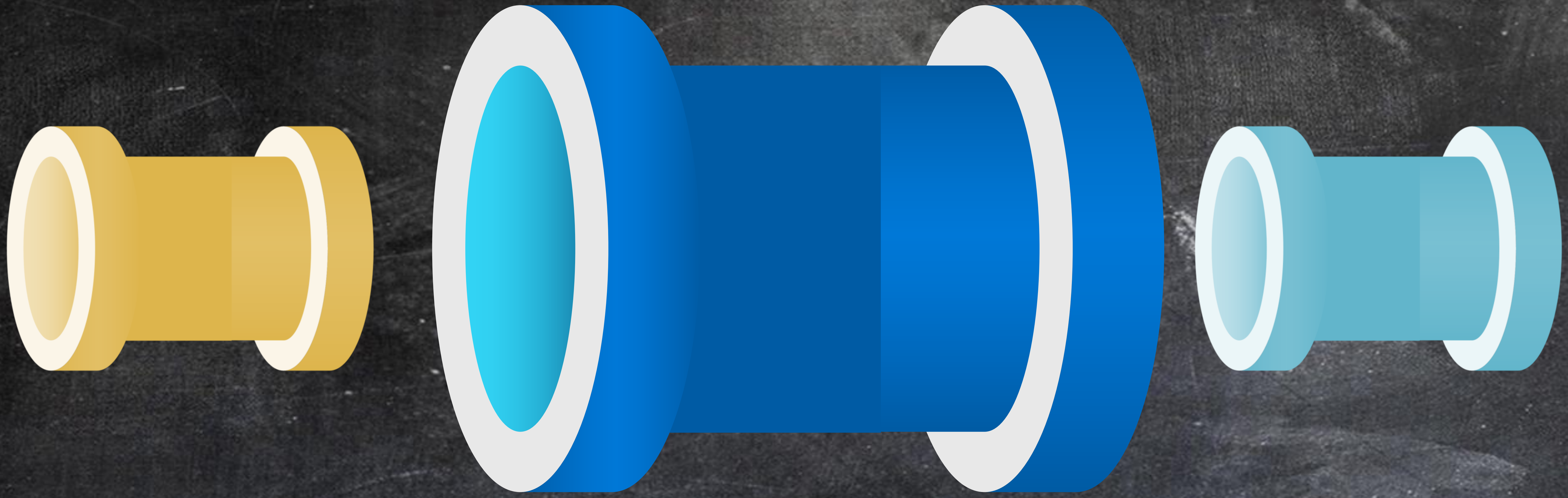
Cloud Formations

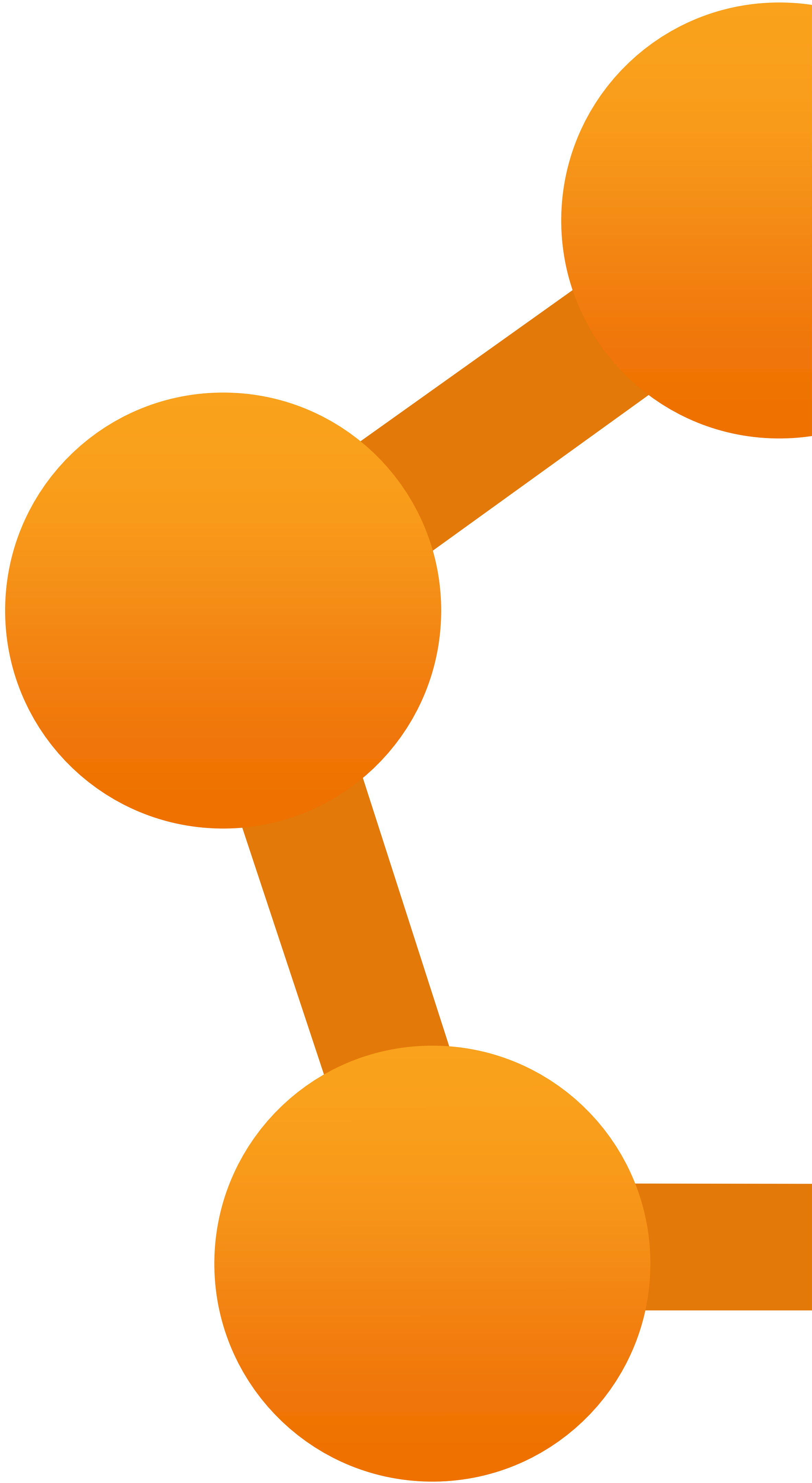
Proudly presents...

Integration Pipelines



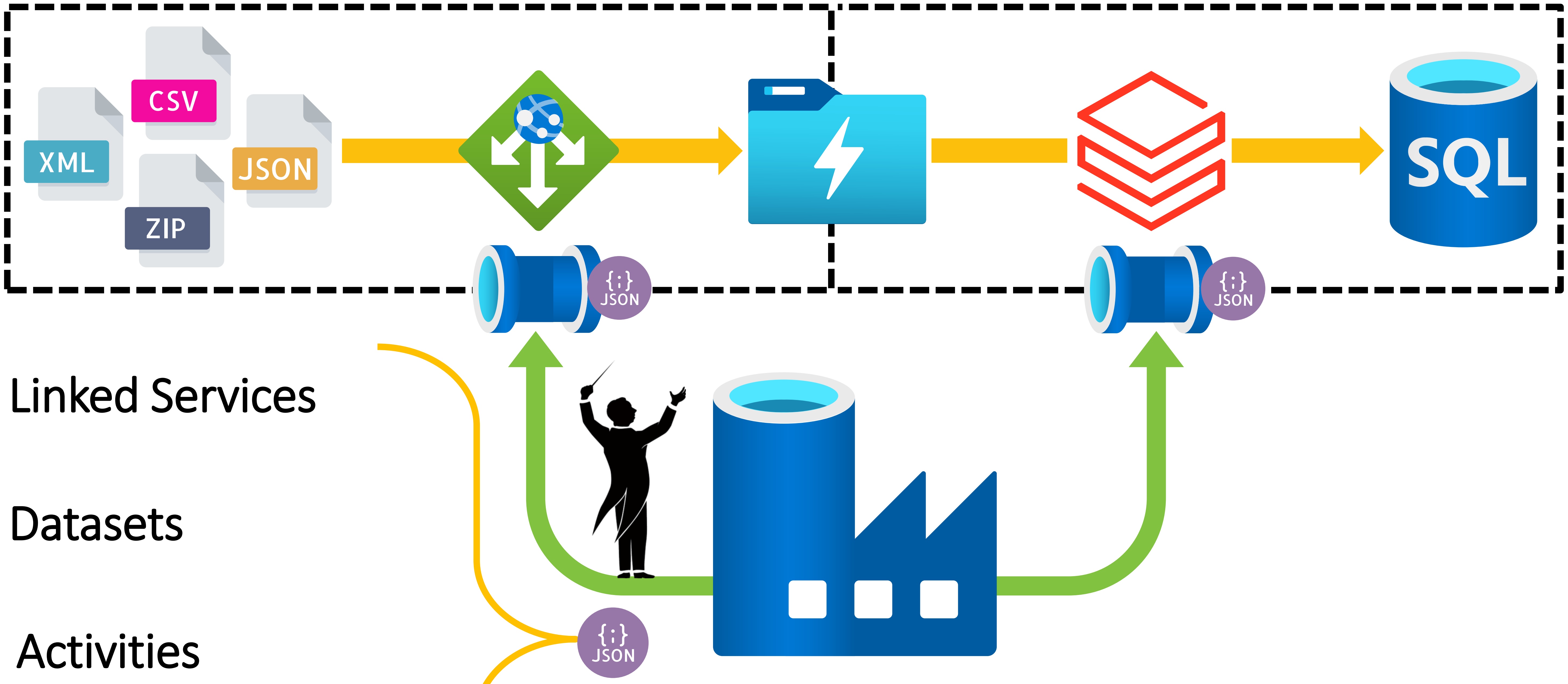
Module 4 – Dynamic Pipelines

Expressions & Interpolation



Cloud Formations

Data Factory Core Components



1 Linked Services

2 Datasets

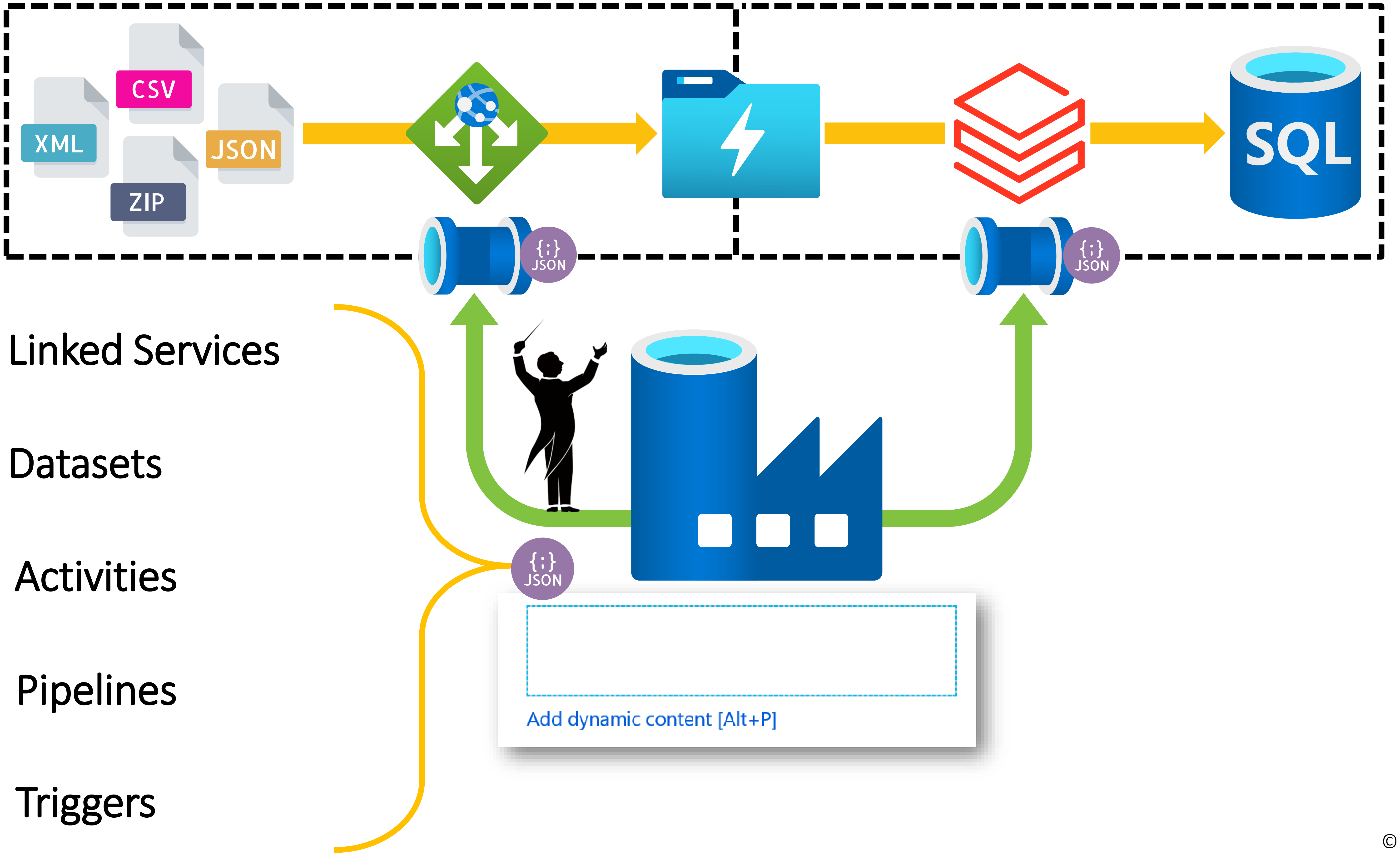
3 Activities

4 Pipelines

5 Triggers

Pipeline Expressions

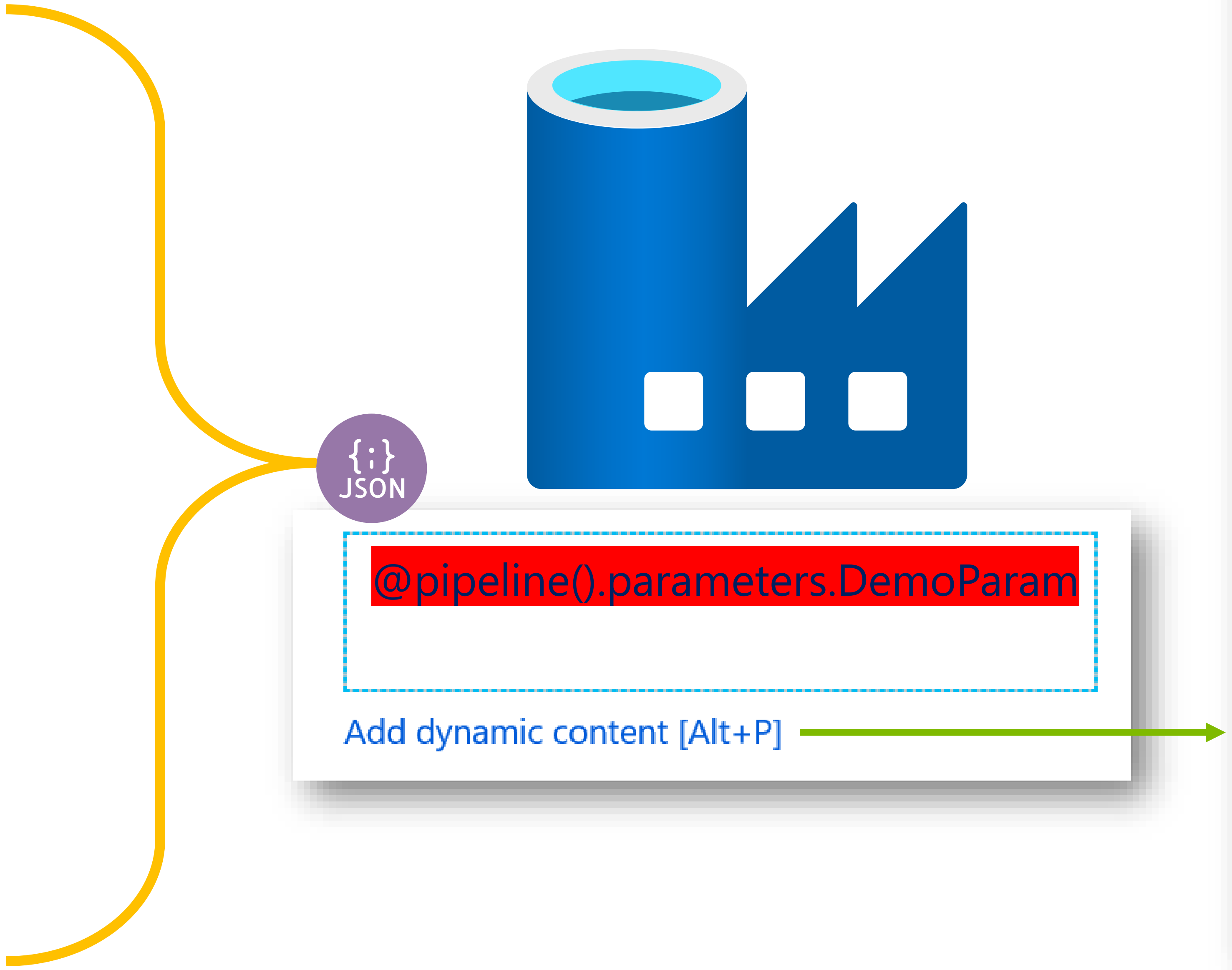
1. Linked Services
2. Datasets
3. Activities
4. Pipelines
5. Triggers



Pipeline Expressions

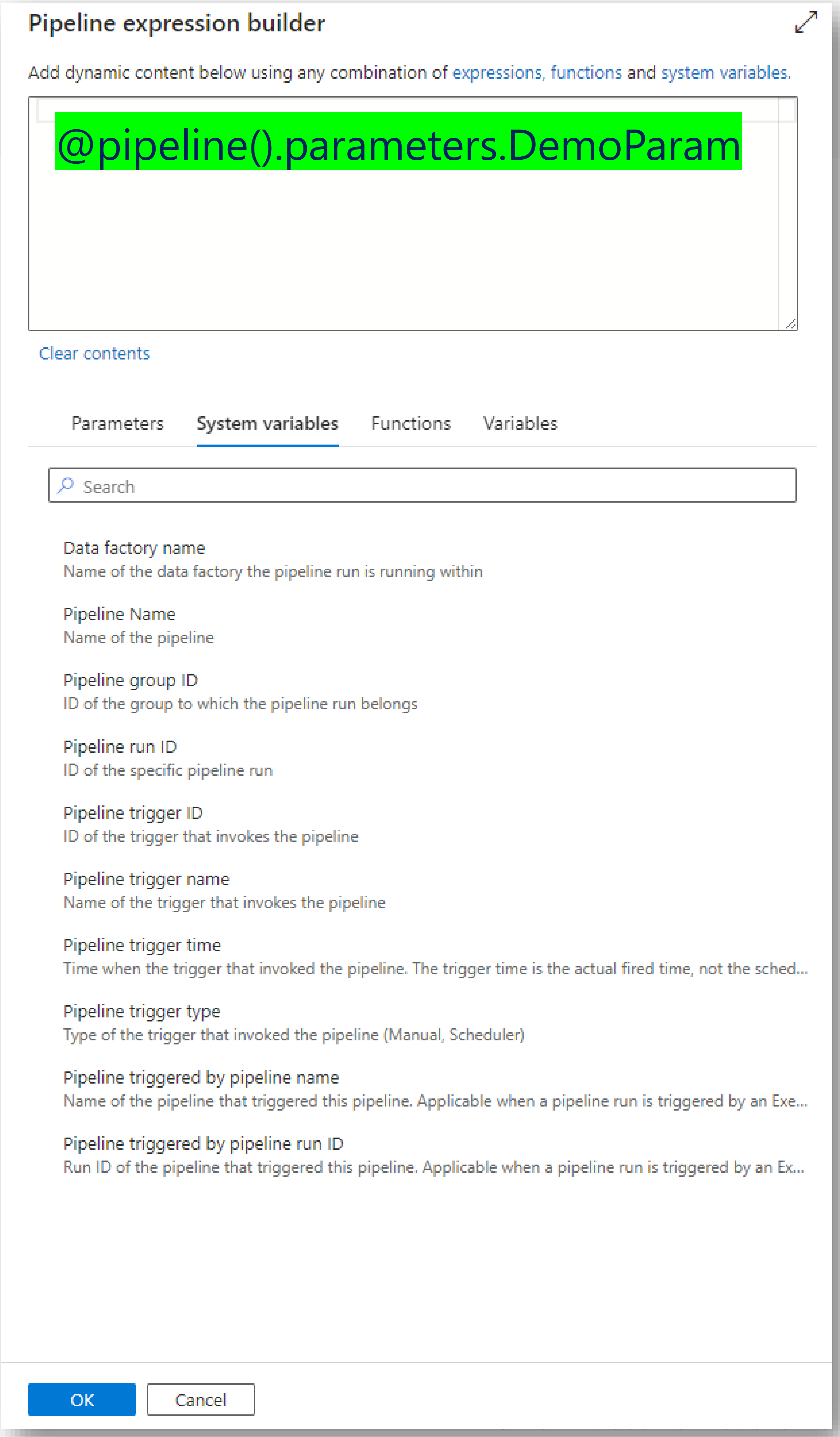
Cloud Formations - Knowledge Transfer & Training

1. Linked Services
2. Datasets
3. Activities
4. Pipelines
5. Triggers



@pipeline().parameters.DemoParam

Add dynamic content [Alt+P]



Pipeline expression builder

Add dynamic content below using any combination of expressions, functions and system variables.

```
@pipeline().parameters.DemoParam
```

Clear contents

Data factory name
Name of the data factory the pipeline run is running within

Pipeline Name
Name of the pipeline

Pipeline group ID
ID of the group to which the pipeline run belongs

Pipeline run ID
ID of the specific pipeline run

Pipeline trigger ID
ID of the trigger that invokes the pipeline

Pipeline trigger name
Name of the trigger that invokes the pipeline

Pipeline trigger time
Time when the trigger that invoked the pipeline. The trigger time is the actual fired time, not the sched...

Pipeline trigger type
Type of the trigger that invoked the pipeline (Manual, Scheduler)

Pipeline triggered by pipeline name
Name of the pipeline that triggered this pipeline. Applicable when a pipeline run is triggered by an Exe...

Pipeline triggered by pipeline run ID
Run ID of the pipeline that triggered this pipeline. Applicable when a pipeline run is triggered by an Ex...

Pipeline Expressions

Cloud Formations - Knowledge Transfer & Training

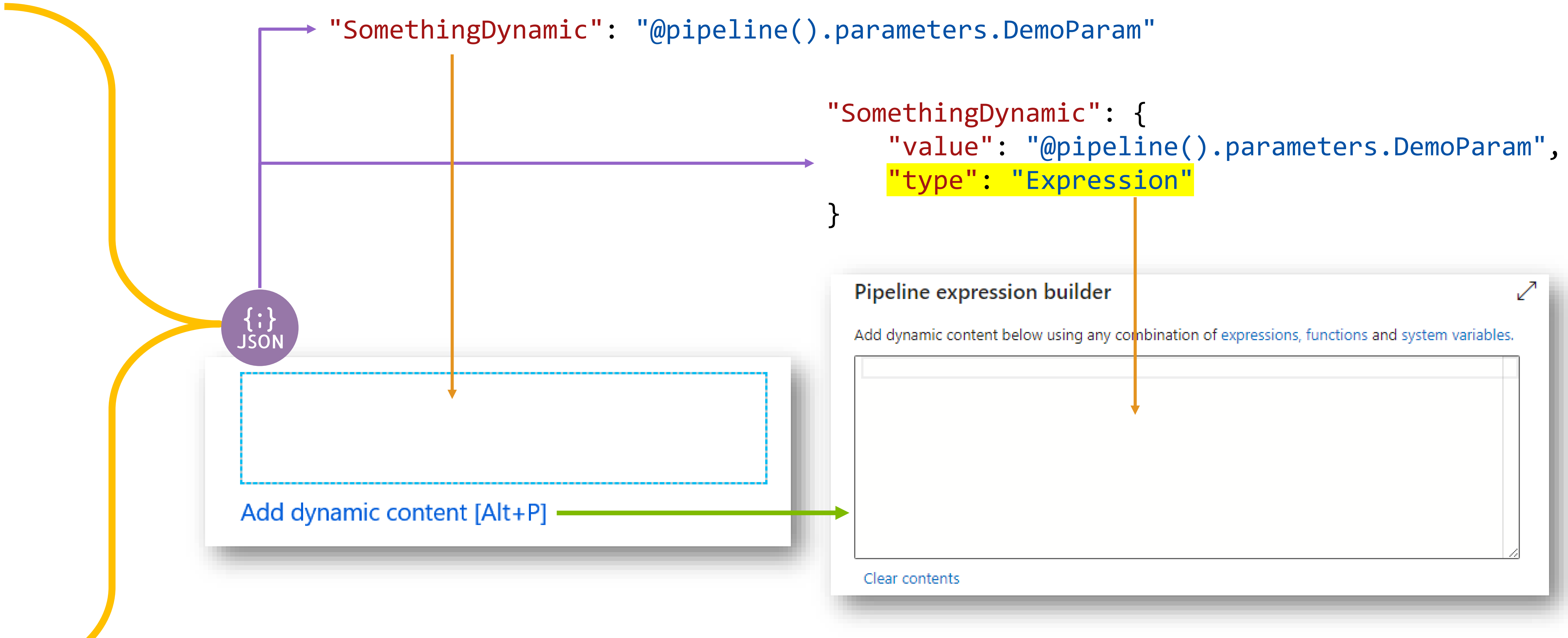
1 Linked Services

2 Datasets

3 Activities

4 Pipelines

5 Triggers



```
"SomethingDynamic": "@pipeline().parameters.DemoParam"
```

```
"SomethingDynamic": {
    "value": "@pipeline().parameters.DemoParam",
    "type": "Expression"
}
```

Add dynamic content [Alt+P]

Pipeline expression builder

Add dynamic content below using any combination of expressions, functions and system variables.

Clear contents

String Interpolation

String interpolation is a technique used in programming to insert the value of variables or expressions into a string. This process involves evaluating a string literal containing placeholders and replacing those placeholders with their corresponding values. It's a more readable and convenient alternative to building dedicated expressions.

```
"parameters": {
   "DemoParam": {
   "type": "string",
   "defaultValue": "World"
}
```

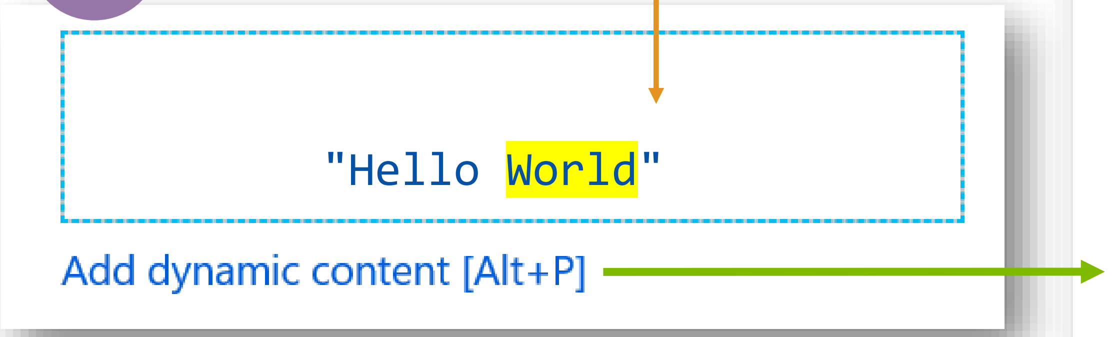
```
"SomethingDynamic": "Hello @{pipeline().parameters.DemoParam}"
```

```
"SomethingDynamic": "@pipeline().parameters.DemoParam"
```

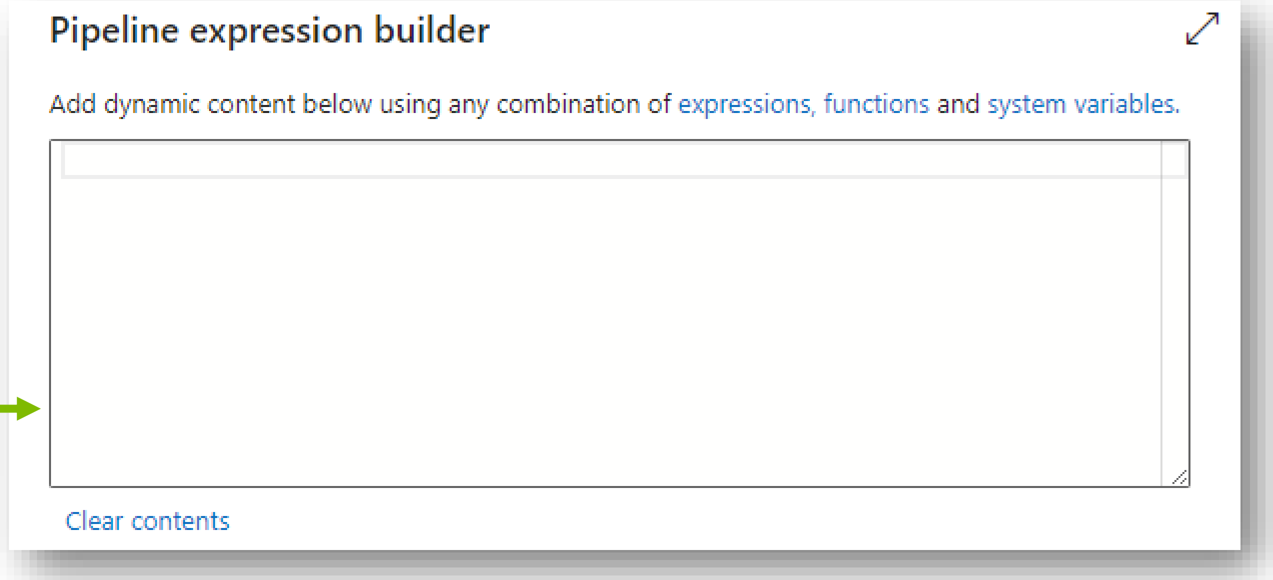
```
"SomethingDynamic": {
    "value": "@pipeline().parameters.DemoParam",
    "type": "Expression"
}
```

1. Linked Services
2. Datasets
3. Activities
4. Pipelines
5. Triggers

{:}
JSON



"Hello World"



Pipeline expression builder

Add dynamic content below using any combination of expressions, functions and system variables.

Cloud Formations - Knowledge Transfer & Training

Expressions Use

String interpolation is a technique used in programming to insert the value of variables or expressions into a string. This process involves evaluating a string literal containing placeholders and replacing those placeholders with their corresponding values. It's a more readable and convenient alternative to building dedicated expressions.

```
"parameters": {
   "DemoParam": {
   "type": "string",
   "defaultValue": "World"
}
```

```
"SomethingDynamic": "Hello @{pipeline().parameters.DemoParam}"
```

✓

```
"SomethingDynamic": "@pipeline().parameters.DemoParam"
```

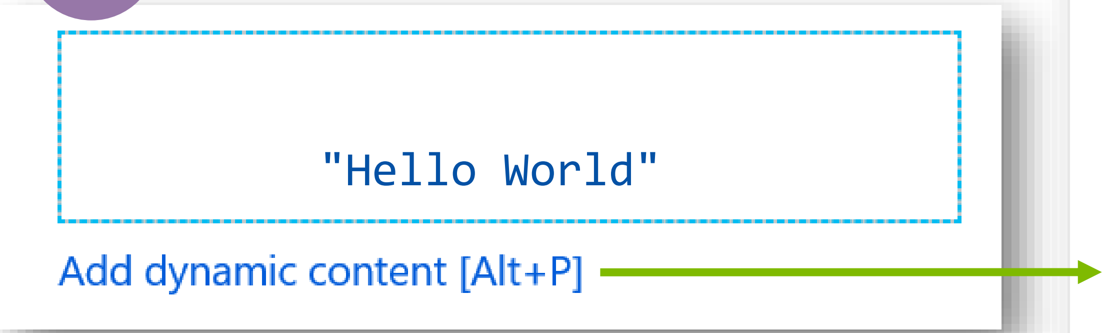
✗

```
"SomethingDynamic": {
    "value": "@pipeline().parameters.DemoParam",
    "type": "Expression"
}
```

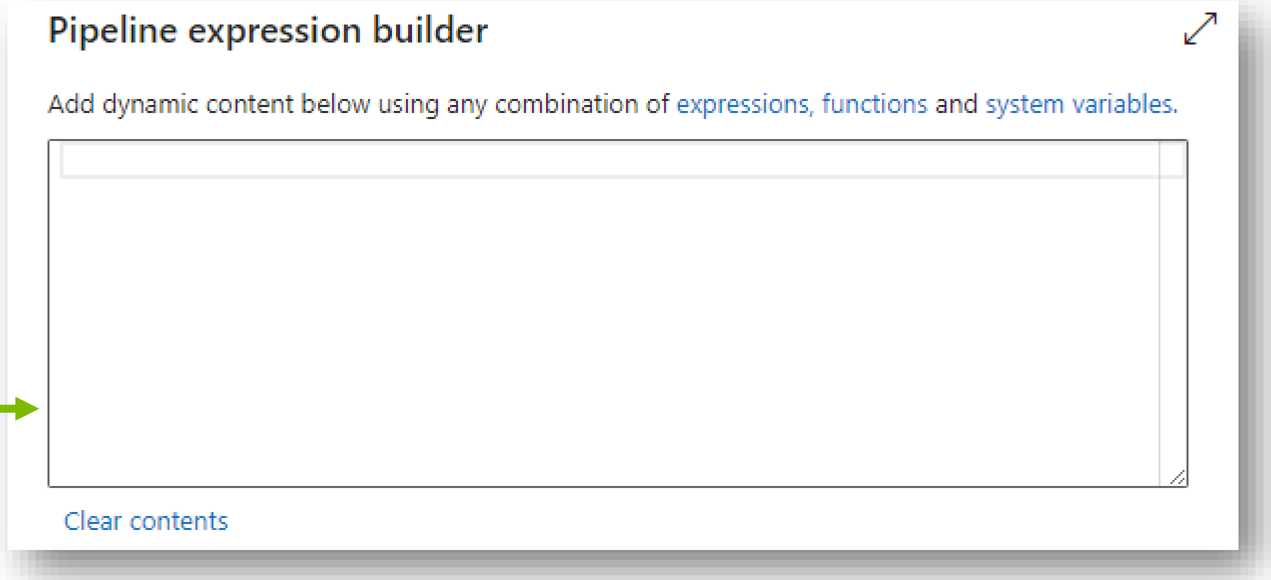
✓

1. Linked Services
2. Datasets
3. Activities
4. Pipelines
5. Triggers

{:} JSON



"Hello World"



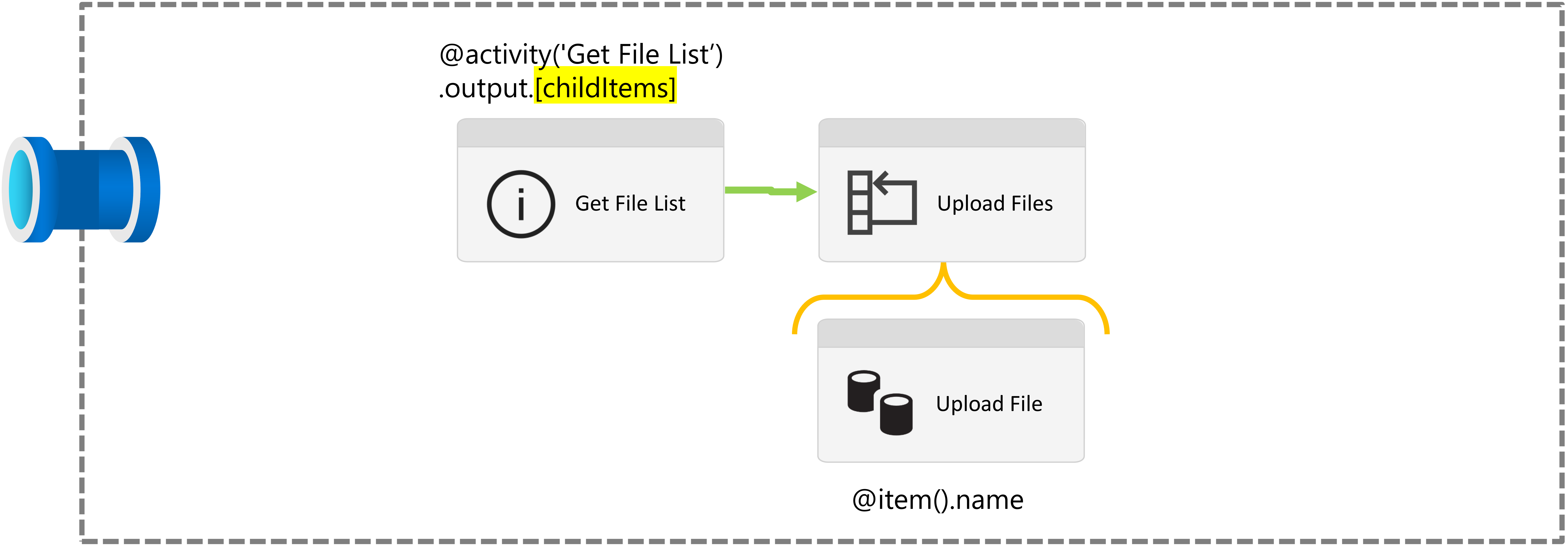
Pipeline expression builder
Add dynamic content below using any combination of expressions, functions and system variables.
Clear contents

Module 4 – Dynamic Pipelines

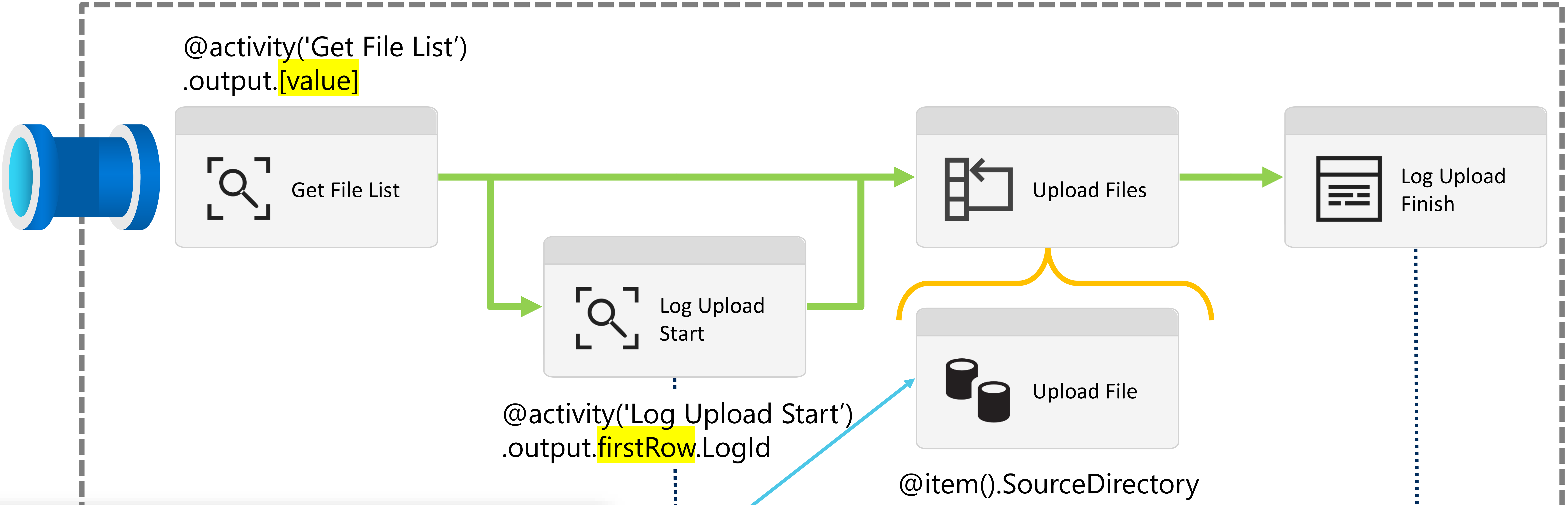
Simple Metadata Driven Execution

Cloud Formations

Data Discovery and Upload



Simple Metadata and Upload



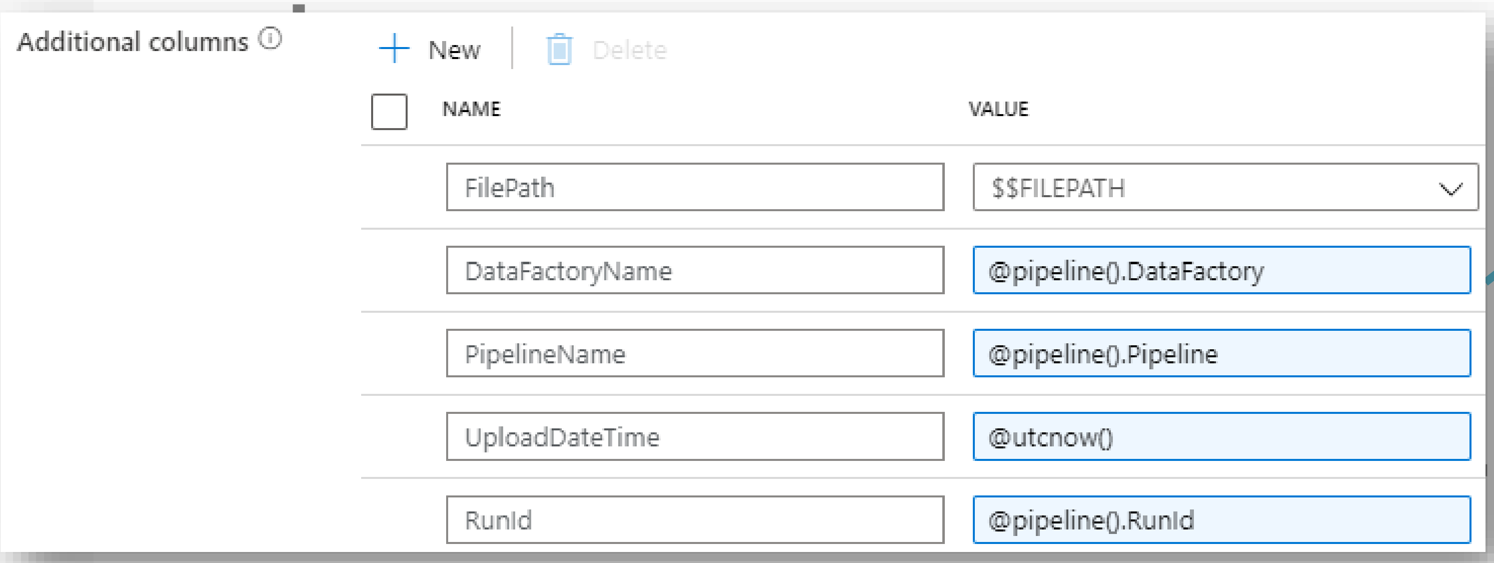
@activity('Get File List')
.output.[value]

@activity('Log Upload Start')
.output.firstRow.LogId

@item().SourceDirectory

@item().TargetDirectory

@item().FileName

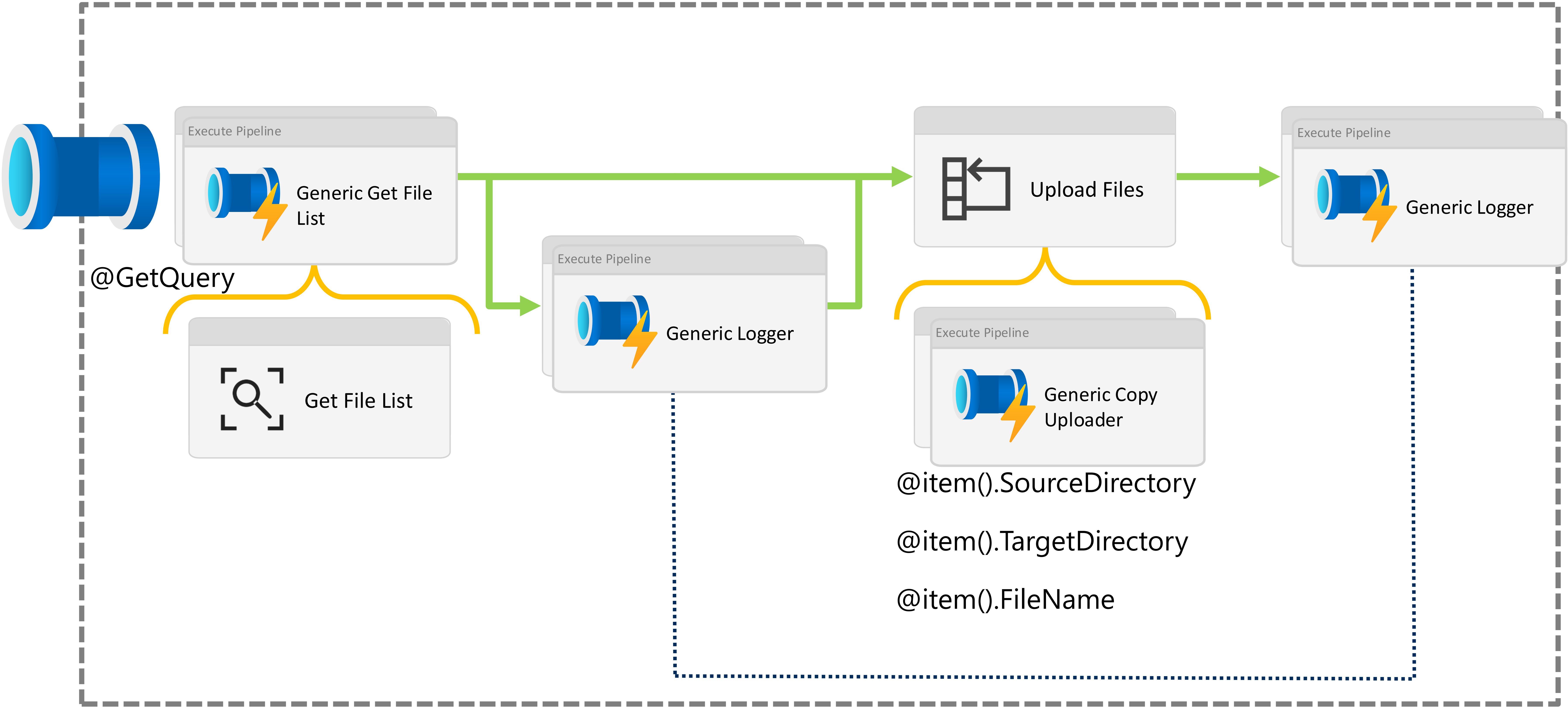


Additional columns ⓘ

+ New | Delete

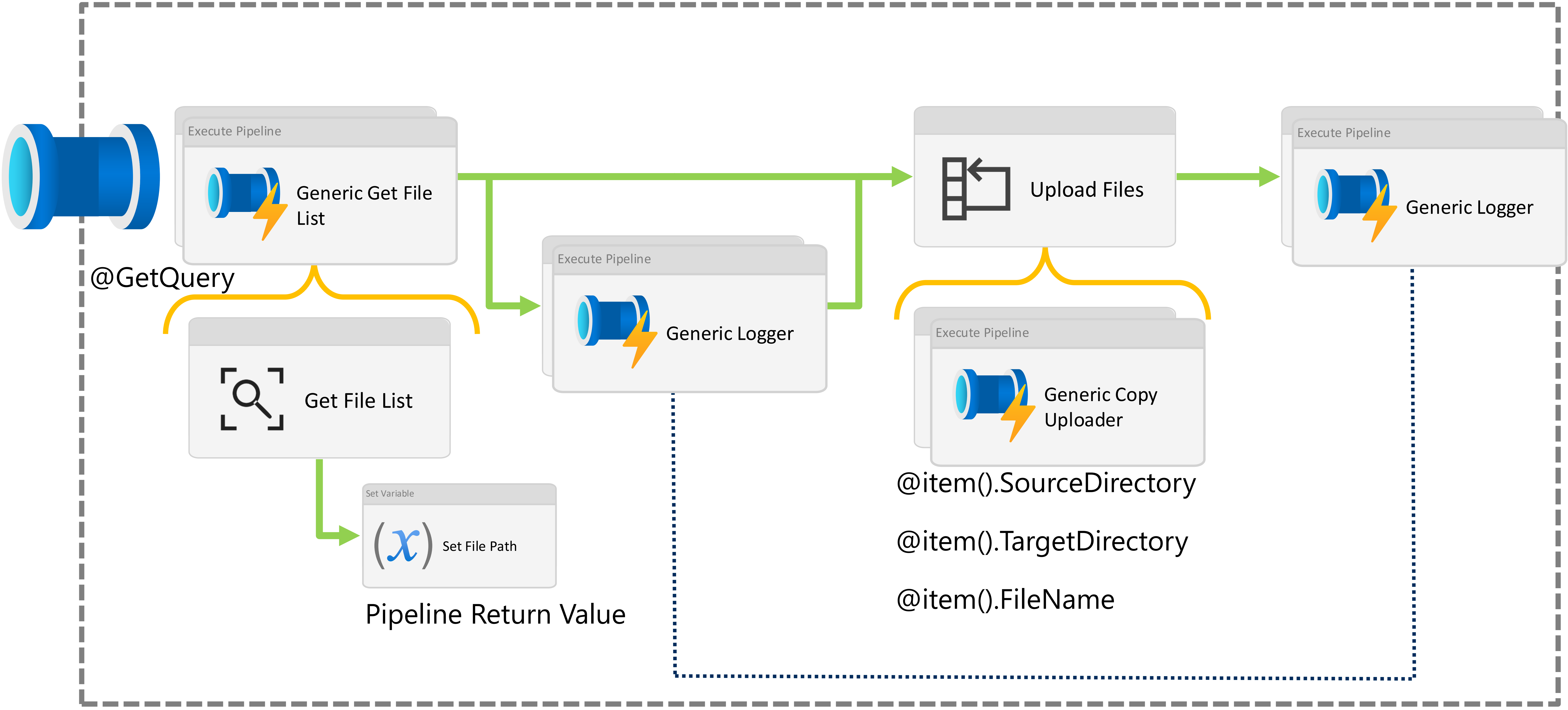
□	NAME	VALUE
	FilePath	$$FILEPATH
	DataFactoryName	@pipeline().DataFactory
	PipelineName	@pipeline().Pipeline
	UploadDateTime	@utcnow()
	RunId	@pipeline().RunId

Using Pipelines as Worker Components



Cloud Formations - Knowledge Transfer & Training

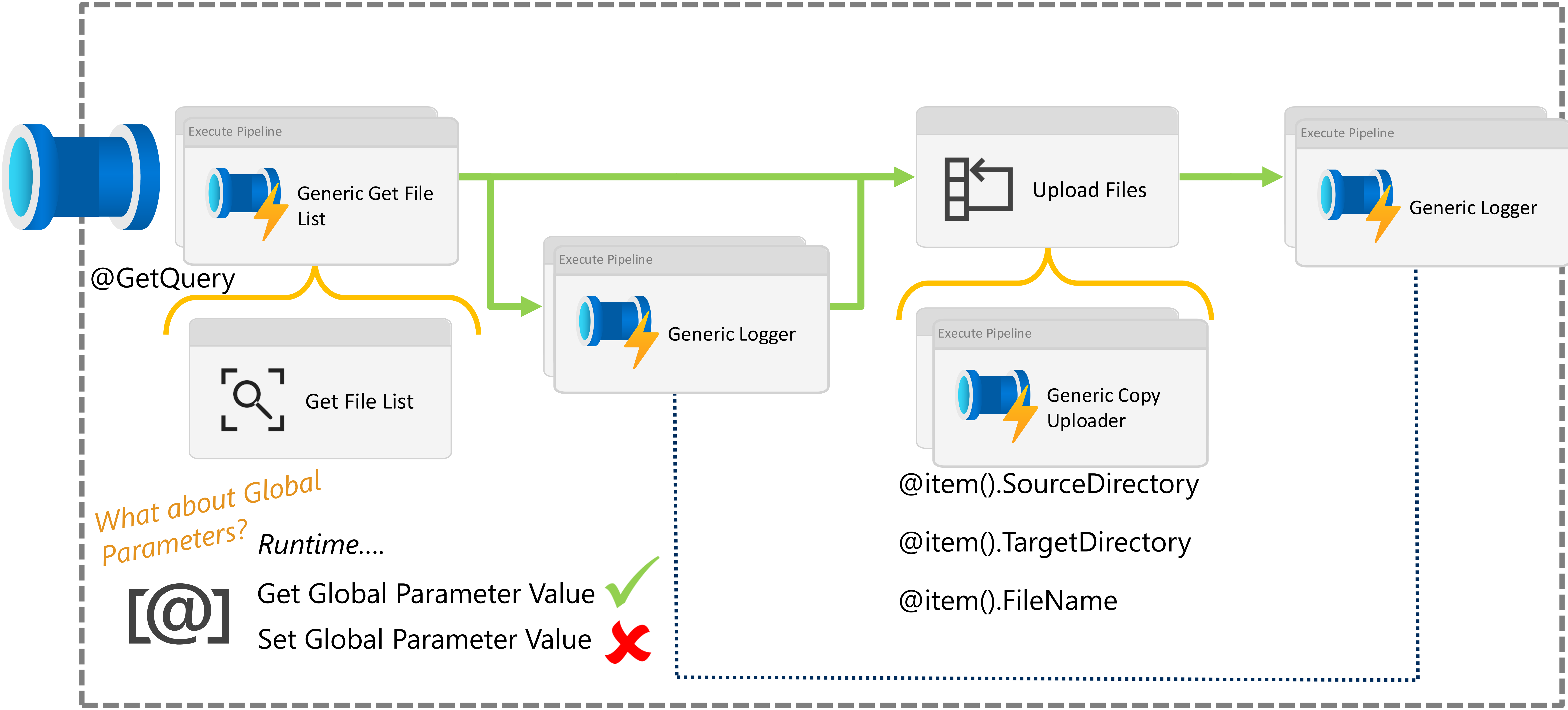
Using Pipelines as Worker Components



Cloud Formations - Knowledge Transfer & Training

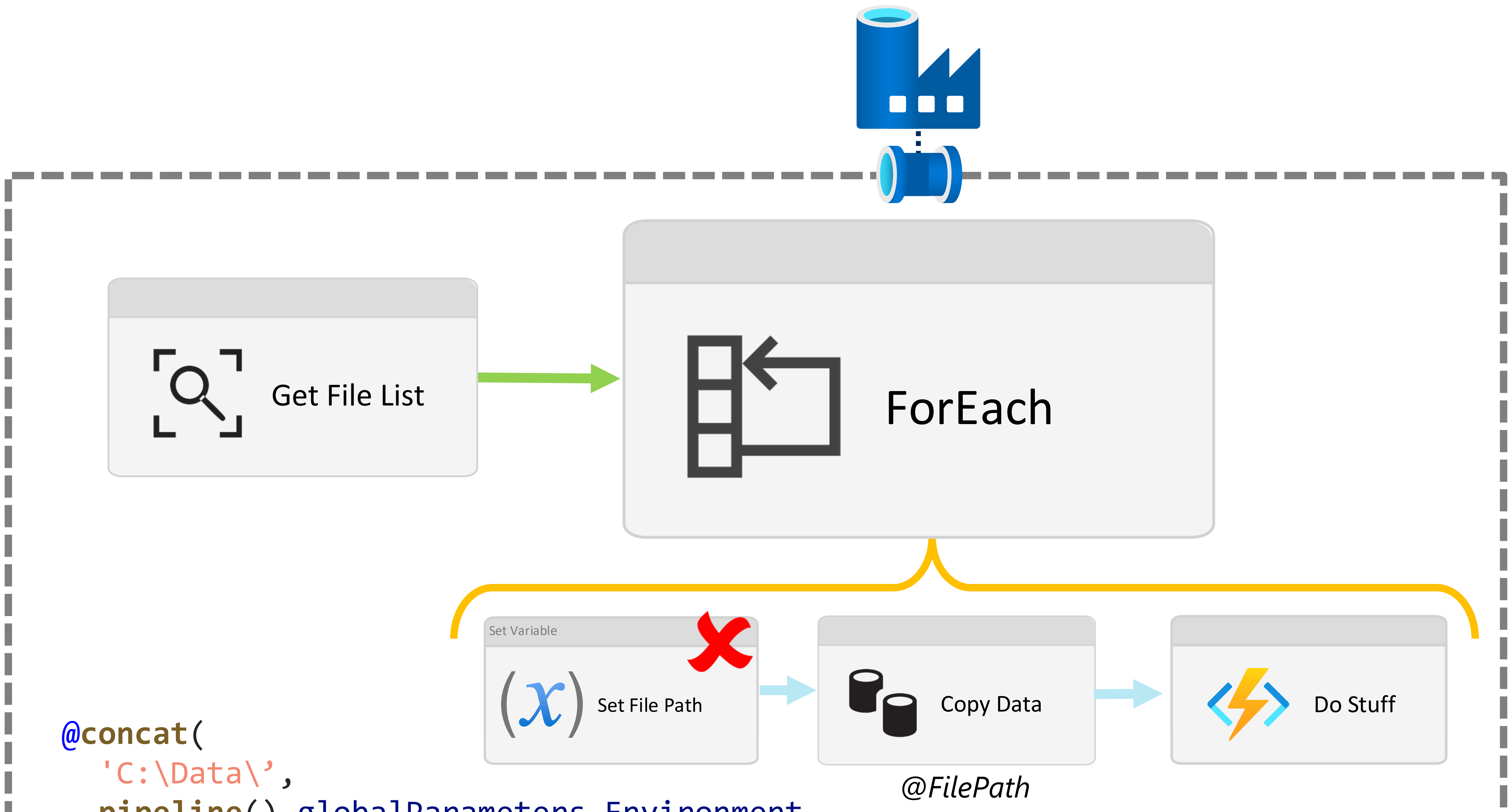
Using Pipelines as Worker Components

With a Global Parameters

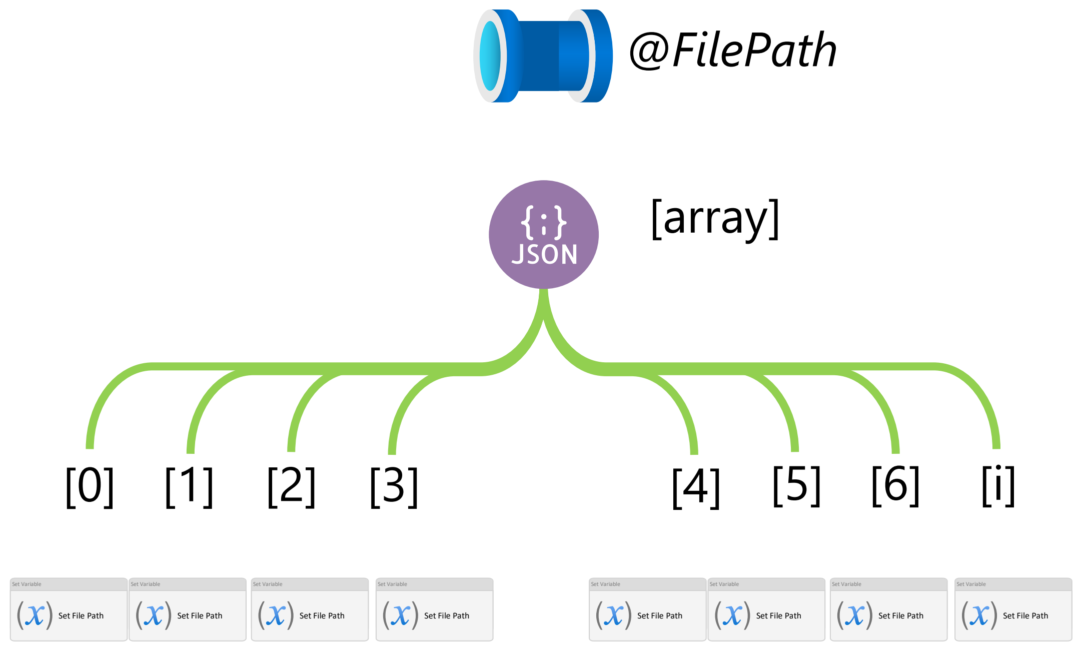


Variable Scope

Thread Safe



```
@concat(
   'C:\Data\’,
   pipeline().globalParameters.Environment,
   '\’,
   item().FileName,
   '.csv’)
```

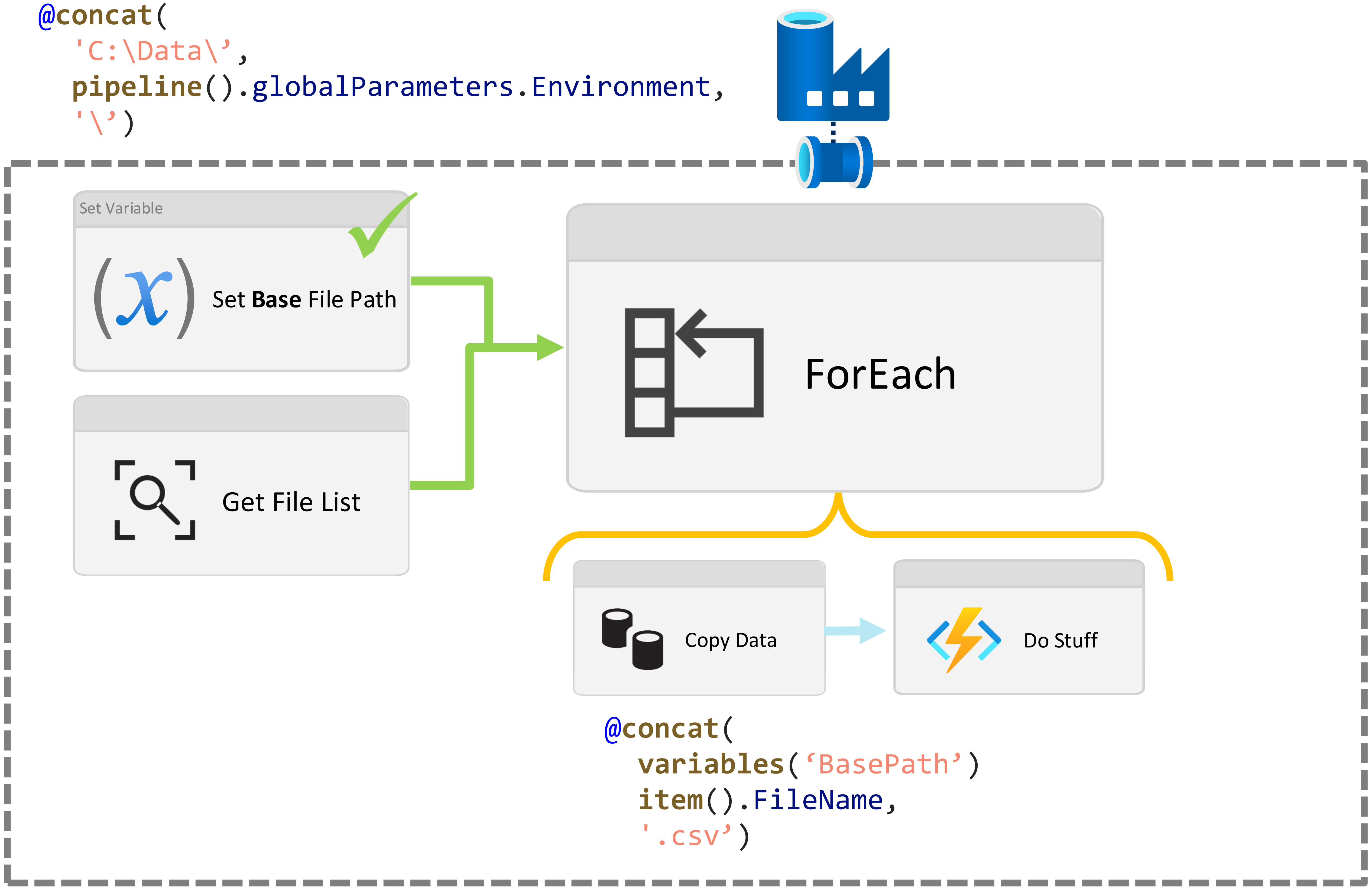


Cloud Formations - Knowledge Transfer & Training

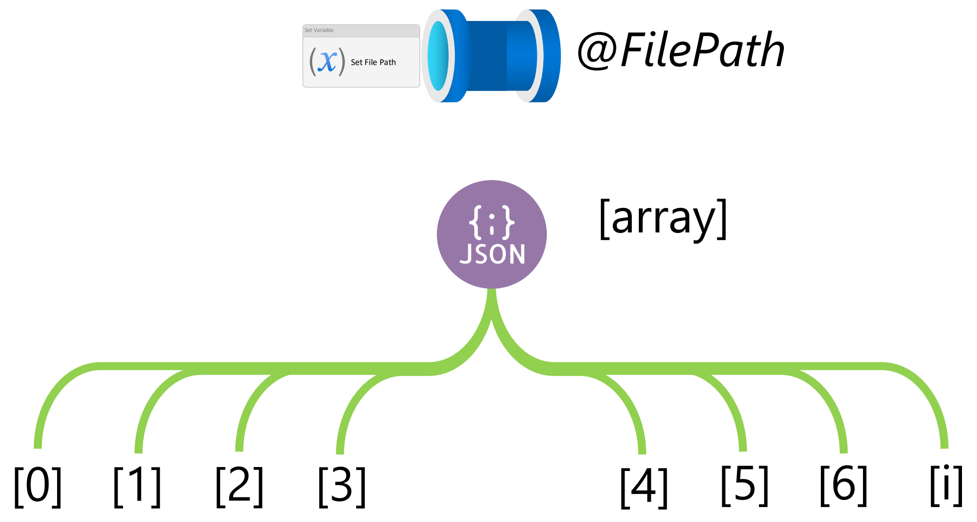
Variable Scope

Layering Expressions

```
@concat(
  'C:\Data\’,
  pipeline().globalParameters.Environment,
  '\’)
```

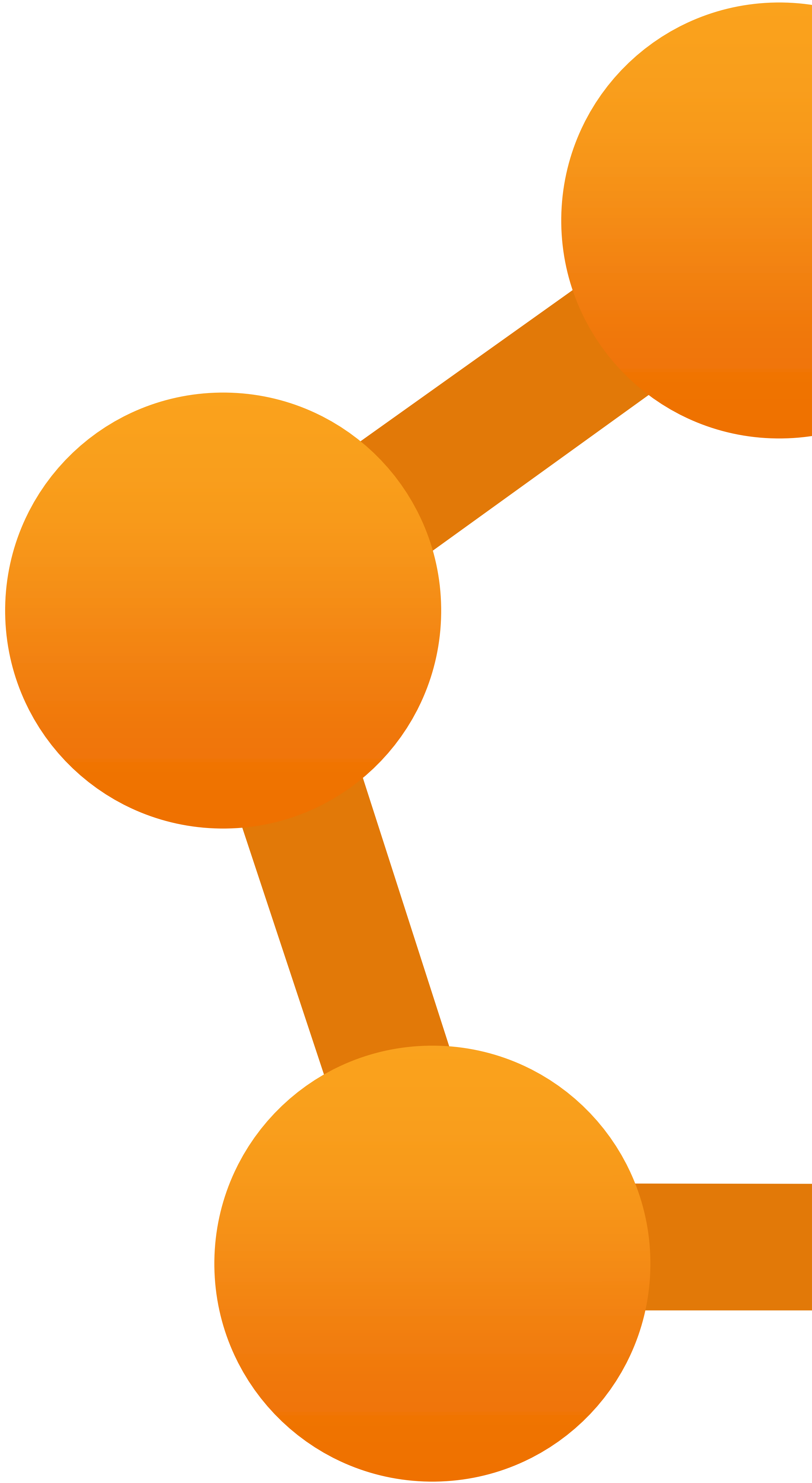


```
@concat(
  variables(‘BasePath’)
  item().FileName,
  '.csv’)
```



Module 4 – Dynamic Pipelines

Dynamic Content Chains

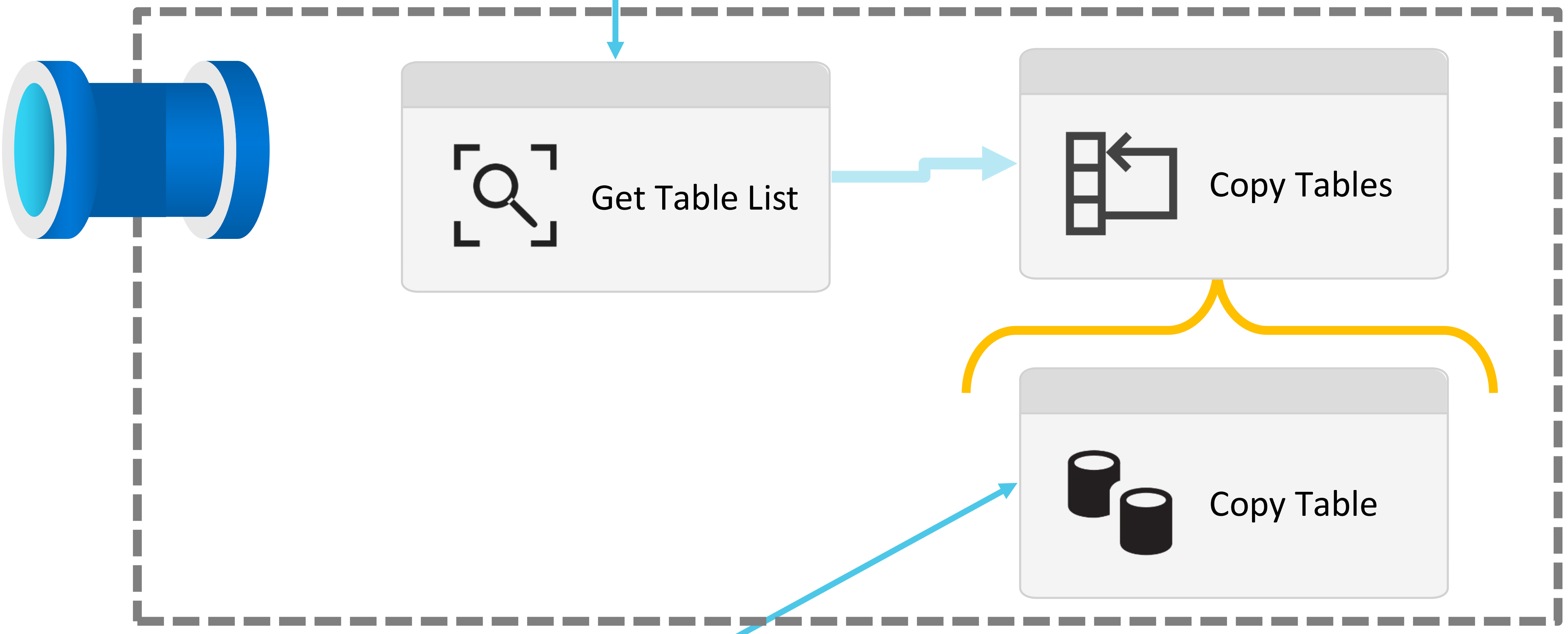


Cloud Formations

Lazy SQLDB Replication

```
SELECT s.name AS SchemaName, o.name AS TableName FROM sys.objects o
INNER JOIN sys.schemas s ON o.schema_id = s.schema_id WHERE o.[type] = 'U'
```

@pipeline().parameters.TableLookupQuery



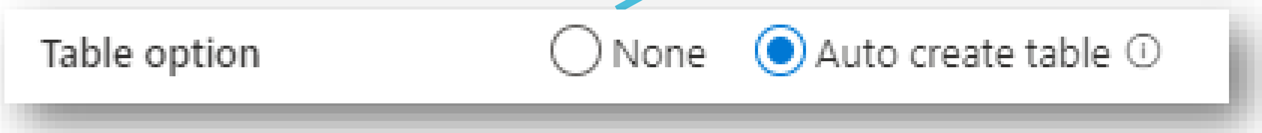


Table option ○ None ● Auto create table ⓘ

```
IF OBJECT_ID(
                '@{item().SchemaName}.@{item().TableName}’
                ) IS NOT NULL

TRUNCATE TABLE @{item().SchemaName}.@{item().TableName}
```

1x

1x

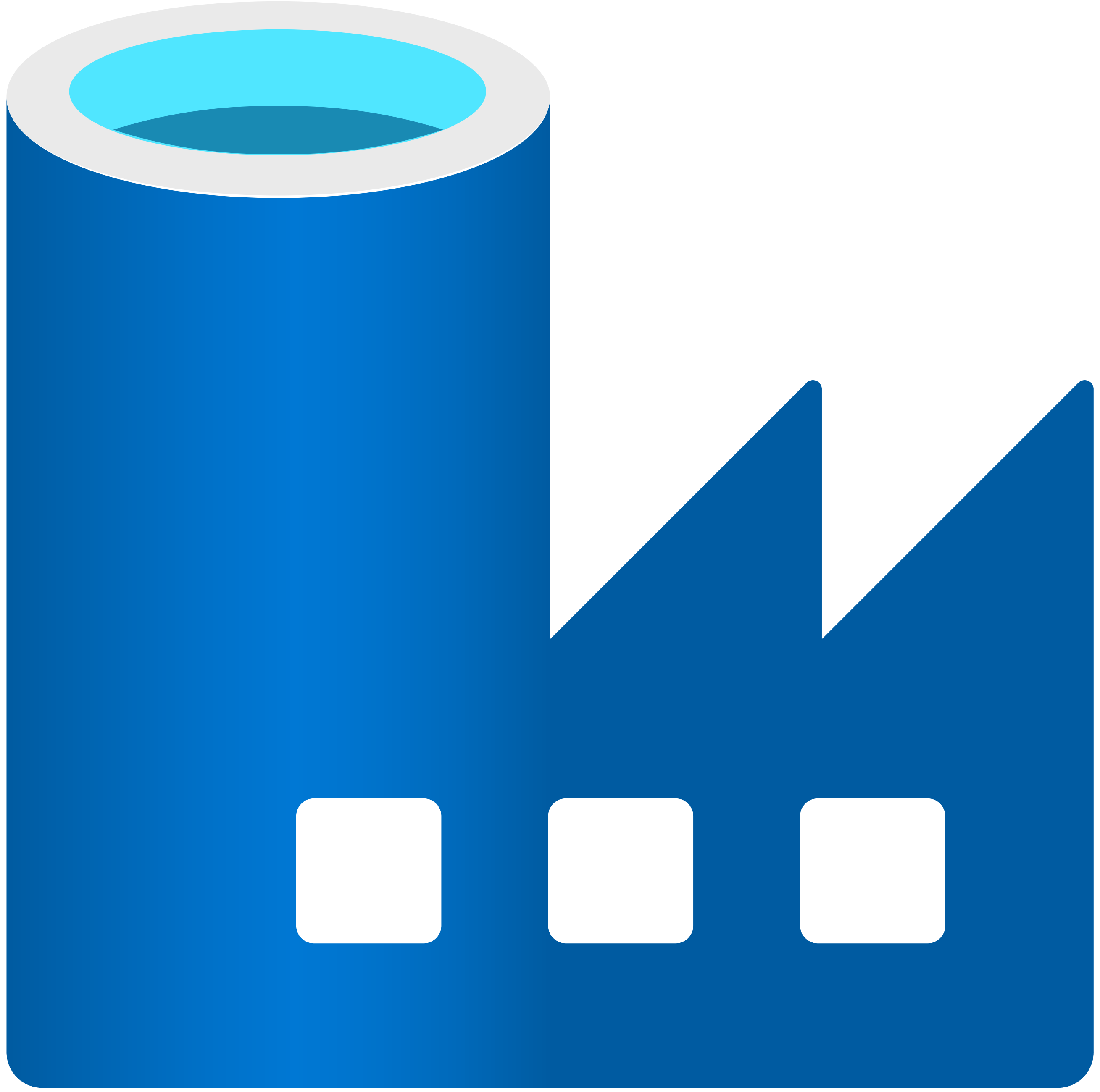
@pipeline().parameters.SourceConnectionSecret

@pipeline().parameters.TargetConnectionSecret

@dataset().LinkedServiceConnectionSecret

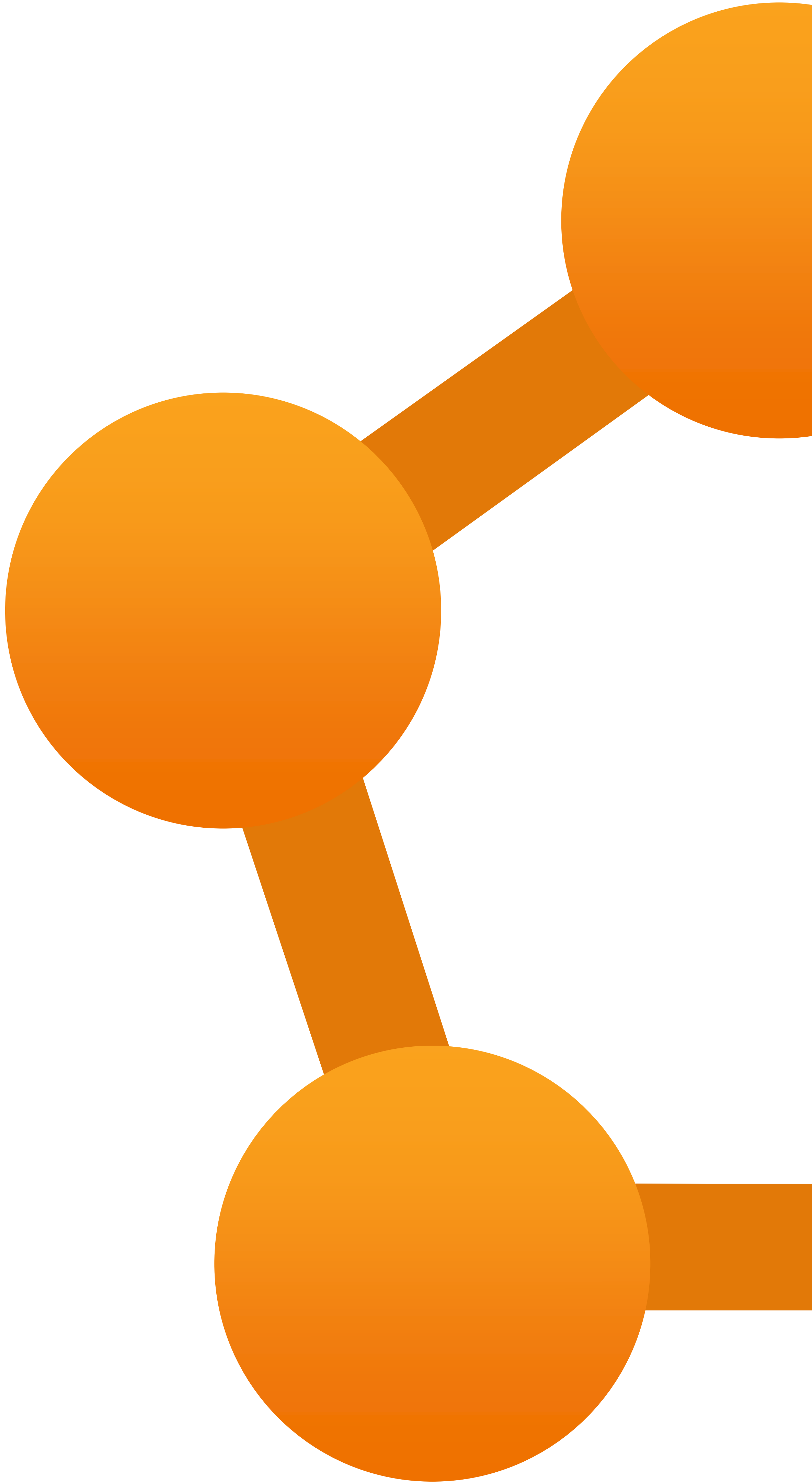
@linkedService().DBConnectionSecret

Azure Portal



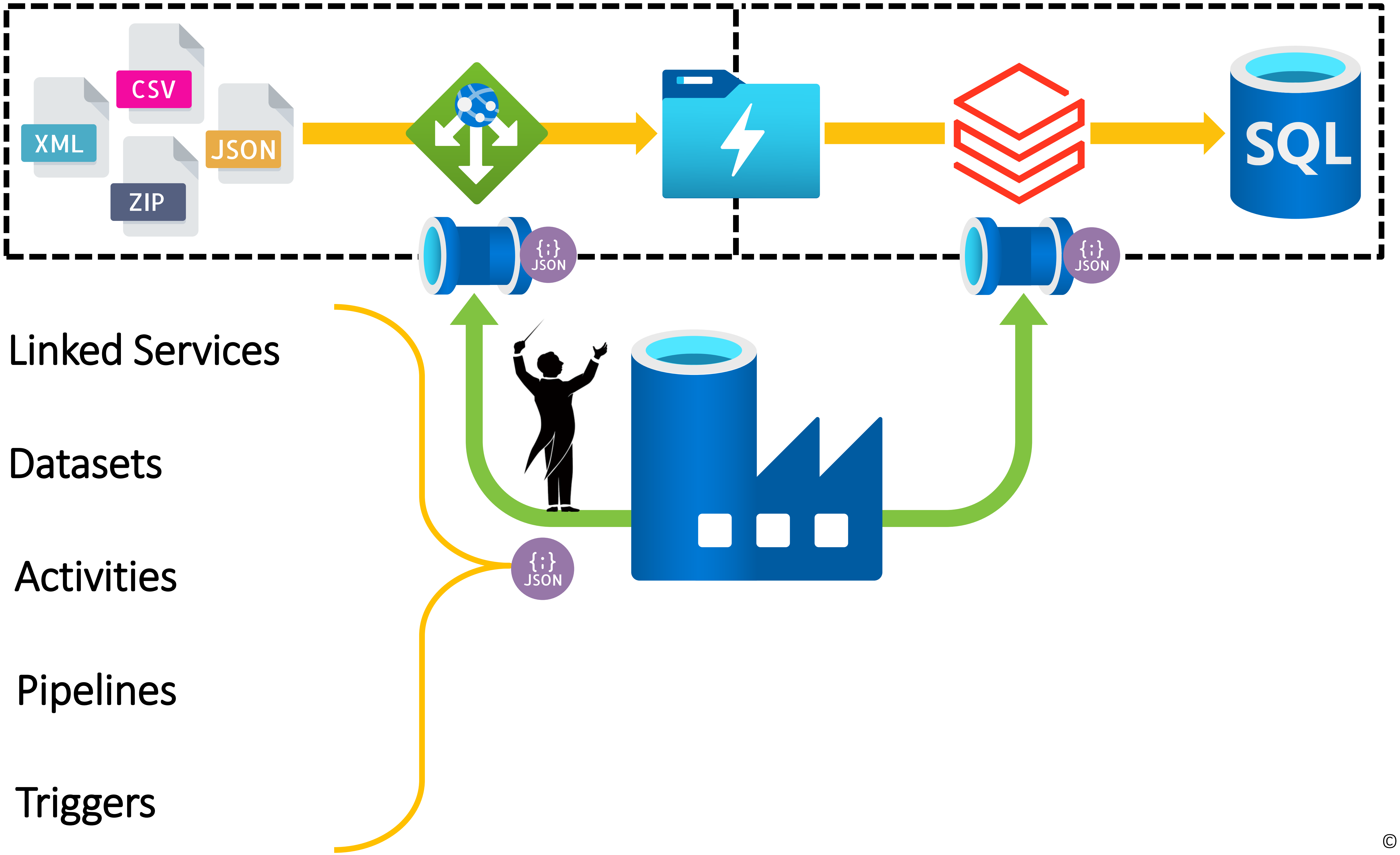
Module 4 – Dynamic Pipelines

Reference Names



Cloud Formations

Reference Names

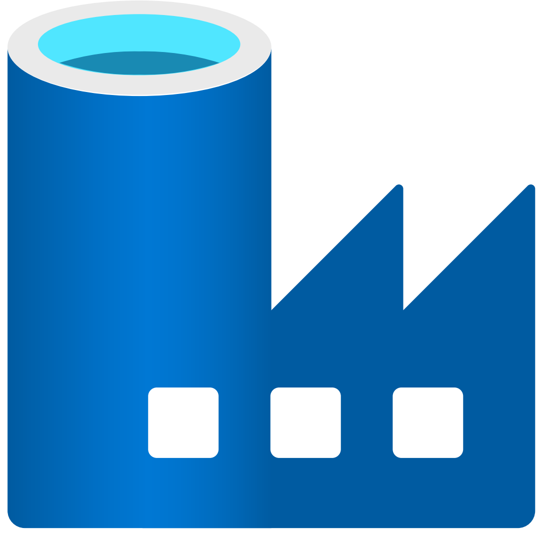


1. Linked Services
2. Datasets
3. Activities
4. Pipelines
5. Triggers

Cloud Formations - Knowledge Transfer & Training

Reference Names

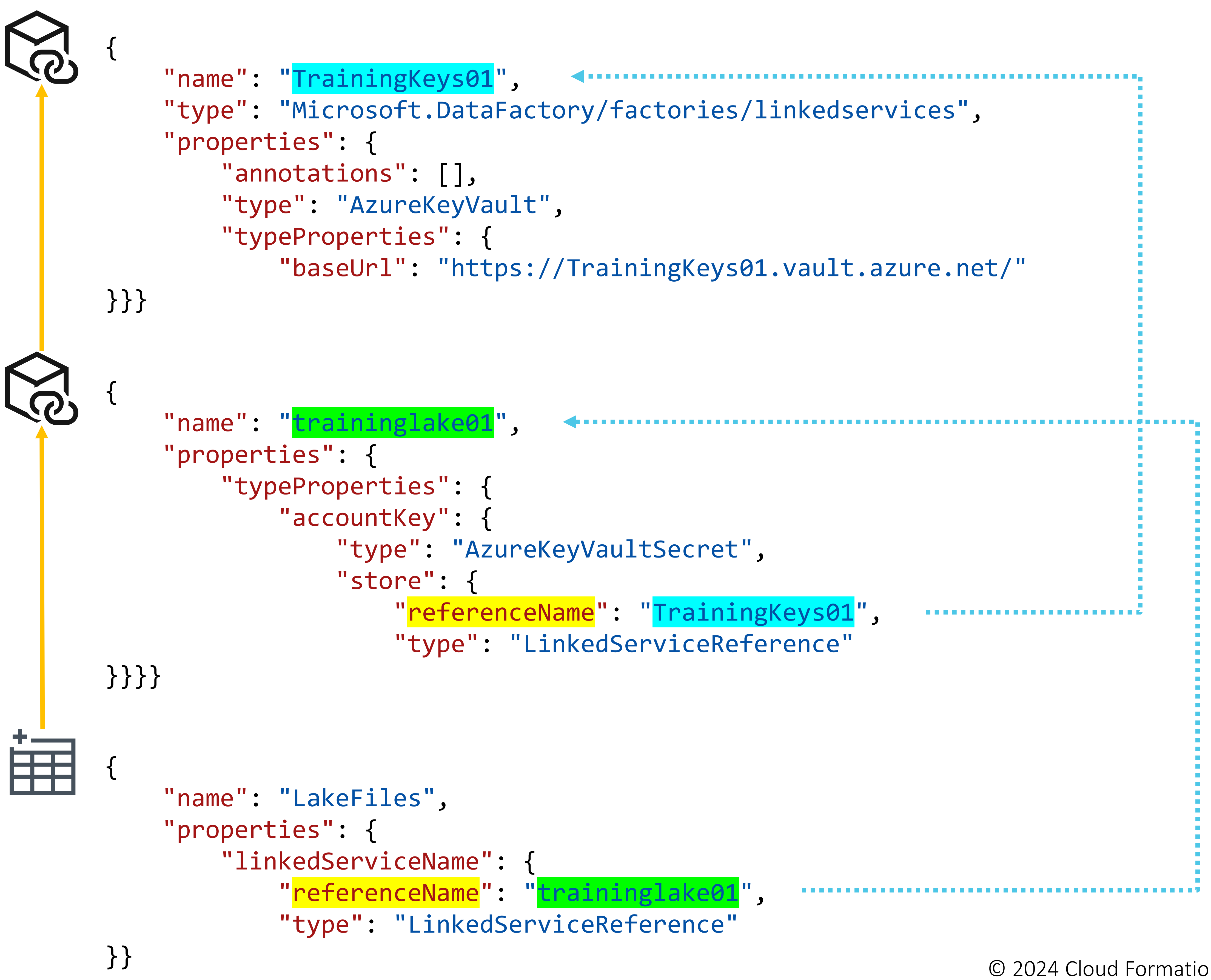
1. Linked Services
2. Datasets
3. Activities
4. Pipelines
5. Triggers



Reference Names

Cloud Formations - Knowledge Transfer & Training

1. Linked Services
2. Datasets
3. Activities
4. Pipelines
5. Triggers



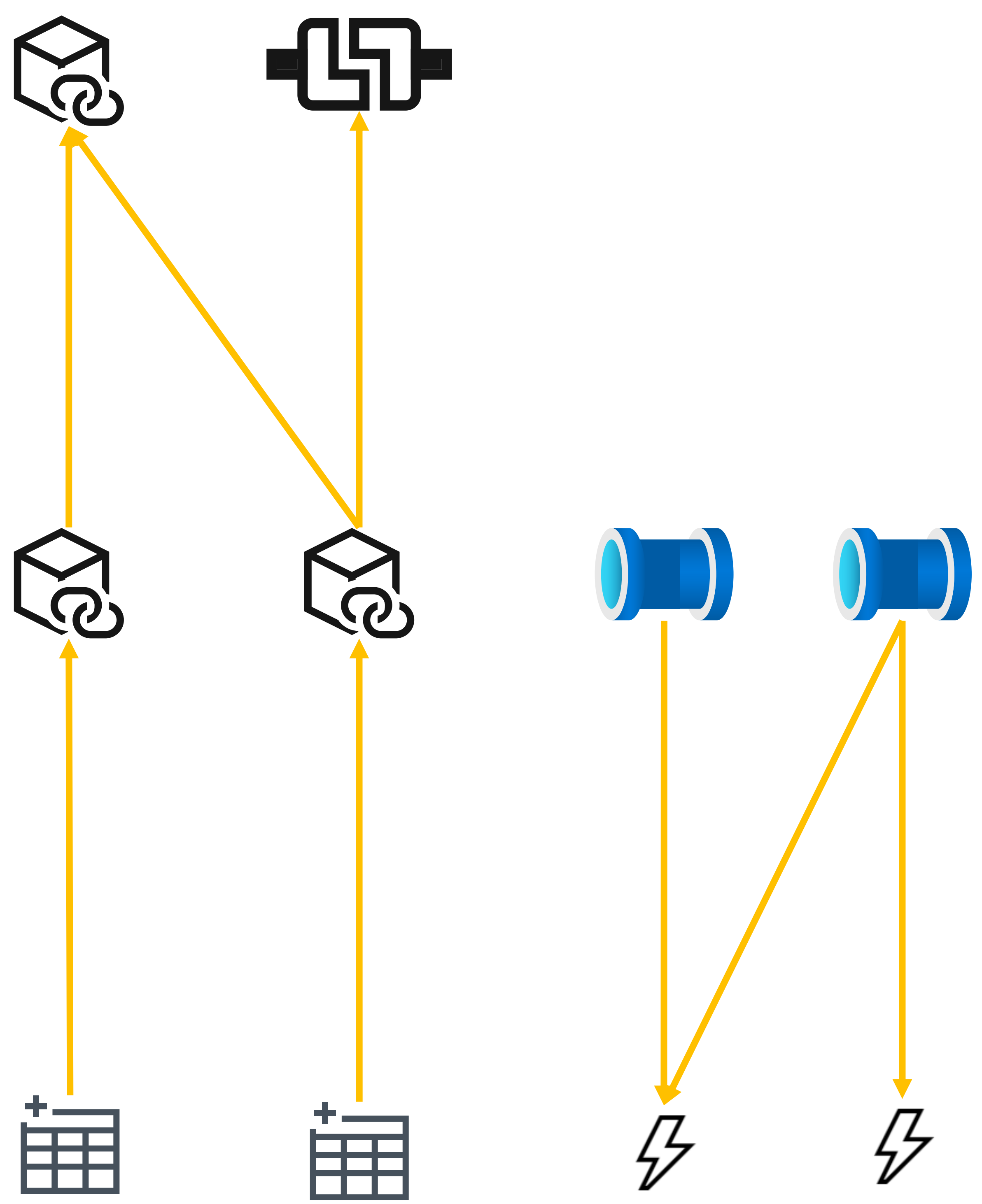
```
{
    "name": "TrainingKeys01",
    "type": "Microsoft.DataFactory/factories/linkedservices",
    "properties": {
        "annotations": [],
        "type": "AzureKeyVault",
        "typeProperties": {
            "baseUrl": "https://TrainingKeys01.vault.azure.net/"
}}}
```

```
{
    "name": "traininglake01",
    "properties": {
        "typeProperties": {
            "accountKey": {
                "type": "AzureKeyVaultSecret",
                "store": {
                    "referenceName": "TrainingKeys01",
                    "type": "LinkedServiceReference"
}}}}
```

```
{
    "name": "LakeFiles",
    "properties": {
        "linkedServiceName": {
            "referenceName": "traininglake01",
            "type": "LinkedServiceReference"
}}
```

Reference Names

1. Linked Services
2. Datasets
3. Activities
4. Pipelines
5. Triggers



Reference Names cannot be dynamic.

- Not at development time.
- Not at runtime.
- At deployment time if being careful.

They are used internally by Data Factory to validate artifact dependencies.

Module 4

Dynamic Pipelines ✓

Any questions?

Cloud Formations