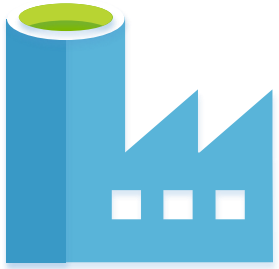
Create a Metadata Driven Processing Framework

For Azure Data Factory



Paul Andrew | Principal Consultant & Solution Architect

Microsoft® Most Valuable Professional

altius

@MrPaulAndrew In/MrPaulAndrew MrPaulAndrew.com

https://github.com/mrpaulandrew

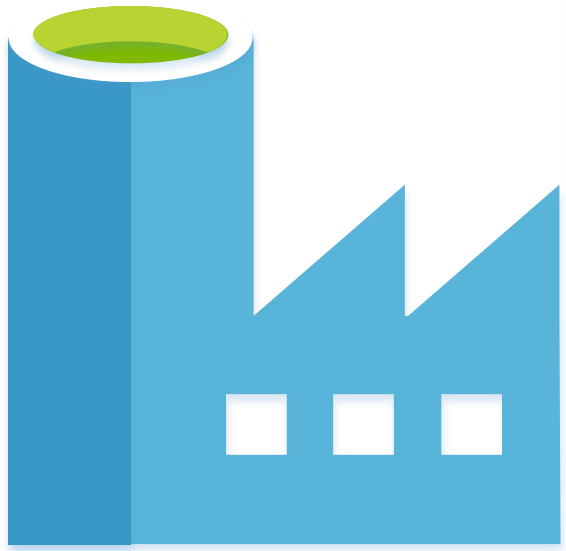
CommunityEvents

Demo code, content and slides from various community events.

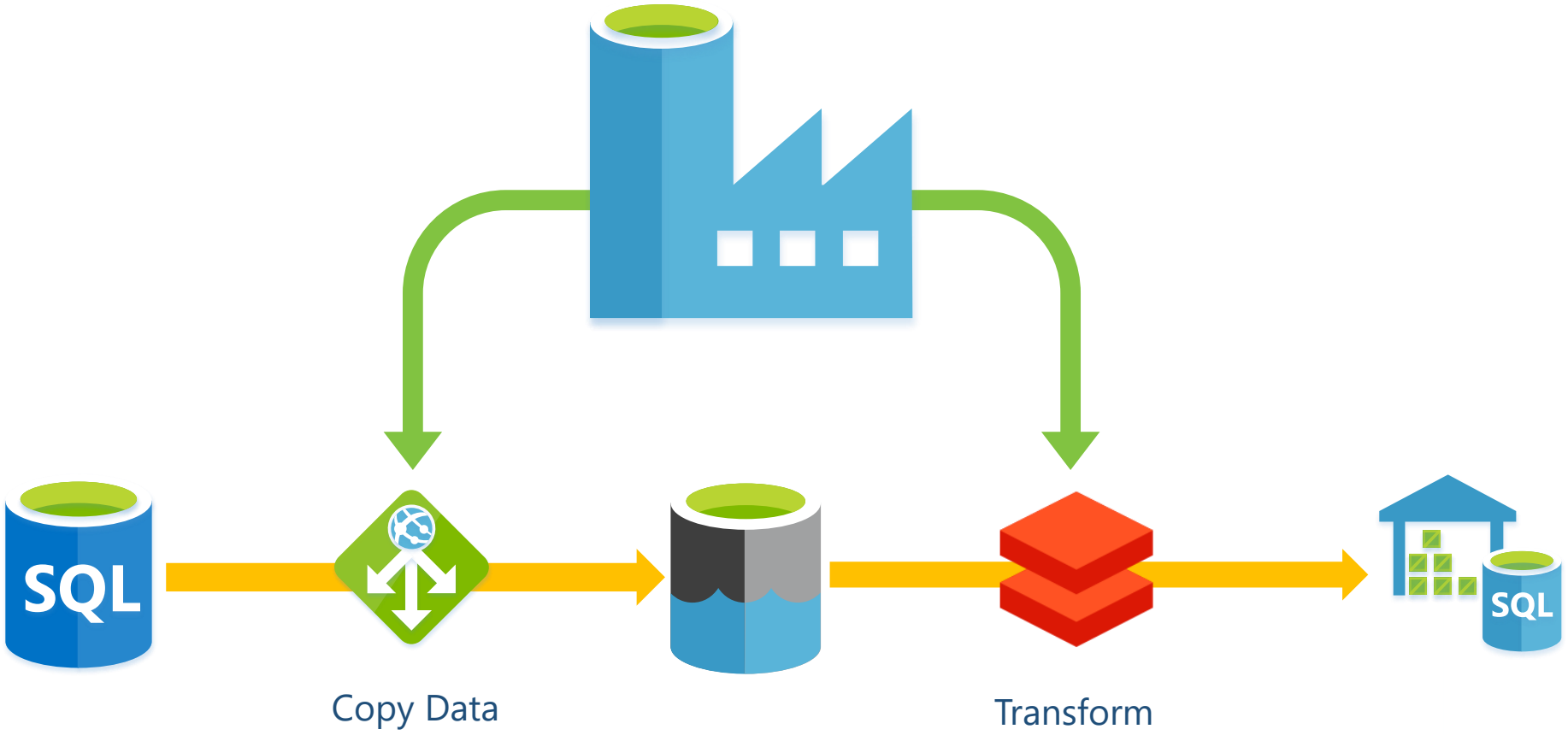
C++

{Event/Location}-{Month}-{Year}

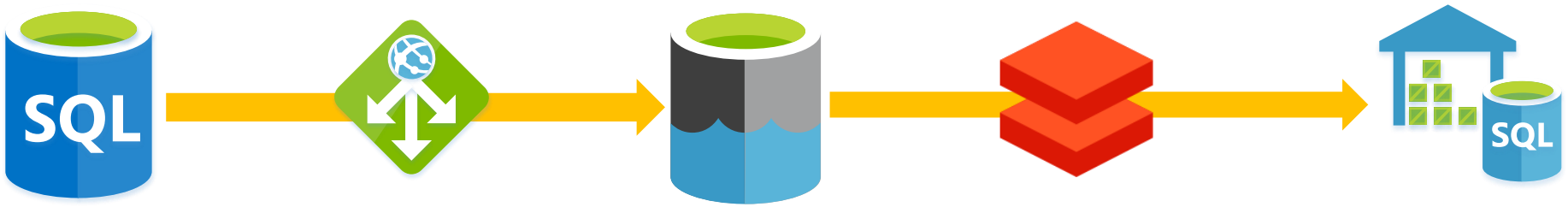
Azure Data Factory



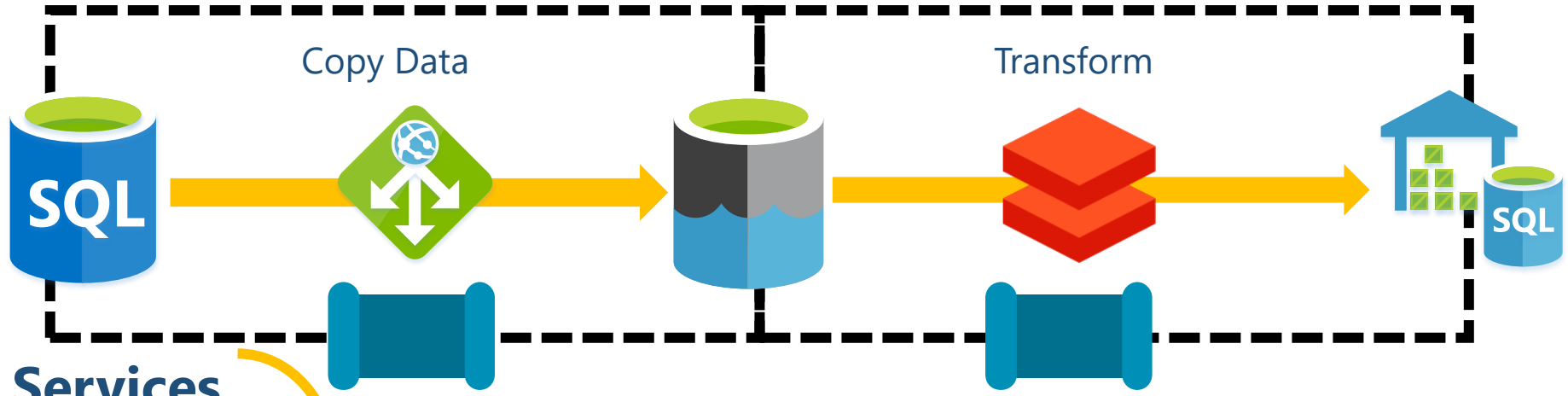
What is Azure Data Factory?



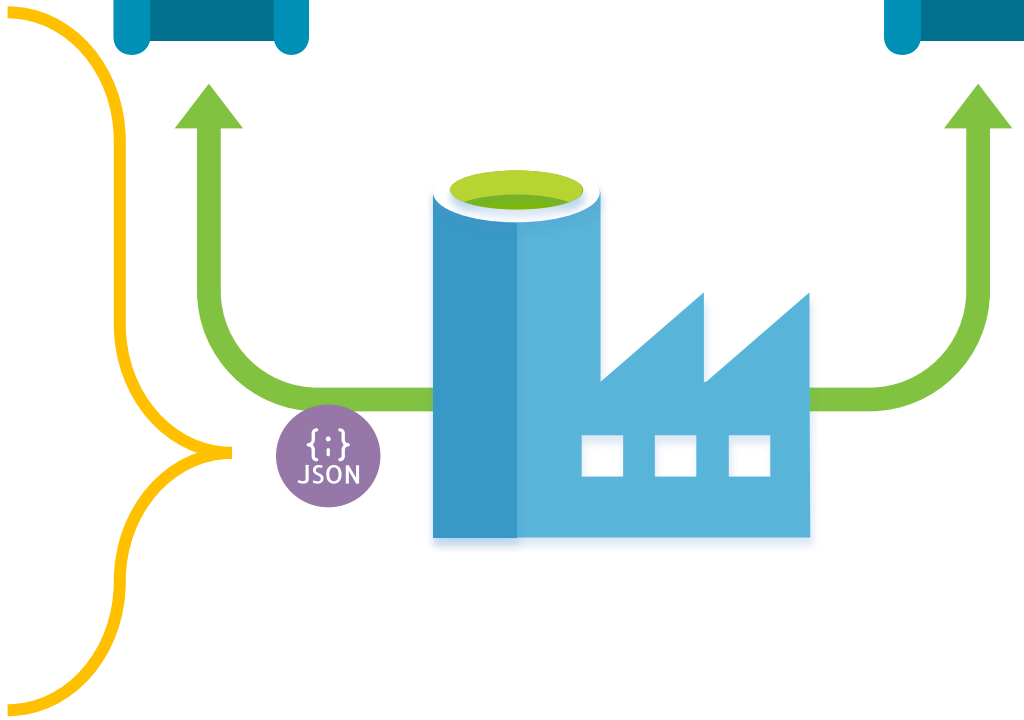
What is Azure Data Factory?



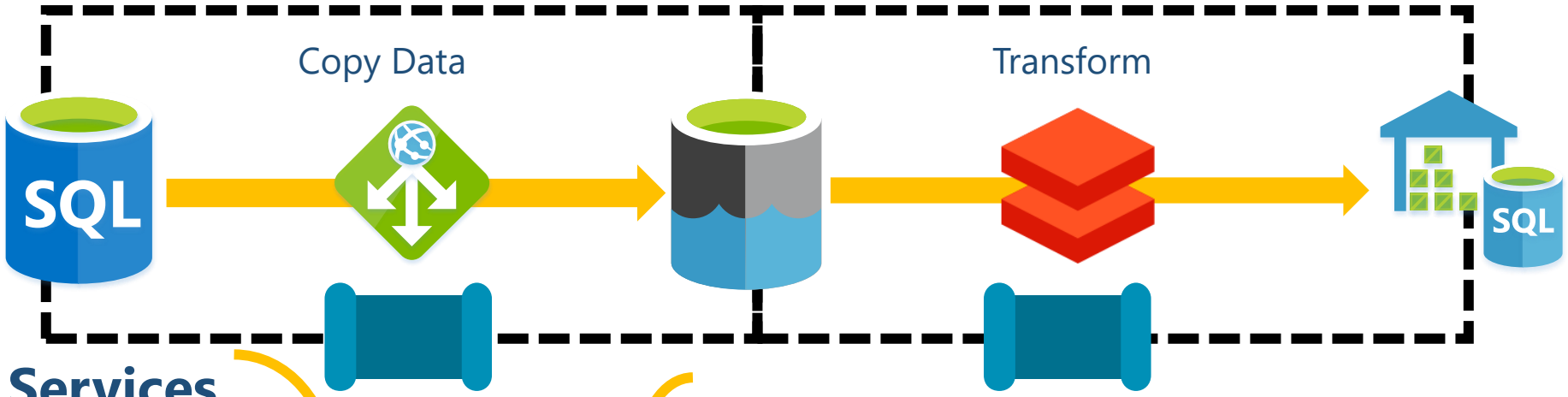
Data Factory Components



1. **Linked Services**
2. **Data Sets**
3. **Activities**
4. **Pipelines**
5. **Triggers**



Data Factory Components

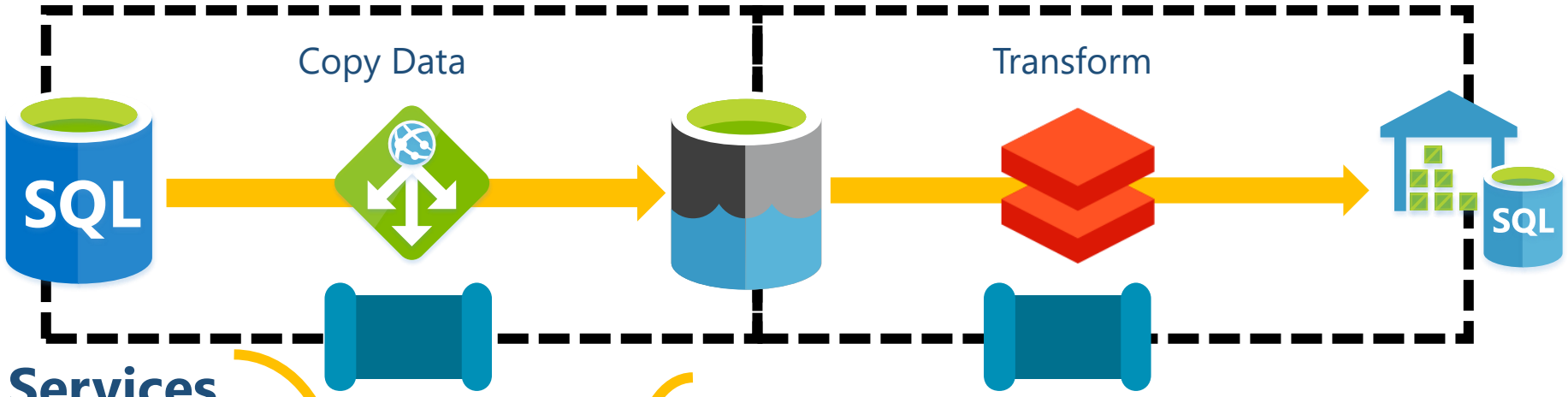


1. Linked Services
2. Data Sets
3. Activities
4. Pipelines
5. Triggers

JSON

```
{
"name": "GenericSQLDB",
"type": "Microsoft.DataFactory/factories/linkedservices",
"properties": {
    "parameters": {
        "ServerInstance": {
            "type": "String"
        },
        "DatabaseName": {
            "type": "String"
        },
        "SQLUser": {
            "type": "String"
        },
        "SQLPassword": {
            "type": "String"
        }
    },
    "type": "AzureSqlDatabase",
    "typeProperties": {
        "connectionString": "Integrated Security=False;Encrypt=True;ConnectionTimeout=30;
            Data Source=@{linkedService().ServerInstance};
            InitialCatalog=@{linkedService().DatabaseName};
            UserID=@{linkedService().SQLUser};
            Password=@{linkedService().SQLPassword}"
    }
}
```

Data Factory Components



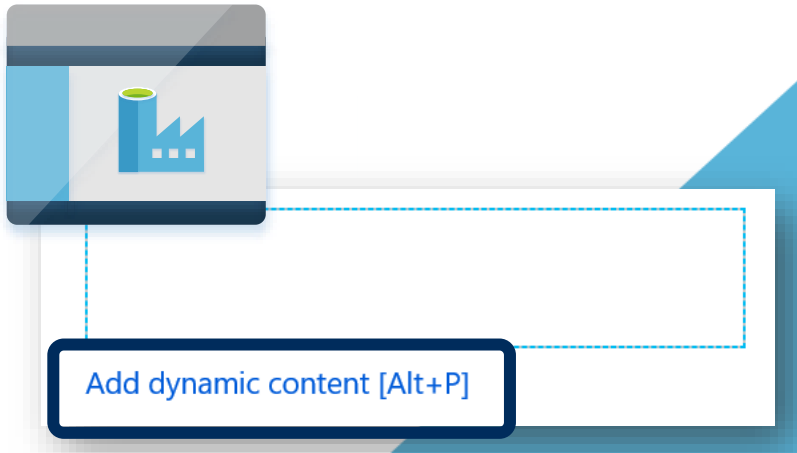
1. **Linked Services**
2. **Data Sets**
3. **Activities**
4. **Pipelines**
5. **Triggers**

JSON

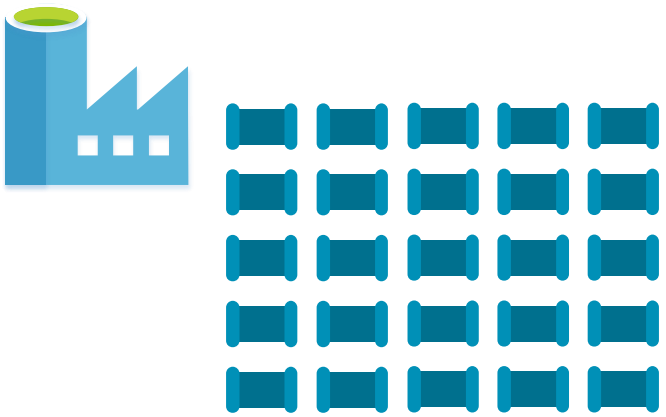
Expression Builder

@{.....} ← Parameters
System Variables

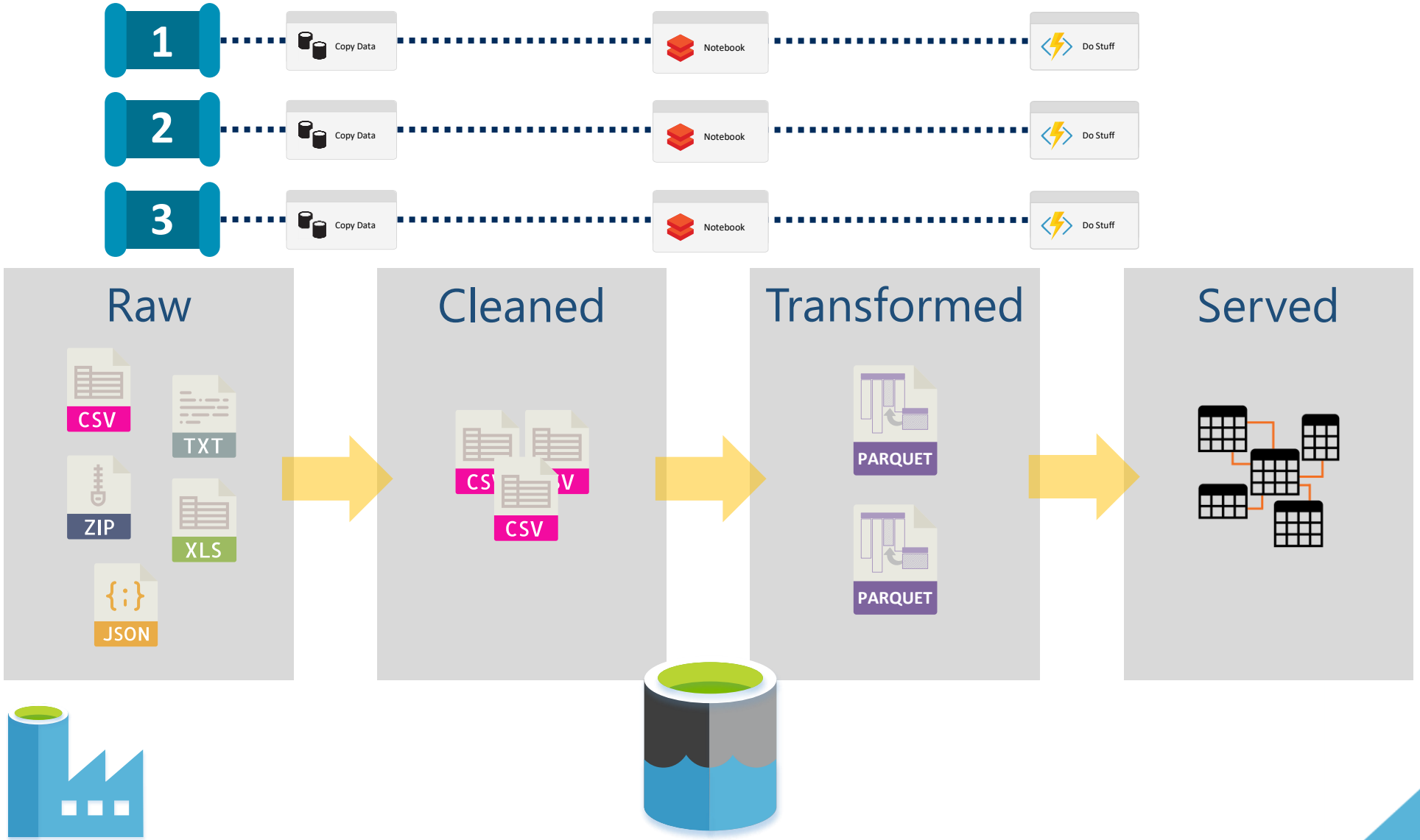
- Collection
- Conversation
- Date
- Logical
- Math
- String



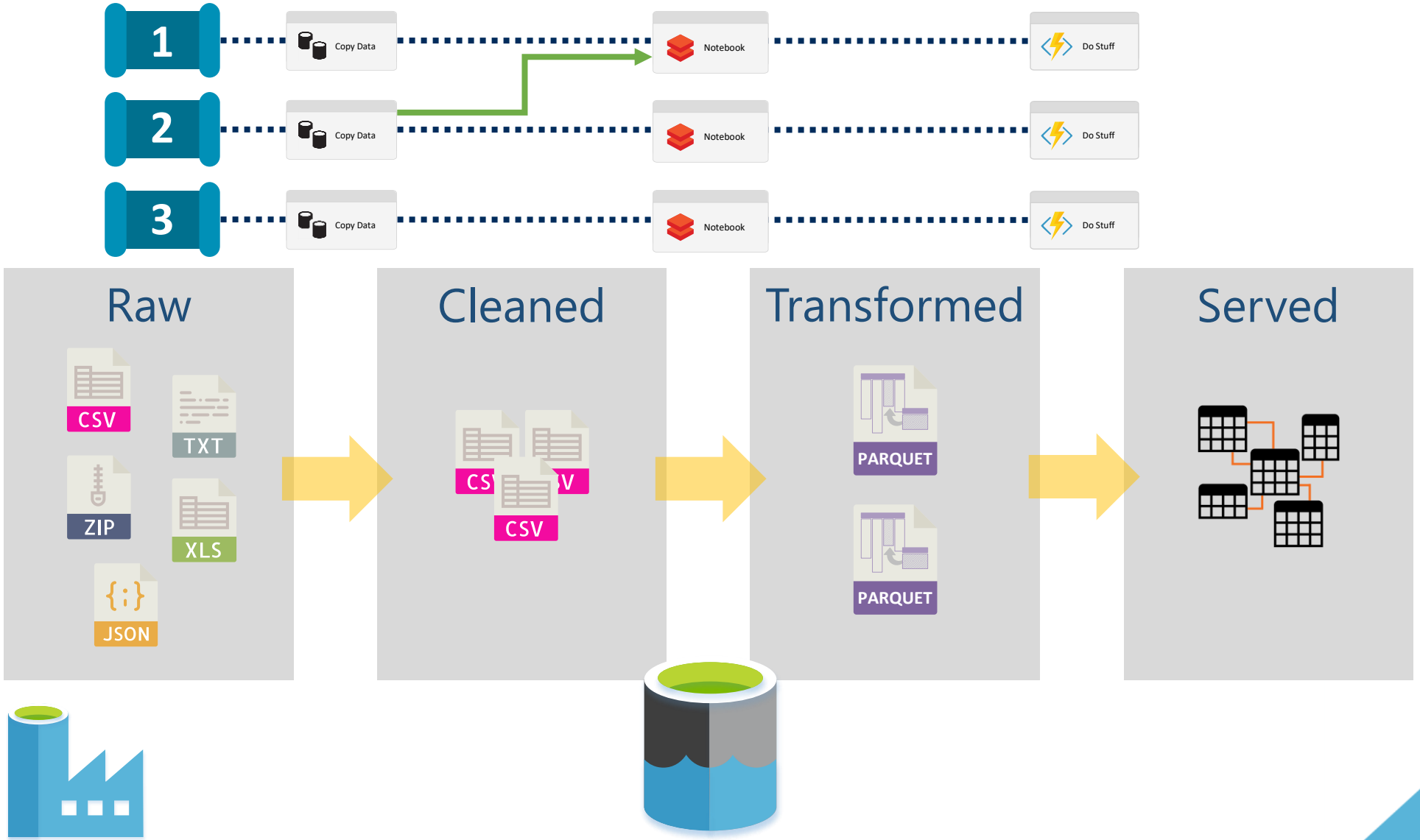
Problem: How do we structure our Data Factory Pipelines?



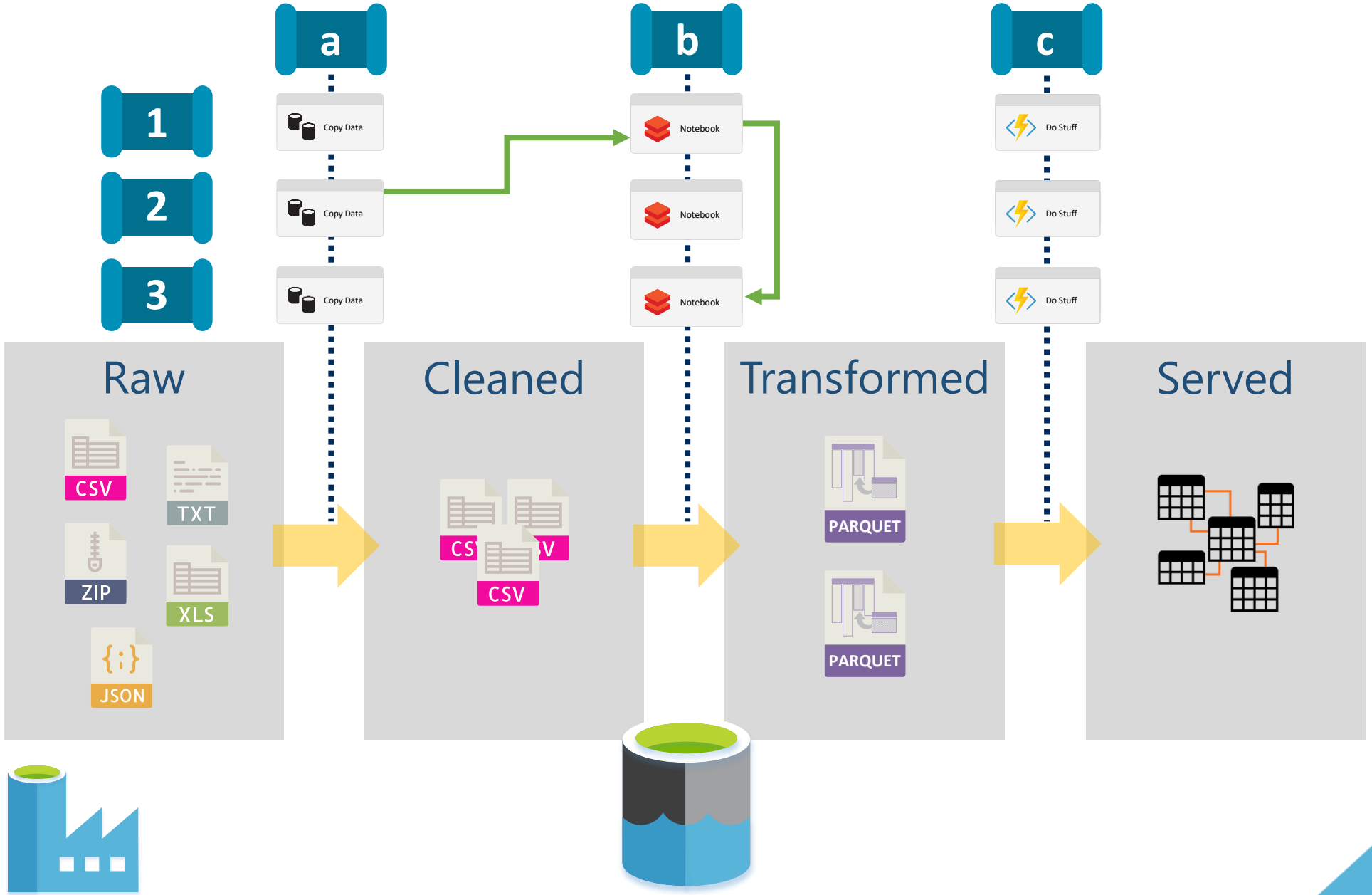
Problem


1
Copy Data
Notebook
Do Stuff
2
Copy Data
Notebook
Do Stuff
3
Copy Data
Notebook
Do Stuff
Raw
CSV
TXT
ZIP
XLS
JSON
Cleaned
CSV
Transformed
PARQUET
PARQUET
Served

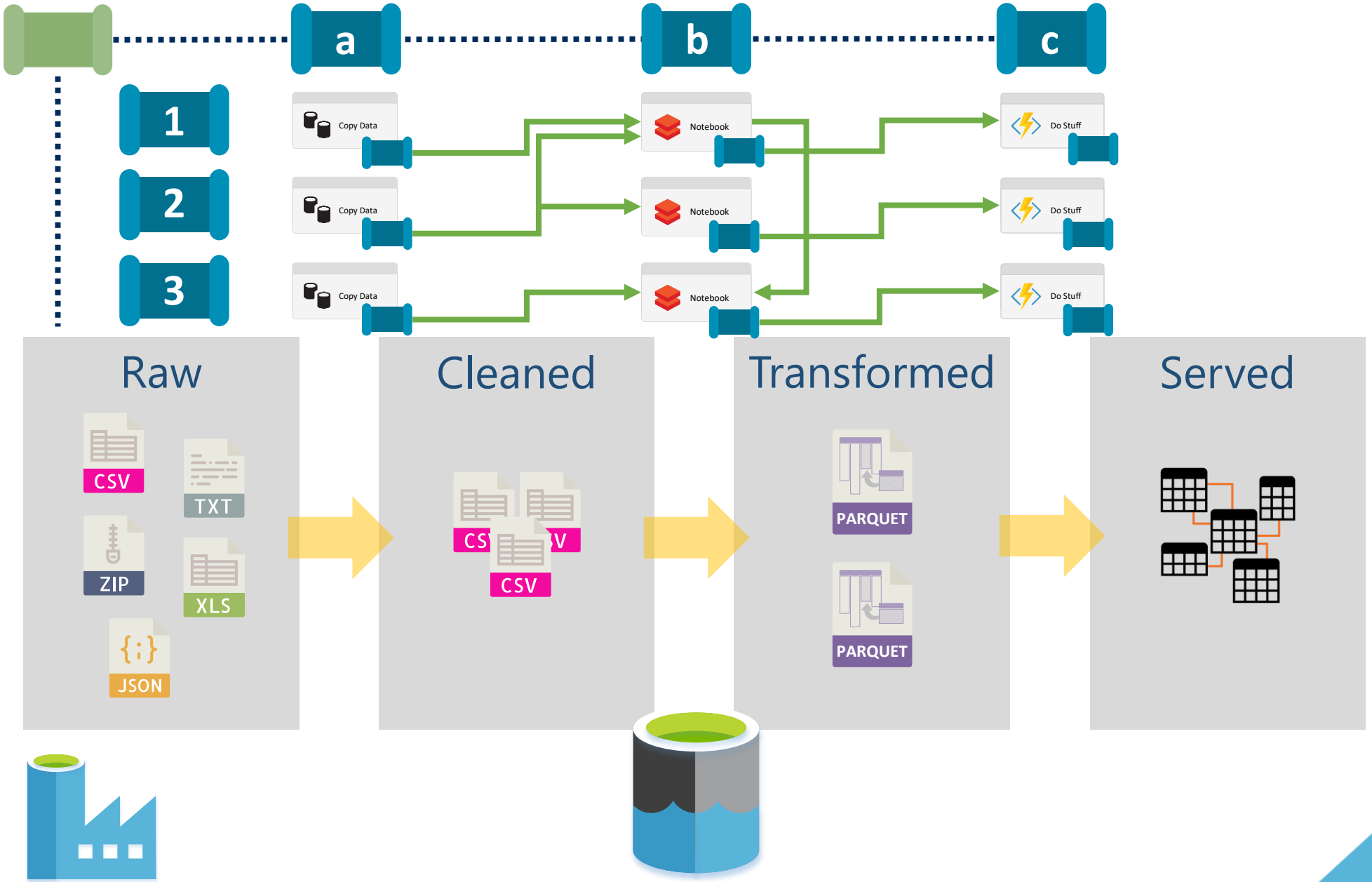
Problem


1
2
3
Copy Data
Copy Data
Copy Data
Notebook
Notebook
Notebook
Do Stuff
Do Stuff
Do Stuff
Raw
Cleaned
Transformed
Served
CSV
TXT
ZIP
XLS
JSON
CSV
PARQUET
PARQUET

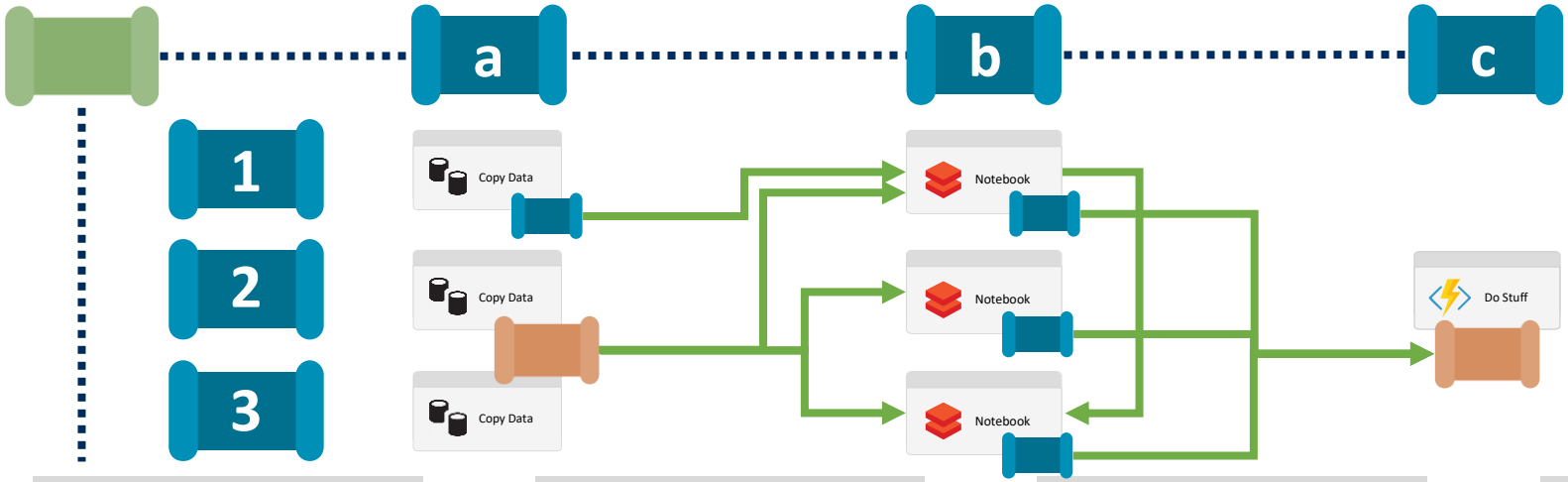
Problem


a
b
c
1
2
3
Copy Data
Copy Data
Copy Data
Notebook
Notebook
Notebook
Do Stuff
Do Stuff
Do Stuff
Raw
Cleaned
Transformed
Served
CSV
TXT
ZIP
XLS
JSON
CSV
PARQUET
PARQUET

Problem


a
b
c
1
2
3
Copy Data
Copy Data
Copy Data
Notebook
Notebook
Notebook
Do Stuff
Do Stuff
Do Stuff
Raw
Cleaned
Transformed
Served
CSV
TXT
ZIP
XLS
JSON
CSV
PARQUET
PARQUET

Problem

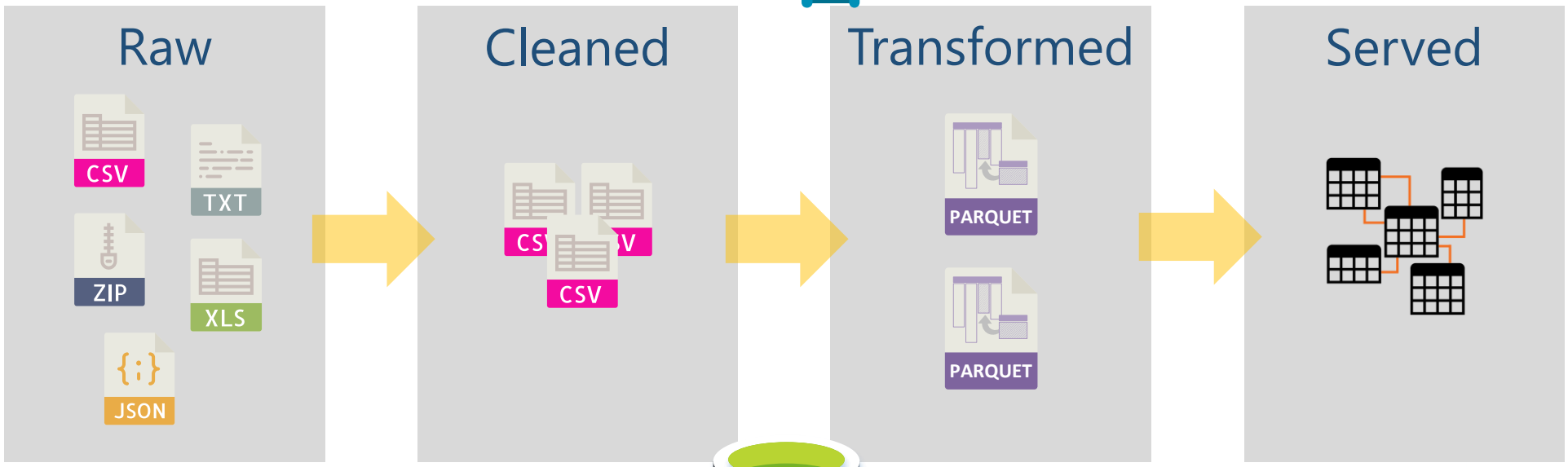


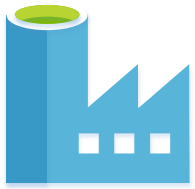
Only 40 Activities per Pipeline.

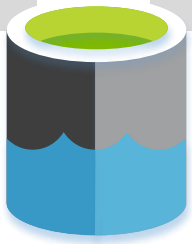
Grandparent pipeline for all processing.

Parent pipeline to consolidate work.

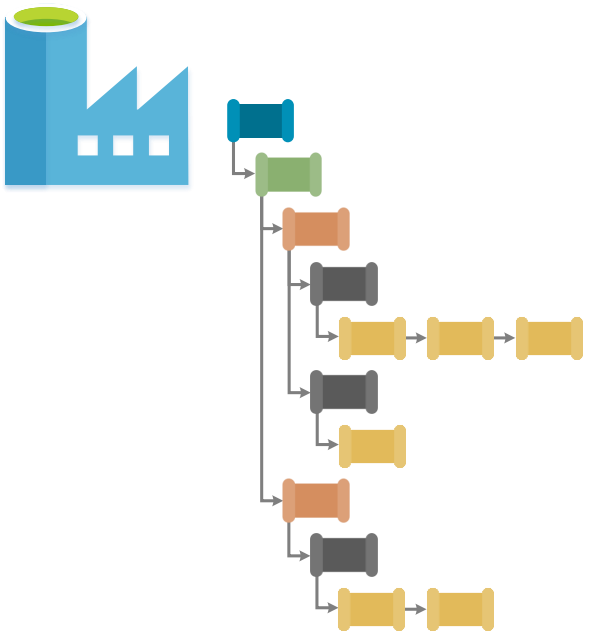
Child pipelines for low level dependencies.



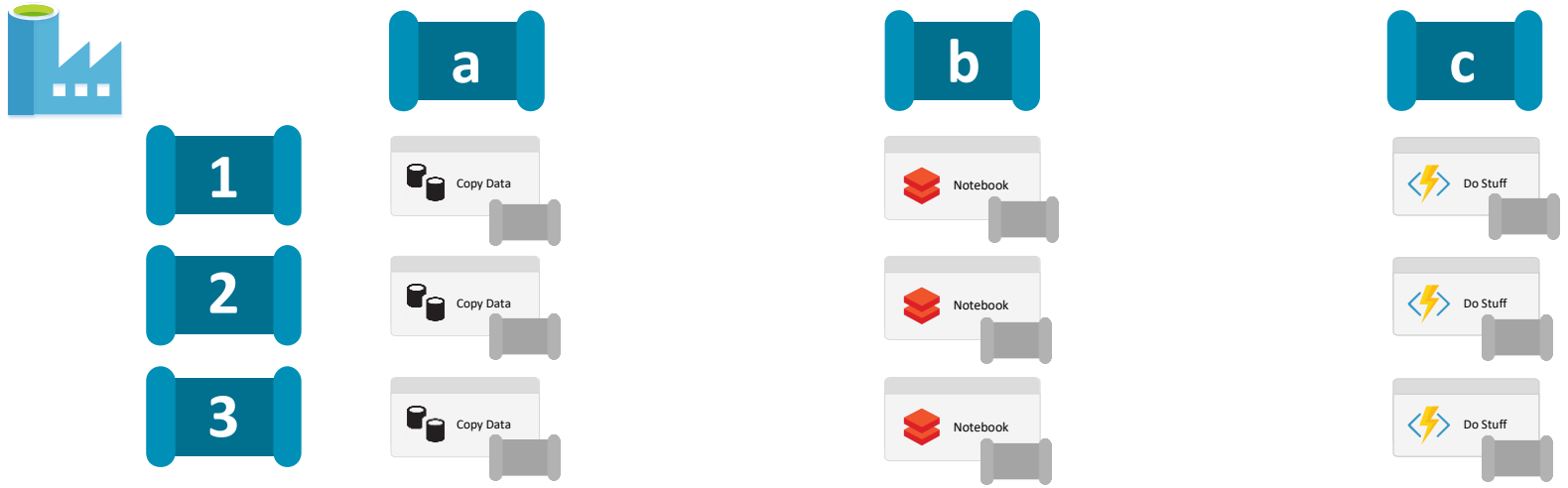




Solution: Use Metadata to Drive Data Factory Pipelines



Solution

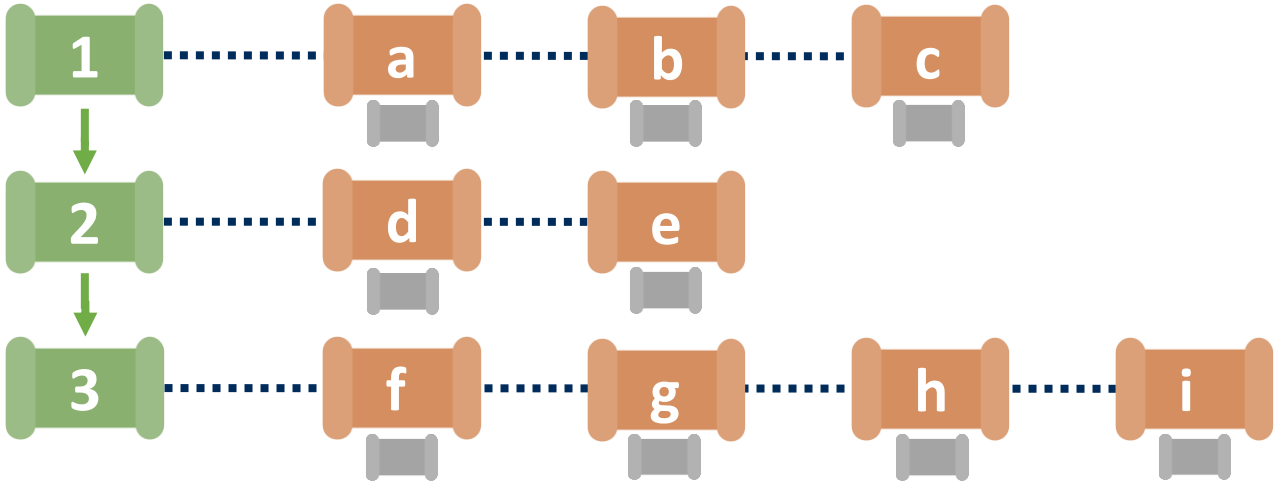


Stages
1
2
3

Pipelines
a
b
c
d
e
f
g
h
i

Stage	Pipeline
1	a
1	b
1	c
2	d
2	e
3	f
3	g
3	h
3	i

Solution

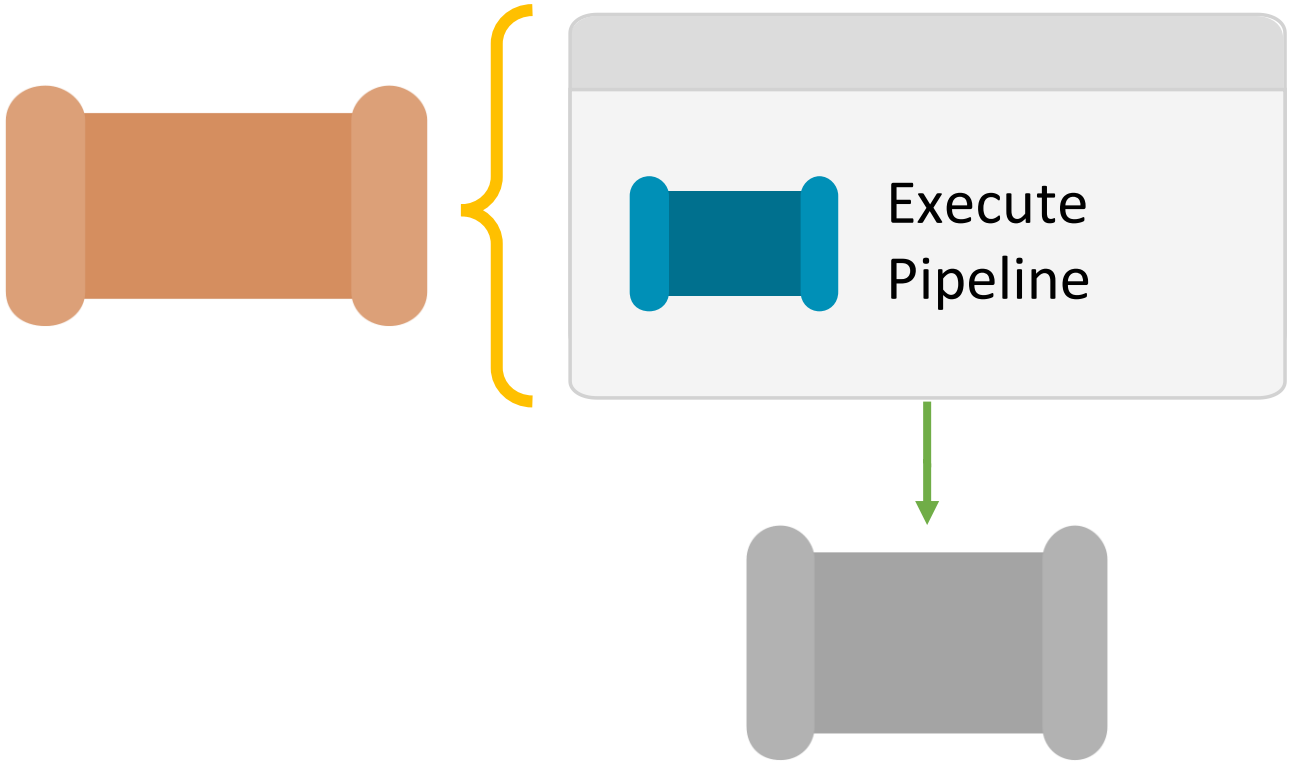


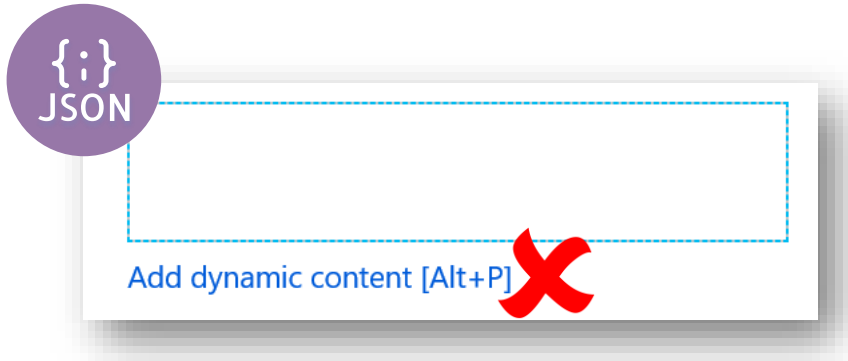
Stages
1
2
3

Pipelines
a
b
c
d
e
f
g
h
i

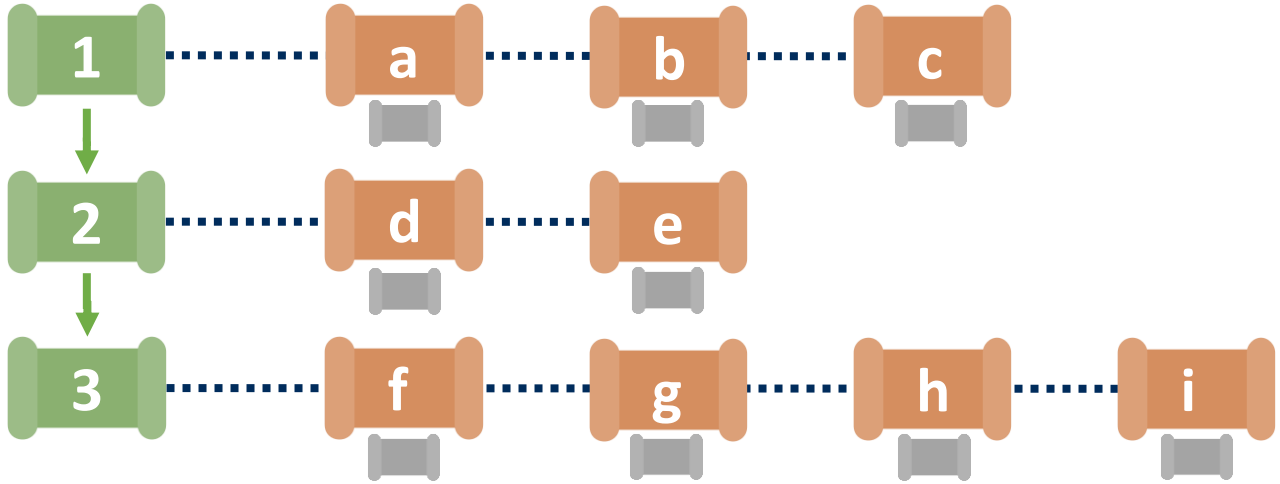
Stage	Pipeline
1	a
1	b
1	c
2	d
2	e
3	f
3	g
3	h
3	i

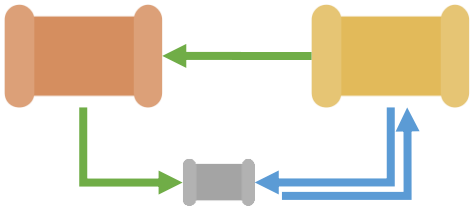
Execute Pipeline Activity





Solution





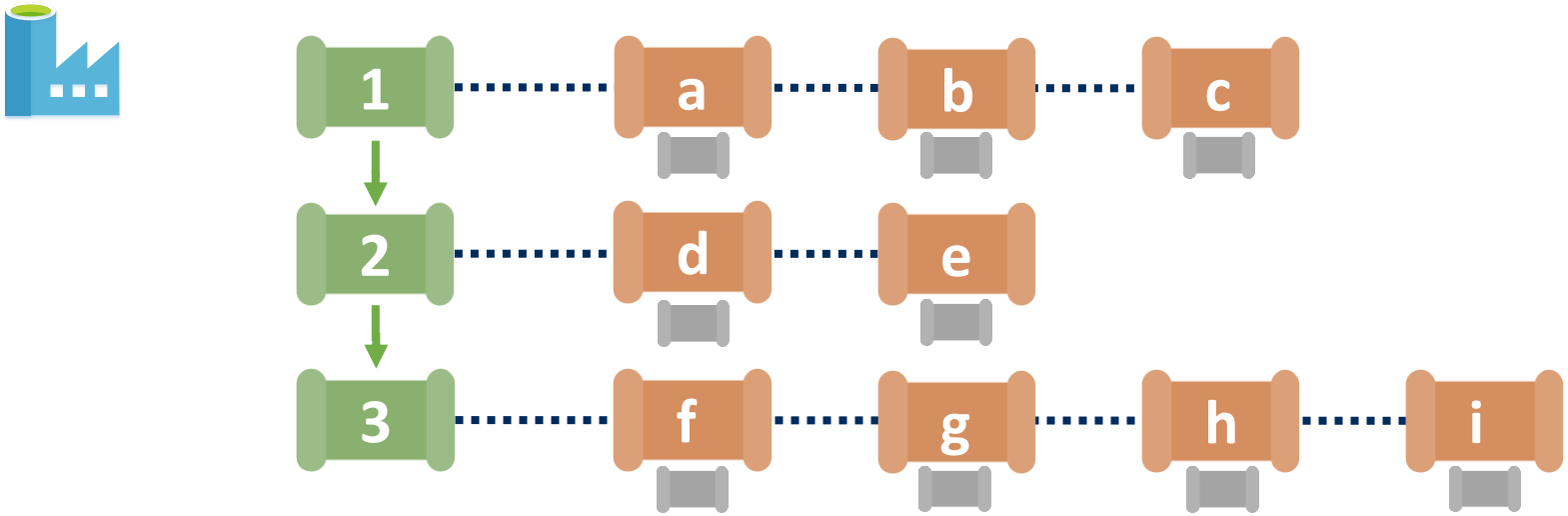
Stages
1
2
3

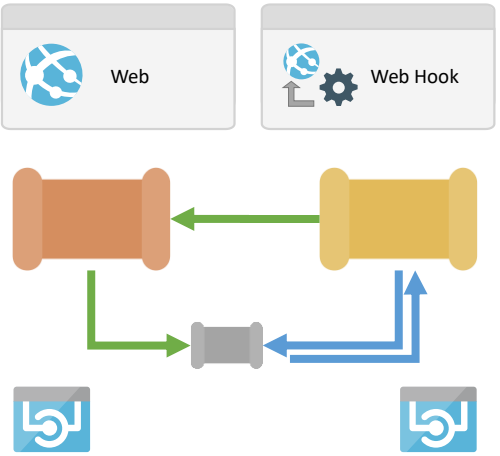
Pipelines
a
b
c
d
e
f
g
h
i

Stage	Pipeline
1	a
1	b
1	c
2	d
2	e
3	f
3	g
3	h
3	i

https://cloudblogs.microsoft.com/industry-blog/en-gb/technetuk/2020/03/19/enterprise-wide-orchestration-using-multiple-data-factories/

Solution – Option 1





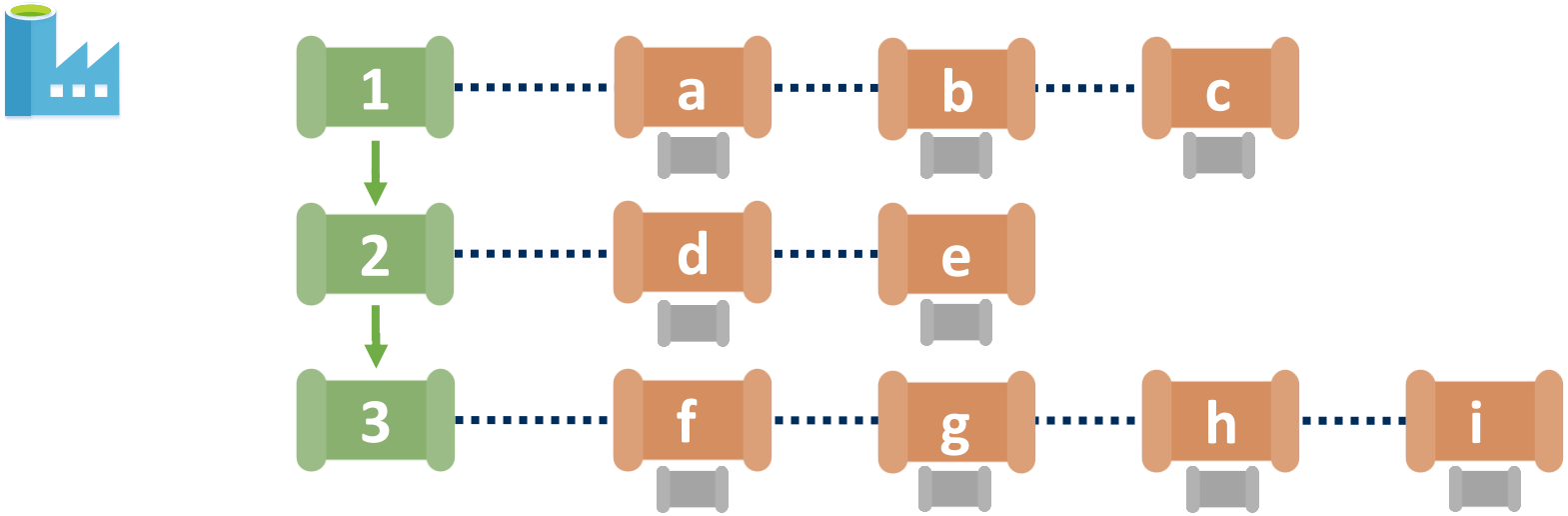
Stages
1
2
3

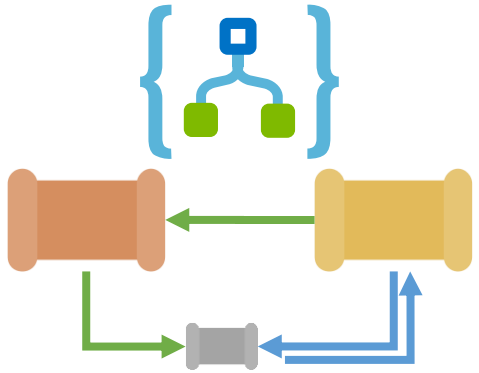
Pipelines
a
b
c
d
e
f
g
h
i

Stage	Pipeline
1	a
1	b
1	c
2	d
2	e
3	f
3	g
3	h
3	i

https://cloudblogs.microsoft.com/industry-blog/en-gb/technetuk/2020/03/19/enterprise-wide-orchestration-using-multiple-data-factories/

Solution – Option 2



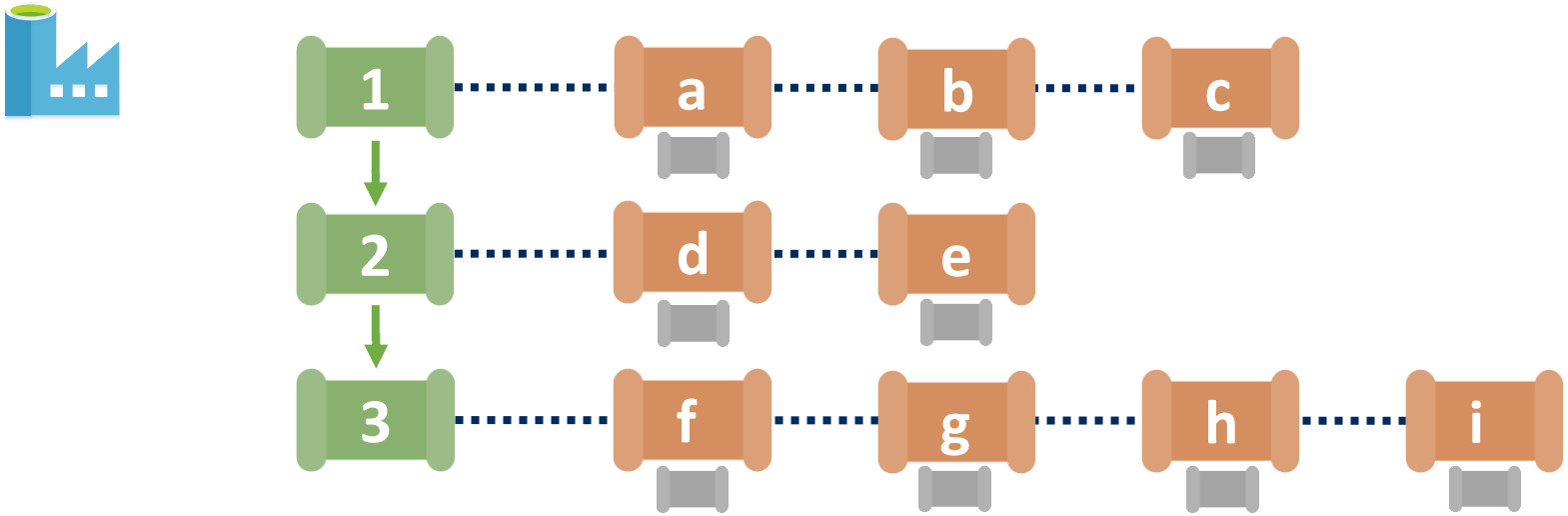


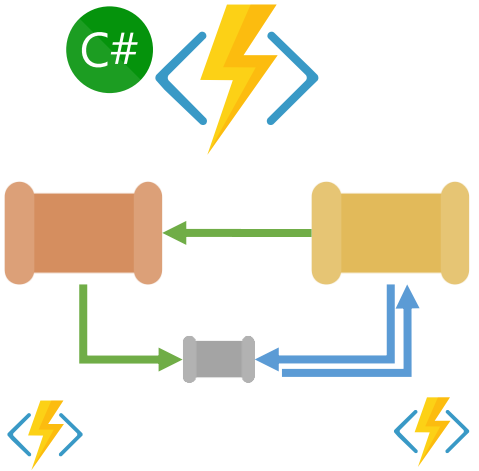
Stages
1
2
3

Pipelines
a
b
c
d
e
f
g
h
i

Stage	Pipeline
1	a
1	b
1	c
2	d
2	e
3	f
3	g
3	h
3	i

Solution – Option 3



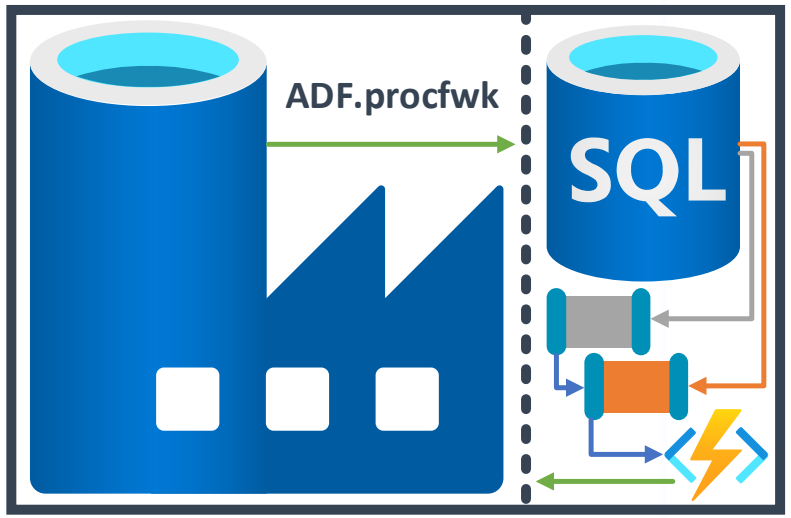


Stages
1
2
3

Pipelines
a
b
c
d
e
f
g
h
i

Stage	Pipeline
1	a
1	b
1	c
2	d
2	e
3	f
3	g
3	h
3	i

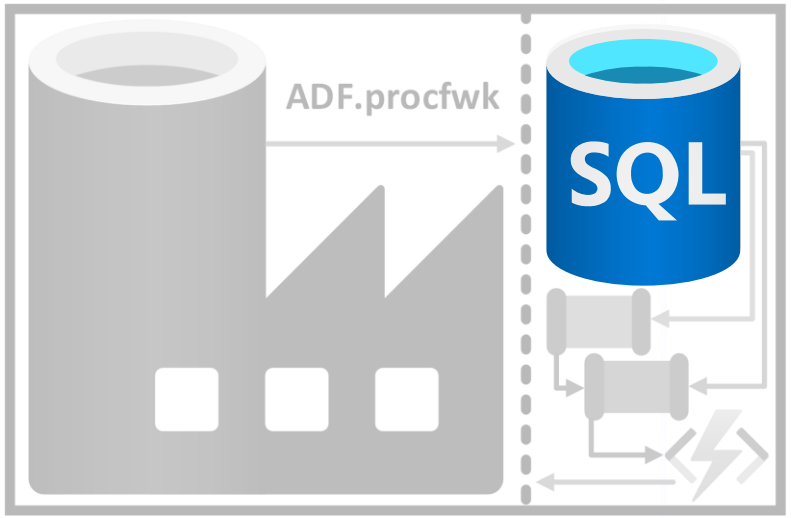
Introducing ADF.procfwk



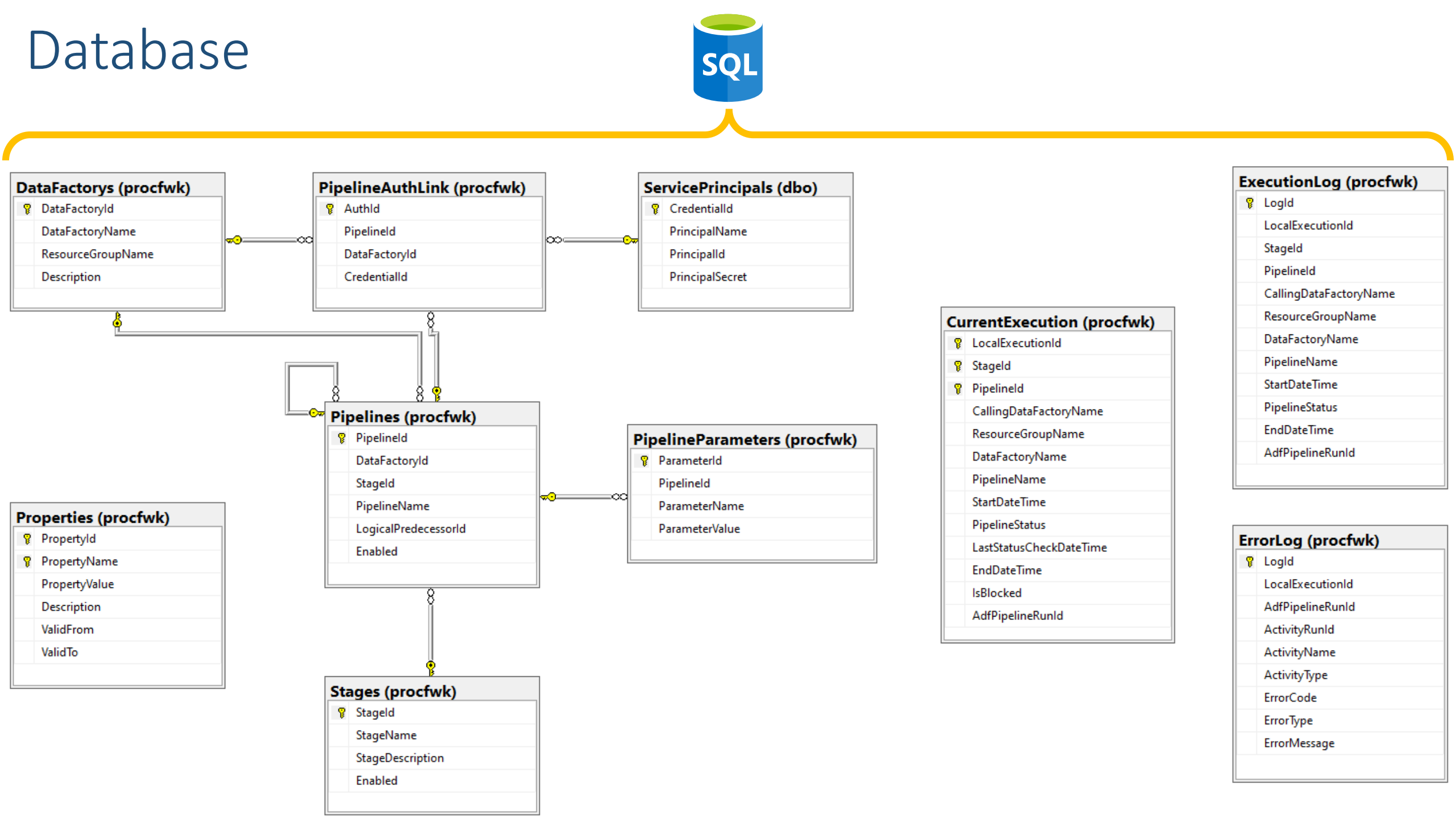
Key Features

- Granular metadata control.
- Metadata integrity checking.
- Global properties.
- Dependency handling.
- Execution restart-ability.
- Parallel execution.
- Full execution and error logs.
- Operational dashboards.
- Low cost orchestration.
- Disconnection between framework and Worker pipelines.
- Cross Data Factory control flows.
- Pipeline parameter support.
- Simple troubleshooting.
- Easy deployment.
- Email alerting.

ADF.procfwk – Database

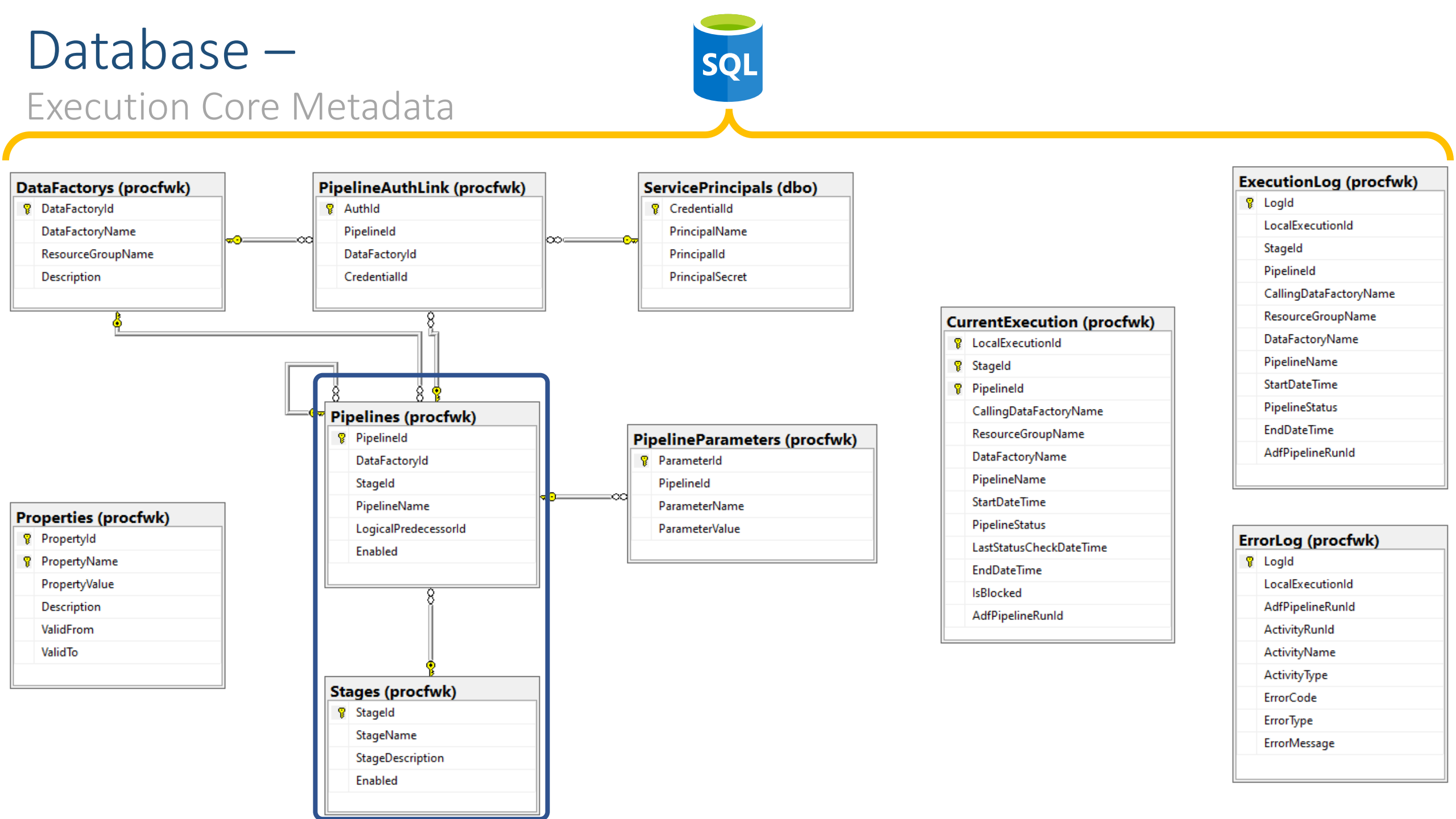


Database


DataFactorys (procfwk)
DataFactoryId
DataFactoryName
ResourceGroupName
Description
PipelineAuthLink (procfwk)
AuthId
PipelineId
DataFactoryId
CredentialId
ServicePrincipals (dbo)
CredentialId
PrincipalName
PrincipalId
PrincipalSecret
ExecutionLog (procfwk)
LogId
LocalExecutionId
StageId
PipelineId
CallingDataFactoryName
ResourceGroupName
DataFactoryName
PipelineName
StartDateTime
PipelineStatus
EndDateTime
AdfPipelineRunId
CurrentExecution (procfwk)
LocalExecutionId
StageId
PipelineId
CallingDataFactoryName
ResourceGroupName
DataFactoryName
PipelineName
StartDateTime
PipelineStatus
LastStatusCheckDateTime
EndDateTime
IsBlocked
AdfPipelineRunId
Pipelines (procfwk)
PipelineId
DataFactoryId
StageId
PipelineName
LogicalPredecessorId
Enabled
PipelineParameters (procfwk)
ParameterId
PipelineId
ParameterName
ParameterValue
Properties (procfwk)
PropertyId
PropertyName
PropertyValue
Description
ValidFrom
ValidTo
ErrorLog (procfwk)
LogId
LocalExecutionId
AdfPipelineRunId
ActivityRunId
ActivityName
ActivityType
ErrorCode
ErrorType
ErrorMessage
Stages (procfwk)
StageId
StageName
StageDescription
Enabled

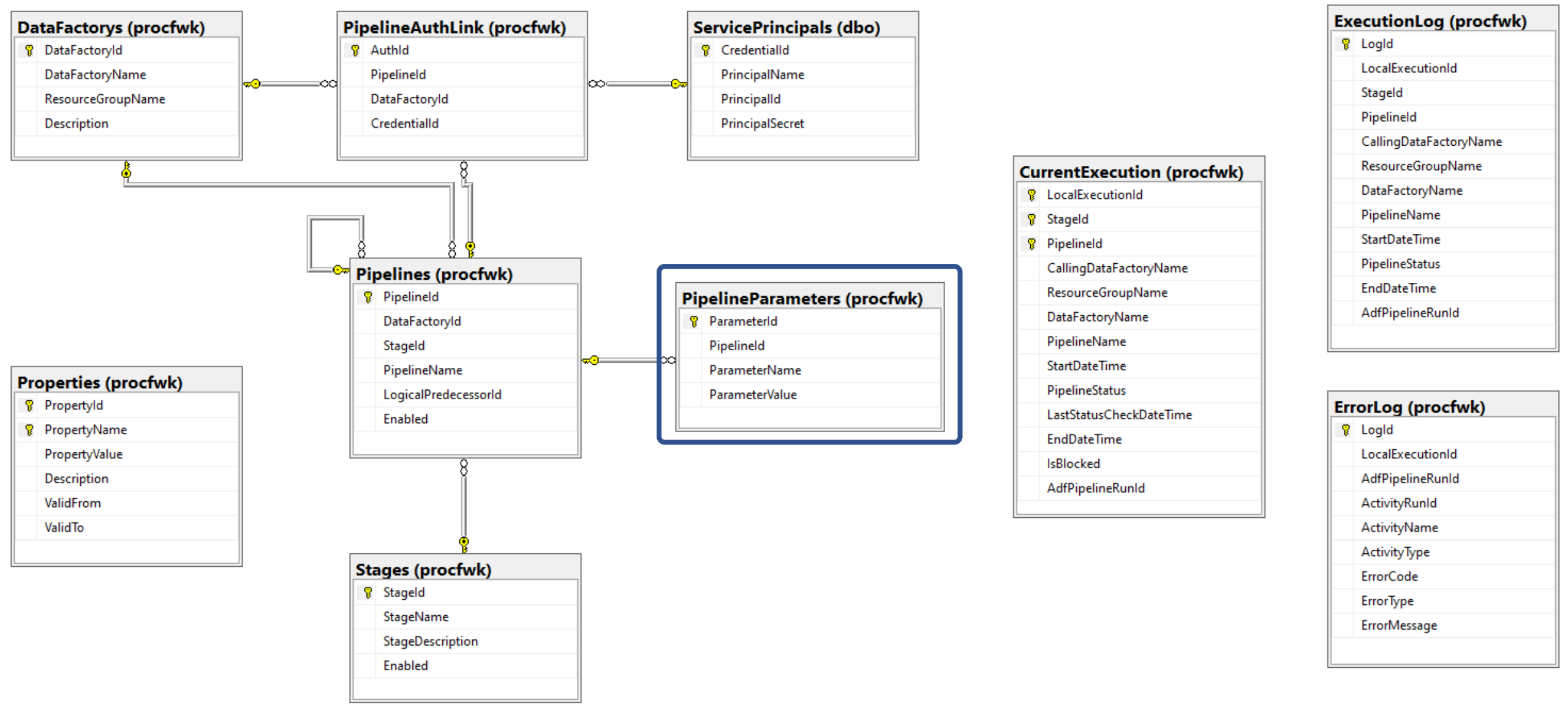
Database –

Execution Core Metadata



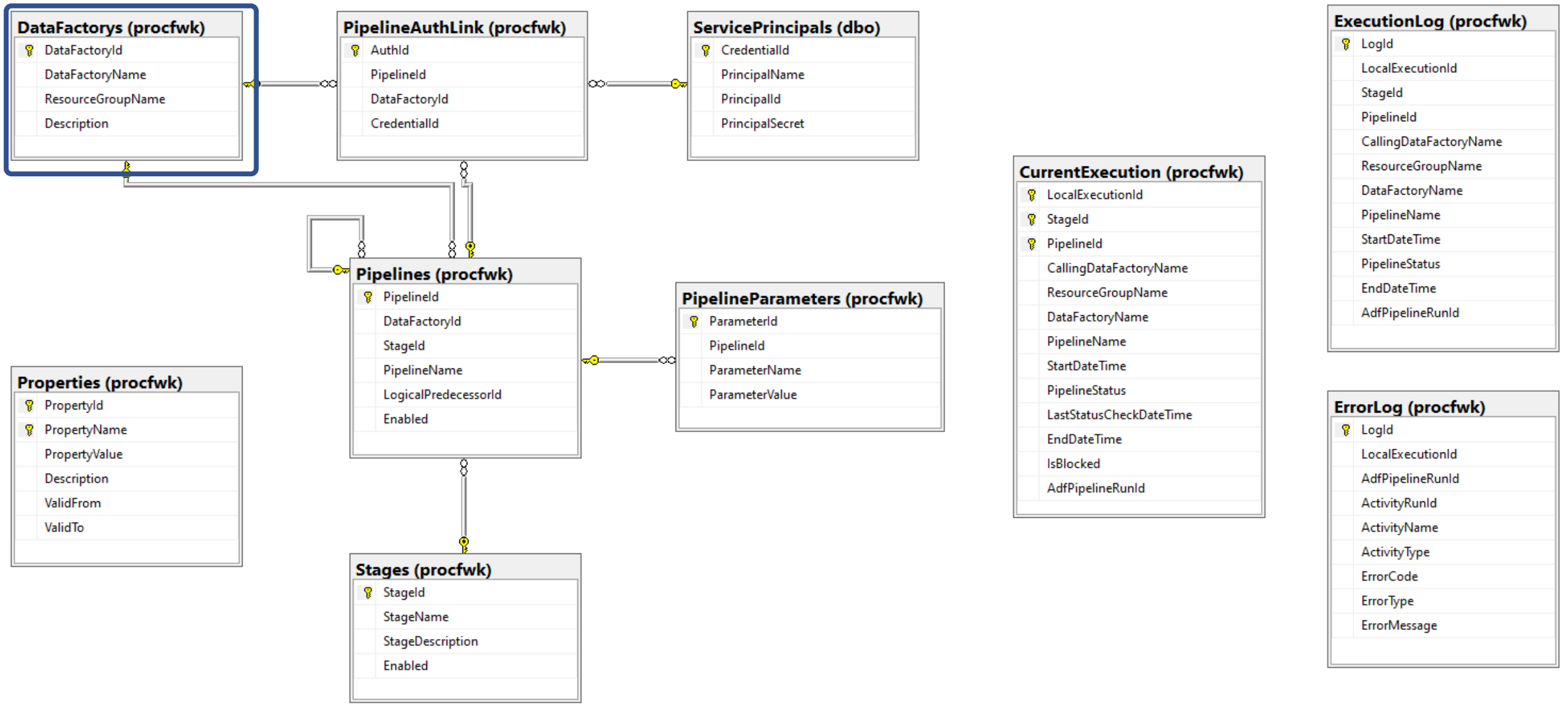
Database –

Pipeline Parameters



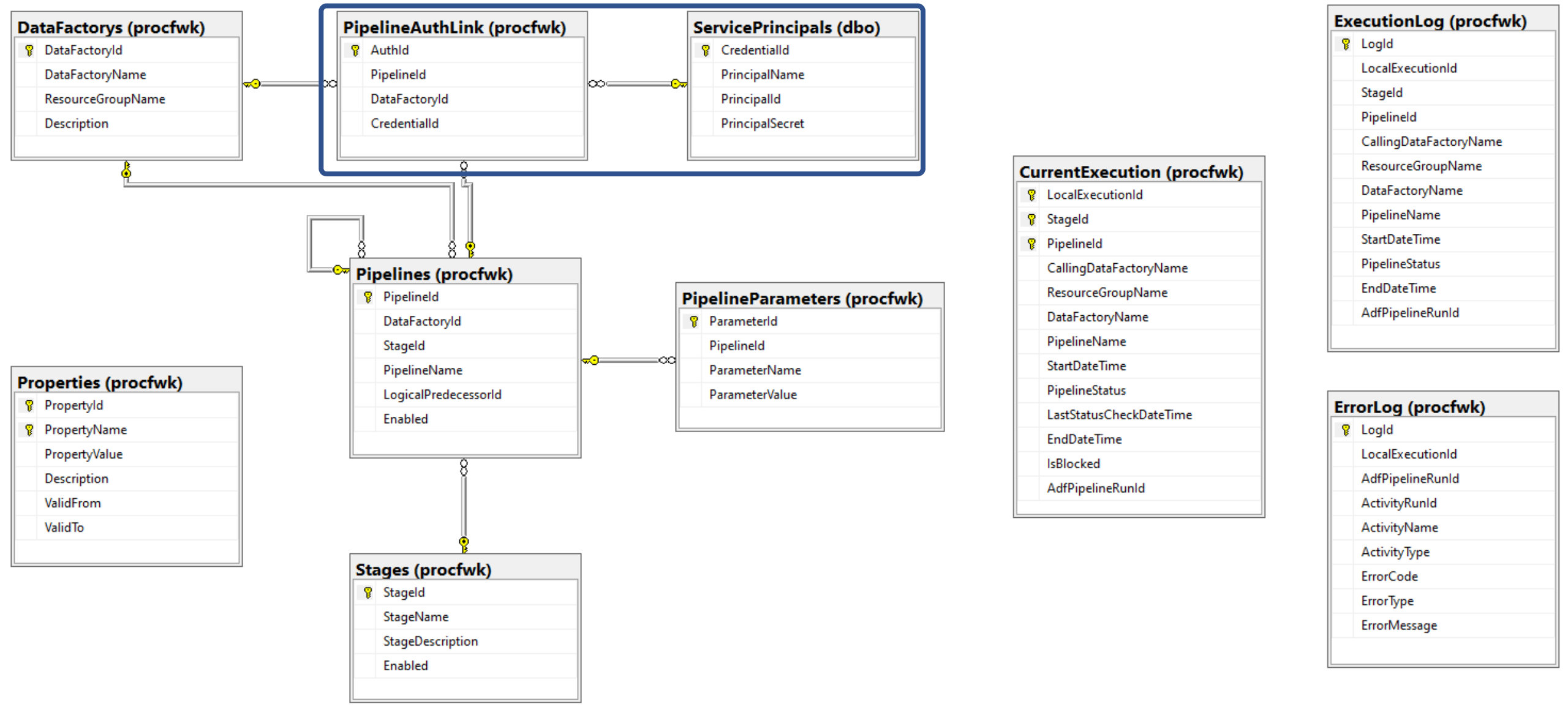
Database –

Data Factory's and Resource Groups

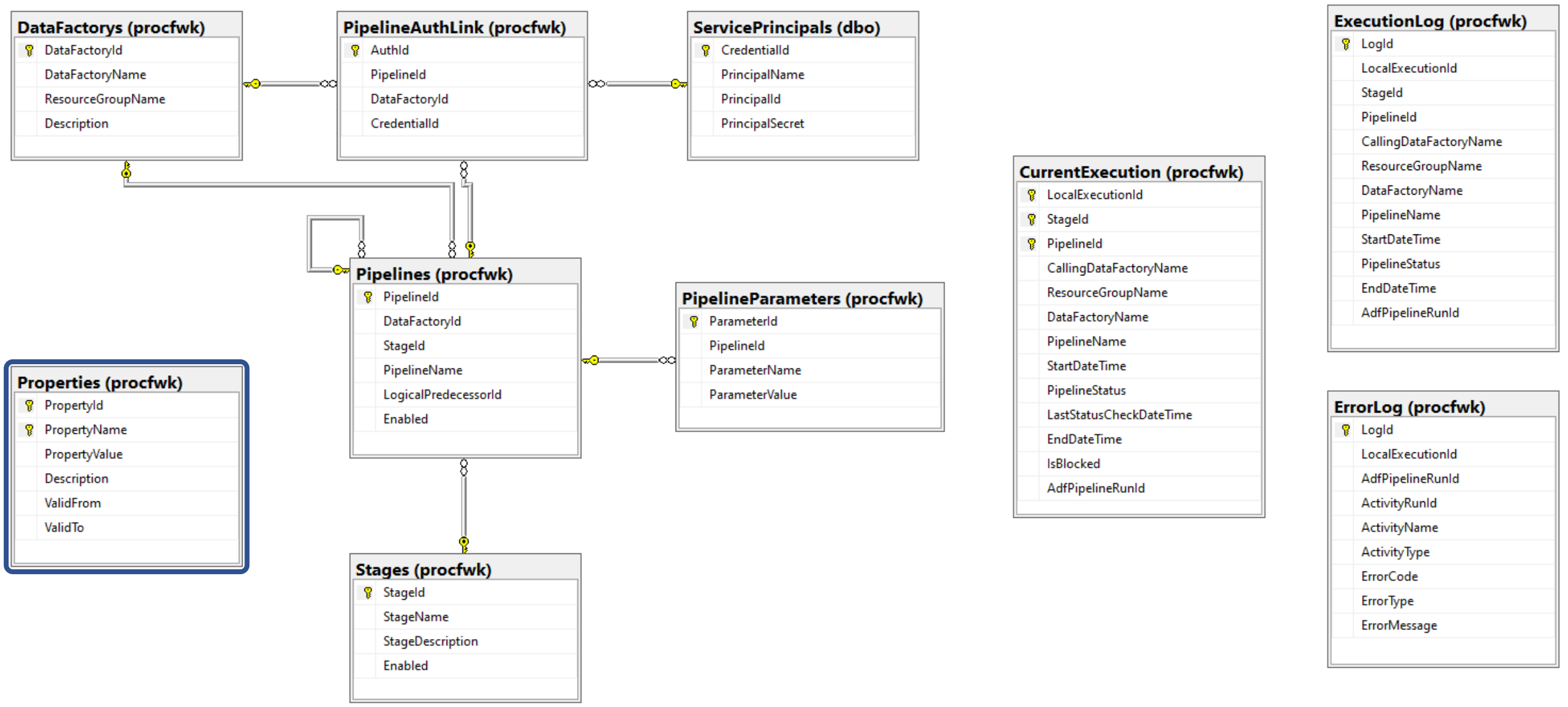


Database –

Authentication

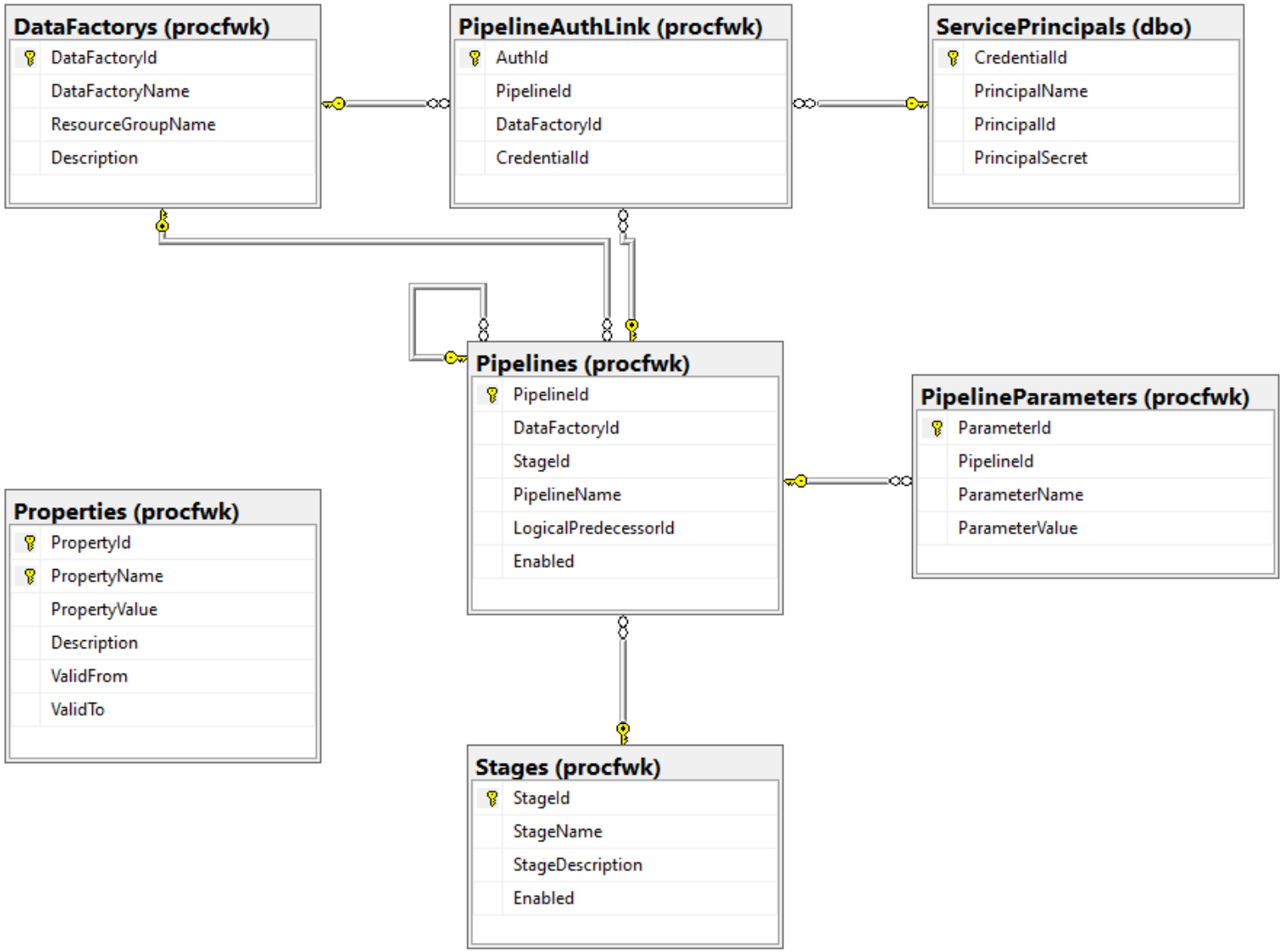


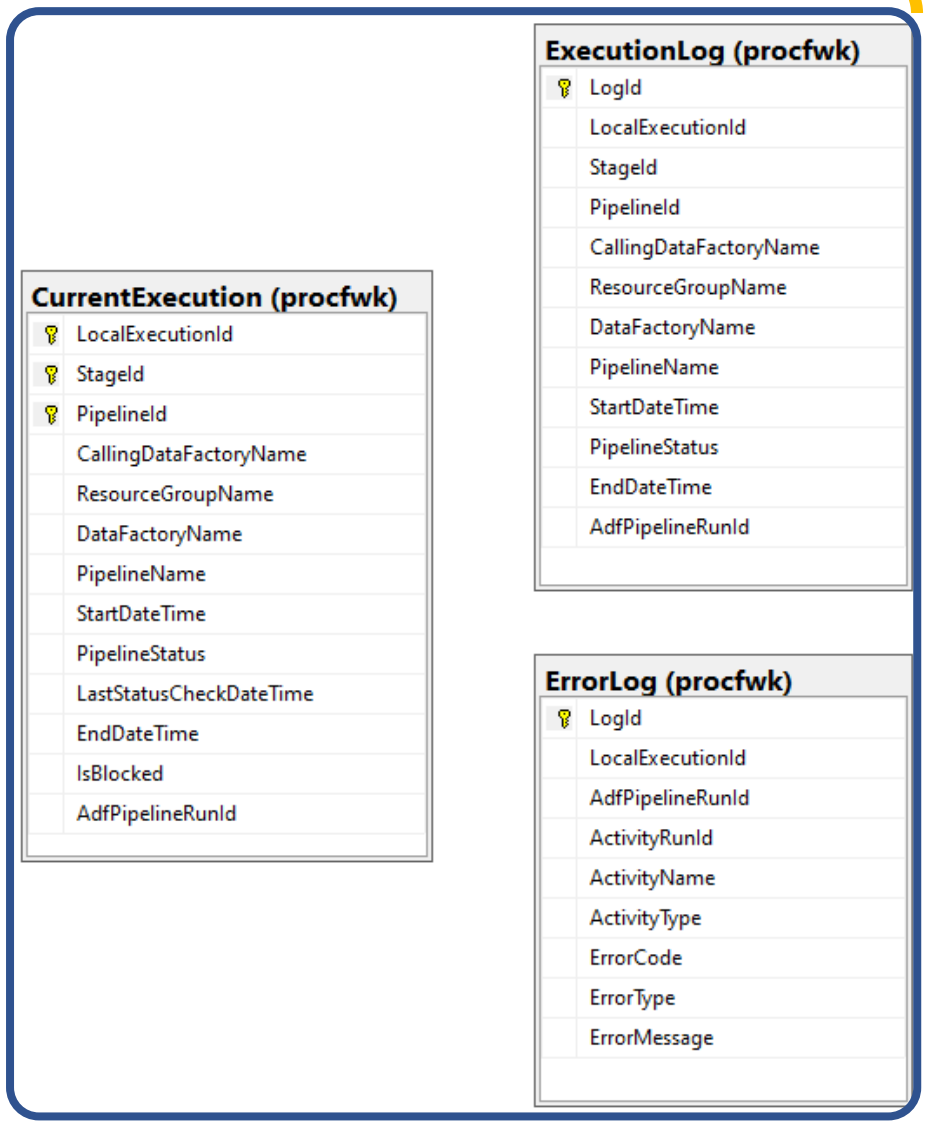
Database –
Global Controls



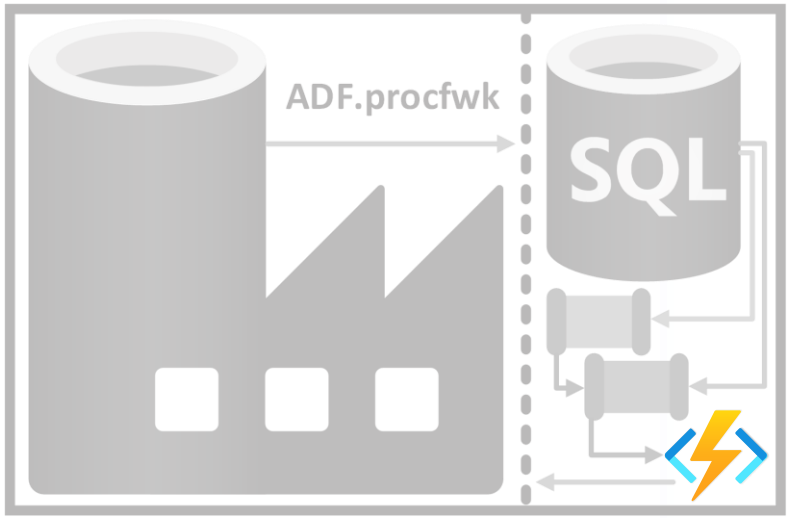
Database –

Runtime Logs





ADF.procfwk - Functions



Functions

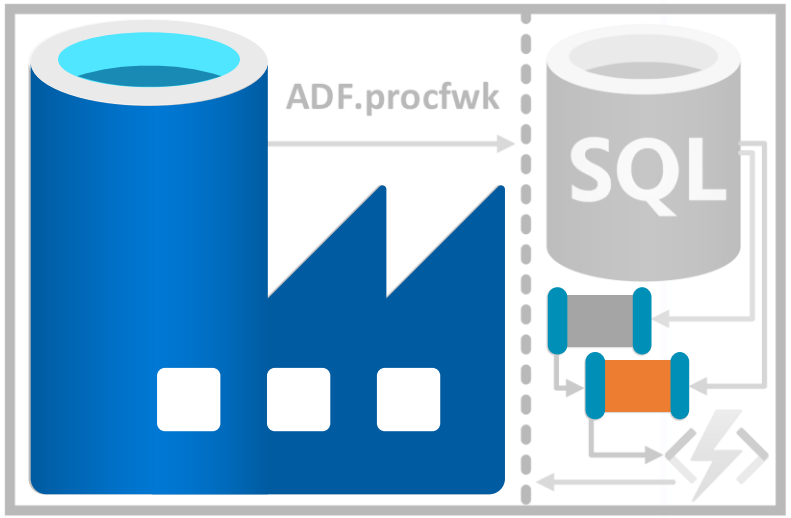
- Execute Pipeline
- Check Pipeline Status
- Get Activity Errors
- Send Email

Created using C# for .Net Core 3.1

Debug via Visual Studio or VSCode

Consumption Tier App Service Plan

ADF.procfwk – Data Factory



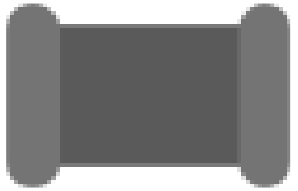
The Pipelines

- Grandparent

- Parent

- Child

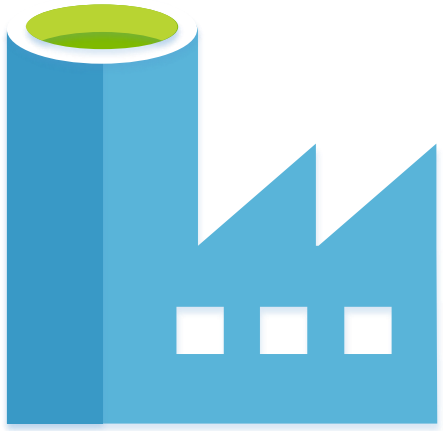
- Infant

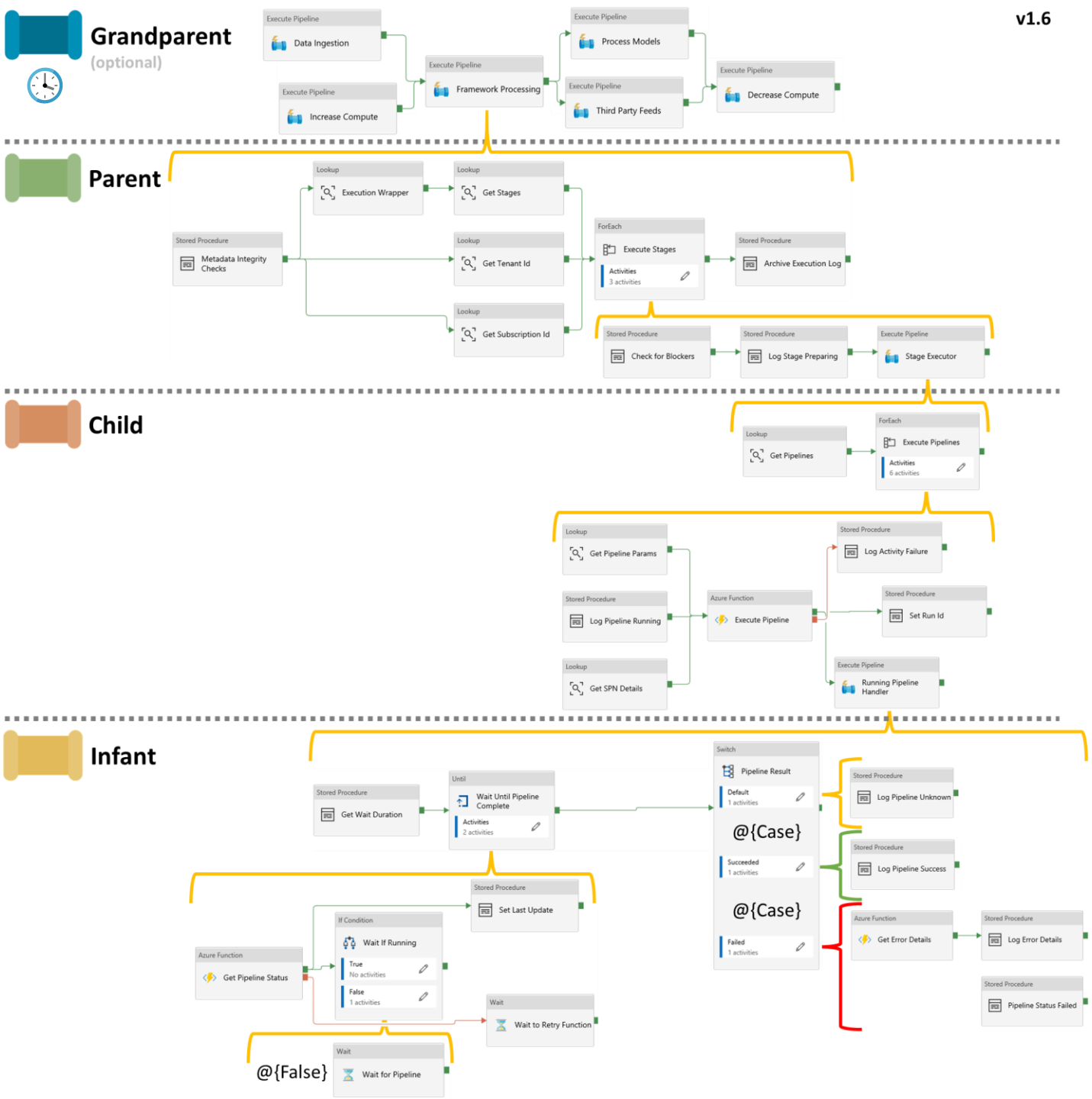


- Worker

https://mrpaulandrew.com/2019/09/25/azure-data-factory-pipeline-hierarchies-generation-control/

Data Factory





Go to Visio file in GitHub:
https://github.com/mrpaulandrew/ADF.procfwk/blob/master/Images/ADFprocfwk%20Designs.vsdx

#ADFprocfwk

Framework Pipeline Hierarchy – Summary

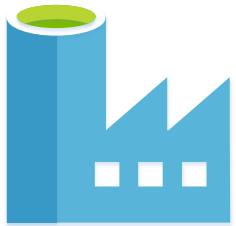
Grand-parent
- Setup Environment
- Attached Trigger(s)

Parent
- Bootstrap Execution (New/Restart)
- Call Processing Stages Sequentially

Child
- Bootstrap Processing Stage
- Call Worker Pipelines in Stage in Parallel

Infant
- Monitor Worker Pipelines
- Log Failure Details

Worker
- Solution Specific Activities
- Code Scoped Outside the Framework



Framework Factory

Worker Factory

https://mrpaulandrew.com/category/azure/data-factory/adf-procfwk/

Thank you for listening…

Paul Andrew

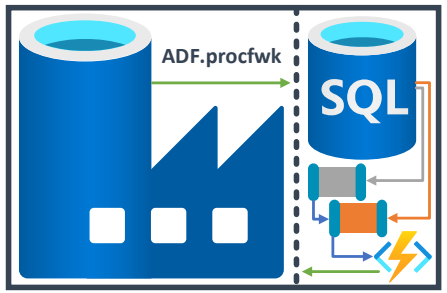
Microsoft® MVP Most Valuable Professional

altius

Blog: mrpaulandrew.com
Email: paul@mrpaulandrew.com

Twitter: @mrpaulandrew
LinkedIn: In/mrpaulandrew

GitHub: github.com/mrpaulandrew



ADFprocfwk.com

#ADFprocfwk

v1.6
Released Today

PDF

Slides:
Community Events
Repository