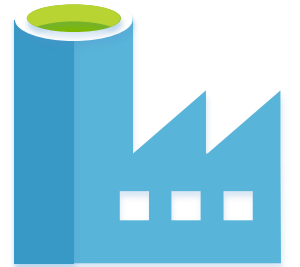


Creating a Metadata Driven Processing Framework

For Azure Data Factory



Paul Andrew | Principal Consultant & Solution Architect



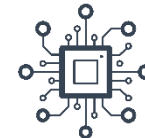
altius



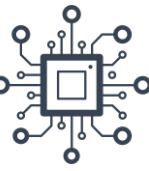
@MrPaulAndrew



In/MrPaulAndrew



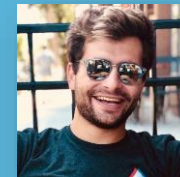
MrPaulAndrew.com



Paul Andrew
@mrpaulandrew

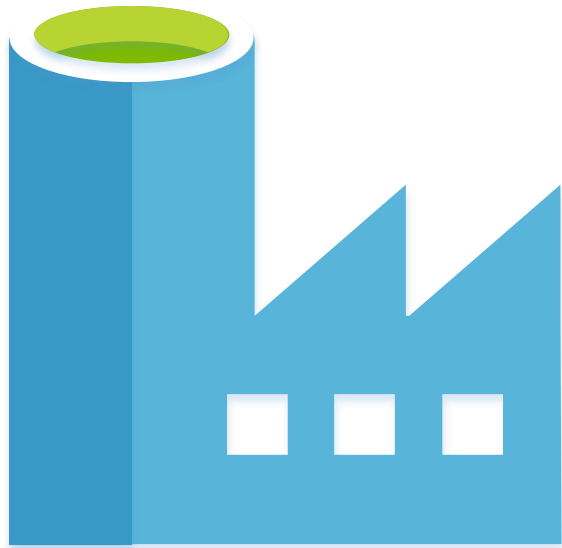


Daniel Perlovsky
@big_data_dan

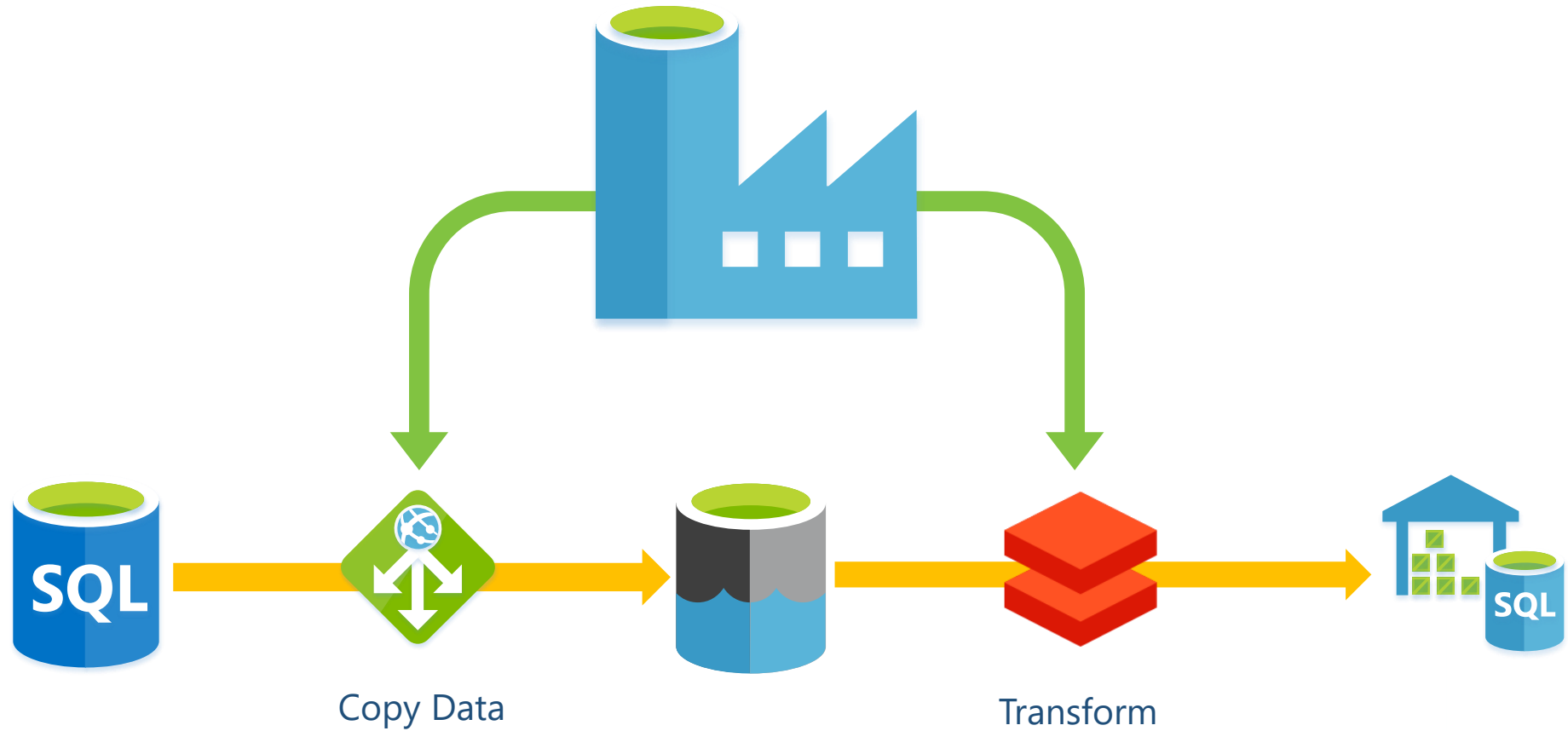


Mark Kromer
@KromerBigData

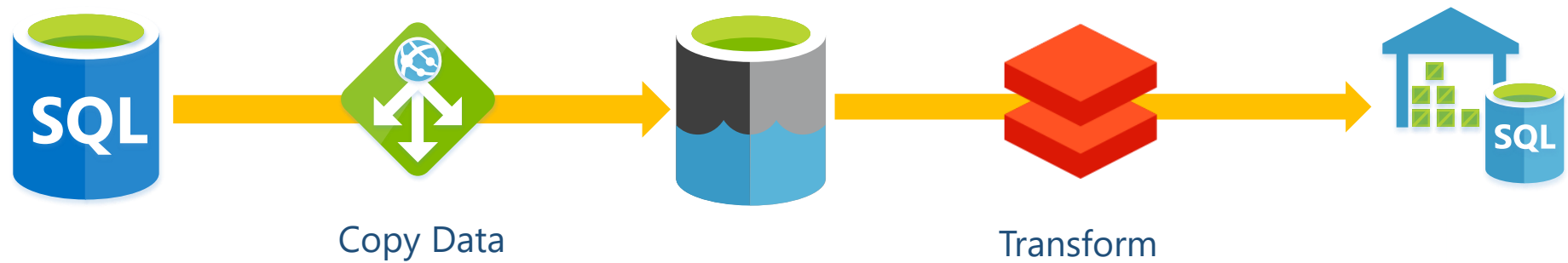
Azure Data Factory



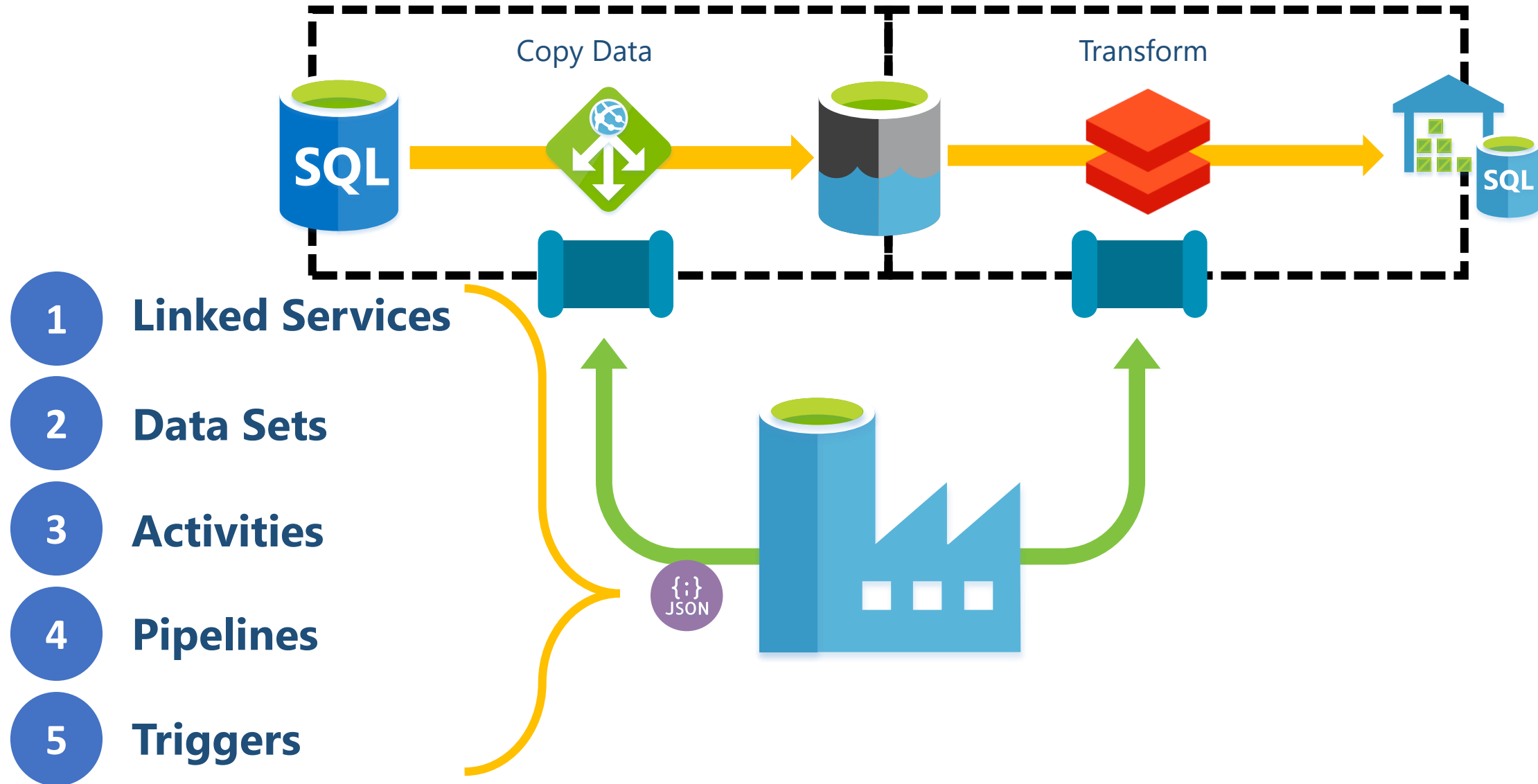
What is Azure Data Factory?



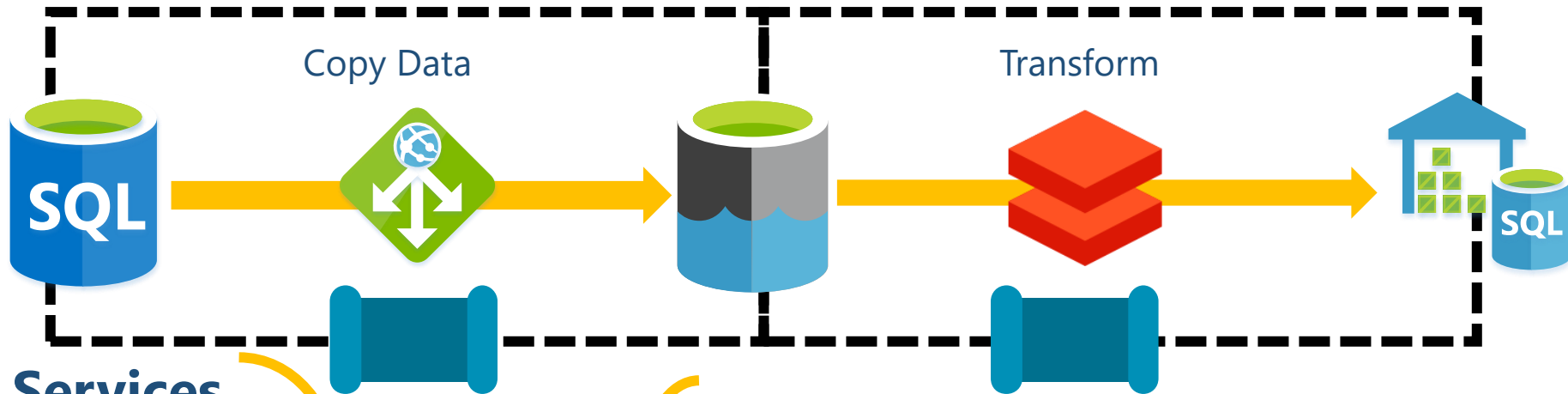
What is Azure Data Factory?



Data Factory Components



Data Factory Components



1 ✓ **Linked Services**

2 ✓ **Data Sets**

3 ✓ **Activities**

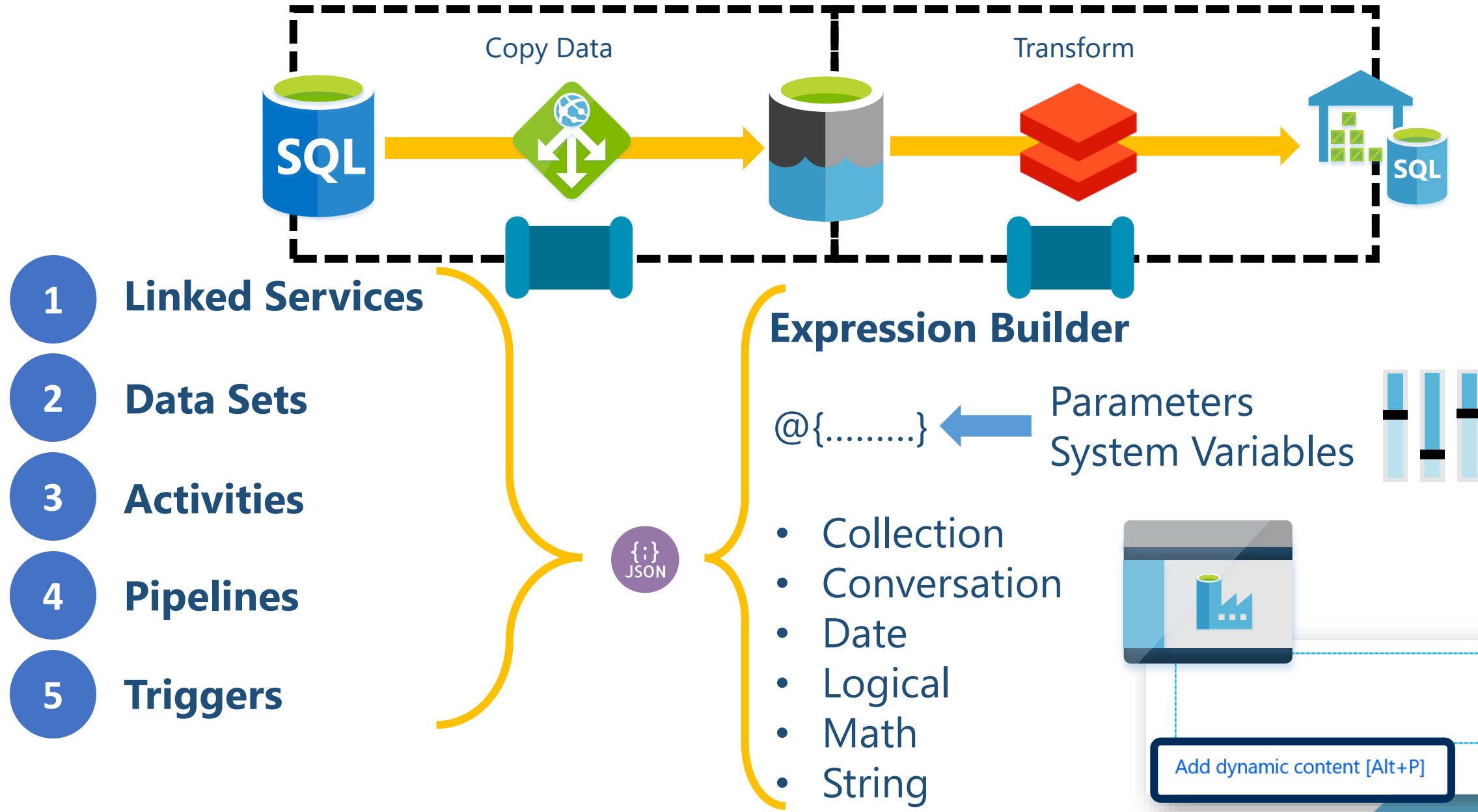
4 ✓ **Pipelines**

5 ✗ **Triggers**

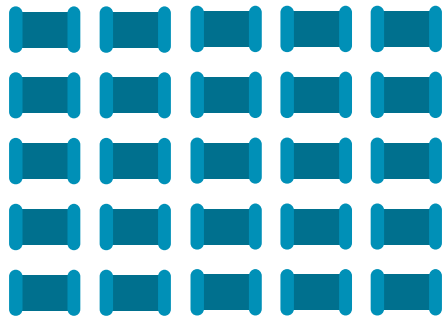
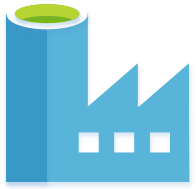
{:}
JSON

```
{
  "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}"
    }
  }
}
```

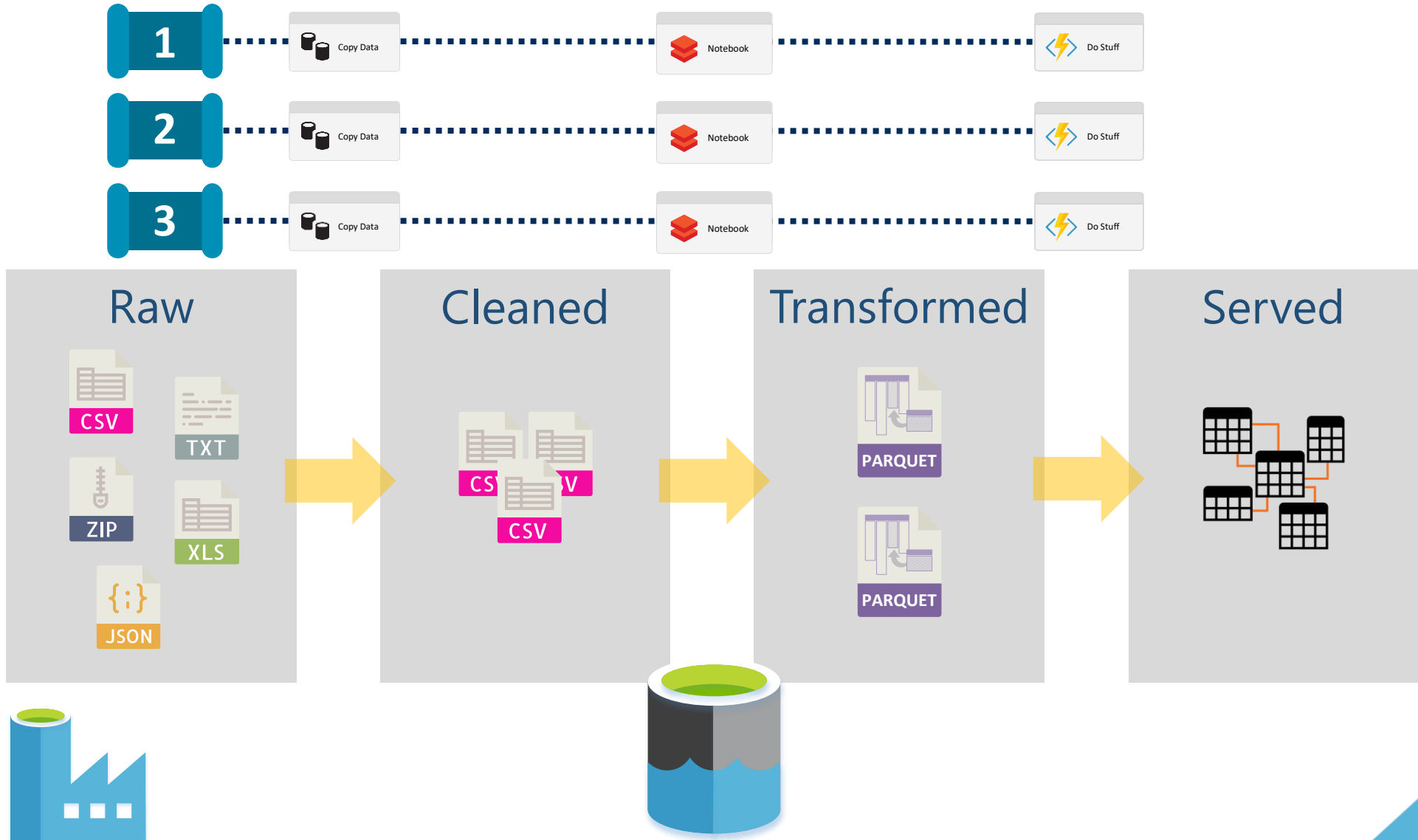
Data Factory Components



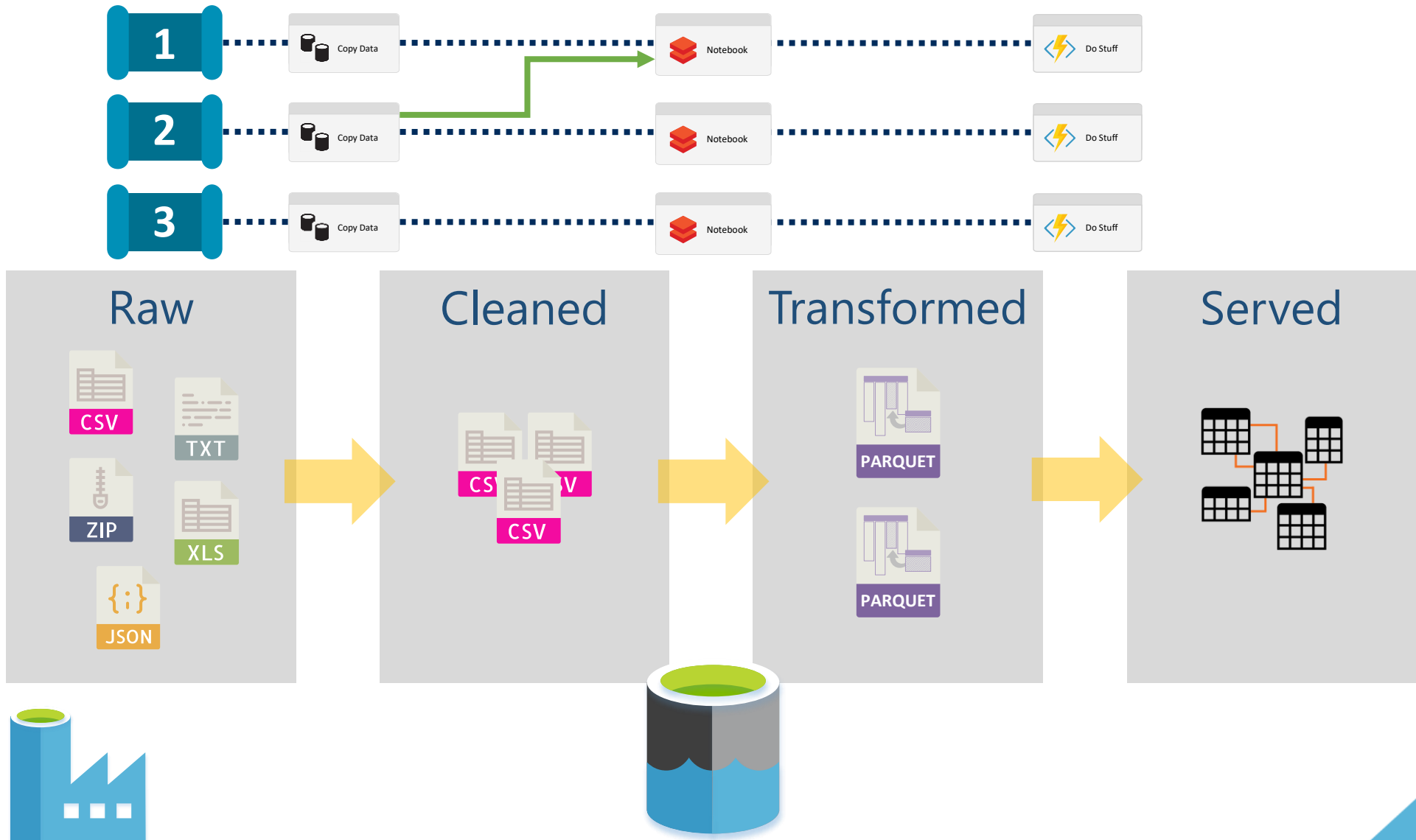
Problem: How do we structure our Data Factory Pipelines?



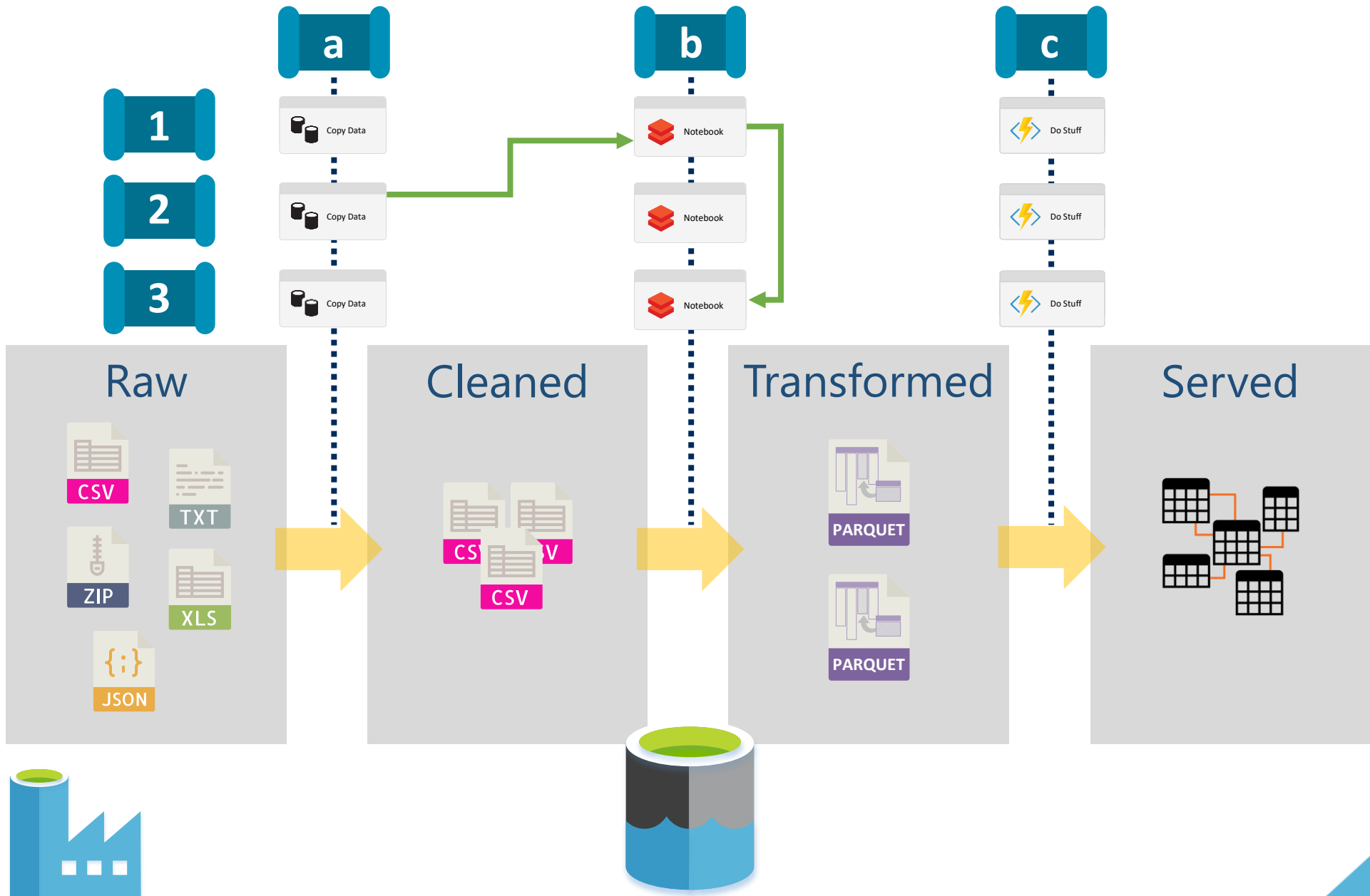
Problem



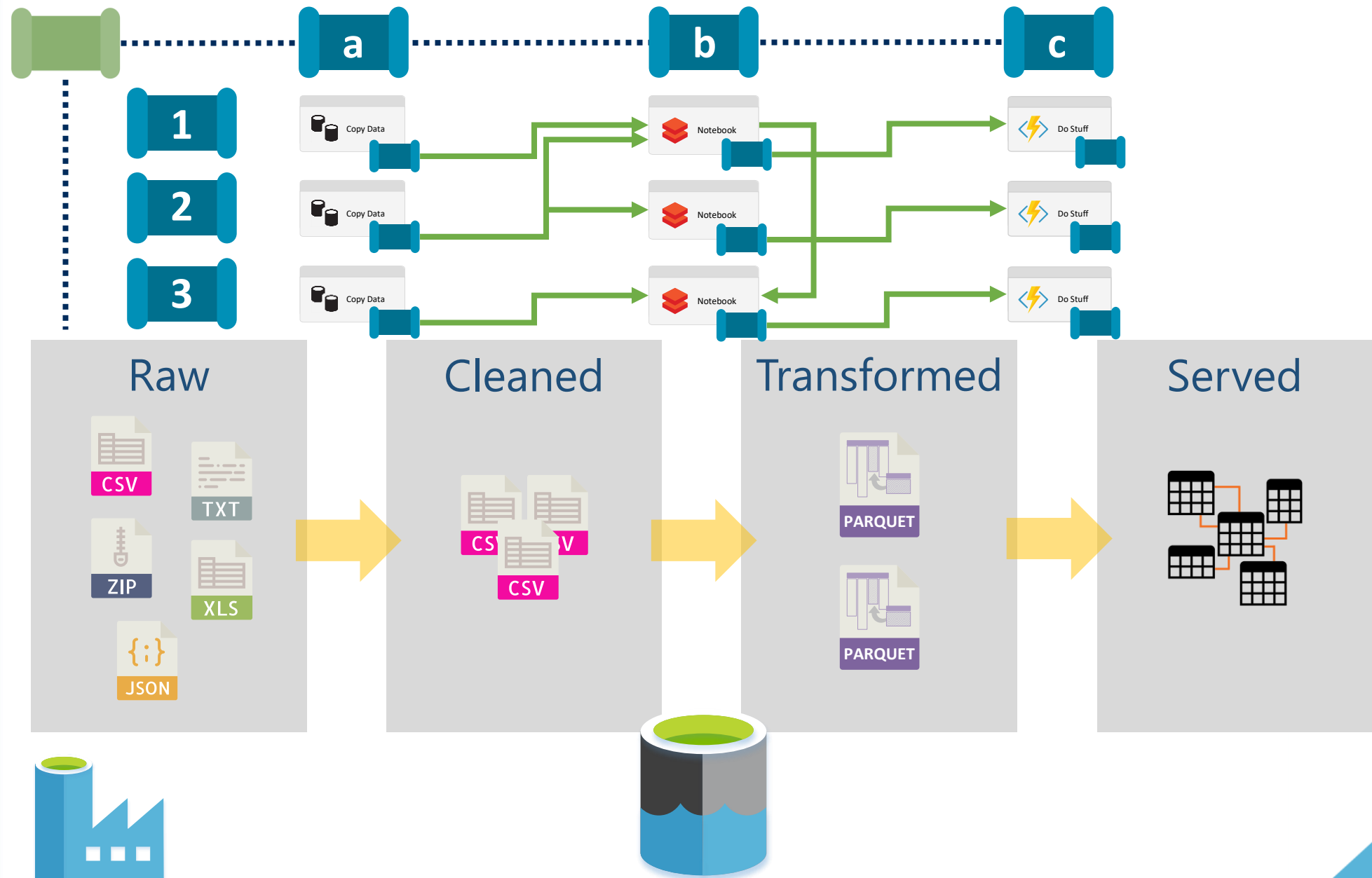
Problem



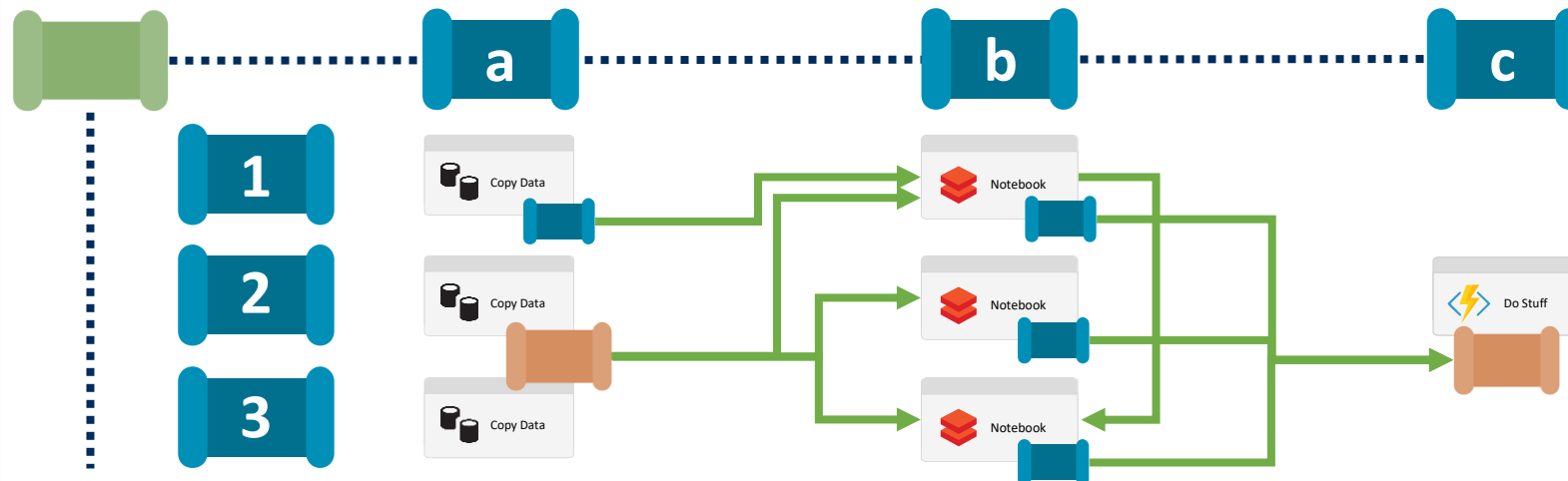
Problem



Problem

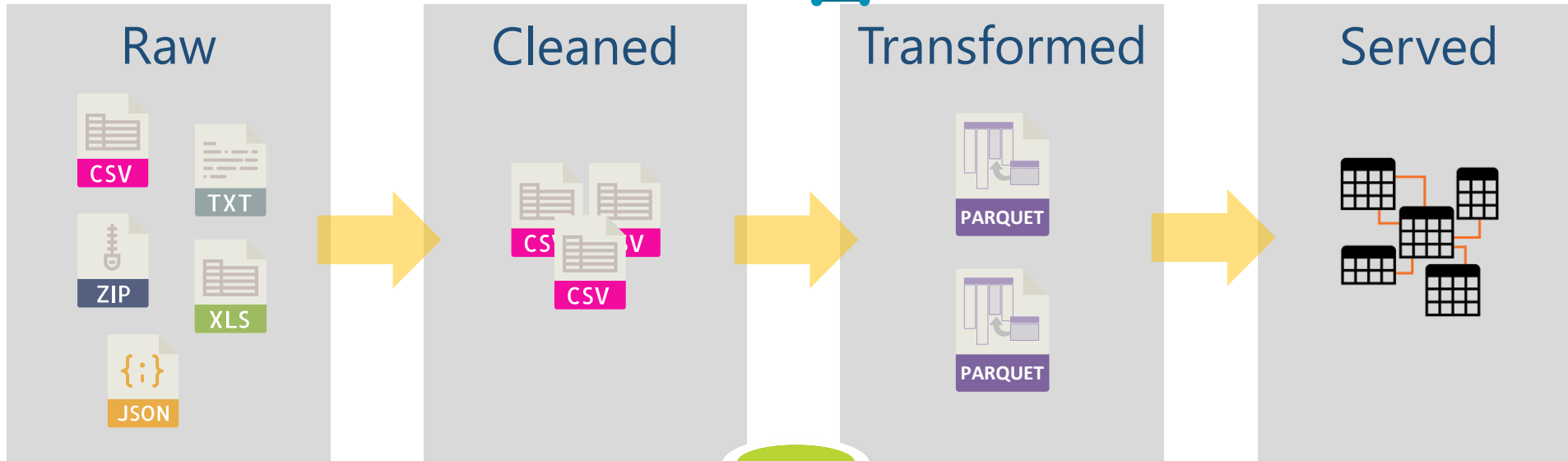


Problem

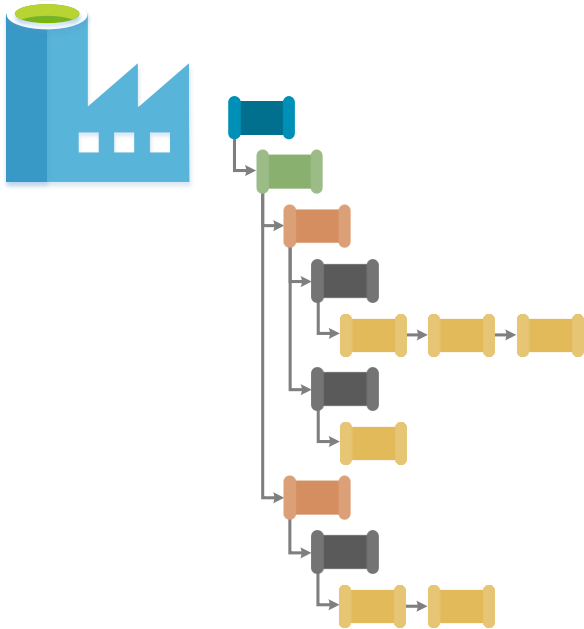


Only 40 Activities
per Pipeline.

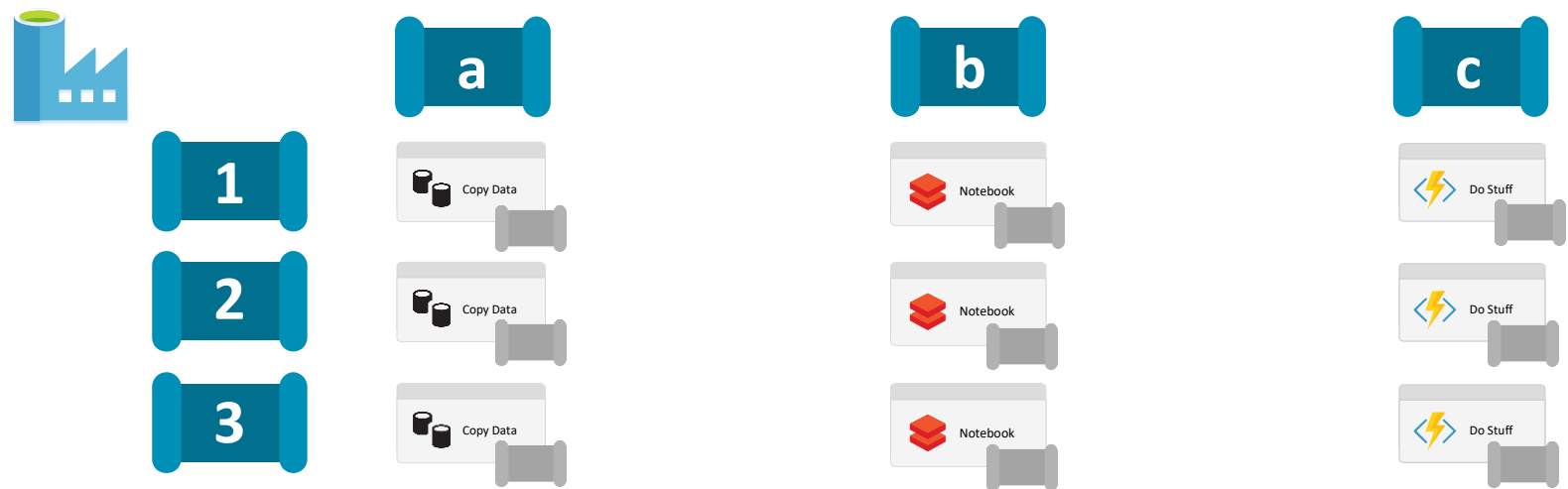
- Grandparent pipeline for all processing.
- Parent pipeline to consolidate work.
- Child pipelines for low level dependencies.



Solution: Use Metadata to Drive Data Factory Pipelines



Solution

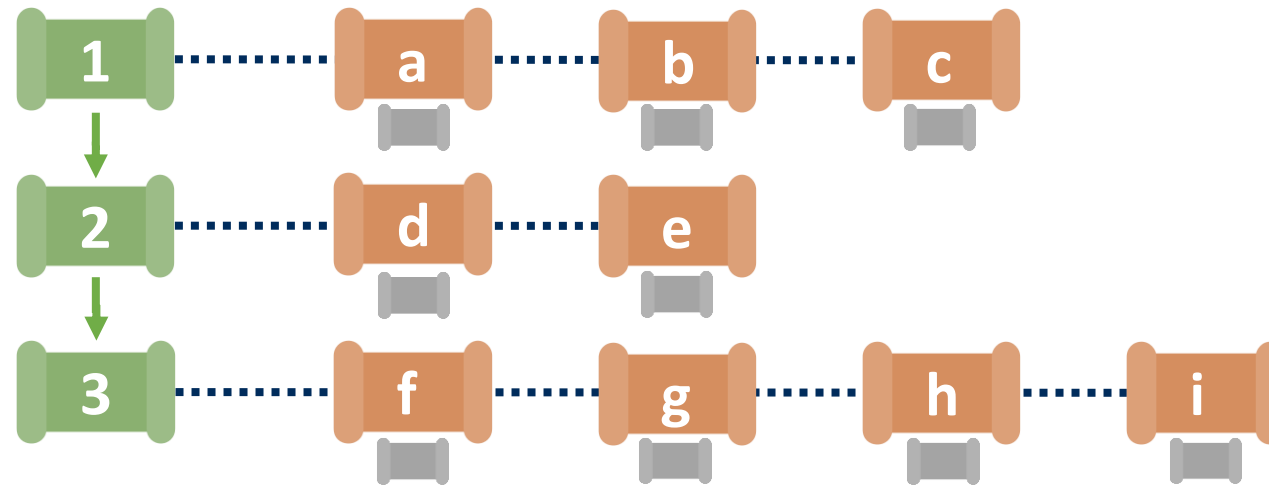


Stages
1
2
3

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

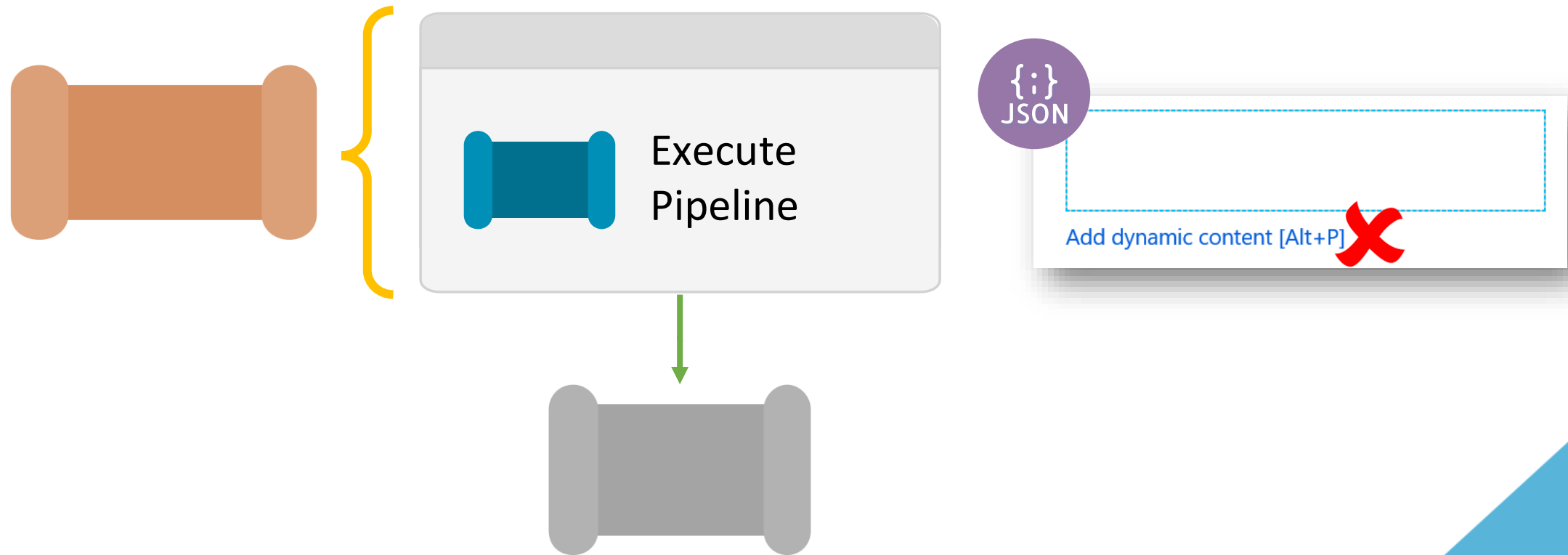


Stages
1
2
3

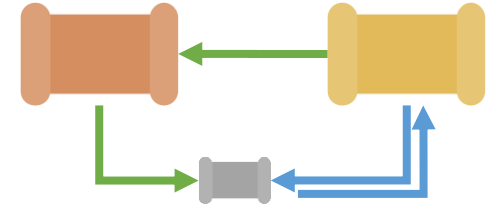
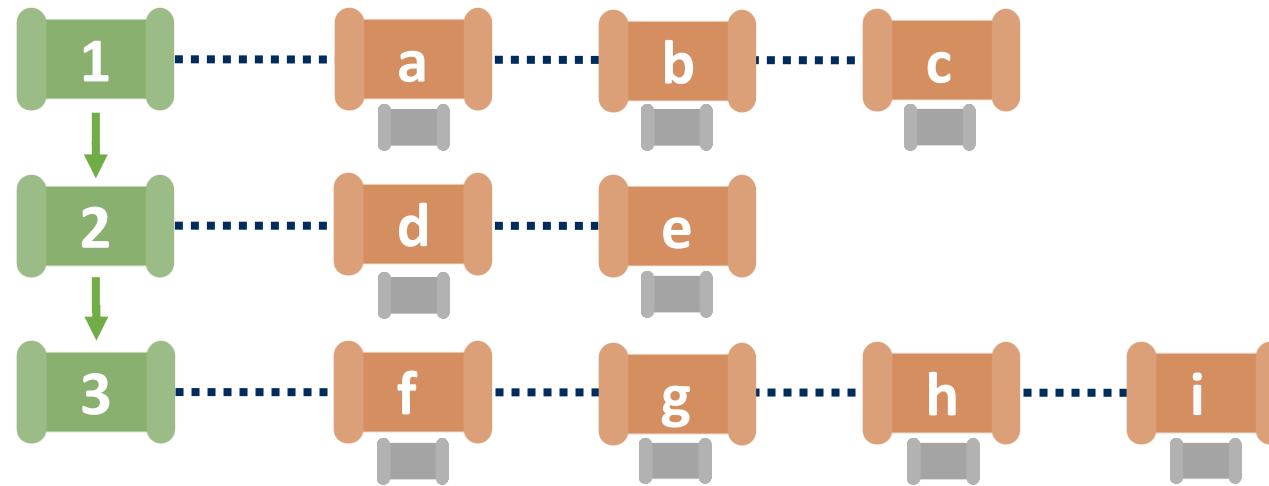
Pipelines
a
b
c
d
e
f
g
h
i

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

Execute Pipeline Activity



Solution

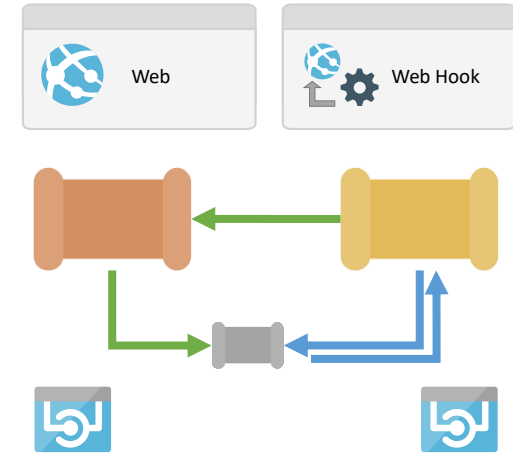
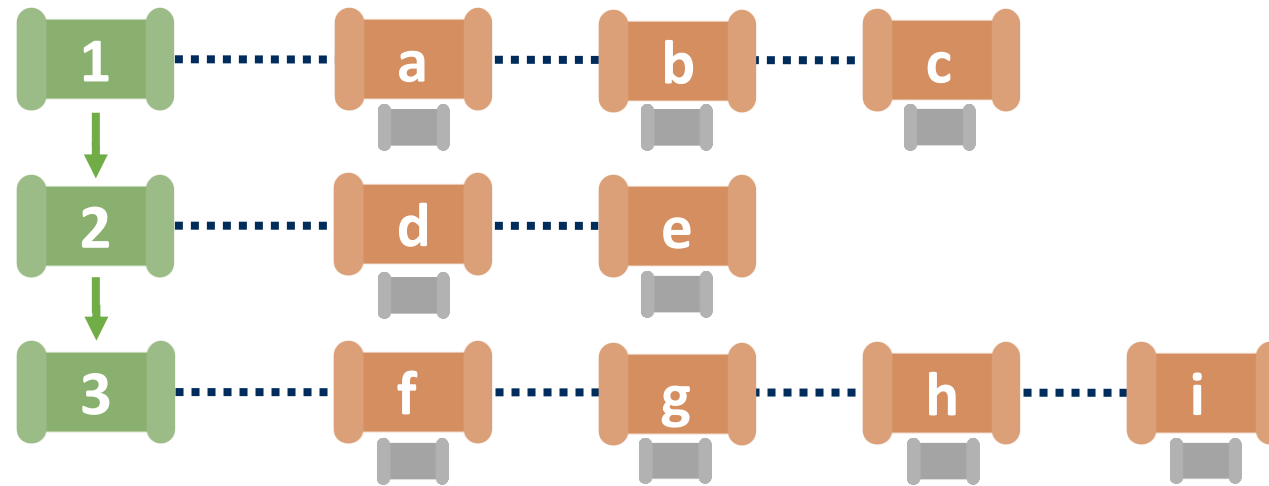


Stages
1
2
3

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

Solution – Option 1

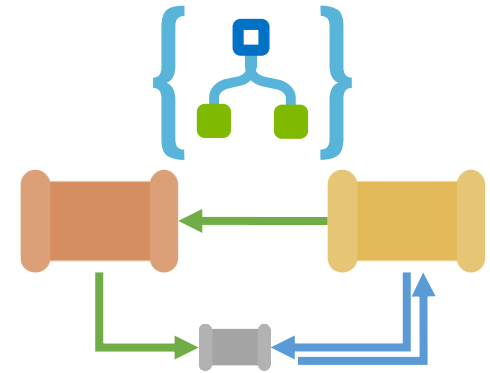
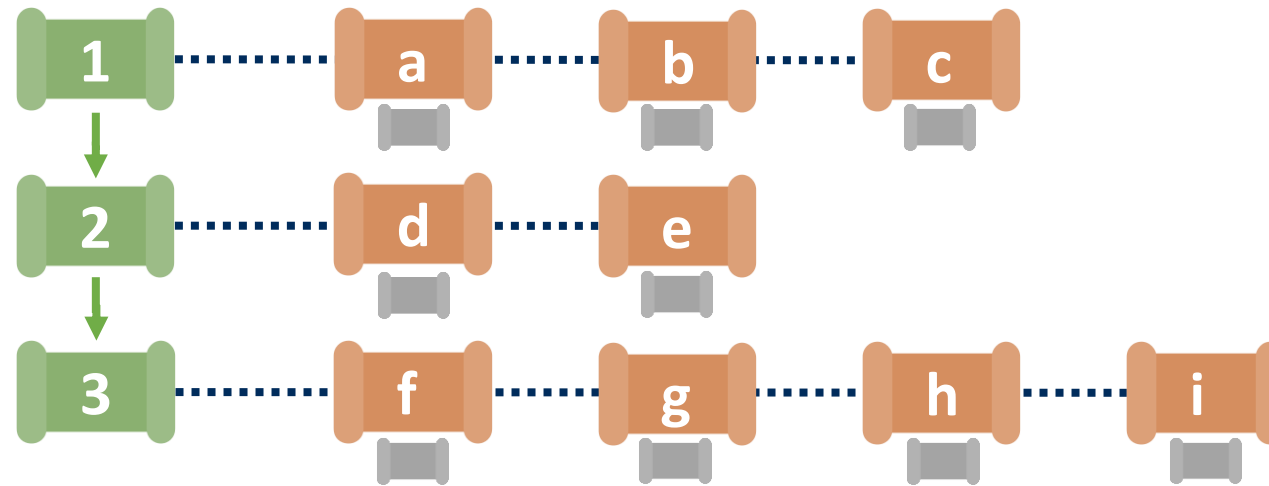


Stages
1
2
3

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

Solution – Option 2

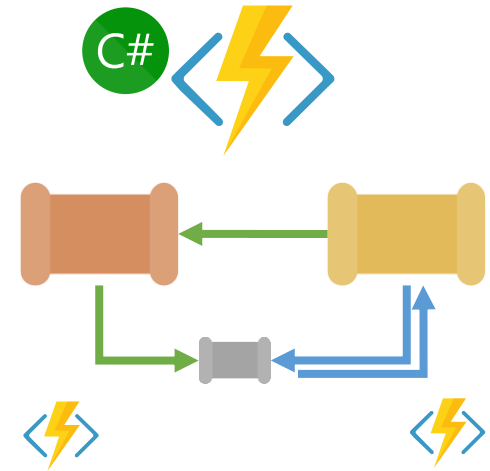
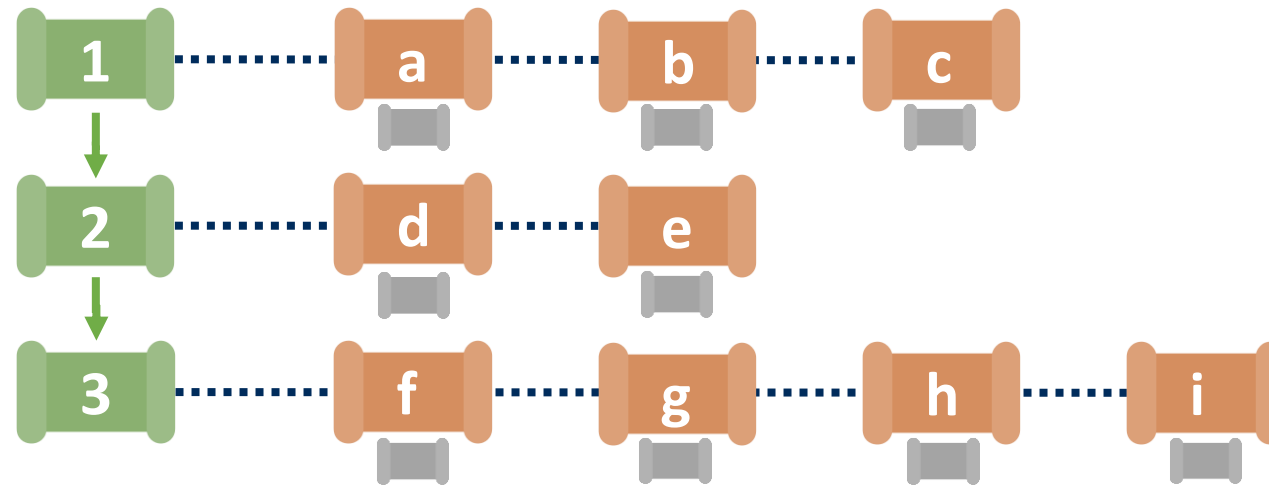


Stages
1
2
3

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

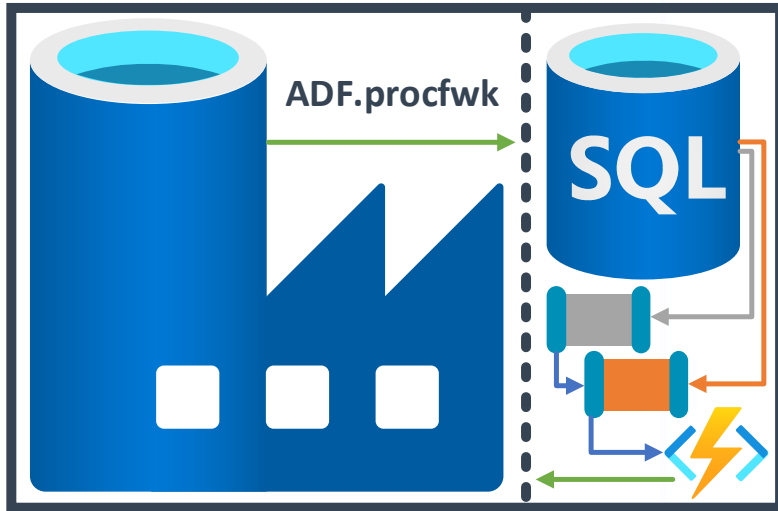


Stages
1
2
3

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

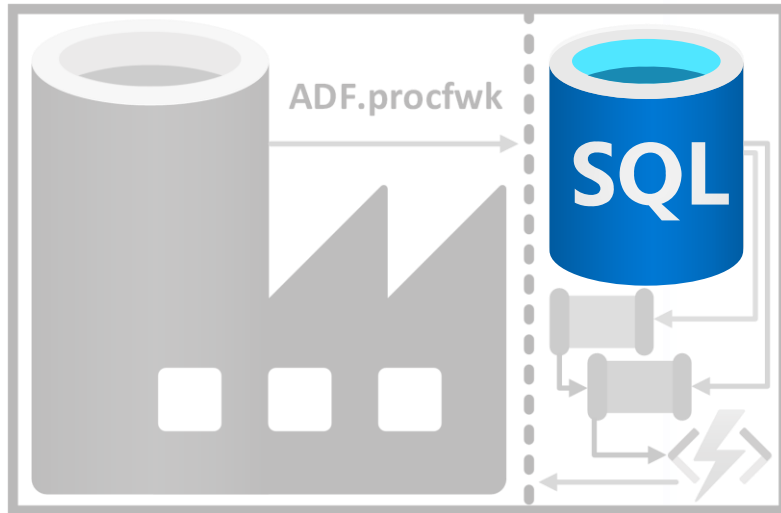
Introducing ADF.procfwk



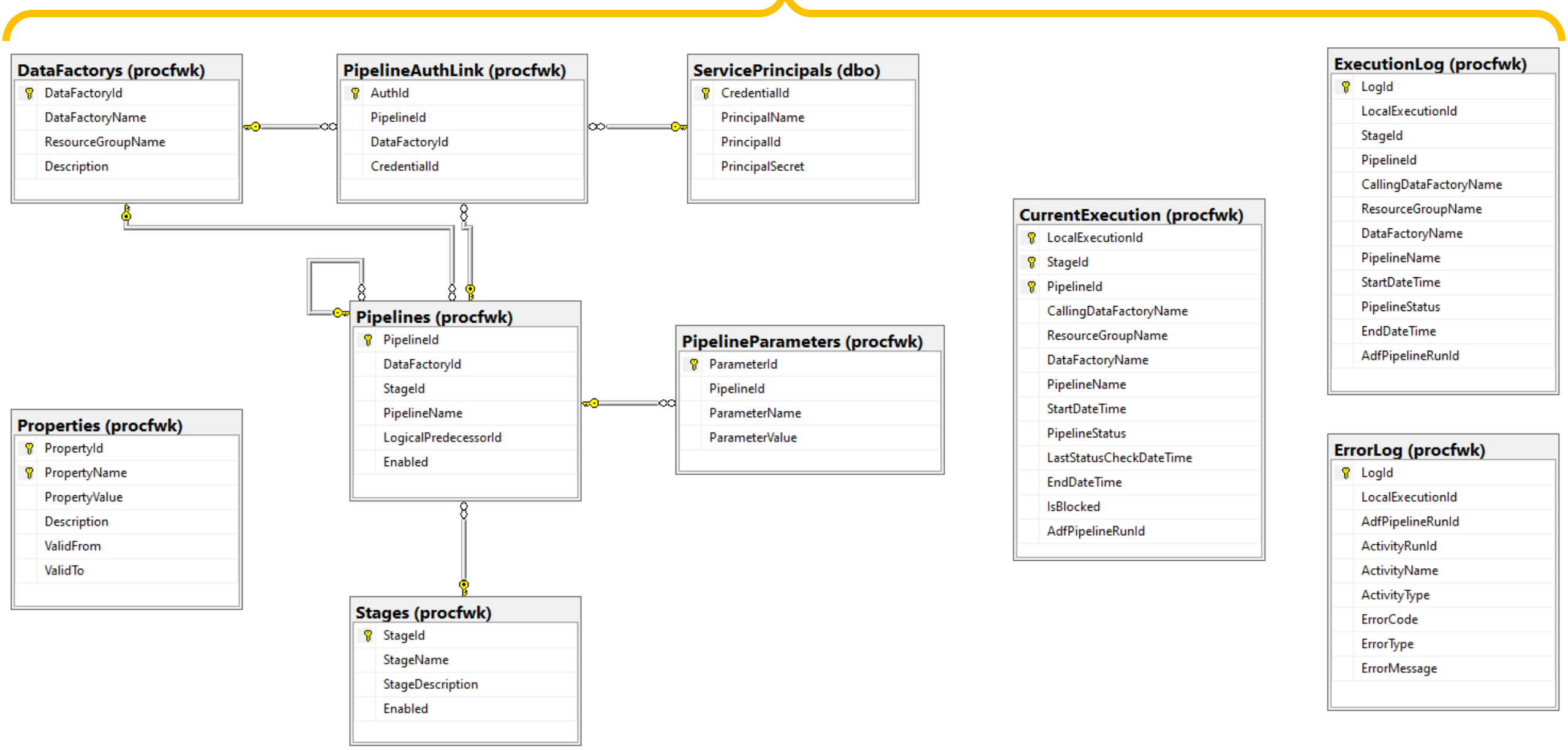
Key Features

- 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.
- Optional disconnection between framework and Worker pipelines.
- Cross Data Factory control flows.
- Pipeline parameter support.
- Simple troubleshooting.
- Easy deployment.
- Email alerting (*coming soon*).

ADF.procfwk – Database

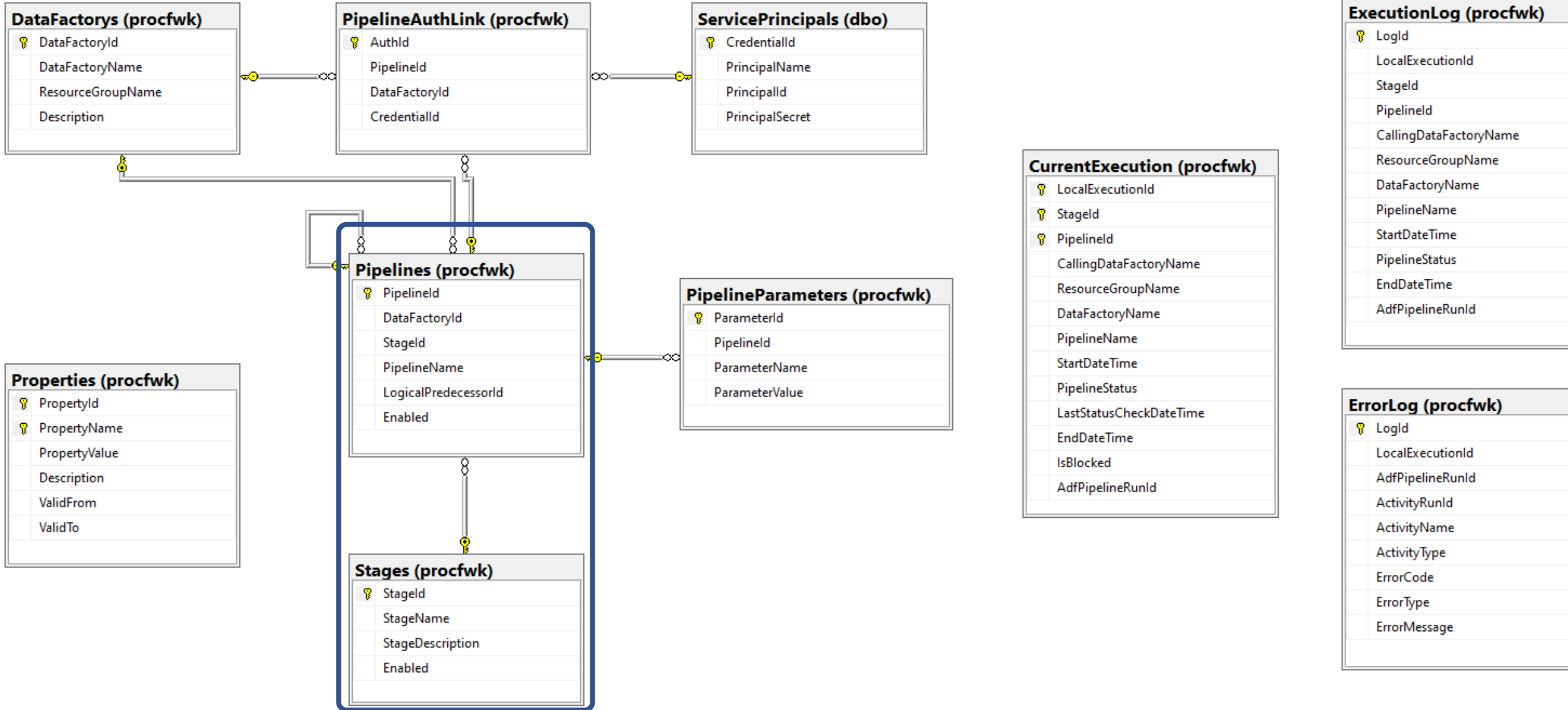


Database



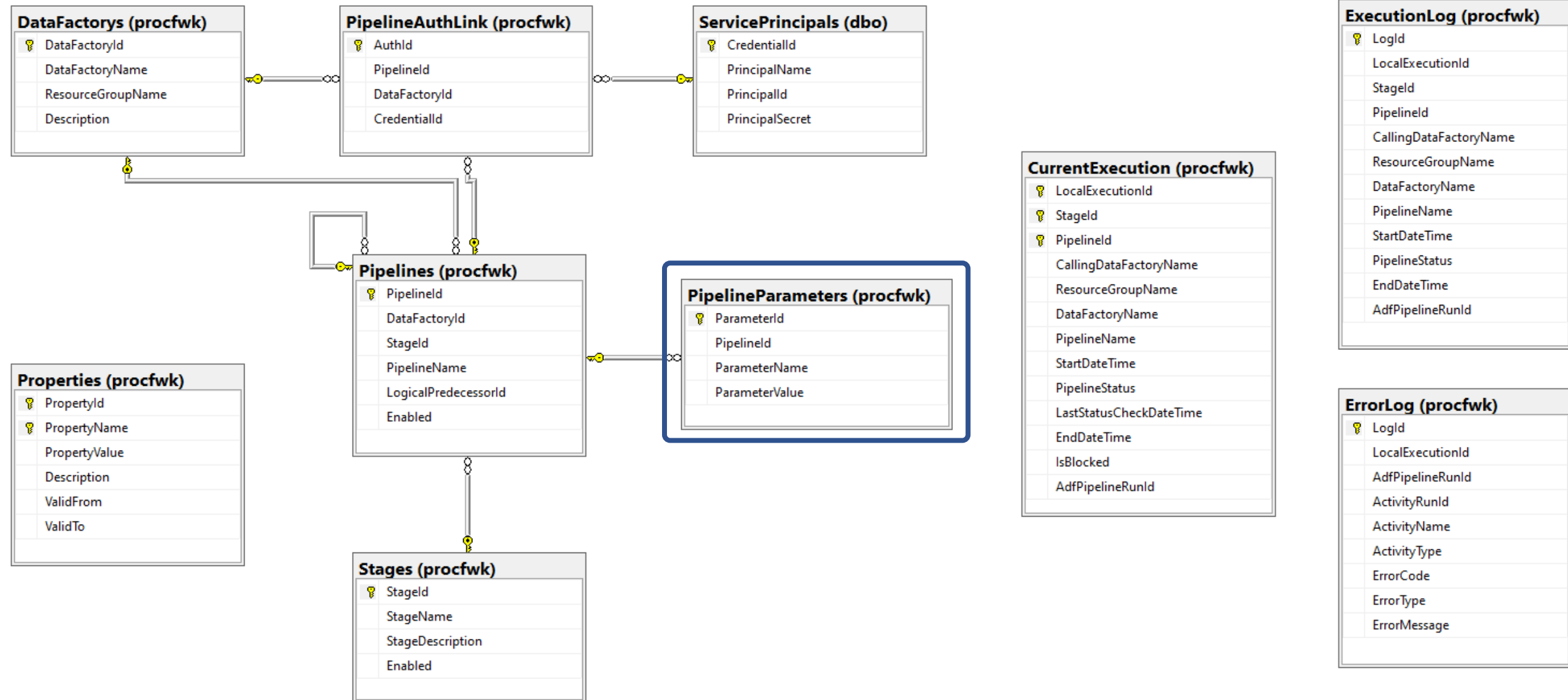
Database –

Execution Core Metadata



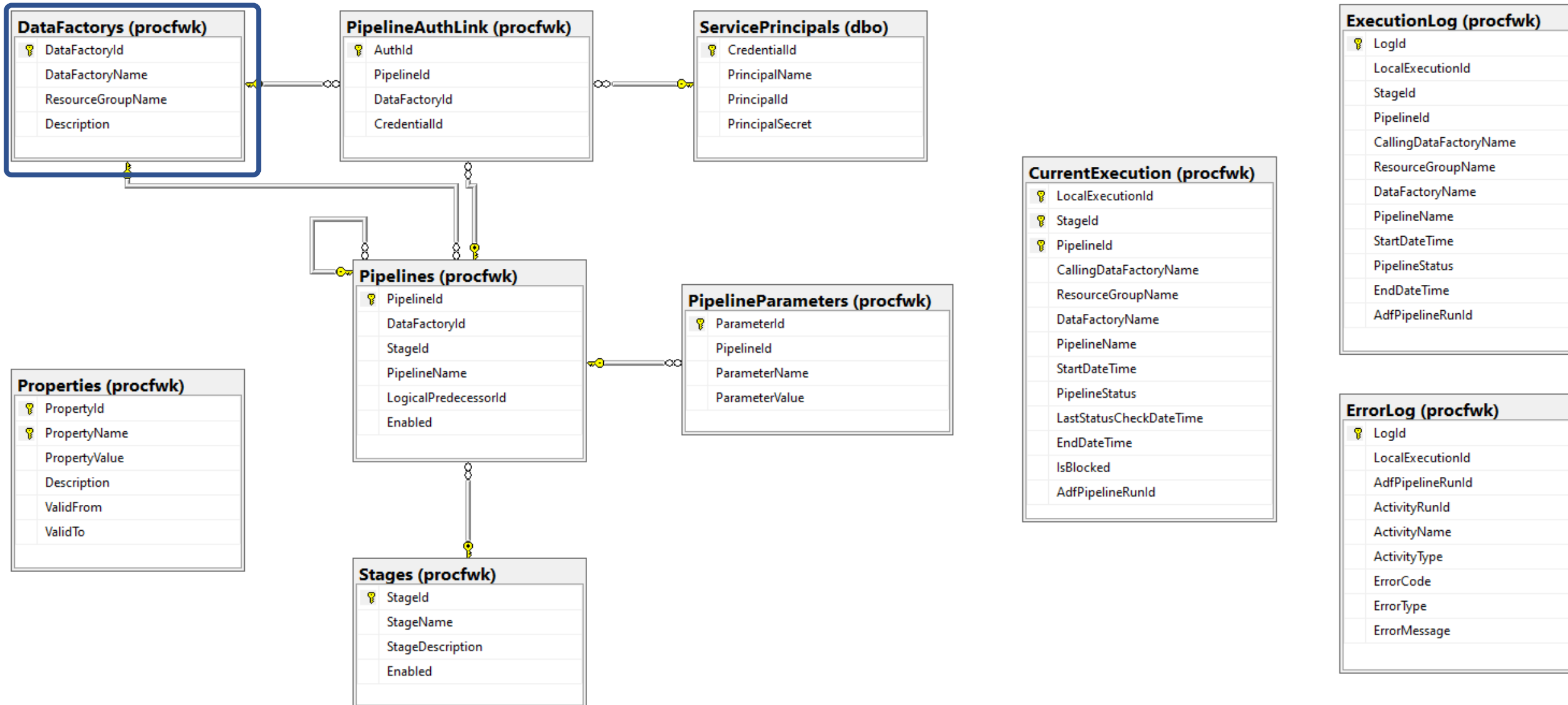
Database –

Pipeline Parameters

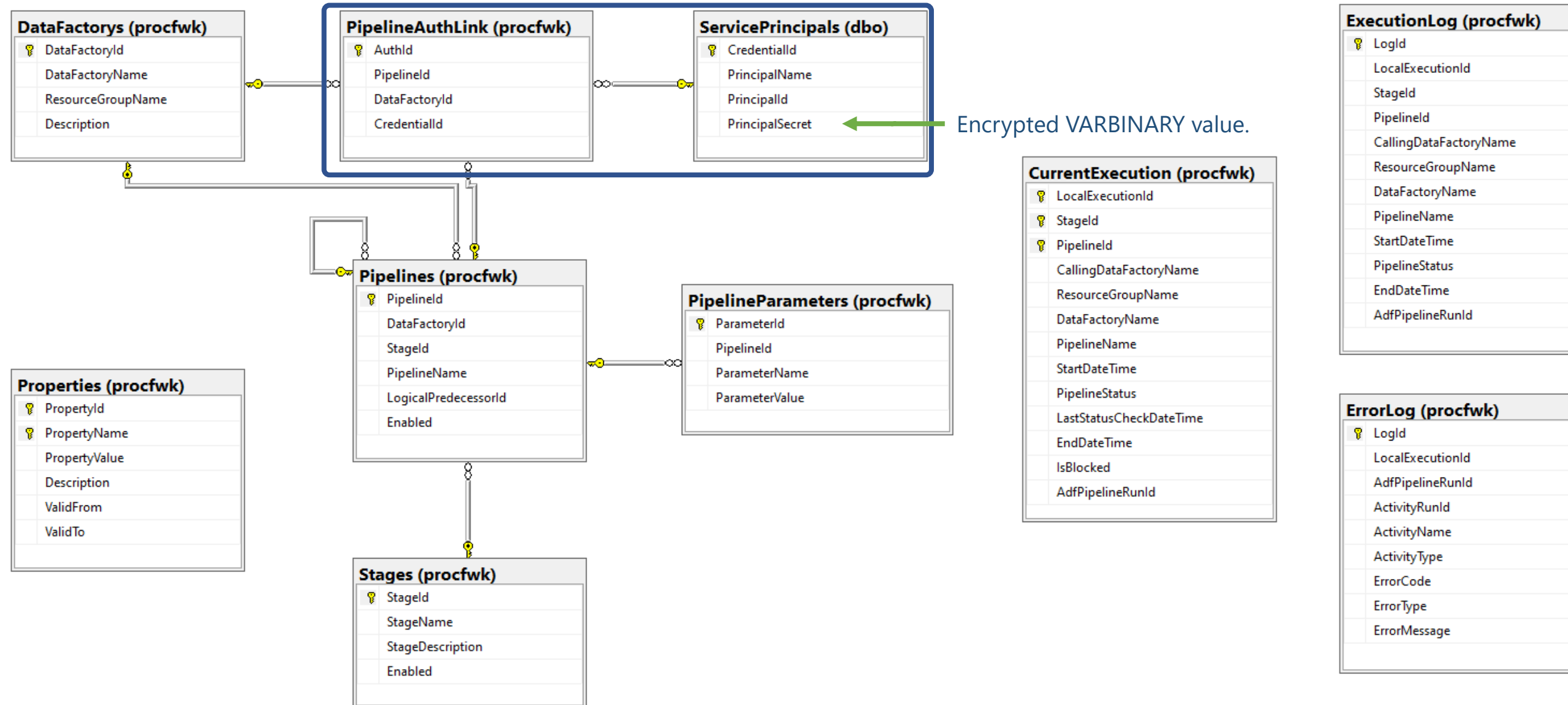


Database –

Data Factory's and Resource Groups

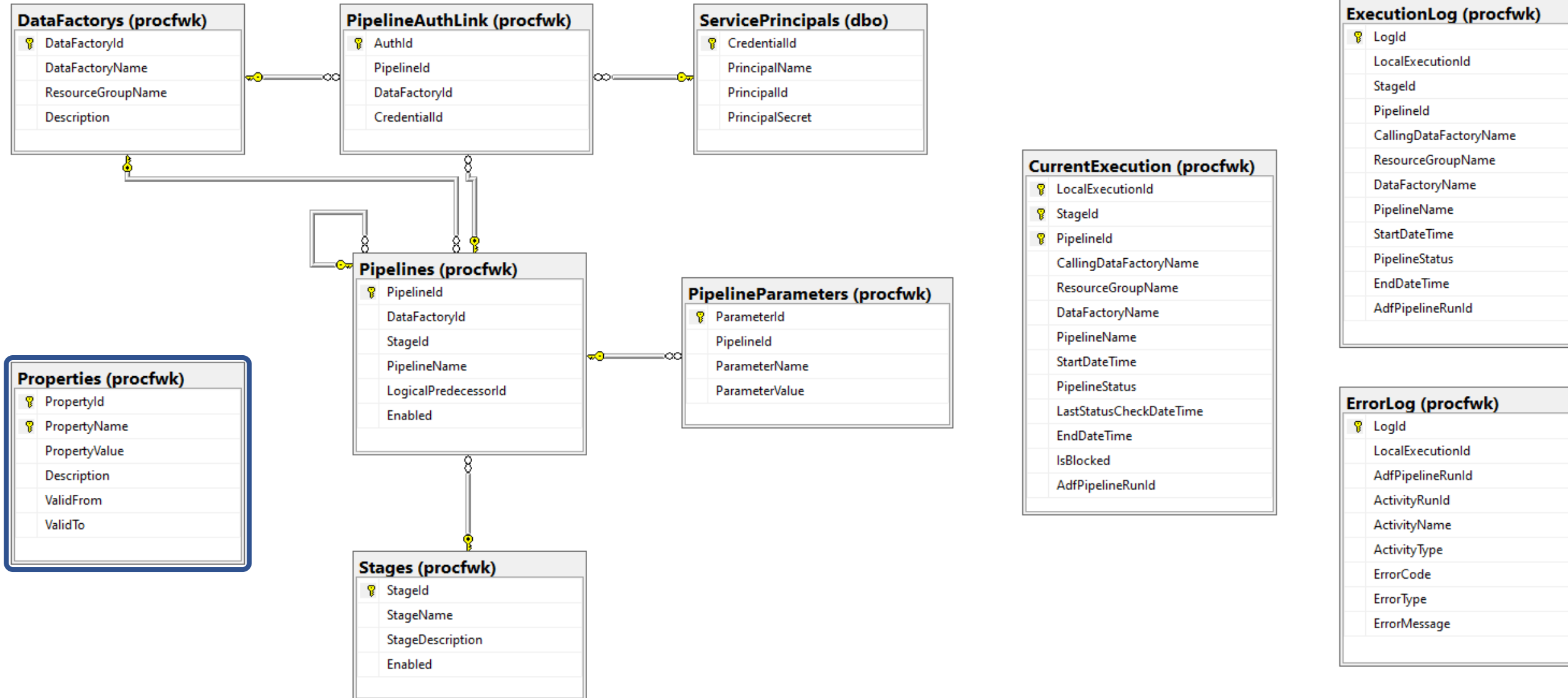


Database – Authentication



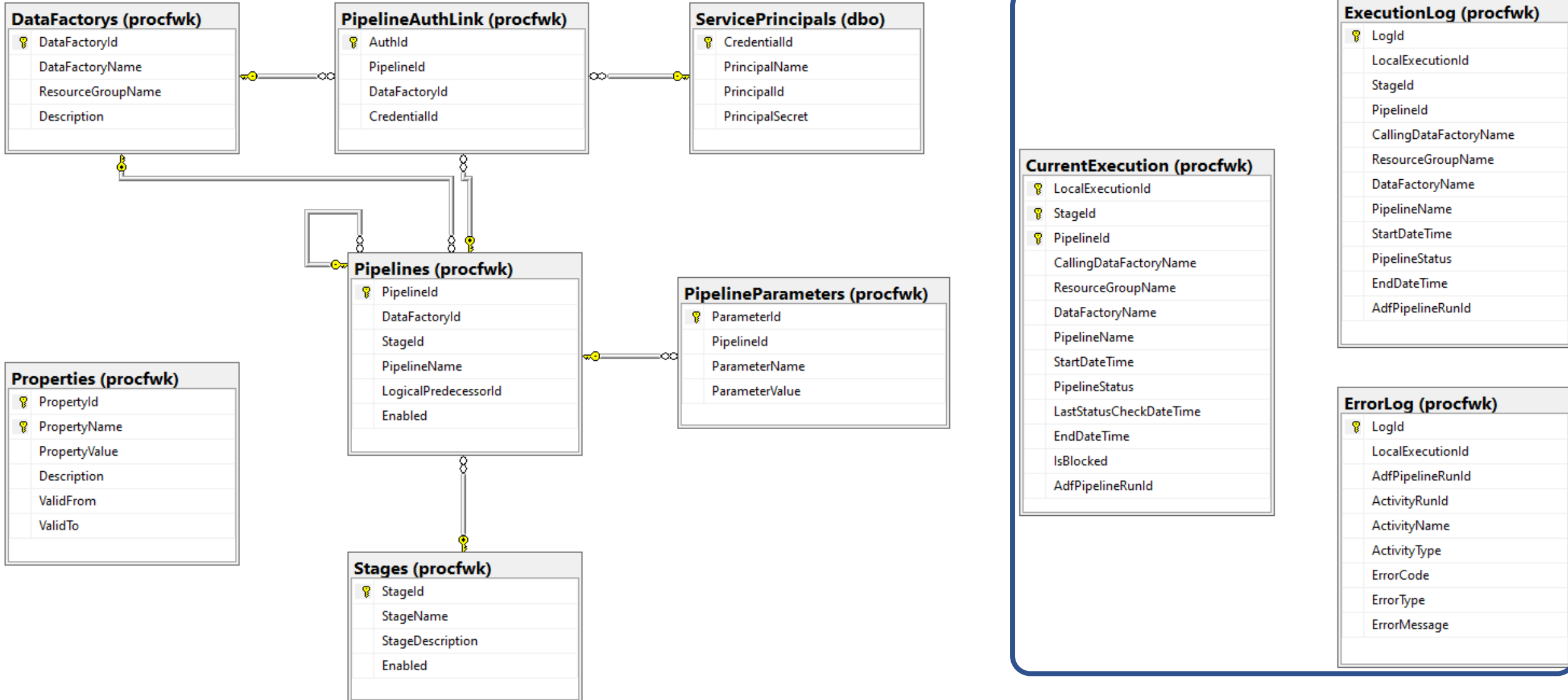
Database –

Global Controls

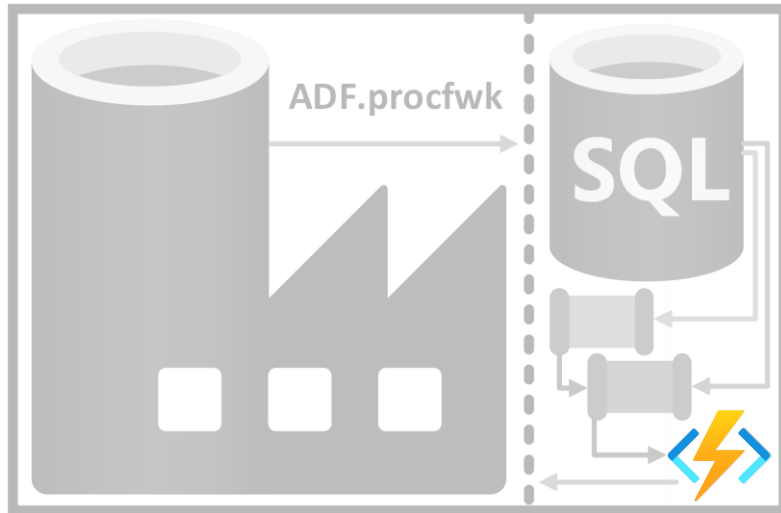


Database –

Runtime Logs



ADF.procfwk - Functions



Functions



Execute Pipeline



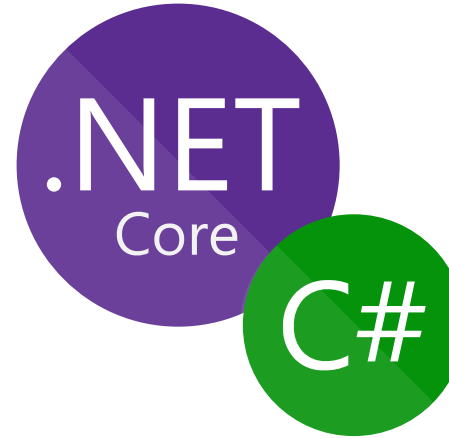
Check Pipeline Status



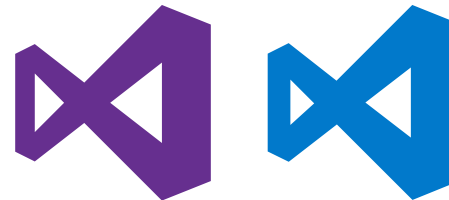
Get Activity Errors



Send Email



Created using C# for
.Net Core 3.1

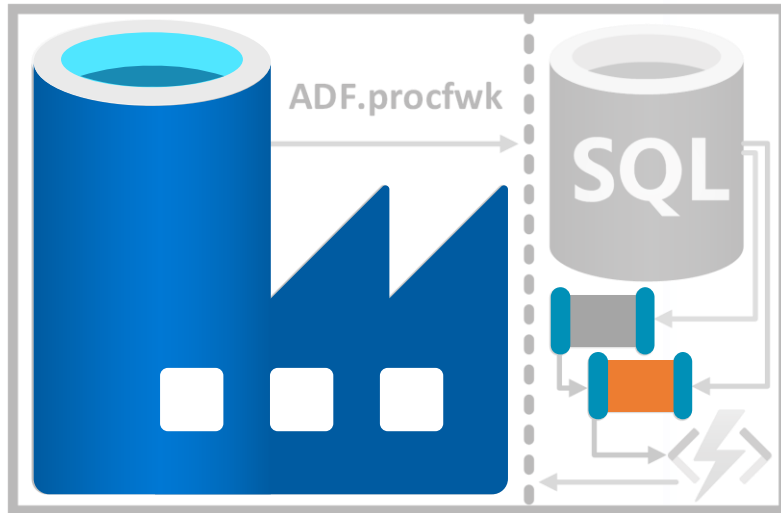


Debug via Visual
Studio or VSCode



Consumption Tier
App Service Plan

ADF.procfwk – Data Factory



The Pipelines



- Grandparent



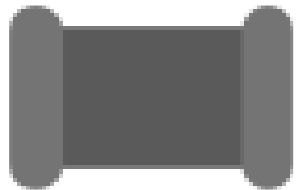
- Parent



- Child



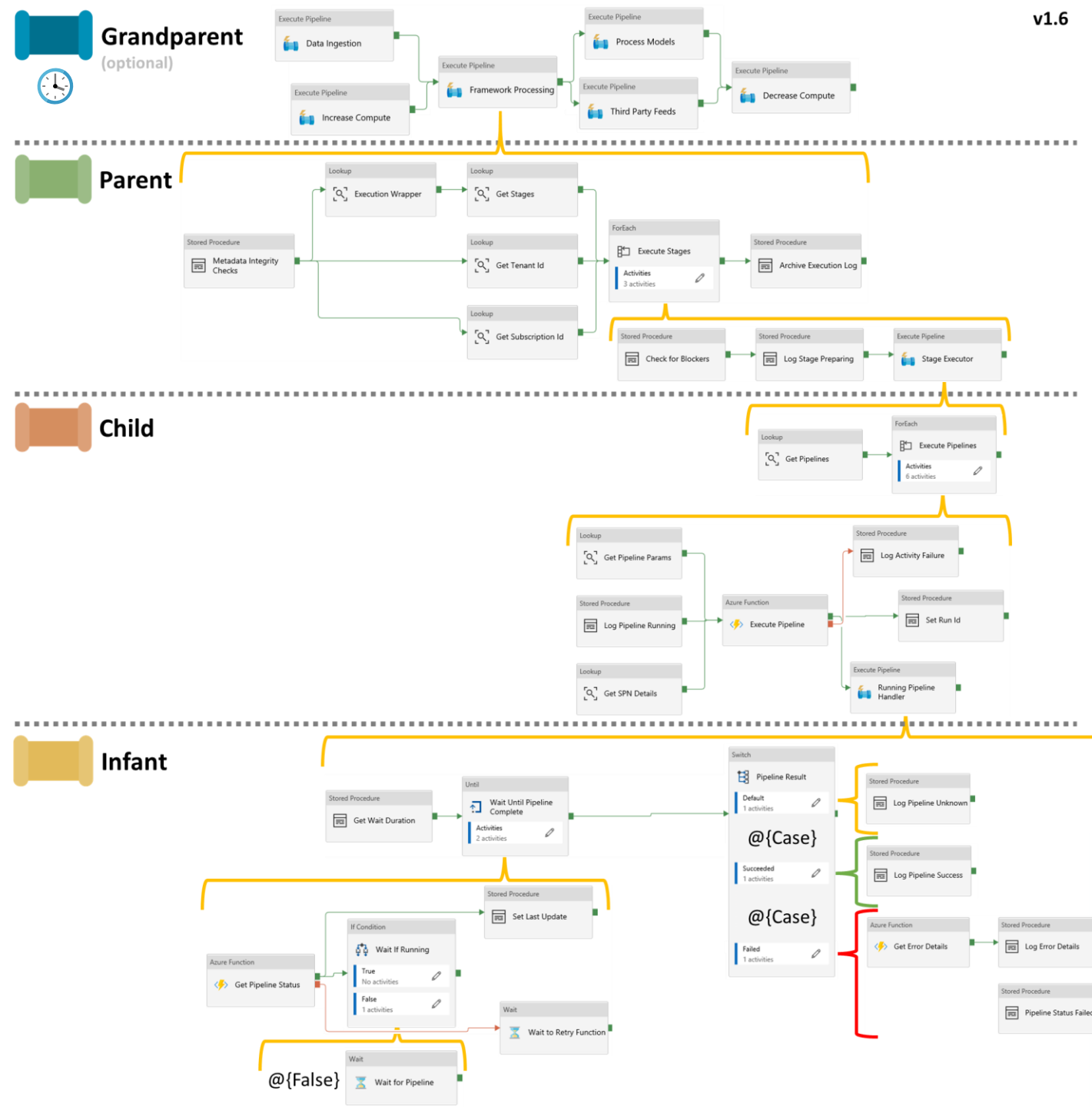
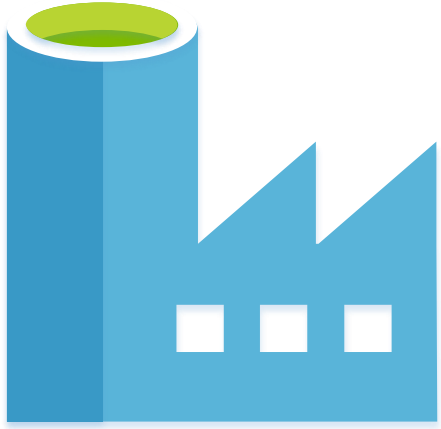
- Infant



- Worker



Data Factory



Go to Visio file in GitHub:
<https://github.com/mrpaulandrew/ADFprocfwk/blob/master/Images/ADFprocfwk%20Designs.vsd>

Framework Pipeline 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



Thank you for listening...

Paul Andrew

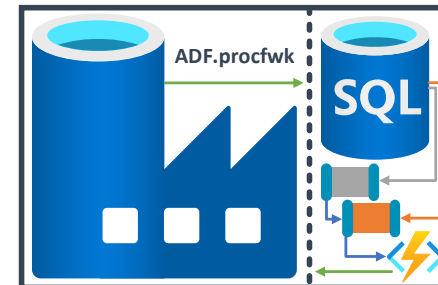


altius

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

Twitter: @mrpaulandrew
LinkedIn: In/mrpaulandrew

GitHub: github.com/mrpaulandrew



ADFprocfwk.com

#ADFprocfwk

v1.6

Now Available



Slides:
[Community Events](#)
Repository