
AWS
re:Invent
DECEMBER 2 – 6, 2024 | LAS VEGAS, NV

DAT347-NEW

Accelerate migrations using AWS DMS Schema Conversion with generative AI

John Winford

Senior Manager, Database Services
AWS

Agenda

01 Intro to AWS Database Migration Service (AWS DMS)

02 Migrations 101

03 AWS DMS Schema Conversion: A close look

04 The AI is coming

05 Bolting it together

06 Something is missing

07 Where do we go from here?

Start at the ground floor

Common customer use cases

Migrate

Migrate business-critical applications and data warehouses

Migrate from NoSQL to SQL, SQL to NoSQL, or NoSQL to NoSQL

Upgrade to a minor/major version

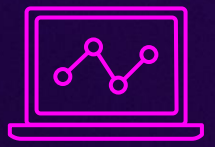
Consolidate and archive data

Replicate

Create cross-region read replicas

Hydrate your data lakes and data warehouses

Replicate to streaming platform



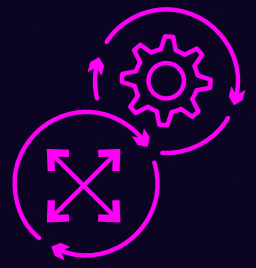
Modernize

Move to AWS modern managed databases like Amazon Aurora, Amazon DynamoDB, and Amazon Redshift for faster innovation

Break free from legacy databases to save, grow, and innovate

AWS Database Migration Service (AWS DMS)

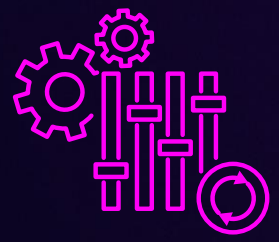
SECURELY MIGRATE YOUR DATABASES AND DATA WAREHOUSES TO AWS WITH EASE AND MINIMAL DOWNTIME



Managed migration

Automated migration service will discover, assess, convert, and migrate your database and analytics workloads to AWS

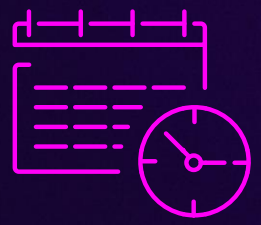
Easy to use with only a few clicks to start the process



Breadth of options

Migrate data to and from widely used commercial and open source databases

Supports SQL, NoSQL, text-based, and data warehouse targets



High availability

Multi-AZ option; highly resilient and self-healing with continuous replication and monitoring for minimal downtime

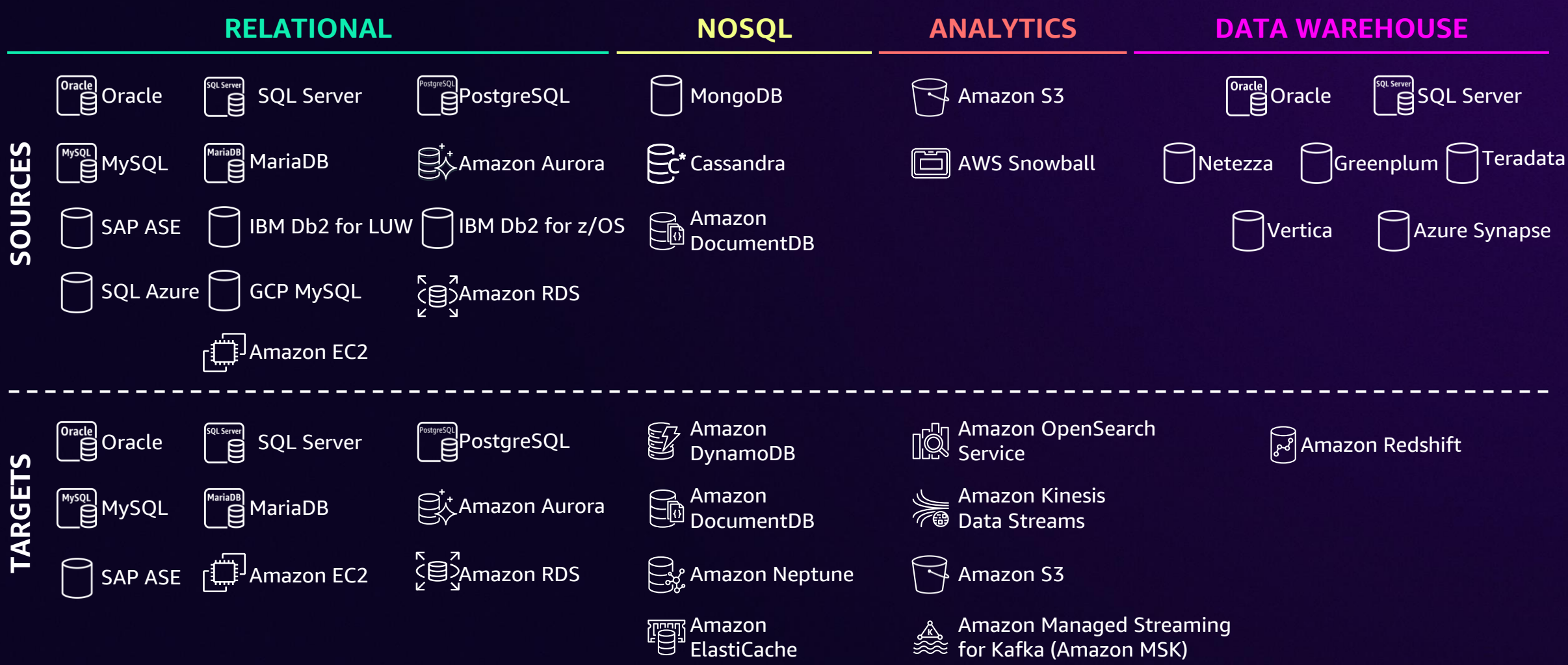
Low cost

Self-service tool

Only pay for compute resources and additional log storage used

1.25M+ databases migrated using AWS DMS

AWS DMS supported sources and targets



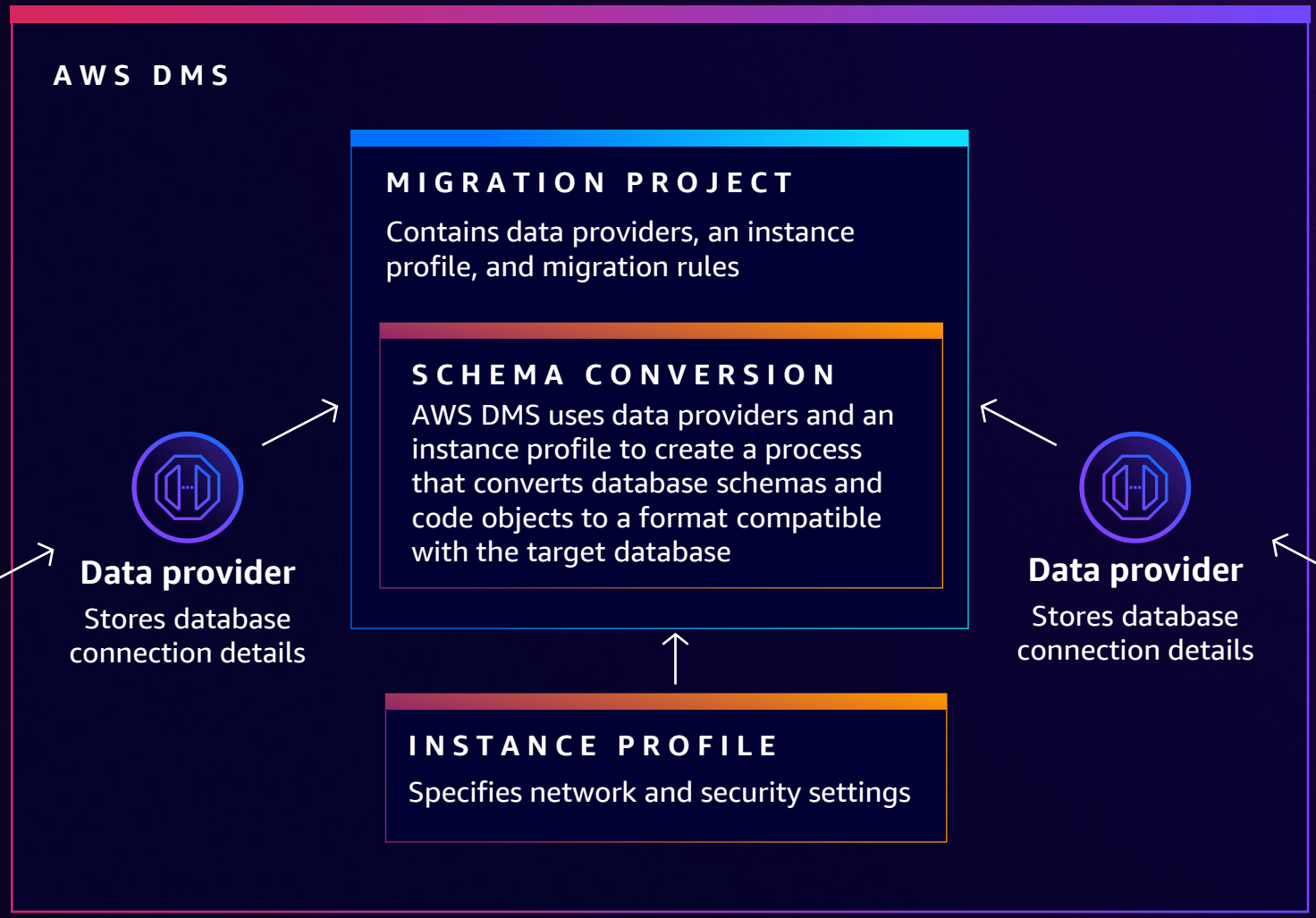
But how, Jim?

Migration basics

- Key components:
  - Database structure or schema
  - Content inside the database
- Type of migration:
  - Homogeneous
  - Heterogeneous
- Outage:
  - Online
  - Offline

DMS Schema Conversion

SCHEMA ASSESSMENT AND CONVERSION CAPABILITIES

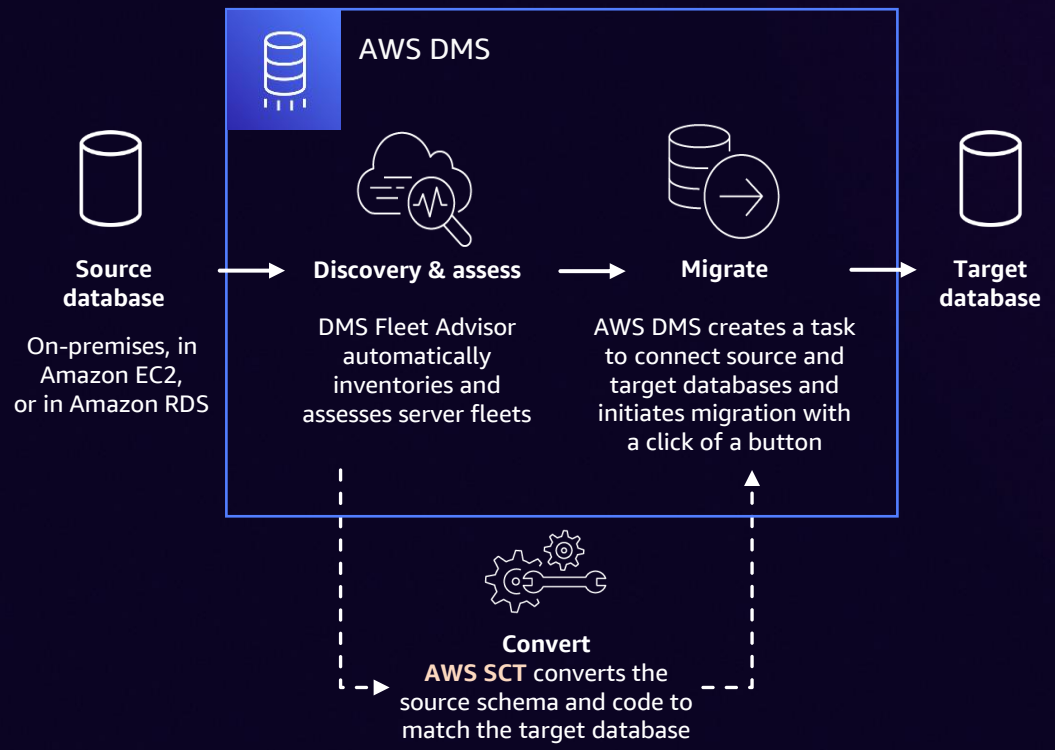


AWS SCT
(“the desktop tool”)

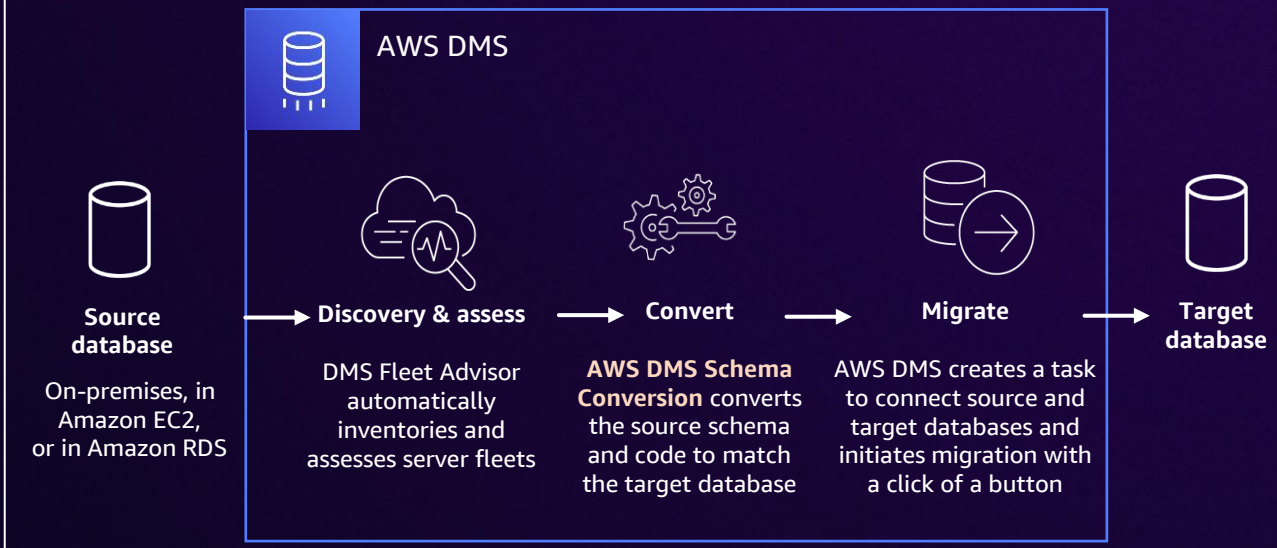
VS.

DMS Schema Conversion
(“the managed service”)

AWS Schema Conversion Tool (AWS SCT)

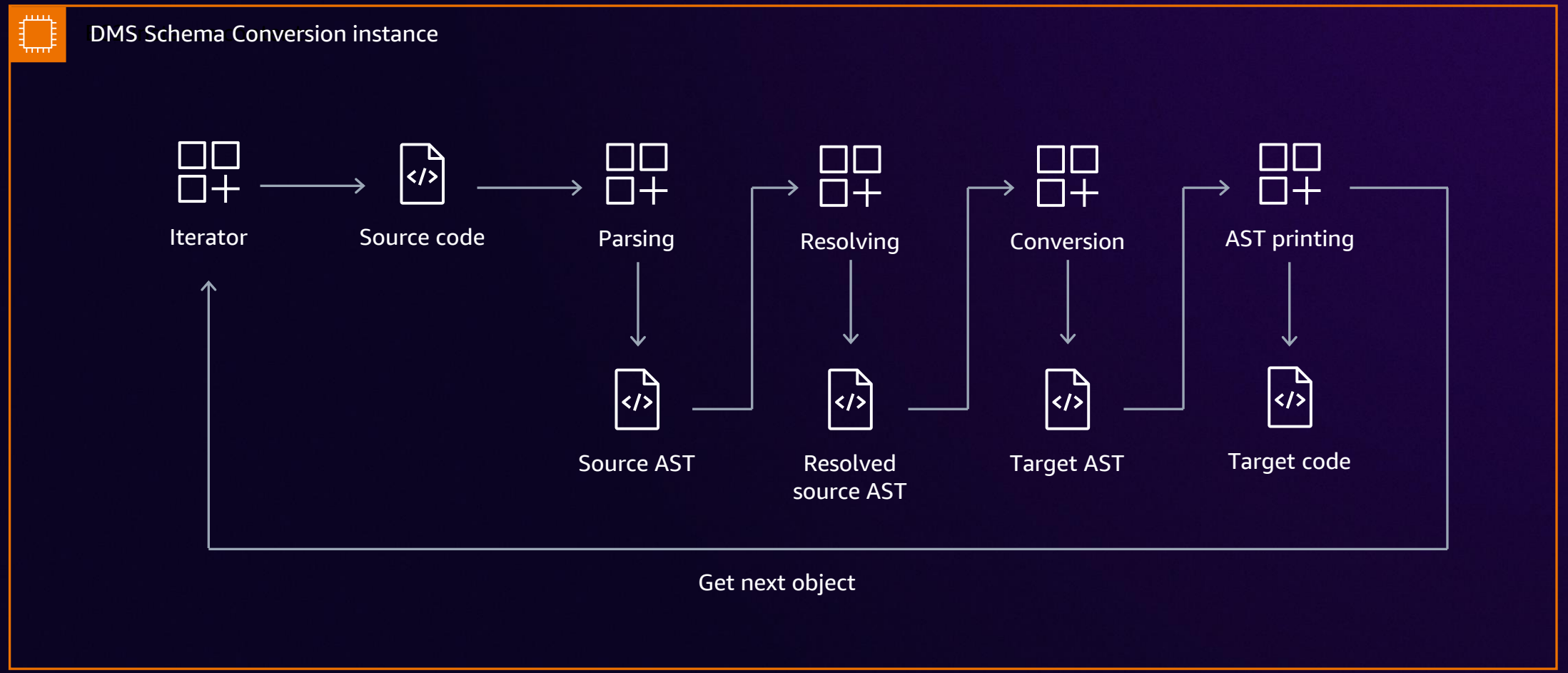


AWS DMS Schema Conversion



DMS Schema Conversion: A close look

DMS Schema Conversion internals



Code example

SOURCE T-SQL STATEMENT

```
CREATE   PROCEDURE dbo.prc_test_ai_ml_2
AS
BEGIN


 DECLARE @var1 varchar(max)
 DECLARE @var2 varchar(max)


 SET @var1 = (SELECT col1 FROM t1 WHERE CURRENT_USER='sa')


 SET @var2 = (SELECT col1 FROM t1 WHERE GETDATE()='2024-09-24')


END
```

RESULTANT TREE

```
<msSql>
    <sqlStatement>
        <statement>
            <createProc>
                <procedureDeclaration  Name="prc_test_ai_ml_2">
                    <schema>
                        <schemaIden>
                            <identifier  Name="dbo"/>
                        </schemaIden>
                    </schema>
                    <routineBody>
                        <statement>
                            <beginEndBlock>
                                <statement>
                                    <declareStatement>
                                        <declareVariable>
                                            <userVariable  Name="@var1"/>
                                            <datatype  Name="VARCHAR" Param1="MAX"/>
                                        </declareVariable>
                                    </declareStatement>
                                </statement>
```

DMS Schema Conversion demo

“

Generative models are a key enabler of machine creativity, allowing machines to go beyond what they’ve seen before and create something new.

Ian Goodfellow

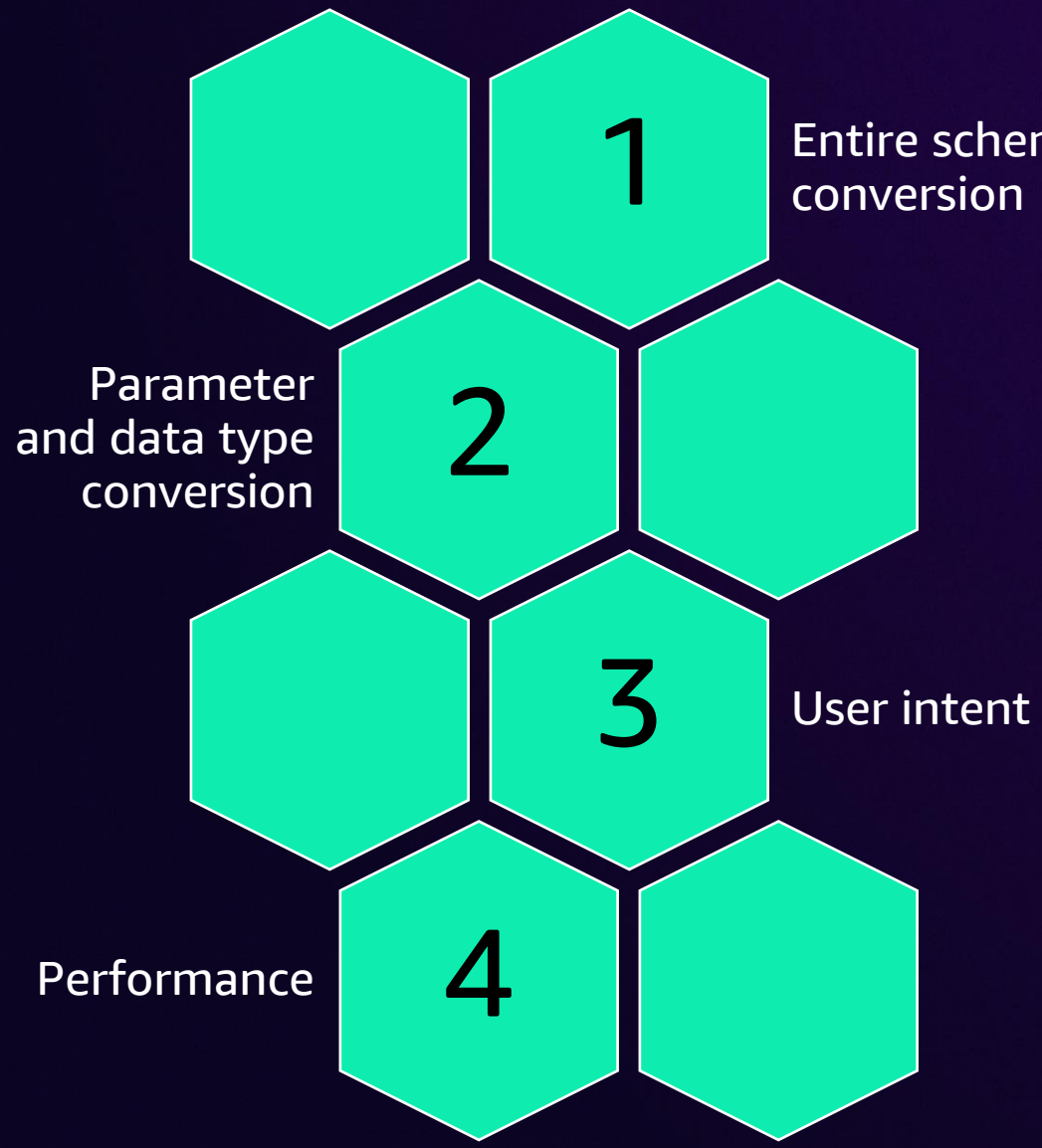
Founder OpenAI research lab

The next wave . . .

ENTER GENERATIVE AI

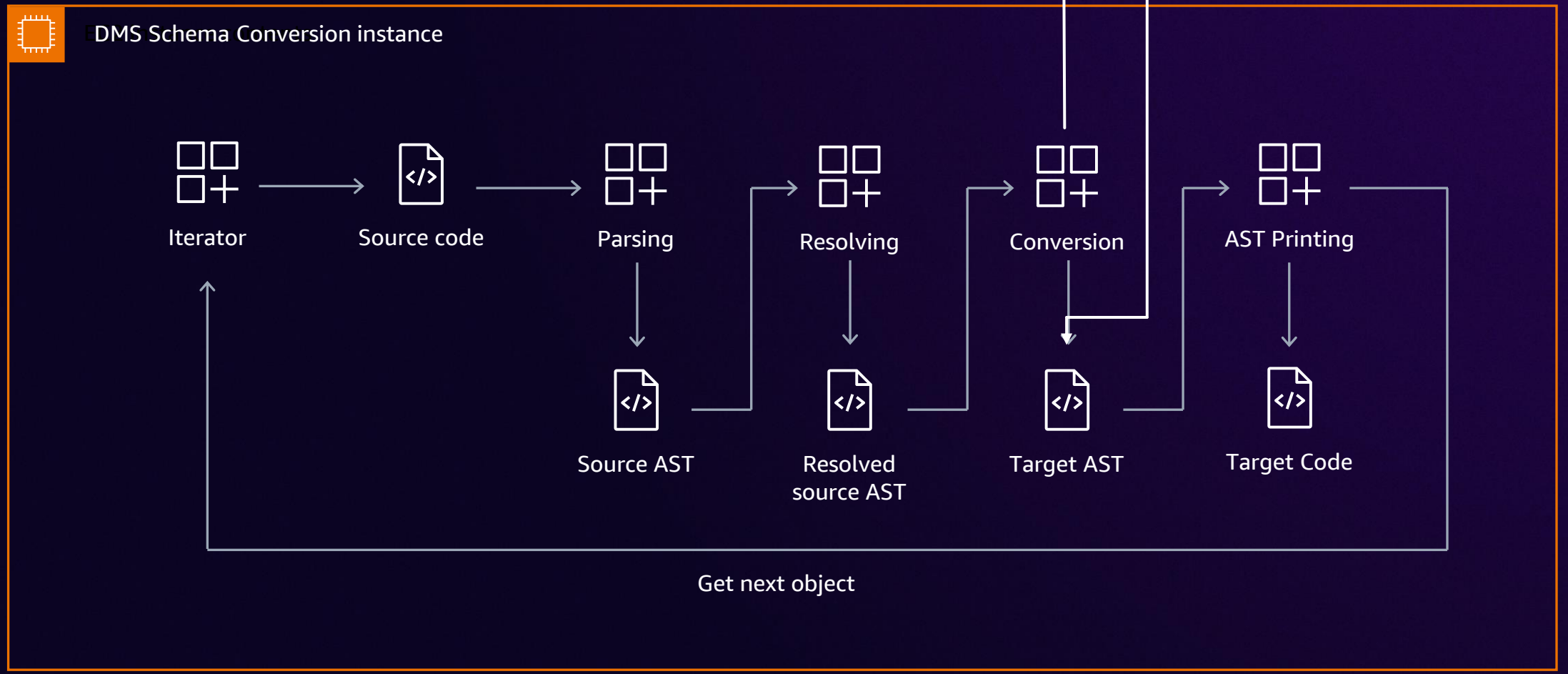
AWS DMS generative AI pilot projects

TRIAL AND ERROR



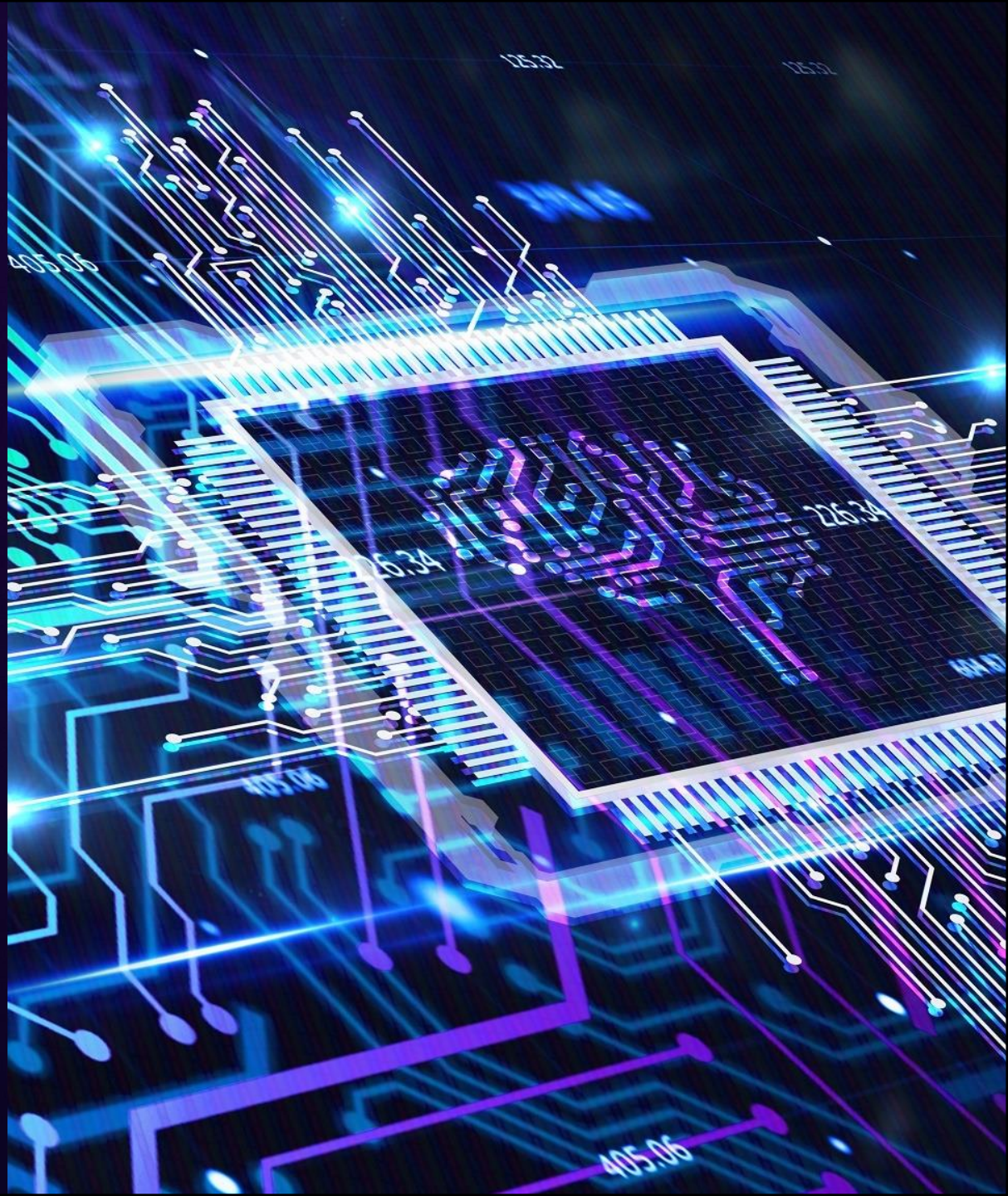
Pulling things together

Generative AI in AWS DMS



Generative AI in AWS DMS

- Context
- Obfuscation
- LLM
- Deobfuscation
- Verification
- AST embedding



Obfuscation example

```
SELECT Col1 AS col INTO a
FROM TEST_AIML_STATEMENT.TBL_AIML_QUALITY_1
taq1
WHERE taq1.COL1 =
PKG_AIML_GLOBAL_VARIABLE.g_variable_table_of
_number(2)
AND rownum < 2
AND rownum < 3;
```

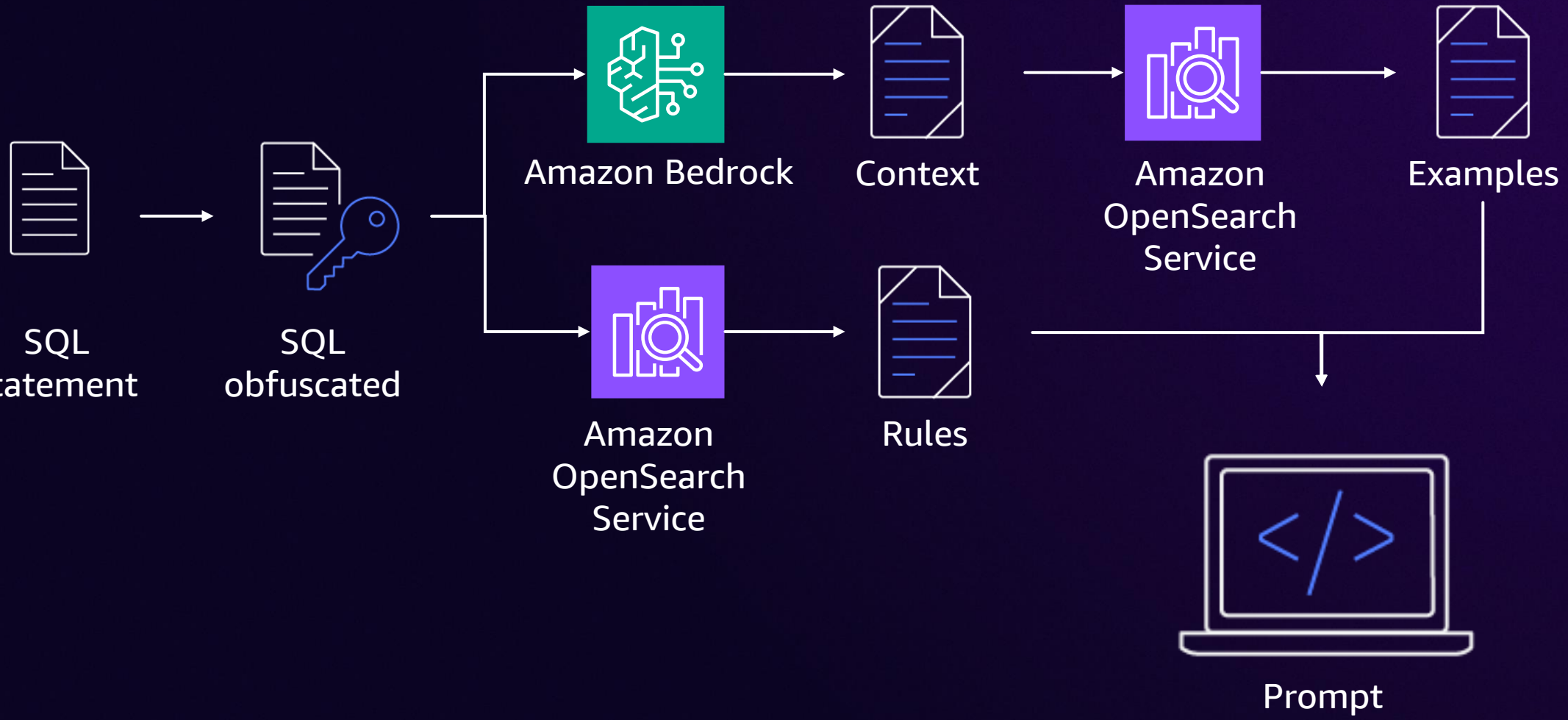
Before, personally identifiable information (PII) is present

After obfuscation, PII is removed

```
SELECT column0 AS col INTO variable0
FROM schema0.table0alias0
WHERE alias0.column1 =
package0.variable1(numericLiteral2)
AND rownum < numericLiteral1
AND rownum < numericLiteral
```

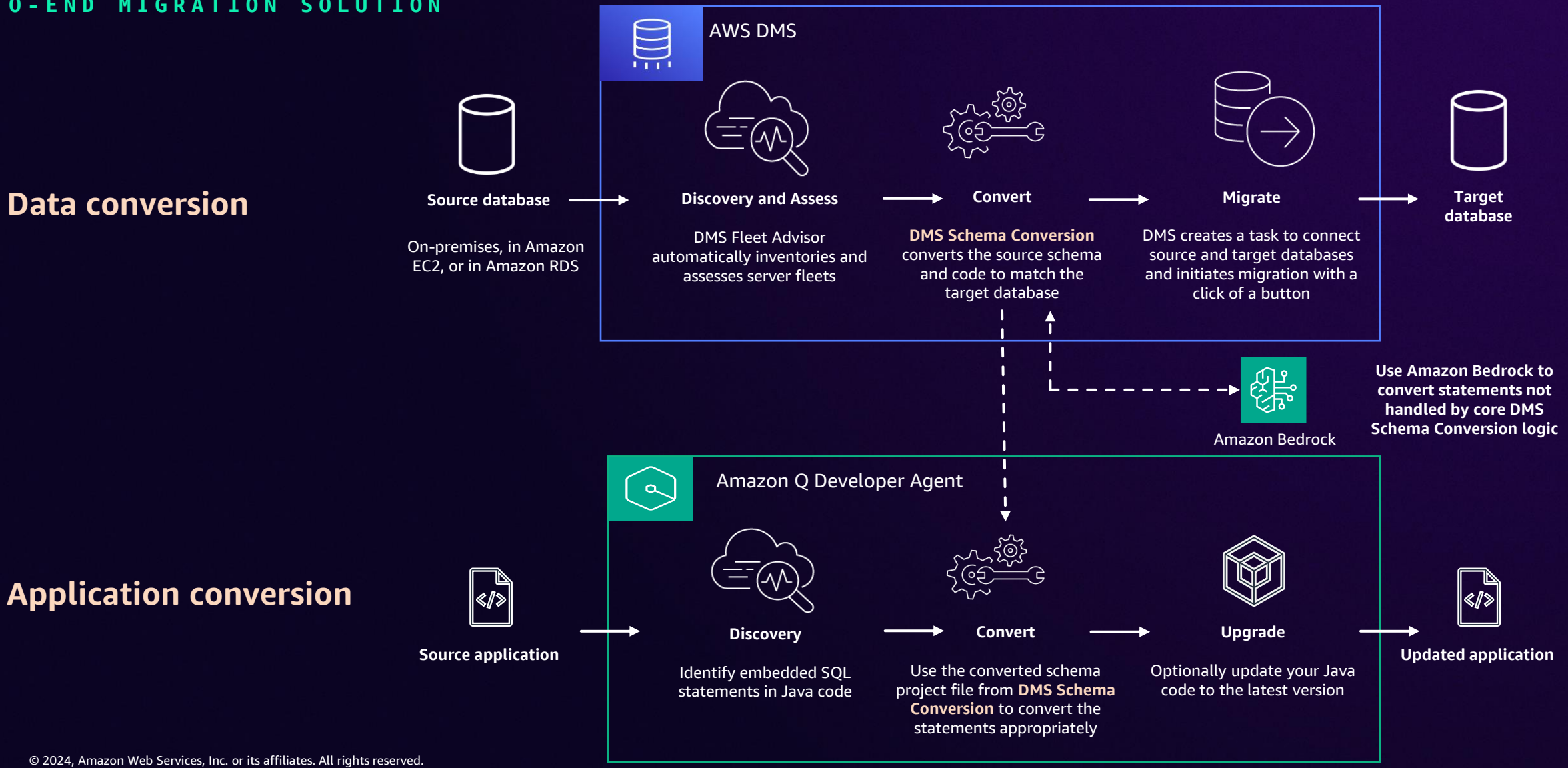
NQR

Prompt engineering

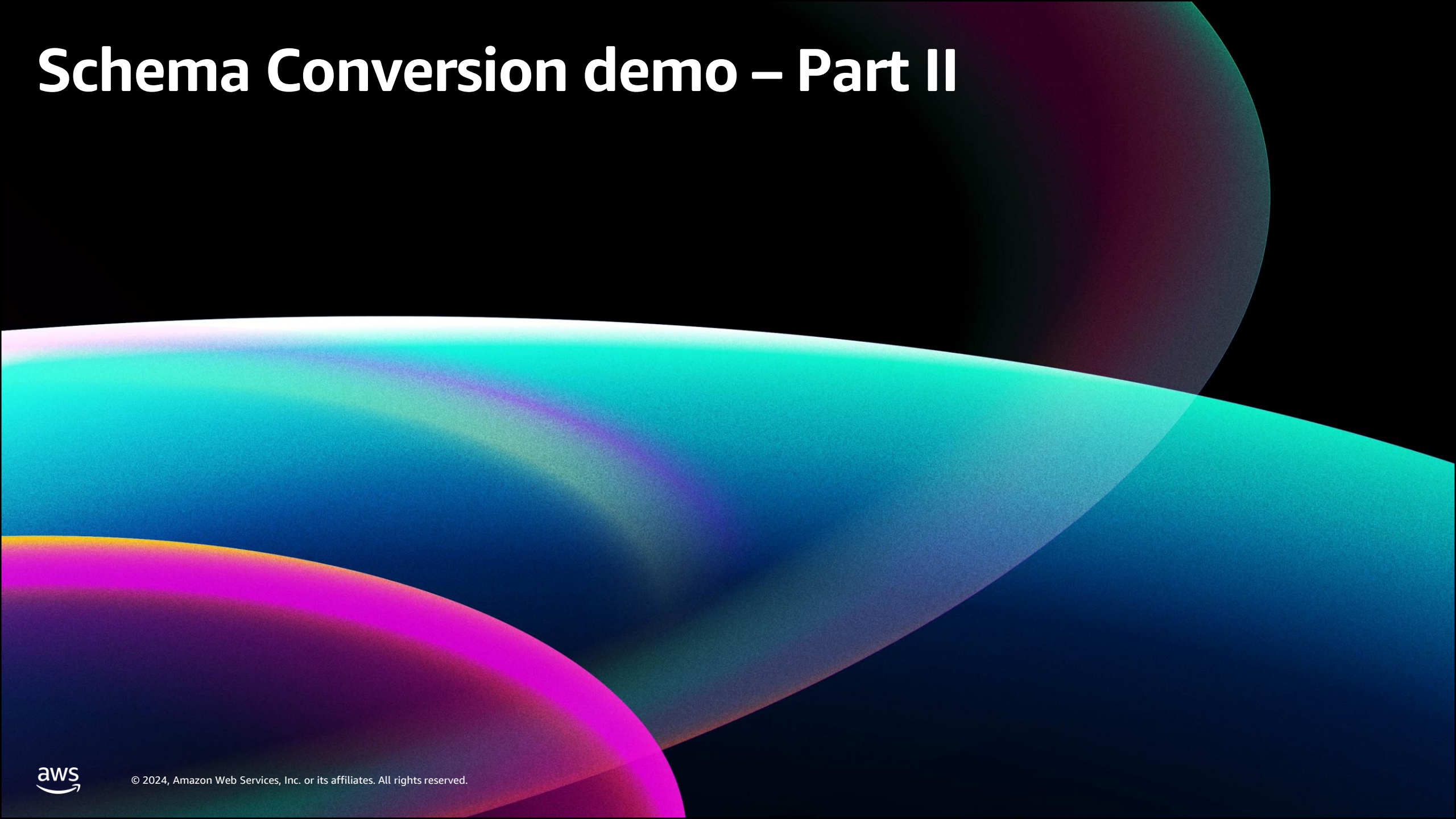


AWS migration components

END-TO-END MIGRATION SOLUTION



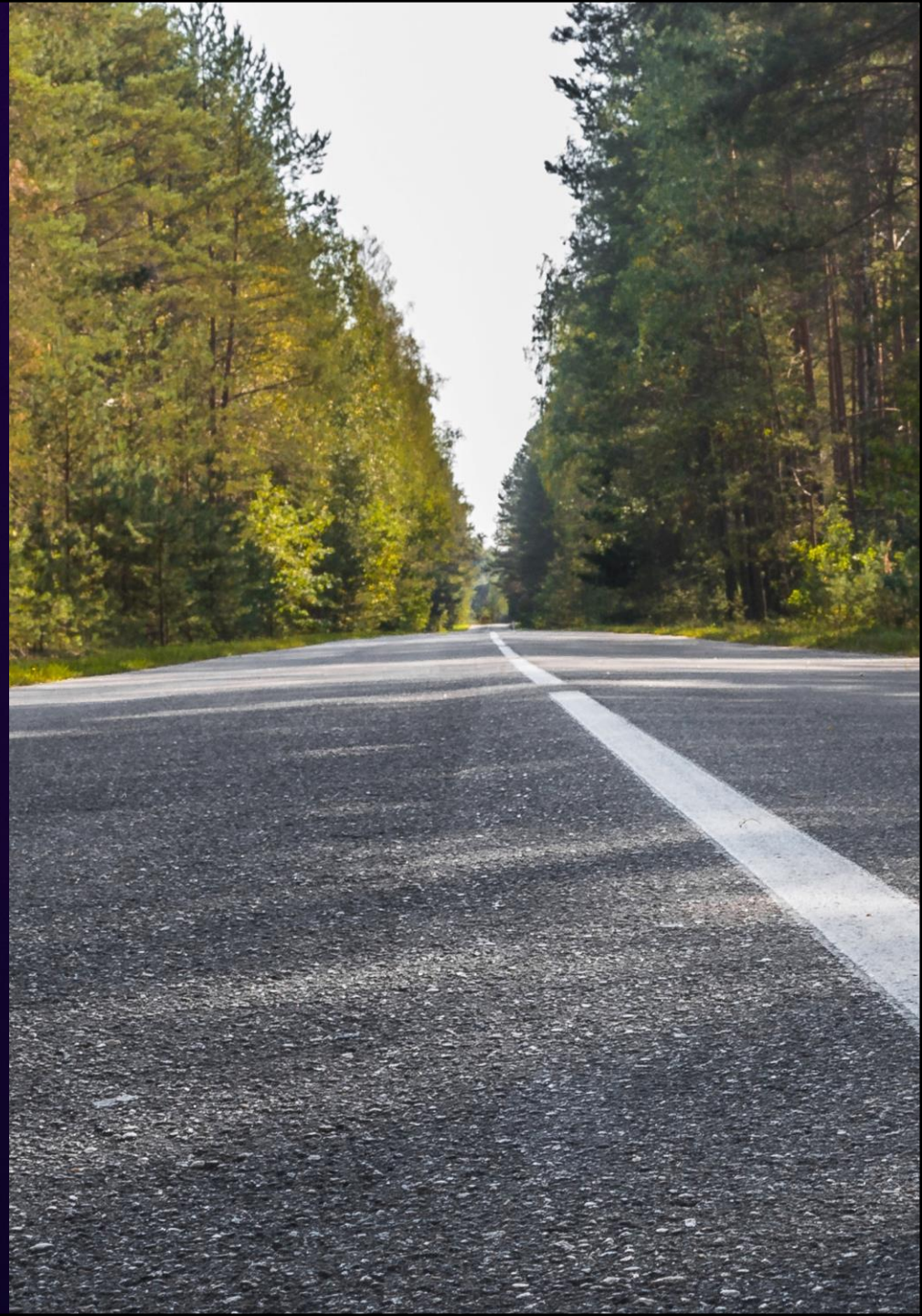
Schema Conversion demo – Part II



What’s next?

Where do we go from here?

- Testing – accuracy and performance
- More endpoints – customer demand
- Model evolution – fast moving
- Scope



Thank you!

Please complete the session survey in the mobile app

John Winford

Senior Manager, Database Services

AWS