

# Change Data Capture (CDC)

Replication

Ingestion

Schema Drift

Kafka

ELT

Streaming Data

File Ingestion

Data Synchronization

---

# Data Ingestion and Replication for Analytics, AI & Operationalization

CMI Product Management Team



Informatica®

# Agenda

1

DI Market Drivers and  
CDW/DL Challenges

2

CDW/DL Reference  
Architecture

3

Why 'Informatica Cloud  
Mass Ingestion' and Use  
Cases?

4

CMI Services and Demo

5

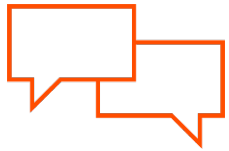
Solution Patterns &  
Customer Examples

6

Summary and  
Call to Action

# Data Integration Market Drivers

Unprecedented Growth in Data Diversity & Volume, Emergence of Fusion Data Teams, and Need to Accelerate Data Operationalization



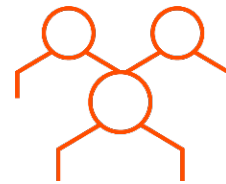
Data Diversity  
(mobile, social, IoT)

**46 billion**  
connected  
devices



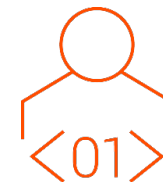
AIML projects  
fail to deliver

**Only 21%** AI  
initiatives in  
production



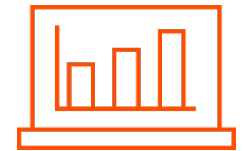
Chronic shortage  
of data engineers

**50% annual  
growth** in open  
data engineering  
positions



Data Engineering  
Democratization

**500 million**  
business data  
users



Explosion in  
Data Volume

**64.2  
zettabytes**  
of data per year

# Businesses struggling with point solutions, complex architecture, lack of resources, runaway costs

## Complexity

75%

of organizations don't have a complete architecture to manage end-to-end set of data activities

## Resource Constraints

50%

year-over-year growth in the number of open data engineering positions

## Cost Overruns

75%

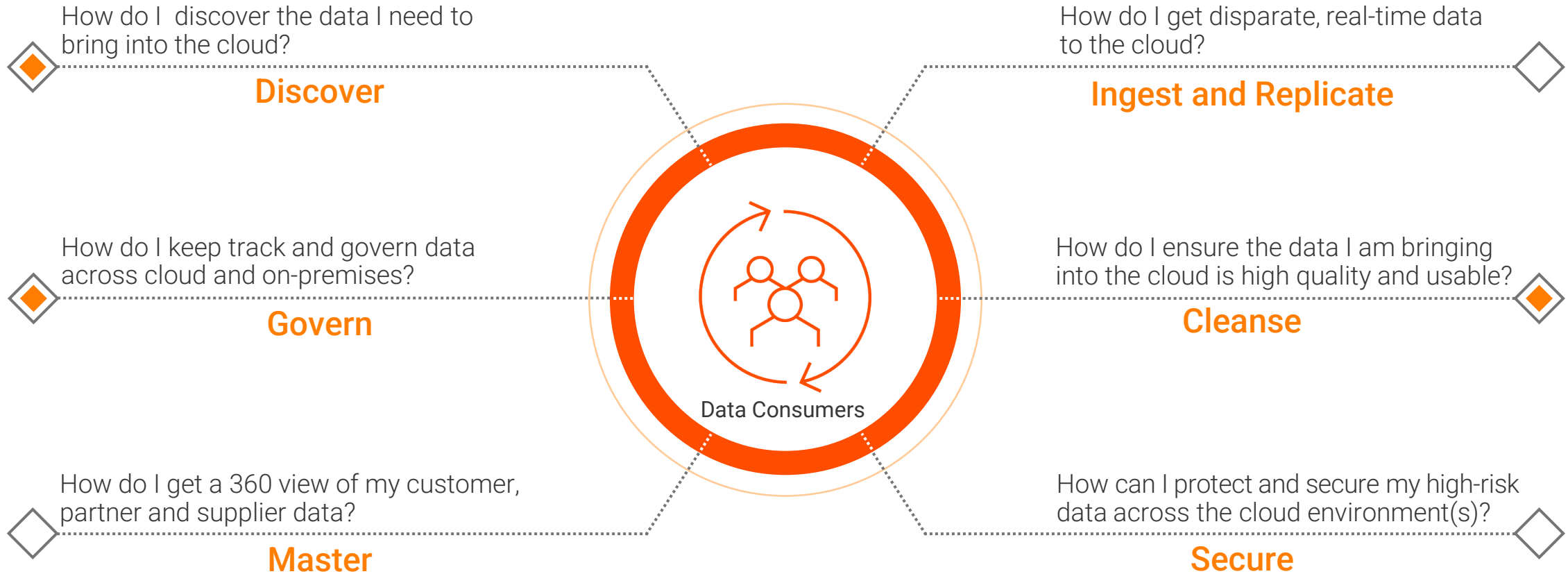
of organizations using cloud data management will encounter budget overruns resulting in their questioning the value of using cloud services

- 75% of point products don't integrate and interoperate
- Data practitioners spend over 80% of their time preparing data instead of analyzing the data
- ~50% of organizations challenged by **data quality**

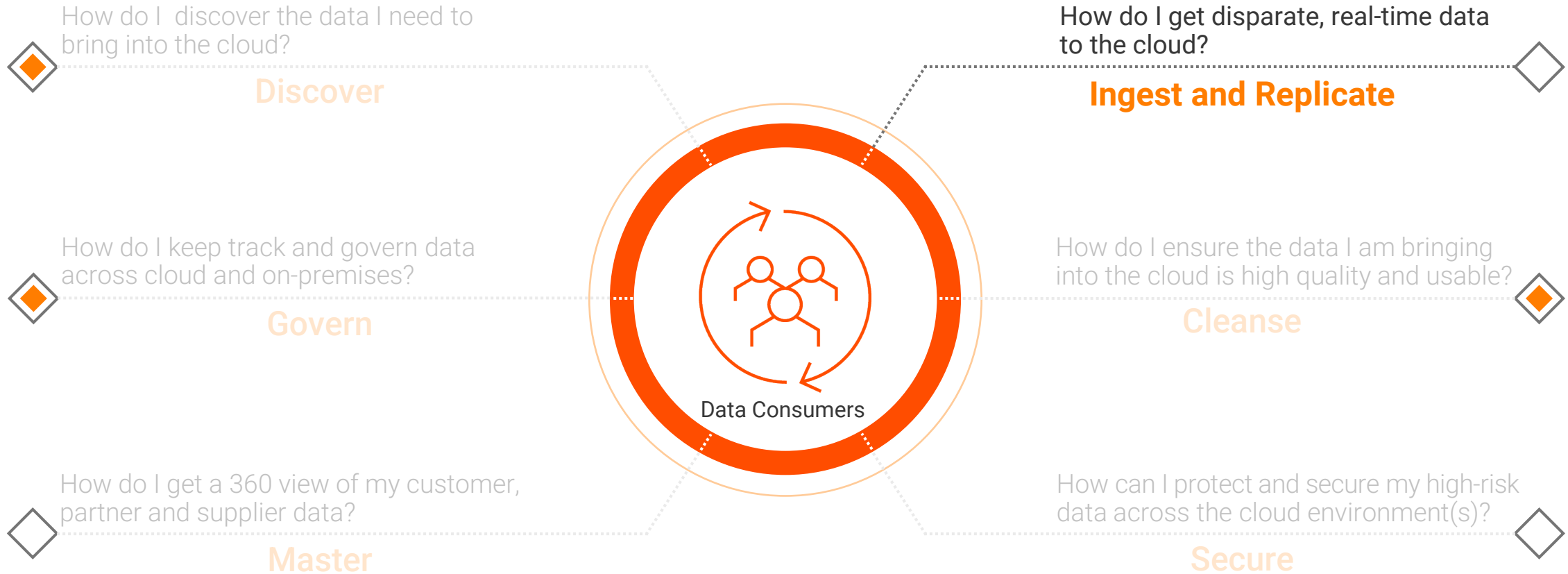
- Difficulty finding specialized skills fast
- Lack of automation impacting the ability to scale
- Lack of self-service access for non-tech users delaying rapid innovation

- Difficulty predicting compute costs
- Lack of visibility and control of users and usage
- Increasing data transfer costs

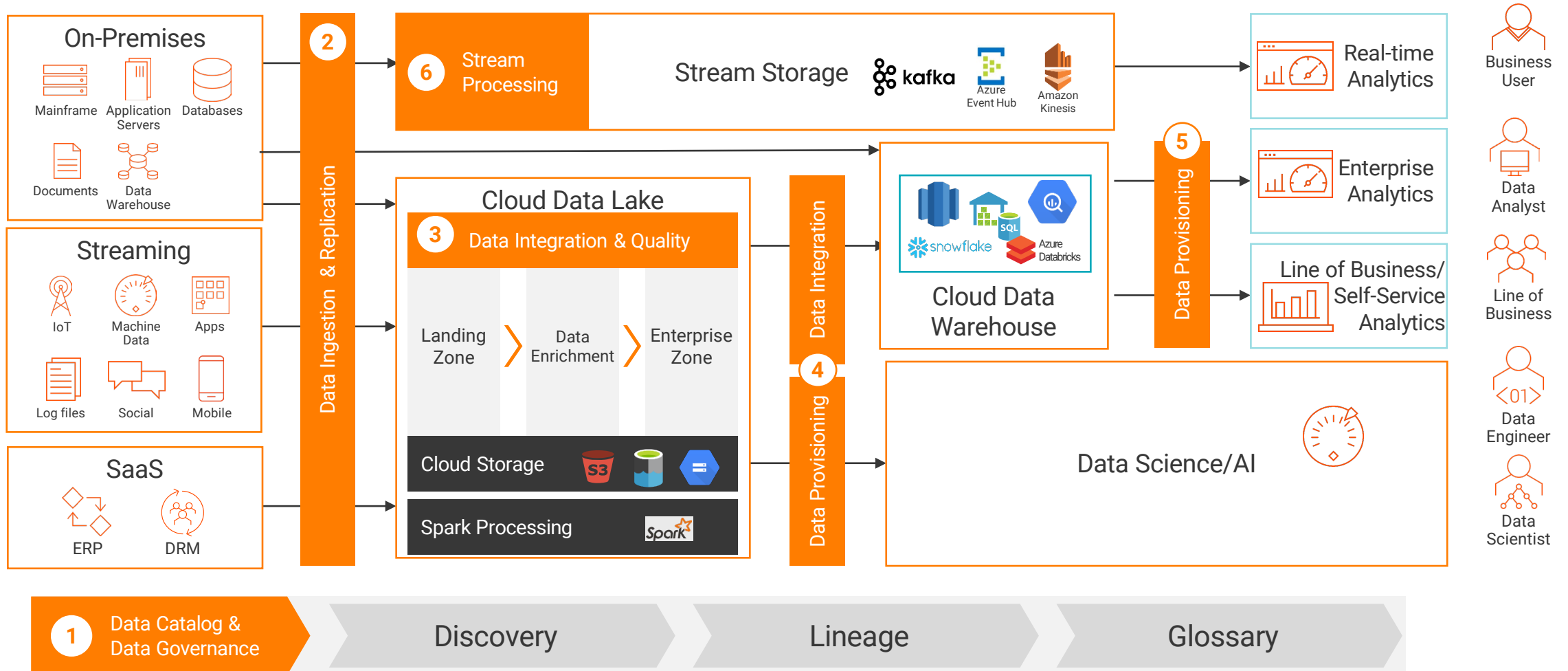
# Challenges at Various Stages of Data Management



# Challenges at Various Stages of Data Management

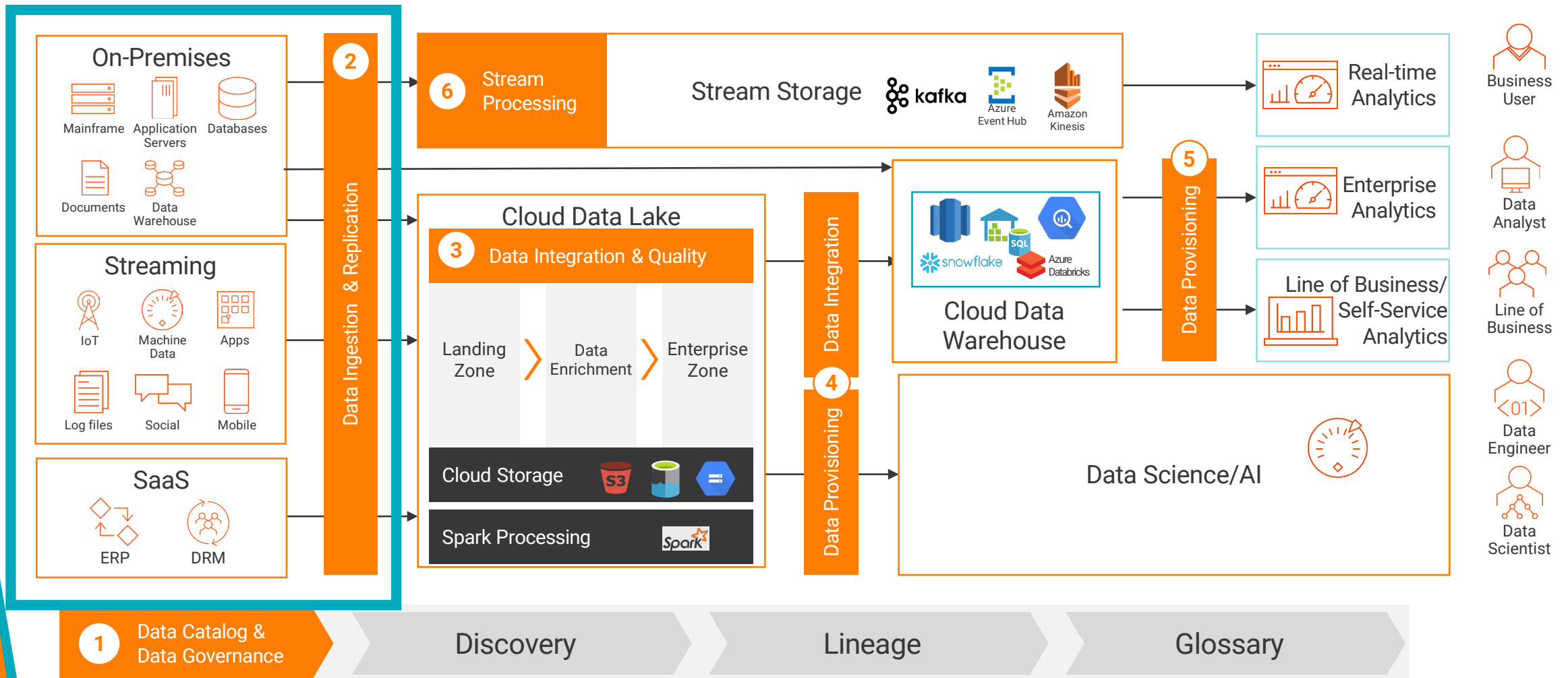


# Informatica Data Management Cloud - Reference Architecture for CDW/DL





# Informatica Data Management Cloud - Reference Architecture for CDW/DL



# Existing Data Ingestion Solutions Requires Hand-Coding And Don't Support All Ingestion Patterns

## Requires Hand-Coding

```

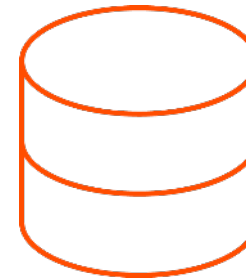
import org.apache.spark._
import org.apache.spark.rdd._
import org.apache.spark.storage.StorageLevel._
import org.apache.spark.sql._
import org.apache.spark.sql.types._
import org.apache.spark.sql.functions._
import org.apache.spark.sql.functions.*
import java.io._
import java.sql.Timestamp
import scala.reflect.ClassTag
import org.apache.spark.sql.catalyst.expressions.Caster
import org.apache.spark.util.LongAccumulator
import org.apache.spark.scheduler.SparkListener
import org.apache.spark.SparkEnv
import org.apache.spark.sql.Row

object Spark {
  def main(args: Array[String]) {
    val sc = SparkContextLoader.getSparkContext
    val sqlContext = SparkContextLoader.getSQLContext
    val ls = new LiveStream(sc.getConf)
    ls.relay(IP.sparkConfToJso(sc.getConf))
    ls.relay(IP.hadoopConfToJso(sc.hadoopConfiguration))
    val lis = new Listener(lis, "TAG")
    sc.addSparkListener(lis)
    sqlContext.sparkSession.experimental.extraPreprocessing = new TaggingRules().rules
    val accs = List()
    ls.relay(IP.sparkAppDetailsToJso(sc.getConf, accs))
    lis.accumulators = accs
    import sqlContext.implicits._
    import org.apache.spark.sql.functions.{stddev_samp, var_samp}
    val icast = caster("MM/DD/YYYY HH24:MI:SS")
    val acast = adapterCaster()
    val jcast = JavaCaster()

    try {
      Tuple2(sqlContext.sql(Params.resolve("DROP TABLE IF EXISTS
'default': 'w8314020283544407639__testpassiveequijoinlkp_ikp_source_lookuptests_newmap'",
"MM/DD/YYYY HH24:MI:SS")), sqlContext.sql(Params.resolve("CREATE TABLE
'default': 'w8314020283544407639__testpassiveequijoinlkp_ikp_source_lookuptests_newmap' ('col0'
DECIMAL(18, 2), 'col1' STRING, 'col2' STRING) ROW FORMAT
SERDE 'hdfs://cdh52.vm.com:8020/tmp/sess8314020283544407639_COPY
_testPassiveEquiJoinLkp_ikp_source_LookupTests_newMap' TBLPROPERTIES
('columns.types='decimal(18,2),string,string',
'pwx.mapping.file.path'='/testPassiveEquiJoinLkp_ikp_source_MAPPING_1204153312640770_1204153
976631823.bin', 'auto.purge'='true', 'columns'='col0,col1,col2)', "MM/DD/YYYY HH24:MI:SS"));
      Tuple2(sqlContext.sql(Params.resolve("DROP TABLE IF EXISTS
'default': 'w8314020283544407639__testpassiveequijoinlpgt_lookuptests_newmap'", "MM/DD/YYYY
HH24:MI:SS")), sqlContext.sql(Params.resolve("CREATE TABLE
'default': 'w8314020283544407639__testpassiveequijoinlpgt_lookuptests_newmap' ('col0' STRING,
'col1' STRING) ROW FORMAT LOCATION
'hdfs://cdh52.vm.com:8020/tmp/sess8314020283544407639//W8314020283544407639__testPassive
EquiJoinLpgTg_LookupTests_newMap' TBLPROPERTIES ('columns.types='string,string',
'pwx.mapping.file.path'='/testPassiveEquiJoinLpgTg_MAPPING_1204153304711020_12041540159301
76.bin', 'pwx.skip.serialization'='true', 'auto.purge'='true', 'columns'='col0,col1)', "MM/DD/YYYY
HH24:MI:SS"));
      val v0 = updatePartitions(asBlock(sqlContext.sql(Params.resolve("SELECT
'w8314020283544407639__testpassiveequijoinlkp_ikp_source_lookuptests_newmap'.col0' as a0,
'w8314020283544407639__testpassiveequijoinlkp_ikp_source_lookuptests_newmap'.col1' as a1,
'w8314020283544407639__testpassiveequijoinlkp_ikp_source_lookuptests_newmap'.col2' as a2
FROM 'default': 'w8314020283544407639__testpassiveequijoinlkp_ikp_source_lookuptests_newmap'",
"MM/DD/YYYY HH24:MI:SS")))
      val v1 = v0(0);
      val v2 = v0(2);
      val v3 = v0.groupBy(v1, v2).agg(v1, v2, last(v0(1), false)).toDF("m");
      val v4 = updatePartitions(asBlock(sqlContext.sql(Params.resolve("SELECT
'w8314020283544407639__src1_lookuptests_newmap'.col0' as a0,
'w8314020283544407639__src1_lookuptests_newmap'.col1' as a1,
'w8314020283544407639__src1_lookuptests_newmap'.col2' as a2 FROM
'default': 'w8314020283544407639__src1_lookuptests_newmap'", "MM/DD/YYYY
HH24:MI:SS")))
      val v5 = v4.join(v3, v3(0) <=> (v4(1)) && (v3(1) <=> (v4(2))), "left_outer").toDF();
      asBlock(sqlContext.sql(Params.resolve("INSERT OVERWRITE TABLE
'default': 'w8314020283544407639__testpassiveequijoinlpgt_lookuptests_newmap' SELECT tbl0.c0 as
a0, tbl0.c1 as a1 FROM tbl0", "MM/DD/YYYY HH24:MI:SS")), v5.select(v5(0),
v5(5)).toDF("TGT_").tag("TGT_TGT_PassiveEquiJoinLpgTg").toDF("c").createOrReplaceTempView("tbl0");
      ) finally {
        sqlContext.sql(Params.resolve("DROP TABLE IF EXISTS
'default': 'w8314020283544407639__testpassiveequijoinlpgt_lookuptests_newmap'", "MM/DD/YYYY
HH24:MI:SS"));
        sqlContext.sql(Params.resolve("DROP TABLE IF EXISTS
'default': 'w8314020283544407639__testpassiveequijoinlkp_ikp_source_lookuptests_newmap'",
"MM/DD/YYYY HH24:MI:SS"));
      }
    }
  }
}

```

## Doesn't Support All Ingestion Patterns



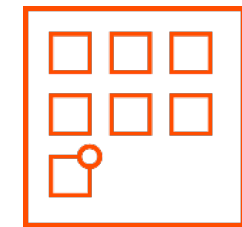
Database



Streaming

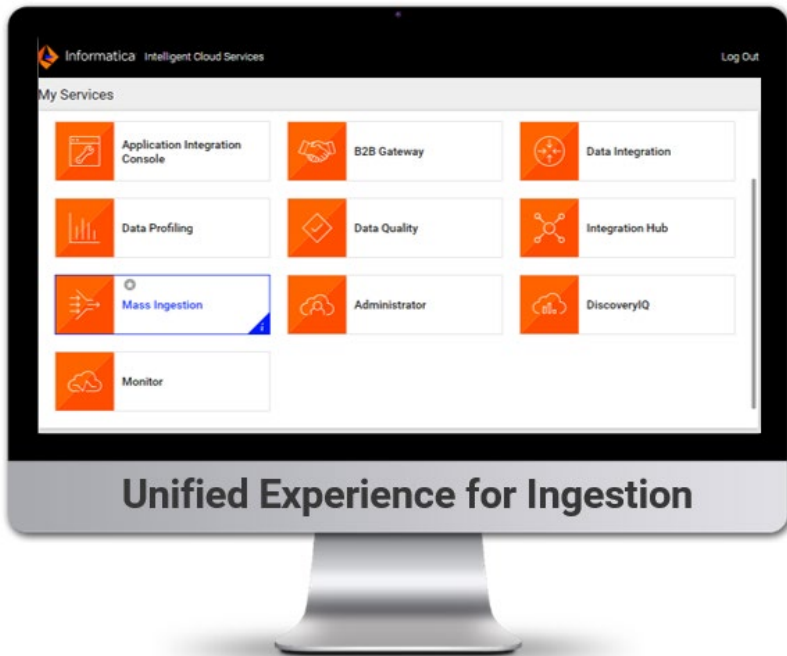


Files

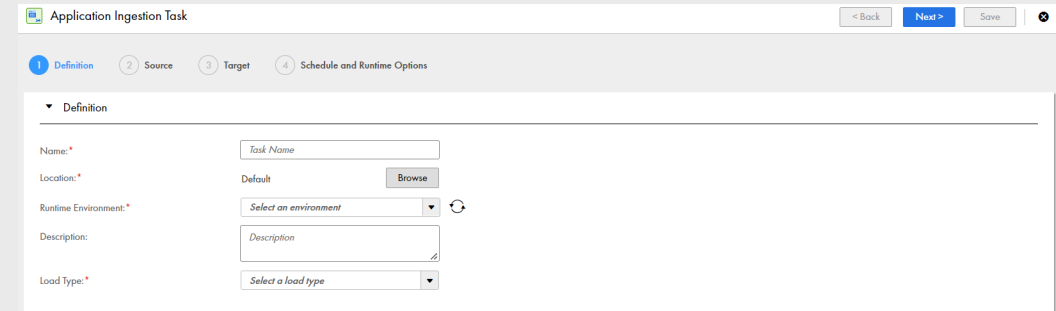


Applications

# Informatica's Data Ingestion and Replication Solution – Cloud Mass Ingestion



- ✓ Step-by-step wizard for designing and creating an ingestion task



- ✓ Deployment, scheduling, real-time monitoring and lifecycle management



Ingest in Real-Time



Real-Time Monitoring

- ✓ Versatile out-of-the-box connectivity to sources and targets



Databases & CDC



Streaming Sources



Files



Applications

# Cloud Mass Ingestion (CMI)

## Use Case Patterns

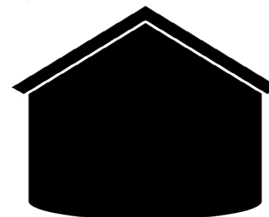
### Data Lake Ingestion

- Mass ingestion of application, on-premises database content into a cloud or on-premises data lake
- Mass ingestion of files into cloud and on-premises data lakes
- Streaming and IoT data ingestion into a data lake



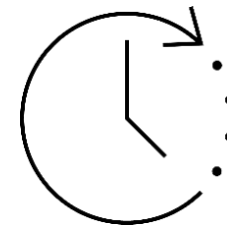
### DB/DWH Replication/Data Warehouse Modernization

- Mass ingestion of on-premises database, data warehouse, applications and mainframe content into a cloud data warehouse (Snowflake, Synapse etc.)
- Synchronize ingested data with Change Data Capture (CDC) and applying



### Real Time Analytics

- Log files and clickstream ingestion
- CDC ingestion
- IoT data ingestion
- App change data ingestion



# Cloud Mass Ingestion

Services



Choose the asset type you want to create.

Application Ingestion  
Task



Database Ingestion  
Task



File Ingestion  
Task



Streaming Ingestion  
Task



Cancel

# Mass Ingestion Applications **New!**

# Average Number of Apps per Customer, by Industry





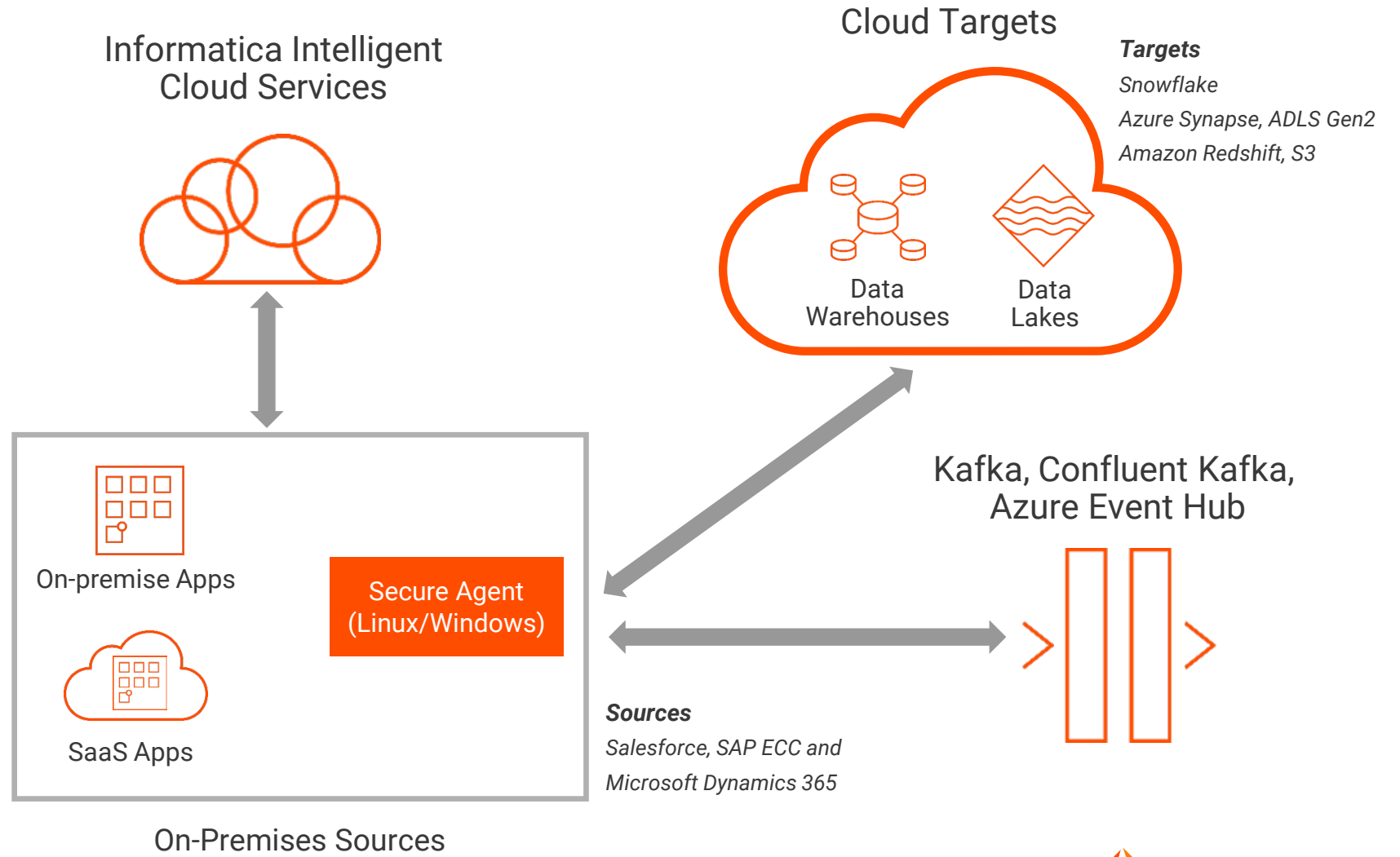
# Cloud Mass Ingestion - Applications

**Provides Application ingestion** capabilities as part of IICS Mass Ingestion service

**Ingest** data from SaaS and on-prem applications like Salesforce, SAP ECC and Dynamics 365. Also supports schema drift for supported applications

**Real-time monitoring** of ingestion jobs with lifecycle management and alerting in case of issues

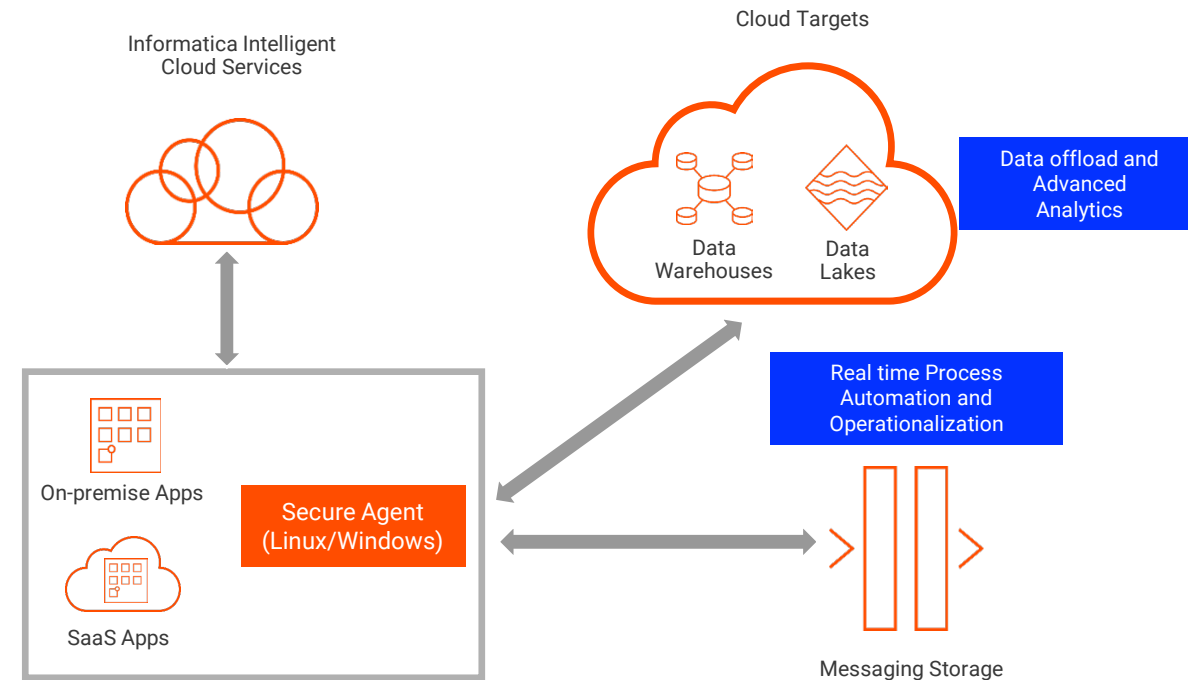
**Orchestrate** Application data ingestion in **hybrid / cloud** as **managed** and **secure** service



# Mass Ingestion Applications – Key Use Cases

- Analytics offload from SaaS application & DW consolidation
- Advanced Analytics on Application Data
  - Initial & change data ingestion, apply changes, schema drift, near real time ingestion
- Real time process automation & action operationalization (for App modernization)
  - CDC data ingestion onto Messaging Systems
  - Integration with CAI and CIH for downstream consumption\*
  - Task flow Integration for downstream processing\*

\*roadmap



# Efficient Capture and Ingestion of CDC Data

The screenshot displays the Informatica Data Integration console for a job named 'Salesforce\_Snowflake\_\_combined'. The interface is divided into two main sections: configuration and execution details.

**Configuration Section (Left):**

- Definition:** Name: Salesforce\_Snowflake\_\_combined, Location: R38\_DEMO, Runtime Environment: USWPF2R1VJK-AAD, Load Type: Initial and Incremental Loads.
- Source:** Type: Microsoft Dyna..., Name: MSD365\_AP...
- Target:** Type: Snowflake Clou..., Name: dbmi\_sf\_R

**Execution Details Section (Right):**

- Job Overview:** Shows 1 Running and 1 Total table.
- Table Name:** MSD365.cr1c6\_all\_dts
- Target Object:** PADMINI.SD\_MD\_cr1c6\_all\_dts
- Status:** Running
- Log:** Select log
- Statistics:** Inserts: 3, Deletes: 0, Updates: 0, LOBs: 0

Efficient change data collection from source system and applies the changes onto the target

# Automatically Addresses Schema Drift at the Source

The screenshot displays the Informatica Mass Ingestion interface for a job named 'Salesforce\_Snowflake\_\_combined'. The interface is divided into several sections:

- Navigation:** A top bar shows the user 'Om Verma' and navigation buttons: '< Back', 'Next >', 'Save', 'View', 'Deploy', and a close button.
- Steps:** A progress bar indicates four steps: 1. Definition, 2. Source, 3. Target, and 4. Schedule and Runtime Options (currently active).
- Schema Drift Options:** A section with a dropdown arrow containing four fields:
  - Add Field: \* (Dropdown: Replicate)
  - Modify Field: \* (Dropdown: Replicate)
  - Drop Field: \* (Dropdown: Ignore)
  - Rename Field: \* (Dropdown: Ignore)
- Advanced:** A section with a dropdown arrow containing:
  - Number of Rows in Output File: \* (Input: 100000)
  - Schedule (Expandable)
  - Custom Properties (Expandable)

An inset window shows a detailed view of the 'Add Field: \*' dropdown menu, which is open and displays the following options: Replicate, Ignore, Stop Job, and Stop Object.

Schema drift recognizes changes on the source application schema and automatically processes and applies the changes

# Real time monitoring

The screenshot displays the Informatica Monitor interface. At the top, a diagram shows a data flow from a Source (Type: Oracle, Name: srao\_ora122\_...) to a Target (Type: Snowflake, Name: srao\_sf\_rh5). To the right, a summary shows 19 Tables Running and 19 Total. Below this, the 'Object Detail' tab is active, showing a table of 19 items. The table columns are Object, Target Object, Stage, State, and Log. All items are in a 'Running' state. A summary of statistics is provided for the first item: LOBs: 0, Inserts: 25, Unload Count: 50, Deletes: 50, Updates: 250, and DDL statements: 0.

Object	Target Object	Stage	State	Log
▼ AUSQA.ORA628_SRC_BK	SRAO.ORA628_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA604_SRC_BK	SRAO.ORA604_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA626B_SRC_BK	SRAO.ORA626B_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA621_SRC_BK	SRAO.ORA621_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA605_SRC_BK	SRAO.ORA605_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA620_SRC_BK	SRAO.ORA620_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA603_SRC_BK	SRAO.ORA603_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA602_SRC_BK	SRAO.ORA602_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA622_SRC_BK	SRAO.ORA622_SRC_BK	Normal	Running	Stage Log
▶ AUSQA.ORA623_SRC_BK	SRAO.ORA623_SRC_BK	Normal	Running	Stage Log

# Out Of The Box Native Connectivity



# Mass Ingestion Databases

# Cloud Mass Ingestion Databases

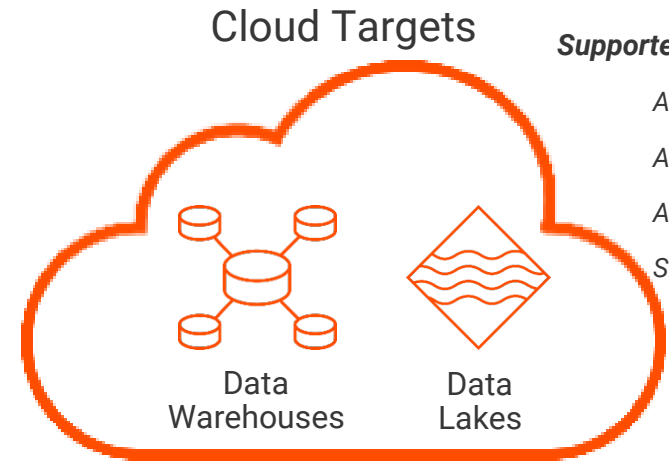
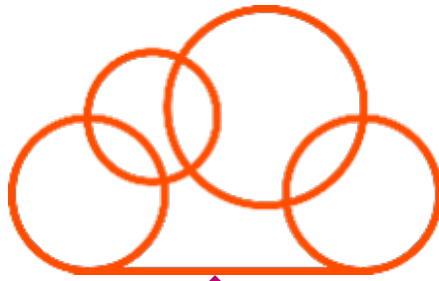
**Provides Database ingestion** capabilities as part of IICS Mass Ingestion service

**Ingest** relational database data from Oracle, SQL-Server & MySQL. Also supporting Schema Drift on CDC supported Databases

**Real-time monitoring** of ingestion jobs with lifecycle management and alerting in case of issues

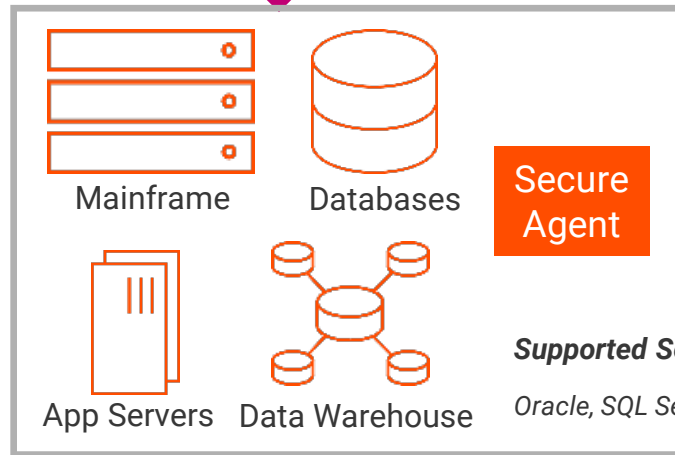
**Orchestrate** Database data ingestion in **hybrid/cloud** as **managed** and **secure** service

Informatica Intelligent Cloud Services



**Supported Targets**

- Amazon S3
- Azure ADLS & SQL DW
- Apache Kafka
- Snowflake



On-Premises Sources

Secure Agent

**Supported Sources**

Oracle, SQL Server, MySQL

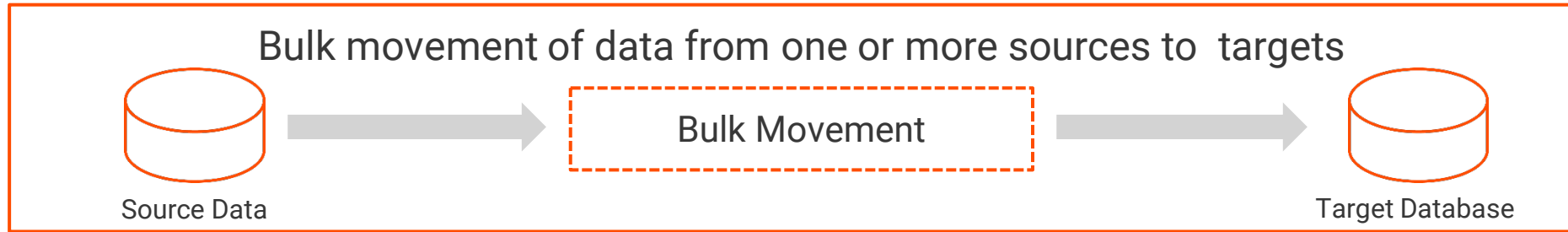
Kafka



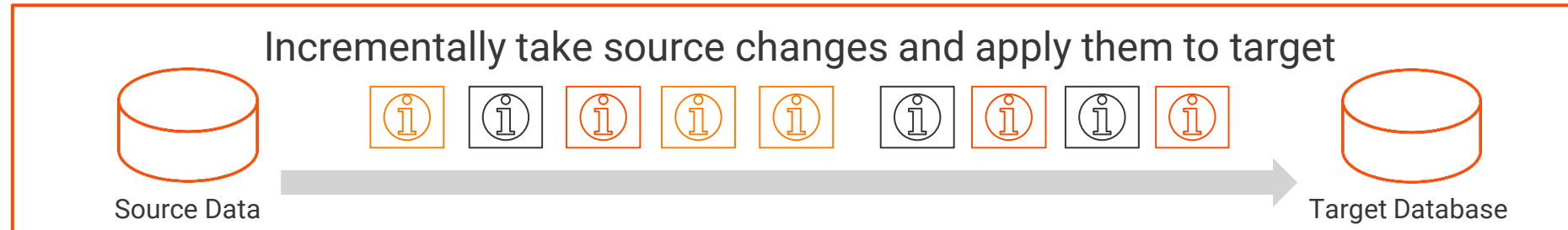


# Data Synchronization between source & target- Typical flow

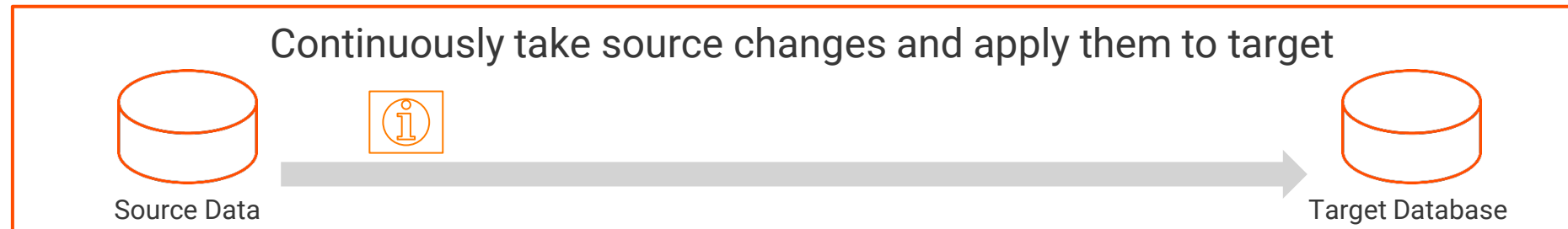
**Step 1**



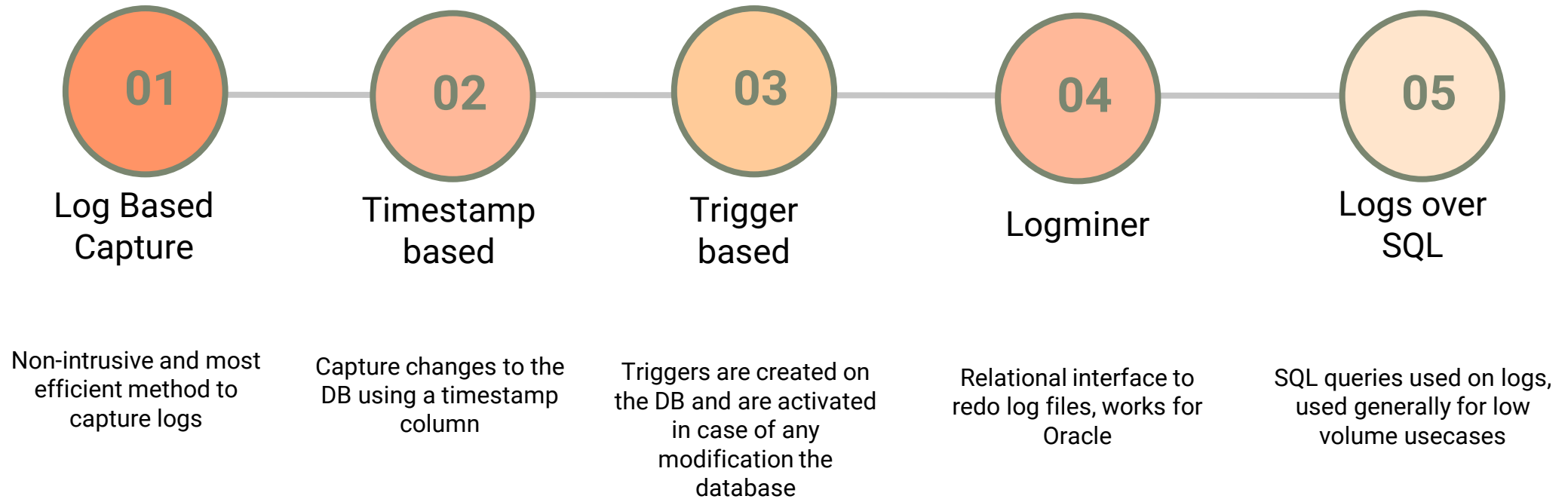
**Step 2**



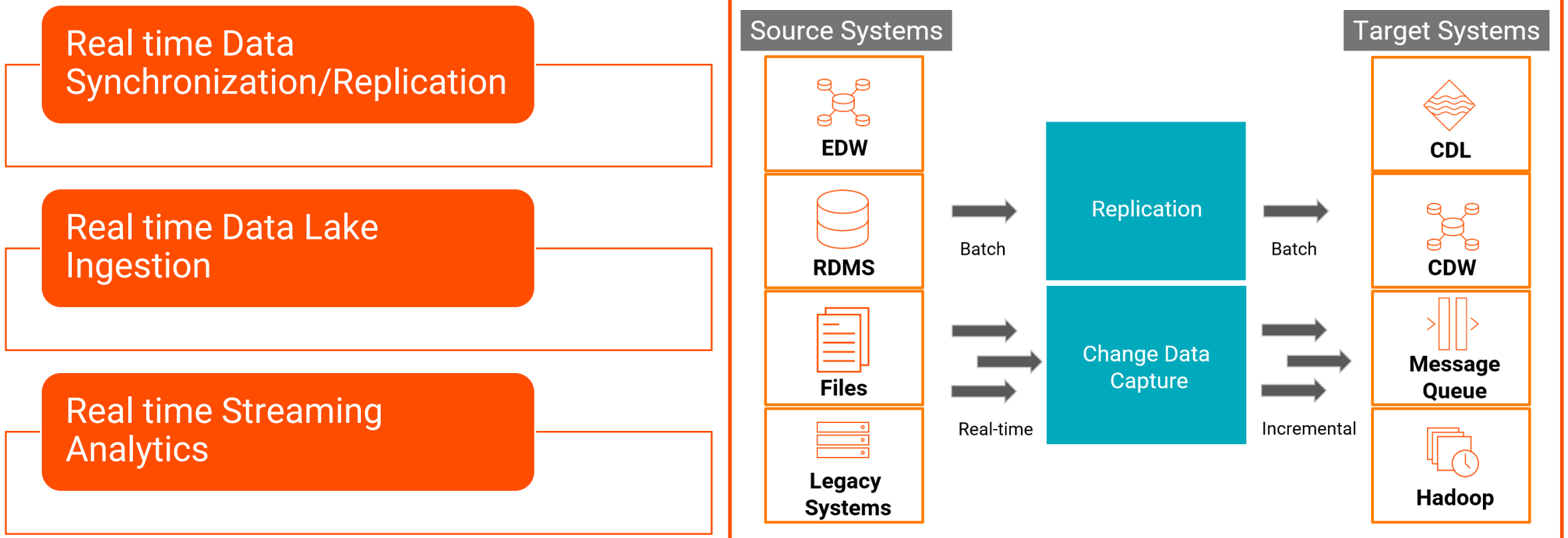
**Step 3**



# Change Data Capture (CDC) – Approaches



# Use cases with Change Data Capture



# Capabilities of DBMI

- Different types of Ingestion
  - Initial load
  - Incremental
  - Combined Load
- Support for invariably variety of different source target combination
- Log-based change data capture for different sources
- Schema drift for change in source schema
- NoSQL Data source support - Replication support of JSON datatype from NoSQL source MongoDB
- LOB columns - Initial Load support for LOB columns

# Capabilities of DBMI

- Combined load support for Lake Targets
- Support for on-prem databases as target for Database Ingestion
- Views – Support for ingestion of views
- Audit column support for target like Snowflake and Databricks
- Custom parameter tuning option for optimized Ingestion
- Data preview while configure your replication job
- Insight and metrics on the replication jobs
- Capturing error logs to do quick RCA in cases of failure

# Mass Ingestion Streaming

# Cloud Mass Ingestion - Streaming

Provides streaming ingestion capabilities as part of IICS data ingestion service

Ingest streaming data: logs, clickstream, social media, Kafka Kinesis, S3, ADLS, Firehose, etc.

Real-time monitoring of ingestion jobs with lifecycle management and alerting in case of issues

Orchestrate streaming data ingestion in hybrid / cloud as managed and secure service



Sensor Data



Machine Data/IOT



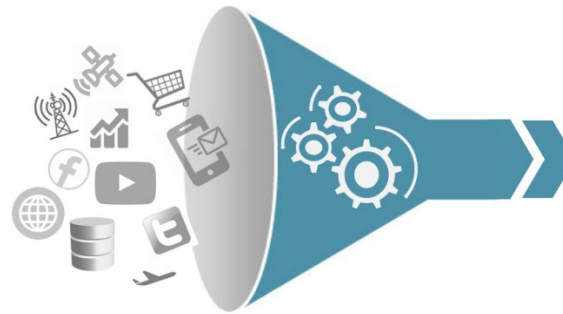
WebLogs



Social Media



Messaging Systems



Messaging Systems

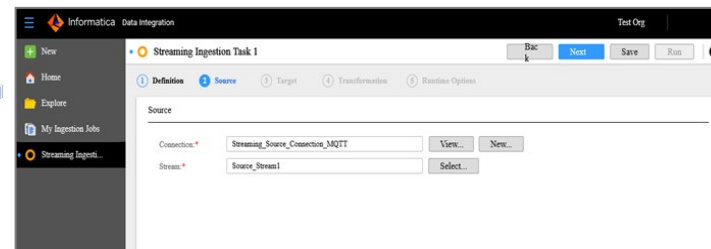


Data Lake & ML



Real-time Analytics

Consumption



Informatica

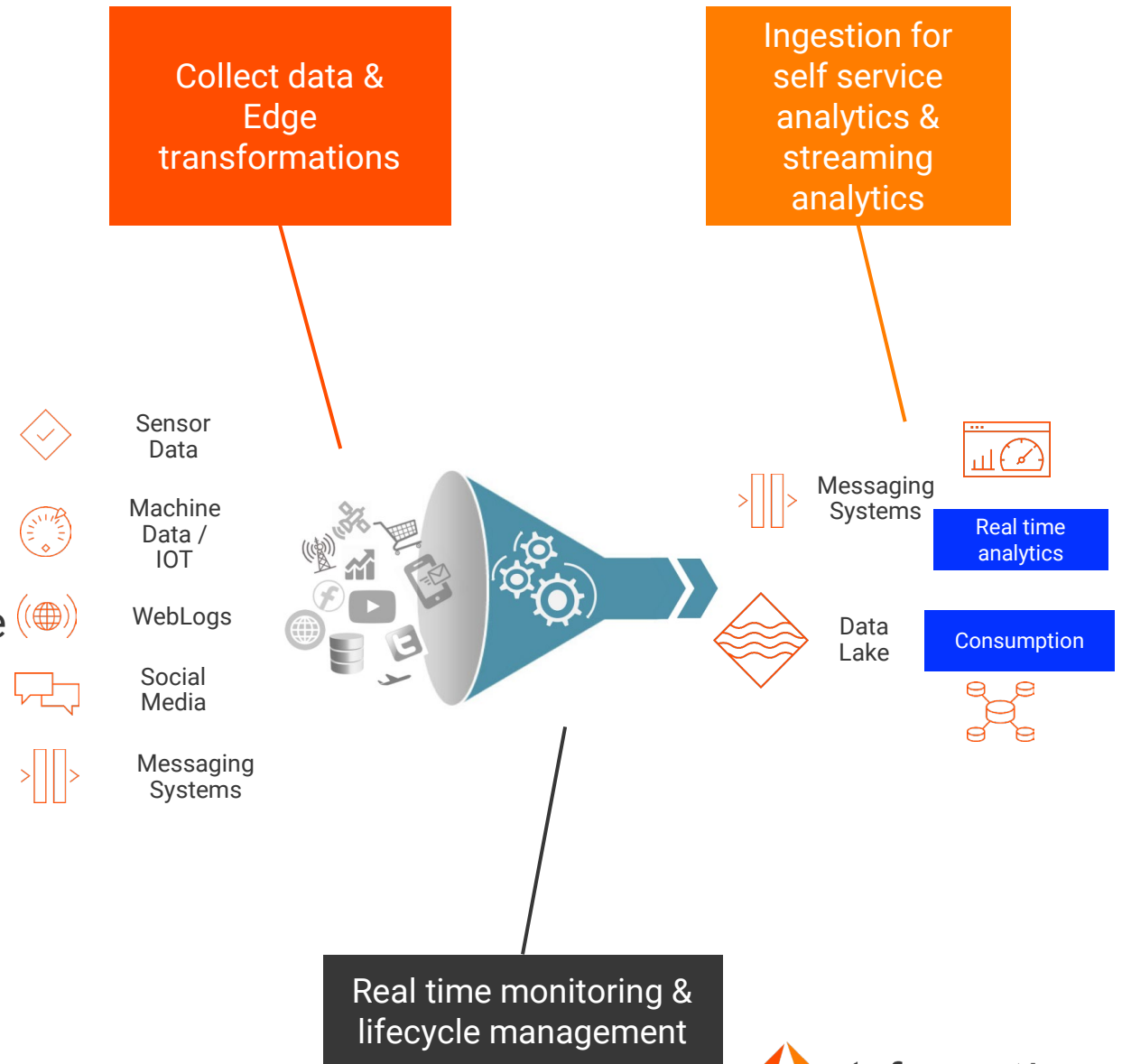
# Common Use Case

## ➤ Lake Ingestion

- ✓ Kafka data ingestion onto Cloud data lake
- ✓ IoT & Industrial IoT Data ingestion onto Cloud Data Lake for batch analytics

## ➤ Real time Analytics

- ✓ IoT & IIoT data ingestion onto Kafka (with simple filtering)
- ✓ Weblogs and Clickstream ingestion onto Kafka for real-time analytics



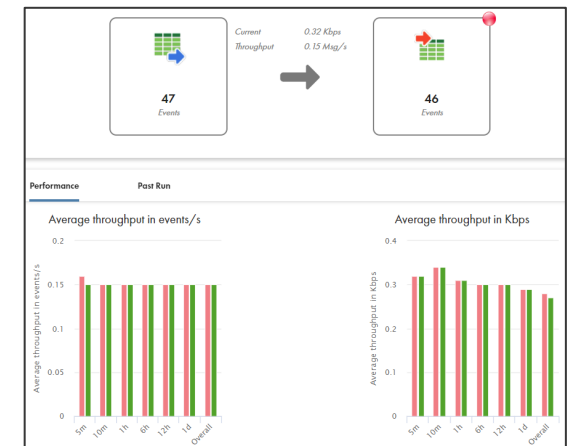


# IICS Cloud Mass Ingestion Streaming - Capabilities

## Key Capabilities and Functionality

- **Authoring Wizard**
  - Common Wizard based authoring experience for creating the data flows
  - Ease of deployment
- **Edge Transformation**
  - Simple Edge transformations—Filter, Python, Splitter, Format Converter, Combiner, Java, and Jolt
- **Real Time Monitoring**
  - Visualization driven real time monitoring
  - Lifecycle management
  - Alerting in case of failures
- **Connectivity**
  - Streaming sources—Tail Files, Kafka, JMS, Azure EventHub, Amazon Kinesis REST API, OPC UA, Google PubSub and MQTT
  - Messaging and data lakes—Kafka, Amazon S3, Amazon Kinesis, Azure Event Hub, Google PubSub & ADLS Gen2

The screenshot shows the 'IngestionDemo' configuration interface. It has a navigation bar with five steps: 1 Definition, 2 Source, 3 Target (selected), 4 Transformation, and 5 Runtime Options. The 'Target Details' section shows 'Connection' set to 'Kafka' and 'Topic' set to 'DemoTopic1'. The 'Advanced Properties' section includes 'Producer Configuration Properties' set to 'key1=value1,key2=value2', 'Metadata Fetch Timeout in milliseconds' set to '5000', and 'Batch Flush Size in bytes' set to '1048576'.



# Mass Ingestion Files

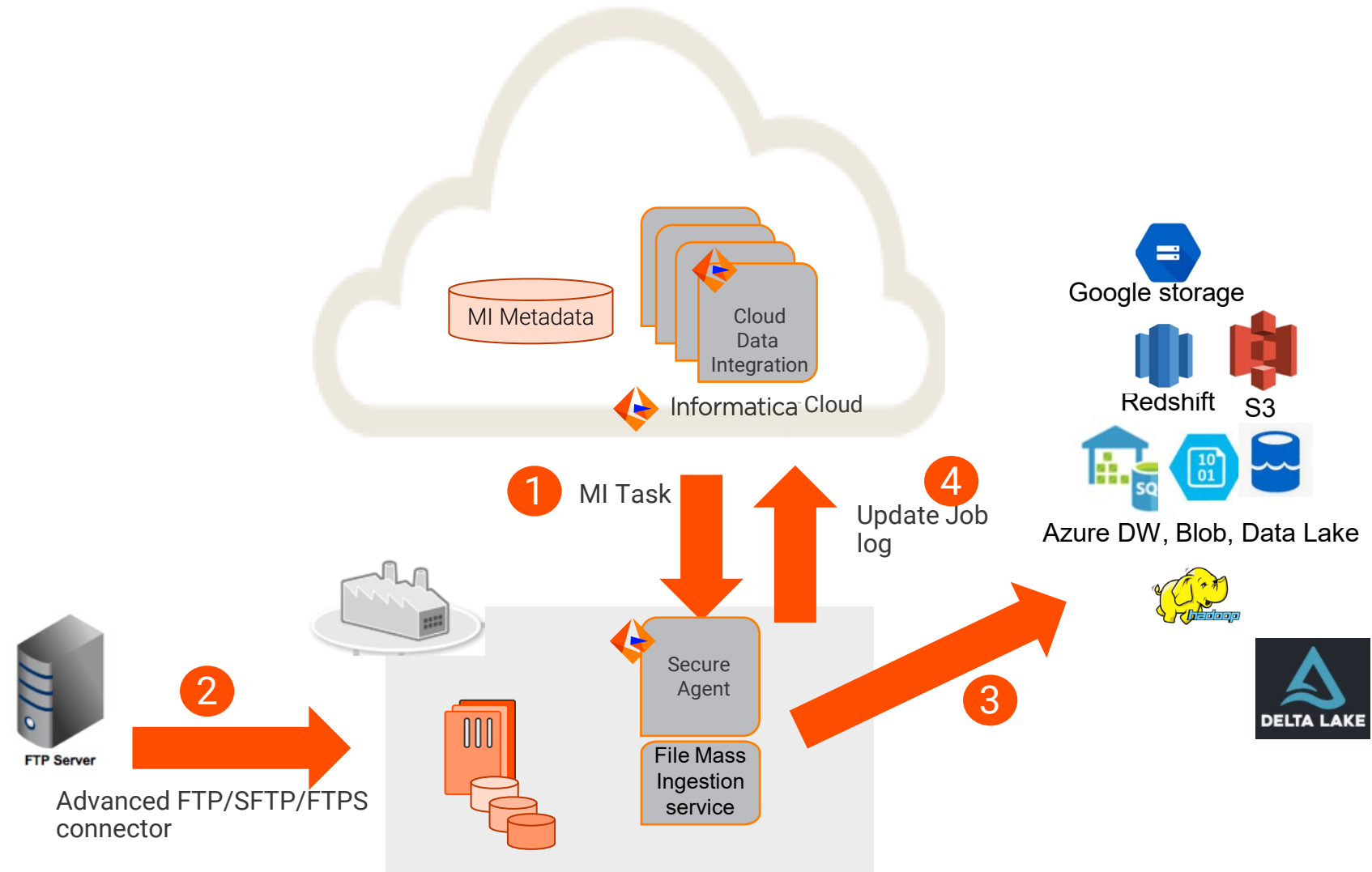
# Cloud Mass Ingestion - Files

Provides **file transfer capabilities** for exchanging files between on premise and Cloud repositories, using standard protocols

Transfer **any file type** with a **high performance and scalability**

Job and file level **tracking and monitoring**

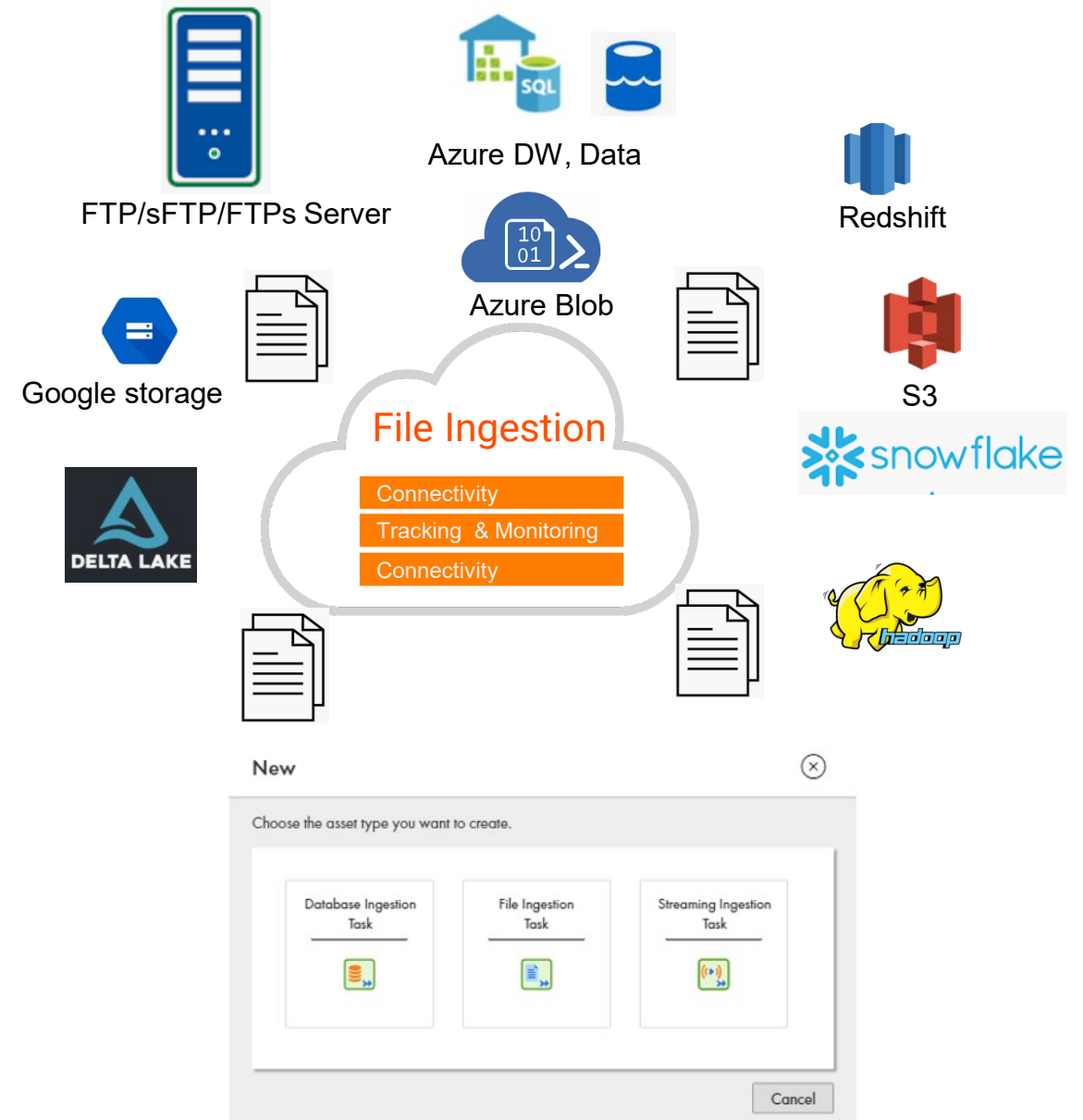
**Orchestrate** File transfer and ingestion in **hybrid/cloud** as **managed and secure** service



# Common Use Cases

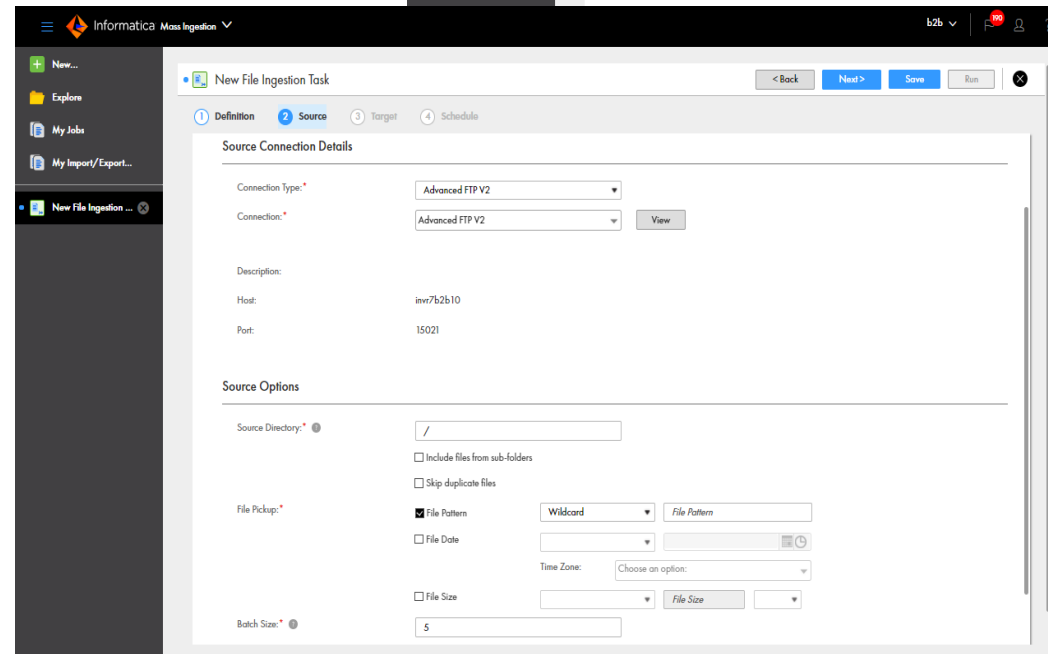
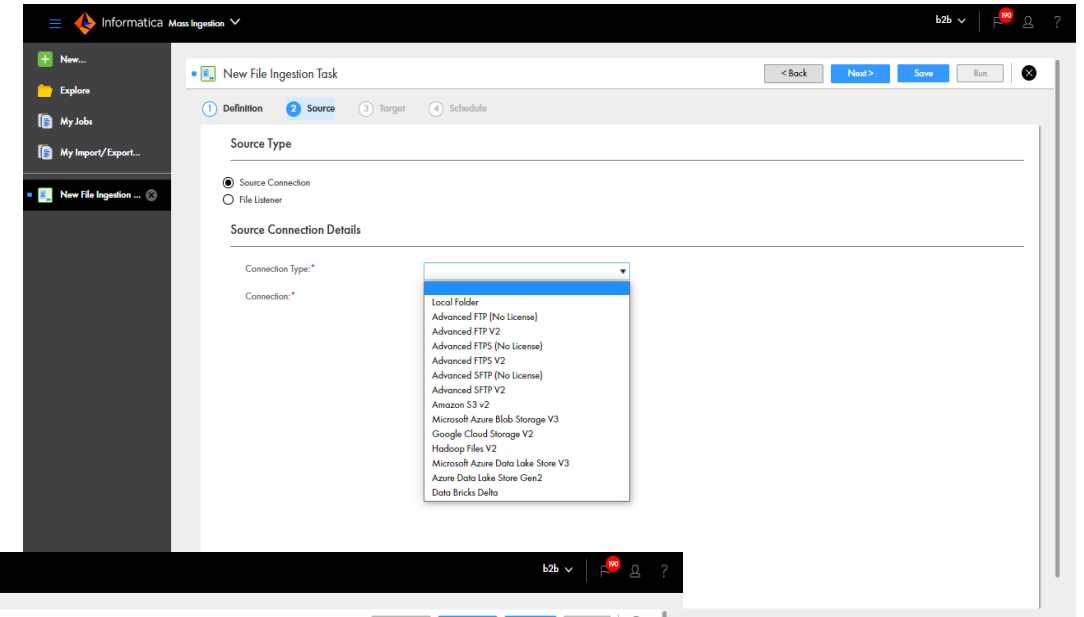
## ➤ Lake Ingestion

- ✓ Ingest data that arrive as files to Data lakes and repositories
- ✓ No data manipulation is needed. Focus on ingestion
- ✓ Transfer files from remote FTP/SFTP/FTPS servers
- ✓ Any file size, any file types



# Key Capabilities

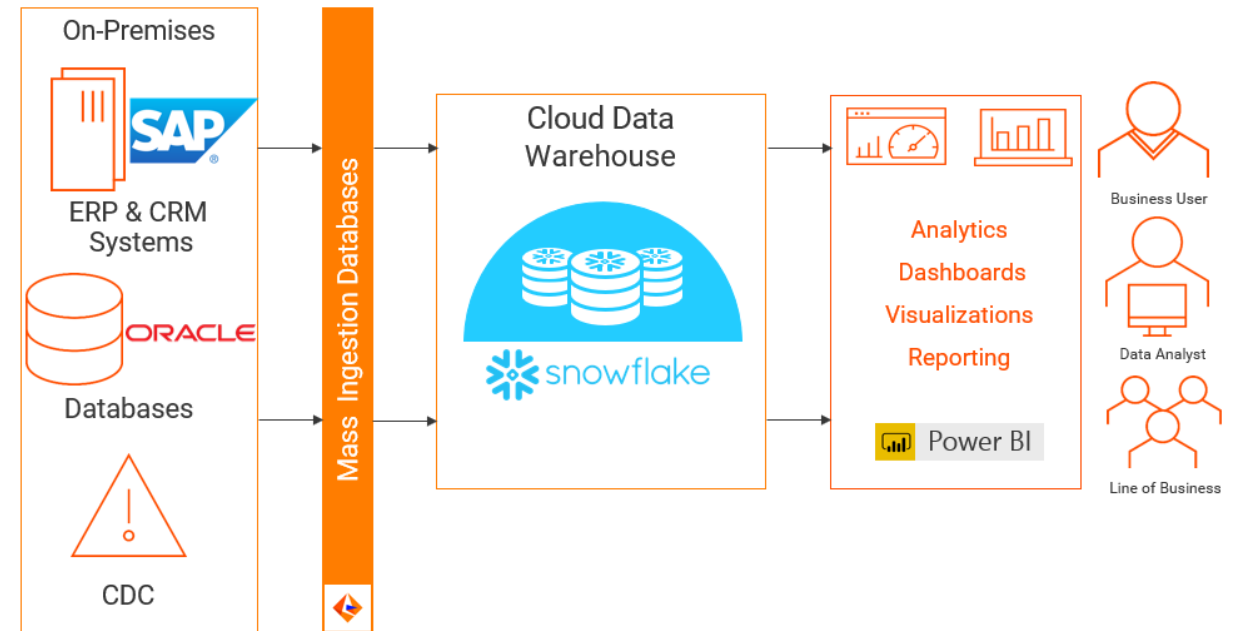
- Simple, wizard-based task definition
- Wide list of supported sources/targets
- Advanced, highly scalable connectors for handling FTP/SFTP/FTPS
- Filter files by file name pattern, file size, file date
- API, schedule or file event triggered
- File actions :
  - Compress/decompress (Zip, Gzip, Tar)
  - Encrypt/decrypt (PGP)
  - Virus Scan (ICAP)
- Highly scalable, any file type
- Unified monitoring and tracking experience
  - Tracking and monitoring - Job and file level



# Customer Stories & Architecture pattern

# Case Study: KLA Migrates 12 Years of Data to Cloud in One Weekend with Informatica

- Delivered trusted information to the data consumers for real-time analytics with Informatica and Snowflake
- Use case:
  - Continuous data replication from Oracle to Snowflake by capturing incremental changes and integrating them in real-time
- Solution:
  - Informatica Cloud Mass Ingestion - Databases
- Results:
  - **Expedited decision-making**, empowering sales and financial teams to drive growth
  - **Enabled two people to migrate** 12 years of ERP data into Snowflake over a weekend
  - **Improved customer satisfaction** using a single platform, consolidating workloads and eliminating data silos



# Case Study: University of New Orleans Increases Student Enrollment and Improves Retention

- The University of New Orleans Modernized its Cloud Data Warehouse with Informatica

- Use case:

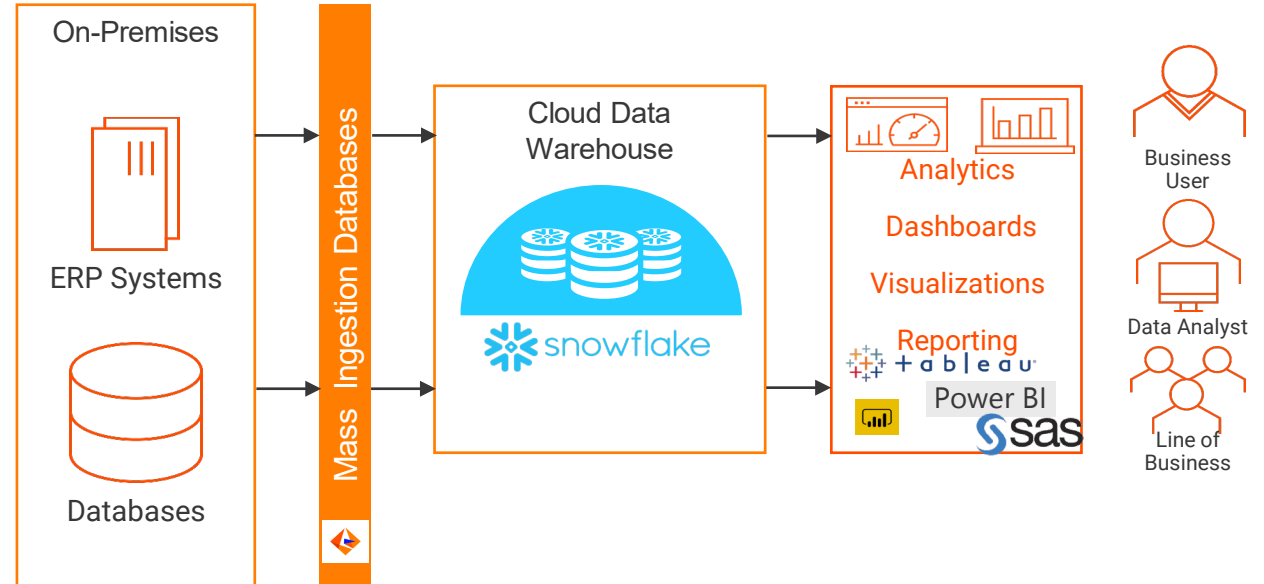
- Accelerate **Cloud Data Warehouse modernization** from Oracle to Snowflake

- Solution:

- Informatica Cloud Mass Ingestion - Databases

- Results:

- **Migrated thousands of tables** with complex data structures from Oracle to Snowflake with no hand-coding
- **Reduced manual effort by 90%** to focus on analytics for improving student recruitment, admission, and retention



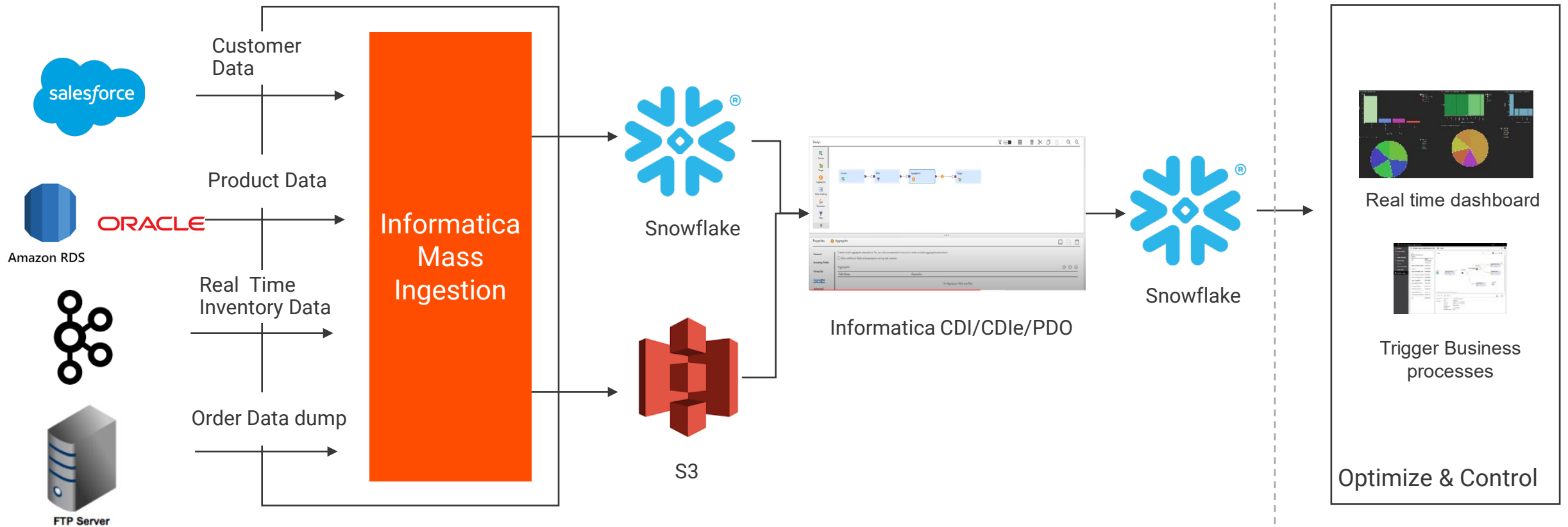


# Common Architecture Pattern

## Extract & Load

## Transform

## Analyze & Act



# Why Informatica Cloud Mass Ingestion?

The infographic features a central header 'Cloud Mass Ingestion' with a red icon of three vertical bars and a jagged arrow. Below it are four panels, each with a red icon and a label: 'Speed' (a speedometer), 'Simplicity' (three people in a circle), 'Scale' (a network diagram), and 'Flexibility' (two interlocking gears). The 'Speed' panel is highlighted with a red border.



## Build Data Ingestion Jobs in Minutes

The screenshot shows the 'Salesforce\_Snowflake\_Replication' job configuration wizard. It has four steps: 1. Definition, 2. Source, 3. Target, and 4. Schedule and Runtime Options. The 'Definition' step is active, showing fields for Name, Location, Runtime Environment, Description, and Load Type. The 'Speed' icon from the infographic is visible in the top left corner of the screenshot.

Build data ingestion jobs in minutes with a simple, easy to use **4-step wizard-based experience**

# Why Informatica Cloud Mass Ingestion?

Cloud Mass Ingestion

Speed

Simplicity

Scale

Flexibility



## Simplify your Data Ingestion Tasks

New

Choose the asset type you want to create.

Application Ingestion Task

Database Ingestion Task

File Ingestion Task

Streaming Ingestion Task

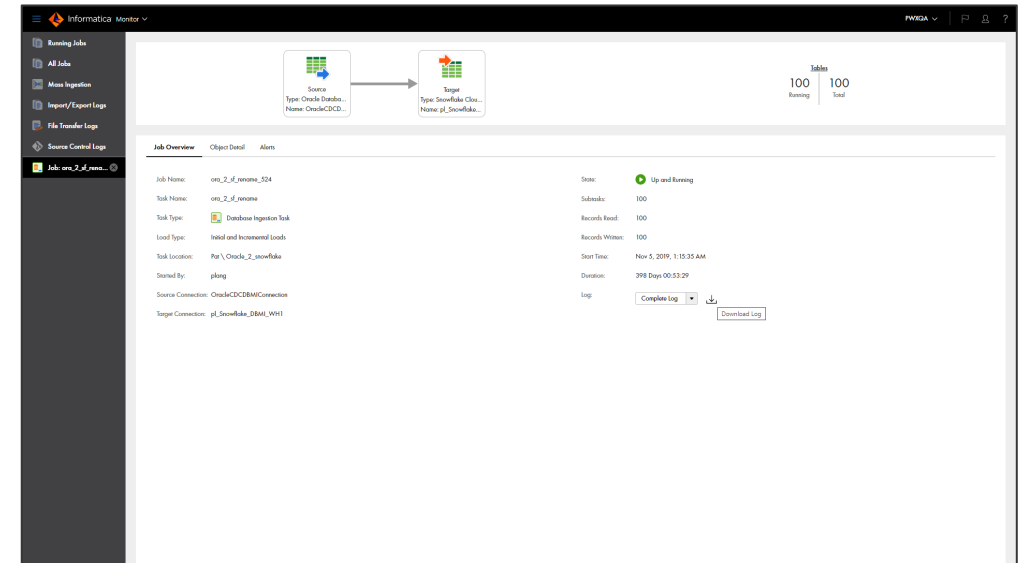
Cancel

Simplify data ingestion with a **single, unified cloud-native data ingestion solution** with out-of-the-box connectivity

# Why Informatica Cloud Mass Ingestion?




## Data Ingestion at Scale



Ingest terabytes of **any data, any pattern, at any latency at scale** in real-time and batch with no data limit



# Why Informatica Cloud Mass Ingestion?



## Cloud Mass Ingestion



### Speed



### Simplicity



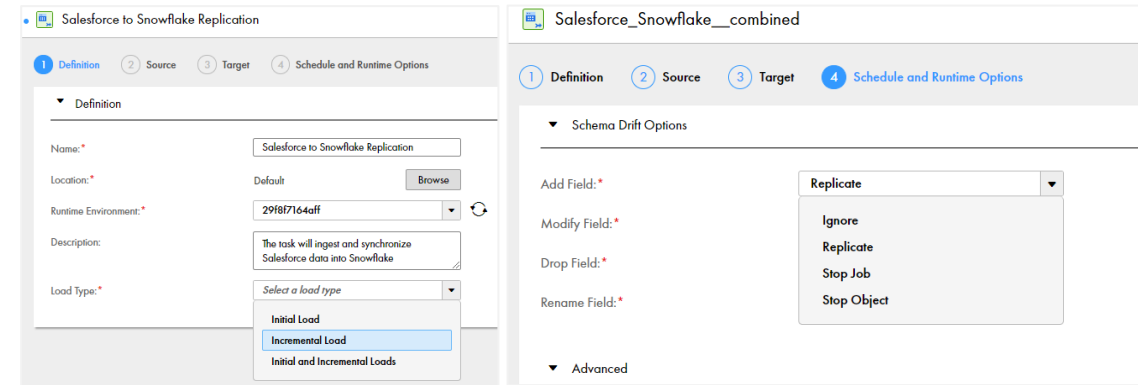
### Scale



### Flexibility



## Flexibility to Track, Capture and Update Changes



Track, capture, and update **changed data** in real-time with **automatic schema drift** support to accelerate application replication and synchronization use cases

# Summary



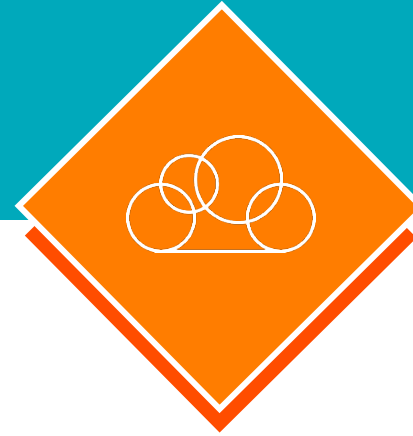
## Cloud Native Ingestion

- Unified service for ingestion from various sources
- Orchestration for ingestion from variety of patterns
- Support for CDC and Schema Drift



## Connectivity

- On-prem database and CDC
- On-prem and cloud files
- IoT and streaming
- Cloud data lakes, data warehouse and messaging hub



## Wizard-driven Design

- Simple, easy-to-use wizard
- Edge transformations
- Intent-driven ingestion



## Real-time Monitoring

- Pictorial view of the ingestion job
- Real-time flow visualization
- Lifecycle management

# Free 30-day Trial

## Free 30-Day Trial: Cloud Mass Ingestion

Ingest any data at scale to make it immediately available for real-time processing, database replication, and application synchronization. Use an automated, wizard-based approach to efficiently ingest databases, applications, files, and streaming data at scale into a cloud or on-premises data lakes or data warehouses.

In this trial, you can:

- Transfer any size or type of file with high performance and scalability
- Collect, filter, and combine data from streaming sources such as IoT endpoints and messaging systems
- Ingest data at scale from common relational databases, SaaS, and on-prem applications and propagate the data into a cloud data warehouse, cloud data lake, and message hub
- Track, capture, and update changed data in real-time with automatic schema drift support

<https://www.informatica.com/trials/data-ingestion.html>

A photograph of a hand raised in a classroom or meeting setting. The hand is in the foreground, and the background is blurred, showing other people and a screen. The word "Questions?" is overlaid in white text on the hand.

Questions?



# Change Data Capture (CDC)

Replication

Ingestion

Schema Drift

Kafka

ELT

Streaming Data

File Ingestion

Data Synchronization

# Thanks



# Simple, Autonomous, Scalable Data Integration for Everyone

**Use Cases**



**Data Consumers**



**Industries**



**Data Patterns**




**SIMPLE**

Simple and easy data access for all data users

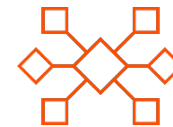
Simple from onboarding to production



**AUTONOMOUS**

Improve productivity for all data practitioners

Intelligent automation across the entire data management lifecycle






**SCALABLE**

Scalable from departmental to enterprise workloads

Scale with optimized data management engine to process data anywhere, anyhow

**Business Value**

-  Reduce Complexity
-  Lower resource need
-  Govern your costs

# Simple, Autonomous, Scalable Data Integration for Everyone

**Use Cases**



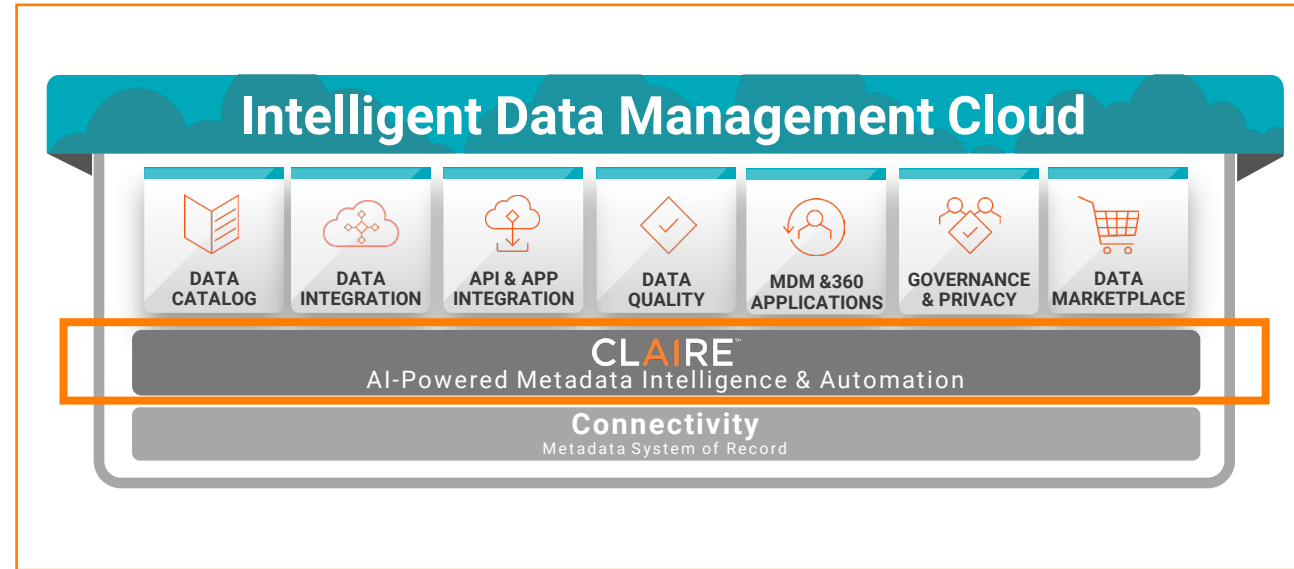
**Data Consumers**



**Industries**



**Data Patterns**






**SIMPLE**

**AUTONOMOUS**

**SCALABLE**

**Business Value**

-  Reduce Complexity
-  Lower resource need
-  Govern your costs