Engineering Machine Learning Data Pipelines

Streaming Data Changes

Paige Roberts
Integrate Product Marketing Manager

Common Machine Learning Applications

- Anti-money laundering
- Fraud detection
- Cybersecurity
- Targeted marketing
- Recommendation engine
- Next best action
- Customer churn prevention
- Know your customer

Data Engineer to the Rescue

Data Scientist

- Expert in statistical analysis, machine learning techniques, finding answers to business questions buried in datasets.
- Does NOT want to spend 50 – 90% of their time tinkering with data, getting it into good shape to train models – but frequently does, especially if there's no data engineer on their team.
- When machine learning model is trained, tested, and proven it will accomplish the goal, turns it over to data engineer to productionize. Not skilled at taking the model from a test sandbox into production, especially not at large scale.

Data Engineer

- Expert in data structures, data manipulation, and constructing production data pipelines.
- WANTS to spend all of their time working with data, but usually has more on their plate than they can keep up with. Anything that will speed up their work is helpful.
- In most successful companies, is involved from the beginning. First gathers, cleans and standardizes data, helps data scientist with feature engineering, provides top notch data, ready to train models.
- After model is tested, builds robust high scale, data pipelines to feed the models the data they need in the correct format in production to provide ongoing business value.

Five Big Challenges of Engineering ML Data Pipelines

1. **Scattered and Difficult to Access Datasets**
   Much of the necessary data is trapped in mainframes or streams in from POS, web clicks, etc. all in incompatible formats, making it difficult to gather and prepare the data for model training.

2. **Data Cleansing at Scale**
   Data quality cleansing and preparation routines have to be reproduced at scale. Most data quality tools are not designed to work on that scale of data.

3. **Entity Resolution**
   Distinguishing matches across massive datasets that indicate a single specific entity (person, company, product, etc.) requires sophisticated multi-field matching algorithms and a lot of compute power. Essentially everything has to be compared to everything else.

4. **Tracking Lineage from the Source**
   Data changes made to help train models have to be exactly duplicated in production, in order for models to accurately make predictions on new data, and for required audit trails. Capture of complete lineage, from source to end point is needed.

5. **Need for Ongoing Real-Time Changed Data Capture and Streaming Data Capture**
   Tracking and detection needs to happen very rapidly. Current transactions need to be constantly added to combined datasets, prepared and presented to models as close to real-time as possible.

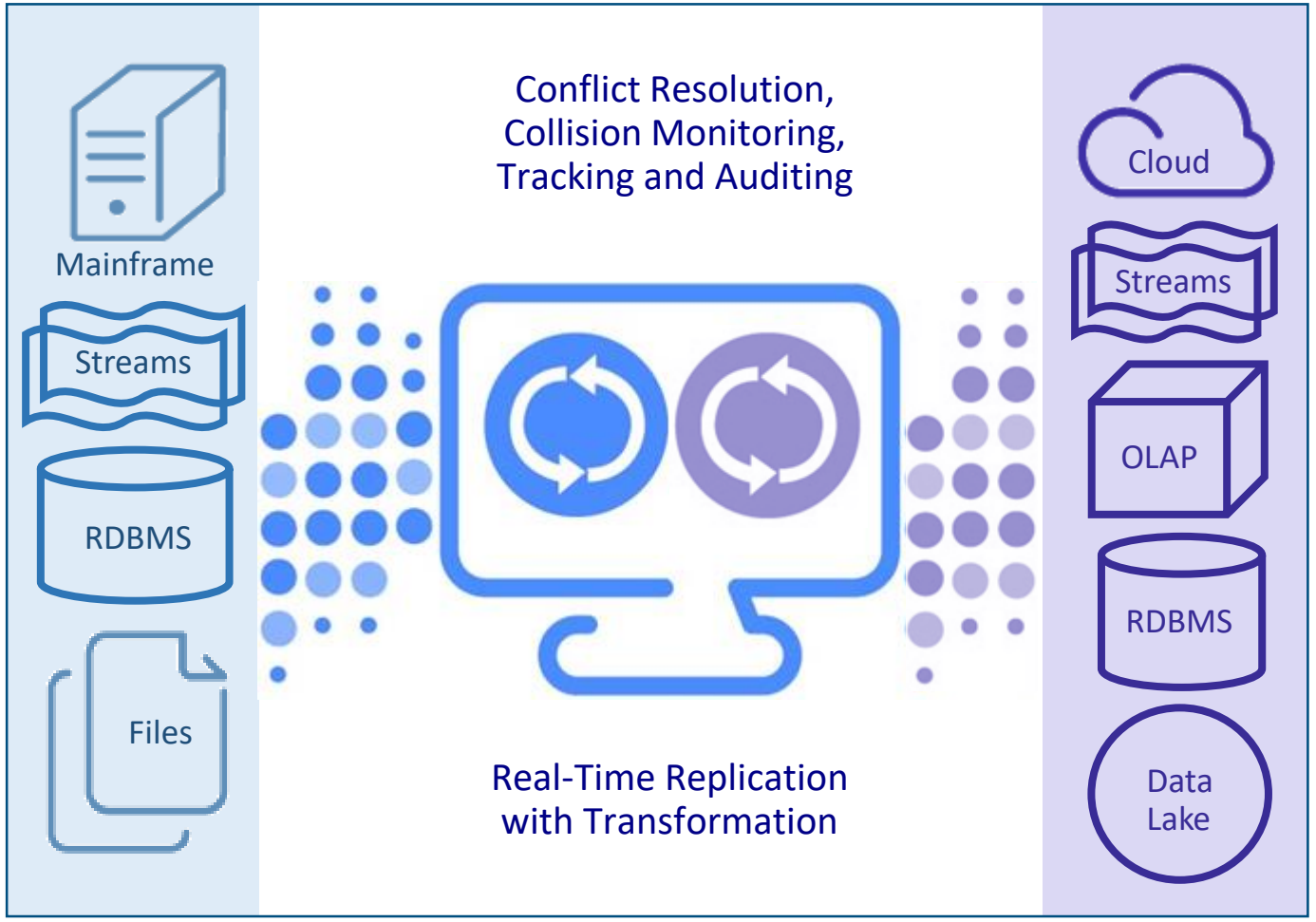
DMX Change Data Capture

Reliable transfer of data you can trust even if connectivity fails on either side.

- Auto restart.
- No data loss.

Keep data in sync in real-time

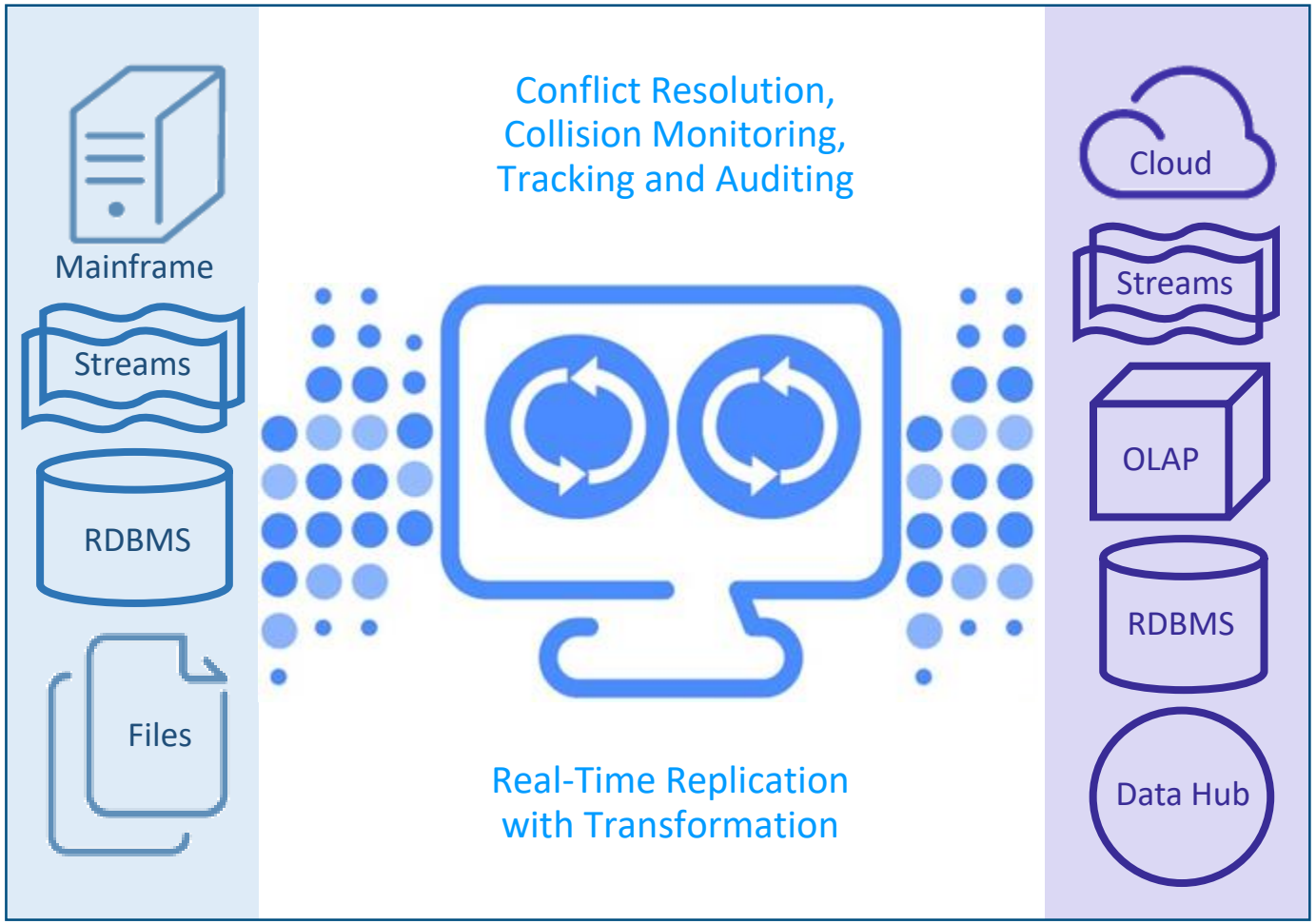
- Without overloading networks.
- Without affecting source database performance.
- Without coding or tuning.



DMX Change Data Capture Sources and Targets

SOURCES

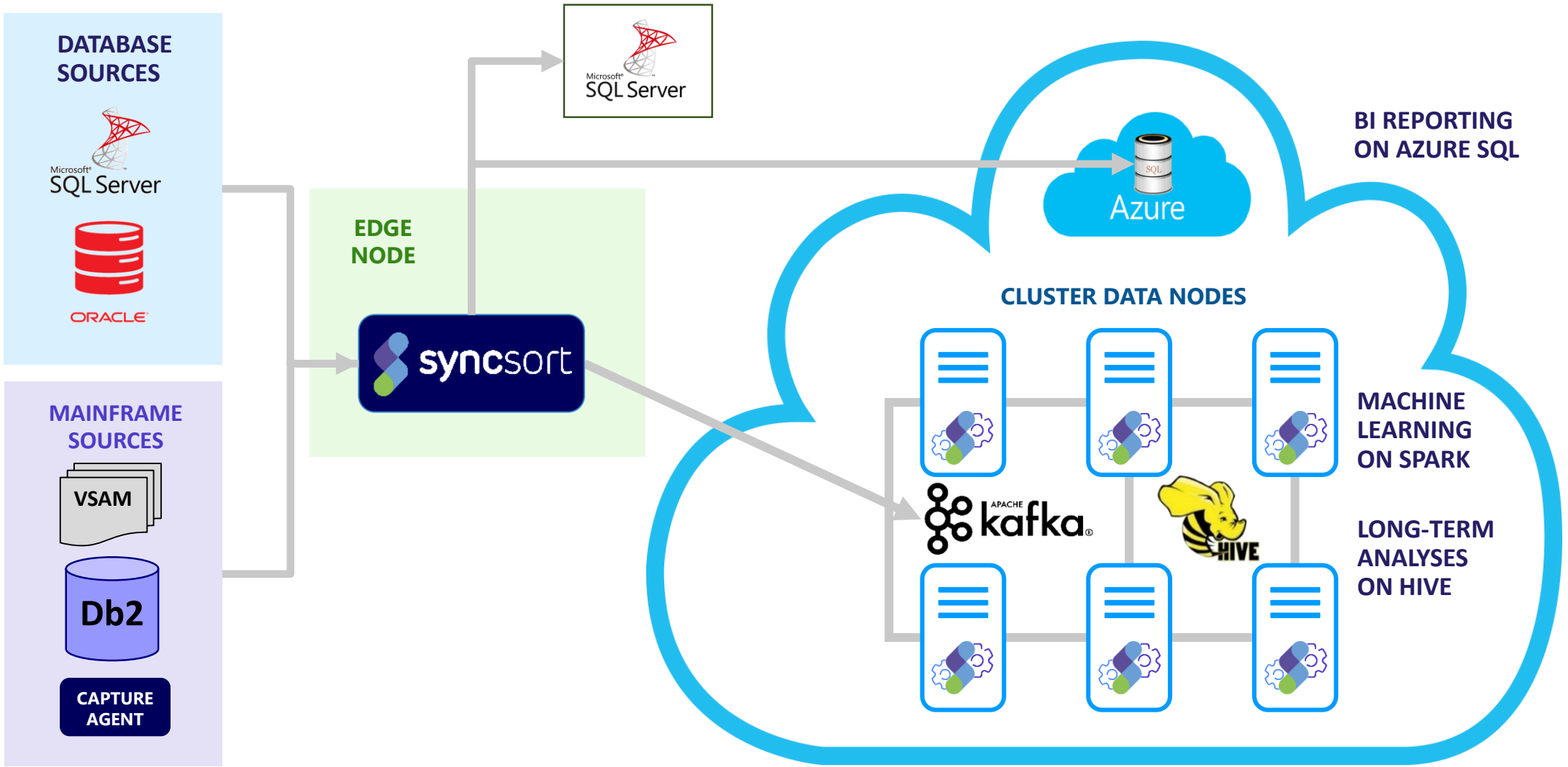
- IBM Db2/z
- IBM Db2/i
- IBM Db2/LUW
- VSAM
- Kafka
- Oracle
- Oracle RAC
  Real Application Clusters
- MS SQL Server
- IBM Informix
- Sybase



TARGETS

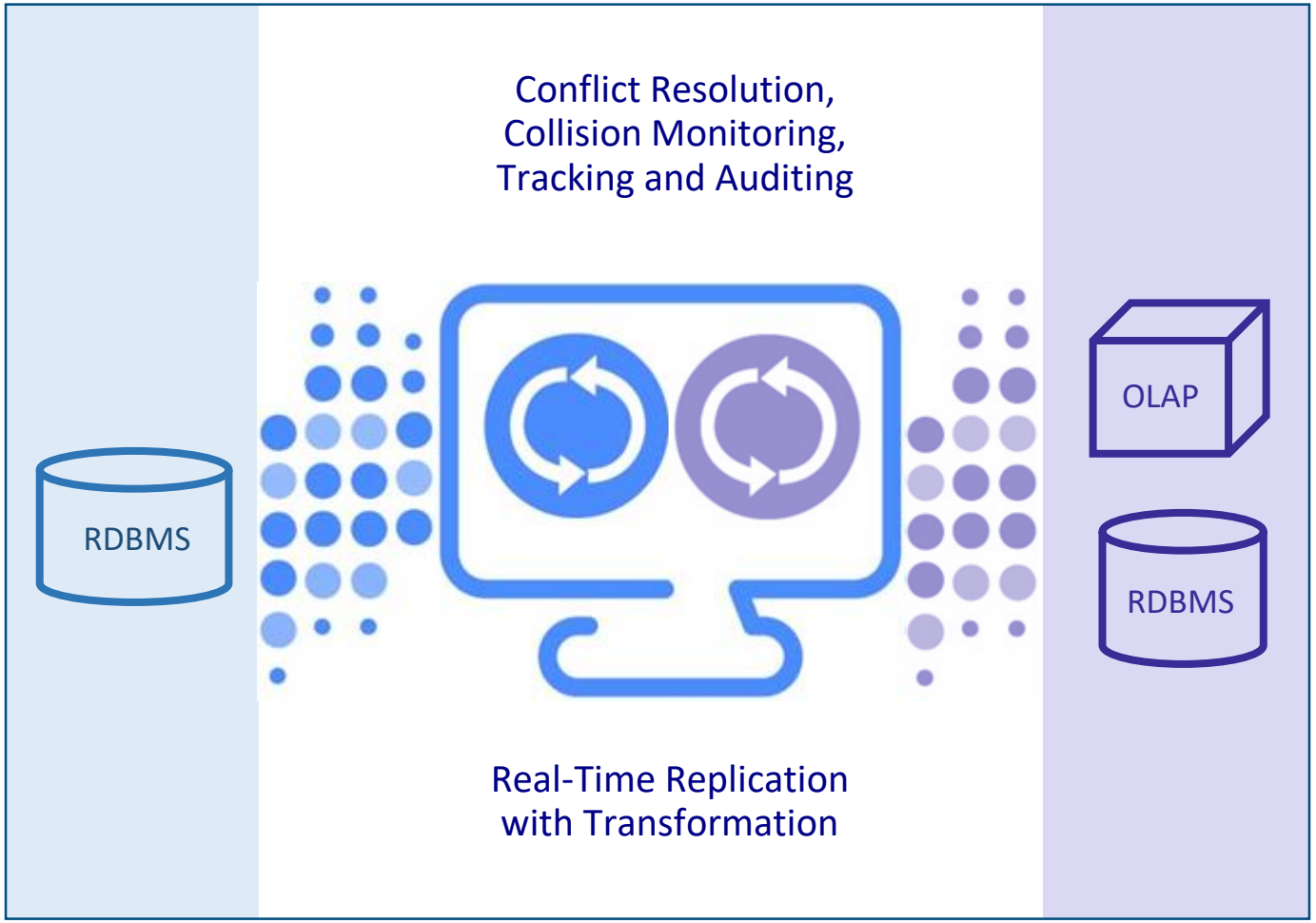
- Kafka
- Amazon Kinesis
- Teradata
- HDFS
- Hive
  (HDFS, ORC, Avro, Parquet)
- Impala
  (Parquet, Kudu)
- IBM Db2
- SQL Server
- MS Azure SQL
- PostgreSQL
- MySQL
- Oracle
- Oracle RAC
- Sybase
- And more …

Simple Customer Example Architecture



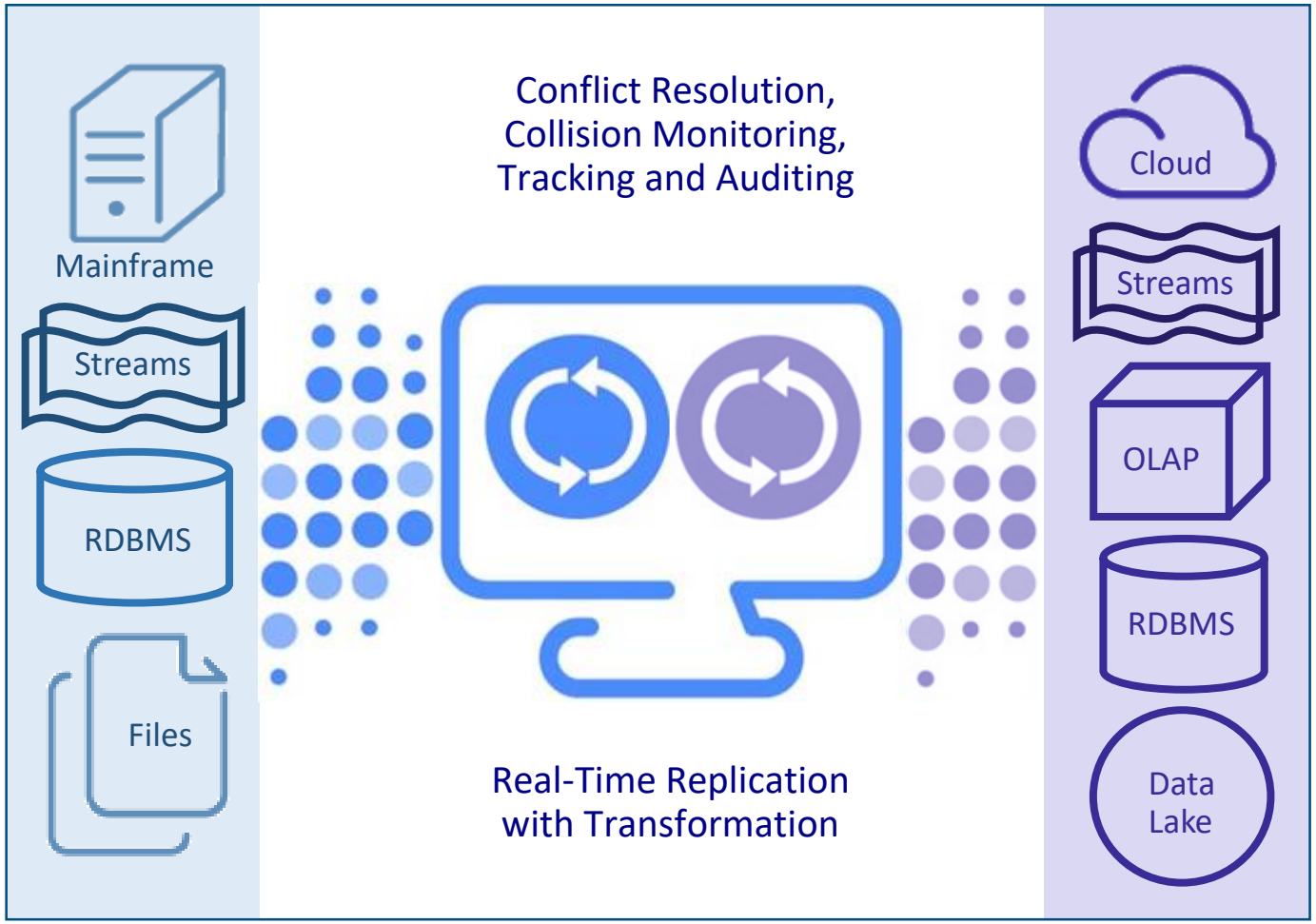
Log-Based Database to Database

- Captures database changes as they happen
- Transforms and enhances data during replication
- Minimizes bandwidth usage with LAN/WAN friendly replication
- Ensures data integrity with conflict resolution and collision monitoring
- Enables tracking and auditing of transactions for compliance
- Latency – sub-second



Anything to Stream, Stream to Anything, Stream to Stream

- Real-time capture
- Minimizes bandwidth usage with LAN/WAN friendly replication
- Parallel load on cluster
- Updates HDFS, Hive or Impala, backed by HDFS, Parquet, ORC, or Kudu.
- Updates even versions of Hive that did not support updating
- Latency – Real-time, actual SLA varies depending on update speed of target, stream settings, etc. Usually, seconds.



Case Study:

Global Hotel Data Kept Current On the Cloud

CHALLENGE

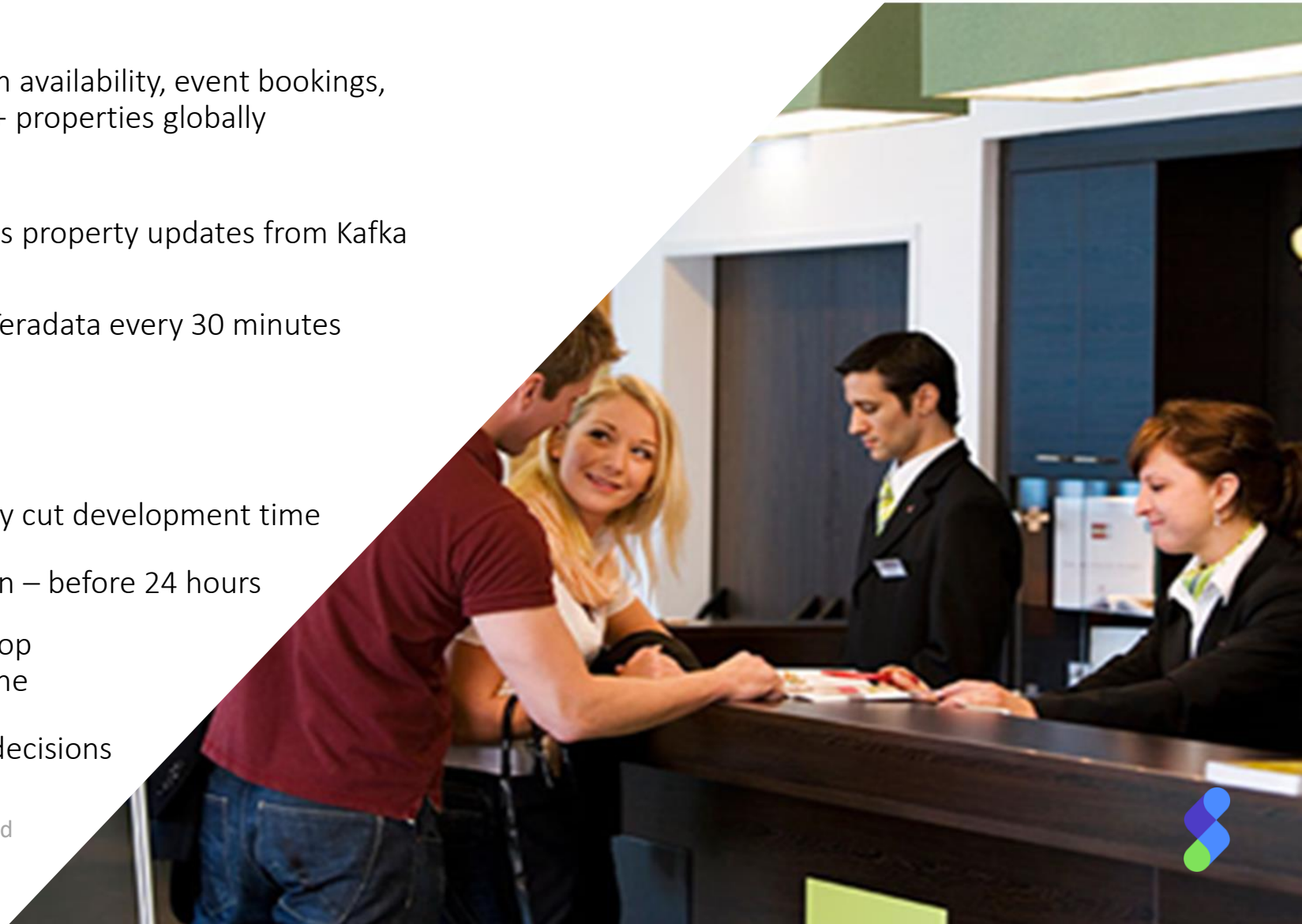
- More timely collection & reporting on room availability, event bookings, inventory and other hotel data from 4,000+ properties globally

SOLUTION

- Near real-time reporting - DMX-h consumes property updates from Kafka every 10 seconds
- DMX-h processes data on HDP, loading to Teradata every 30 minutes
- Deployed on Google Cloud Platform

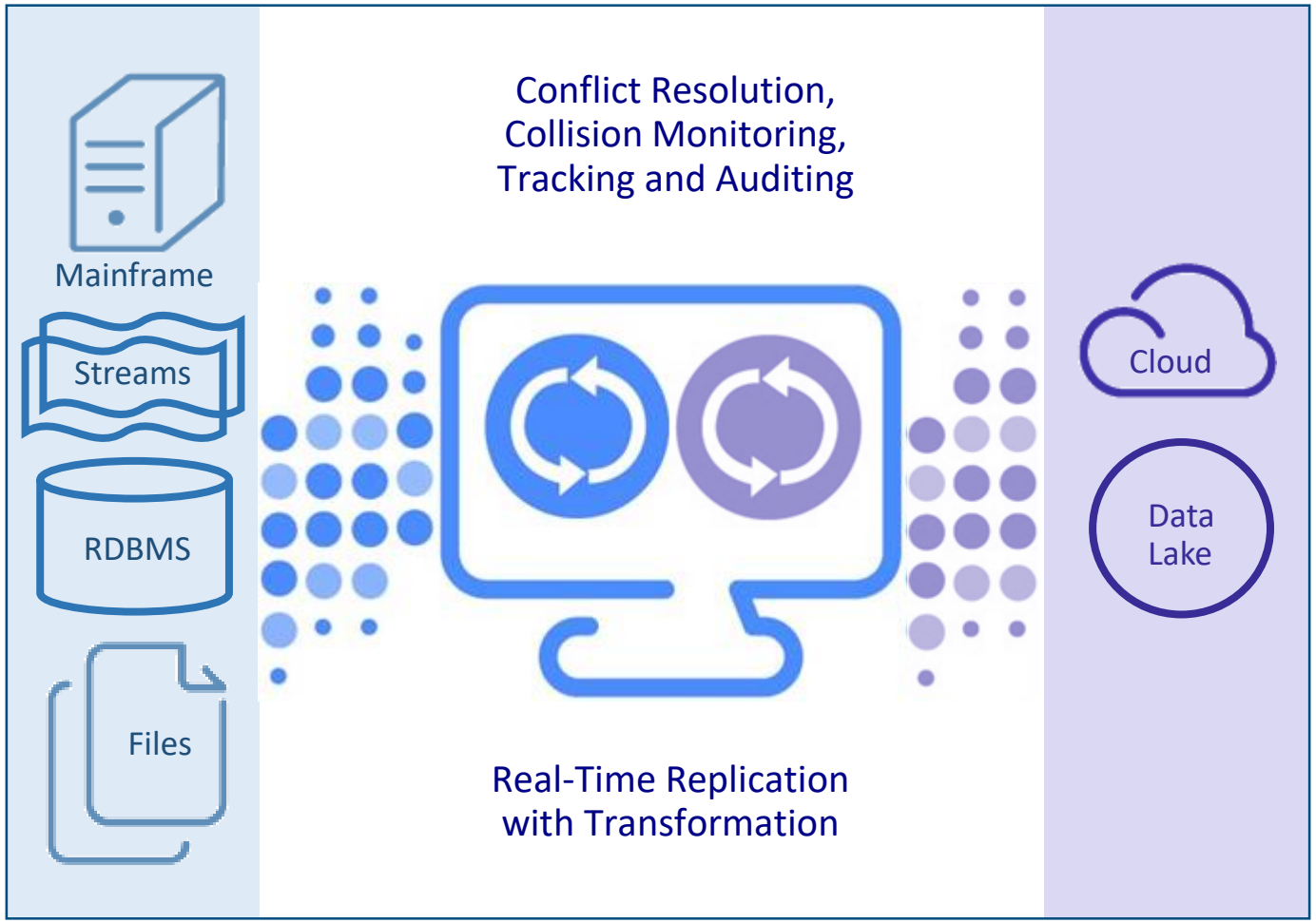
BENEFITS

- **Time to Value:** DMX-h ease of use drastically cut development time
- **Agility:** Global reports updated every 30 min – before 24 hours
- **Productivity:** Leveraging ETL team for Hadoop (Spark), visual understanding of data pipeline
- **Insight:** Up-to-date data = better business decisions = happier customers



Log-Based Change Capture to Hadoop

- Real-time capture
- Minimizes bandwidth usage with LAN/WAN friendly replication
- Parallel load on cluster
- Updates HDFS, Hive or Impala, backed by HDFS, Parquet, ORC, or Kudu.
- Updates even versions of Hive that did not support updating
- Latency – Minutes (< 3)



Guardian Life Insurance

"We found DMX-h to be very usable and easy to ramp up in terms of skills. Most of all, Syncsort has been a very good partner in terms of support and listening to our needs."

– Alex Rosenthal, Enterprise Data Office

GUARDIAN®

Need to enable ML, visualization and BI on broad range of datasets, and reduce time-to-market for analytics projects.

- Reduce data preparation, transformation times – long delay before new analyses.
- Make data assets available to whole enterprise – including Mainframe data.

SOLUTION

- Hadoop, NoSQL data lake.
- DMX DataFunnel quickly ingested hundreds of database tables at push of a button.
- DMX-h adds new transformed, standardized data with each new project.
- DMX Change Data Capture pushes changes from DB2 and other sources to the data lake in real-time. Current data up-to-the minute.

Data Marketplace – centralized, reusable, up-to-the-minute current, searchable, accessible, managed, trustworthy data for analytics.

Fast Time-to-Market for new analytics and reporting.

Symphony Health Provides Healthcare Data Science with DMX-h

"Before, part of the data wasn't available for a day, and other parts, not for a week. Now it's all available for analysis within minutes of the data arriving."

Robert Hathaway
Senior Manager Big Data

Symphony Health
A PRAHEALTHSCIENCES COMPANY

Data scientists need fresh data and constantly seek to do new analyses.

Expensive Oracle solution took days to get data to data scientists. Required new schemas from DBA work queues for each new analysis.

Hadoop helped, but expensive ETL tool bottlenecked all data processing on overloaded edge node. Blamed poor performance on unoptimized workflows.

SOLUTION:

- DMX-h
- Apache Spark on Cloudera CDH
- Amazon Redshift

Costs saved both on Hadoop storage and DMX-h data processing. And, data scientists can define their own new schemas – no waiting.

DMX-h also does low latency push to Amazon Redshift for fast, advanced interactive queries, and so Symphony Health can display results to clients in web application.

Data scientists can ask more questions now, find things out sooner.

Data available for analysis in minutes, not days.

- **No tuning required:** *"DMX-h is already optimized. We use its Intelligent Execution and it just performs."*
- **Average 3 - 5X processing speed increase:** On one project, dropped processing times from 20 minutes to 20 seconds.
- **No lock-in** – If part of a workflow works better in something like PySpark, DMX-h makes it easy to plug in.

"We get the same end result, faster, cheaper, and with a bigger pool of developers to draw from who can do the work. I'm a C# and Java developer who even knows some Scala, and I still like using DMX-h because I can get a lot more done in the same time."


thank you
danke
謝謝
ngiyabonga
teşekkür ederim
gracias
merci
obrigado
go raibh maith agat
dziękuję
bedankt
спасибо
рахмат
Баярлалаа
dank je
tapadh leat
mochchakkeram
terima kasih
sukriya
kop khun krap
arigatô
dakujem
grazie
takk
tack
barka
mersi
spas
welalin
mahalo
paldies
grazzi
misaotra
matondo
vinaka
blagodaram
faafetai lava
kiitos
dankie
dhanyavad
nanni
nandri
mauruuru
hvala
köszönöm
gràcie
bayarlalaa
enkosi
akun
dankon
aciu
djiere dieuf
tau
дякую
mamnun
chnorakaloutioun
gratias ago
gràcies
sulpáy
taiku
sagolun
sobodi
dèkuji
mèsi
didi madloba
najis tuke
kam sah hamnida
rahmat
tanemirt
rahmet
ありがとう
diolch
dhanyavadagalu
merci
xièxie
감사합니다
তোমাকে ধন্যবাদ
ευχαριστώ
хвала
asante
manana
obrigada
tenki
chokrane
murakoze
trugarez
мерси
shukriya
mercé


syncsort
advancing data