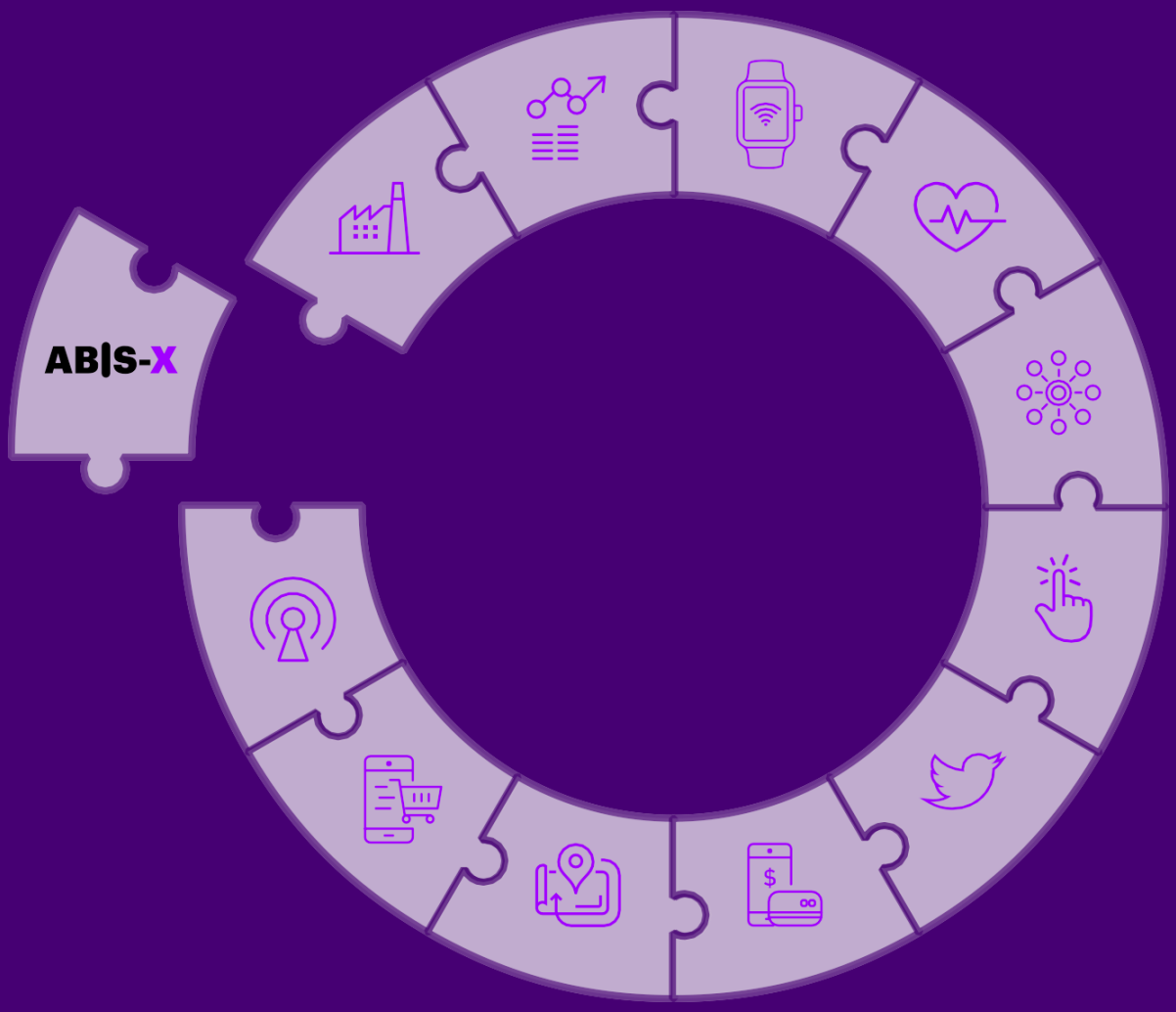
accenture

ACCENTURE BIG DATA INTEGRATOR SUITE

ABIS-X

A SUITE FOR BIG DATA INGESTION, CONNECTING SOURCES TO DIFFERENT STORAGE GO FAST, AT SCALE. DON'T CODE!

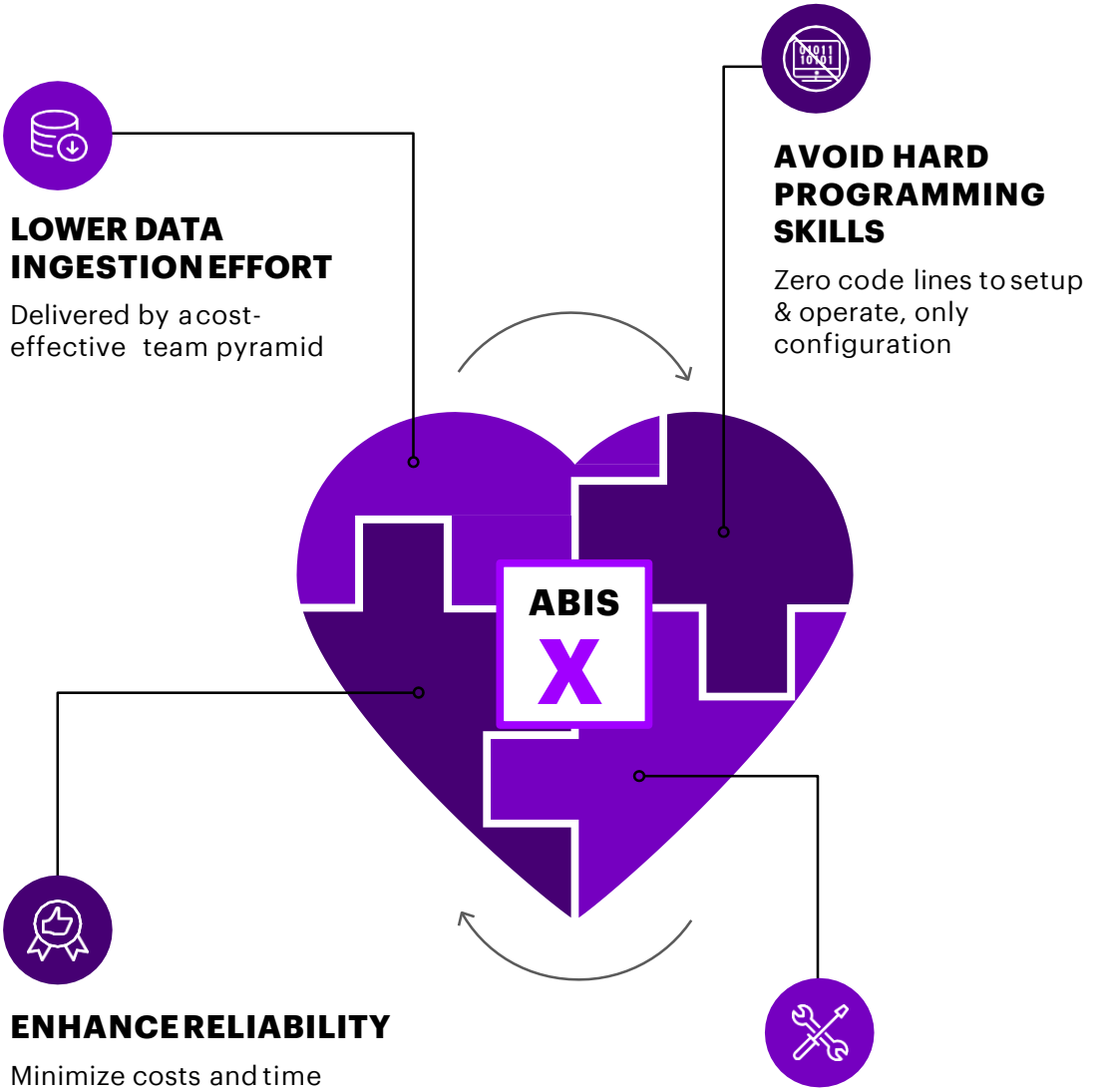


WE INSTANTLY REALIZED NEEDS HAD TO BE ADDRESSED PROCESSING 100'S OF DATAFLOWS

WE SOUGHT TO FIT THEM SMART. ZERO-CODE APPROACH MAKES A DIFFERENCE.

THE ABIS-XTEAM

ZERO CODE-APPROACH



SOME ANSWERS TO QUESTIONS YOU MAY HAVE IN MIND...

1. **I NEED TO PERFORM BIG DATA INGESTION, MAY ABIS-X HELP ME?**
   Yes, ABIS-X can be used in any Big Data ingestion activity as it has been developed with market standard open source software.

2. **I ALREADY USE AN ETL TOOL ON A BIG DATA ENVIRONMENT, CAN I USE ABIS-X?**
   Yes, ABIS-X can be used jointly with any ETL tool on the market.

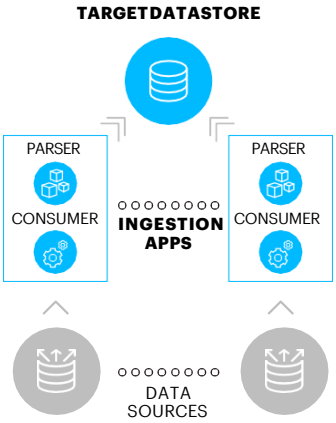
3. **WHAT KIND OF SKILLS ARE NEEDED TO USE ABIS-X?**
   ABIS-X doesn't require any hard programming skill to be used, as additional ingestion flows are set up by simple, text format, configuration files.

4. **I NEED A CAPABILITY NOT CURRENTLY IMPLEMENTED ON ABIS-X.**
   ABIS-X can be quickly enhanced to fit your needs at the best.

5. **WHY SHOULD I USE ABIS-X?**
   ABIS-X is a plug and play centralized engine that allows to reduce delivery costs, maintenance costs and overall TCO while improving reliability and scalability.

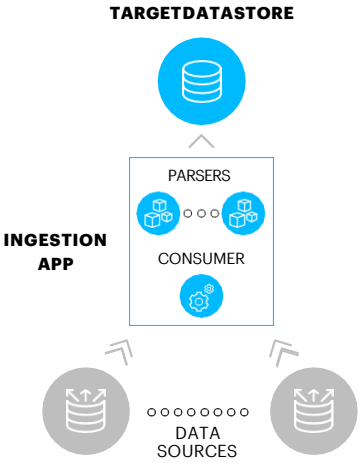
ANY OTHER OPTIONS?

INDIVIDUAL PER-FLOW SILOED APPS



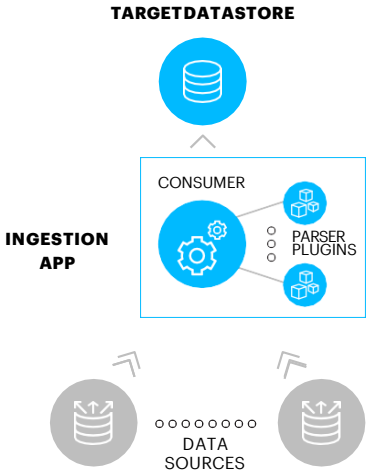
Individual **siloed ingestion apps**, that **require coding**, consume and parse a data flow each

PER-FLOW PARSERS EMBEDDED IN CENTRALIZED APP



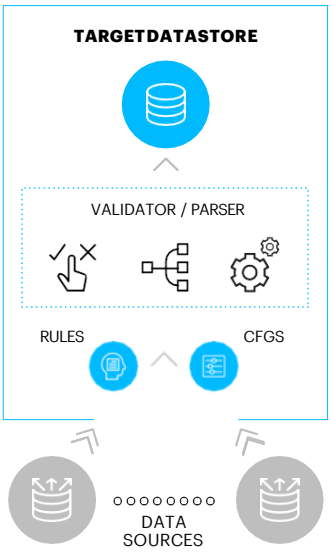
Data flow-specific embedded parsers –that **require coding** – drive centralized ingestion app

PER-FLOW PARSER PLUGINS TO CENTRALIZED APP



Data flow-specific parser plugins – that **require coding** – drive centralized ingestion app

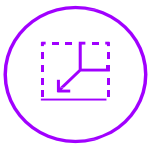
ABIS-X



The parser and a **rich** set of **configurable capabilities** fit all data flows:
ZERO CODING

REUSABLE AND SCALABLE BY DESIGN

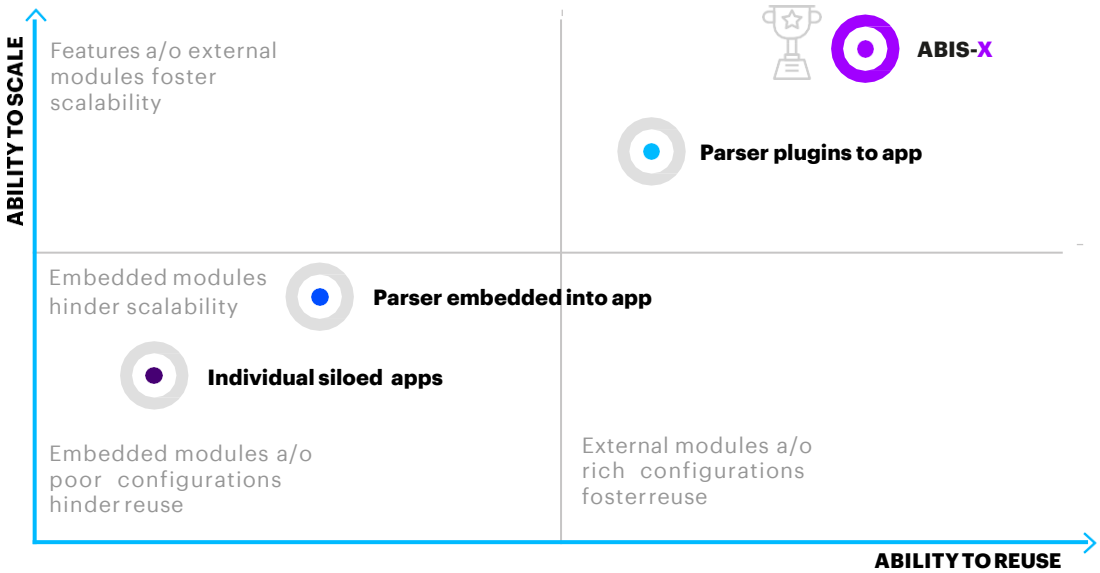
TOP REUSABILITY



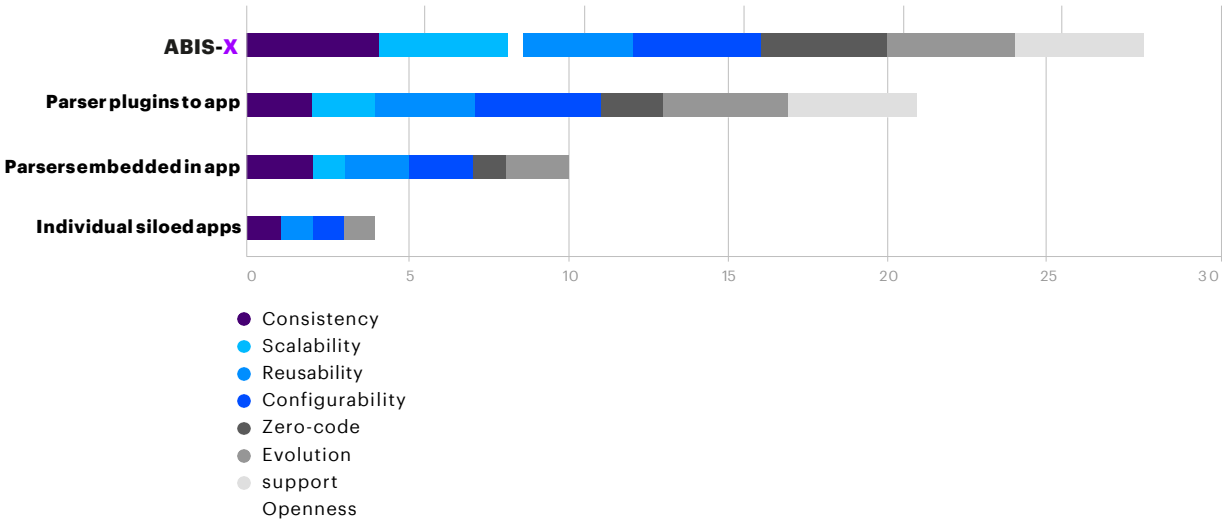
TOP SCALABILITY

TOP CONSISTENCY

POSITIONING BY REUSE AND SCALABILITY



CAPABILITIES ACROSS SOLUTIONS



OTHER OPTIONS STRUGGLE TO KEEPUP...

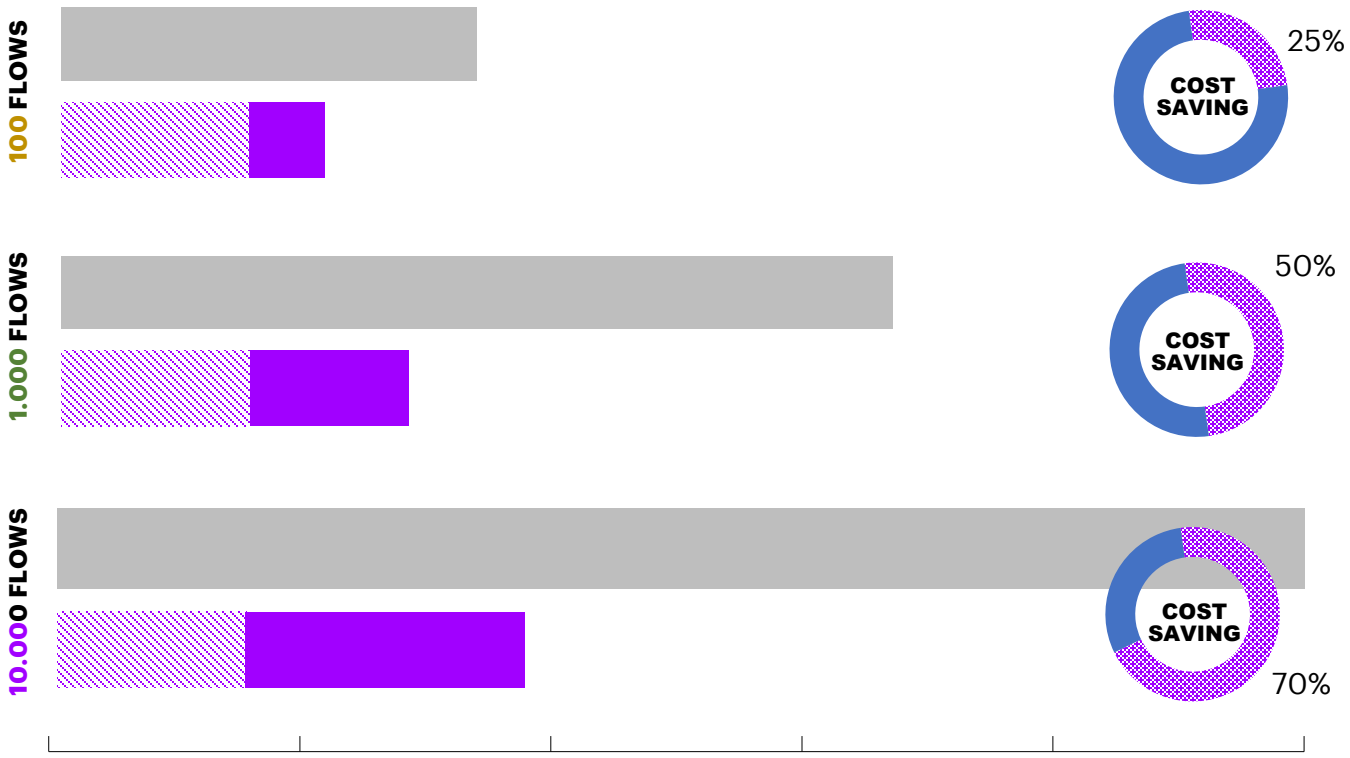
	INDIVIDUAL SILOED APPS	PARSERS EMBEDDED IN APP	PLUGINS EMBEDDED IN APP	ABIS-X
CONSISTENCY	★☆☆☆	★★☆☆	★★☆☆	★★★★
SCALABILITY	☆☆☆☆	★★☆☆	★★☆☆	★★★★
REUSABILITY	★☆☆☆	★★☆☆	★★★☆	★★★★
CONFIGURABILITY	★☆☆☆	★★☆☆	★★★★	★★★★
ZERO-CODE	☆☆☆☆	★☆☆☆	★★☆☆	★★★★
EVOLUTION SUPPORT	★☆☆☆	★★☆☆	★★★★	★★★★
OPENNESS	☆☆☆☆	☆☆☆☆	★★★☆	★★★★
OVERALL EASE-TO-DELIVER				WOW!

ABIS-X

STANDS OUT OF THE BUNCH

HOW MUCH YOU CAN SAVE

COST/TIME OF DELIVERY OF ABIS-X VS. STANDARD ETL SCENARIO (*)



(*) Saving forecast; actual saving depends from specific context and ABIS-X module used.

STANDARD ETL | **ABIS-X SET UP** | **ABIS-X SCALE OUT**

TOP SAVINGS

UP IN MINUTES

POWERFUL ACCELERATOR

ONLY CONFIGURATION

SMARTEST PYRAMID TO SET UP

BASIC PYRAMID TO SCALE

DELIVERY ACCELERATOR DRAMATICALLY SAVES COST & TIME

- Set **ABIS-X** up via **Linux installer** and ingest **tens of data flows** in **short time**
- **Zero-code application** only requires configuration file update
- **Basic Linux & Big Data skills** are required for installation and setup
- Delivered by a **cost-effective Team pyramid**

ABIS-X DELIVERY

1. ENGAGE ABIS-X TEAM
Get in touch, explain your use case, business goals, technical landscape. Get support in fitting ABIS-X to your needs, at best.

2. ANALYZE RAW DATA
Review the data schema you have been required to ingest, understanding features, volumes, schemas, key-candidate fields.

3. INSTALL AND CONFIGURE
Install ABIS-X, or be supported to, and configure its rich general and specific property set to fit the identified case.

4. RUN ABIS-X, APPS & MODELS
Start ABIS-X: it will nicely, reliably and consistently ingest data into storage, ready for your apps & ML models.

5. CHECK RESULTS, TUNE CONFIGS
Confirm delivered value, check that outcomes meet requirements, leverage evidences to refine configurations.

6. GET TRAINED
Learn how to instantiate new flows, or update existing configurations to keep up with future use cases.

ABIS-X DELIVERY

UPCOMING

DATA ENRICHMENT
Enrich records with relevant info from external lookup data addressed by values of a field existing in raw data.

PRIORITY

SENSITIVE DATA MANAGEMENT
Anonymization or pseudonymization of sensitive data fields through an array of internal or custom-defined algorithms.

DATA FILTERING
Exclusion of irrelevant data, e.g. technical records or fields, for storage efficiency and better focus on relevant data.

CUSTOMIZED PROCESSING
Calculations, conversions, reformatting based on fields in raw data to enrich records with essential info.

MORE SOURCES & FORMATS
Ingestion from a monitored directory in HDFS; parsing of standard separated or XML files stored as text or Parquet.

CONFIGURATION GUI
Graphical User Interface to configure operation parameters with no need to edit properties files.

OVERVIEW

ABIS X

Since day-0, ABIS-X endeavored to address a wide and diverse scope of requirements: functional ones from Businesses, technical ones from ICT Depts and operational ones from AMS Teams, all conveyed into a single fast application in reliably operated at scale, that demands no coding and is open to evolve.

INGESTION CORNERSTONE

Proven architecture, proven modules, delivered outcomes. Multi-vendor e.g. Hortonworks and Cloudera Hadoop.

CONFIGURE FLOWS, DON'T CODE THEM

Truly zero code lines to setup & operate, only configuration. Single-command solution deployment via Linux installer. Anything that may influence operation can be configured.

DISTINCTIVE OPERATION CAPABILITIES

Distinctive features to recover operation from app shutdowns. Recognition and recording of invalid data and their issues.

FAST, RELIABLE, AT SCALE

Speed, reliability, scalability inherent to design and build. Kafka, Spark, Kudu, Hive & HBase are fast and fault-tolerant. Kafka, Spark Kudu, Hive & HBase scale with the cluster they live in.

HIGH FLEXIBILITY

Allow adaptation to changing or evolving requirements. Flexibility through asset template agnosticism and parametrizations.

CYBER SECURITY

Enabling Big Data Cybersecurity with Kerberos and Sentry, re-design of existing applications and 360° user profiling to ensure data security across the whole organization.

ENHANCE RELIABILITY

Leverage proven engineered solutions. Minimize costs and time of the test activities.

MAIN FEATURES

ABIS X

- 100% CONFIGURABLE
- RELEVANT SKILLS IN OUR PRACTICE
- STABLE, PROVEN, FAULT-TOLERANT
- PROVEN SW DEVELOPMENT PATTERNS
- NATIVLEY INTEGRATED IN HADOOP
- MULTI-VENDOR
- HIGHLY INTEROPERABLE
- SUPPORTED BY VENDORS
- SCHEMA AGNOSTICISM
- KERBEROS COMPLIANT

OVERVIEW

ABIS-KH, Kafka-to-HBase, consumes JSON data messages from one or more Apache Kafka topics, parses, enriches, processes them and stores results into one or more tables of the Apache HBase wide-column store.

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

STREAMING ITERATION

Periodic streaming iterations (based on data micro-batches) are executed.
Each streaming iteration includes:

- Parallel Json messages consumption from Apache Kafka topics.
- Check Jsons' validity and parse valid Jsons.
- Valid data processing through a configurable multi-stage transformation pipeline.

The available transformation base is expandable and pluggable and e.g. actually includes:

- Data enrichment
- ISO 8601 DateTime part extraction and conversion
- String padding and formatting
- Field renaming

PARSED DATA STORAGE INTO HBASE

- Valid parsed JSON data messages are stored into target data HBase tables, with multi-threaded writing.
- Processed data Journaling to provide bookmarks for downstream application.
- Invalid JSON data messages are stored into the discarded HBase table for troubleshooting and remediation.
- Recording on specific HBase Table of first and last offsets of valid or invalid messages processed during the streaming iteration, in order to support product restartability in case of failure or planned maintenance.

ABIS KH

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	Kafka (Version 0.10)
SOURCE DATAFORMAT	JSON
TARGET STORAGE	APACHE HBASE (Version 1.2.0)
ENGINE	APACHE SPARK STREAMING (Version 2.1.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	UPSERT
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 3 consumption start modes • 50+ configuration properties • 15+ transformation pipeline tools • 45 plus Preliminary Validations & Checks • Yarn Cluster Deployment • Exploit HBase Authentication Tokens in order to minimize KDC login number • Json flattening • Consumption start offset flexibility • Multi-execution prevention • ABIS-KH can also be used in not Kerberized Environment • Tested on over 3BLN messages/day

OVERVIEW

ABIS-KU, Kafka-to-Kudu, consumes and deserializes AVRO data messages from one or more Apache Kafka topics, parses, enriches, processes them and stores results into one or more Apache Kudu Tables.

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

STREAMING ITERATION

Periodic streaming iterations (based on data micro-batches) are executed.
Each streaming iteration includes:

- Parallel Json data messages consumption from Apache Kafka topics.
- Check Avro' validity and valid Avro deserialization that includes two levels:
- First deserialization to obtain the source table name of the message and the schema hash
- Second deserialization of the payload with the appropriate schema for each group of homogeneous records.
- Grouping all AVRO messages in different flows and their deserialization with specific schema on relative target tables.

PARSED DATA STORAGE INTO KUDU

- Each group of valid deserialized AVRO data message is stored into the relative target Kudu table.
- Failed Upsert or Delete Action Check.
- Invalid AVRO data messages are stored in two ways:
- For failed deserialization, the discarded AVRO Messages are stored into HBase table for troubleshooting and remediation.
- For Failed Upsert or Failed Delete, the discarded AVRO Messages are saved into a specific path on HDFS.
- Correct Validation Flow and then recording, on specific HBase Table, the last batch offset of valid or invalid messages processed during the streaming iteration.

ABIS KU

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	APACHE KAKFA (Version 0.10)
SOURCE DATAFORMAT	AVRO
TARGET STORAGE	APACHE KUDU (Version 1.6.0)
ENGINE	APACHE SPARK STREAMING (Version 2.1.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	UPSERT, DELETE
CONFIGURATION FILE TYPE	TEXT FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 60+ configuration properties • 3 consumption start modes • Yarn Cluster Deployment • Multi-execution prevention • Logs File on HDFS • Logs File on HBase table • ABIS-KU can also be used in not Kerberized Environment

OVERVIEW

ABIS-FT, File-to-Table, reads data from one or more CSV Source files, enriches and stores read data into Apache Hive Tables.

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

BATCH PROCESSING

Data batch processes are executed. Each Data batch includes:

- Reading data from CSV files on HDFS
- Check CSV validity
- The valid data processing includes:
  - Addition of any calculated columns during processing phase, only modifying the Json Configuration File.

LOADING DATA STORAGE INTO HIVE TABLES

- Valid CSV are stored into target data Hive tables and can be stored in HDFS path for troubleshooting. Each record can be updated in two different ways:
  - ranking update algorithm
  - update through an operation identification field
- Invalid CSV are stored into the specified error HDFS path for troubleshooting and remediation.

ABIS FT

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	APACHE HDFS (Version 2.6.0)
SOURCE DATAFORMAT	CSV
TARGET STORAGE	APACHE HIVE (Version 1.1.0)
ENGINE	APACHE SPARK (Version 2.1.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	OVERWRITE, APPEND, UPDATE
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 50+ configuration properties • Yarn Client Deployment • Logs File on HDFS • Process Summary Report File available on HDFS • It's possible to compress the archived files in Gzip format • ABIS-FT can also be used in not Kerberized Environment

OVERVIEW

ABIS-KT, Kafka-to-Tables, consumes and deserializes JSON data messages from one or more Apache Kafka topics, parses, enriches, processes them and stores results into one or more Apache Hive Tables.

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

STREAMING ITERATION

Periodic streaming iterations (based on data micro-batches) are executed.
Each streaming iteration includes:

- Parallel Jsons' data messages consumption from Apache Kafka topics
- Check Jsons' validity and deserialize valid Jsons
- Grouping all data messages in different flows and their deserialization with specific schema on relative target tables.

PARSED DATA STORAGE INTO HIVE TABLES

- Each group of valid deserialized JSON data message is stored into the relative target Hive table.
- Failed Loading Data Check.
- Invalid JSON data messages are stored in two ways:
  - For failed deserialization, the discarded JSON Messages are stored into HBase table for troubleshooting remediation.
  - For Failed Loading Data, the discarded JSON Messages are saved into a specific path on HDFS for troubleshooting and remediation.
- Correct Validation Flow then recording on specific HBase Table the last batch offset of valid or invalid messages processed during the streaming iteration.

ABIS KT

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	Kafka (Version 0.10)
SOURCE DATAFORMAT	JSON
TARGET STORAGE	APACHE HIVE (Version 1.1.0)
ENGINE	APACHE SPARK STREAMING (Version 2.1.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	APPEND
CONFIGURATION FILE TYPE	TEXT FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 60+ configuration properties • 3 consumption start modes • Yarn Cluster Deployment • Multi-execution prevention • ABIS-KT can also be used in not Kerberized Environment

OVERVIEW

ABIS-**DT**, **D**atabase-to-**T**able, reads data from one or more RDBMS tables and stores read data into Apache Hive Tables

ABIS DT

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

BATCH PROCESSING

Data batch processes are executed.
Each periodic Spark process includes:

- Data extraction from one or more RDBMS table, using JDBC
- It's possible to import only some source table columns configuring the "select" property in the configuration file
- It's possible to import only source table filtered data configuring the "where" property in the configuration file
- It's possible to parallelize the read process from RDBMS only configuring the configuration file

LOADING DATA INTO HIVE TABLES

Extracted data are loaded into one or more Hive Tables, based on different file format

ABIS DT

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	GENERIC RDBMS
SOURCE DATAFORMAT	TABLE
TARGET STORAGE	APACHE HIVE (Version 1.1.0)
ENGINE	APACHE SPARK (version 2.4.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	OVERWRITE, APPEND
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 15+ configuration properties • Yarn Client Deployment • ABIS-DT can also be used in not Kerberized Environment • Parametric Bash Shell Launcher

OVERVIEW

ABIS-**DU**, **D**atabase-to-**K**udu, , reads data from one or more RDBMS tables and stores read data into data into Apache Kudu Tables

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

BATCH PROCESSING

Data batch processes are executed.
Each periodic Spark process includes:

- Data extraction from one or more RDBMS table, using JDBC
- It's possible to import only some source table columns configuring the "select" property in the configuration file
- It's possible to import only source table filtered data configuring the "where" property in the configuration file
- It's possible to parallelize the read process from RDBMS only configuring the configuration file

LOADING DATA INTO HIVE TABLES

Extracted data are loaded into one or more Kudu Tables

ABIS DU

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	GENERIC RDBMS
SOURCE DATAFORMAT	TABLE
TARGET STORAGE	APACHE KUDU (Version 1.6.0)
ENGINE	APACHE SPARK (version 2.4.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	UPSERT
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 15+ configuration properties • Yarn Client Deployment • ABIS-DT can also be used in not Kerberized Environment • Parametric Bash Shell Launcher

OVERVIEW

ABIS
H3

ABIS-H3, HBase-to-AWS Bucket S3, fetches data from the source HBase table and stores them into files in a specific AWS S3 bucket directory, as nested per-rowkey JSONs.

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

LOADING DATA INTO AWS BUCKET S3

- Writing the extracted JSONs data messages into the AWS S3 file according to a predefined scalable and interoperable data model.

BATCH PROCESSING

Data batch processes are

executed. Each Data batch

includes:

- Reading the HBase Source table via HBase Scan:
  - The start time and the end time for the data extraction are passed by the application parametric launcher.
- The read data is used to JSONs Marshalling that wraps individual JSONs

ABIS H3

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	HBASE (Version 1.2.0)
SOURCE DATAFORMAT	TABLE
TARGET STORAGE	AWS S3 BUCKET DIRECTORY
ENGINE	APACHE SPARK (Version 2.1.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	APPEND
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 15+ configurations properties • Yarn Client/Cluster Deployment • Parametric Launcher • It's possible to clean the AWS S3 bucket target directory up before starting data extraction. • ABIS-H3 can also be used in not Kerberized Environment

OVERVIEW

ABIS 3H

ABIS-3H, AWS Bucket S3-to-HBase, uploads data from files in a 'directory' of the AWS S3 into the destination HBase table, based on read JSON.

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

BATCH ITERATION

Data batch processes are executed.

Each Data batch includes:

- Reading the contents of one or more S3 files in parallel.
- Validation of each line read from the file and parsing as a JSON.
- Valid data is unmarshalled based on a scalable and interoperable dataModel and converted into an object to be loads into HBase.

PARSED DATASTORAGE INTO HBASETABLES

- Valid data are stored into the target HBase table
- The invalid data are stored into the target HBase table: each Put is built for the rowkey built by current timestamp, application name, S3 file name, number of file line (coordinates of the entry), for troubleshooting remediation.

ABIS 3H

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	AWS S3 BUCKET DIRECTORY
SOURCE DATAFORMAT	JSON
TARGET STORAGE	HBASE (Version 1.2.0)
ENGINE	APACHE SPARK (Version 2.1.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	UPSERT
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 15+ configuration roperties • Yarn Client/Cluster Deployment • Parametric Launcher • It's possible to define Time-To-Live of HBase Tables modifying parametric launcher • It's possible to delete successfully loaded files from the source AWS S3 bucket directory Table • It's possible to truncate the target HBase table before loading data ABIS- 3H can also be used in not Kerberized Environment

OVERVIEW

ABIS **KK**, **K**afka-to-**K**afka, consumes data from Apache Kafka Topics, performs filtering, conversion and transformation and publishes on Apache Kafka Topic.

ABIS KK

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

STREAMING ITERATION

Periodic streaming iterations (based on data micro-batches) are executed.
Each streaming iteration includes:

- Parallel Jsons' data messages consumption from Apache Kafka topics
- Data conversion, filtering and transformation are performed

LOADING DATA INTO HIVE TABLES

Extracted data are published on Apache Kafka Topic

ABIS KK

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	KAFKA
SOURCE DATAFORMAT	JSON
TARGET STORAGE	KAFKA
ENGINE	APACHE SPARK STREAMING (Version 2.1.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	APPEND
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	15+ configuration properties Yarn Client Deployment ABIS-KK can also be used in not Kerberized Environment Parametric Bash Shell Launcher

OVERVIEW

ABIS T1

ABIS-T1 enables Data Quality and Profiling controls on data stored in Cloudera tables.
It provides tools to prepare data and define quality & profiling rules to be evaluated on data.
Rules outcome can be stored in different store to enable simple logging or reporting on controls.

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

BATCH PROCESSING: RULES EVALUATION

Quality and Profiling rules evaluation engine on data stored in Cloudera including:

- Data source configuration (e.g. Hive tables, Kudu tables,...);
- Data preparation to normalize dataset and facilitate rule evaluation;
- Rule evaluation on data storing all exceptions and calculating metrics on good vs bad records;
- Outcome configuration in order to define log or table structure to analyze and visualize exceptions and trends

PARSED DATASTORAGE INTO HDFS

The result of Spark action will be stored into Hive tables or simple log files.

ABIS T1

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	HDFS / KUDU
SOURCE DATAFORMAT	TABLE
TARGET STORAGE	HDFS / APACHE HIVE
ENGINE	APACHE SPARK (Version 2.4.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	INSERT
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 30+ configuration properties • 45 plus Preliminary Validations & Checks • Yarn Cluster Deployment • Multi-execution prevention • Configurable Data Preparation based on SQL transformations • Configurable validation check output structure & format

OVERVIEW

ABIS
T2

ABIS-T2, the «compactor», consumes files from HDFS to resolve the large number of small files within HDFS locations . It uses defined thresholds to understand which file should be compacted and it stores results into defined Hive tables.

PRELIMINARY VALIDATIONS AND CHECK

Performs validations and checks to confirm that properties are correctly configured, and elements required to successfully execute are in place.

BATCH PROCESSING: COMPACTION OF DATA

The action of compacting small files is executed through the declaration of defined thresholds including:

- size threshold to take only the smallest files;
- number of files that has to be compacted;
- threshold for a min and max files that has to be selected for compacting;

PARSED DATASTORAGE INTO HDFS

- The result of Spark action will be stored into Hive tables based on different data file format(ORC, Parquet, TextFile);
- In order to guarantee the consistency of data, ABIS-T2 provides the ability to make a backup before the files are merged.

ABIS T2

SUMMARY TABLE

MAIN FEATURES	DESCRIPTION
SOURCE DATASYSTEM	HDFS
SOURCE DATAFORMAT	TABLE
TARGET STORAGE	APACHE HIVE
ENGINE	APACHE SPARK (Version 2.4.0)
PROGRAMMING LANGUAGE	SCALA (Version 2.11.8)
WRITING MODE	OVERWRITE
CONFIGURATION FILE TYPE	JSON FORMAT ON HDFS (Version 2.1.0)
CAPABILITIES	• 50+ configuration properties • 45 plus Preliminary Validations & Checks • Yarn Cluster Deployment • Multi-execution prevention • ABIS-T2 can also be used in not Kerberized Environment • Configurable for multilevel partition • Ability to perfom a set up of backup of processed file

COMING SOON...

ABIS HT

OVERVIEW

ABIS-HT, HBase-to-Hive Tables, reads data from a table of Apache HBase wide-column and stores read data into a corresponding Apache Hive Table.

ABIS SH

OVERVIEW

ABIS-SH, Web Service-to-HBase, Requests Data from Web Service through REST API and stores data into HBase Target Table.

ABIS SK

OVERVIEW

ABIS-SK, Web Service-to-Kafka, Requests Data from Web Service through REST API and stores data into Apache Kafka Target Topic.

CONTACTS

Federico Laschi
Associate Director
Accenture S&C
Applied Intelligence ICEG
federico.laschi@accenture.com

ABOUT ACCENTURE

Accenture is a leading global professional services company, providing a broad range of services and solutions in strategy, consulting, digital, technology and operations. Combining unmatched experience and specialized skills across more than 40 industries and all business functions – underpinned by the world's largest delivery network – Accenture works at the intersection of business
and technology to help clients improve their performance and create sustainable value for their stakeholders. With 469,000 people serving clients in more than 120 countries, Accenture drives innovation to improve the way the world works and lives. Visit us at **www.accenture.com**.