DATA ARCHITECTURE SUMMIT

November 13 - 16, 2017 | Chicago, Illinois

ETL is No Longer King, Long Live SDD

How to Close the Loop from Discovery to Information (Data) to Insights (Analytics) to Outcomes (Business Processes)

A presentation by Brian McCalley of DXC Technology, Glenn Field of SiriusIQ and Gavin Robertson of WhamTech, Inc.

Produced By:
DATAVERSITY®

#DASummit

No secret that most organizations face major data-related hurdles

Location	Dirty
System	Typo/Transposition
Access	Missing
Security	Meaning
Container	Duplication
Format	Obfuscation
Age	Governance

…and ANALYTICS is the prime driver to lower costs and increase revenue

…which, in turn, drives the need for applications* to have clean and understood data in specific formats

*Reporting, BI, analytics, CDI-MDM, CRM, SCM, fraud detection, anti-money laundering, ERP, etc.

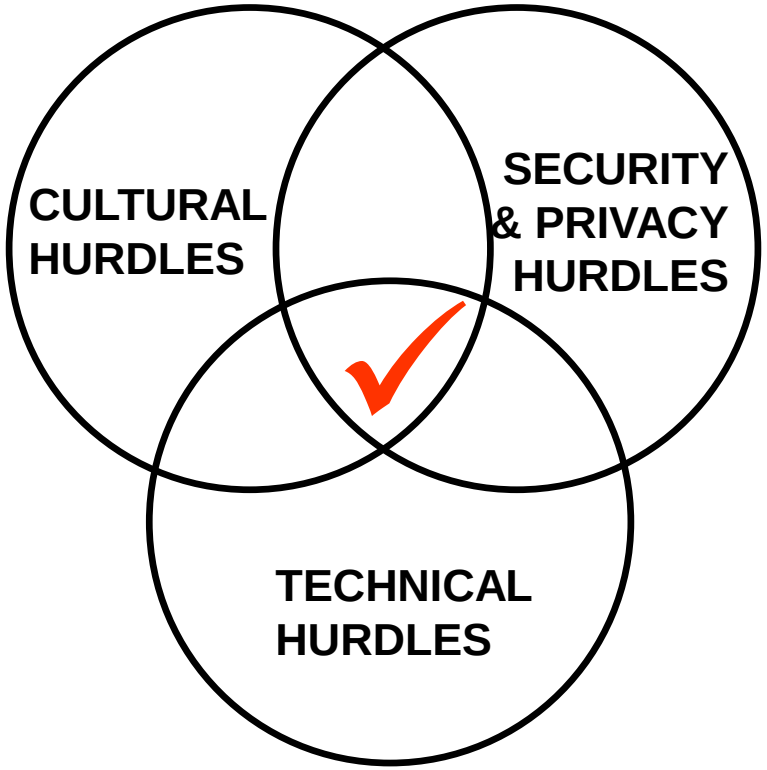
Goals for an optimal data architecture

1. Complete, clean, transformed, standardized and secure data, and master data, for multiple applications
2. Near real-time – minimal update and query latency
3. Automation, including workflow and event processing
4. Support reporting, BI and analytics, including graph database
5. Minimize copies of data
6. Data discovery, metadata repository and data governance
7. Write back to data sources

Managing data in, or from, multiple disparate systems requires a new approach

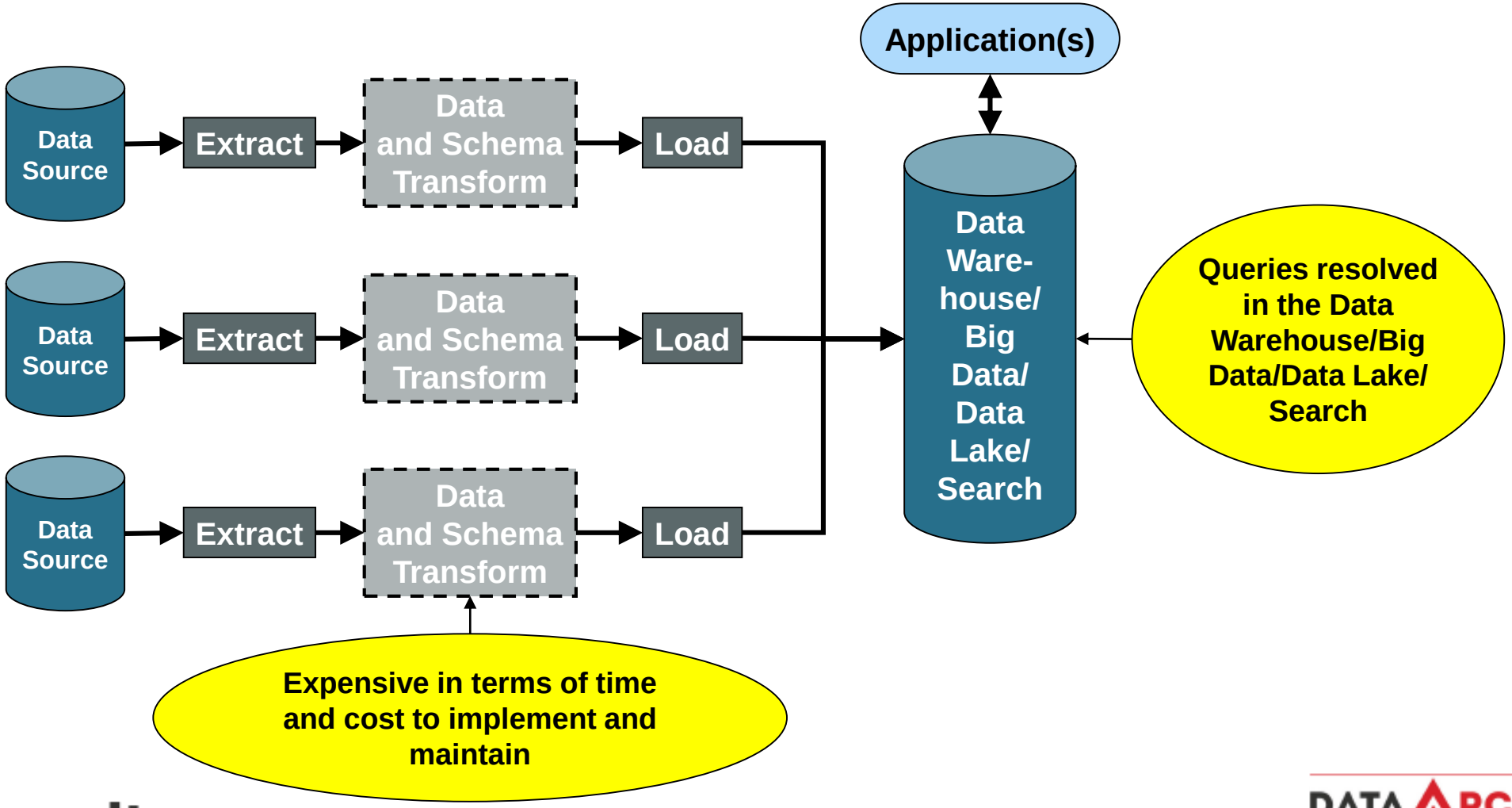
CONVENTIONAL APPROACHES

- **ETL**: Copy and transform schemas and data to a one-size-fits-all data warehouse
- **Copy**: To a single Data Lake/Big Data repository
- **Federate**: Submit queries through adapters to source systems
- **Search**: E.g., Solr™/Elasticsearch™ - copy, read, parse and index data, process queries to provide data – Big Data options for storing data

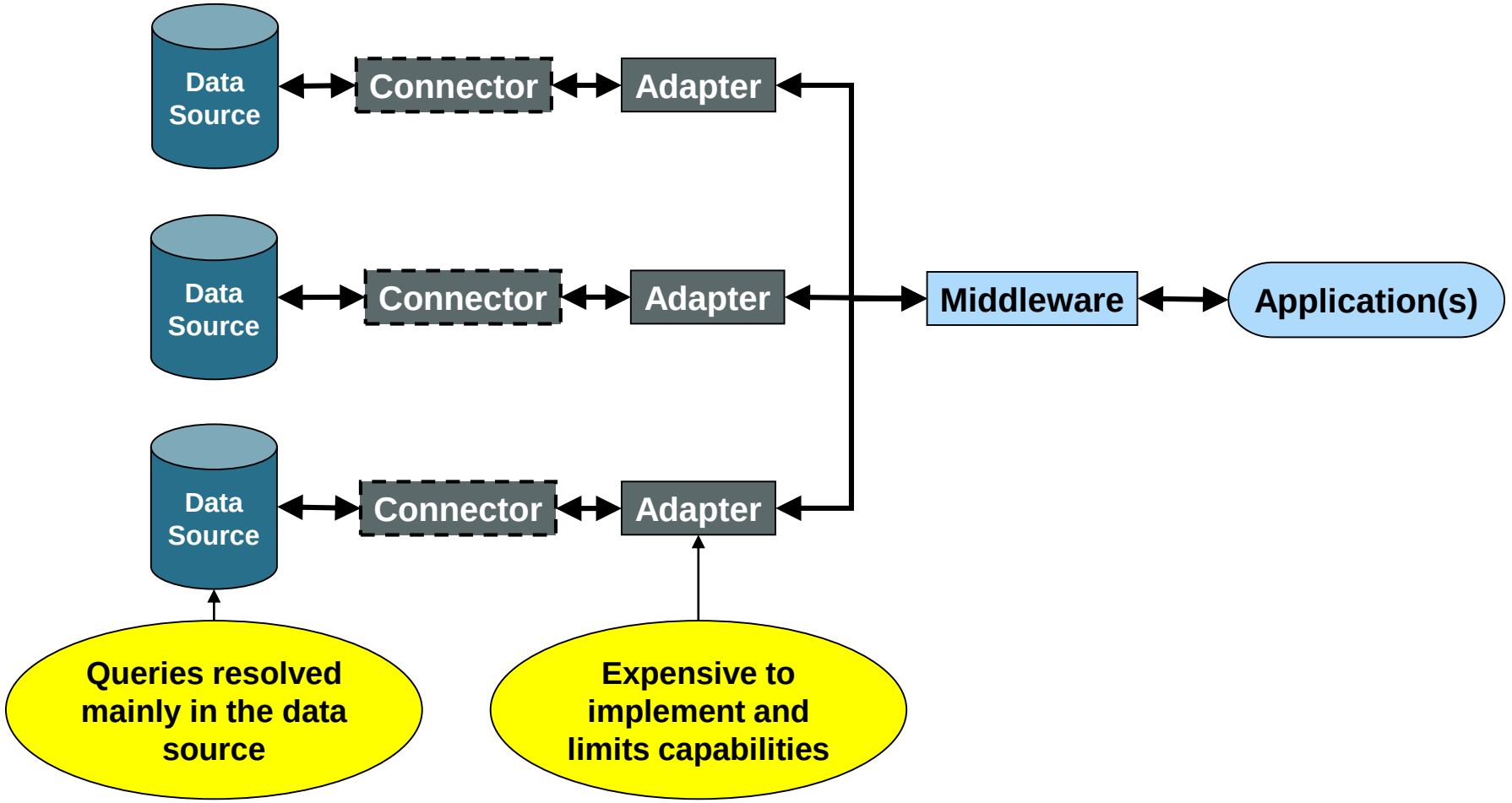


The new approach leverages the advantages of each conventional approach

Typical Data Warehouse/Big Data/Data Lake/Search



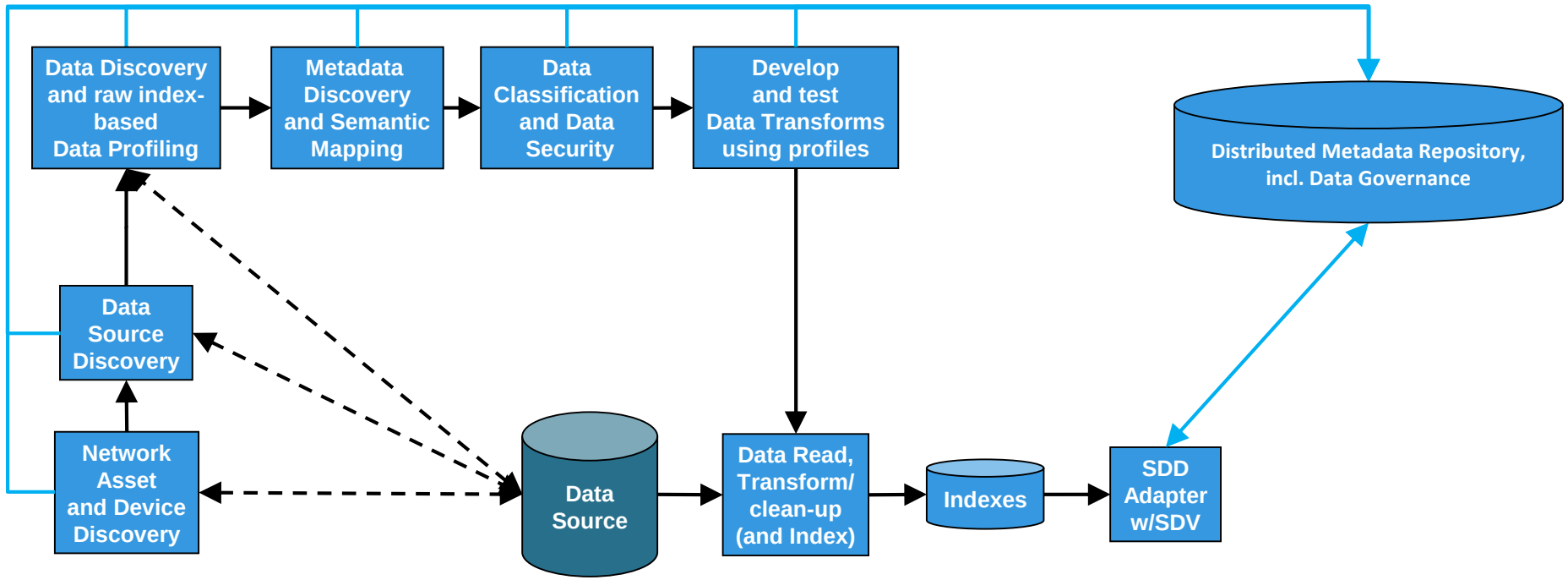
Typical federated data access with conventional adapters



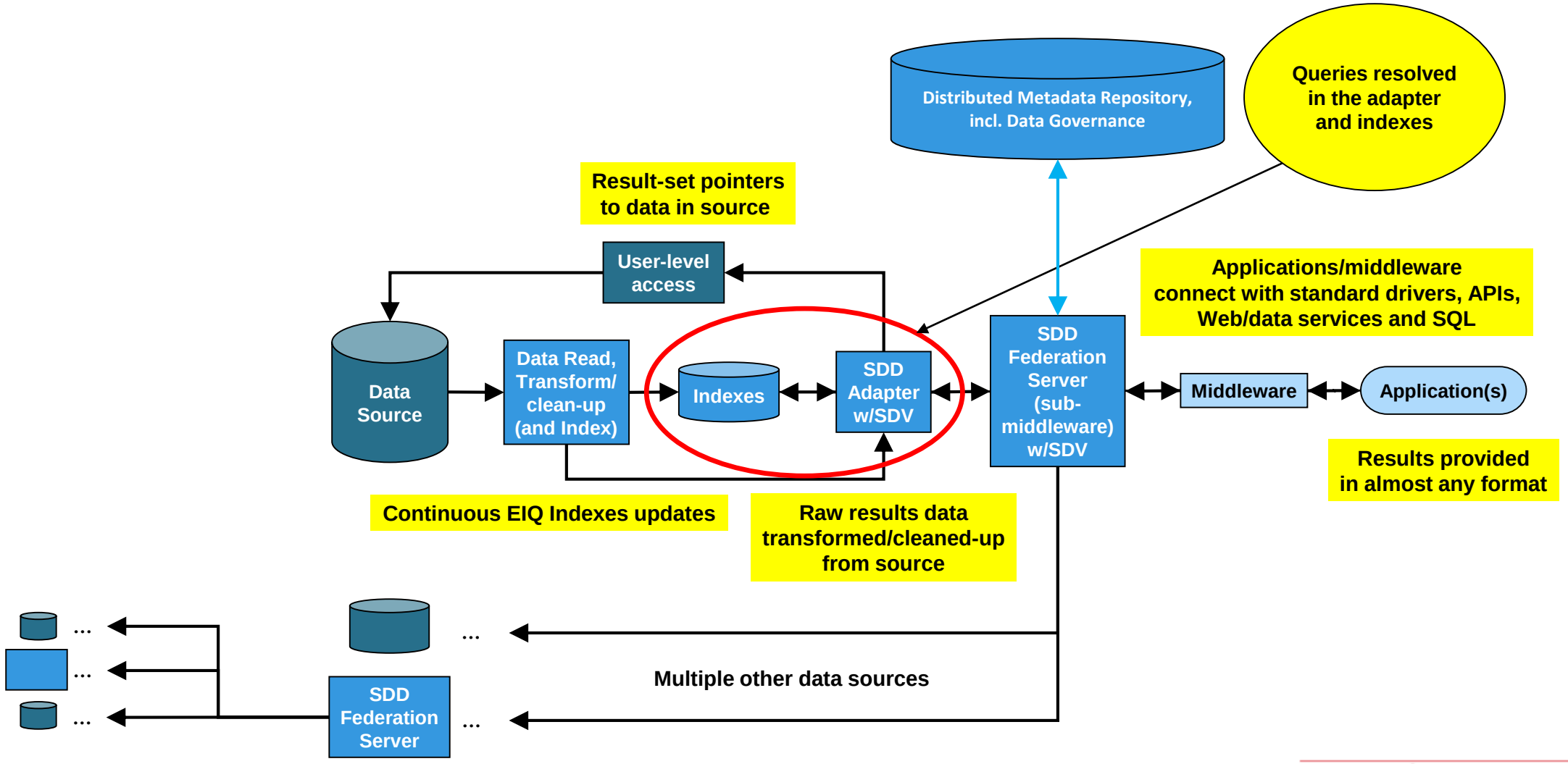
ETL has been the only option to come close to meeting the goals for an optimal data architecture, until now…

Introducing Software Defined Data (SDD) consisting of unconventional federated adapters that Read, Transform (process) and Index (RTI) source data and process queries against these indexes

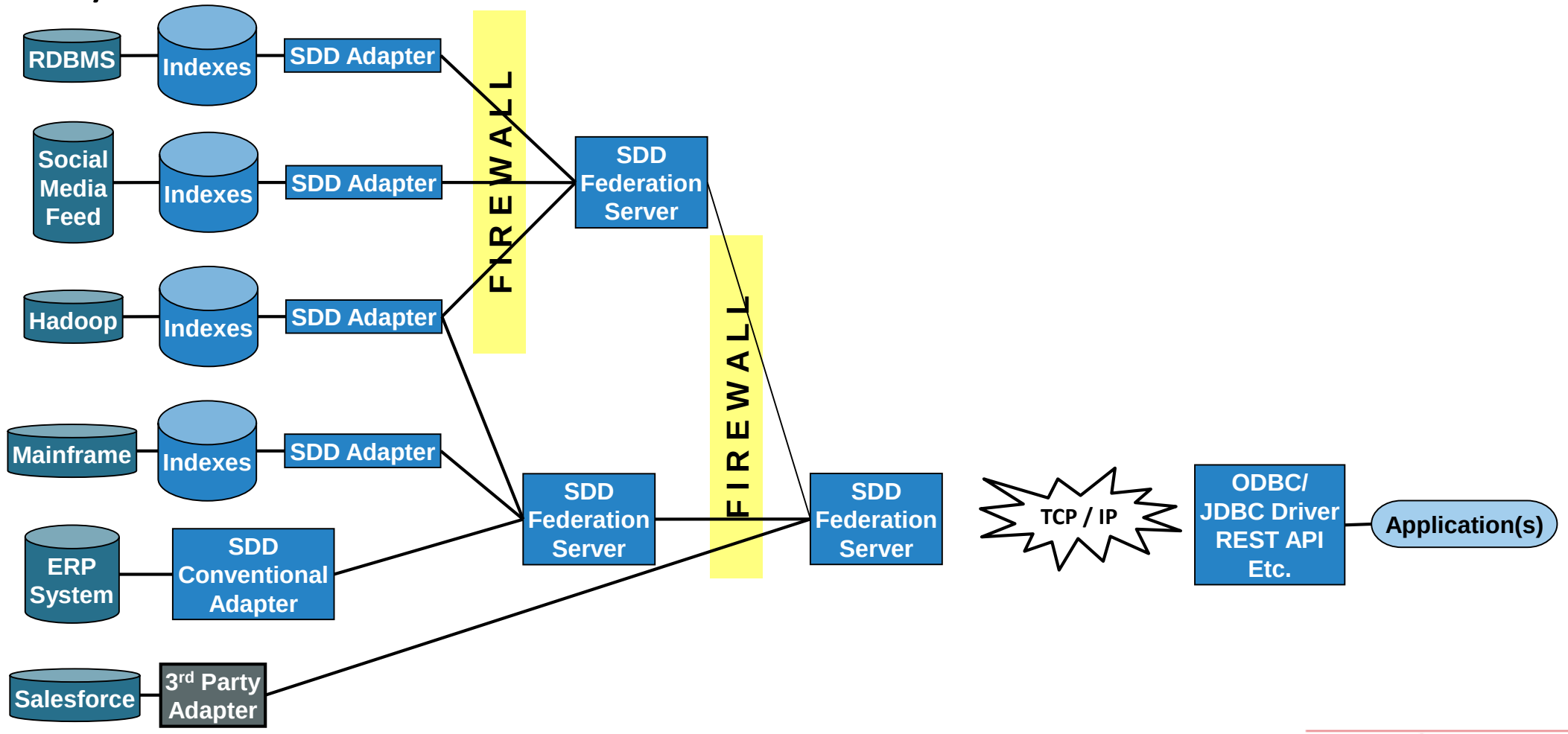
SDD initial discovery, index and adapter configuration, index build and Standard Data View (SDV) mapping



SDD index update, query processing and results retrieval

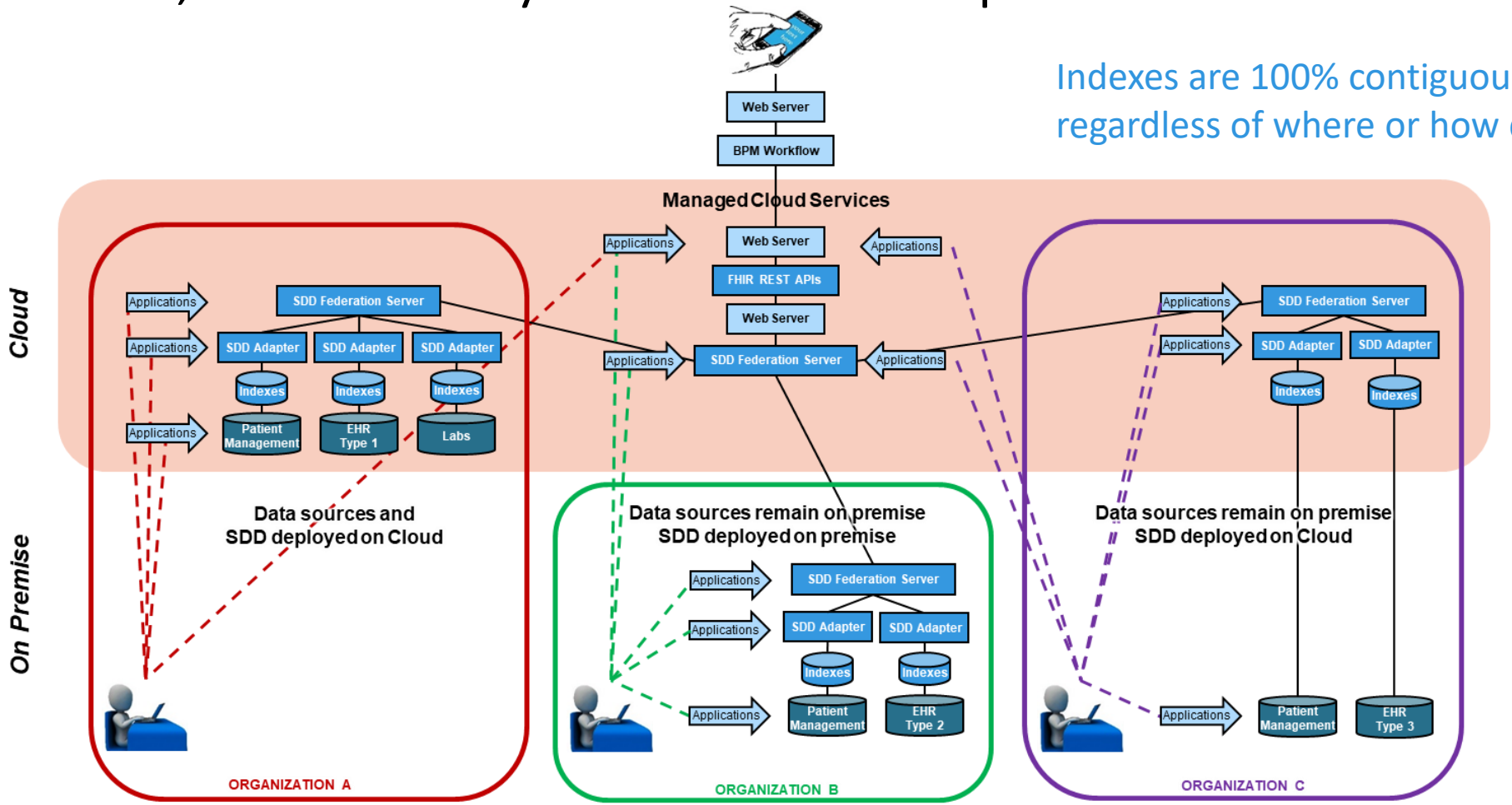


SDD adapters can co-exist with other types of adapters to System of Records



SDD indexes and adapters can be deployed and accessed anywhere, and at any level in multiple combinations

Indexes are 100% contiguous, regardless of where or how deployed



How does Software Defined Data compare with other approaches?

Goal #1: Complete, clean, transformed, standardized and secure data, and master data, for multiple applications

Goal		SDD	ETL to a Data Warehouse	Big Data Lake	Conventional Federated Adapters	Solr/ Elastic-search
#1	Complete, clean, transformed, standardized and secure data, and master data, for multiple applications	✓	✓	✗	(✗)	✗

Goal #2: Near real-time – minimize update and query latency

Goal		SDD	ETL to a Data Warehouse	Big Data Lake	Conventional Federated Adapters	Solr/ Elastic-search
#1	Complete, clean, transformed, standardized and secure data, and master data, for multiple applications	✓	✓	✗	(✗)	✗
#2	Near real-time – minimize update and query latency	✓	(✗)	(✓)	(✓)	(✓)

Goal #3: Automation, including workflow and event processing

Goal		SDD	ETL to a Data Warehouse	Big Data Lake	Conventional Federated Adapters	Solr/ Elastic-search
#1	Complete, clean, transformed, standardized and secure data, and master data, for multiple applications	✓	✓	✗	(✗)	✗
#2	Near real-time – minimize update and query latency	✓	(✗)	(✓)	(✓)	(✓)
#3	Automation, including workflow and event processing	✓	✗	✗	(✓)	✗

Goal #4: Support reporting, BI and analytics, including graph database

Goal		SDD	ETL to a Data Warehouse	Big Data Lake	Conventional Federated Adapters	Solr/ Elastic-search
#1	Complete, clean, transformed, standardized and secure data, and master data, for multiple applications	✓	✓	✗	(✗)	✗
#2	Near real-time – minimize update and query latency	✓	(✗)	(✓)	(✓)	(✓)
#3	Automation, including workflow and event processing	✓	✗	✗	(✓)	✗
#4	Support reporting, BI and analytics, including graph database	✓	(✓)	(✓)	(✓)	(✓)

Goal #5: Minimize copies of data

Goal		SDD	ETL to a Data Warehouse	Big Data Lake	Conventional Federated Adapters	Solr/ Elastic-search
#1	Complete, clean, transformed, standardized and secure data, and master data, for multiple applications	✓	✓	✗	(✗)	✗
#2	Near real-time – minimize update and query latency	✓	(✗)	(✓)	(✓)	(✓)
#3	Automation, including workflow and event processing	✓	✗	✗	(✓)	✗
#4	Support reporting, BI and analytics, including graph database	✓	(✓)	(✓)	(✓)	(✓)
#5	Minimize copies of data	✓	✗	✗	✓	✗

Goal #6: Data discovery, metadata repository and data governance

Goal		SDD	ETL to a Data Warehouse	Big Data Lake	Conventional Federated Adapters	Solr/ Elastic-search
#1	Complete, clean, transformed, standardized and secure data, and master data, for multiple applications	✓	✓	✗	(✗)	✗
#2	Near real-time – minimize update and query latency	✓	(✗)	(✓)	(✓)	(✓)
#3	Automation, including workflow and event processing	✓	✗	✗	(✓)	✗
#4	Support reporting, BI and analytics, including graph database	✓	(✓)	(✓)	(✓)	(✓)
#5	Minimize copies of data	✓	✗	✗	✓	✗
#6	Data discovery, metadata repository and data governance	✓	✓	(✗)	(✓)	(✗)

Goal #7: Write back to data sources

Goal		SDD	ETL to a Data Warehouse	Big Data Lake	Conventional Federated Adapters	Solr/ Elastic-search
#1	Complete, clean, transformed, standardized and secure data, and master data, for multiple applications	✓	✓	✗	(✗)	✗
#2	Near real-time – minimize update and query latency	✓	(✗)	(✓)	(✓)	(✓)
#3	Automation, including workflow and event processing	✓	✗	✗	(✓)	✗
#4	Support reporting, BI and analytics, including graph database	✓	(✓)	(✓)	(✓)	(✓)
#5	Minimize copies of data	✓	✗	✗	✓	✗
#6	Data discovery, metadata repository and data governance	✓	✓	(✗)	(✓)	(✗)
#7	Write back to data sources	✓	✗	✗	✓	✗

How SDD meets goals for an optimal data architecture

Goal		Software Defined Data
#1	Complete, clean, transformed, standardized and secure data, and master data, for multiple applications	• Process source data as building and maintaining indexes and master data, and as reading raw results data • Multiple indexes, views, means of access and result formats
#2	Near real-time – minimize update and query latency	• Changed data capture • High performance, parallel distributed processing – almost no load on data sources
#3	Automation, including workflow and event processing	• Index monitoring, REST APIs and workflow integration
#4	Support reporting, BI and analytics, including graph database	• Indexed views, provision highly curated data to analytics, run analytics, and built-in virtual graph database and link analysis
#5	Minimize copies of data	• Can leave and secure data in sources, a Data Lake or indexes
#6	Data discovery, metadata repository and data governance	• Use raw indexes for discovery, metadata and combining with IAM and RBAC for data governance – from edge/bottom up
#7	Write back to data sources	• Can read as well as insert, delete and update

Software Defined Data (SDD)

- Implementation and alignment of use-cases is the key to driving Enterprise IP. Technology prohibits this due to binding of data elements within applications
- Freeing data to create workflows will dramatically reduce time to market
- Incrementally developing enterprise use-cases through SDD drives innovation to next-gen path allowing a reinvention of the enterprise
- Agnostic de-coupling of silo solutions drives speed to market
- Use-case consumption for any user, on any device, anywhere securely enhances collaboration and productivity

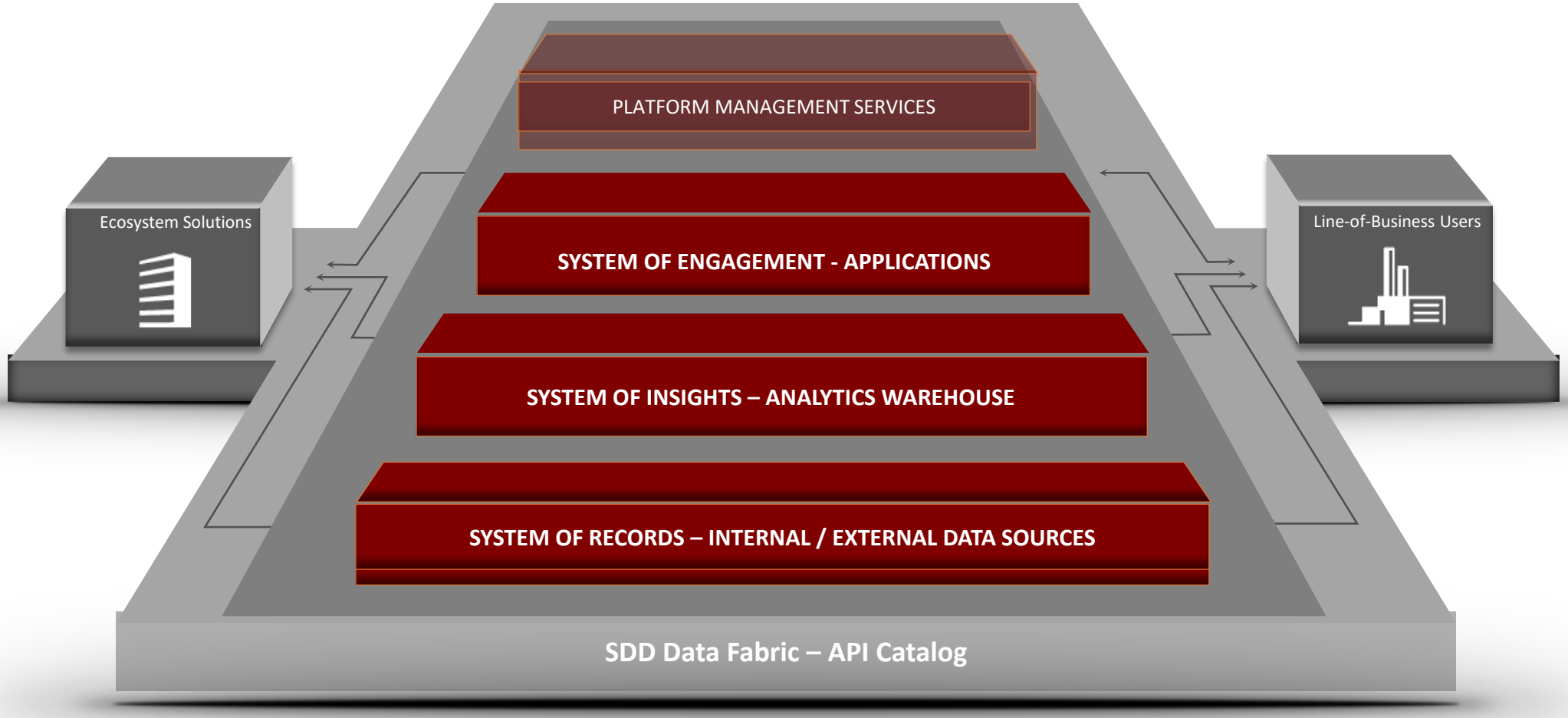
Software Defined Data (SDD)

- Process to eliminate upwards of 95% of today's regression issues – translates into 50-75% calendar time savings
- Business logic and code base functions grow incrementally as business dictates
- Cumulative code growth ensures reuse, optimal performance and agnostic access. You also benefit from globally available logic
- Zero impact deployments eliminates downtime and simplifies the SDLC
- Dynamic, intelligent workflows consume new features when live

AI and NLP are the new UI for many applications

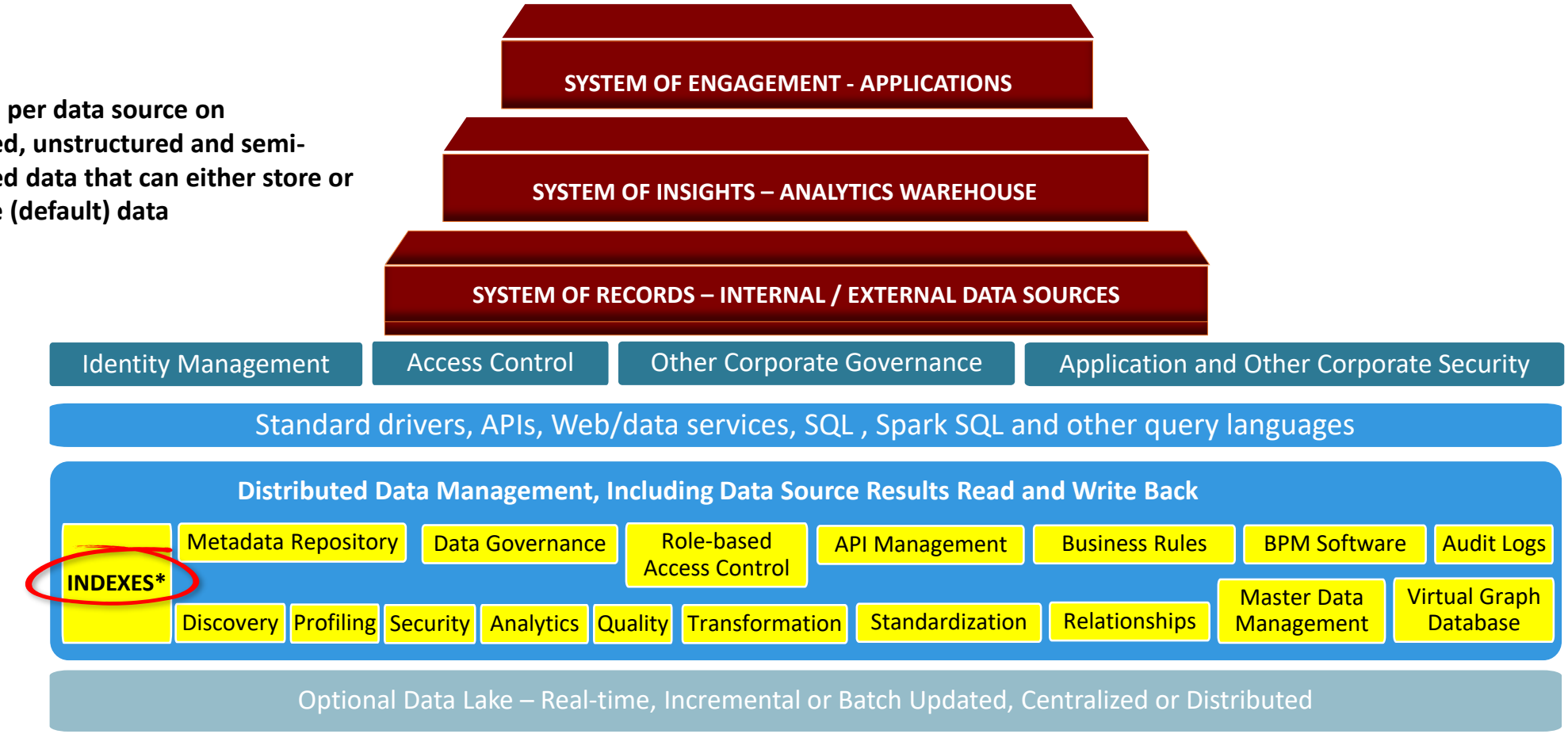
- Leveraging next-gen cognitive rapidly delivers functionality and results
- Millennial workforce alignment
- Dramatic decrease in training requirements
- Dramatic decrease in time to market on features and results
- Disconnect 3rd party backend solutions from natural language UI
- Allow seamless app upgrades by using AI UI, which will interact with both old and new systems

Abstracted 3-tier architecture connected through a Smart Data Fabric

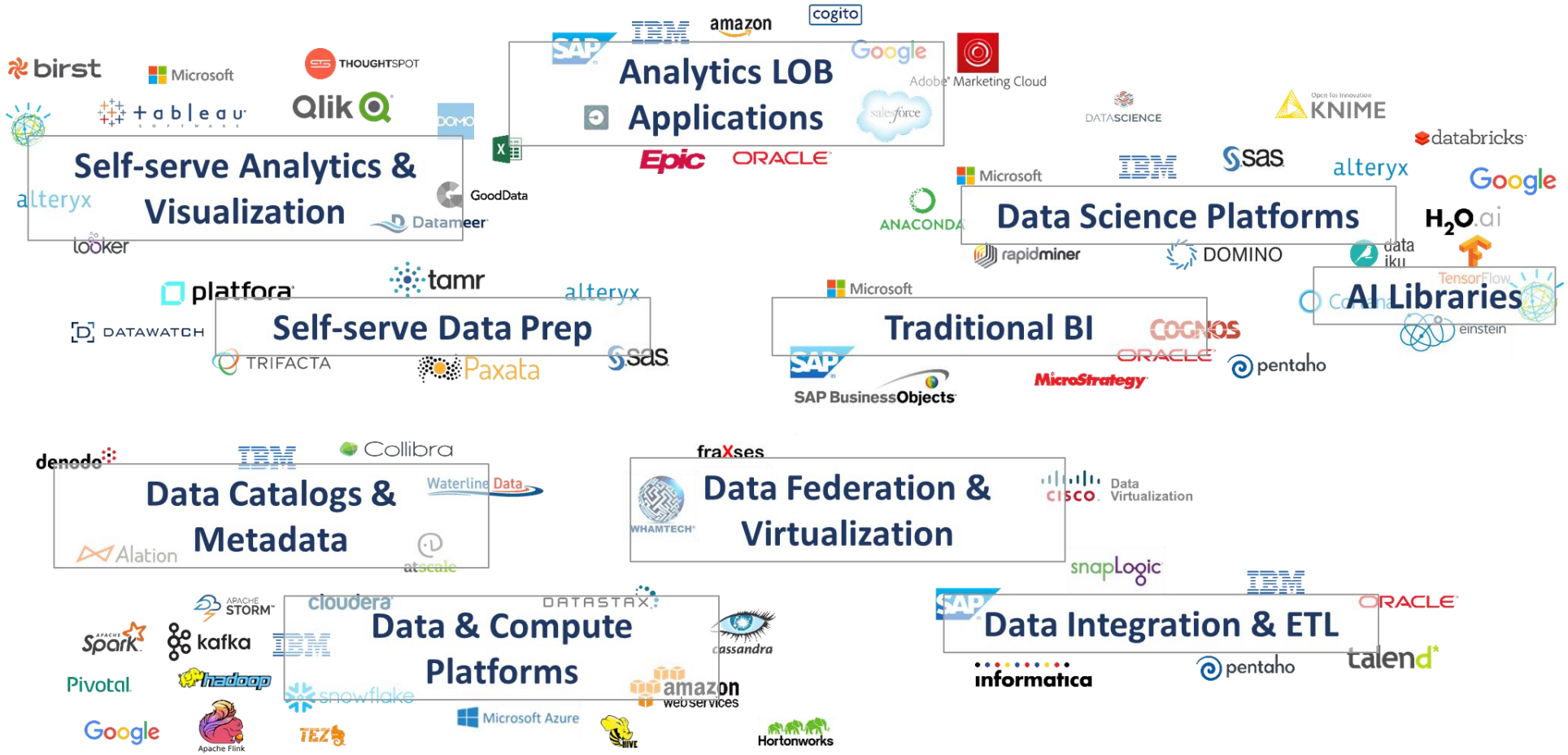


SDD is the First Paradigm Shift in how data and analytics are managed in a common meta-object framework

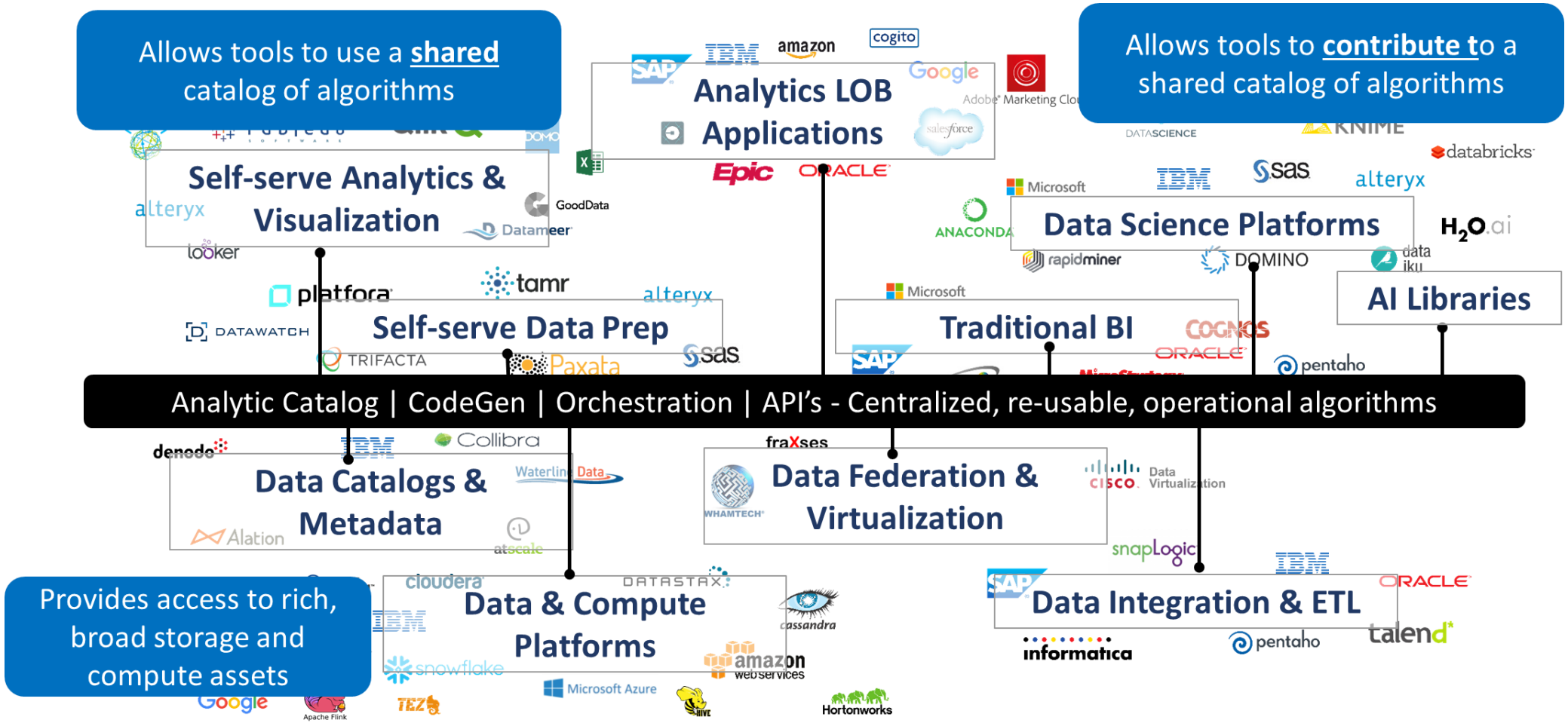
***Indexes per data source on structured, unstructured and semi-structured data that can either store or not store (default) data**



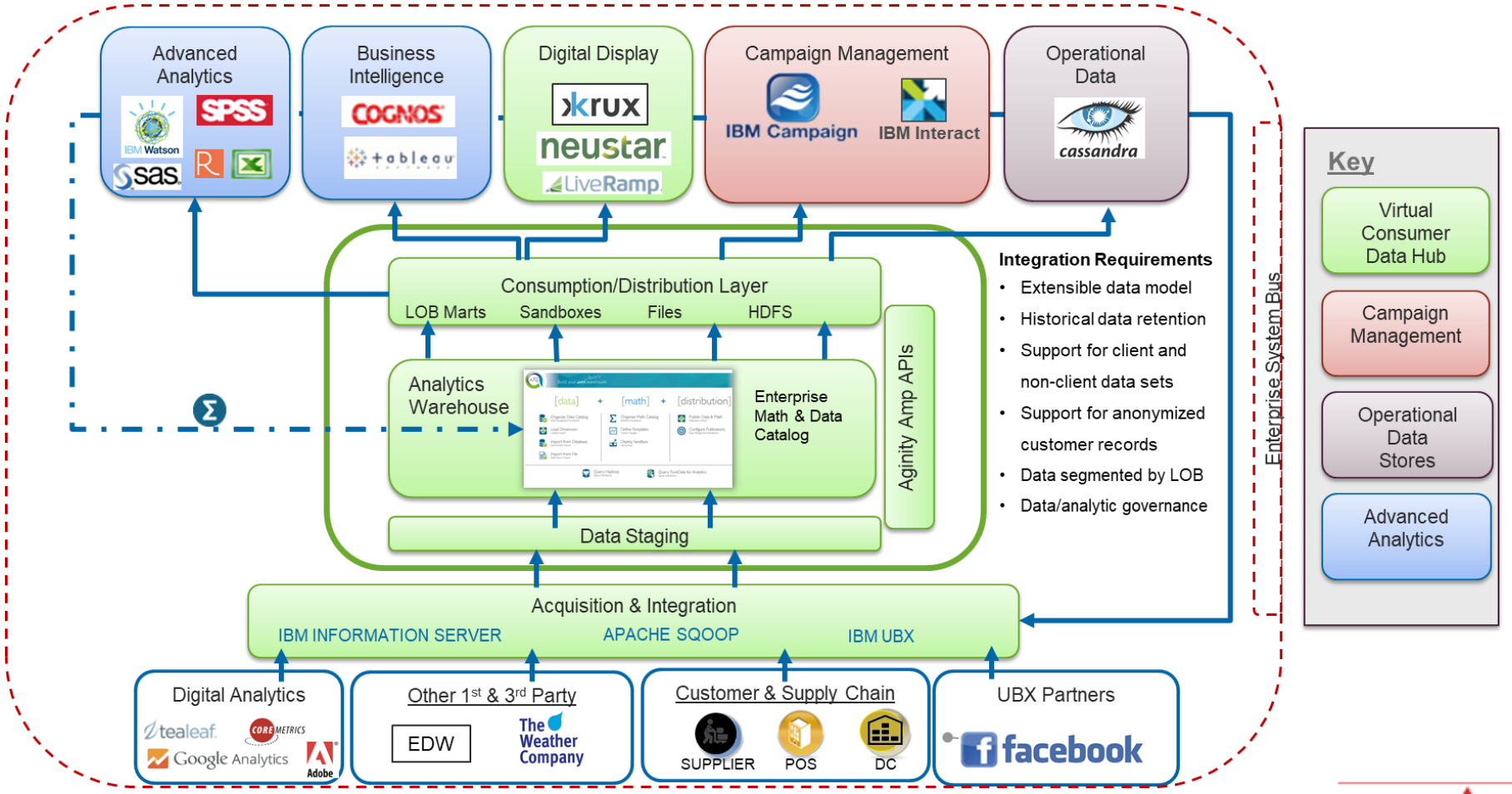
The Second Paradigm Shift is the concept of the Analytics Warehouse



The Second Paradigm Shift is the concept of the Analytics Warehouse

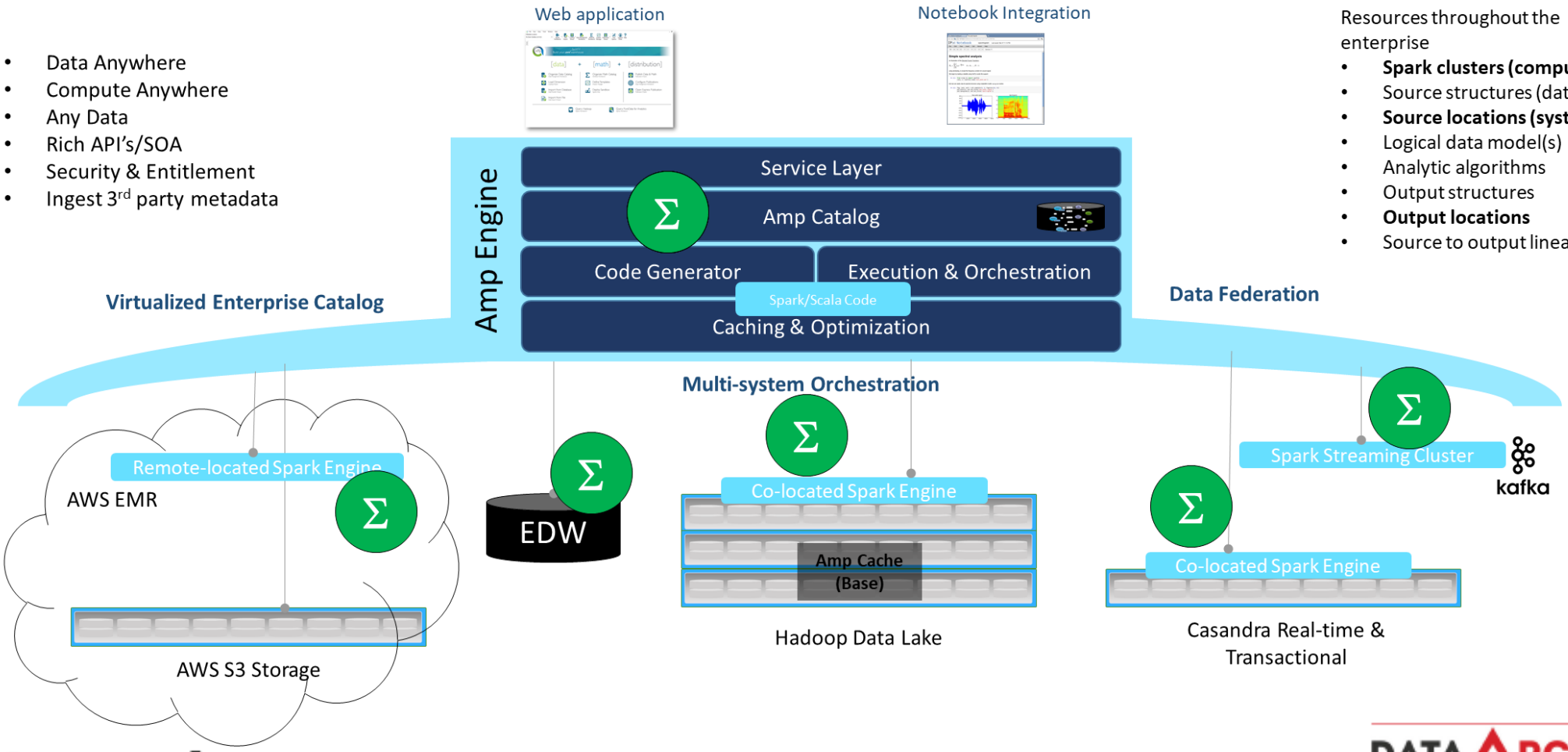


An example of an Analytics Warehouse architecture – data ingestion/warehouse model



An example of an Analytics Warehouse architecture spanning enterprise systems – federated model

- Data Anywhere
- Compute Anywhere
- Any Data
- Rich API's/SOA
- Security & Entitlement
- Ingest 3rd party metadata

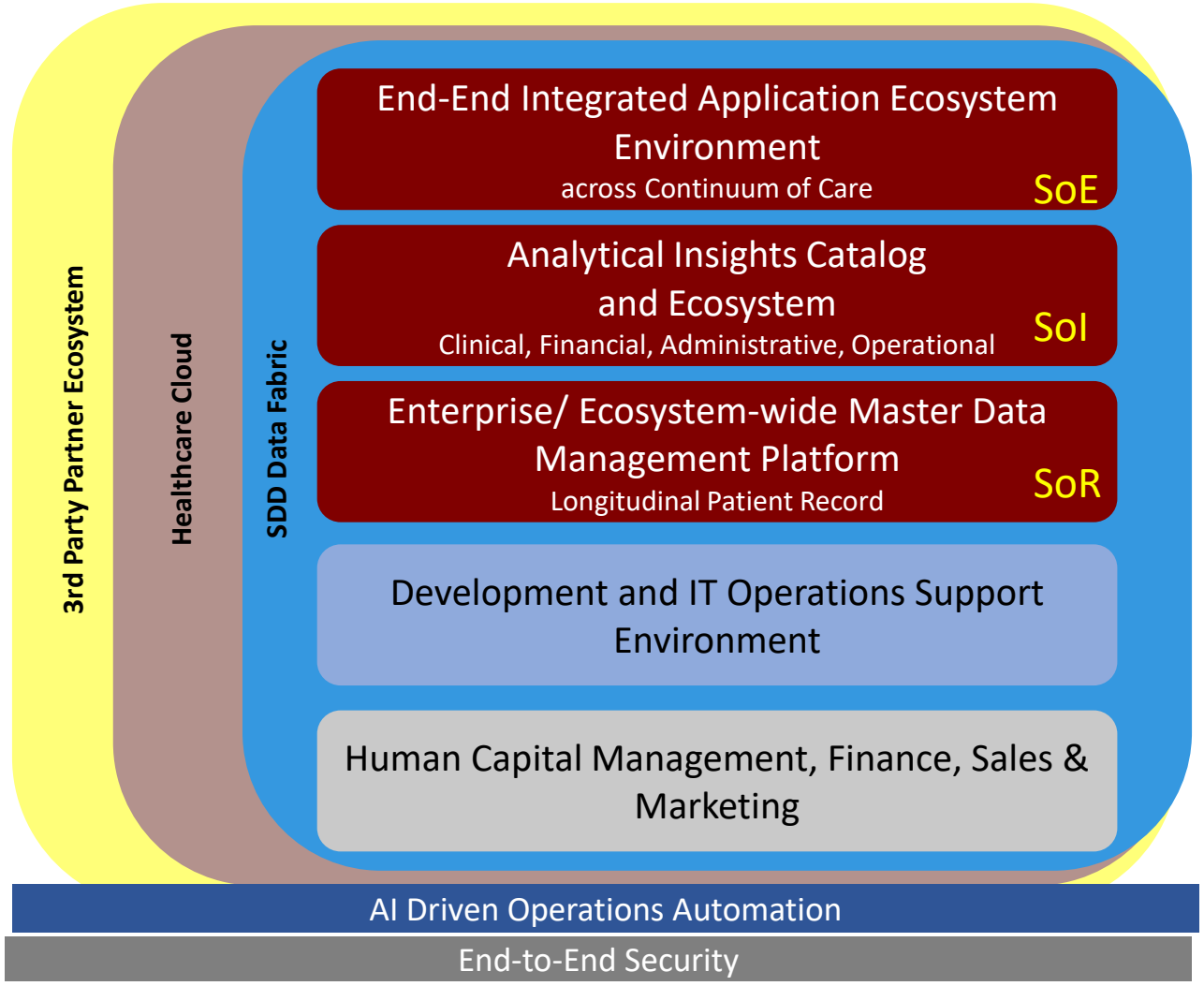


Resources throughout the enterprise

- **Spark clusters (compute)**
- Source structures (data)
- **Source locations (systems)**
- Logical data model(s)
- Analytic algorithms
- Output structures
- **Output locations**
- Source to output lineage

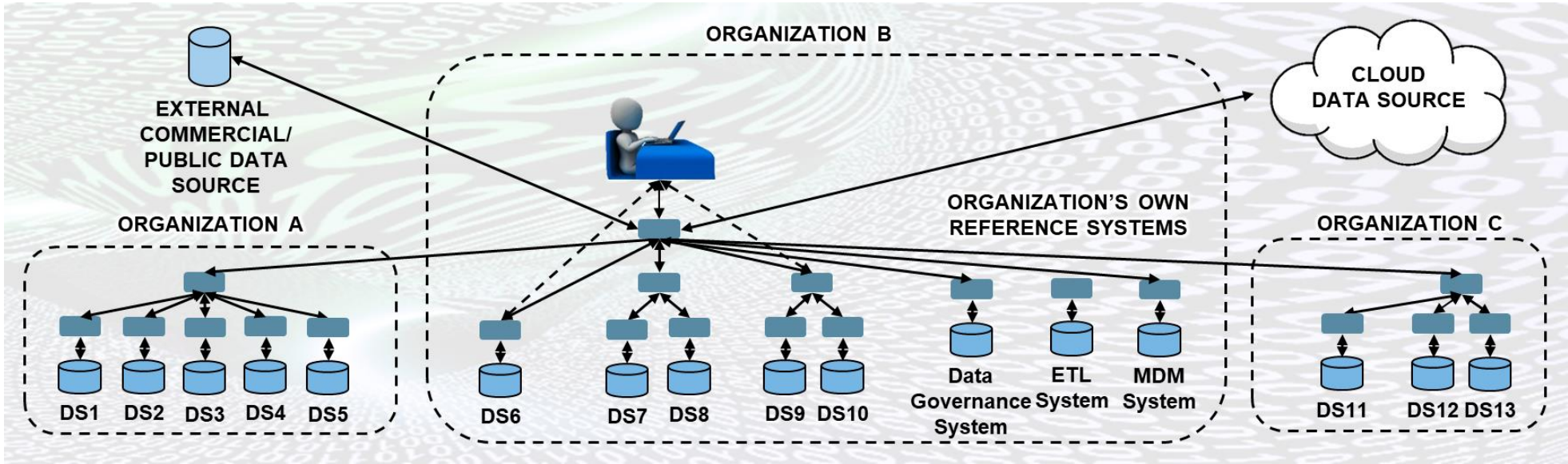
Examples of applying SDD and an Analytics Warehouse to healthcare analytics

Healthcare Clinical Network Management 3-Tier Architecture



SDD Data Fabric enables a Longitudinal Patient Record (LPR) view across multiple System of Records, across multiple enterprises

- Transparent distributed data management layer that plugs-and-plays in existing IT infrastructures
- Complements and leverages existing IT systems, tools and applications
- Leave and guard data in sources, copies, e.g., Data Lake, or stored in indexes – a hybrid approach
- Address upfront data discovery, security, quality, standards, MDM and other data-related processes



Use cases from healthcare that combine data and analytics management

Use Cases		Applications
Clinical Applications		✓ Diabetes, Hypertension, Heart Failure, etc. ✓ Gaps in Care ✓ Predictive Readmissions Management ✓ Clinical Wellness Management
Operational Management		✓ Operational Management – Hospital ✓ Operational Management – Physician Practices ✓ Physician Quality Reporting Scores
Financial Performance		✓ Financial Management – Hospital ✓ Financial Management – Physician Practices ✓ Claims Analytics
Regulatory Reporting		✓ Hospital Value Based Purchasing (HVBP) ✓ HEDIS ✓ Patient Centered Medical Home Scorecard ✓ MU 2 Clinical Quality Measures-Hospitals/Physicians ✓ MU2 Usage Scorecard - Physicians ✓ ACO Quality Reporting ✓ Hospital Outpatient Quality Reporting
Ability to create patient cohorts		✓ Cohort Manager/Chronic Condition Management
Population Management		✓ Population Focus & Population Care

- Patient Similarity
- Comparative Effectiveness Research
- Predictive models – Chronic disease management

Conclusion of why ETL is no longer King, long live SDD

- SDD enables a data management paradigm shift
- SDD supports an analytics management paradigm shift
- ETL still has value, but not exclusively
- SDD can greatly enhance, complement and/or replace ETL in the future
- SDD is more suited than ETL to the new world of:
  - Data everywhere
  - API data services/catalog
  - Parallel distributed processing
  - Event and workflow processing
  - Near real-time architectures

Thank you

A presentation by Brian McCalley of DXC Technology, Glenn Field of SiriusIQ and Gavin Robertson of WhamTech, Inc.

Q&A