Global Practices
Big Data & Analytics Practice

Data Fabric primer

December 2022
Rahul Barik

GlobalLogic®
A Hitachi Group Company

Intended Audience

The intended audience of this document is Architects that are looking to understand practical approaches to notable Big Data & Analytics Architecture-level aspects. It is NOT the intent of this document to cover Implementation level aspects to the lowest granularity, however, it IS intended that such detailed levels of associated content could be subsequently developed.

The Problem

A few months ago, this author was involved in a conversation with the Data & Analytics leaders of one of the leading online travel shopping companies in the world. The organization overwhelmingly leverages data and analytics to power its consumer and business travel products.

The company has many databases for separate applications powering its booking, shopping and loyalty programs. There are different reporting and BI applications as well. The organization also has an Enterprise Data Platform for Analytics and a centralized Data Governance team. But over a period of time, the organization has realized that Data Silos have developed in spite of their best efforts. On top of the existing situation, the online travel company has acquired many other online travel portals. Additional data repositories and platforms from these acquisitions continue to exist in their enterprise ecosystem which haven't yet been consolidated or merged with their own data platforms.

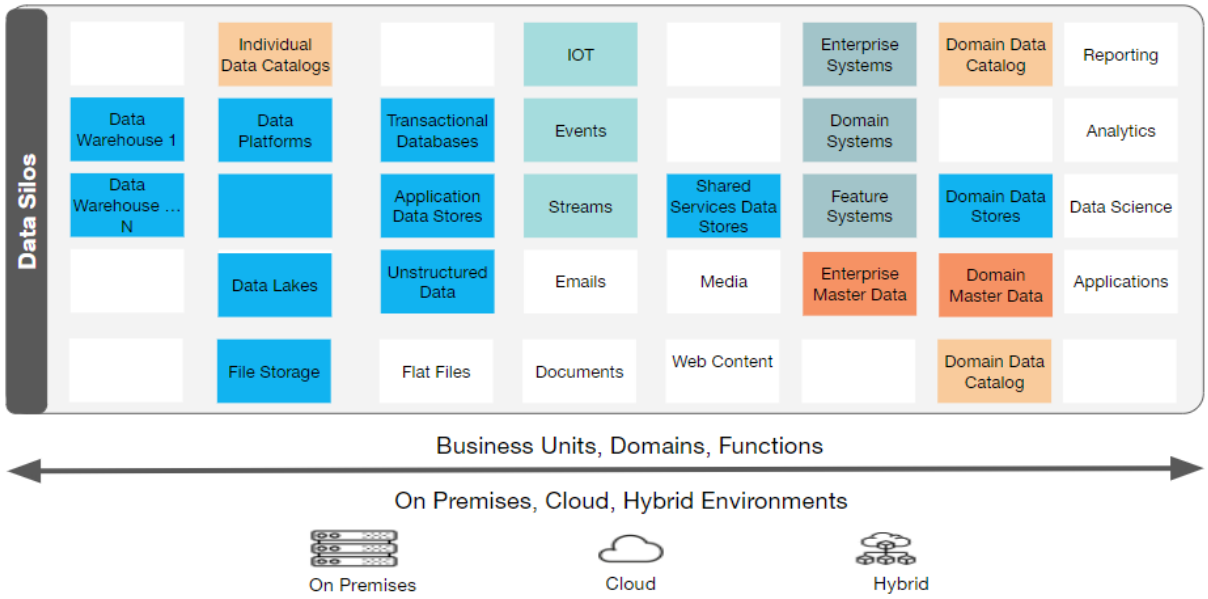

Figure 1. Representation of the diverse Data Ecosystem for the Organization (Source here)

Now they have data everywhere including streaming data, real time transactions, data in enterprise systems such as Salesforce and data in systems which are remnants of mergers and acquisitions. They have different databases, data warehouses, data lakes, reporting and analytical applications running either on premises or on the cloud. Not all of this data is available in the centralized Enterprise Data Platform for analytics. This despite the organization trying to pursue the goal of having a centralized system of record to power their Analytics. In fact, this is not surprising considering that only 6% of respondent enterprises of a BARC survey were actually able to achieve a centralized data platform to power decision making.

Individual departments and business units are making use of their own sets of tools based on their ease of use and creating their own data assets using business rules and domain knowledge. These data assets are useful but not being leveraged by other teams as they are not discoverable. A representation of the challenges being faced within an enterprise due to a diverse data landscape are provided below:

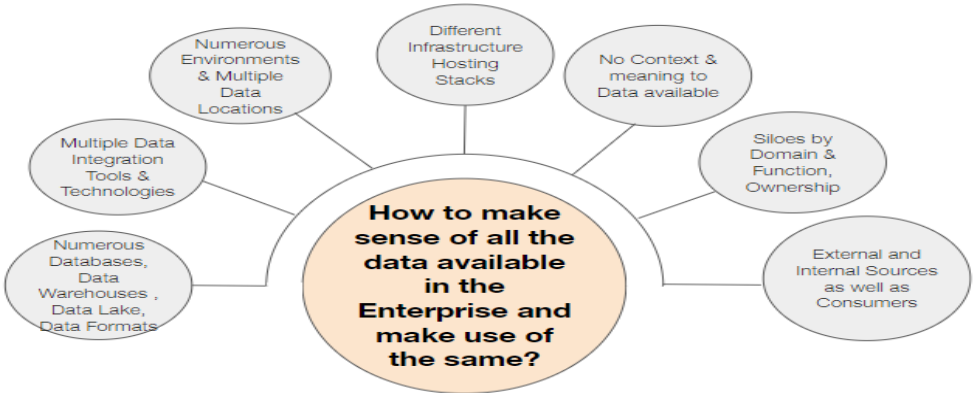

Figure 2. Challenges within today's diverse Enterprise Data Landscape (Source here)

The company wanted options to democratize data and provide access to the right data to the users who need it at the right time. They wished to have a better understanding of the data, its location across environments either on premises or on the cloud, the value of the data asset and a way to make the data accessible across the enterprise without worrying about centralization of the data.

An enterprise Data Fabric architecture offers a new way forward to resolve this kind of a situation which is common nowadays for many enterprises (especially the large ones). The data fabric brings together data assets from different data sources across silos to create an information map that can help power business. Put simply, a Data Fabric architecture can be utilized to support the full breadth of today's complex and connected enterprise.

NOTE: It is not the intent of this document to imply that a Data Fabric capability replaces a Data Platform capability, rather, that a Data Fabric capability can be used in addition to a Data Platform capability by adding "data intelligence" aspects which will be described in following sections.

The Context

The Data Fabric concept was coined by **Forrester** in 2016 and as per them it is meant to be an abstraction layer which can link disparate data assets for better data management, discovery, access and use. As per Forrester it should provide self-service capabilities, as well as some graph capabilities to identify connected data.

Over the last couple of years, **Gartner** has taken the lead in driving thought leadership for the industry for Data Fabric. Gartner defines data fabric as "A design concept that serves as an integrated layer (fabric) of data and connecting processes. A data fabric utilizes continuous analytics over existing, discoverable and inferenced metadata assets to support the design, deployment and utilization of integrated and reusable data across all environments, including hybrid and multi-cloud platforms."

The nuances and goals of such a definition need to be comprehended to get a proper understanding of the Data Fabric architecture.The intention of this document is to unpack why Data Fabrics have emerged, where it fits into the Enterprise City Map and provide an overview of the Data Fabric architecture, its key pillars and its internal components / layers as well as related taxonomy. In the subsequent sections, we will also discuss how a Data Fabric aligns with GlobaLogic's Enterprise Data Platform reference architecture and a high level approach to implement the Data Fabric architecture.

Scoping within the Enterprise

The reality is that data silos still exist in enterprises! They may exist due to very good reasons such as separation of certain data storage to comply with regulation or due to the need for local control and governance for business reasons. Data silos may also exist if there are huge costs, efforts and risks in consolidating or modernizing legacy systems. Or it may be due to political reasons where data or business owners are loath to relinquish control.

Below is a view of where a Data Fabric fits into the Enterprise City Map. It is the opinion of this author that a Data Fabric is an evolution of the Enterprise Data Platform capability which can leverage the relevant capabilities already existing in the Data Platform and add additional capabilities to create the information network of data assets in the enterprise for business use.

The data in various data repositories, system of records, data stores, data lakes, data warehouses and even data platforms continues to stay where they already are. These platforms and systems continue to be used as well by their respective owners or business units or domains. The focus is on using metadata to create the data fabric which can be used across the enterprise for discovering and accessing data in whichever system or infrastructure it may be. This is in contrast to a centralized data platform where there is a heavy focus on moving data from different systems of records and data stores for storing and aggregating it in the centralized enterprise data platform.

NOTE: While Data Fabrics & Data Catalogs both leverage metadata, a directional difference is that Catalogs leverage metadata in order to establish data dictionary aspects where-as Fabrics leverage metadata in order to establish an interconnected knowledge network for understanding & accessing data.

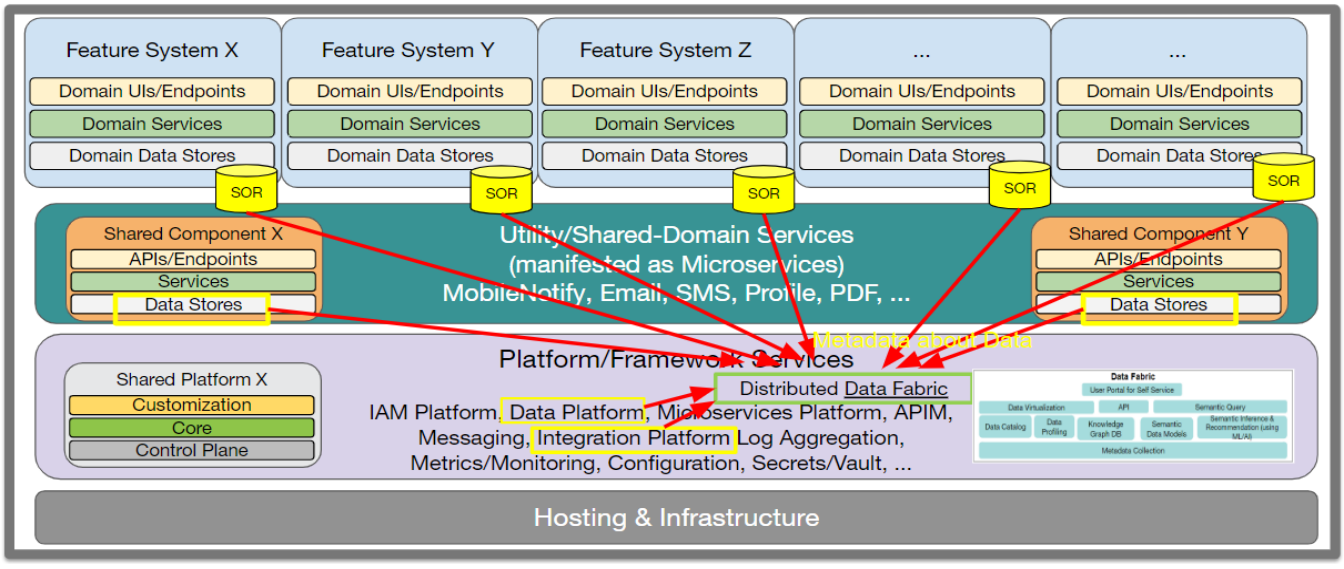

Figure 3. Data Fabric within the scope of an Enterprise City Map (Source here)

A Data Fabric can encompass the data ecosystem already in place in the Enterprise across domains, feature systems and data repositories. It can enable a distributed data platform architecture as it is meant to utilize metadata about data wherever the data may be located or available in the enterprise landscape & add details about relationships, interconnections, and context for better usability.

What is a Data Fabric?

A **Data Fabric** is an architecture which intends to connect and provide the complete knowledge and holistic view of data assets across an enterprise utilizing metadata for self-service data discovery, data access and data use.

Data Fabrics and Catalogs can both manifest an inventory of data assets across the enterprise but the purpose of this manifestation and how they do so differ. Also a data catalog is a layer within the Data Fabric architecture.

A data fabric manifests the inventory of data assets across the enterprise with its details, meaning, interconnections & relationships. This allows understanding, accessing and querying data based on this knowledge (manifested in the Knowledge Graph). While a catalog is a collection of metadata that describes the data assets within an enterprise but does not provide direct access to the data itself.

There have been various perspectives on what a Data Fabric should be. But industry is settling towards a standardized view of the Data Fabric architecture. This standardized view of the Data Fabric architecture enables support for discovering, understanding, accessing data from all sources across the enterprise by all users for their use cases.

Below is a simple depiction of the Data Fabric architecture with its key layers.

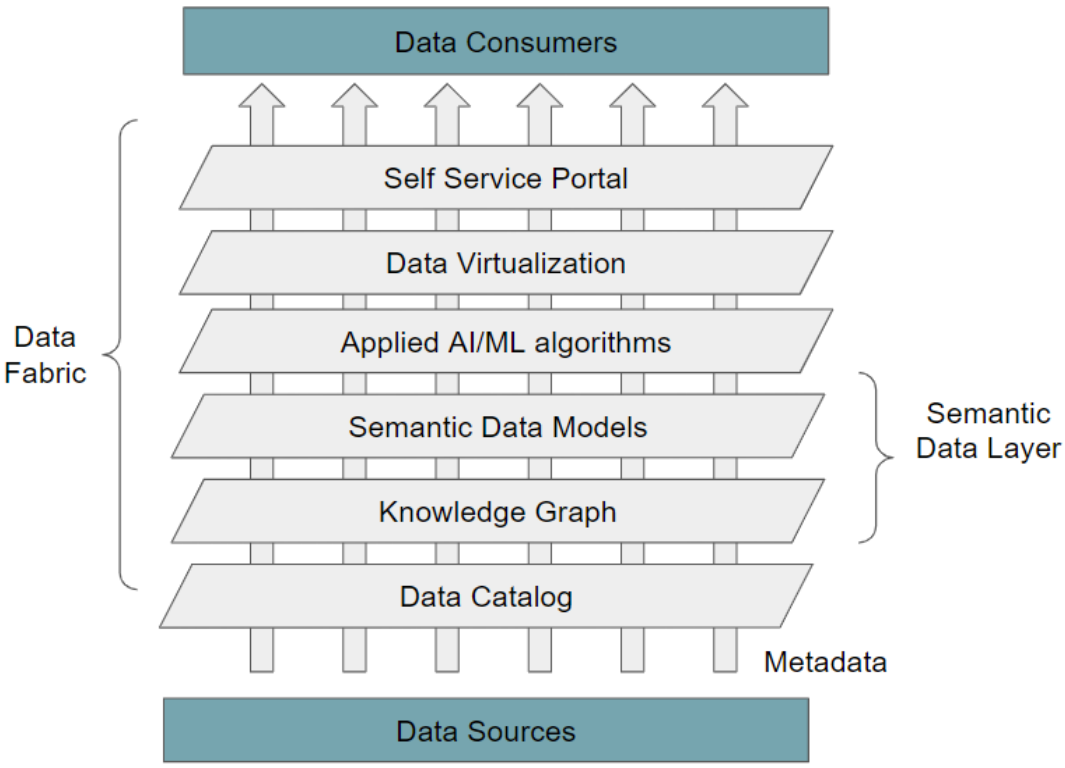

Figure 4. Data Fabric Architecture & its layers (Source here)

As depicted, a Data Fabric architecture consists of Data Catalog, Knowledge Graph, Semantic Data Models, Applied AI/ML Algorithms layers along with Data Virtualization for data access and Self Service user portal.

We will discuss details of each layer in the context of the Data Fabric subsequently but before we do so, a few key points on the Data Fabric architecture:

- It is an Architecture and not a product
- There is no single tool currently to provide data fabric capabilities and it needs multiple components.
- Its foundation are the data assets and sources across the enterprise and their metadata.
- It utilizes metadata about data assets and data to build the information network for use by data consumers
- It Intends to use a semantic data layer to
    - Connect Data Assets across the enterprise
    - Provide Unified Knowledge and a consolidated view about Data Assets
    - Infer relationships and patterns across data assets by applying AI & ML
- Enables simplified data access through Data virtualization
- Has a Self-Service layer for data discovery, access and consumption

Let us unpack the taxonomy and the various layers / components and how it translates to value for the enterprise.

Taxonomy

Metadata

Metadata provides information about data assets and is defined as data about other data. With Metadata - the structure, nature, and context of the data can be understood.
Metadata can be of various types:

- **Technical Metadata**: which covers data's defining aspects such as data types, schema, fields / columns etc. This can include descriptive and structural information. For example, for a customer records dataset, it may include the name of the dataset, number of records, data types of the fields (such as name, address and phone number) along with information such as format of the dataset, type of compression to be used for storage etc.
- **Operational Metadata**: which is focused on data's operational aspects in various systems, applications and data pipelines. For example, for the same customer records dataset - operational metadata will include information of date of creation, data of updates, source of the customer records, lineage, use of the dataset in applications and pipelines along with error, execution information apart from provenance (change & versioning) details etc.
- **Business Metadata**: covers the descriptive business information about data such as business glossary, business tags and classifications for example.
- **Social Metadata**: is about social aspects of data with respect to usage, ownership, accountability, handling, governance by people and entities. For example, this would include details about frequency of access, access control information, feedback/comments/tags about

data by users, data owners & stewards for the dataset, roles and expertise of people using the data.

- **Passive metadata** is just technical metadata collected and curated from systems without any linkages between them. This is typically static and may need manual curation efforts as well.
- **Active metadata** is data that defines data, plus data and context about everything that happens to the data and is done to the data. So it's operational, business, and social metadata tied and linked to the technical metadata.

For a Data Fabric, metadata is the starting point to create the knowledge network of data assets across the enterprise.

Data Catalog

A data catalog is an organized collection of metadata which covers the detailed inventory of data assets. A Data catalog can store collected technical, business, and operational metadata which can help organizations manage and understand their data.

While both a Data Catalog and Data Fabric leverage metadata, the main difference is in their focus and purpose. A data catalog is primarily focused on organizing and using metadata to establish data dictionary aspects. In contrast, a data fabric is focused on using metadata to establish a network of interconnected data details, relationships, context and meaning so as to enable data to be understood, shared, used and accessed across an organization. Also a Data Fabric can make use of the information already available in Data Catalogs.

Semantic Data Layer

Semantics is concerned with meaning.

A semantic data layer is concerned with representation of data with its meanings and context and provides the framework to do so. It can enable accessing and understanding data using standard real world terms — such as customer, recent purchase, and prospect. It also provides human-readable terms to data sources that otherwise would be impossible to discover (e.g., table slsqtq121 becomes Sales West 1st Quarter 2021).

A semantic data layer includes the Semantic Data Models & Knowledge Graphs. A semantic data layer provides the overall infrastructure and framework for knowledge graphs and semantic data models to operate and provide value. The semantic data layer enables a consistent mechanism of organizing and describing data so that this knowledge can be queried and accessed in a meaningful way.

To provide more context, let’s consider an example of a semantic data layer for a healthcare organization which is represented below.

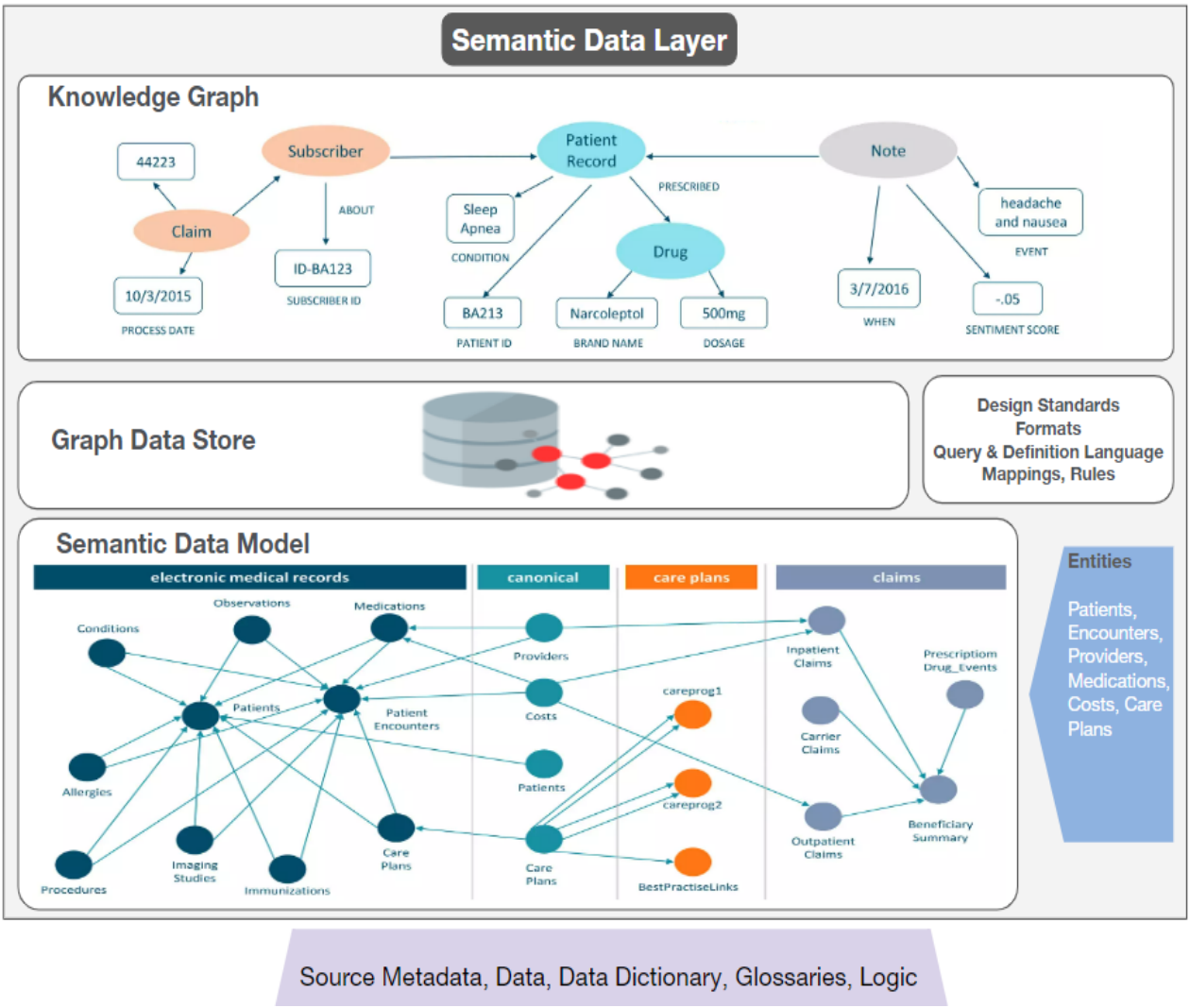

Figure 5. Semantic Data Layer for an healthcare example (Source here)

The healthcare organization has large and complex datasets related to patients, treatments and outcomes which are stored in different systems. The semantic data layer can include a set of standardized terms and definitions for the data elements - such as patient demographics, medical conditions and treatment outcomes which represent entities. The entities will be an input for the Semantic Data Model to capture and navigate the data relationships. It will include the physical Graph Data Store for storing the knowledge. It will also include rules, guidelines and mappings for organizing and linking the data elements, such as how to represent relationships between patients, treatments, and outcomes which will manifest the knowledge of data. The knowledge will be represented as a Knowledge Graph.

The knowledge manifested by the semantic data layer can be used by doctors and other hospital staff to find and access the information they need. This can also be used by applications such as patient or treatment management systems to make more intelligent decisions about how to use the patient data for uses such as personalized recommendations for treatment or anomalies in health indicators.

Next, let's understand how Semantic Data Models and Knowledge Graphs make up the Semantic Data Layer.

Semantic Data Models

A semantic data model is a data modeling technique / way of representing the data in the semantic data layer using a formal structure where the real world entity classes and relationships are first class citizens.

Essentially Semantic Data Models is a type of Data Modelling similar to Relational Data Modelling in DBs or Dimensional Data Modelling in DWHs but in this case is applied to form the Semantic Data Layer. The main value of semantic data models is that it provides a well-defined model which allows the knowledge in graphs to be represented in a structured and consistent manner.

Before we dive in further, let's understand a few basic concepts:

- Entities can be real world objects (things, places, people, organization) and abstract concepts (incidents, activities, professions).
- Semantic description indicates the meaning of an object or relation, e.g. Orders is a table, Barack Obama is a Person.
- Relationships are logical connections between two entities.

Let's take the case of an Insurance company. An insurance company is concerned with issuing policies and processing claims by its customers for insured items. To create the Semantic Data Model, the entities that are relevant for the insurance company would need to be identified such as Customers, Insured Items, Policies and Claims. While a semantic data model can be created using different mechanisms and tools, let's take the example of using Stardog to do so.

Using Stardog's Designer tool, the semantic data model can be created visually on a canvas. The initial step is to create the specific classes for each of these entities such as Products, Sellers, and Orders.

Attributes can then be added for example:

- Name, age, and address for the Customer entity class
- Model, Value for Vehicles class which can be insured item
- Policy type, coverage amount, insured item for the Policy entity class
- Date, Amount for the Claim entity class

Next, these classes can be linked together using relationships that represent the connections between the entities such as "Customer has Policy", "Customer owns Vehicle", "Policy covers Vehicle" and "Claim is against Policy". This would create a semantic data model that represents the data and relationships for the insurance company. Once the semantic data model has been defined, the same can be used to constrain the data in the knowledge graph using restrictions or constraints for a particular entity (for example a Customer must have an Address property).

Below is a depiction of the Semantic Data Model for the Insurance example created using Stardog's Designer tool.

Figure 6. Creating a Semantic Data Model using Stardog Designer (Source here)

While it is not necessary to define a Semantic Data model before creating a knowledge graph, it is beneficial to do so as a Semantic Data model provides a way to organize and represent the data in the knowledge graph along with the meaning and structure of the data. This makes it easier to query, analyze and reason about the relationships between different entities in the knowledge graph. This can also be represented visually without needing technical knowledge about data models in general.

Knowledge Graphs

A knowledge graph represents a network of real-world entities such as objects, events, situations, or concepts and illustrates the relationship between them using nodes and edges. Entities are represented by nodes while relationships are represented by the edges.

Let's take the Insurance company example forward. Once the semantic data model has been created, the information and knowledge about entities & instances and their relationships can be populated in the Graph Database. Metadata from different sources such as Claims Reports, Policy transactions, Customer master data can be gathered / mapped to the entities using tools like Stardog, AWS Glue, Gremlin or other mapping mechanisms.

After the mappings have been completed and knowledge about entities, instances, relationships and semantic descriptions has been added, the connected knowledge network will get manifested as a Knowledge Graph depicted below for the insurance example:

Figure 7. Knowledge Graph example for Insurance (Source here)

Stardog's Designer tool provides capabilities to connect to Data Sources and map the metadata from sources such as datasets with information about claims, customer demographics and risk factor to map the data to the appropriate attributes. This would add the knowledge about demographics and risk into the graph. Below depicts mapping metadata & contextual knowledge into the Semantic Data Model to create the knowledge graph for the Insurance example using Stardog Designer.

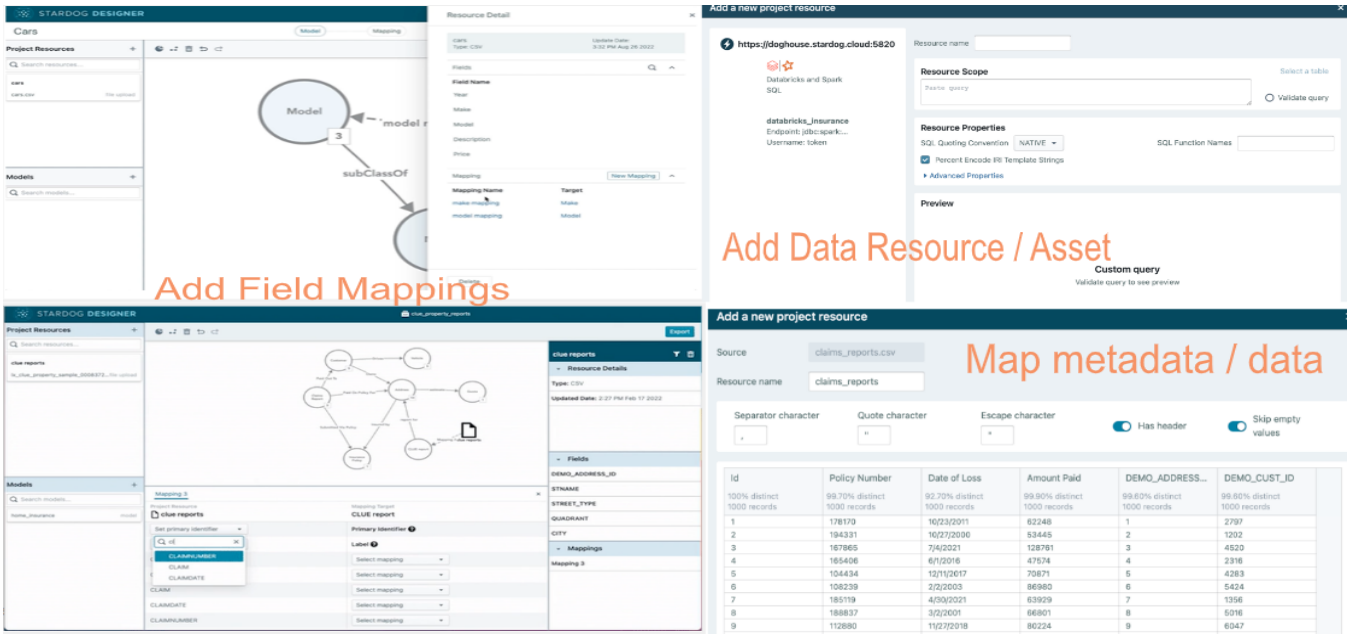

Figure 8. Adding knowledge to the Semantic Data Model using Stardog Designer (Source here) to create the Insurance Knowledge Graph in Figure 6

A knowledge graph combines characteristics of a database which stores the graph information and can be queried. The graph network can be analyzed like a network data structure and a knowledge base as the data has context and meaning associated with it.

A knowledge graph can also link data coming from disparate sources and formats in the order of billions for extracting insights using metadata stored in Graph databases.

Knowledge Graphs are widely used especially by search engines such as Google to provide additional context and information about a searched entity such as a person and support automated reasoning. For example, a search on Michelle Obama will provide search results and additional context about her along with her relationships. These details are powered by Google's knowledge graph on billions and billions of facts about billions of entities across the web.
In addition to Google, knowledge graphs are used by tech giants such as Facebook, Amazon, Apple, Twitter and industrial manufacturers such as Siemens.

It is important to have a business sponsor and funding sorted for implementing a Knowledge Graph for an Enterprise wide Data Fabric as it is not a trivial activity. This is covered in a section later in this document.

Semantic Inferencing

Inferencing based on semantic meanings and relationships is called Semantic Inferencing.

How can Semantic Inferencing help business? Let's take an example cited by Ontotext below.

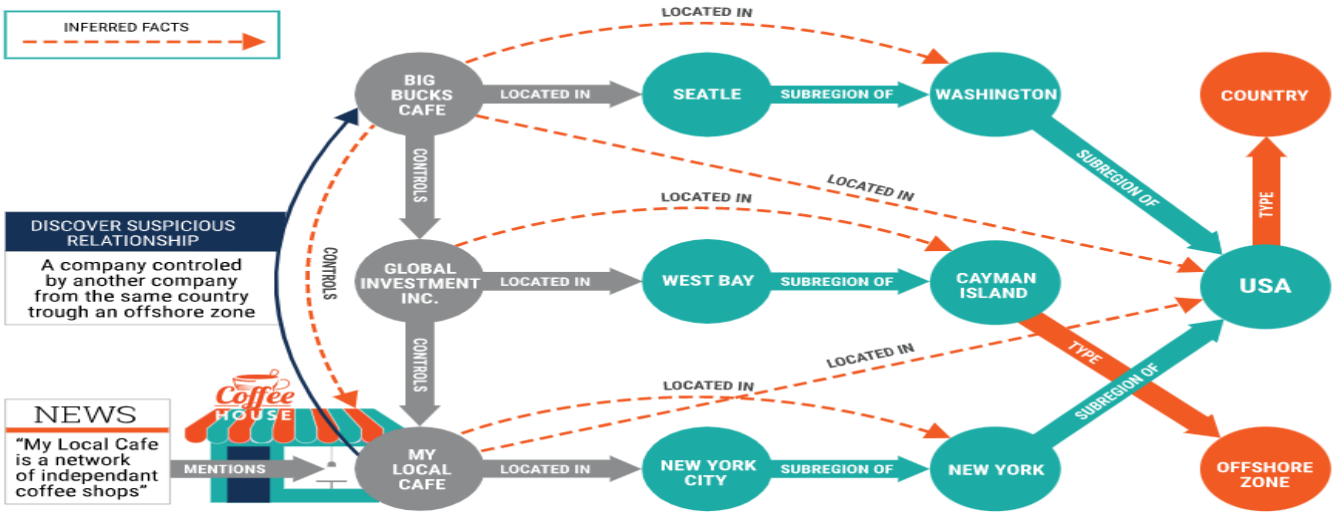

Figure 9. Semantic Inferencing business example (Source here)

Let's say the business is involved in contracts between companies and needs to know if there are unknown ownerships or suspicious relationships amongst them for compliance reasons. Given that there is information about ownership of Global Investment by Big Bucks and separately that Global Investment controls My Local Cafe. In such a scenario, information based on news on My Local Cafe being independent can be refuted by inferring that Big Bucks Cafe controls My Local Cafe.

Semantic Inferencing can leverage recommendation / inference algorithms to learn feature vectors, uncover relationships and patterns using similarity & graph based approaches. AI / ML algorithms can be applied on knowledge graphs to learn feature vectors, uncover relationships and patterns using similarity & graph based approaches.

There are tools such as Informatica CLAIRE and Stardog's Semantic Inference Engine which already apply predefined AI / ML algorithms for semantic inferencing. Informatica CLAIRE applies machine learning models on metadata knowledge graphs for entity matching, inferring similar data & fields, discovering relationships among data, automating data integration using schema matching and recommending data sets of interest.

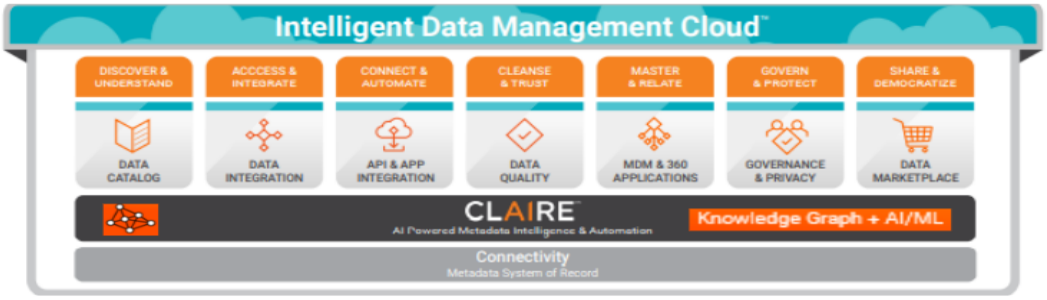

Figure 10. Informatica CLAIRE engine within the Informatica solution landscape (Source here)

Informatica is delivering an integrated combination of metadata and AI/ML with CLAIRE and this has been used for delivering Data Fabric solutions for their clients like BMC. The metadata already available in Informatica Data Catalog provides a vast trove of information for CLAIRE to create the Knowledge Graph and apply built in algorithms to make intelligent recommendations, automate the development and monitoring of data management projects and adapt to changes from within and outside the enterprise. More details are available here.

Below is a representation of the application of Machine Learning algorithms on the Semantic Data Layer for Semantic Inferencing using similarity and graph based approaches

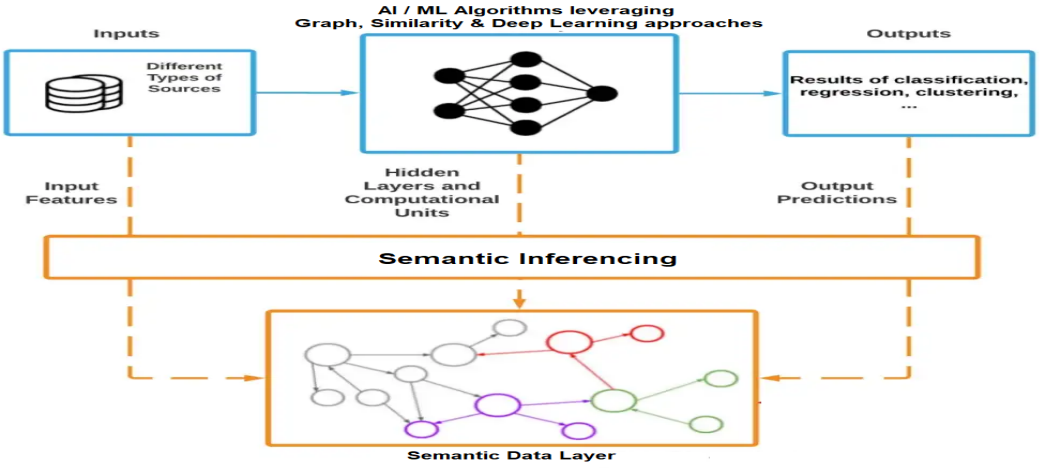

Figure 11. Semantic Inferencing schematic representation (Source here)

Separately, semantic inference algorithms can also be applied using tools such as Databricks / AWS Sagemaker / Azure Machine Learning.

Data Virtualization

Data virtualization is a mechanism to connect to source data directly without the need to move or copy data. It eliminates the need to move or store data at different places. Because of this, data can stay where it is and it can also help enterprises avoid the redundancy of duplicated data while saving time and effort needed for replication / aggregation if there is no actual need.

Some tools which can be used for data virtualization (out of many more) are mentioned below:

- Denodo: Can create virtual views of data and integrate it with other systems and applications.
- Informatica Data Virtualization: cloud-based data virtualization platform
- Red Hat Data Virtualization: Is an open-source data virtualization platform
- AWS Athena: a serverless query service that enables analysis of data in S3 & RDS using SQL.
- Presto: a popular open-source distributed SQL query engine that is designed for high-performance querying of large-scale data sets.

Data Fabric Logical Architecture

Now that we have defined the taxonomy related to a Data Fabric and scoped it within the Enterprise City Map, let's drill down further into the logical architecture of a Data Fabric and the purpose of the different components.

The logical architecture is represented below along with the metadata sources integrating into the fabric, the end users and the use cases of a Data Fabric architecture.

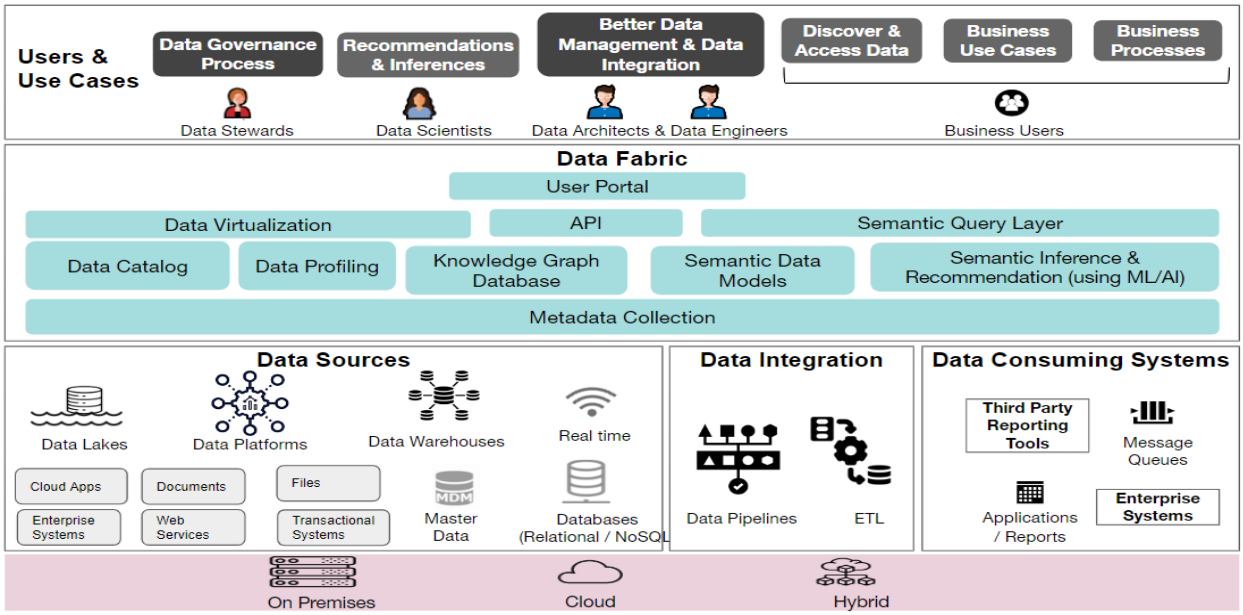

Figure 12. Data Fabric Logical Architecture blocks (Source here)

Below are the key logical components in the Data Fabric architecture:

- **Metadata Collection** Layer for collecting metadata across data sources, data integration pipelines and data consuming systems

- **Data Catalog** to store data categorization and metadata augmented with **Data Profiling** of data and metadata
- **Knowledge Graph Database and Semantic Data Models** to extract semantics and create semantic data models graphs for the data & metadata and store the relationships between the data.
- **Inference and Recommendation layer using ML & AI** which can apply analytical / data science models on the metadata collected and semantic data models to suggest improvements for data management and data integration apart from inferring data relationships.
- **Semantic Query layer** to query and discover relations, inferences and recommendations
- **Data Virtualization** for federated data access
- **User Portal** for use by end users to discover, explore and access

Now why are all these components needed in a Data Fabric? Can a Data Catalog not suffice to provide a Data Fabric? Why is a Knowledge Graph needed at all along with the other components?

The answer lies in the definition of the Data Fabric itself as a Data Fabric architecture will enable a connected knowledge network of disparate data assets.

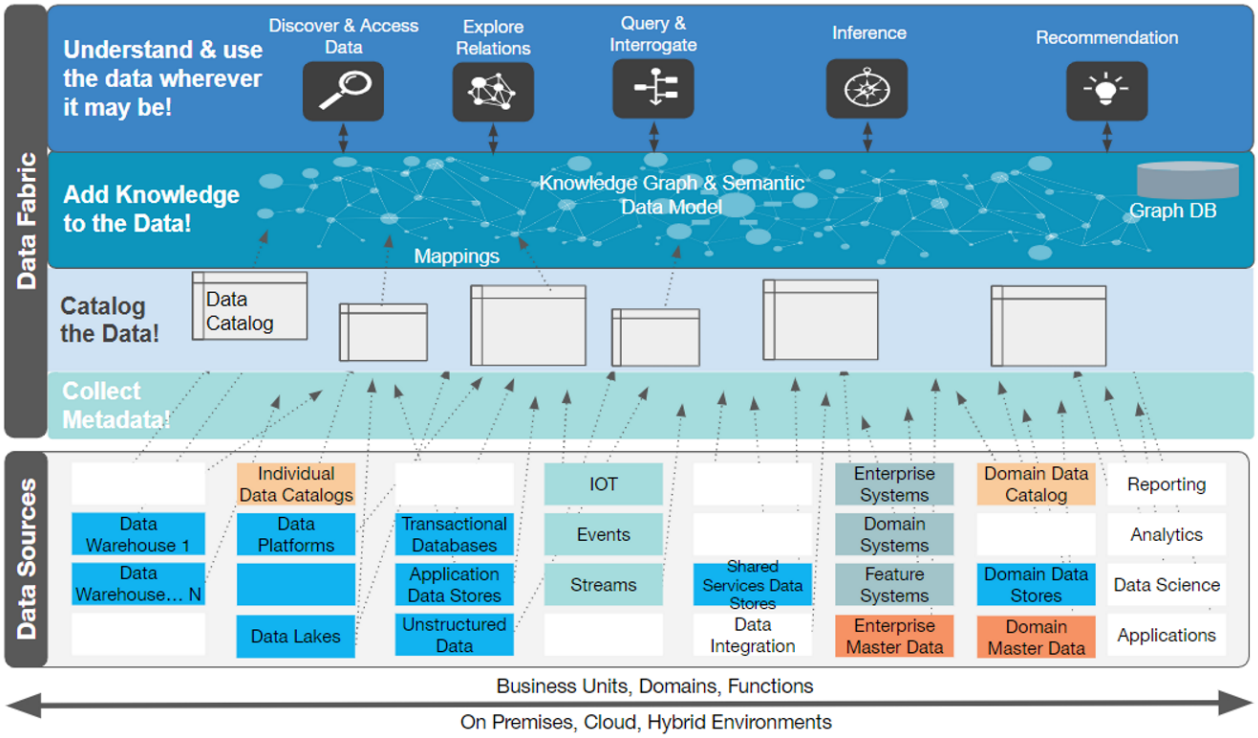

Figure 13. Data Fabric - connecting disparate Data Sources (Source here)

The data assets need to be crawled and all of the metadata for the data assets and its current usage needs to be collected. Once the data assets have been crawled, the technical and business metadata can be stored in a Data Catalog. Once this information is available in the Data Catalog, users / data consumers can definitely use the catalog to understand datasets better.

What happens at this point is that people start realizing that enterprise data is a big mess! There are hundreds of different data sources with thousands of objects such as tables and tens of thousands of fields / columns. If the data consumer knows which data they are looking for, then they can view the dataset's details in the Catalog and then access the data using other data access, query or virtualization tools. Otherwise it is akin to finding a needle in a haystack unless the context and inter connections come to the foreground.

That's where the Semantic Data Layer comes in and this gets built on top of the metadata and the Data Catalog by mapping the information and context to build a Knowledge Graph based on the defined Semantic Data Model. Once the knowledge is available, it can be exposed to consumers to explore, discover query and use the data.

Below is a depiction of the Data Fabric components on GL's Enterprise Data Platform reference model. The GlobalLogic Enterprise Data Platform reference model which brings together many tools and methodologies.

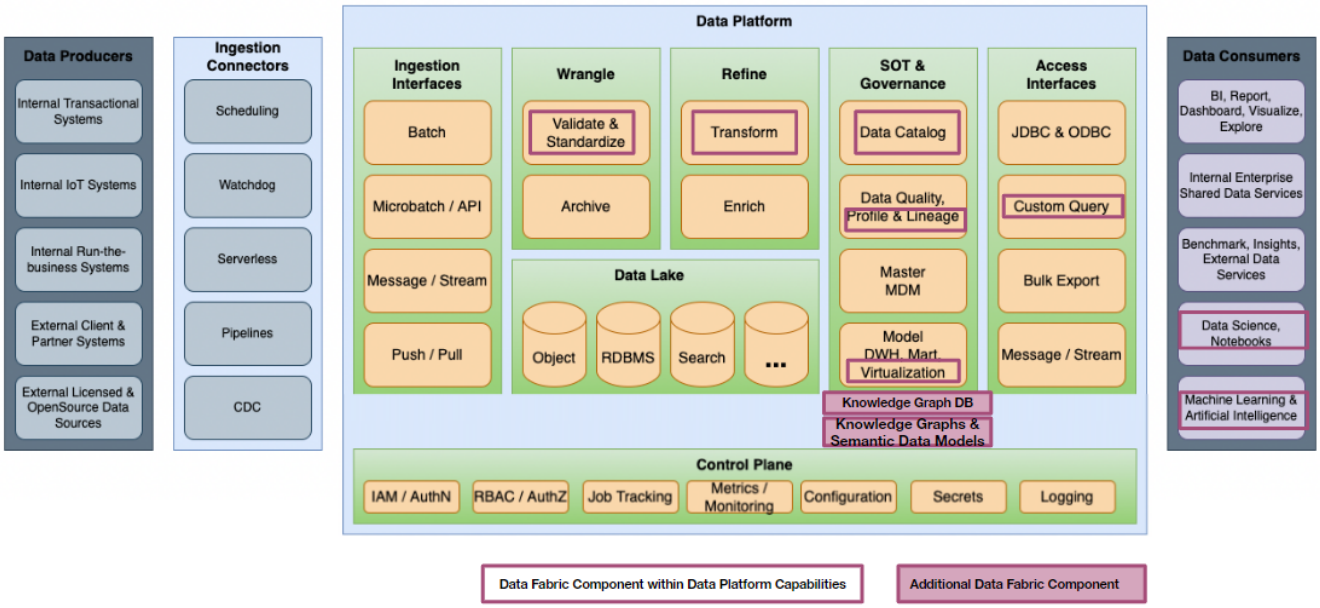

Figure 14. GL Data Platform Reference Model - Updated with Data Fabric Components (Source here)

As is evident, many components are already part of the reference model. A Data Fabric architecture makes use of elements / tools in the Data Governance, Data Access and Data Science pillars which already exist in GlobaLlogic's Enterprise Data Platform reference model and are illustrated in the diagram above.

Implementing a Data Fabric architecture will make use of these components and add Semantic Data Models, build Knowledge Graphs and utilize Applied AI / ML algorithms on top of the Enterprise Data Platform capabilities. The new / additional components are also highlighted.

The amalgamation of these components will enable the Data Fabric architecture on top of an Enterprise Data Platform which will allow discovery and utilization of data silos across the enterprise.

It needs to be noted that a Data Fabric is not a solution to all problems and needs to be applied in particular cases and scenarios which are covered here.

Knowledge Graph in the Data Fabric

A knowledge graph is the core or beating heart of the data fabric. Below is an example of a representational Knowledge Graph which can power the Data Fabric for business use.

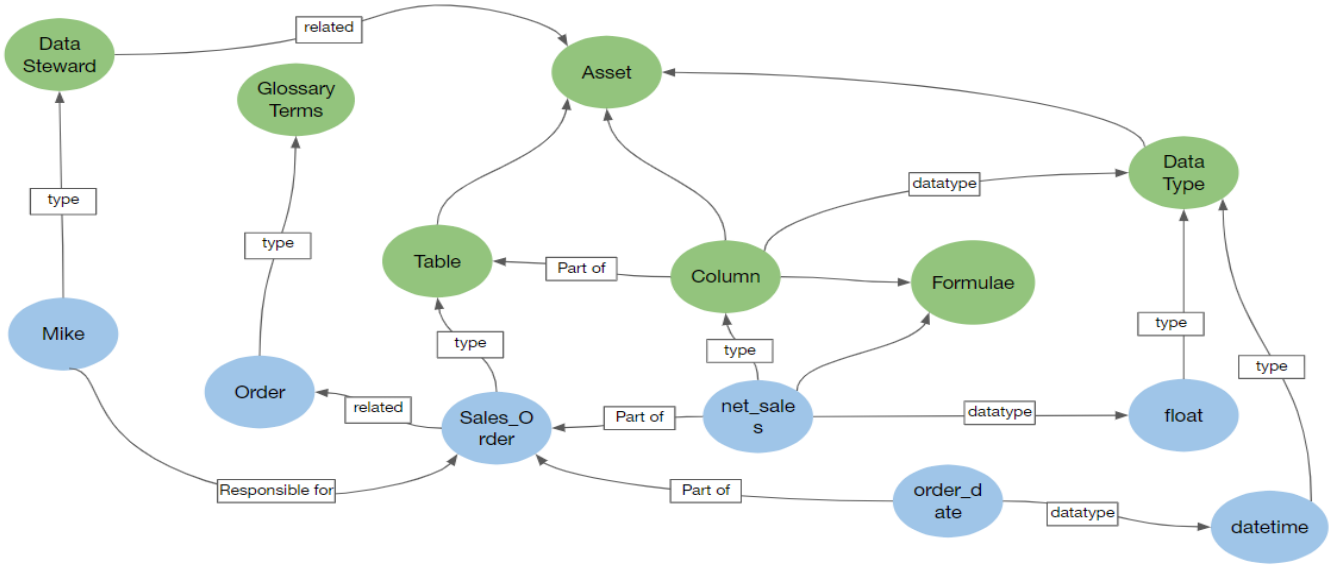

Figure 15. Representational Knowledge Graph in a Data Fabric (Source here)

Nodes in the Knowledge graph represent entities which can be:

Entity Classes, Entity Instances, Tables, Columns, Data types, Business Terms, Pipelines, Transformations, Dashboards, Reports, Users, Environments, Infrastructure, etc as examples

Edges represent their relationships:

Derived from, associated with, related to, owner of, type etc as examples

Green nodes are classes while Blue nodes are instances of the class.

Knowledge graphs can represent everything that happens to enterprise data because they serve as a universal format for data, regardless of its source structure or location or format. It can represent data of various structures and supports multiple schemas. It creates the semantic understanding of enterprise and external data which business can understand and then leverage. The other key thing is Knowledge Graphs can be extended easily to newer data assets, entities and instances of entities. For example the representational Knowledge Graph can exist independently or also be linked to other knowledge graphs which cover Customers or Orders placed by them (for example).

The key to understanding how a knowledge graph links data is to understand that it connects related data, rather than transforming it. Each data object is assigned a unique ID, to which all related information is linked. This unique process allows data owners to continue to maintain control of source data while enabling enterprise-wide collaboration.

Algorithms can be applied to knowledge graphs as they are both human and machine readable to activate metadata & uncover different patterns and learn new relationships amongst data assets.

Self Service User Portal of the Data Fabric

Once the Knowledge is there, consumers can discover data, explore relations and access data and get inferences and recommendations through a self-service layer with virtualized data access.

To gain the maximum value of a Data Fabric, it should be possible for business users who do not have technical backgrounds to use this portal to discover data and its relations easily. This will also effectively provide a simplified and usable portal for other users such as Data Engineers, Data Architects, Data Stewards, Data Scientists, SMEs and domain experts.

This self service user portal should have capabilities to search the Knowledge Graphs and Semantic Data Models. Search should also be integrated with semantic querying. On searching for an entity, all details of the entity can be provided to the user to understand more about the entity and its relationships. The details can include the relationships, the graph visualization, recommendations and graph analytics. The user can then proceed to access and use the dataset.

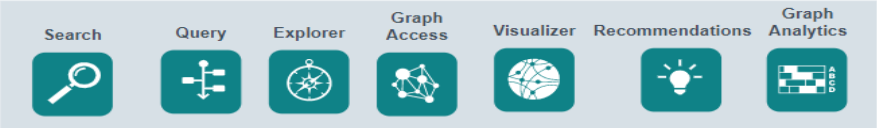

Figure 16. Self Service Portal capabilities of the Data Fabric (Source here)

A data fabric is meant to provide answers to users across the enterprise. This is enabled through its self service user portal which is supported by powerful querying, search, visualization and analytics capabilities. These capabilities can also be exposed via APIs for other applications to use. The querying is done at the compute level against the knowledge graphs and semantic data models above the actual data storage layer. It is at this level that a Data Fabric connects otherwise disconnected data silos.

The self service user portal can either leverage off the shelf tools or be custom built integrated with frameworks so that one doesn't need to start from scratch. In case advanced functionalities are needed or in case the commercial tools do not suffice, custom self service portals can be explored.

Below is a short list of tools which provide a web portal for self service such as:

- Stardog: provides Stardog Explorer for text based and advanced searches along with visualizations which includes data virtualization to access the data as well

Figure 1. Stardog Explorer (Source here)

- Metaphacts: provides UI for visualization, querying, knowledge graph management, knowledge discovery, exploration, analytics, authoring.

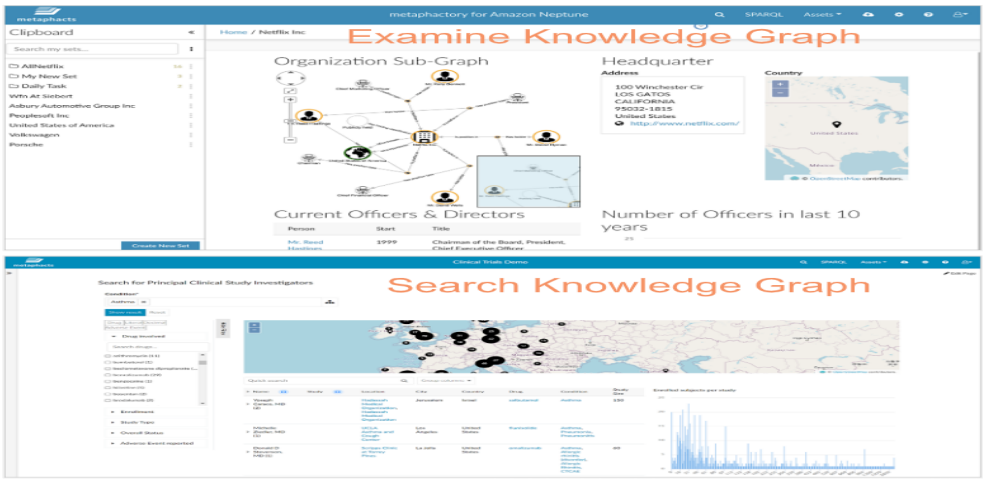

Figure 2. Metaphacts Self Service Portal (Source here)

- Tom Sawyer Graph Database Explorer: has an application for viewing and analyzing connections, networks, and dependencies offering clean, interactive graph layout. It's easy to use and connects directly to Amazon Neptune, Neo4j etc.

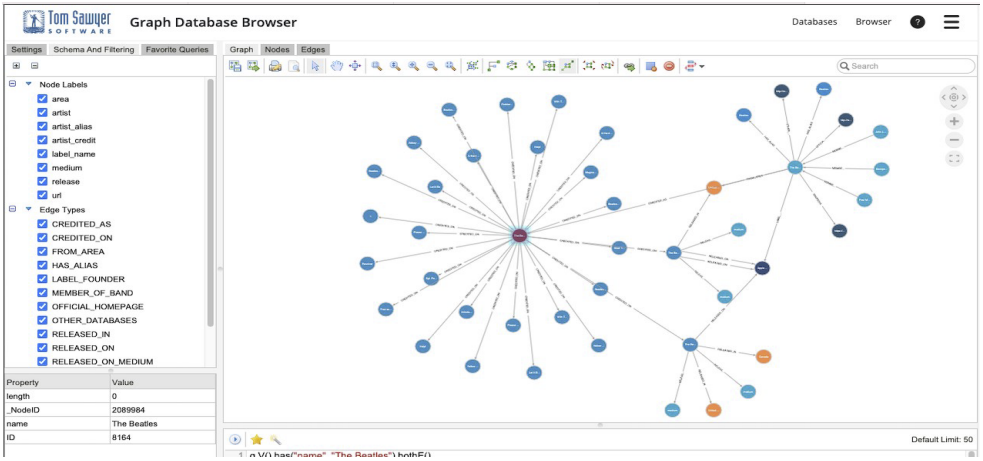

Figure 3. Tom Sawyer Graph DB Browser (Source here)

Low code platforms / Toolkits from Cambridge Intelligence and Graphistry provide frameworks for graph exploration and visualization which can be integrated to a custom self service portal.

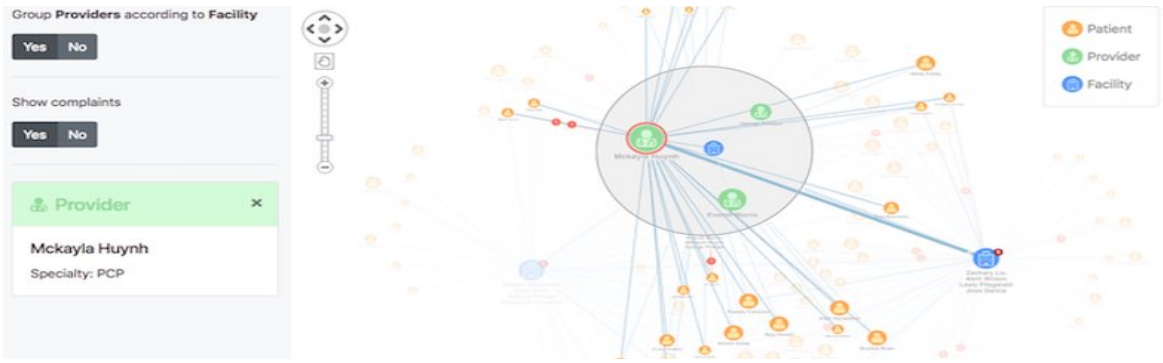

Figure 4. Cambridge Intelligence Toolkit / Framework (Source here)

Below is a custom self service user portal example which would need to have integrations with graph visualization & exploration toolkits (provided for example by Cambridge Intelligence or Graphistry) and enable search and data access on top of the Semantic Data Layer.

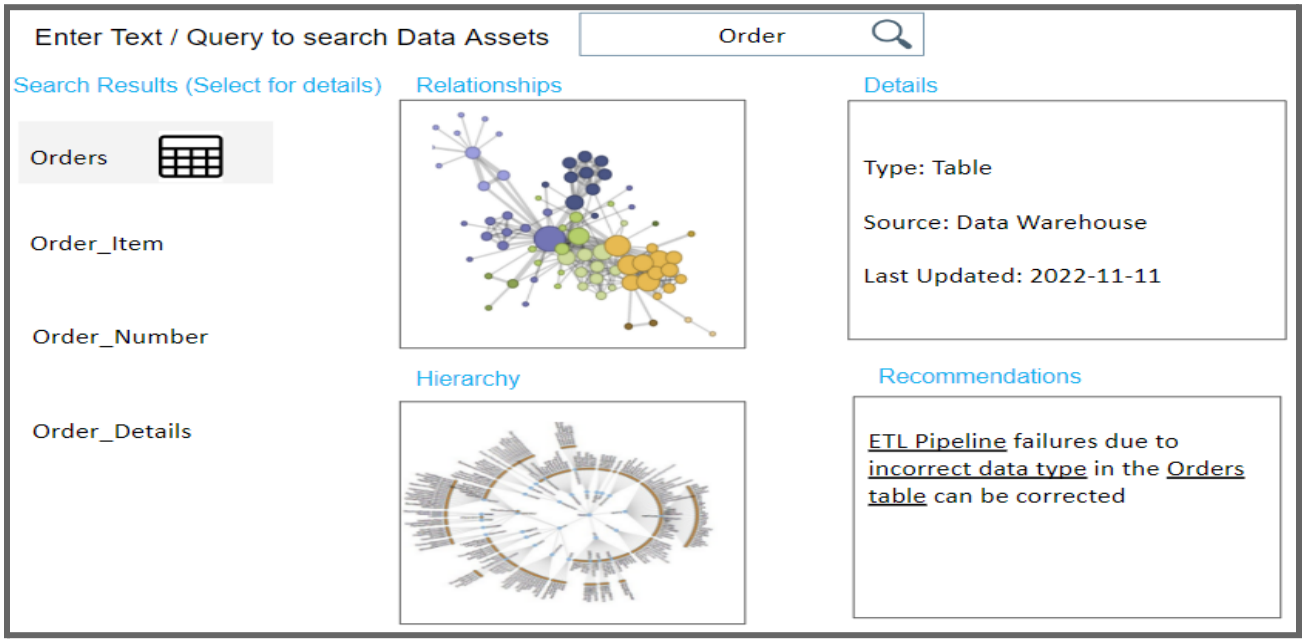

Figure 17. Self Service custom User Portal example (Source here)

With a Data Fabric architecture and its self service layer, Enterprises can easily resolve questions around data management such as "What is the impact of changing a field in a Table across the Enterprise?", "How to improve failures in data engineering pipelines through links between failure and errors?" and also business questions such as "Show me all the materials and suppliers involved in a particular product.", "How is a person related to a department and what skills does he have?" and "Which part can fail because of historical failures?".

This is a game changer as the self service user portal helps less technical users quickly find, access & share data and it also reduces the cycle time of making use of already available data assets. This helps large enterprises unlock enterprise wide collaboration and allows data democratization across the enterprise.

Benefits of a Data Fabric

To reiterate briefly, a Data Fabric architecture enables the below benefits:

- Provides a unified and holistic view of the enterprise data landscape
- Removes silos around data and data systems in the enterprise
- Makes data assets discoverable for business users
- Enables enterprise wide collaboration and democratizes data access
- Provides a mechanism to get more details about data and specific entities / objects
- Utilizes metadata to provide recommendations through AI, ML which can be used for automating tasks for better data management, improving reliability of data pipelines and other actions

When to use a Data Fabric

The question arises as to when should a Data Fabric architecture be employed in the Enterprise context. Is it a good idea to implement a data fabric architecture always or is it better to consider the scenario and check for applicability? What are the considerations for the same?

Data Fabric is an architectural approach which can be considered when:

- Organization has highly interrelated data but is having challenges in unifying data
- Data cannot be discovered due to data silos existing between different domain focused teams
- Many different data assets / data lakes / data platforms / data warehouses exist in the organization which are used independently by business
- Organizational and business context of data is not available
- Data changes frequently and a flexible model is needed to keep track of it
- Data Management is complex
- There exist data swamps
- Data is stored across Multi Cloud environments / Hybrid environments

A Data Fabric may not be needed at all in case

- There is not enough data available in terms of the variety and velocity of data. Therefore, applicability needs to be considered as this architecture may be an overkill.
- The enterprise ecosystem and data landscape is not very complicated where-in a simple Data Lake or Data Warehouse capabilities will suffice.
- A centralized Data Platform already aggregates all data across the enterprise

Data Fabric Initiative Ownership & Sponsorship

It is important to have defined ownership of the initiative to set up or implement the Data Fabric architecture with approved sponsorship and funding. Creating a data fabric is a non trivial amount of work and it will need a sponsor in the business with an actual budget for the implementation part.

In an enterprise setup, typically the unit managing the organization's data & analytics strategy or a corporate office unit like the CIO's office or the CTO's office would most likely own the implementation of the data fabric capability. The specific Business Units involved may vary depending on the specific requirements and goals of the organization as well as the overall business strategy. Ultimately, the decision to fund and sponsor a data fabric capability would be made by the leadership of the organization in collaboration with relevant stakeholders. But this needs to be aligned internally before embarking on the Data Fabric journey so as to avoid issues later on.

Data Fabric Implementation Approach

A combination of tools and technologies (from the same provider such as Informatica or different providers such as AWS, Stardog etc) can be considered for best fitment against the Data Fabric architectural blocks.

While this document does not intend to go into full implementation details of a Data Fabric, a high level approach to set up the Data Fabric architecture is discussed here. Tools and services which can be potentially leveraged on AWS & Azure are mentioned in the subsequent section.

Below is a representation of the end to end setup to power an enterprise-wide data fabric:

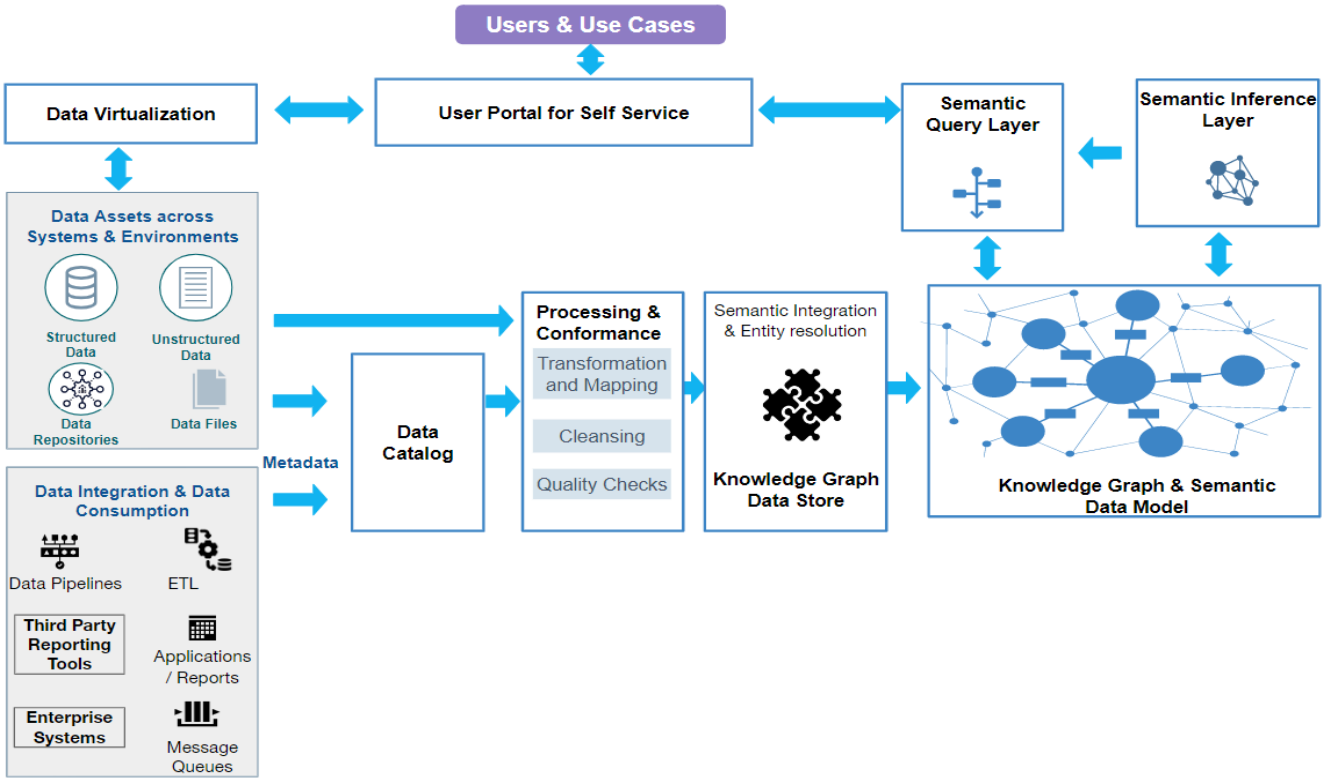

Figure 18. Data Fabric Implementation approach (Source here)

When a data fabric is being implemented, it's best to start small and in a place where the data teams already have familiarity. This place would be tools already being used for data integration and ETL from data sources into the relevant data repositories or for specific use cases and then build / expand on the initial successes.

Existing pipelines can be **extended** to **collect and push metadata** and data tags into **Data Catalogs** on an automated basis. This can leverage existing Data Integration tools already available in the Enterprise. Additionally, new automated pipelines need to be developed to connect to sources and data platforms across environments (on-premises cloud, hybrid, & multi-cloud) and collect the requisite metadata. Care should be taken to ensure the metadata includes details about the origin point of the data, how it was created, what business and operational processes use it, what is its format etc. By making use of metadata and data catalogs for maintaining data's Lineage including its

transformations and usage, Data Fabrics can check data for reliability and ensure that it can be trusted.

The collected metadata will need to be mapped to the **Semantic Data Models** using mappings, domain classes, taxonomies and extracted meanings to create the **Knowledge Graphs**. A good reading on how this was done on data in the Databricks Lakehouse using **Stardog** is available here. Similar processes can be extended to data assets across the enterprise to create the enterprise wide knowledge graph but the key point here is that the semantic layer can be built up steadily focusing on specific areas or data sources and then expand further.

The Semantic Inference Layer can leverage applied AI/ML algorithms on top of the semantic data models and knowledge graph for converting passive metadata to active metadata. Tools such as Informatica Claire are available which can run its own set of algorithms on the Knowledge Graph based on metadata for semantic inference. These algorithms can check how all of the organization data within the scope of the data fabric is working together and which combinations of data are used most often in different business and operational contexts. They can also learn new relationships and uncover patterns using similarity and graph based approaches using supervised or unsupervised learning algorithms. An example of this is illustrated below.

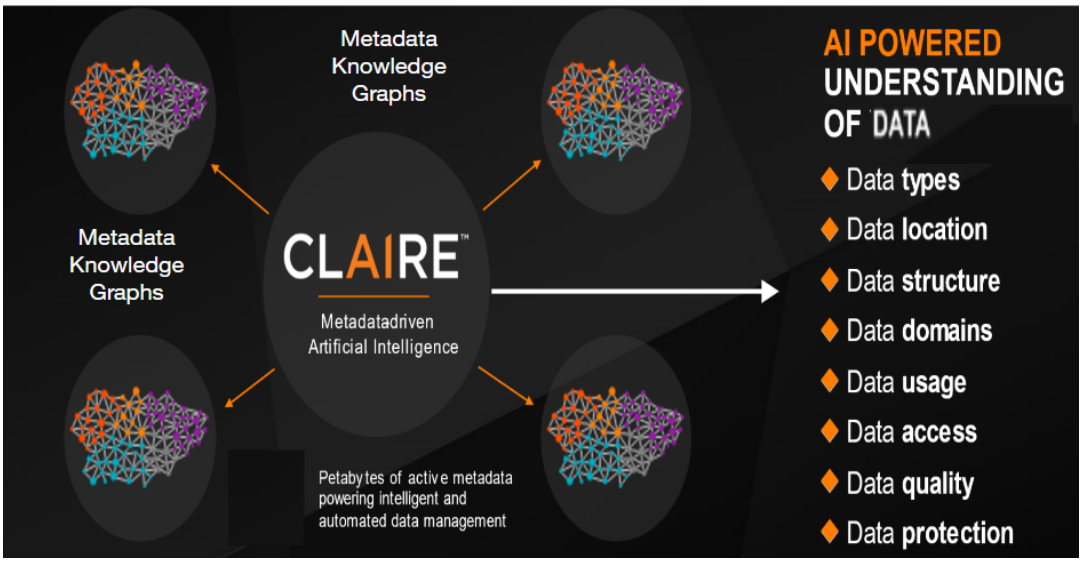

Figure 19. Semantic Inference algorithms applied on Knowledge Graphs in Informatica CLAIRE to learn new relationships and uncover patterns within Data (Source here)

The knowledge graphs and data catalog can be exposed to end users and data consumers with proper Role based Access Control (RBAC) through a **Self Service User Portal** leveraging off the shelf tools / toolkits. The user portal can also be a custom implementation with Search, Querying, Visualization & Analytics capabilities which allows simplified discovery and access to data. **Data Virtualization** needs to be leveraged for accessing & consuming all data.

Data Fabric implementation on AWS & Azure

Now that we have detailed the logical architecture and implementation approach for an enterprise Data Fabric, let's see which tools and services which can be leveraged for implementing a Data Fabric architecture on AWS & Azure.

Below are reference architectures for data fabric implementation on AWS & Azure. Preference is to utilize Cloud native or managed services / tools if available.

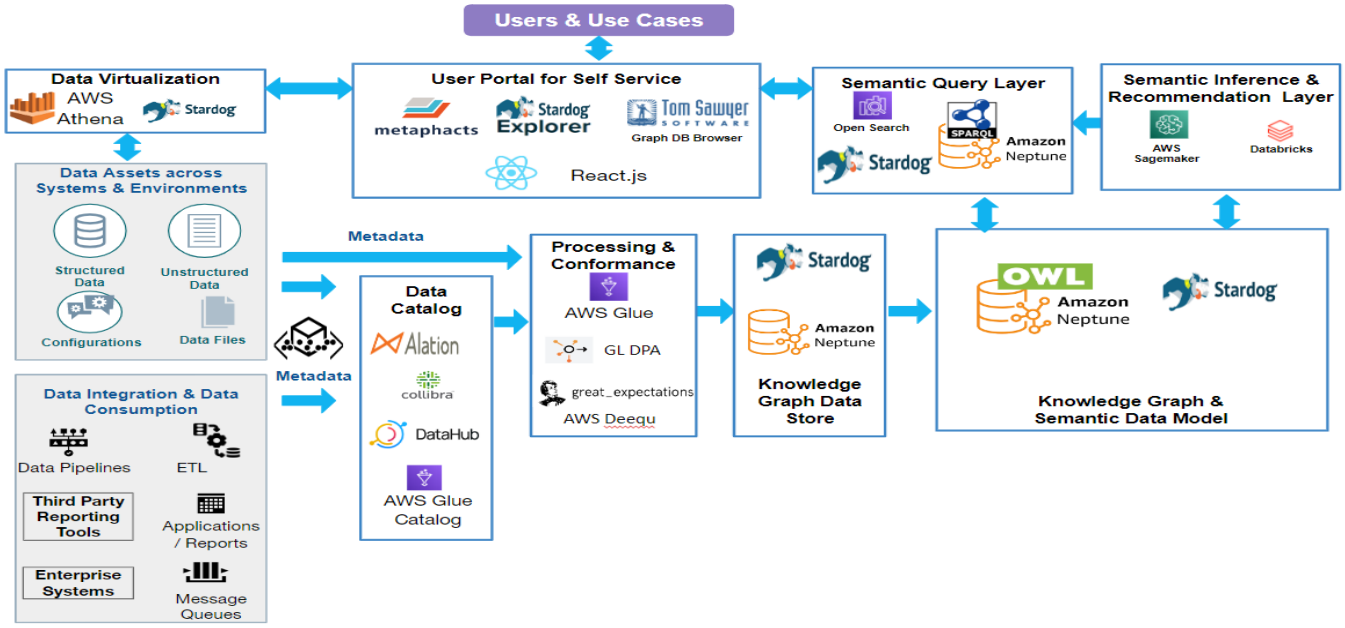

Figure 20. Data Fabric reference architecture on AWS (Source here)

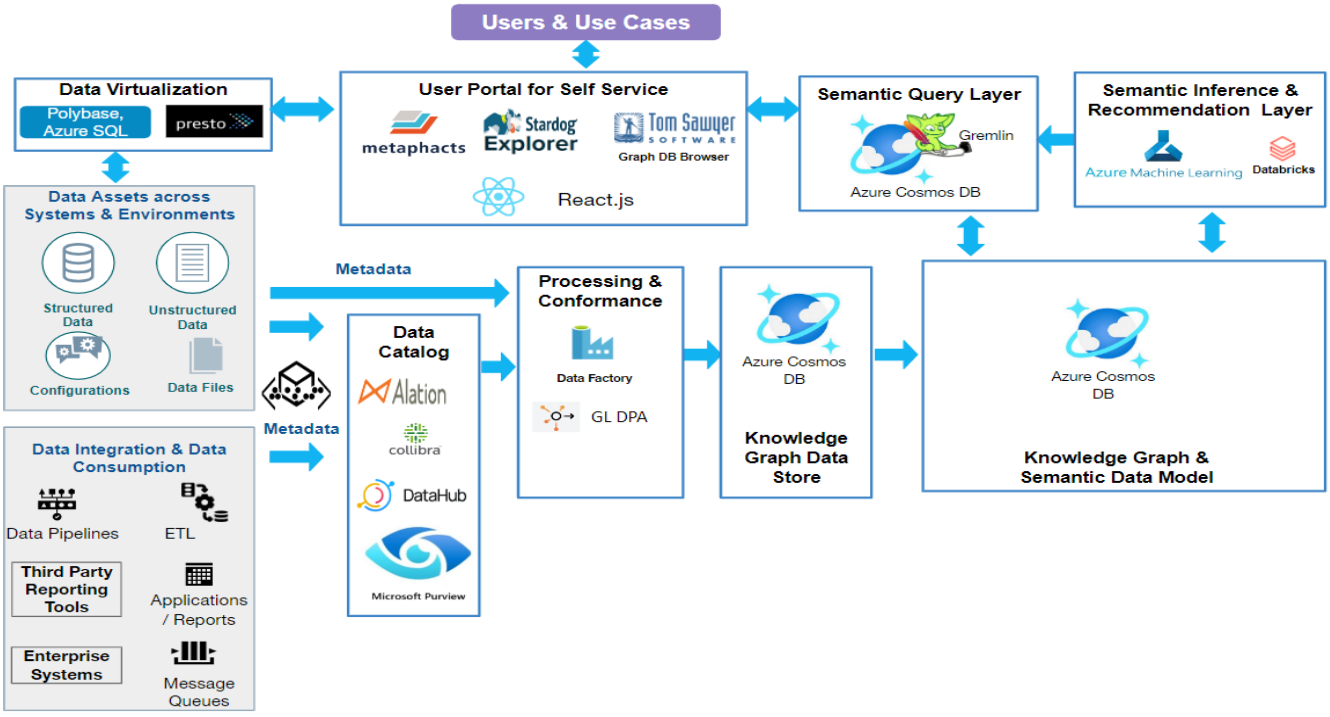

Figure 21. Data Fabric reference architecture on Azure (Source here)

Details of the tools for each layer are provided below:

	AWS	Azure
Metadata Collection	Metadata will be collected by connecting to data sources. Existing data pipelines need to be extended with the data orchestration tools to collect the metadata. Data Catalog tools or AWS Glue can be leveraged to collect metadata. This will form the mainstay of the Data Fabric.	The data pipelines will need to be extended with automated metadata extraction and collection processes. Data Catalog tools or Azure Data Factory can be leveraged to collect metadata. This will form the mainstay of the Data Fabric.
Data Catalog	The Data Catalog with either AWS Glue Data Catalog can be utilized to store the data tags and metadata. Tools such as Alation, Collibra and Dathub can also be looked at for the Data Catalog component.	Azure Purview can be leveraged for the Data Catalog for storing the metadata. This can also be a source for data consumers for discovering the data they need. Tools such as Alation, Collibra and Dathub can also be looked at for the Data Catalog component.
Processing & Conformance	Metadata collected and data profiling available in the Data Catalog will be parsed, mapped and conformed to the Semantic Data Model using AWS Glue and pushed to the Knowledge Graph Database (aka Graph Data Store)	Metadata collected and data profiling available in the Data Catalog will be parsed, mapped and conformed to the Semantic Data Model using Azure Data Factory and pushed to the Knowledge Graph Database (aka Graph Data Store)
Graph Database / Data Store	The vertices, properties and edges for data based on the metadata will be stored in AWS Neptune or Stardog which will act as the Graph Data Store. The Knowledge graph can be utilized for semantic analysis and querying.	Azure ComosDB can be used for storing the graph details of entities.
Knowledge Graph	The data in the Graph data store can be utilized to create Knowledge Graphs in AWS Neptune or Stardog which will allow easier interpretation & discovery of data across the enterprise.	Azure ComosDB can be used for deriving relationships and storing the knowledge as graphs for analysis and querying.

Semantic Query Layer	The Knowledge graphs in AWS Neptune / Stardog can be queried using Graph traversal and query languages (such as Sparql/Gremlin) to explore relationships and meaning.	The Knowledge graphs in CosmosDB can be queried using Graph traversal and query languages (such as Gremlin) to explore relationships and meaning.
Semantic Inference / Recommendation (using AI/ML)	AWS Sagemaker or Databricks can be utilized for running algorithms on the metadata & knowledge graphs for metadata activation & recommendations for data management, data discovery and data integration.	AzureML or Databricks can be utilized for running analytical models on the metadata and knowledge graphs for recommendations and activating metadata.
Data Virtualization	Data access can be virtualized through the use of AWS Athena on AWS which has connectors to connect to different source systems. If Stardog is being leveraged for the Knowledge Graphs, Stardog can also be utilized as it has a virtualization layer.	SQL Server's PolyBase and Azure SQL Managed Instance can be used on Azure for Data Virtualization. Presto can also be deployed on Azure VMs for use as a distributed query engine.
Self Service User Portal	A user portal on top of these engines can be leveraged for self service. The self service portal can utilize existing off the shelf tools or will need to be a custom web application in case the capabilities provided by the tools are not enough. Off the Shelf tools are available on AWS such as Metaphacts, Tom Sawyer Graph Browser or Stardog Explorer. Self service user portal can also be a custom built portal using React.js with search and visualization capabilities for data entities. The custom portal would need to have integrations with graph visualization & exploration toolkits (provided for example by Cambridge Intelligence or Graphistry) and enable search powered by Opensearch which is mapped to the Semantic Data Models.	A user portal on top of these engines can be leveraged for self service. The self service portal can utilize existing off the shelf tools or will need to be a custom web application in case the capabilities provided by the tools are not enough. Off the Shelf tools stardog available such as Metaphacts, Tom Sawyer Graph Database browser or Stardog Explorer can be leveraged to get up and running. Custom self service user portal can also be explored. This can be developed using React.js integrated with graph visualization & exploration toolkits (provided for example by Cambridge Intelligence or Graphistry) enabling search functionalities.

Operational Aspects of a Data Fabric

The operational aspects of a data fabric are important because they ensure that the data fabric is maintained properly and it continues to function effectively. Without proper operation and management, a data fabric will become outdated, unreliable, or difficult to use thus reducing its value and effectiveness. By regularly maintaining and updating the data fabric, organizations can ensure that it continues to support their goals and enables them to make data-driven decisions.

Maintaining a data fabric will involve:

- Using automated pipelines to regularly update and maintain the Data Fabric layers including the Data Catalog and Semantic Data Layer
- Updating the Semantic Data Model based on updates to business and data assets
- Curating the Knowledge Graph with a dedicated team / Knowledge Stewards to ensure that the data fabric is accurate, up-to-date and relevant for the organization's needs
- Working closely with stakeholders and users to ensure that the data fabric is meeting their needs and supporting their goals for collaboration.
- Support, training, coordination with business users apart from gathering & incorporating feedback
- Onboarding new data assets and systems into the Data Fabric

Summary

Data Fabric is an important distributed data architecture as it brings together the knowledge about data assets across the enterprise. As it constantly keeps the knowledge of data updated for enterprise wide applications, a Data Fabric also provides flexibility to organizations to evolve their data applications for upcoming business needs. A Data Fabric also enables data democratization through its self service user portal which can help Enterprises realize more value from their data assets.

Given this, a Data Fabric can add value to existing investments as it doesn't intend to make existing data platforms redundant. Therefore there is considerable interest from Enterprises to see if it can make an actual difference in their landscapes and be the game changer it promises to be. Software vendors focussed on Data Governance & Data Management (such as Informatica who has tools such as Informatica Data Catalog & Informatica Claire Engine which includes Knowledge Graphs,AI,ML etc) and Knowledge Graphs (Cambridge Semantics, Stardog etc) have a head start on Data Fabric solutions as they are looking to combine these capabilities and add other elements which can be used to implement the Data Fabric on top of enterprise data assets. Gartner lists data fabric as a Top strategic technology trend for 2022 and predicts that by 2024, 25% of data management vendors will provide a complete framework for data fabric. It'll indeed be interesting to see who takes the lead in providing an end to end framework / platform to implement the Data Fabric architecture.

In this document, we have delved deeper into the Data Fabric architecture. This understanding can be leveraged further by GlobalLogic colleagues to initiate conversations on improving client data landscapes and to check for applicability of the Data Fabric architecture & approach discussed in this document in their respective engagements.

References / Further Reading

- https://www.forrester.com/report/the-forrester-wave-enterprise-data-fabric-q2-2020/RES157288
- https://www.gartner.com/smarterwithgartner/data-fabric-architecture-is-key-to-modernizing-data-management-and-integration
- https://www.talend.com/resources/what-is-data-fabric/
- https://www.stardog.com/blog/5-steps-to-building-a-data-fabric/
- https://www.qlik.com/us/data-management/data-fabric
- https://www.ibm.com/blogs/journey-to-ai/2022/03/what-is-a-data-fabric-architecture/
- https://www.databricks.com/blog/2022/06/17/using-a-knowledge-graph-to-power-a-semantic-data-layer-for-databricks.html
- https://blog.google/products/search/about-knowledge-graph-and-knowledge-panels/
- https://docs.aws.amazon.com/neptune/latest/userguide/notebooks-visualization-tools.html
- https://atlan.com/types-of-metadata/
- https://www.oracle.com/in/big-data/data-catalog/what-is-a-data-catalog/#make-use
- https://www.informatica.com/blogs/data-fabric-vs-data-mesh-3-key-differences-how-they-help-and-proven-benefits.html
- https://www.gartner.com/en/documents/4006916
- https://www.gartner.com/en/documents/4008708