## Big Data Application Provider

Big Data Application Provider is the role of executing a generic data life cycle including data collection from various sources, multiple data transformations being implemented using both traditional and new technologies, and diverse data usage.

The Big Data Application Provider activities would typically be specific to the application and therefore are not candidates for standardization. However, the metadata and the policies defined and exchanged between the application’s sub-blocks could be standardized.

As the data propagates through the ecosystem, it is being processed and transformed in different ways in order to extract the value from the information. Each activity within the Big Data Application Provider can be implemented by independent stakeholders and deployed as stand-alone services.

Each activity can utilize a separate specialized Big Data Framework best suited for its requirements, and can have its own privacy and other policy considerations.

In its role, Big Data Application Provider typically executes the manipulations of the data lifecycle of a specific vertical system to meet the requirements or instructions established by the System Orchestrator.

The Big Data Application Provider activities include:

* Collection:
  + Obtains connection to Data Provider APIs to collect into local system, or to access dynamically when requested. At the initial collection stage, sets of data (e.g., data records) of similar structure are collected (and combined) resulting in uniform security considerations, policies, etc. Initial metadata is created (e.g., subjects with keys are identified) to facilitate subsequent aggregation or lookup method(s).
* Curation:
  + Prepares the data through cleansing, outlier removal, and standardization for the ingestion and storage processes.
  + Aggregates data from different data providers with easily correlated metadata (e.g., identical keys) into a single data set. As a result, the information about each object is enriched or the number of objects in the collection grows. Security considerations and policies concerning the resultant data set are typically similar to the original data.
  + Matches data from different data providers with disparate metadata (e.g., keys) into a single data set. As a result, the information about each object is enriched. The security considerations and policies concerning the resultant data set may differ from the original policies.
  + Optimizes data by determining the appropriate data manipulations and indexing to optimize subsequent transformation processes.
* Analysis:
  + Implements the techniques to extract knowledge from the data based on the requirements of the data scientist, who has specified the algorithms to process the data to produce new insights that will address the technical goal.
* Visualization
  + Ensures relevant visualization tools are integrated into data life cycle system.
  + Formats and presents data in such a way as to optimally communicate meaning and knowledge.
  + Develops appropriate statistical charts and diagrams to reflect analysis results.
* Access
  + Identifies and stores data in persistent repositories for use, sharing, and re-use.
  + Ensures descriptive, administrative, and preservation metadata and metadata schemes are developed and used.
  + Conducts appropriate curation activities to account for changes in storage media, file formats, etc., over time in order to ensure data access for specified periods.

While many of these tasks have traditionally existed in data processing systems, the scale, velocity and variety present in big data systems radically changes their implementation. The algorithms and mechanisms need to be re-written and optimized to create applications that are responsive and can grow to handle ever growing data collections.

The Big Data Application Provider will expose a collection of interfaces (or services) for consuming the results of the data processing performed by the big data system. These interfaces can be called and composed by 3rd party applications that have the permissions to consume the data. While the specifics of this interface may be application-specific, commonality will exist in a number of areas, including:

Data Transfer: Facilitates secure transfer of data between different repositories and/or between the Big Data Application Provider and the Big Data Framework Provider RA blocks.

Identity Management and Authorization: Individual vertical Big Data Application Providers may implement their own schemes for usage of their services. Identity management enables Big Data Application Providers to implement charging schemes, provide levels of differentiated service and protect confidential services from unauthorized access.

Discovery: Data Consumers require a directory that defines the services that a Big Data Application Provider can support.

Code execution services: A Big Data Application Provider may allow data consumers to push analytics in the form of a code to execute on the big data system. The usage services will define the precise form that these requests support, for example the software languages that can be used, the constraints (e.g., execution time) on the code that is provided to the service, and how the results will be delivered to the Data Consumer.

Charging: Big Data Application Providers may implement charging schemes to generate revenue from Data Consumers. The usage services will enable users to discover the amounts they are being charged, monitor usage and make payments.