# 

## Big Data Framework Provider

The Big Data Framework Provider ensures a computing fabric (e.g., system hardware, network, storage, virtualization, computing platform) to enable execution of certain transformation applications while protecting the privacy and integrity of data. The computing fabric facilitates a mix-and-match of traditional and state-of-art computing features from software, platforms, and infrastructures based on the needs of each application. The computing fabric consists of the following:

* Processing Frameworks
* Platforms
* Infrastructures
* Physical and Virtual Resources

The Framework Provider consists of one or more instances of components typically hierarchically organized based on the IT Value Chain as shown in the RA diagram. There is no requirement that all instances at a given level in the hierarchy be of the same technology and in fact most big data implementations are hybrids combining multiple technology approaches in order to provide flexibility or meet the complete range of requirements which are driven from the Application Provider Framework. The four component areas that make up the Framework provider are Resources, Infrastructures, Platforms, and Processing. The following paragraphs briefly describe each of these component areas and more detailed discussions can be found in other NIST Big Data publications including the Technology Roadmap document and the Definitions and Taxonmy document.

### Physical and Virtual Resources

This component area represents the raw storage and computing power that will be leveraged by the other framework areas. While at some level all resources have a physical representation the Big Data Framework components may be deployed directly on physical resources or on virtual resources (e.g. Amazon Web Services). Frequently Big Data Framework deployments will actually employ a combination of physical and virtual resources. Virtual resources may be used to deploy either components that require additional elasticity (e.g. can change size dynamically) or that have very specific high availability requirements that are easier to meet using virtual resource managers. Physical, resources in turn are frequently used to deploy horizontally scalable components that will be duplicated across a large number of physical nodes. Environmental resources such as Power and HVAC are also part of this component layer since they are in fact finite resources that must be managed by the overall framework.

### Infrastructures

The infrastructure component defines how the physical and virtual resources and organized and connected. The three key sub elements of this are the cluster/computing, storage, and networking infrastructures. The cluster/computing infrastructure logical distribution may vary from a dense grid of physical commodity machines in a rack, to a set of virtual machines running on a cloud service provider, to a loosly coupled set of machines distributed around the globe providing access to un-used computing resources (e.g. SETI@Home, etc.). This infrastructure also frequently includes the underlying operating systems and associated services used to interconnect the cluster resources. The storage infrastructure may be likewise be organized as anything from isolated local disks to Storage Area Networks (SANs) or global distributed object stores as shown in Figure 3.3.2-1 along with their associated technology readiness.



The cluster/compute and storage components are in turn interconnected via the network infrastructure. The volume and velocity of Big Data often is a driving factor in the implementation of this component of the architecture. For example, if the implementation requires frequent transfers of large multi-gigabyte files between cluster nodes then high speed and low latency links are required. Depending on the availability requirements redundant and fault tolerant links may be required. Other aspects of the network infrastructure include name resolution (e.g. DNS) and encryption along with firewalls and other perimeter access control capabilities. Finally, this layer may also include automated deployment/provisioning capabilities/agents and infrastructure wide monitoring agents that are leveraged by the management elements to implement a specific model.

### Platforms

The platform element consists of the logical data organization and distribution combined with the associated access APIs or methods. As shown in Figure 3.3-1 below that organization may range from simple delimited flat files to fully distributed relational or columnar data stores. Accordingly, the access methods may range from POSIX style file access APIs to fully blown query languages such as SQL or SPARQL. Typically, most big data framework implementations support basic file system style storage and one or more indexed storage approaches. It should be noted that while many big data implementations will implement this logical organization distributed across a cluster of computing resources there is no requirement that it be so.

The platform may also include data registry and meta data services along with semantic data descriptions such as formal ontologies or taxonomies.

### Processing Frameworks

Processing frameworks define how the computation/processing of the data is organized. Typically, processing frameworks are generally divided along the lines of batch oriented and streaming oriented. However, reality is that depending on the specific data organization and platform most frameworks in fact span a range from high latency to near real time processing. Most, big data implementations tend to implement multiple frameworks depending on the nature of the application or application processing step which needs to be implemented as show below in figure 3.3.4-1.



The green shading above illustrates the general sensitivity of that phase of the processing to latency which is defined as the time from when a request or piece of data arrives at a system until its processing/delivery is complete. For Big Data the ingestion may or may not require near real time performance to keep up with the data flow, and some types of analytics (specifically those categorized as Complex Event Processing) may or may not require that type of processing. At the far right generally sits the data consumer depending upon the use case and application batch responses (e.g. a nightly report is emailed) may be sufficient. In other, cases the user may be willing to wait minutes for the results of a query to be returned, or they may need immediate alerting when critical information arrives at the system. Another way to look at this is that batch analytics tend to better support long term strategic decision making where the overall view or direction is not going to be affected by a recent change to some portion of the underlying data. Streaming analytics are better suited for tactical decision making where new data needs to be acted upon immediately. A primary use case for this would be electronic trading on stock exchanges where the window to act of a given piece of data can be measured in microseconds.

Typically big data discussions focus around batch and streaming frameworks for analytics, however retrieval frameworks which provide interactive access to big data are becoming a more prevalent. Of course the lines between these categories are not solid or distinct with some frameworks providing aspects of each element.