**A “Small” Definition of Big Data**

The term ‘big data’ seems to be popping up everywhere these days. And there seems to be as many uses of this term as there are contexts in which you find it: ‘big data’ is often used to refer to any dataset that is difficult to manage using traditional database systems; it is also used as a catch-all term for any collection of data that is too large to process on a single server; yet others use the term to simply mean “a lot of data”; sometimes it turns out it doesn’t even have to be large. So what exactly is big data?

A precise specification of ‘big’ is elusive. What is considered big for one organization may be small for another. What is large-scale today will likely seem small-scale in the near future; petabyte is the new terabyte. Thus, size alone cannot specify big data. The complexity of the data is an important factor that must also be considered.

Most now agree with the characterization of big data using the 3 V’s coined by Doug Laney of Gartner:

· Volume: This refers to the vast amounts of data that is generated every second/minute/hour/day in our digitized world.

· Velocity: This refers to the speed at which data is being generated and the pace at which data moves from one point to the next.

· Variety: This refers to the ever-increasing different forms that data can come in, e.g., text, images, voice, and geospatial.

A fourth V is now also sometimes added:

· Veracity: This refers to the quality of the data, which can vary greatly.

There are many other V's that gets added to these depending on the context. For our specialization, we will add:

· Valence: This refers to how big data can bond with each other, forming connections between otherwise disparate datasets.

The above V’s are the dimensions that characterize big data, and also embody its challenges: We have huge amounts of data, in different formats and varying quality, that must be processed quickly.

It is important to note that the goal of processing big data is to gain insight to support decision-making. It is not sufficient to just be able to capture and store the data. The point of collecting and processing volumes of complex data is to understand trends, uncover hidden patterns, detect anomalies, etc. so that you have a better understanding of the problem being analyzed and can make more informed, data-driven decisions. In fact, many consider value as the sixth V of big data:

· Value: Processing big data must bring about value from insights gained.

To address the challenges of big data, innovative technologies are needed. Parallel, distributed computing paradigms, scalable machine learning algorithms, and real-time querying are key to analysis of big data. Distributed file systems, computing clusters, cloud computing, and data stores supporting data variety and agility are also necessary to provide the infrastructure for processing of big data. [Workflows](http://words.sdsc.edu/words-data-science/workflows) provide an intuitive, reusable, scalable and reproducible way to process big data to gain verifiable value from it in and enable application of same methods to different datasets.

With all the data generated from social media, smart sensors, satellites, surveillance cameras, the Internet, and countless other devices, big data is all around us. The endeavor to make sense out of that data brings about exciting opportunities indeed!

Source: <http://words.sdsc.edu/words-data-science/big-data>

**Workflows**

As the Internet of Things and other data acquisition and generation technologies advance, data being generated is growing at an exponential rate at all scales in many online and scientific platforms. This mostly unstructured and variable data growing and moving between different applications dynamically in vast quantities is often referred to as "Big Data". The amount of potentially valuable information buried in Big Data is of interest to many data science applications ranging from natural sciences to marketing research. In order to analyze and digest such heterogeneous data, challenges for integration and distributed analysis include: scalable data preparation and analysis techniques; new and distributed programming paradigms; repeatable and verifiable process development; and innovative hardware and software systems that can serve applications based on their needs.

An important aspect of Big Data applications is the variability of technical needs and steps based on applications being developed. These applications typically involving data ingestion, preparation (e.g., extract, transform, and load), integration, analysis, visualization and dissemination are referred to as Data Science Workflows. A data science workflow development is the process of combining data and processes into a configurable, structured set of steps that implement automated computational solutions of an application with capabilities including provenance management, execution management and reporting tools, integration of distributed computation and data management technologies, ability to ingest local and remote scripts, and sensor management and data streaming interfaces. Each data science workflow has a set of technological challenges that can potentially employ a number of Big Data tools and middleware. Rapid programmability of applications on a use case basis requires workflow management tools that can interface to and facilitate integration of other tools. New programming techniques are needed for building effective and scalable solutions span across the data science workflows. Flexibility of workflow systems to combine tools and data together makes it an ideal choice for development of data science applications involving common Big Data programming patterns.

Big Data workflows have been an active research area since the introduction of scientific workflows. After the development and general adoption of MapReduce as a Big Data programming pattern, a number of workflow systems were built or extended to enable programmability of MapReduce applications including Oozie, Nova, Azkaban and Cascading. The Kepler Workflow Environment also provide a distributed data-parallel (DDP) programming module on MapReduce and other BigData programing patterns on top of well-known Hadoop and Stratosphere engines to build and execute big data workflows. The actor-oriented approach of Kepler provides flexibility and improves application programmibility due to: (i) its heterogeneous nature in which Big Data programming patterns are placed as part of other workflow tasks; (ii) its visual programming approach that does not require scripting of Big Data patterns; (iii) its adaptability for execution of data parallel applications on different execution engines.