Data Analysis Use Cases

August 1, 2019

Version 1.1

These use cases describe how scientists analyze large amounts of scientific data. The data is typically large in volume (larger than one would use on a personal or business computer), but is organized and stored in different ways for different kinds of research. Data might be generated by a single source (a large simulation, for example) or it might come from many sources (observational results from many different instruments or different research teams). The methods of analysis also vary from field to field and problem to problem.

DA-01: Discover data analysis resources and documentation

A **researcher** needs to find information about the data analysis resources, applications, and services available to the community. This is a pure discovery process; no data manipulation or analytics are performed.

In most cases, the researcher wants to experience it as follows.

1. First, the researcher visits the community website and navigates to a section on "Data Analysis." (Specific wording may vary.)
2. Then, the website displays a categorized list of resources, applications, and services available in the community.
3. If the researcher selects an item from the list, the researcher can access related information such as user guides, allocation instructions, and current availability data.

It will always be like this unless Step 1 is replaced by the researcher entering a search term similar to "data analysis" in the search field of the community website. In that case, the experience should continue unchanged with Step 2.

We'll take any solution, as long as the following are true.

1. The researcher doesn't need to authenticate to access this information.
2. The researcher doesn't need an active or prior allocation to access this information.
3. In Step 2, a glossary of data analysis terms is available.
4. In Step 2, the list of available resources includes references to community experts that can assist with data analysis, training workshops, and reference materials.
5. In Steps 2 and 3, the information combines details from many sources, including the community website, community information services, service provider websites and documentation.
6. As resources come and go and details of each resource change, the information about the application and resources available via the community is continuously updated.
7. The community website provides sufficient information to understand the available resources, apply for access, and make basic use of the resources.
8. Whenever information is gathered from community partners or independent service providers, the community website should clearly present contextual information, including: how access, use, or policies differ from ordinary community expectations; specific requirements for obtaining access; and especially any related costs.

DA-02: Prepare data for analysis

A **researcher** needs to prepare a set of data for large-scale analysis. The researcher must gather the required data from its source(s) and then translate, reformat, organize, or annotate the data as needed for data mining, modeling, or other analysis activities. We assume the researcher is a member of a project with an allocation on a computational resource suitable for the necessary data analysis.

In most cases, the researcher wants to experience it as follows.

1. First, the researcher logs into the resource where the analysis will take place.
2. Then, the researcher fetches data for processing from one or more sources.

3. Then, the researcher uses commands or applications to clean, filter, and organize the data for successful mining, modeling, and/or analysis, to solve or avoid problems in the data, and to present the data to the modeling or analysis system in an optimal way. The researcher may also attach appropriate metadata and provenance information to the resulting data.
4. Finally, the researcher may create an archival copy of the prepared data, metadata, and provenance information on a long-term storage resource.

It will always be like this except when the researcher uses a data transfer service to transfer data to the resource. In that case, Step 1 is delayed until before Step 3, and Step 2 is replaced by the researcher using the data transfer service to transfer data to the resource.

We'll accept any solution as long as the following are true.

1. In Steps 2 and 4, the resource must allow the researcher to transfer data from (and to) remote systems, ideally using the community's standard data transfer mechanism (see use case CAN-02).
2. In Step 3, the researcher may submit jobs to the resource's scheduler to prepare the data.
3. In Step 3, the researcher may specify general workflows to prepare the data. Each workflow is a sequence or pattern of jobs. The process may be iterative with breaks for human review and adjustments.
4. Data preparation applications--including those submitted as jobs--may collect data from a wide range of Internet sites. (A.k.a. "Internet scraping.")
5. Software that support data preparation and community standards for data exchange and reuse are pre-installed and tested.
6. The pre-installed software enables metadata creation and management.
7. The pre-installed software includes libraries for reading standard file formats such as NetCDF, HDF5, and others.
8. Documentation for pre-installed software is accurate and accessible, including clear examples.
9. Pre-installed software includes both current versions and older versions where backward compatibility isn't guaranteed.
10. The researcher's data and metadata may be in multiple formats including flat files, relational RDBMS, Hadoop data file system, text, images, and video.

DA-03: Analyze data from research instruments

A **researcher** needs to analyze data collected from research instruments. Examples of instruments include: radio, microwave and optical telescopes; genome sequencers; satellites; electron microscopes; and sensor networks. We assume the researcher is a member of a project with an allocation on a computational resource suitable for the required data analysis.

In most cases, the research wants to experience it as follows.

1. First, the researcher assembles and prepares the data as described in use case DA-02.
2. Then, the researcher submits jobs to analyze the data, specifying a location to store the results.
3. Then, the researcher inspects the analysis results and uses them to complete the goals of the allocation project.

4. Finally, the researcher optionally transfers the analysis results to another system (e.g., for use in later stages of the research) and/or creates an archival copy on a long-term storage resource.

It will always be like this except when the researcher also needs to continuously ingest new data from the instrument and analyze it as it becomes available. In that case, the researcher first sets up automatic data ingestion as described in use case DM-03, and Step 1 includes extracting the data to be analyzed from the resulting data collection.

We'll accept any solution as long as the following are true.

1. In Step 1, the data may exist as files or as database tables, and the data may be transferred from other resources (e.g., the facility where the instrument is located).
2. In Steps 1 and 4, the resource allows the researcher to transfer data from (and to) remote systems, ideally using the community's standard data transfer mechanism (see use case CAN-02).
3. Software that supports data mining and analysis and that satisfies community standards for data exchange and reuse is pre-installed and tested.
4. Documentation for pre-installed tools and applications is accurate and accessible, including clear examples.
5. Pre-installed tools and applications include both current versions and older versions where backward compatibility isn't guaranteed.

DA-04: Analyze data generated by a simulation

A **researcher** needs to analyze data that was generated by a simulation. We assume the researcher is a member of a project with an allocation on a computational resource suitable for the required data analysis.

In most cases, the researcher wants to experience it as follows.

1. First, the researcher gathers and prepares the data as described in use case DA-02.
2. Then, the researcher submits jobs to analyze the data.
3. Then, the researcher inspects the analysis results and uses them to complete the goals of the allocation project.
4. Finally, the researcher optionally transfers the analysis results to another system (e.g., for use in later stages of the research) and/or creates an archival copy on a long-term storage resource.

It will always be like this except when the researcher needs to analyze the data while the simulation is running (a.k.a., *in situ* analysis). In this case, Step 1 is replaced by the researcher submitting jobs to run the simulation, specifying a location where the simulation data will be gathered for analysis. (See use case HPC-01 or other HPC use cases.)

We'll accept any solution as long as the following are true.

1. Software that supports dynamic workflow analysis and manipulation and that satisfies community standards for data exchange and reuse is pre-installed and tested.
2. Documentation for pre-installed software is current, accurate, and accessible, including clear examples.

3. Pre-installed software includes both current versions and older versions where backward compatibility isn't guaranteed.

DA-05: Steer a large computation while it runs

A **researcher** needs to "steer" a large computation while it is running to focus on important or interesting parameters. The researcher inspects or visualizes the output of the computation while it is running and adjusts the parameters used for the remainder of the computation. We assume the researcher is a member of a project with an allocation on a resource appropriate for the computation.

In most cases, the researcher wants to experience it as follows.

1. First, the researcher logs into the resource where the computation will run.
2. Then, the researcher modifies the computation's code so it can detect and respond to modified parameters while it runs, enabling computational steering.
3. Then, the researcher creates one or more parameter files. These files include parameters that cause the program to periodically produce intermediate results or other diagnostics that the researcher can inspect.
4. Then, the researcher submits the computation job(s).
5. While the computation's job(s) execute, the researcher inspects or visualizes the intermediate results or diagnostics.
6. As determined by the intermediate results or diagnostics, the researcher modifies the parameter files, and the computation's job(s) detect and respond to the changes.
7. When the computation completes, the researcher inspects the results and uses them to complete the goals of the allocation project.
8. Finally, the researcher optionally transfers the results to another system (e.g., for use in later stages of the research) and/or creates an archival copy on a long-term storage resource.

We'll accept any solution as long as the following are true.

1. Software that supports dynamic workflow analysis and manipulation and that satisfies community standards for data exchange and reuse is pre-installed and tested.
2. Documentation for pre-installed software is current, accurate, and accessible, including clear examples.
3. Pre-installed software includes both current versions and older versions where backward compatibility isn't guaranteed.

History

	Version	Date	Changes	Author
Entire document	0.1	9/18/2012	First Version submitted to A&D	Data Analytics Team
Entire document	0.2	4/26/2013	Iterating Architects feedback	Data Analytics Team
DA-01, DA-02	0.3/1.0	3/16/2013	Significant revisions to first two use cases based on feedback	Data Analytics Team
Entire Document	1.1	8/1/2019	Reformatted to the XSEDE-2 use case format; removed unnecessary XSEDE terminology	L. Liming