

## Chapter 11

# Visualization of Large-Scale Distributed Data

**Jason Leigh**

*University of Illinois at Chicago, USA*

**Venkatram Vishwanath**

*University of Illinois at Chicago, USA & Argonne  
National Laboratory, USA*

**Andrew Johnson**

*University of Illinois at Chicago, USA*

**Tom Peterka**

*Argonne National Laboratory, USA*

**Luc Renambot**

*University of Illinois at Chicago, USA*

**Nicholas Schwarz**

*Northwestern University, USA*

### ABSTRACT

*An effective visualization is best achieved through the creation of a proper representation of data and the interactive manipulation and querying of the visualization. Large-scale data visualization is particularly challenging because the size of the data is several orders of magnitude larger than what can be managed on an average desktop computer. Large-scale data visualization therefore requires the use of distributed computing. By leveraging the widespread expansion of the Internet and other national and international high-speed network infrastructure such as the National LambdaRail, Internet-2, and the Global Lambda Integrated Facility, data and service providers began to migrate toward a model of widespread distribution of resources. This chapter introduces different instantiations of the visualization pipeline and the historic motivation for their creation. The authors examine individual components of the pipeline in detail to understand the technical challenges that must be solved in order to ensure continued scalability. They discuss distributed data management issues that are specifically relevant to large-scale visualization. They also introduce key data rendering techniques and explain through case studies approaches for scaling them by leveraging distributed computing. Lastly they describe advanced display technologies that are now considered the “lenses” for examining large-scale data.*

DOI: 10.4018/978-1-61520-971-2.ch011

## INTRODUCTION

The primary goal of visualization is insight. An effective visualization is best achieved through the creation of a proper representation of data and the interactive manipulation and querying of the visualization. Large-scale data visualization is particularly challenging because the size of the data is several orders of magnitude larger than what can be managed on an average desktop computer. Data sizes range from terabytes to petabytes (and soon exabytes) rather than a few megabytes to gigabytes. Large-scale data can also be of much greater dimensionality, and there is often a need to correlate it with other types of similarly large and complex data. Furthermore the need to query data at the level of individual data samples is superseded by the need to search for larger trends in the data. Lastly, while interactive manipulation of a derived visualization is important, it is much more difficult to achieve because each new visualization requires either re-traversing the entire dataset, or compromising by only viewing a small subset of the whole. Large-scale data visualization therefore requires the use of distributed computing.

The individual components of a data visualization pipeline can be abstracted as:

Data Retrieval → Filter / Mine → Render → Display

The degree to which these individual components are distributed or colocated has historically been driven by the cost to deploy and maintain infrastructure and services. Early in the history of scientific computing, networking bandwidth was expensive and therefore scarce. Consequently early visualization pipelines tended to minimize the movement of data over networks in favor of collocating data storage with data processing. However, as the amount and variety of data continued to grow at an exponential pace, it became too costly to maintain full replicas of the data

for each individual that needed to use it. Instead, by leveraging the widespread expansion of the Internet and other national and international high-speed network infrastructure such as the National LambdaRail<sup>1</sup>, Internet-2<sup>2</sup>, and the Global Lambda Integrated Facility<sup>3</sup>, data and service providers began to migrate toward a model of widespread distribution of resources.

In this chapter we will first introduce the various instantiations of the visualization pipeline and the historic motivation for their creation. We will then examine individual components of the pipeline in detail to understand the technical challenges that must be solved in order to ensure continued scalability. We will discuss distributed data management issues that are specifically relevant to large-scale visualization. We will also introduce key data rendering techniques and explain through case studies approaches for scaling them by leveraging distributed computing. Lastly we will describe advanced display technologies that are now considered the “lenses” for examining large-scale data.

## THE LARGE-SCALE DATA VISUALIZATION PIPELINE

### Collocated Data, Filtering, Rendering and Display Resources

Most visualization software packages have a pipeline architecture where raw data comes in at one end of the pipeline from disk or the network, moves through a sequence of filters that process the data on the CPU and generate computer graphics primitives (e.g. lines, triangles, splats, pixels) which are rendered on the GPU, and displayed on a monitor at the other end of the pipeline. Some filters deal with accessing data or generating data. Other filters convert data from one form to another. Finally there are filters that deal with the creation of computer graphics. Each filter has an explicit input and output format allowing

31 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:  
[www.igi-global.com/chapter/visualization-large-scale-distributed-data/62830](http://www.igi-global.com/chapter/visualization-large-scale-distributed-data/62830)

## Related Content

---

### Performance Evaluation of Chi-Square and Normal Distributions of Mesh Clients for WMNs Considering Five Router Replacement Methods

Admir Barolli, Kevin Bylykbashi, Ermioni Qafzezi, Shinji Sakamoto and Leonard Barolli (2022). *International Journal of Distributed Systems and Technologies* (pp. 1-14).

[www.irma-international.org/article/performance-evaluation-of-chi-square-and-normal-distributions-of-mesh-clients-for-wmns-considering-five-router-replacement-methods/296247](http://www.irma-international.org/article/performance-evaluation-of-chi-square-and-normal-distributions-of-mesh-clients-for-wmns-considering-five-router-replacement-methods/296247)

### Service Level Provisioning for Cloud-Based Applications Service Level Provisioning for Cloud-Based Applications

Valeria Cardellini, Emiliano Casalicchio and Luca Silvestri (2012). *Grid and Cloud Computing: Concepts, Methodologies, Tools and Applications* (pp. 1479-1500).

[www.irma-international.org/chapter/service-level-provisioning-cloud-based/64549](http://www.irma-international.org/chapter/service-level-provisioning-cloud-based/64549)

### Real-Time Attributes in Operating Systems

Tommaso Cucinotta and Spyridon V. Gougouitis (2012). *Achieving Real-Time in Distributed Computing: From Grids to Clouds* (pp. 275-287).

[www.irma-international.org/chapter/real-time-attributes-operating-systems/55253](http://www.irma-international.org/chapter/real-time-attributes-operating-systems/55253)

### Load Balancing in Peer-to-Peer Systems

Haiying Shen (2010). *Handbook of Research on Scalable Computing Technologies* (pp. 163-190).

[www.irma-international.org/chapter/load-balancing-peer-peer-systems/36408](http://www.irma-international.org/chapter/load-balancing-peer-peer-systems/36408)

### Guaranteeing Correctness for Collaboration on Documents Using an Optimal Locking Protocol

Stijn Dekeyser (2011). *International Journal of Distributed Systems and Technologies* (pp. 17-29).

[www.irma-international.org/article/guaranteeing-correctness-collaboration-documents-using/58631](http://www.irma-international.org/article/guaranteeing-correctness-collaboration-documents-using/58631)