Preface

We are generating data more than ever. The ubiquity of the Internet has dramatically changed the size, speed, and nature of the generated data. Almost every human became a data generator and every business became a digital business. As a result, we are witnessing a data explosion. In the past few years, several technologies have contributed to this data explosion including mobile computing, Web 2.0, social media, social network, cloud computing and Software-as-a-Service (SaaS). In the future, it is expected that the Internet of Things will further amplify this challenge. In particular, several *things* would be able to get connected to the Internet, and thus there will be lots of data passed from users to devices, to servers, and back. Hence, in addition to the billions of people who are currently using the Internet and daily producing a lot of data, watches, cars, fridges, toaster, and many other devices will be online and continuously generating data as well. It is quite expected that in the near future, our toasters will be able to recommend types of bread based on suggested information from our friends on the social networks.

With the recent emerging wave of technologies and applications, the world has become more connected than ever. Graph is a popular neat data structure which is used to model the data as an arbitrary set of objects (vertices) connected by various kinds of relationships (edges). With the tremendous increase in the size of the graph-structured data, large-scale graph-processing systems have been crucially on demand and attracted a lot of interest. This book is intended to take you to a journey with **Apache Giraph**, a popular distributed graph-processing platform, which is designed to bring the power of big data processing to graph data that would be too large to fit on a single machine. We describe the fundamental abstractions of the system and its programming models and describe various techniques for using the system to process graph data at scale. The book is designed as a self-study step-by-step guide for any reader with an interest in large-scale graph processing. All the source codes presented in the book are available for download from the associated Github repository of the book.

Organization of the Book

Chapter 1 starts with a general background of the big data phenomena. We then introduce the big graph problem, its applications, and how it differs from the traditional challenges of the big data problem and motivates the need for domain-specific systems that are designed to tackle the large-scale graph-processing problem. We then introduce the Apache Giraph system, its abstraction, programming model, and design architecture to set the stage for the reader and provide him with the fundamental information which is required to smoothly follow the other chapters of the book.

Chapter 2 takes Giraph as a platform. Keeping in view that Giraph uses Hadoop as its underlying execution engine, we explain how to set up Hadoop in different modes, how to monitor it, and how to run Giraph on top of it using its binaries or source code. We then move to explaining how to use Giraph. We start by running an example job in different Hadoop modes and then approach more advanced topics such as monitoring Giraph application life cycle and monitoring Giraph jobs using different methods. Giraph is a very flexible platform and its behavior can be tuned in many ways. We explain the different methods of configuring Giraph and end the chapter by giving a detailed description of setting up a Giraph project in Eclipse and IntelliJ IDE.

Chapter 3 provides an introduction to Giraph programming. We introduce the basic Giraph graph model and explain how to write a Giraph program using the vertex similarity algorithm as a use case. We explain three different ways of writing the driver program and their pros and cons. For loading data into Giraph, it comes packaged with numerous input formats for reading different formats of data. We describe each of the formats with examples and end the chapter with the description of Giraph output formats.

Chapter 4 discusses the implementation of some popular graph algorithms including PageRank, connected components, shortest paths, and triangle closing. In each of these algorithms, we give an introductory description and show some of its possible applications. Then using a sample data graph, we show how the algorithm works. Finally, we describe the implementation details of the algorithm in Giraph.

Chapter 5 sheds light on the advanced Giraph programming. We start by discussing common Giraph algorithmic optimizations and how those optimizations may improve the performance and flexibility of the algorithms implemented in Chap. 4. We explain different graph optimizations to enable users to implement complex graph algorithms. Then, we discuss a set of tunable Giraph configurations that controls Giraph's utilization of the underlying resources. We also discuss how to change Giraph's default partitioning algorithm and how to write a custom graph input and output format. We then talk about common Giraph runtime errors and finalize the chapter with information on Giraph's failure recovery.

Recently, several systems have been introduced to tackle the challenge of large-scale graph processing. In Chap. 6, we highlight two of these systems, GraphX and GraphLab. We describe their program abstractions and their programming models. We also highlight the main commonalities and differences between these systems and Apache Giraph.

Target Audience

We hope this book serves as a useful reference for students, researchers, and practitioners in the domain of large-scale graph processing.

To Students: We hope that the book provides you an enjoyable introduction to the field of large-scale graph processing. We have attempted to properly describe the state of the art and present the technical challenges in depth. The book will provide you with a comprehensive introduction and hands-on experience to tackling large-scale graph-processing problem using the Apache Giraph systems.

To Researchers: The material of this book will provide you with a thorough coverage for the emerging and ongoing advancements on big graph-processing systems. You also can use this book as a starting point to tackle your next research challenge in the domain of large-scale graph processing.

To Practitioners: You will find this book a very useful step-by-step guide with several code examples, with source codes available in the Github repository of the book, and programming optimization techniques so that you can immediately put the gained knowledge from this book into practice due to the open-source availability of Apache Giraph system.

Sydney, Australia — Sherif Sakr
Aalborg, Denmark — Faisal Moeen Orakzai
Thuwal, Saudi Arabia — Ibrahim Abdelaziz
Thuwal, Saudi Arabia — Zuhair Khayyat

http://www.springer.com/978-3-319-47430-4

Large-Scale Graph Processing Using Apache Giraph

Sakr, S.; Orakzai, F.M.; Abdelaziz, I.; Khayyat, Z.

2016, XXV, 197 p. 102 illus., 87 illus. in color.,
Hardcover
ISBN: 978-3-319-47430-4