Databricks Lakehouse Platform Cookbook

100+ recipes for building a scalable and secure Databricks Lakehouse

Dr. Alan L. Dennis

www.bpbonline.com

First published: 2024

Published by BPB Online
WeWork
119 Marylebone Road
London NW1 5PU

UK | UAE | INDIA | SINGAPORE

ISBN 978-93-55519-566

www.bpbonline.com

Dedicated to

My loving and supportive wife,

Kim

Foreword

There is no denying that data is the lifeblood of industry. Everyone understands that businesses that harness their data well, will thrive while the ones that do not, will fall by the wayside. Choosing the correct platform for your data estate is, perhaps, the most critical decision a business can make. The second most important factor is, of course, to hire the right team to build on this chosen platform.

2023 has shown us that GenAI is the future and I am convinced that there is only one data platform that is natively suited for the demands that GenAI will impose on the data estates of the future: the Lakehouse built on Databricks. Over the last 10 years, Databricks has transformed itself from a niche Spark-focused shop to a visionary organization building a holistic data platform that can support Analytics, Data Integration (a fancier term for ETL or ELT) and AI/ML: including the now white-hot GenAI use cases. While a lot of competing data platforms and data clouds make similar claims, there is only one platform, that is, Databricks: that has been doing cloud-native, multi-language data processing at scale: for 10 years now.

I consider myself fortunate that I discovered and fell in love with Apache Spark in 2014 and then got a chance to join Databricks in 2019. I am truly grateful that I got to pick the brains of some of the smartest minds in the universe during that time. The author of this book, Dr. Alan Dennis, is one such individual and it is my honor to count him as a friend and a business partner.

Whether you are a seasoned data professional or someone looking to unlock the potential of data in your organization, this book is your key to a data revolution. Databricks and the Lakehouse paradigm offer a fresh perspective on data management, and this book is your guide to mastering it. Alan's recipe-driven approach to teaching is perfect for the real world: it will enable you to deliver results quickly in the immediate term: and will help you connect the dots and build a strong foundation for self-learning over the longer term. The context-setting sections give you a quick history of features and approaches over the years. It will help you appreciate how the platform has evolved and most importantly, help you avoid old pitfalls and anti-patterns.

Apache Spark, Databricks and Lakehouse have transformed my life for the better: and it is my sincere hope and best wishes that you, the reader, have a similar fulfilling experience. Onward!

Subramanian Iyer
Principal, Speedboat Professional Services
Award-winning Architect and Certified Instructor on Databricks and Lakehouse
Ex-Brickster (2019-2023) and Spark fanboy since 2014

About the Author

Dr. Alan L. Dennis has been writing software for over 30 years. His experiences range from being one of the first employees at a startup to leading a team of over twenty developers. He has held titles such as Programmer, Architect, Chief Technical Officer, and member of technical staff. He has worked for many Fortune 50 companies, with a wide range of industry experience.

He holds a Doctorate in Computer Science with a concentration in Big Data Analytics, a Master's in Computer Science with specialization in Artificial Intelligence, and a Bachelor's of Business Administration with focus on Computer Information Systems. He teaches graduate classes at several universities and is a Databricks Certified Trainer.

About the Reviewers

- **Jay Kalathia** is an experienced Senior Software Engineer, with over two decades of experience in designing, developing, and optimizing cloud-based solutions on the Azure platform. He has a diverse background with proficiency in various programming languages such as Python, C#, JavaScript, and more. Jay is skilled in building infrastructure as code, developing CI/CD pipelines on Azure DevOps, and working on cloud-native solutions and tools including Azure, AWS, AKS, Kubernetes, and Terraform. Additionally, he has extensive experience with Azure Databricks and other cloud-based Big Data solutions. Jay is also a learner; from taking courses to completing certifications to stay up to date with latest technology trends.

- **Mahesh Das** is a Technology Evangelist and a Databricks Certified Data Science Professional. He is currently focused on the development of cloud-native solutions for ingesting and refining data from various sources using Azure Cloud, Databricks, Azure Data Factory, and Terraform scripts. Mahesh is actively engaged in projects implementing Large Language Models for diverse applications and remains a dedicated learner in the field of Machine Learning and Artificial Intelligence. With over 18 years of experience, he has contributed to numerous significant IT transformation projects for Fortune 500 clients across a range of industries, including Manufacturing, FMCG, oil and gas, Managed Print Services, and metals and mining. Mahesh possesses additional skills in AWS Cloud and SAP Master Data Governance, covering various functional domains such as sales and distribution, finance, and manufacturing. He is not only an avid reader of cutting-edge AI technologies but also serves as a technical reviewer for books within the same domain.

Acknowledgement

Many people have had a hand in this book. First, I would like to thank my parents for always supporting and encouraging me. They taught me to figure out how things work and, ideally, put them back together again afterward. My mother passed before this book could be completed. She will be missed, but her Heavenly Father called, and she went. I would also like to thank my wife Kim, for supporting and encouraging me and understanding when I disappear for weeks on end.

I would like to thank Jay, and Mahesh for their input to the book. They provided valuable feedback throughout the process. I would also like to thank Subramanian for his kind words in the foreword.

Lastly, I am incredibly thankful for you, gentle reader. I wrote this for you and I hope you find value in it. There is something for everyone in this book, let me know what you find.

Preface

It is commonly understood that valuable insights can be found in an organization's data. One way to extract that value is to construct a Data Lakehouse. This book helps you create a Lakehouse on the Databricks Platform. It is the culmination of decades of data processing design and implementation.

It is not easy to create a data ecosystem. There are many competing priorities and technical challenges. This book walks you through the process, providing hands-on examples. We organize those steps into recipes. This keeps the author from waxing on about theory and helps the reader find the information needed in a given situation. We cover the theory behind the approaches used and guide the reader to avoid common pitfalls.

We start with the basics, such as explaining what a Databricks Lakehouse is, why we need them, and what value it brings. We move on to applying the concepts in practice. Part of the reason for constructing a Data Lakehouse is to enable users to access its data. We then discuss the various personas that benefit from a Databricks Lakehouse.

While we start with the fundamentals, we rapidly move on to more advanced topics. A good understanding of SQL, Python, Spark, and cloud computing would benefit the reader but is not required.

Chapter 1: Introduction to Databricks Lakehouse – This chapter provides a brief history of Big Data, Spark, and Databricks. It introduces the reader to the community edition of Databricks as a starting point for using Databricks. We discuss why we construct a Lakehouse and present the overall architecture. We provide clear definitions for each of the layers of a Databricks Lakehouse. We discuss design considerations and compare Lakehouses to other data technologies.

Chapter 2: Setting-up a Databricks Workspace – This chapter presents the information necessary to provision and effectively use a Databricks environment. This includes examining core Databricks concepts, service tier selection, and cloud selection considerations. Deployment details are examined, including those with long-lasting implications. Access control and other configurations are discussed, along with the types of clusters and performance levels.

Chapter 3: Connecting to Storage – This chapter covers the approaches and tradeoffs to connect to storage. The Databricks File System is discussed in detail as it is an important element of the Lakehouse platform. The background of the file system is reviewed, and

various ways of connecting to storage are explored. The approaches to Lakehouse design are presented, with recommendations on how to organize a Lakehouse. Recommendations are provided regarding the documentation of allowed operations. Recipes containing various examples of connecting to Azure storage systems are provided.

Chapter 4: Creating Delta Tables – This chapter describes how to construct a Delta Lake, including a discussion of managed and external tables. Guidance is provided to help decide which type of table to create. Examples are provided of creating tables using SQL and the Spark API. Core concepts such as secrete scopes are discussed, along with example of creating tables from AWS S3, GCP buckets, and Azure ADLS.

Chapter 5: Data Profiling and Modeling in the Lakehouse – This chapter examines two of the more important activities when constructing a Data Lakehouse. Various ways of performing profiling are examined, including Databricks' native Data Profile feature. Discussion of the Databricks Describe and Summary features are included, along with analysis at scale using ydata_profiling.

Chapter 6: Extracting from Source and Loading to Bronze – This chapter covers the first step in refining data. A discussion is presented regarding using the raw zone or skipping it and going from source to bronze. Several ways of incrementally ingesting data are presented, which is essential for a high-performance Databricks Lakehouse. These methods include self-managed watermarks, Auto Loader, Delta Live Tables, and streaming data.

Chapter 7: Transforming to Create Silver – This chapter continues the refinement journey, picking up data at the Bronze layer and moving it to Silver. Both incremental and full refinement are discussed. Several approaches to processing are discussed, including the importance of data quality rules and expectations. Common Silver-to-Silver operations are discussed, including denormalization, JSON exploding, and projection reshaping.

Chapter 8: Transforming to Create Gold for Business Purposes – This chapter continues the discussion of refining data, with the goal of answering business questions. Gold tables are built to answer a specific question. The sources for Gold tables are discussed, with implementations in PySpark and Delta Live Tables. As Gold tables are optimized for consumption, a brief discussion of support-related operations such as vacuuming and optimizing tables is present.

Chapter 9: Machine Learning and Data Science – Data scientists are common users of the Databricks Lakehouse. We examine using Machine Learning in Databricks, and the use of AutoML. Next, we discuss MLflow, and the importance it plays in deploying models to production. Lastly, we briefly discuss the Databricks feature store.

Chapter 10: SQL Analysis – SQL is one of the most widely known languages. We discuss the SQL Analysis features built into Databricks, including Databricks SQL. We show how to create and manage a SQL Warehouse. We discuss the usage of the SQL Editor and use it to write common queries. We create dashboards and alerts using those queries. We close with a discussion of cost and performance considerations.

Chapter 11: Graph Analysis – There are many ways to perform analysis; one way is to use mathematical graph algorithms. We discuss the nature of graphs and when using graph algorithms is appropriate. We discuss GraphX and GraphFrames, along with the operations they enable and associated algorithms. Lastly, we discuss reading data from Neo4J's AuraDB from Databricks.

Chapter 12: Visualizations – There are many ways to present data; visualizations can be very powerful. We discuss visualization best practices and how to create a Databricks dashboard. We also discuss native visualization support within a Databricks notebook. We conclude the chapter by discussing the use of Power BI with Databricks.

Chapter 13: Governance – Without proper governance, a Databricks Lakehouse will not be successful. We discuss the role of data governance and the use of Databricks' Unity Catalog. We walk through the installation and usage of Unity Catalog and review the major benefits. We discuss the steps to install and use Azure Purview in combination with Unity Catalog.

Chapter 14: Operations – This chapter covers the steps necessary to keep a Lakehouse working effectively, including source code management and orchestration. Preventive scheduled maintenance can help avoid unacceptable processing time and outages. We also discuss how to manage and maintain visibility of costs.

Chapter 15: Tips, Tricks, Troubleshooting, and Best Practices – This final chapter contains important elements that did not make it into other parts of the book. We revisit ingesting data, by ingesting relational data. Discuss performance optimizations such as using pools. We discuss how to orchestrate notebooks. Lastly, we conclude with a discussion of best practices and guiding principles.

Code Bundle and Coloured Images

Please follow the link to download the
Code Bundle and the ***Coloured Images*** of the book:

https://rebrand.ly/llidt00

The code bundle for the book is also hosted on GitHub at **https://github.com/bpbpublications/Databricks-Lakehouse-Platform-Cookbook**.
In case there's an update to the code, it will be updated on the existing GitHub repository.

We have code bundles from our rich catalogue of books and videos available at **https://github.com/bpbpublications**. Check them out!

Errata

We take immense pride in our work at BPB Publications and follow best practices to ensure the accuracy of our content to provide with an indulging reading experience to our subscribers. Our readers are our mirrors, and we use their inputs to reflect and improve upon human errors, if any, that may have occurred during the publishing processes involved. To let us maintain the quality and help us reach out to any readers who might be having difficulties due to any unforeseen errors, please write to us at :

errata@bpbonline.com

Your support, suggestions and feedbacks are highly appreciated by the BPB Publications' Family.

Piracy

If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at **business@bpbonline.com** with a link to the material.

If you are interested in becoming an author

If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit **www.bpbonline.com**. We have worked with thousands of developers and tech professionals, just like you, to help them share their insights with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions. We at BPB can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about BPB, please visit **www.bpbonline.com**.

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com

Table of Contents

Chapter 1
Introduction to Databricks Lakehouse

Introduction

Welcome to our journey of learning and mastering the Databricks Lakehouse Platform. This is a hands-on book. While we will cover each topic's theoretical and technical foundations, you will have code to help you learn how to build a Lakehouse and succeed using Databricks.

Structure

In this chapter, we will cover the following topics:

- Background
- Brief history of Big Data, Spark, and Databricks
- Databricks community edition
- Data Lakehouse value proposition
- Lakehouse architecture
- Design considerations
- Lakehouse compared to other data technologies

Objectives

This chapter introduces nomenclature commonly used when discussing the Databricks Lakehouse Platform. By the end of the chapter, you should be able to describe a typical Lakehouse configuration and understand the architectural components and the value proposition of the lakehouse architecture.

Background

It is often important to understand a phenomenon's history that influenced its creation, and Data Lakehouse is no exception. We start with a brief history of Big Data, Spark, and Databricks. We briefly discuss the Databricks community edition and perform our first analysis in Databricks. We close this section by discussing the value proposition that drives the adoption of Data lakehouse, particularly Databricks Lakehouse.

Brief history of Big Data, Spark, and Databricks

When looking at how things came to be, we often discuss supporting and challenging forces. In the case of Big Data, several forces were driving its adoption. One key supporting force was the shift of the Internet from companies and government entities sharing information with their customers to users of platforms creating content in social media. Companies also learned that online sales had many advantages over traditional outlets, including lower operating costs. This shift generated vast amounts of data that previously was minimal. Online retailers learned that examining those log files could give insights into their customers that previously was not possible. The desire to process this information, which was too large to process with traditional file-processing approaches, led to the creation of a new set of technologies. Big Data was used to label these distributed, software-based fault-tolerant algorithms and technologies.

One of the early success stories of Big Data was Hadoop. Hadoop is a collection of open-source projects related to processing large, fast, or variant data. An early processing approach in Hadoop was called MapReduce. MapReduce was a framework that simplified the process of creating distributed solutions. Before Big Data frameworks like MapReduce, software developers coordinated activities between various workers attempting to

work together to solve a problem. Often, one or more of those workers would become unavailable. The software developer's job was to determine how to address this and many other challenges. With MapReduce, a developer was tasked with writing a few functions called by a framework to simplify the process. While MapReduce was a significant advancement, it was limited by its original design and purpose. MapReduce was focused on processing large or numerous files. Due to this design goal, it failed to support iteration and relied heavily on disk drives.

Spark was developed to address many of these challenges. Spark is a computational solution that relies on other technologies for storage. It also favors processing data in memory, resulting in significant performance improvements over MapReduce. Spark also enabled iteration during processing. These advancements lead to its rapid adoption and increasing popularity. Many of the creators of Spark formed Databricks in 2013. Databricks is a cloud company supporting the major cloud vendors. In 2017 Azure Databricks was announced. This partnership was notable because of the high integration between Azure and Databricks.

A data lakehouse is an architecture that combines the best elements of data lakes to address data warehousing needs. It is an open standards-based set of technologies. A key distinction of data lakehouse from data lakes is that it uses a schema and **Atomic, Consistent, Isolated, and Durable** (**ACID**) transactions. Lakehouses allow updates to a record, while data lakes treat data as immutable. In Databricks, Spark is the computational engine supporting all lakehouse processing, and Delta Lake is the storage format used to enable ACID transactions and schemas. Delta Lake is based on the Parquet format, with transaction logs in **JavaScript Object Notation** (**JSON**) to journal interactions with data. A Delta Lake exists on top of data lakes and cloud storage containers.

Databricks community edition

Databricks understands that learning technology is essential for its adoption. Databricks offers a community edition of its platform to enable learning and smaller workloads. The community edition offers limited functionality compared to the enterprise-class versions available on AWS, Azure, and **Google Cloud Platform** (**GCP**). The community edition has several restrictions, including little computational power and lacks automation capabilities via an API. To learn more about the Databricks community edition, go to **https://docs.databricks.com/getting-started/community-edition.html**.

Recipe 1: Signing up for the Databricks community edition

To sign up for the Databricks community edition, go to **https://www.databricks.com/try-databricks** and fill out the form, as shown in *Figure 1.1*. You will be asked for your name, email, company, and job title:

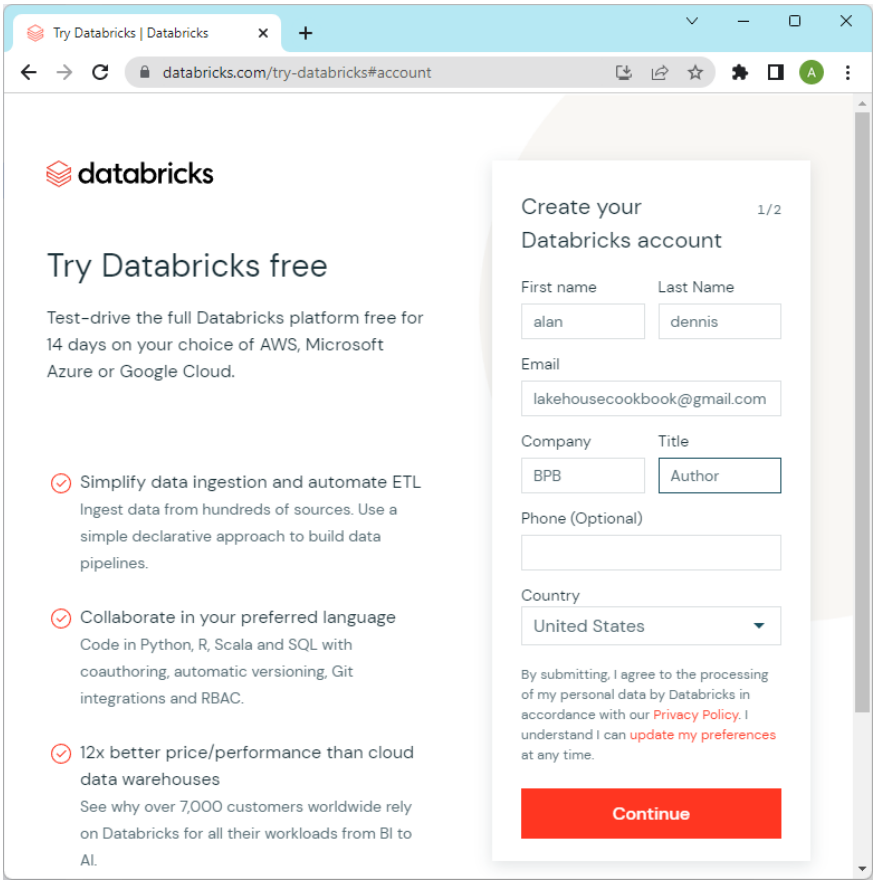

Figure 1.1: *Sign-up for Databricks Community Edition*

After clicking **Continue**, you are presented with a page asking you to choose your cloud provider, as shown in *Figure 1.2*. Under the section that refers to not having a cloud account, there is a link titled **Get Started** with Community Edition. It is relatively small and easy to miss, but it is how to sign up for the free community edition.

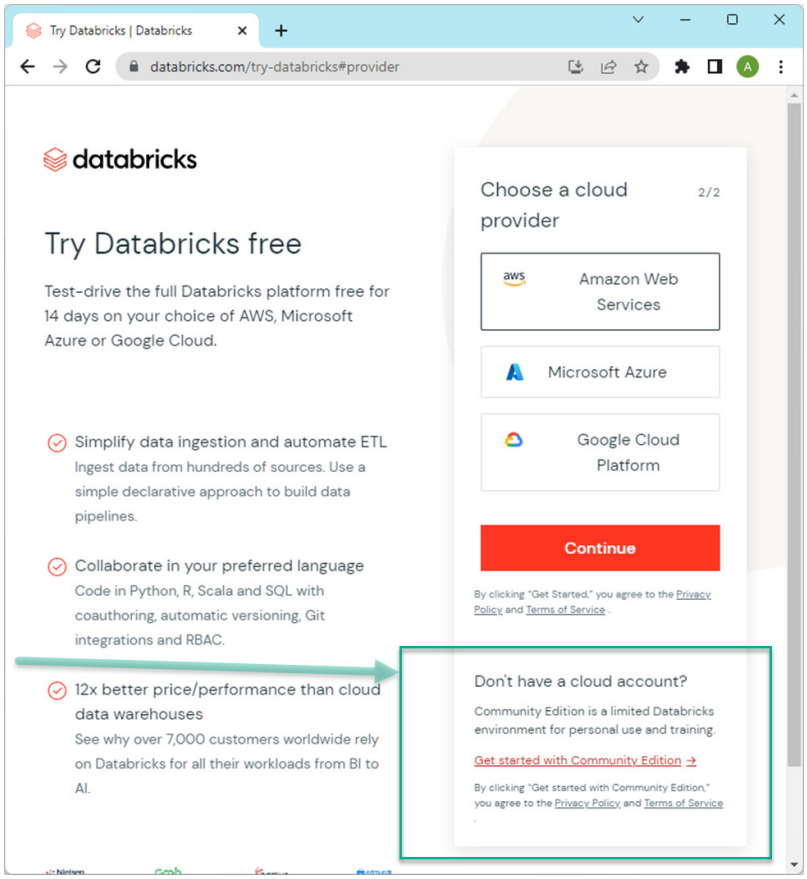

Figure 1.2: *Select getting started with Databricks Community Edition*

After clicking the link, you will likely be asked to prove you are a human by solving a simple puzzle. After solving the puzzle, you will be redirected to a page asking you to confirm your email address. Check your email and click the link in the body to confirm receipt of the email message. You are then asked to provide a password for logging into the tool. After supplying a password, you will be redirected to the Databricks community edition home page.

Recipe 2: Creating a notebook in the Databricks Community edition

The areas of the community edition Databricks workspace are similar to that of the enterprise-class cloud-hosted versions, as shown in *Figure 1.3*. The layout is organized