**Chapter 33. Ecosystem and Community**

One of Spark’s biggest selling points is the sheer volume of resources, tools, and contributors. At the time of this writing, there are over 1,000 contributors to the Spark codebase. This is orders of magnitude more than most other projects dream of achieving and a testament to Spark’s amazing community—both in terms of contributors and stewards. The Spark project shows no sign of slowing down, as companies large and small seek to join the community. This environment has stimulated a large number of projects that complement and extend Spark’s features, including formal Spark packages and informal extensions that users can use in Spark.

**Spark Packages**

Spark has a package repository for packages specific to Spark: [Spark Packages](https://spark-packages.org/). These packages were discussed in Chapters [9](https://www.safaribooksonline.com/library/view/spark-the-definitive/9781491912201/ch09.html#s2c6---data-sources) and [24](https://www.safaribooksonline.com/library/view/spark-the-definitive/9781491912201/ch24.html#s6c1---advanced-analytics-and-machine-learning). Spark packages are libraries for Spark applications that can easily be shared with the community. [GraphFrames](http://graphframes.github.io/) is a perfect example; it makes graph analysis available on Spark’s structured APIs in ways much easier to use than the lower-level (GraphX) API built into Spark. There are numerous other packages, including many machine learning and deep learning ones, that leverage Spark as the core and extend its functionality.

Beyond these advanced analytics packages, others exist to solve problems in particular verticals. Healthcare and genomics have seen a surge in opportunity for big data applications. For example, the [ADAM Project](http://bdgenomics.org/) leverages unique, internal optimizations to Spark’s Catalyst engine to provide a scalable API & CLI for genome processing. Another package, [Hail](https://hail.is/), is an open source, scalable framework for exploring and analyzing genomic data. Starting from sequencing or microarray data in VCF and other formats, Hail provides scalable algorithms to enable statistical analysis of gigabyte-scale data on a laptop or terabyte-scale data on cluster.

At the time of this writing, there are nearly 400 different packages to choose. As a user, you can specify Spark packages as dependencies in your build files (as seen in this book’s [book GitHub repository](https://github.com/databricks/Spark-The-Definitive-Guide/)). You can also download the pre-built jars and include them in your class path without explicitly adding them to your build file. Spark packages can also be included at runtime by passing a parameter to the spark-shell or spark-submit command-line tools.

**An Abridged List of Popular Packages**

As mentioned, there are nearly 400 Spark packages. Including all of these is not relevant to you as a user because you can search for specific packages on the Spark package website. However, it is worth mentioning some of the more popular packages:

[Spark Cassandra Connector](https://github.com/datastax/spark-cassandra-connector/)

This connector helps you get data in and out of the Cassandra database.

[Spark Redshift Connector](https://github.com/databricks/spark-redshift)

This connector helps you get data in and out of the Redshift database.

[Spark bigquery](https://github.com/spotify/spark-bigquery)

This connector helps you get data in and out of Google’s BigQuery.

[Spark Avro](https://github.com/databricks/spark-avro)

This package allows you to read and write Avro files.

[Elasticsearch](https://github.com/elastic/elasticsearch-hadoop)

This package allows you to get data into and out of Elasticsearch.

[Magellan](https://github.com/harsha2010/magellan)

Allows you to perform geo-spatial data analytics on top of Spark.

[GraphFrames](http://graphframes.github.io/)

Allows you to perform graph analysis with DataFrames.

[Spark Deep Learning](https://github.com/databricks/spark-deep-learning)

Allows you to leverage Deep Learning and Spark together.

**Using Spark Packages**

There are two core ways you can include Spark Packages in your projects. In Scala or Java, you can include it as a build dependency, or you can also specify your packages at runtime (for Python or R). Let’s review the ways in which you can include this information.

**IN SCALA**

Including the following resolver in your *build.sbt* file will allow you to include Spark packages as dependencies. For example, we can add this resolver:

*// allows us to include spark packages*

resolvers += "bintray-spark-packages" at

"https://dl.bintray.com/spark-packages/maven/"

Now that we added this line, we can include a library dependency for our Spark package:

libraryDependencies ++= **Seq**(

...

*// spark packages*

"graphframes" % "graphframes" % "0.4.0-spark2.1-s\_2.11",

)

This is to include the GraphFrames library. There are slight versioning differences between packages, but you can always find this information on the Spark packages website.

**IN PYTHON**

At the time of this writing , there is no explicit way to include a Spark package as a dependency in a Python package. These sorts of dependencies must be set at runtime.

**AT RUNTIME**

We saw how we can specify Spark packages in Scala packages, but we can also include these packages at runtime. This is as simple as including a new argument to the spark-shell and spark-submit that you would use to run your code.

For example, to include the magellan library:

$SPARK\_HOME/bin/spark-shell --packages harsha2010:magellan:1.0.4-s\_2.11

**External Packages**

In addition to the formal Spark Packages, there are a number of informal packages that are built on or leverage Spark’s capabilities. A prime example is the popular gradient-boosted, decision-tree framework [XGBoost](https://github.com/dmlc/xgboost), which makes use of Spark for scheduling distributed training on individual partitions. A number of these are liberally licensed, public projects available on GitHub. Using your favorite search engine is a great way to discover projects that may already exist, rather than having to write your own.

**Community**

Spark has a large, robust community. It is so much larger than the packages and direct contributions. The ecosystem of end users who build Spark into their products and write tutorials is an ever-growing group. As of this writing, there are over 1,000 contributors to the repository on Github.

The [official Spark website](http://spark.apache.org/community.html) maintains the most up-to-date community information, including mailing lists, improvement proposals, and project committers. This website also includes many resources about new Spark versions, documentation, and release notes for the community.

**Spark Summit**

Spark Summits are events that occur across the globe at various times a year. This is the canonical event for Spark-related talks, where thousands of end users and developers attend these summits to learn about the cutting edge in Spark and hear about use cases. There are hundreds of tracks and training courses over the course of several days. In 2016, there were three events: New York (Spark Summit East), San Francisco (Spark Summit West), and Amsterdam (Spark Summit Europe). In 2017, there were Spark Summits in Boston, San Francisco, and Dublin. Coming in 2018—and beyond—there will be even more events. Find out more at at the [Spark Summit website](https://spark-summit.org/).

There are hundreds of freely available [Spark Summit videos](https://www.youtube.com/user/TheApacheSpark) for learning about use cases, Spark’s development, and strategies and tactics that you can use to get the most out of Spark. You can browse historical Spark Summit talks and videos on the [website](https://spark-summit.org/).

**Local Meetups**

There are many Spark-related meetup groups on [meetup.com](https://www.meetup.com/). Figure 33-1 shows a map of Spark-related meetups on Meetup.com.



*Figure 33-1. Spark meetup map*

Spark’s “official meetup group” in the Bay Area (founded by one of the authors of this book), can be found [here](https://www.meetup.com/spark-users/). However, there are over 600 Spark-related meetups around the world, totaling nearly 350,000 members. These meetups continue to spring up and grow, so be sure to find one in your area.

**Conclusion**

This whirlwind chapter discussed nontechnical resources that Spark makes available. One important fact is that one of Spark’s greatest assets is the Spark community. We are extremely proud of the community’s involvement in the development of Spark and love to hear about what companies, academic institutions, and individuals build with Spark.

We sincerely hope that you’ve enjoyed this book and we look forward to seeing you at a Spark Summit!