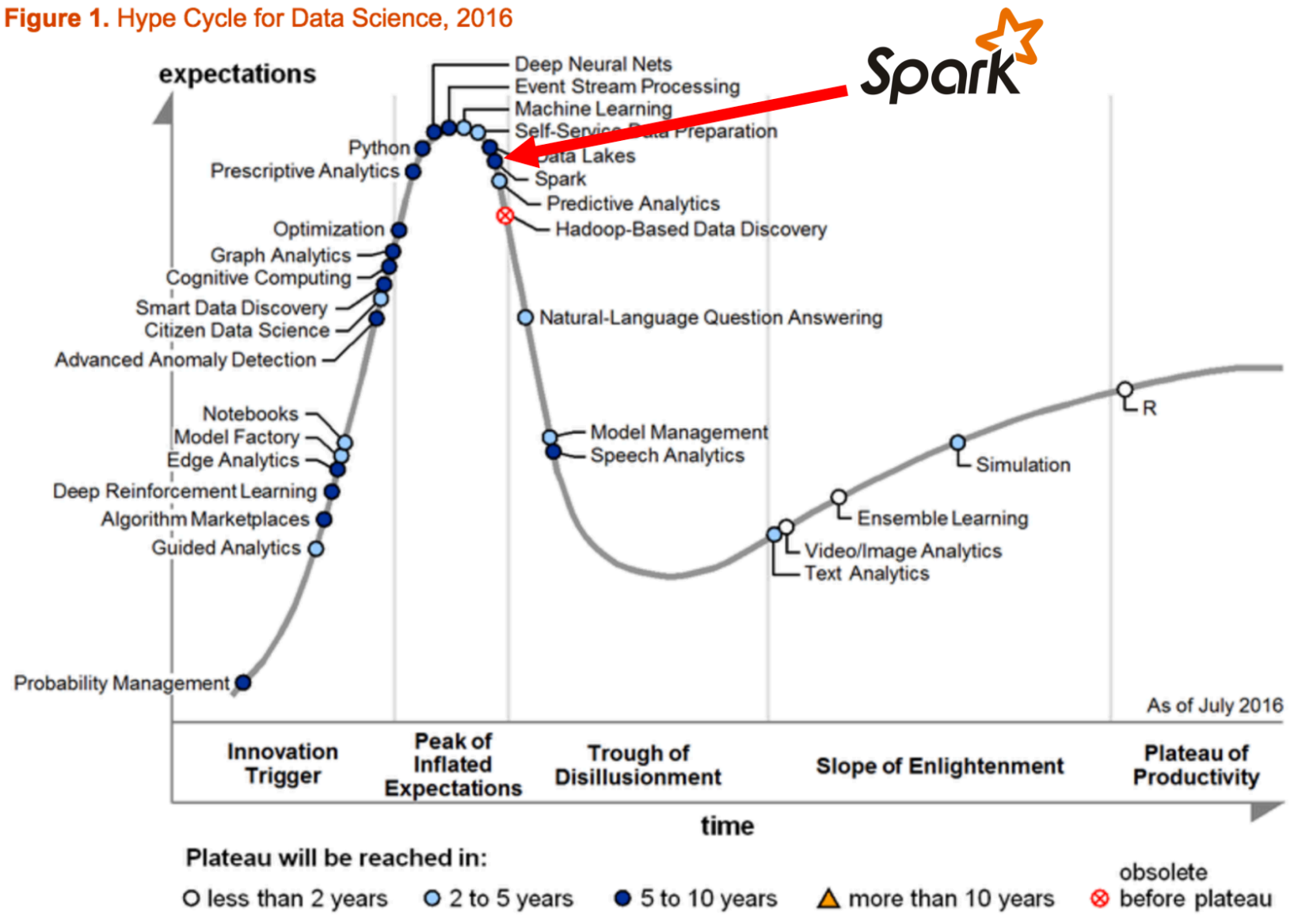
SPARK IS THE FUTURE OF ANALYTICS

At the 2016 Spark Summit, Gartner Research Director Nick Heudecker asked: *Is Spark the Future of Data Analysis?* It's an interesting question, and it requires a little parsing. Nobody believes that Spark alone is the future of data analysis, even its most ardent proponents. A better way to frame the question: *Does Spark have a role in the future of analytics? What is that role?*

Unfortunately, Heudecker didn't address the question but spent the hour throwing shade at Spark.

Spark is overhyped! He declared. His evidence? This:



Source: Gartner (July 2016)

One might question an analysis that equates real things like optimization with fake things like "Citizen Data Science." Gartner's Hype Cycle by itself proves nothing; it's a conceptual salad, with neither empirical foundation nor predictive power.

If you want to argue that Spark is overhyped, produce some false or misleading claims by project principals, or documented cases where the software failed to work as claimed. It's possible that such cases exist. Personally, I don't know of any, and neither does Nick Heudecker, or he would have included them in his presentation.

Instead, he cited a Gartner survey showing that organizations don't use Spark and Flink as much as they use other tools for data analysis. From my notes, here are the percentages:

- EDW: **57%**
- Cloud: **44%**
- Hadoop: **42%**
- Stat Packages: **32%**
- Spark or Flink: **9%**
- Graph Databases: **8%**

That 42% figure for Hadoop is interesting. In 2015, Gartner concern-trolled the tech community, trumpeting the finding that "only" **26%** of respondents in a survey said they were "deploying, piloting or experimenting with Hadoop." So — either Hadoop adoption grew from 26% to 42% in a year, or Gartner doesn't know how to do surveys.

In any event, it's irrelevant; statistical packages have been available for 40 years, EDWs for 25, Spark for 3. The current rate of adoption for a project in its youth tells you very little about its future. It's like arguing that a toddler is cognitively challenged because she can't do integral calculus without checking the Wolfram app on her iPad.

Heudecker closed his presentation with the pronouncement that he had no idea whether or not Spark is the future of data analysis, and bolted the venue faster than a jackrabbit on Ecstasy. Which begs the question: *why pay big bucks for analysts who have no opinion about one of the most active projects in the Big Data ecosystem?*

Here are eight reasons why Spark has a central role in the future of analytics.

(1) Nearly everyone who uses Hadoop will use Spark.

If you believe that **42%** of enterprises use Hadoop, you must believe that **41.9%** will use Spark. Every Hadoop distribution includes Spark. Hive and Pig run on Spark. Hadoop early adopters will gradually replace existing MapReduce applications and build most new applications in Spark. Late adopters may never use MapReduce.

The only holdouts for MapReduce will be those who want their analysis the way they want their barbecue: low and slow.

Of course, Hadoop adoption isn't static. Forrester's Mike Gualtieri argues that **100%** of enterprises will use Hadoop within a few years.

(2) Lots of people who don't use Hadoop will use Spark.

For Hadoop users, Spark is a fast replacement for MapReduce. But that's not all it is.

Spark is also a general-purpose data processing environment for advanced analytics. Hadoop has baggage that data science teams don't need, so it's no surprise to see that most Spark users *aren't* using it with Hadoop. One of the key advantages of Spark is that users aren't tied to a particular storage back end, but can choose from many different options. That's essential in real-world data science.

(3) For scalable open source data science, Spark is the only game in town.

If you want to argue that Spark has no future, you're going to have to name an alternative. I'll give you a minute to think of something.

Time's up.

You could try to approximate Spark's capabilities with a collection of other projects: for example, you could use Presto for SQL, H2O for machine learning, Storm for streaming, and Giraph for graph analysis. Good luck pulling those together. H2O.ai was one of the first vendors to build an interface to Spark because even if you want to use H2O for machine learning, you're still going to use Spark for data wrangling.

"What about Flink?" you ask. Well, what about it? Flink may have a future, too, if anyone ever supports it other than ten guys in a loft on the Tempelhofer Ufer. Flink's event-based runtime seems well-suited for "pure" streaming applications, but that's low-value bottom-of-the-stack stuff. Flink's ML library is still pretty limited, and improving it doesn't appear to be a high priority for the Flink team.

(4) Data scientists who work exclusively with "small data" still need Spark.

Data scientists satisfy most business requests for insight with small datasets that can fit into memory on a single machine. Even if you measure your largest dataset in gigabytes, however, there are two ways you need Spark: to create your analysis dataset and to parallelize operations.

Your analysis dataset may be small, but it comes from a larger pool of enterprise data. Unless you have servants to pull data for you, at some point you're going to have to get your hands dirty and deal with data at enterprise scale. If you are lucky, your organization has nice clean data in a well-organized data warehouse that has everything anyone will ever need in a single source of truth.

Ha ha! Just kidding. Single sources of truth don't exist, except in the wildest fantasies of data warehouse vendors. In reality, you're going to muck around with many different sources and integrate your analysis data on the fly. Spark excels at that.

For best results, machine learning projects require hundreds of experiments to identify the best algorithm and optimal parameters. If you run those tests serially, it will take

forever; distribute them across a Spark cluster, and you can radically reduce the time needed to find that optimal model.

(5) The Spark team isn't resting on its laurels.

Over time, Spark has evolved from a research project for scalable machine learning to a general purpose data processing framework. Driven by user feedback, Spark has added SQL and streaming capabilities, introduced Python and R APIs, re-engineered the machine learning libraries, and many other enhancements.

Here are some projects under way to improve Spark:

— Project Tungsten, an ongoing effort to optimize CPU and memory utilization.

— A stable serialization format (possibly Apache Arrow) for external code integration.

— Integration with deep learning frameworks, including TensorFlow and Intel's new BigDL library.

— A cost-based optimizer for Spark SQL.

— Improved interfaces to data sources.

— Continuing improvements to the Python and R APIs.

Performance improvement is an ongoing mission; for selected operations, Spark 2.0 runs 10X faster than Spark 1.6.

(6) More cool stuff is on the way.

Berkeley's AMPLab, the source of Spark, Mesos, and Tachyon/Alluxio, is now RISELab. There are four projects under way at RISELab that will extend Spark capabilities:

— Clipper is a prediction serving system that brokers between machine learning frameworks and end-user applications. The first Alpha release, planned for mid-April 2017, will serve scikit-learn, Spark ML and Spark MLLib models, and arbitrary Python functions.

— Drizzle, an execution engine for Apache Spark, uses group scheduling to reduce latency in streaming and iterative operations. Lead developer Shivaram Venkataraman has filed a design document to implement this approach in Spark.

— Opaque is a package for Spark SQL that uses Intel SGX trusted hardware to deliver strong security for DataFrames. The project seeks to enable analytics on sensitive data in an untrusted cloud, with data encryption and access pattern hiding.

— Ray is a distributed execution engine for Spark designed for reinforcement learning.

Three Apache projects in the Incubator build on Spark:

— Apache Hivemall is a scalable machine learning library implemented as a collection of Hive UDFs designed to run on Hive, Pig or Spark SQL with MapReduce, Tez or Spark.

— Apache PredictionIO is a machine learning server built on top of an open source stack, including Spark, HBase, Spray, and Elasticsearch.

— Apache SystemML is a library of machine learning algorithms that run on Spark and MapReduce, originally developed by IBM Research.

MIT's CSAIL lab is working on ModelDB, a system to manage machine learning models. ModelDB extracts and stores model artifacts and metadata, and makes this data available for easy querying and visualization. The current release supports Spark ML and scikit-learn.

(7) Commercial vendors are building on top of Spark.

The future of analytics is a hybrid stack, with open source at the bottom and commercial software for business users at the top. Here is a small sample of vendors who are building easy-to-use interfaces atop Spark.

— Alpine Data provides a collaboration environment for data science and machine learning that runs on Spark (and other platforms.)

— AtScale, an OLAP on Big Data solution, leverages Spark SQL and other SQL engines, including Hive, Impala, and Presto.

— Dataiku markets Data Science Studio, a drag-and-drop data science workflow tool with connectors for many different storage platforms, scikit-learn, Spark ML and XGboost.

— StreamAnalytix, a drag-and-drop platform for real-time analytics, supports Spark SQL and Spark Streaming, Apache Storm, and many different data sources and sinks.

— Zoomdata, an early adopter of Spark, offers an agile visualization tool that works with Spark Streaming and many other platforms.

All of the leading agile BI tools, including Tableau, Qlik, and PowerBI, support Spark. Even stodgy old Oracle's Big Data Discovery tool runs on Spark in Oracle Cloud.

(8) All of the leading commercial advanced analytics platforms use Spark.

All of them, including SAS, a company that embraces open source the way Sylvester the Cat embraces a skunk. SAS supports Spark in [SAS Data Loader for Hadoop], one of SAS' five different Hadoop architectures. (If you don't like SAS architecture, wait six months for another.)

Magic Quadrant for Advanced Analytics Platforms, 2016

— IBM embraces Spark like Romeo embraced Juliet, hopefully with a better ending. IBM [contributes] heavily to the Spark project and has rebuilt many of its software products and cloud services to use Spark.

— KNIME's [Spark Executor] enables users of the KNIME Analytics Platform to create and execute Spark applications. Through a combination of visual programming and scripting, users can leverage Spark to access data sources, blend data, train predictive

models, score new data, and embed Spark applications in a KNIME workflow.

— RapidMiner's Radoop module supports visual programming across SparkR, PySpark, Pig, and HiveQL, and machine learning with SparkML and H2O.

— Statistica, which is no longer part of Dell, offers Spark integration in its Expert and Enterprise editions.

— Microsoft supports Spark in AzureHD, and it has rebuilt Microsoft R Server's Hadoop integration to leverage Spark as well as MapReduce. VentureBeat reports that Databricks will offer its managed service for Spark on Microsoft Azure later this year.

— SAP, another early adopter of Spark, supports Vora, a connector to SAP HANA.

You get the idea. Spark is deeply embedded in the ecosystem, and it's foolish to argue that it doesn't play a central role in the future of analytics.