Microsoft

The scientist, the engineer, and the warehouse

Building the right team for data analytics in the age of cloud

Donald Farmer
Principal, TreeHive Strategy

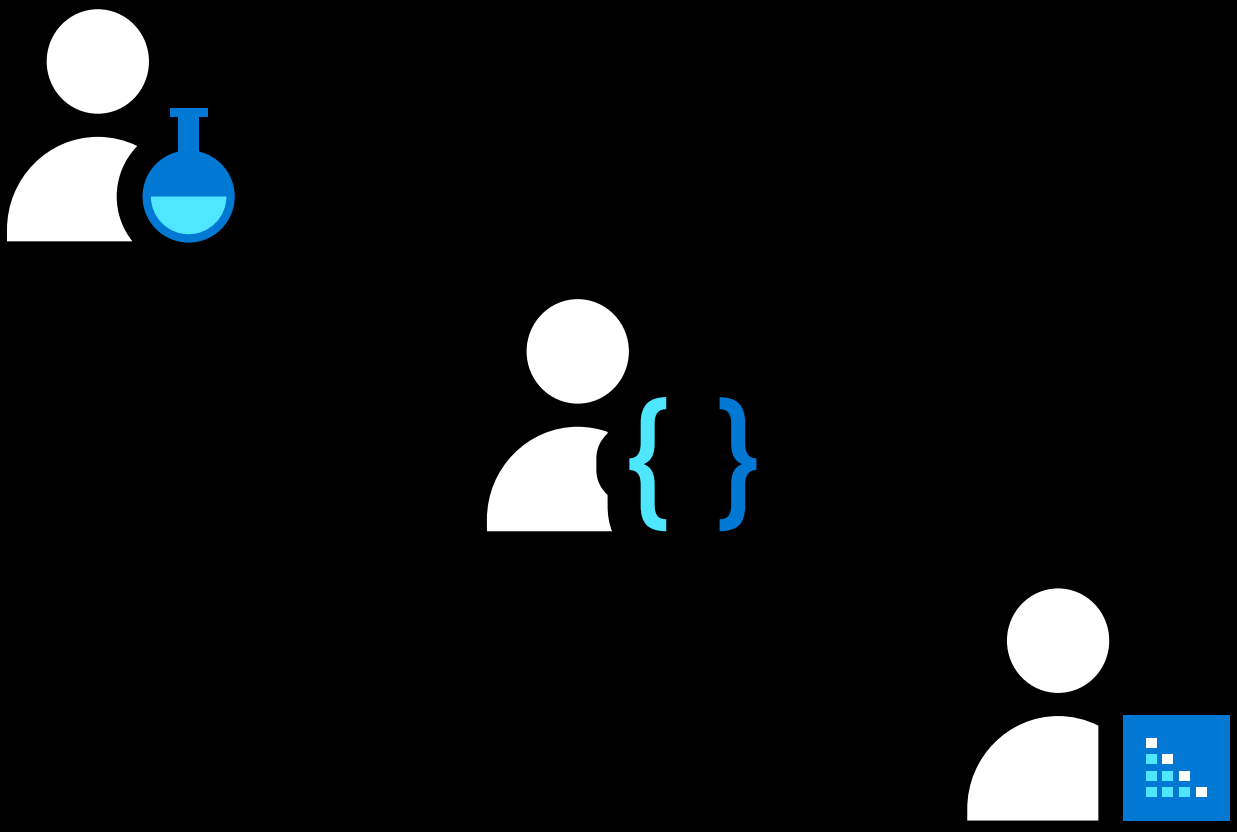


Table of contents

Introduction

"I need to hire a data scientist!"

We hear this call, urgent and animated, from Chief Information Officers (CIOs) and Chief Technical Officers (CTOs) almost every week. We understand. It's easy to get excited about new technology, especially when it promises a shortcut to those most tantalizing of strategic objectives: innovation, competitive advantage, and efficiency. Few fields have recently generated such enthusiasm as data science, which broadly covers machine learning, artificial intelligence, and big data. Much of this eagerness is justified: the headline achievements have often been startling.

Too few organizations have given thought to supporting data science as both a technical and a business practice.

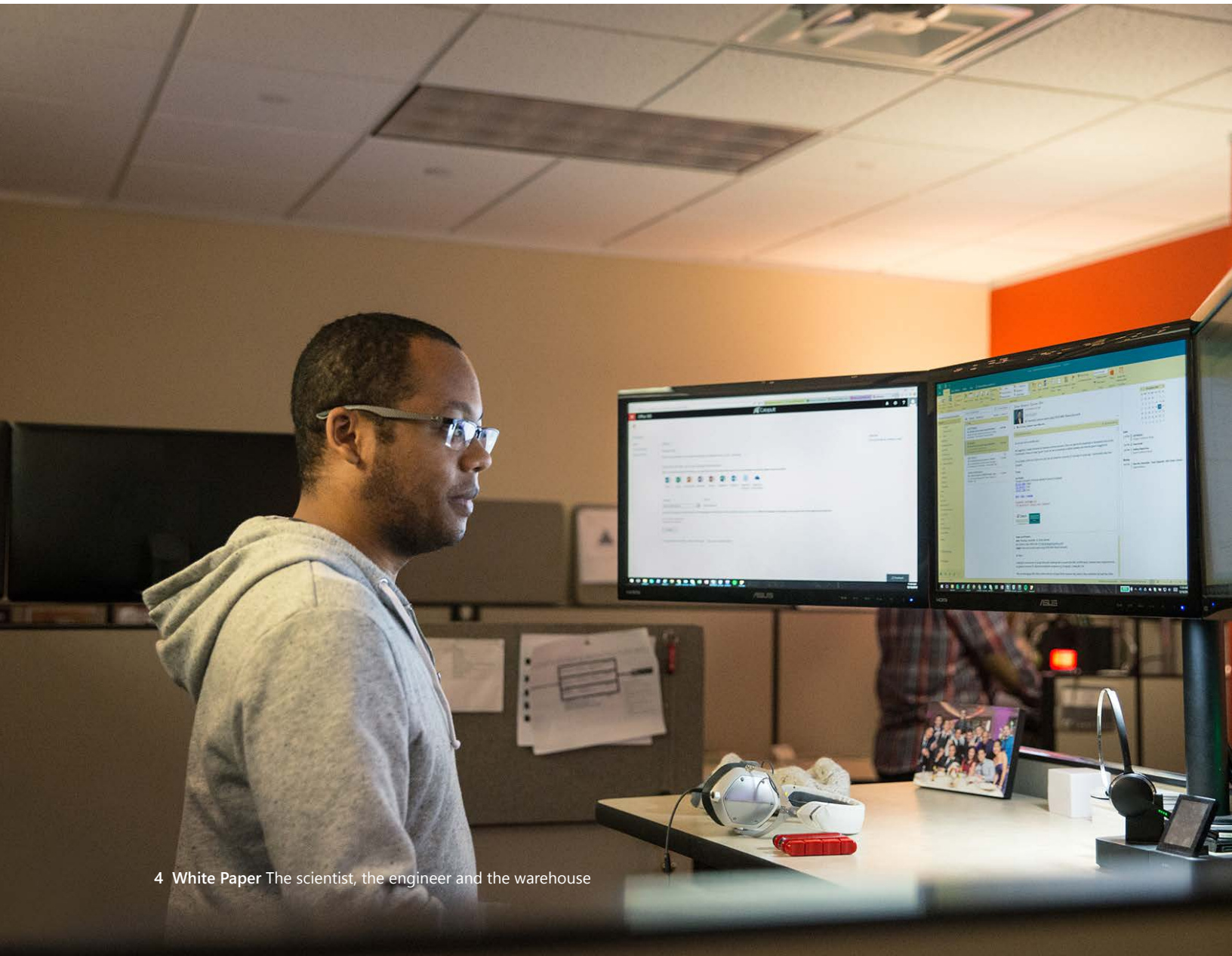
Less exciting—at least to the headline writers—are the managerial and architectural changes needed to support radical new practices as they emerge. Too few organizations have given thought to supporting data science as both a technical and a business practice This is partly driven by a simple lack of knowledge about how machine learning, and particularly artificial intelligence work. These practices have a mystique of their own which can seem quite distant from IT's day-to-day work. Companies often lack foresight, too. Results from predictive models need to be available and applicable in the real world of your operations, at a scale, and with a reliability that matches company demands.

Data warehousing is a core technology that enables data science to power business at an enterprise scale and is well-established and widely available. The concept of a data warehouse, first defined by Barry Devlin at IBM in 1985, is still a powerful technology and is far from being left behind by data lakes, pipelines, scripts, and algorithms. In fact, the data warehouse's central role—to serve integrated data and a canonical model of operations—has never been more authoritative. Thanks to new cloud architectures and in-memory engines, the technical platform is still highly relevant.

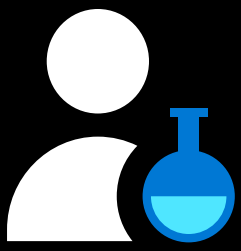
In companies that stand out in the field of data science, three new organizational roles are emerging: the data scientist, the data engineer responsible for ensuring predictive models are production-ready, and a new generation of data literate analysts in marketing, finance, and sales operations.

From one perspective, business users are remarkably independent. We saw this when they brought their own mobile devices into the working environment, with or without IT's permission. Today, these same users develop data literacy skills at college and through their everyday participation in corporate systems. They adopt self-service analytic tools as needed. Yet, IT must still plan to support business workers, even as their skills and demands arise organically in the modern enterprise.

To hire a data scientist, however, is a deliberate step towards a new strategic horizon. In this paper, we'll explore how we can serve these new roles, and how the modern cloud data warehouse plays critically in this space. We think you will soon realize why you be saying, "I need to hire a data engineer, too."



For the data scientist:
the experiment



The term data science as we now use it so much, was coined to define a business-focused role working particularly with big data.

The term data science as we now use it so much, was coined to define a business-focused role working particularly with big data. Although the buzz around the role is quite contemporary, data scientists have been around for a long time. Previously, diverse industries from railroads and insurance to pharmaceuticals and agriculture, employed specialists in statistical modeling, and often linear programming. These specialists worked with large volumes of raw data, scripts, and algorithms. However, the term data science, as we now use it, was coined by D.J. Patil and Jeff Hammerbacher around 2008, to define a business-focused role working particularly with big data.

No one becomes a good data scientist by solely working with data. Rather, they approach their role with three key skills: mathematics and statistics, coding and data preparation, grounded in the commercial realities of their employer's trade. To be fair, these latter skills are often learned on the job, particularly in specialized vertical markets. But no data scientists can afford to think of his or her work as an abstract study, removed from the real world.

A notable difference between this new role and a traditional business analyst is the importance of experimentation within the practice. You will remember the empirical scientific method from high school: a sequence of steps that leads from the formulation of a hypothesis to testing that hypothesis through experiments, to finally refining the hypothesis in the light of experimental results.

The data scientist works in a similar way. First, they formulate a hypothesis of what will be interesting and useful for the business, pulling together data relevant to that hypothesis. This 'data ingestion', which sounds rather messy and biological, simply means that data scientists bring data they need from various sources into their own system in order to work on it without impacting anyone else.

The second step in our data science method is experimenting with models and data. Then, just as in the natural sciences, based on results, we typically refine the hypothesis to improve its applicability and accuracy. It is in these phases that more advanced mathematical work will be applied. Finally, we apply our findings in practice, often as a set of rules derived from the model.

The purpose of analytic experiments is to uncover insights otherwise hidden to management and operations.

One fundamental implication of the machine learning workflow is the need for raw data, as close to its original state as possible. This is because every process of preparing and cleaning data, and every effort made to apply a model to it, involves a set of assumptions about how business works. We may think these assumptions are essential, as we accept them as true or certain. But the data scientist often wants to dig beneath the surface of our suppositions, to find out what our data really says about our operations, rather than what we think it should say. Many of the most exciting discoveries in data science, as in natural science, have been challenges to existing ideas about how things work. The purpose of analytic experiments, in other words, is to uncover insights otherwise hidden to management and operations. We can't do that, if we in turn hide the raw data behind a standardized process.

On the other hand, as we'll see later (The data warehouse as a source for data science) the warehouse schema can be a useful source of structural knowledge and the conformed data is an invaluable reference set. It's one thing to analyze raw, unprocessed, data from the log files of, say, a conveyor belt controller. But if the IDs, model numbers, and locations of all the company's conveyor belts have already been collected and standardized in dimension tables, there's no need to do that work again, and using the approved reference data can help when it comes to applying the resulting models back into the business.

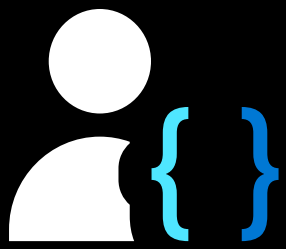
Another telling aspect of a data scientist's method, which may be very different from the work of a business analyst, is the diversity of data sources in use and the volumes of data involved. If our hypothesis requires analyzing log files from the Internet of Things, or customer activity on social media sites, data sources can be varied and the volumes very challenging, if not impossible, using standard BI techniques. Therefore, our data science toolset includes advanced mathematical routines, as well as special technologies for moving and preparing data.

We see new role emerging: the data engineer.

The workflow of a data scientist as described above is also significantly different from other roles, for example a database developer or an app developer. Too often, this leads to confusion or even confrontation when putting a model into production with IT support. To IT, the cleaning and formatting done during data acquisition and preparation can seem haphazard and makeshift. You'll see a lot of cut-and-paste into spreadsheets, numerous Python scripts, and sometimes quite sophisticated linear algebra. This can sit uneasily with enterprise requirements of governance, audibility, security, availability, and scale.

As a result, in modern enterprises, we see new role emerging: the data engineer.

For the data engineer: the process



We have seen that a data scientist's skill set is wide and demanding and with its emphasis on mathematics and statistics it sounds hard. Compared to this, the title data engineer perhaps has a less glamorous ring to it: a practical Scottie to the mathematical Mr. Spock. Yet for all their specialized knowledge, there is much to be done in any enterprise application of their work which is not in their responsibility or their skillset. Increasingly, this is the very domain of data engineering.

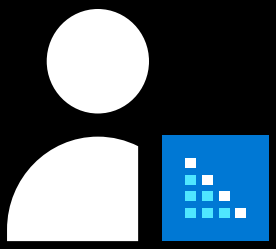
For example, it may not be appropriate to give a data scientist direct access to production systems for their research and experimentation. Although they likely want raw data, there can still be numerous business rules regarding confidentiality and regulatory compliance—especially in these days of The General Data Protection Regulation (GDPR) and other privacy legislation—that must be applied to data before sharing it for a given purpose. Don't expect a data scientist to know the regulations, external or internal, with the rigor required, even though these regulations are themselves often an answer and a reaction to the work of data science. It is the data engineer who can craft the necessary data sets and make them available with correct permissions. Nevertheless, the work of the data engineer at this phase is not completed when extracts are delivered. There will be numerous issues found, no doubt, and they may be required to help with everything from importing reference data sets to investigating missing values.

Similarly, when it comes to putting models into production, don't ask the data scientist to estimate data storage requirements, or costs for processing data streams, likely cloud compute expenses, or the solution's scalability. Getting to the bottom of those issues requires a good grasp of how a model will be deployed, used and supported in production. That is the data engineer's job.

Data scientists build robust data pipelines and deployments which can run models in production under demanding conditions.

But let's make it clear: the data engineer certainly knows about machine learning and coding. They can make sense of algorithms and scripts, but perhaps without a state-of-the-art statistical or mathematical understanding of models and experimentation. What they do have is the ability to build robust data pipelines and deployments which can run models in production under demanding conditions. Their work is rooted in a deep, and often hard-won, knowledge of how modern software is deployed and administered.

For the data warehouse: the model



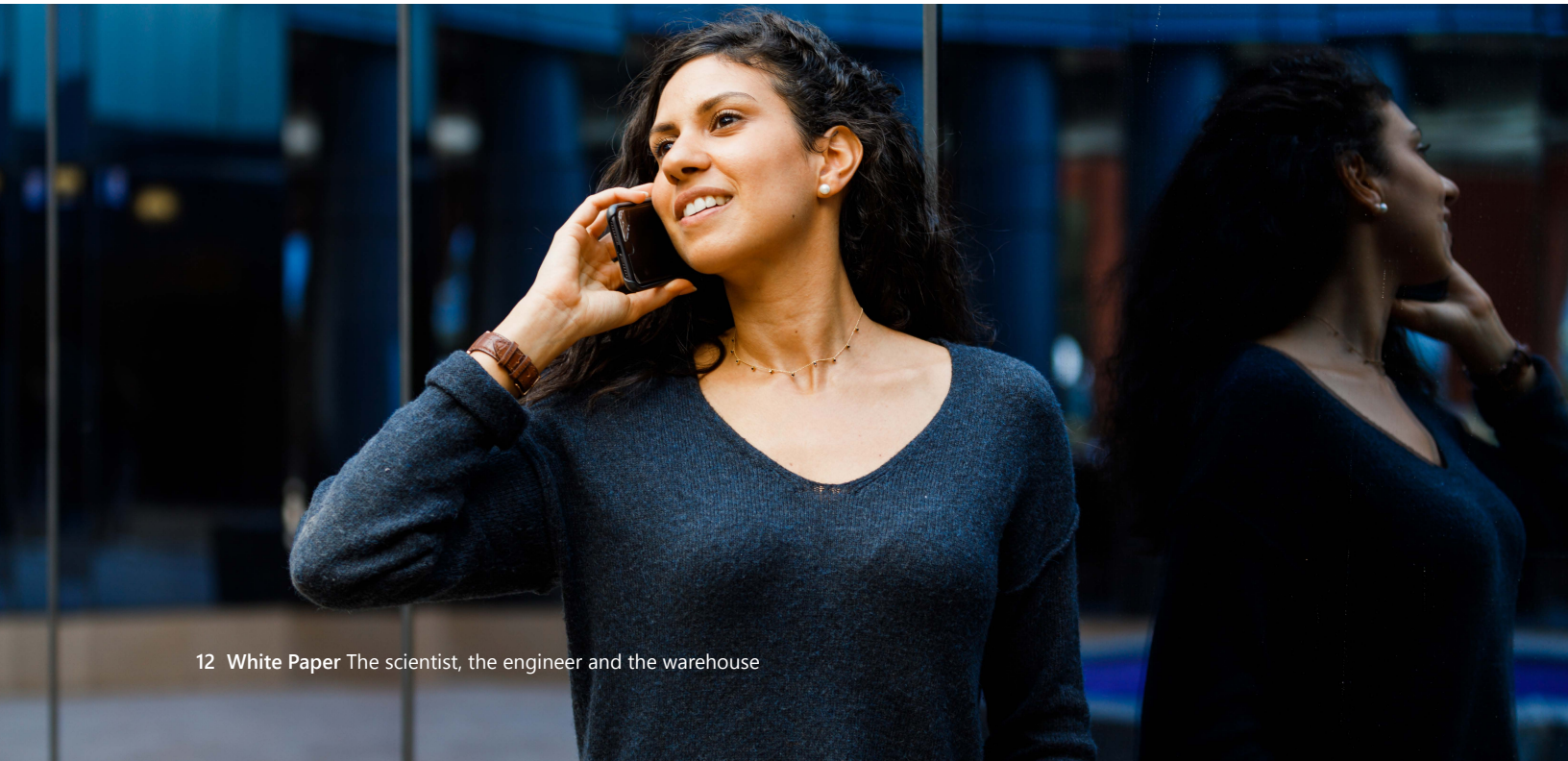
The model schema is crucial because it represents how different data sources and operational systems are related to each and it also captures any business rules involved. For example, it is from the enterprise data model that we know stores in Frankfurt are in Germany for tax purposes, but in the DACH region as far as marketing are concerned; meanwhile staff who work there are overseen by the human resources team for EMEA. These relationships may not be present in un-integrated data captured from operational systems, but can be represented in our enterprise data model.

Yet modern companies also find that they want to store and manage that raw data, and not only for machine learning which, as we have said, looks beneath the standardized business model.

A data lake stores data—often vast amounts of data—in its natural state.

This is one reason that the data lake has been such a successful architectural innovation in recent years. A data lake stores data—often vast amounts—in its natural state. This data may be messy and unstructured but provides raw material for data science.

While the Data Scientist may work with raw data in a data lake, report designers and BI users are much more likely to work with data which has already been cleaned and modeled to reflect enterprise demands and standards. Nevertheless, the modern data warehouse can be a useful source of knowledge and data. See the sidebar: *The data warehouse as a source for data science*.



The data warehouse as a source for data science

As in any data analysis project, identifying and accessing data sources is fundamental. Mostly data scientists like to work with raw data, from log files and production systems, but the modern data warehouse can be useful at this phase in two ways.

Firstly, the enterprise data model itself represents a specific understanding of commercial operations. A star schema is not just data. The modeling of facts reflects how an organization measures its activities. The modeling of dimensions reflects how management sees a firm's internal structure and its relationships to clients, suppliers, and other entities. In other words, a smart data scientist can learn about the business, including its blind spots, from the data warehouse.

Secondly, the data warehouse serves data that has been through a process of consolidation and cleaning, particularly when it comes to dimension data. Names of departments, geographies, job titles, and every other dimension have been reconciled and agreed upon. Although the data scientist will mostly wish to use raw numeric or text data for analysis, this authoritative source of master data can help them to ensure that their model is readily applicable in business terms.

The data warehouse serves models

Data warehouse integration is critical. A smart data scientist can learn about the business, including its blind spots, from the data warehouse.

As we have seen, the data warehouse serves data structured in an enterprise data model, but it can serve and support other models too. Business analysts working with self-service applications such as Microsoft Power BI, often create their own models which may be temporary, or limited, to their department to use. For example, a marketing team may create a model for a summer campaign which will only be used for a few months and integrates data from Microsoft Excel spreadsheets, including some ad-hoc calculations. However, when running the campaign, that data may also integrate with customer or store data from an Azure Cloud Data Warehouse. The campaign model may live only in Power BI or in Microsoft Analysis Services, but data warehouse integration is critical for its consistency, governance, and scalability.

It is also useful for the modern data warehouse to serve predictive models developed through machine learning. For example, a customer churn model may be developed through experiments which can predict the likelihood that a customer will abandon your service within, for example, 60 days. Running the algorithm, perhaps a logistic regression or a decision tree, generates a score for each customer; that score can be stored in the data warehouse alongside other attributes. Now our data scientist's work is readily available to any business analyst or call center operator, with their regular tools such as Microsoft Power BI or Microsoft Dynamics, and with great scalability and governance, through the enterprise data warehouse.

To see better how data scientists and data engineers work in this environment, it will be useful to examine their workflow and tools in more detail.

The cloud data warehouse

The data scientists have often been described as someone who work with very large volumes of data; they are not the only ones. We also see that the volume of data handled by data warehouses is increasing dramatically. We find this especially when data about facts—measures of the business—includes, for example, log files from the Internet of Things, web activity, or mobile applications, all of which can generate millions of new data points each day. It is also true that significant quantities of data today are generated in the cloud, by Software as a Service (SaaS) applications. These large data sets can challenge IT teams who try to store and administer them on premises. The IT budget must then support all hardware, tools, and administration time required in order to serve the data with high availability and good governance.

Dynamic elasticity is a key technical advantage of data warehousing in the cloud.

The cloud data warehouse has developed in recent years as an effective answer to such problems. The Azure Synapse Analytics is, in industry jargon, elastic. That is, it scales up or down as needed without additional effort, planning, or locking in a hardware investment from your IT teams. Dynamic elasticity is a key technical advantage of data warehousing in the cloud, but its financial benefits are compelling too.

Another advantage of the cloud data warehouse is the simplicity and speed of deploying models and applications. The design of analytic systems is iterative as models evolve over time and as new data sources are added. This is a necessary criterion for many teams who must choose when to use on-premises and/or a cloud solution.

In the past, the security of cloud services was a common focus of anxiety for some IT departments.

In the past, the security of cloud services was a common focus of anxiety for some IT departments. However, attitudes are rapidly changing as it becomes clear that cloud service providers can develop or deploy new security technologies more rapidly than in-house teams. This ensures that your cloud data warehouse is continually updated using best practices. You can read more about security best practices here: Seven Key Principles of Cloud Security and Privacy.

Finally, in today's global economy, most IT departments find it easier to serve a worldwide workforce with consistent data and models using a data warehouse in the cloud. Azure Synapse Analytics ensures the replication, consistency, and availability of your services with less effort on your part.

The architecture of data science in the organization

The following diagram shows a high-level view of an effective architecture for data science at scale, with the modern data warehouse at its heart.

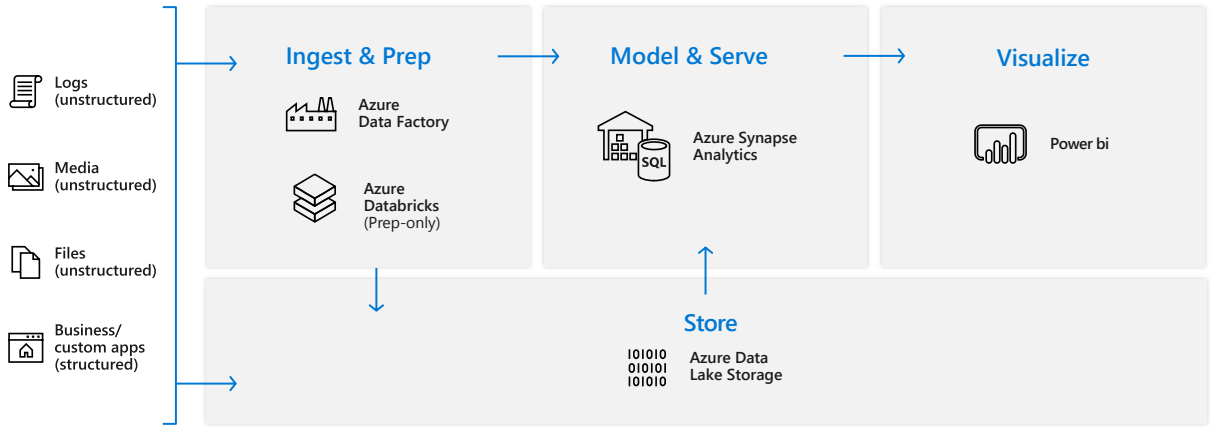


Figure 1: The modern data warehouse and data science process

Data ingestion and storage

Having discussed the need for raw data, you will understand how foundational this phase is to an enterprise data science process. The aim is to bring together structured data from business applications, along with unstructured and semi-structured data from logs and media, using on-premises and cloud sources.

You may have to deal with transforming data types between different formats, handling missing values, filtering outliers, and perhaps performing aggregations.

Frequently, the data must be prepared first. Indeed, practitioners sometimes refer to this phase as data wrangling because the tasks can be quite complex and time-consuming. For example, you may have to deal with transforming data types between different formats, handling missing values, filtering outliers, and perhaps performing aggregations.

Azure Data Factory

Azure Data Factory enables you to create, schedule and orchestrate data transformations with SQL Server Integration Services, bulk copying of data, and Python scripts. And, as previously described, the Azure Cloud Data Warehouse can also play a useful role here, serving reference data where needed.

If the project at hand is simply for research, this work may well be done by a data scientist alone. However, if their research comes up with compelling results which could be useful in production, this phase will need to be revisited by the data engineer and data scientist together, to deliver a process robust enough for enterprise use.

Often enough, the project definition is not just research, but includes a long-term goal of delivering a production-ready process. In such cases, you need at least regular reviews between data science and data engineering to identify potential issues and alternative approaches early enough to be useful.

Azure Data Lake

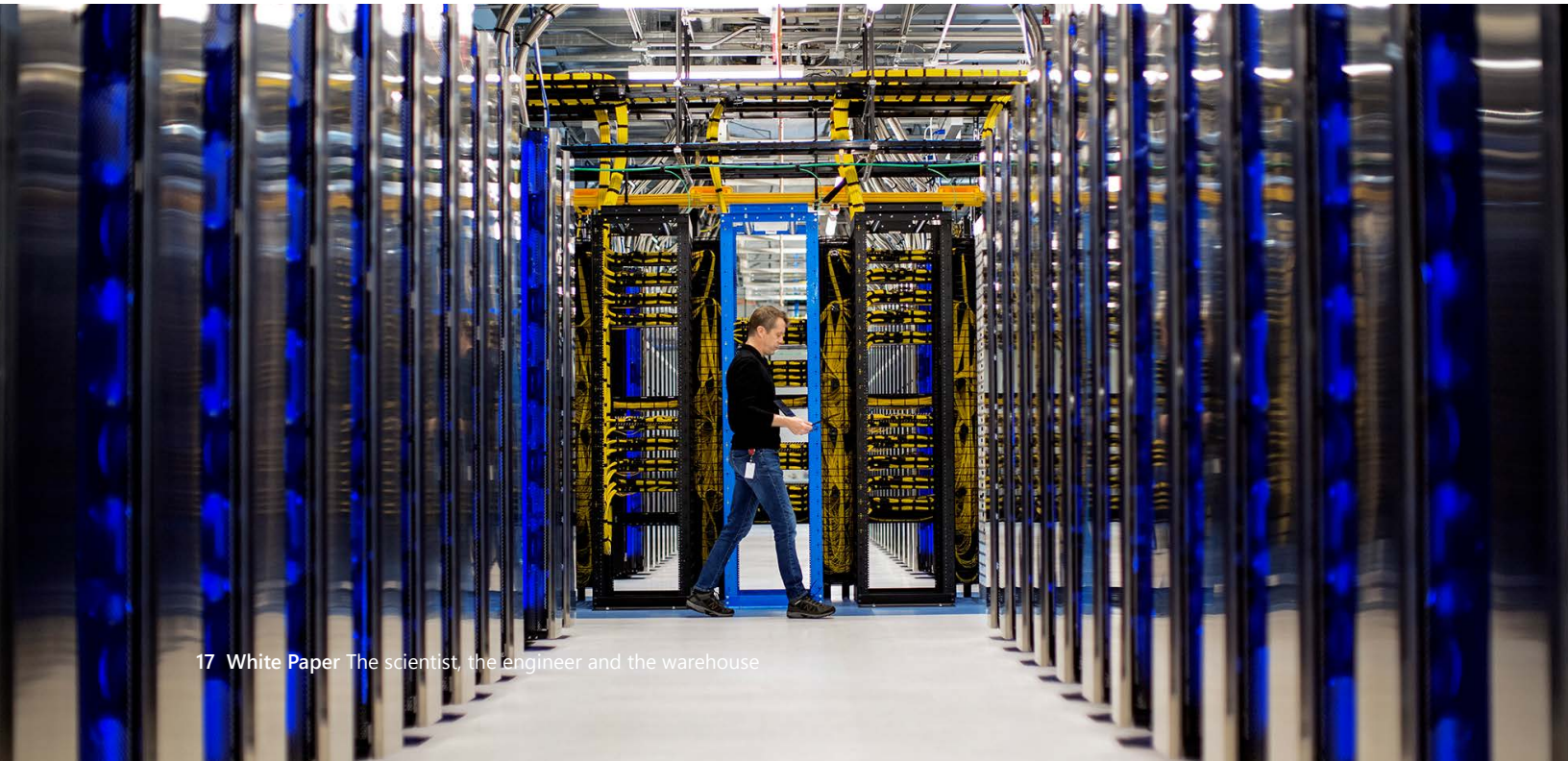
It is quite usual to find data sourced from one zone, prepared in some way and perhaps integrated with external data, and then landed in another zone for further use.

Although this paper focuses on the role of an Azure Synapse Analytics, for most scenarios involving advanced analytics, data sourced and prepared with the Azure Data Factory will land in Azure Data Lake storage. This is true, even for data which has itself been sourced from a data lake. In fact, most enterprise data lakes have several zones, including:

A landing zone where raw data is ingested and stored. This zone acts as a data source for both data scientists and data engineers to work with data in its raw form.

A development zone where data scientists and engineers, and sometimes applications developers, can store their prepared working data.

A trusted zone where users who work with visualization tools or business intelligence apps (Power BI for example) that can access unique sources such as social media data that has been collected and curated by IT for their use. It is quite usual to find data sourced from one zone, prepared in some way and perhaps integrated with external data, and then landed in another zone for further use.



Data preparation and training

Data preparation and exploration can be a time-consuming task. The art of analysis—rather than the science—lies in knowing from experience, and sometimes a hunch of intuition, what to look for in data and where to look for.

Some data exploration is performed visually but scripts are also used to analyze correlations between values, to discover outliers and to understand distributions of data values.

It is worth pointing out that at this stage, visualization does not necessarily mean creating charts in Power BI or other business intelligence tools. Most often the data scientist will visualize these raw data sets during preparation using a visualization script such as Matplotlib or Plotly in Python. BI tools typically come later in the process when presenting data to less specialized users. In contrast, our emphasis in this phase of data science is on exploration and finding patterns rather than communicating findings.

Having explored your data, selecting which features are most likely to be significant, and labelling your data, you are now ready to build a model. To do so, you must choose a machine learning technique, such as a specific algorithm.

Automated machine learning also enables you to run many more experiments and can help to ensure that interesting possibilities are not overlooked.

Increasingly, faced with the necessary slowness and meandering paths of manual data preparation, even experienced data scientists are looking to automated machine learning for help. With automation we can quickly develop many models, with multiple algorithms, and numerous variations in how algorithms look at the data. The result is not just one model crafted by a specialist (with their own inherent biases and preferences) but a wide range of models generated by the system itself, which you can compare and select from based on your criteria for a successful project. The result is generally higher accuracy while spending less time experimenting. Automated machine learning also enables you to run many more experiments and can help to ensure that interesting possibilities are not overlooked.

One key advantage of the Azure Machine Learning service is that once you have built a good model, you can easily use it in a web service or from a business intelligence tool such as Power BI.

Azure Machine Learning service

The Azure Machine Learning service is a cloud platform used by data scientists to develop and automate machine learning models using a variety of resources or environments. These resources—compute targets—may be local machines or cloud resources, and often Azure Databricks.

Machine learning automation is supported with a simple workflow that involves selecting data, configuring the compute target, and setting parameters such as how many iterations to run and what metrics to look at to determine the best model.

One key advantage of the Azure Machine Learning service is that once you have built a good model, you can easily use it in a web service or from a business intelligence tool such as Power BI.

Azure Databricks

Azure Databricks is an Apache Spark-based analytics platform widely used in machine learning for exploration and modeling. It enables the data scientist and data engineer to write code in data science notebooks using Java, Python, R, Scala, or SQL while also leveraging the power of distributed processing with automated cluster management.

Databricks environments and clusters can be created and managed automatically with Azure Resource Manager templates and PowerShell scripts. Azure Databricks also supports various types of visualizations out of the box using the display function.

It is useful to remember that the work done here by a data scientist will, in general, center on supporting their experiments. It is too early to be thinking about scalability or resilience. But, if the experiment results in a compelling discovery for business, it will fall to the data engineer to deploy the project. In that case, this work does not necessarily need to be rebuilt. Azure Data Factory has connectors which enables a data engineer to trigger the running of Azure Databricks notebooks in a pipeline. They can assign compute power to scale the execution of notebooks on a cluster. With this capability the engineer can run notebooks consistently for the enterprise, orchestrated with other processes.

Serving and presenting models

The traditional role for an enterprise data warehouse is to serve a canonical model of the business: an approved view of all data necessary to report on and analyze your processes. This model will integrate many data sources from numerous divisions of the organization. The task of integrating and conforming this data, which can involve numerous ETL jobs and staging areas, along with error recovery and scheduling, is more complex than Power BI is designed to handle. Azure Synapse Analytics serves enterprise data models for all scenarios, including as a data source for business intelligence and Power BI. Business intelligence models today are typically built by users themselves, using tools such as Power BI. Therefore, we commonly call these applications self-service business intelligence. A user connects to data sources which may be simple, or which may require some integration. But primarily they do this work themselves in their own toolset. Nevertheless, the cloud data warehouse has a fundamental role to play in well-governed self-service BI.

You have the simplicity and ease-of-use with Power BI for business users, but the exceptional scalability of a cloud data warehouse at the back end.

With Power BI, when you connect to your data source, it is common to import a copy of the data where you can then work on it. While this is possible when connecting to Azure Synapse Analytics, there is more for when you wish to create dynamic reports based on your Azure Synapse Analytics model. This can be especially useful for near-real-time scenarios such as reporting over data from the Internet of Things or online commerce. For such scenarios, which we'll describe more when talking about streaming data in Azure Databricks, rather than importing data to Power BI, you can connect directly to the data source using DirectQuery. With DirectQuery, queries are sent back to your Azure Synapse Analytics in real time as you explore the data. In this way, you have simplicity and ease-of-use with Power BI for business users, but the exceptional scalability of a cloud data warehouse at the back end.

As these business users are not data scientists, we need to find a way to provide them with predictive insights.

The result of a data science project will often be a predictive model that can be put into practical production. For example, a model which identifies groups of customers by their shared characteristics can be useful for sales, marketing, and product support to help recognize where a new customer will fit into existing commercial patterns and how to better serve them. As these business users are not data scientists, we need to find a way to provide them with predictive insights through their existing tools, such as Power BI, which mostly connect to data through either BI models, or through the modern data warehouse.

So now we should look at how to integrate machine learning models with this classic architecture.

Serving data from Azure Databricks with Azure Synapse Analytics

Azure Synapse Analytics has several significant roles to play in the enterprise machine learning process.

We have mentioned previously that it can be a useful source of reference or master data for some scenarios. But its real power comes into play when serving models and data for analysis by the business users.

If you have used Azure Databricks for data preparation, transformation and cleansing, the resulting data set may be useful not only for machine learning, but for other operational or reporting use cases. For example, a dataset used to analyze sales of products to customers with a view to optimizing special offers, can also be integrated with CRM data to create a compelling single view of each customer and their activity. That new combined data set could be used with visual tools like Power BI.

To enable this, you can load large volumes of data from Azure Data Bricks directly into Azure Synapse Analytics using a specialized and highly efficient Synapse Analytics connector. This connector uses Azure Blob Storage, and PolyBase (a Microsoft data virtualization technology) in Synapse Analytics to transfer large volumes of data between a Databricks cluster and a Synapse Analytics instance. In some scenarios, where source data is streaming from a constantly updated system—often the case in online retail, for example, or from the Internet of Things—you can directly stream data into Azure Synapse Analytics using Structured Streams. This enables business users to work with near-real-time data in Azure Synapse Analytics.

Serving machine learning models with Azure Synapse Analytics

As previously stated, the cloud data warehouse can usefully serve results of machine learning to business users. This is particularly the case where an algorithm generates scores.

For example, when executing a notebook, Azure Databricks uses a trained Spark MLlib model to generate predictions for the observations in the scoring data set. These predictions are stored in the results store, which is a new data set on the Databricks data store. These results can be loaded into Azure Synapse Analytics using the connector described above.

With the Azure Machine Learning service, you can use the Score Model module to generate predictions using a trained classification or regression model. The module's scored dataset output can then be loaded into Azure Synapse Analytics.

It's worth noting that different models generate different kinds of scores. For classification models, Score Model outputs a predicted value for the class, as well as the probability of the predicted value. For regression models, Score Model only generates the predicted numeric value. For image classification models, the score might be the class of object in the image, or a Boolean value indicating whether a specific feature was found.

The modern data warehouse is an exciting source of data for recommendation systems as it likely already contains cleansed and prepared data for users, items, and the sales or selections that connect them.

Some other types of models generate their own kinds of output, rather than scores, but the results can be loaded into Azure Synapse Analytics and used in similar ways.

A recommendation system points out one or more items to users of the system. Well known examples of items include movies, restaurants, books, or songs. Most people have been users of such systems. But users can also be a group of people, or some other entity which has item preferences. The modern data warehouse is an exciting source of data for recommendation systems as it likely already contains cleansed and prepared data for users, items, and the sales or selections that connect them. Similarly, when you generate results, it can be very effective to save these new data points back to your warehouse schema, where they can be readily used (and easily understood) by business analysts, sales teams, and call center operators.

Azure Machine Learning features the Matchbox recommender which is a sophisticated system combining two different approaches in an automated hybrid technique. You can find more details of the research behind this fascinating algorithm here: Matchbox: Large Scale Bayesian Recommendations.

The Matchbox recommender generates different datasets depending on the scenario you choose:

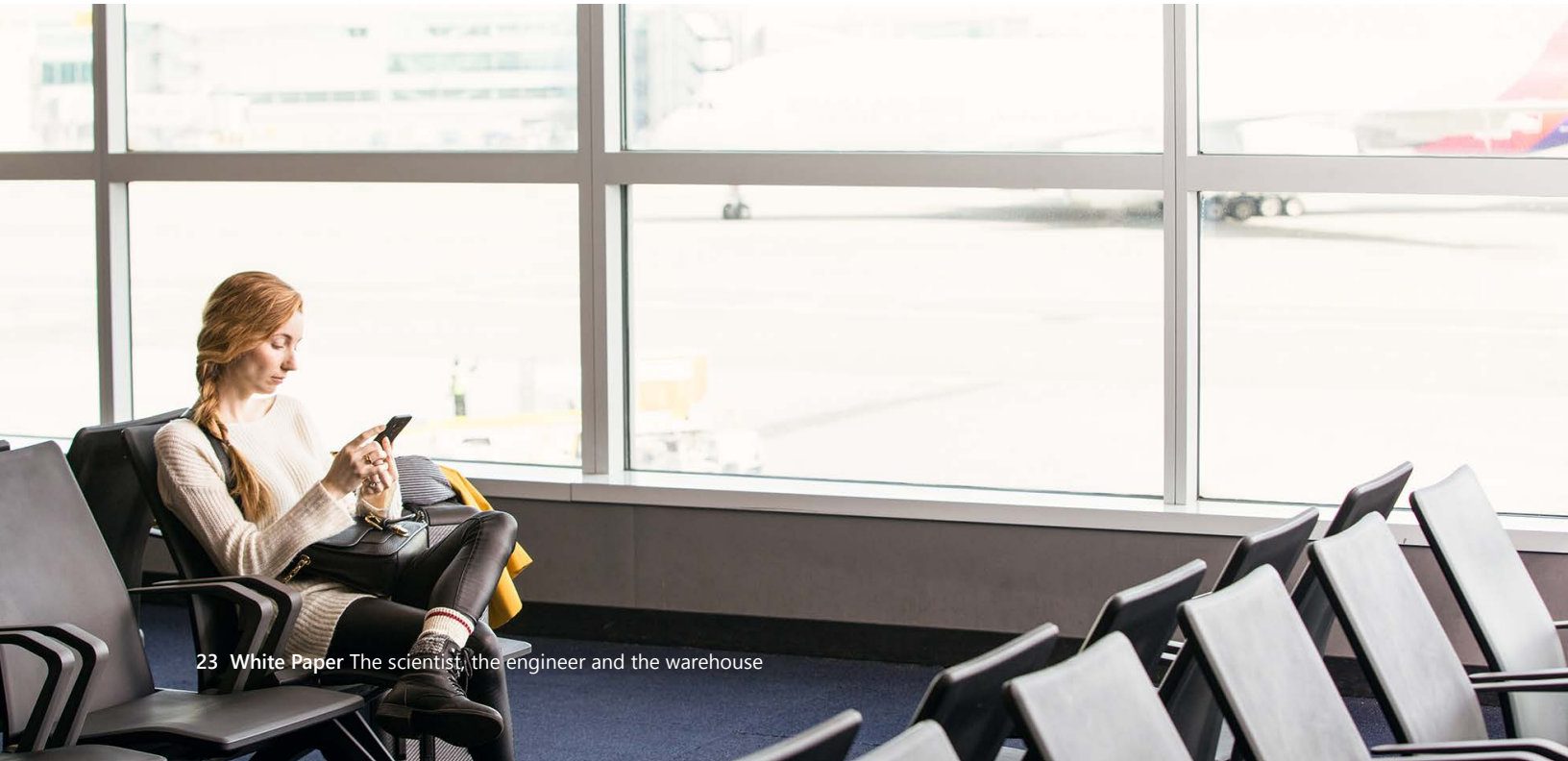
Predict ratings outputs three columns, containing the user, the item, and the predicted rating for each input user and item.

Recommend items rreturns a dataset listing the recommended items for each user, depending on how many recommendations you request.

Find related users lists the users who are related to each user in the input dataset, depending on how many results you request.

Find related items similarly outputs the items related to each item in the source data.

All these data sets can be loaded easily into the Azure Synapse Analytics.



Another commonly used type of model is clustering. Clustering models group data points into similar clusters. Sometimes you will hear this called segmentation.

Clustering algorithms use features of individual items to find similar items. For example, items in a shipping warehouse may be clustered by size, weight, and destination. People may be clustered by demographic properties.

Azure Machine Learning includes an Assign to Clusters module which can be used with a trained clustering model. This module returns a dataset containing the cluster assignments for each case, and a distance metric that gives you some indication of how close this case is to the center of the cluster.

[Clustering] helps to find interesting patterns in data and is often used for exploration of data prior to analysis with other algorithms.

Clustering is a little different from other techniques described, in that it is most useful at the exploration stage. It helps to find interesting patterns in data and is often used for exploration of data prior to analysis with other algorithms.

Nevertheless, there are some use cases, notably in Customer Relationship Management, where clustering algorithms may be used to segment people into, for example, Committed Customers, Casual Customers, and Non-Customers. It can be significant for some customer analytics to be aware if someone is moving from one cluster to another, because that can be meaningful for their relationship with the firm. For these purposes, loading the result set back into Azure Synapse Analytics is useful.

Conclusion

The aim of this whitepaper has been to reflect on the emergence of data engineers in the analytic enterprise, and to underline the importance of a cloud data warehouse to support that role. These two aims are significantly related.

The data engineering role has emerged because businesses must put data science into practice with enterprise-class governance, scalability, and resilience.

Firstly, the data engineering role has emerged because businesses must put data science into practice with enterprise-class governance, scalability, and resilience. It is too much to ask the specialist data scientist to take on those requirements in addition to their experimentation and research, thus the data engineer steps in.

Alongside the emerging role of the data engineer, the cloud data warehouse has also developed, with technological goals which mirror the concerns of a data engineer. The cloud data warehouse enables IT to serve these diverse manifestations of business models (data models, BI models, and machine learning insights) all with enterprise-class governance and manageability as well as with the scalability and elasticity we expect from the cloud.

Microsoft's Azure Synapse Analytics is uniquely suitable for this new landscape, thanks to its deep integration with other elements of the Azure platform, its enterprise capabilities rooted in a market-leading database platform, and its exceptional performance and security.

The smaller organization

In this paper, we've looked at the newly emerging role of data engineer, working alongside the data science team to put advanced analytic models into production. In particular, the emphasis has been on the importance of being able to run such models with enterprise-class scalability and resilience.

However, many businesses don't have the resources to employ both a data scientist and a data engineer. Yet, in these days of big data, globalization, and online commerce, even modestly-sized teams may be handling challenging issues. In such cases, there are four key recommendations which will enable you to address the same concerns that we've discussed in this whitepaper.

Firstly, train existing IT team members in the basics of machine learning. They do not need to become experts, but the more familiar an IT team becomes with the data science methodology, how models work, and the data that is required, the more they can effectively support machine learning. There are numerous online courses available, reasonably priced and often free, which cover the fundamentals, and can even be quite advanced.

Secondly, the data scientist will need to accept that they have quite a lot of work to do, putting models into production. This work will require close co-operation with IT to ensure that data pipelines, scripts, notebooks, and so on are ready for the enterprise. The data scientist will also need to learn about governance, compliance, and security needs of the business. This white paper, Seven Key Principles of Cloud Security and Privacy, is a good place to start.

Thirdly, the data scientist (singular) must take on some data engineering work. That data scientist will be your first hire. But if you are serious about expanding your machine learning footprint—and that is highly likely as your work progresses—your second hire should be a data engineer, not another data scientist.

Finally, your choice of tools and platforms will be central to your success. The Azure Synapse Analytics and Azure Machine Learning Service, with Power BI at the front end, are simple to deploy and maintain, while scaling and growing with your business very effectively. There is no better platform on which to start a data science practice.