DATAOPS

The Authoritative Edition

JOHN G **SCHMIDT**

KIRIT **BASU**

DataOps

The Authoritative Edition

John G. Schmidt
and Kirit Basu

This edition first published 2019 by Panther Publishing, 3102 Rexford Dr., Austin, TX 78723, USA

For more information about Streamsets please visit www.streamsets.com.

Streamsets Headquarters
150 Spear Street Suite 300, San Francisco, CA 94105, USA

Library of Congress Cataloging-in-Publication Data

ISBN 978-0-980-21694-3

Cover design by Kris Hackerott

Set in size of 11pt and Minion Pro by Manila Typesetting Company, Makati, Philippines

Printed in the USA

10 9 8 7 6 5 4 3 2 1

We gratefully acknowledge the following people for valuable contributions to this book: Girish Pancha, Arvind Prabhakar, Sean Anderson, Judy Ko, Melissa Gillard, Pat Patterson, and Rick Bilodeau.

Contents

About the Book

DataOps is an innovative breakthrough that lets people use data as easily as they plan a trip around the world or buy a pair of shoes on the web. DataOps is the practice of operationalizing data movement to improve quality and accelerate delivery for new business demands for data, and to deliver continuously with confidence, in a world of ceaseless change. With DataOps you operate data rather than engineer data. In other words, it creates continuous data flows with automated processes and self-service tools so that users can discover and deliver data by themselves in a few days or hours.

Compare that to the traditional method of launching an IT project, which brings together data architects, application developers, infrastructure technical architects, software engineers, testers, and others all coordinated by a Project Management team to orchestrate a complex symphony of efforts to create a one-time data flow usally in months. And the following month, when one of the data sources changes or the company moves an in-house application to the cloud or a new web app needs to be added, the data flow is broken and the symphony must be replayed.

This is the first book on the full range of advice to create and optimize the DataOps capabilities for any enterprise. Topics include:

- Defining the Data Management Breakthrough!
- The astonishing speed of DataOps practices
- DataOps Playbook for responsibilities, services & practices
- Starting DataOps, gaining executive support and overcoming resistance
- Institutionalizing the DataOps culture and continually improving
- Teaching about Continuous Design, Continuous Operations, Continuous Governance, Continuous Data and Program Execution
- Building and using DataOps systems and technologies
- Measuring progress and Business Value

Executive Foreword (by Girish Pancha)

I spent two decades leading efforts at Oracle and Informatica to develop successful and innovative products for large enterprises. The main focus of my efforts was to address the challenge of providing integrated information as a mission-critical, enterprise-grade solution to support business intelligence. For the past two decades the best in class technologies and methodologies have focused on the ability to deploy dashboards and reports at scale. But there is a tidal wave of change when it comes to the types of data being analyzed, the platforms and cloud infrastructures where data is analyzed, the speed at which data needs to be analyzed, and the applications through which they are consumed.

The solution to all this accelerating change in the data analytics landscape is the emerging field of DataOps. First coined a few years ago by Andy Palmer at Tamr, it's now on the hype cycles of prominent industry analysts, and on the tongues of data vendors and leaders in large enterprises.

I have had the privilege of working with both John Schmidt and Kirit Basu over my career. John wrote the first 2 definitive books in this space around organizing teams and implementing processes to create a factory like approach to data integration. Kirit is the founding VP Product at StreamSets where he has been driving the technology vision on modernizing data integration in support of DataOps since 2015. They have collaborated to develop a comprehensive and coherent view of the people, processes and products across the data analytics ecosystem that's needed to implement a DataOps discipline. The end result is that businesses can operate at the speed of need, and with confidence, when embarking on their next generation of digital transformation.

Preface

DataOps is one of three technology breakthroughs that allow people to use data as easily as they plan a trip around the world or buy a pair of shoes on the web. Before I explain, let's first step back.

Before the internet, the effort to buy a pair of shoes from home involved waiting for a catalog to be delivered (typically quarterly), calling the company to describe what you want and waiting for it to be delivered. The process of paying was even longer, since it involved receiving a paper invoice by mail, mailing a paper check, the company moving documents around internal departments until it all showed up at Accounts Receivables, and eventually posting the payment at the bank, which was also a paper-based process. The typical time from a customer's order to cash-in-hand was several months; not to mention the time involved in printing and delivering the catalog.

In the modern internet age, the order-to-cash process is just a few seconds! There were basically three technology breakthroughs that enabled this dramatic result; the internet protocol, the browser (universal user interface) and online electronic payments. Other developments like the digitization of catalogs, Amazon's 1-click order process, and delivery tracking on cell phones helped, but it was the three inventions of the internet, browser and online payments that made the breakthrough a reality.

I've been wondering for years when IT professionals would invent the same capabilities for finding, delivering and using data as quickly as we can buy a pair of Nike shoes. What three innovations will emerge to make data easy enough for your mother to use, rather than a pain? We don't yet have all three inventions for data, but we have two of them!

The first breakthrough was big data, which emerged around 2005. Big data enabled advanced analytics by making it possible to gather massive amounts of data in any format, without the strict rules of historical database structure, and store it in a distributed compute platform running on commodity hardware and open-source software, at less cost than traditional

data warehouses. The first challenge of managing the exponential growth in data scale, variety and speed has been solved.

The second breakthrough is DataOps, which was first mentioned in 2014. It is the alignment of people, automated technologies, and business agility to enable an *order of magnitude improvement* in the quality and reduced cycle time of data analytics. *DataOps expedites the flow of data* for effective operations on both traditional and big data, by leveraging self-service capabilities to bypass traditional methods of engineering customized programs. DataOps has demonstrated its capabilities and effectiveness in multiple industries on a global basis to give us the confidence to label this book as **The Authoritative Edition of DataOps,** building on the first edition in early 2019.

The third technology breakthrough has not yet arrived, but elements are starting to emerge. The idea is to make it easy to find and assess the value of the wide variety of information in today's complex data landscape using sophisticated tools such as natural language metadata, automatic classification of data in written or spoken form, artificial intelligence to connect with real-world objects or processes, showing data as 3D holograms, and so on. But let's return to DataOps, which is the latest innovation!

John Schmidt

1

DataOps – A Data Management Breakthrough!

DataOps is not named randomly. It builds on the use of DevOps, which is a widely adopted and well-understood practice that accelerates software development by leveraging automation and monitoring to enable agile collaboration across application designers, operations staff and business users. While DataOps does have some similarities to DevOps, it is a more comprehensive capability and the comparison downplays its significance. DataOps is a paradigm shift: It is a fundamental change in the basic concepts and practices of data delivery and completely challenges the usual and accepted way of integrating data.

In today's world, there isn't a single IT organization that can control and engineer all aspects of the data that their enterprise needs. Data is now the connective tissue upon which complex enterprise logic is being built, spanning numerous applications. DataOps enables faster delivery of existing and new data services and products in the face of changing environments, requirements, infrastructure and semantics while preventing data threats. DataOps enables applications to be good citizens of the ecosystem by respecting the implied contracts between them despite unexpected data drift that emerges from changing technologies.

For a current example of a paradigm shift, consider what Tesla has realized. Cars have been around for 100 years and car technology has obviously evolved since the Model T; then Tesla came along. Recently, one of the authors drove a Tesla from Florida to Toronto. In the 3,000-mile round-trip journey, we didn't buy any gas, the electricity was free, the car accelerated faster than any others on the road, the "engine" was silent, and it drove itself most of the time.

Furthermore, you never need to add oil or antifreeze (it doesn't need either) and it doesn't need any maintenance other than filling the tires with air and adding windshield wiper fluid. In some cases, you can enhance the performance of your car in your driveway with an online payment; the

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (1–12)

60KW Tesla S can be upgraded to a 75KW battery through the car's computer screen. This is possible because the car already had the larger battery installed at the factory and it just needed to be activated by a software license. Tesla is more of a computer with wheels than a car. In short, a paradigm shift.

In the same way that Tesla has flipped the polluting, high-maintenance, manually controlled automobile into a clean, friction-free, self-driving computer, **DataOps transforms stodgy, centralized Business Intelligence "dashboards and reports" into real-time and democratized analytics capability that unlocks the huge potential of all your data. DataOps transforms the traditional approach of designing and building custom data movement software into self-service capabilities that people simply operate.** That looks like a paradigm shift!

1.1 Why Care About DataOps?

> ***"The world as we have created it is a process of our thinking. It cannot be changed without changing our thinking."***
>
> **— Albert Einstein**

DataOps is an innovative and powerful capability in the world of data management. It is a rather new practice being adopted by leading organizations, but it is already becoming the default method for managing data needed by enterprises to support their digital transformation programs and modern data analytics practices.

As per Einstein's words, to take advantage of this capability, you need to change your thinking. The very needs of analytics and business intelligence have changed in the context of digital transformation. Technologies like Business Intelligence and Data Warehousing serve for descriptive and diagnostic analytics, but *predictive and prescriptive analytics have changed the game*. As the hardware landscape becomes more capable (even the smallest of devices are able to run neural networks and machine learning algorithms) decisions can be made on that data at the time of creation while it is in motion; and certainly when it's back in the realm of centralized teams in the cloud or in datacenters. Analytics has become pervasive.

Leaders across all industries have realized that their long-term viability and organizational existence in many cases depend on the ability to do a LOT more with data. New roles like the Chief Data Officer are exploring forward-looking analytics, smart products and services and perhaps even

productizing data in support of a completely innovative business model. Data is no longer an intangible asset; it is a vital pillar of corporate strategy and competitive differentiation.

This new world demands agility, both for developing data management practices and for operating them in the real, ever-changing world. DataOps is a new capability that aligns with this need for agility and changes to your methods. Specifically, stop developing brittle data integration software that has to be rebuilt whenever something changes and stop architecting complex data models that take months to create and can't keep up with the rapid changes in today's world.

> **DataOps:** an automated, process-oriented methodology, used by analytic and data teams, to improve the quality and reduce the cycle time of data analytics.

DataOps uses smart tools to discover data, detect changes, direct it where it is needed, and monitor operations in a much more automated fashion. In short, focus on the outcome of delivering value (driving the car) rather than building or worrying about perfecting every underlying component (customizing and tuning the engine). The diagram below summarizes why you should care about DataOps.

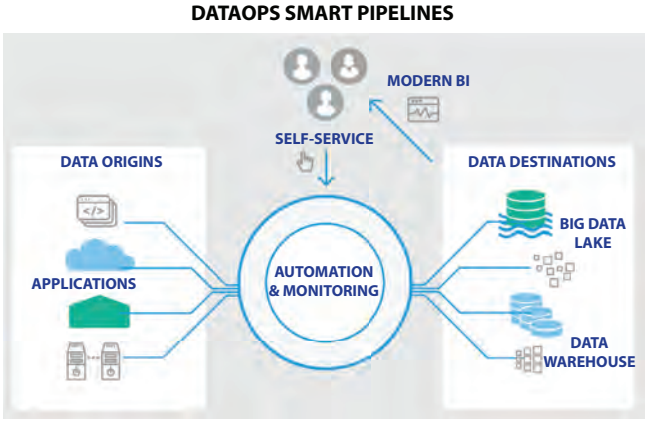

Figure 1.1 Simple, automated and self-service DataOps features.

This image is the traditional Integration Hairball. It shows data flows between a company's application systems—both in-house and in the cloud—through a complex series of integrations, each of which were custom built for a fixed data flow by different projects using a variety of technologies.

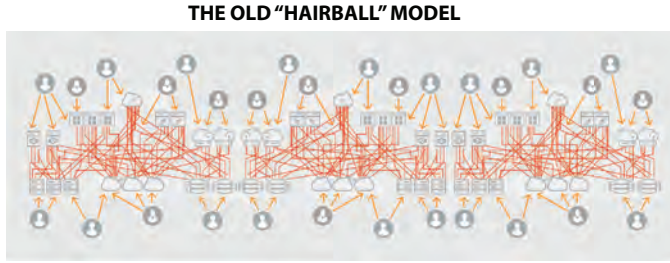

Figure 1.2 Traditional methods of acquiring and delivering data.

For example, at a *Fortune* 500 retailer, the ICC team implemented a metadata repository and a process to capture and maintain the integration points going forward. Within three years, they had 5,000 integration points captured in the repository. Seven years later they were up to 45,000 through a combination of new development plus documenting historical integrations.

The fact that they had an actual number is positive since it means they had standardized processes to capture integration data facts. If you were to compare this to other companies, you will find that only a few could give you an accurate number. The reality is that most organizations are building integration points ad hoc as needed to support individual projects, without a master plan or holistic view for all data flows. The result is high complexity which is consuming a huge portion of the budget. A rough estimate is that 20% of IT budgets are consumed by the Integration Hairball. An even larger problem is that the complexity of these custom integrations is adding up to 25% to every effort; three months of a one-year project is consumed just dealing with the hairball complexity.

No organization needs an Integration Hairball although it remains the reality in many enterprises. See 1.3 DataOps in a Nutshell to understand how DataOps eliminates the Hairball!

1.2 A Brief History of Data Evolution

To build on why you need to change your thinking, a brief history of computer technology, how priorities have changed, and why DataOps is critical is justified. Starting with the introduction of electronic computers in the 1950s, there are four areas of computer evolution: 1) hardware, 2) software, 3) internet, and 4) data.

1.2.1 Hardware Era

Hardware became interesting in the 1950s with the transistor to replace vacuum tubes, new magnetic core memory and creation of integrated circuits (IC). Solutions based on mainframe computers evolved to mini-computers, Personal Computers (PC), smartphones and more recently the Internet of Things (IoT) and even quantum processors. Technology continues to advance with no end in sight, but the point is that hardware is basically a commodity. Computer speed and power has increased, and hardware still demands management attention and investments even with dramatic reductions in cost. That said, software, the internet, cloud and data became the center of attention for innovation and competitive.

1.2.2 Software Era

The early computer hardware used first generation (machine level) and second generation (assembly) programming. It wasn't until third generation languages like Fortran, Cobol and Basic caught the attention of programmers in the 1960s that software began to demand efforts from IT organizations. Organizations could now write programs for airline reservations, bank account processing, accounting systems, and replace manual activities like connecting telephone calls and printing sales invoices.

By the 1990s, software companies, like Microsoft, Oracle, IBM, SAP, and eventually Google were computer leaders and by the 2000s, more than 80% of CIOs were promoting the principle of "buy rather than build". Big enterprises ended up with hundreds or thousands of applications (one bank I worked at had 18,000 applications), each hoarding its own siloed treasure trove of data, which kept their programmers busy developing data integrations. The software application fragmentation in turn generated a wave of software vendors specializing in integration tools and ultimately fueling the migration to cloud software and the need for DataOps.

1.2.3 Internet, Cloud & Mobile Era

Since the advent of the internet, the world has undergone a sea change thanks to the power of the World Wide Web. The internet is now influencing not just how individuals learn, work and communicate, but also how organizations use and manage computers and information technology. The cloud era is led by vendors who are *renting* software tools and complete application systems by the hour, minute, second, or even less, rather than selling them. The ease and exponential growth with which business units

acquire new cloud and mobile applications drives the need for DataOps due to the dramatic increase in the volume, variety and number of data sources. Traditional methods of extracting, transforming and integrating data can't adapt to the flood of internet-generated data.

1.2.4 Data Era

Data is now leading technological innovation more than hardware, software or the internet. During the earlier eras, data was combined with hardware and software and was basically an integral component of a system to serve a business function. But data now comes from more than just application systems—it also comes from IoT and similar devices (cell phones, digital cameras, health monitoring devices, home security systems, etc.) and a range of cloud applications.

There is more data than ever, coming from more places both inside and outside the organization, and changing constantly. This data provides tremendous opportunity to make better decisions, run faster, and even build entire new business models.

Data started to change from a by-product of business activities to a foundational driver for business innovation during the Information Economy, a concept introduced in *The Third Wave* by Alvin Toffler in 1980. Google, Facebook and other social media companies leverage this new paradigm, treating data as their product/service and the source of financial results. Other data-driven businesses include the world's biggest taxi service, Uber, which has no taxis, and the world's biggest hotel chain, Airbnb, which has no hotels. Being data-driven is not reserved for just digital-native companies but is incumbent on players in every industry—e.g., health care, education, manufacturing and government.

Data is creating a "virtual reality" and taking on a life of its own. The opportunity is tremendous, but it also can quickly become extraordinarily complex and challenging.

1.3 DataOps in a Nutshell

What does DataOps do that addresses the emerging era of data dominance, and how does it do it? DataOps expedites the onboarding of new and uncharted data and directs the data to effective operations within an enterprise and its partners, customers and stakeholders; and at the same time preventing data loss, operational breakages and security threats. Unlike traditional point solutions, DataOps uses "smart" capabilities of automation and monitoring:

1. Self-service tools for professionals to find, move, consolidate and annotate data.
2. Discovery of technical blueprint of data sources like structured relational stores, semi structured NoSQL stores and unstructured binary data.
3. Automated creation of data processing jobs without specifying schema and structure in advance.
4. Monitoring of data-in-motion including capturing operational events, timing and volume, generating reports and statistics.
5. Expedited handling of data errors and exceptions during data flow processing.
6. Discovering and automatically handling data source changes. For example, if a data source has a new schema with new fields since the last flow, automatically include the changes and continue flowing the data.

The monitoring capabilities of DataOps are critical, as they provide global visibility of the entire interconnected system, and notify operators of significant events, errors or deviations from the norm. Monitoring is especially important now because the data landscape is more fluid and continues to evolve dynamically. With more actors in the modern data supply chain, the data infrastructure is no longer a static plan that can be crafted once and executed; it is now a constantly emerging picture that shifts to align with business imperatives at clock speed.

The skills and competencies used in the past to manage and control the effective use of data are not sufficient. In many situations what was learned or practiced 10 years ago is worse than irrelevant; it is wrong!

In today's world, there isn't a single IT organization that can control and engineer all aspects of the data that their enterprise needs.

Across companies of all sizes, the data changes, systems and applications are evolving, and infrastructure changes with the emergence of new platforms. The net result is that you can't control the actual data because you don't control all the systems and variations of sources and infrastructure. Because of this, Data Drift is continuous.

> ***Data Drift:*** *The unpredictable, unannounced and unending mutation of data characteristics caused by the operation, maintenance and modernization of the systems that produce the data.*

In addition to Data Drift, IT professionals, and those in the business who are using self-procured cloud native services because they don't want to deal with IT, should care about DataOps because their biggest systems are not the core application systems that they operated during prior decades. Rather their biggest system, and therefore the largest cost, is the collection and sharing of data regardless of source (internal enterprise, external partners, customers, public and other) and regardless of technology. The biggest "system" is in fact the continuous and rapid flow of data. And because the holistic flow of data is generally unmanaged, it presents the greatest opportunity for efficiency improvements, risk reduction and business value.

Why do you need a professional DataOps team now? You haven't had DataOps to this point, so what has changed? There are three forces which are pushing data behavior into new concepts that demand new thinking and new capability. These three forces have united to create a "perfect storm" in data management; a combination of events which are not individually worrying but occurring together produce a disastrous outcome. The three forces that are creating the data storm are outlined below.

1. **The first force** is the four V's of big data (volume, variety, velocity, and veracity) which is like a hurricane, making it hard to find, deliver, and access data. Once data is under control, it changes. The quantity, speed and endless variety of data (unstructured, structured, batch, real-time, streaming, cloud, IoT) feels like the chaos of a hurricane. It all must be rationally defined to be trusted, make sense, be truthful and be protected from people who may damage it or steal it. This is a scale of complexity that didn't exist even 10 years ago.
2. **The second force** is an unceasing wave of technology change. Data management technology is endlessly being improved to find data in new devices and structures. It needs to be transformed, delivered to where is needed, and cataloged, analyzed, monitored, secured, compressed, archived, and the list goes on and on. In its totality, it feels like a tidal wave—a tsunami of technology.
3. **The third force** is that data is more valuable than ever. Data is now independent from applications and must be managed explicitly in all states and forms in order for the enterprise to operate its business-critical requirements. Data is highly valuable since it is no longer just facts about business or operations; analytics are predictive and prescriptive and,

> in many respects, data "is" the business. The third force is that data is an asset that needs to be governed and secured and at the same time needs to be democratized and used widely. As one CIO said, the biggest cost of large data volumes is not the storage capacity, it is management's time to talk about it.

To sum up this section, data is the new oil, the most valuable asset in many companies. No longer is data confined to a data warehouse that gets processed and lifecycle managed by career professionals. It is now the foundation on which modern enterprises build their business-critical logic. DataOps is a new capability that builds on a combination of new technologies, data management practices, and collaboration across functions in the enterprise.

1.4 The Astonishing Speed of DataOps Practices

The following scenarios show how traditional methods for a specific data analytics solution require eight months, while DataOps delivers the same result in two weeks. If this sounds unbelievable, read on.

You are a lead architect tasked with building an architecture for a new data science application that will apply machine learning for predicting customer purchasing patterns and, in real time, recommend items that a customer may buy. The Data Scientist is clear that the more data she gets, the better the models and predictions will be.

You learn from the Data Scientist that she expects live feeds (real time) from the company's e-commerce website; historic data from several data-marts/warehouses around the organization (customer profile, transaction histories, preferences and habits data, product data, service history data, etc.); data from external APIs such as weather and crime statistics near customer location; and promotional data from third parties looking to promote similar products.

After several discussions with the Data Scientist, several days to collect ideas from other architects and scanning the enterprise portfolio of applications and data stores, you build a list of 15 high-priority internal data sources, five additional data sources that seem likely, and 10 key websites with public or social media data that is relevant. The data sources run the spectrum from bulk loads of historic data, to real-time data streams, to API calls from outside sources. They contain structured, semistructured and unstructured data and some of the data include sensitive PII (personally identifiable information) that, if leaked, could be a major liability for the company.

This example below shows how Scenario A consumes eight months to build the solution while Scenario B delivers the same result in two weeks based on a DataOps COE (center of excellence).

1.4.1 Scenario A

The lead architect scans the demands and realizes the following needs:

- Developers with the right skills to handle the unique characteristics of all of these unique feeds. They should understand the mechanics of getting data out of core applications, a mobile app, as well as semantics of reading out of large data warehouses.
- Operations Engineers who need to understand how to bring up and maintain the execution engines that run these varied and dynamic workloads; when doing massive batch loads, during peak shopping hours that need a lot of capacity, and at other times. These folks also need to make sure they don't build up unnecessarily large infrastructures that are idling 50% of the time.
- Operations Leaders that are monitoring every aspect of this architecture. They need to know what data is flowing for normal operations and when the flows deviate from the norm.
- Security Engineers who need to ensure that no sensitive data flows into areas that are classified as being insecure and that those without the right clearances are unable to see secure data.

The lead architect also has to contend with some constaints:

- Developers are not equal; engineers that have built mobile applications that are transmitting data back to the data center don't necessarily understand the semantics of reading large volumes of data from the central warehouse.
- Developers will have to resort to using different tools/execution engines to solve each of these different problems.
- Operations Engineers have to spend a lot of time understanding the technology and operational characteristics of each execution engine and develop different automation

strategies for each of them. Not all the tools used will scale elastically; as a result, they will need to size environments for peak load — resulting in potentially large infrastructure expenditures and many more machines to manage — or significant architecture and design innovations to work around capacity variations.
- Operations Engineers will also have to set up monitoring dashboards for every tool they use and set up alerts unique to each tool — which also consumes architecture time or additional implementation steps.
- Security Engineers will need to examine every data flow point, check to see what PII exists, and define policies to protect the data and communicate violations to developers who need to go enforce those policies. Developers will do this for every tool/dataflow they've developed based on the underlying technology.
- If changes occur to the datasets (i.e., new PII fields show up), if and when the change is detected, both Security Engineers and Developers will scramble to figure out how to protect the new data, and what if anything is at risk since they didn't detect the new pattern initially.

In summary, in Scenario A the lead architect needs to plan at least five projects (or more depending on how many different technologies are involved) with each team needing an average of five staff. The typical cycle-time of each project is about three months, but they can't start at the same time due to the availability of key staff or needing to work around system or operational constraints, so let's assume that all five projects finish at the end of six months. At that time the final integration test across the 20 applications, capacity validations, corrections and bug-fixes, and production deployment consume another two months. Voilà – after eight months the Data Scientist can start her analysis.

1.4.2 Scenario B

When using the DataOps COE based on StreamSets technology, this project is much simpler:

- **Any Developer**, without specialized skills, can create pipelines that get data from any data warehouse, external APIs or execute directly on big data clusters. When they connect

to the source systems, they don't need to understand the underlying schema of the data or the particulars of extracting data from that source. These automatic pipelines allow the data engineers to express their intent in an easy manner and do the heavy lifting automatically. For example, format changes, filtering by value, aggregate functions and more can be applied without the data engineer knowing the full schema of the data and will continue to work even when the schemas change at runtime, as long as the intended requirements are satisfied.

- The **Operations Engineer** can build up a Kubernetes-based infrastructure that they already use for the other microservices- based applications the company uses. StreamSets will run its execution engine on this infrastructure and the engineer can choose not to get into too many details of how the execution environment works. System-level monitoring they've set up for any and all Kubernetes applications will easily be automated and replicated for the StreamSets execution environment. Overall, they get tremendous efficiencies of scale.
- The **Operations Engineer** also pulls up the StreamSets dashboards and monitors runtime execution of all the dataflows. They get alerted for issues no matter where in the architecture the problem arises. And finally, they can monitor the overall topology (living map) and share it with the **Data Architect**. This living map not only represents the current state of the architecture exactly how the architect originally imagined it, but automatically tracks changes in the architecture over time.
- The **Security Engineer** sets up centralized policies for handling secure data and ensures that each underlying pipeline (no matter who developed them) protects the data. They can see an audit report of all PII flowing through the environments and prove that they have taken all necessary steps to protect it.

In summary, Scenario B is able to complete the solution with just five specific individuals. Because of StreamSets technology, the team is able to support all 20 internal and 10 external data sources and requires less effort due to the level of automation. Voilà – within two weeks the Data Scientist can start her analysis.

2

DataOps Playbook Primer

Now that we've demonstrated the power of a DataOps approach, let's build on the earlier description of **DataOps in a Nutshell,** by detailing and defining a) its purpose and responsibilities, b) the services it provides, c) the problems and challenges it resolves, and d) the required practices and skills compared to traditional methods.

2.1 DataOps Purpose and Responsibilities

To repeat, there isn't a single IT organization that can control and engineer all aspects of the data that their enterprise needs. Data is now the connective tissue that complex enterprise logic is being built on, spanning numerous applications. DataOps delivers flexible data flows that maintain the connectivity between systems in the face of changing environments, requirements, infrastructures and semantics, all while preventing data losses and threats.

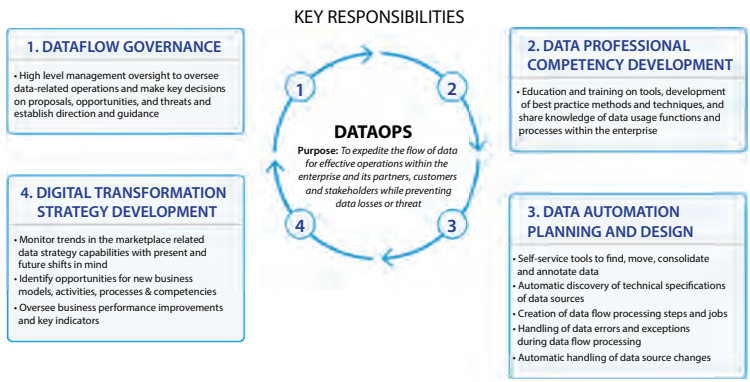

Figure 2.1 DataOps purpose and responsibilities.

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (13–22)

DataOps does not change the application behavior, it changes the dataflows to ensure that all of the application's implied contracts are respected in the face of drift. It accomplishes this through four key responsible areas as outlined in Figure 2.1 above:

1. **Dataflow Governance**: High-level management oversight to oversee data-related operations, to make key decisions on proposals, opportunities, and threats, and to establish direction and guidance.
2. **Provide Information to Professionals and Stakeholders**: Education and training on tools, development of best practice methods and techniques, and knowledge sharing of data usage functions and processes within the enterprise.
3. **Data Strategy Development**: Support digital transformation programs and modern analytics in several areas:
   - Monitor trends in the marketplace related to data strategy capabilities with present and future shifts in mind,
   - Identify opportunities for data-driven business models, activities, processes and competencies in line with digital transformation strategy, and
   - Oversee efforts for business performance improvements and developing new key performance indicators.
4. **Planning and Design**: Ensure selection of appropriate tools that are built for DataOps; allow for data security by design and not as an afterthought. Disallow tooling that is opaque and doesn't allow for a high degree of automation and monitoring; enhance acquired tools to support automation and self-service capabilities for data users. Ensure data engineers, operators and architects are appropriately trained in the relevant tools and techniques.

2.2 DataOps Services

We describe the DataOps Center of Excellence services in Figure 2.2.

This figure shows the services the DataOps COE provides to a) the enterprise from a holistic perspective, b) function owners for business

Figure 2.2 DataOps services and supported stakeholders.

units operational teams, and c) data professionals. The primary DataOps services to support the enterprise are:

- **Assurance of Data Flow Service Capacity**: Ensures data movement and exchanges between sources and destinations, effective operations of data pipelines and control systems, and the ability to support data volumes peaks and preparation for future demands.
- **Assurance of Data Usability and Protection**: Approve access for enterprise stakeholders and for establishing the relationship between data owners and data consumers and analysts (e.g., data scientists).

- **Oversight of Data Flow Protection and Regulations**: Guidance for measuring, analyzing and reporting data delivery operations, quality, security, and compliance rules.
- **Data Flow Performance Measurement and Alignment**: Monitor the data flow operations and measure volume, speed and trends including usage and availability across enterprise users.

The primary DataOps services to support business functions and their leaders are:

- **Support for the Productive Use of Data Resources**: Ensures capturing, collecting, and publishing data about data and data processes to support data analysis and strategic planning functions. Use cases include business definitions, data source and owner information, and metadata related to data capture, update, change history, and distribution.
- **Establishing & Monitoring of Data Analytics Management Practices**: Implementation and oversight of analytics policy, standards and procedures, resolving analytics and reporting issues, discovering new meaningful patterns in data, communicating business insights to relevant planning functions, oversight of sandbox environments, and initiating production analytics capabilities.
- **Digital Transformation Opportunity Assessment & Strategy Initiation**: Enable Digital Transformation with a formalized framework, best practices, and modeling tools to identify improvement opportunities, define business priorities, sequence migration phases, develop roadmaps, and create business cases.

The primary DataOps services to support data professionals in IT and business functions are:

- **Establishing and Monitoring of Data Flow Practices**: Develop procedures for operational roles, rules and processes to ensure that data access, distribution and quality meets performance requirements. Share information and train data users for the appropriate use of data flow practices.
- **Development of Data Flow Standards and Guidelines**: Ensure that DataOps strategies and policies are followed by

developing and maintaining related framework, methodology, tools and standards.
- **Data Process Automation Capabilities Modeling**: Support the development of self-service and automated data discovery, delivery and quality by defining operational and technology architectural models.

2.3 Challenges and Problems that DataOps Resolves

Effective use of data is indispensable to your business; you should have an advanced and mature professional data management practice. This section explains some of the causes of potential data issues, why traditional methods don't address them effectively, and how DataOps is different.

In short, the nightmare for data professionals is the inability to keep up with the dramatic increase in data complexity, variety and scale. Building on the perfect storm outlined in Section 1.3, there are three specific categories which keep data professionals awake at night: data sprawl, data drift and data urgency.

Data sprawl is the dramatic variety of data sources and their volume. Consider systems such as mobile interactions, sensor logs and web clickstreams. The data that those systems create changes constantly as the owners adopt updates or even re-platforms those systems. Modern enterprises experience new data constantly in different formats, from various technologies and new locations.

Data drift is the unpredictable, unannounced and unending mutation of data characteristics caused by the operation, maintenance and modernization of the systems that produce the data. It is the impact of an increased rate of change across an increasingly complex data architecture. There are three forms of data drift: structural, semantic and infrastructure.

- **Structural drift** occurs when the data schema changes at the source, such as application or database fields being added, deleted, reordered, or the data type being changed.
- **Semantic drift** occurs when the meaning of the data changes even if the structure hasn't. Consider the evolution from IPv4 versus IPv6. This is a common occurrence for applications that are producing log data for analysis of customer

behavior, personalization recommendations, and so on. Another example of semantic drift is accepting international postal codes rather than just zip codes; alphanumeric rather than digit codes!

- **Infrastructure drift** occurs when changes to the underlying software or systems create incompatibilities. This includes moving in-house applications to the cloud, mainframe apps to client-server systems, or moving from a traditional database to big data solutions, such as Hadoop, NoSQL, Hive and Sqoop, to mention just a few of the many technology options.

Data urgency is the compression of analytics timeframes as data is used to quickly make real-time operational decisions. Examples include Uber ride monitoring, fraud detection for financial services, next best offers in e-commerce, or real-time notification of customer issues. In addition, the Internet of Things (IoT) is creating ever-increasing sources of transactions that need immediate attention: doctors are demanding input from medical sensors connected to their patients, utilities need to balance energy generation on the grid in real time, companies are investing in automating sensors on equipment, trucks and buildings to monitor their status, potential failure events, and on and on.

> You might be thinking "*The issues of data sprawl, drift and urgency aren't new and have been around for years. Why do we need DataOps now and what is wrong with traditional processes?*"

You would be correct in thinking that these aren't new issues, but their increased frequency and magnitude are new requirements! In past years, these issues were generally isolated and could be dealt with using standard exceptions methods. Let's look at how data service incidents were resolved in the past (and are still used today in many enterprises).

1. **First,** an exception event occurs. Maybe it is flagged by a computer mainline job that "abends" (ends with an error) and is noticed by a data center operator or maybe a monitoring job sends out an event pager. Alternatively, a business owner may see odd results in the monthly sales performance report, or a customer calls the service desk to complain about a slow website. In any event, someone in the incident management team or help desk is notified of the exception.

2. **Second,** the help desk gathers as much information as they can and makes an assessment of the "severity level". For a low severity, they send an email to the application owner and ask them to look into it when they can. If it's a Severity 1, they take a more dramatic action and initiate the "Severity 1 Group Page" which notifies dozens of staff to organize a conference call.
3. **Third,** the staff on the conference call work to a) understand the current issue and its impact, b) analyze the problem and determine the root cause, and c) figure out how to correct the situation and return to normal operations. Dozens of staff are involved because it's not clear up front what the precise problem or correction is, so anyone that "might" be able to help is required to attend. In any event, usually the team is able to return functional operations and data flows are restarted. But the incident recovery often does not result in a permanent solution and the company needs to know the root cause and how to avoid future recurrences.
4. **Fourth,** a postmortem process is initiated to fully understand the root cause and how to avoid it in the future. It could take several weeks to understand what happened, followed by a group review meeting by multiple SMEs and managers, followed by a formal report and recommendations for division leaders, internal audit, or senior management. Hopefully the defined recommendations are approved, and a permanent resolution is implemented.

It is clear that this process is painful, expensive, and simply won't work in today's reality of increasing data complexity, data variety and data scale. We need a better solution built on the assumption that data sprawl, data drift and data urgency are the new normal.

It's worth noting that with DataOps, the number of incidents is dramatically reduced not because the issue frequency slows down, but because an infrastructure that executes with DataOps can automatically handle a vast majority of the typical changes and emerging characteristics of systems. These same changes could cause devastating damage if not handled via DataOps and could lead to data corruption, loss, and SLA breaches that create cascading failures downstream.

2.4 Practices & Skills Compared to Traditional Methods

DataOps sounds compelling, but what exactly are the procedures and techniques for applying the practice? The table below compares DataOps methods with traditional means, organized by data needs. The methods listed below could leverage DataOps products or could be developed from open-source or commonly available technologies, or simply are processes that could be applied by skilled managers or subject matter experts (SMEs).

Data Usage Needs	Traditional Data Integration & Execution Requirements	DataOps Practices and Advantages
Design Solutions	Requires exact schema-to-schema mapping specifications.	Requires minimal schema specifications to accommodate a range of change (data drift) adoption without redesign.
Infrastructure Dependency	Integration logic is tightly coupled with the underlying infrastructure; therefore, infrastructure cannot be changed without redesign.	Provides a decoupled "run-anywhere" semantic that is infrastructure-agnostic and allows the same integrations to operate in diverse environments. Provides portability and freedom from technology lock-in.
Data Delivery Speed	Requires applications to work with the execution modes offered by the data integration system. Batch and streaming require different technical solutions.	Enables a "run at any speed" semantic that allows integrations to adjust to application requirements. Provides portability across batch, microbatch, streaming, and real-time operations modes.
Data Map	Relies on manual design-time documentation to produce a visual map for individual integration (not complete maps for large enterprises).	Self-documented through metadata to enable live views of data interconnectedness within the enterprise, with capabilities that can track evolution over a period of time.
Dataflow Consistency & Variation	Requires consistently matched metadata scenarios. Data variations are hard-coded for specific specifications.	Intent-oriented design that applies broadly whenever the same type of data is in motion. Supports reuse across scenarios even when lower-level semantics are different.

Figure 2.3 DataOps methods and advantages compared to traditional methods. (*Continued*)

Data Usage Needs	Traditional Data Integration & Execution Requirements	DataOps Practices and Advantages
Continuous Integration & Deployment	Provides black box integrations that must be designed to specific requirements. Requires manual and explicit handoffs for deployments in production for dataflow integrations.	Whitebox integrations that protect data corruption, loss or failure, and enable operation in the face of changing requirements. Provide criteria based on the automatic promotion of integrations to ensure agility of operations.
Continuous Data Privacy	Requires data privacy and protection logic to be designed for each pipeline. Demands complex intertwined logic between orthogonal concerns like integration and security.	Tiered data privacy rules applied to integrations automatically based on governance policies powers information security teams without requiring awareness of lower-level details. Decouples security and integration concerns.
Entity Centric Services	Active support for entity-centric services such as live catalog and glossary management through custom-developed software.	Operates on the basis that all integrations are directly or indirectly related to the entities within the business domain and therefore works to keep the reference systems like catalog and glossary in sync automatically. Provides up to the minute visibility into how and where data infrastructure is aligned with the business processes it supports.
Zero Downtime Evolution	Requires taking integrations offline while making changes to logic or underlying systems. Treats every integration as a black box during runtime and does not enable the rapid evolution of integration in the face of changing requirements.	Enables the rapid evolution of integrations while ensuring the continuity and integrity of data flowing through them even as change occurs. The capability to make this happen is an approach to "instrument everything" towards data integration using dataflow sensors. In such scenarios, every change is captured and evaluated at different context levels (that of the immediate integration, the application being integrated, the topology that this application is a part of, etc.). Different context levels result in different insights that spotlight the emergent designs within the infrastructure, thus enabling rapid iterations without the need for expensive redesigns.

Figure 2.3 (Continued) DataOps methods and advantages compared to traditional methods.

Data Usage Needs	Traditional Data Integration & Execution Requirements	DataOps Practices and Advantages
Bad Record Management	Handle bad records as exceptional cases with little or no capabilities to understand, anticipate and reprocess them. As errors increase, data flows stop.	Treat bad records as a natural consequence of operations and provide support for the direct handling and reprocessing of such data, thereby ensuring that there is minimal data corruption or loss along the way.
Data Metrics and Alerts	Provide limited design time support for profiling data and consequently visibility into how data is changing and evolving at runtime.	Provide fine-grained visibility into how the data evolves at runtime based on constant runtime data profiling, ensuring the operational integrity.
Data Confidence	Requires manual subject-matter experts and knowledge across a group of people to ensure data integrity and correctness.	Provides direct tie-in and automation to ensure the integrity and correctness of data that flows through the infrastructure at all times. The underlying technology is based on ML (Machine-learning). Use ML to process large amounts of sensor data and thus identify the outcomes that support data confidence.
Data Marketplace	Traditional DI does not provide a systematic means to further the sharing and access to data in the enterprise. Different roles such as designers, architects or data analysts generally interact with others within the same function across rather than systematically across functions.	DataOps is based on the principle that data powers business processes at the core. Sharing and enablement provides access to data in all parts of the enterprise while being compliant with security, privacy and other requirements. Consequently, DataOps lends itself towards better sharing and secure sharing of data across the enterprise.

Figure 2.3 (Continued) DataOps methods and advantages compared to traditional methods.

3

Starting the DataOps Practice

There are two main strategies for implementing DataOps: bottom-up evolution and top-down transformation. The quickest way to *start* is bottom-up by data professionals simply applying new methods incrementally. The quickest way to *finish* a mature practice that is embedded company-wide is top-down by following a transformation roadmap with senior management support.

A common need for both strategies is to leverage change agents. DataOps will change how the company ingests, propagates and uses data so it is critical to have one or more change agents who:

- Are voracious learners
- Do not wait for orders to take action on new ideas
- Express excitement freely concerning new ideas and change
- Demonstrate a sense of urgency to capitalize on innovations and opportunities
- Challenge the status quo
- Transcend silos to achieve enterprise results
- Skillfully influence peers and colleagues to promote and sell ideas
- Display personal courage by taking a stand on controversial and challenging changes

Start DataOps quickly from the bottom up by finding a few change agents to begin applying practices. They may be new employees or long-term established staff with a network of relationships and the ability to get things done across the enterprise. There are some change agents in every organization, so find them; maybe you are one of them. Collaborate with them, start applying simple Agile or Lean methods such as flow of value, waste elimination and fail fast, and evolve the capability as you have successes.

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (23–34)

The top-down transformation of DataOps leverages same of the same methods as bottom-up, but with more structure and formality.

1. Identify an Executive Sponsor
2. Define the Vision and Charter and Inform Stakeholders
3. Develop a Roadmap to Map the Journey
4. Execute and Advertise the COE
5. Periodically Assess and Renew Plan
6. Reinforce the DataOps Culture

First, you need support from an executive sponsor since you will run into resistance from team processes, policy changes, funding needs and other roadblocks. It is essential to have a senior director, VP or C-level officer, that you can work with to support the DataOps vision.

Second, formally define and document your vision and charter. One way to start is to simply ask your executive sponsor, *"What keeps you up at night?"* and *"If our DataOps turns out to be successful, what would that look like from your perspective? How would you measure the results or talk about the outcomes?"* You should also review your company's annual report and incorporate priorities of the CEO or chairperson.

3.1 Developing the DataOps Roadmap and Gaining Executive Support

Launch your DataOps blueprint; the blueprint consists of three elements:

- **Strategic Roadmap** is a "checklist" of milestones or outcomes arranged in multiple tracks and phases. Find specific leaders/managers to assume responsibility for the tracks and phases, but the strategic roadmap does not specify "how" the milestones are to be accomplished, only "what" the outcomes will be.
- **Program Roadmap** is to define and gain approval for specific initiatives including business justification, costs, change drivers, timelines, current/future state models, risks and constraints. This map adds concrete initiatives to the strategic roadmap and plays them out in phases.
- **Project Plan** details efforts for a program initiative showing a detailed breakdown of activities, resources, dependencies, costs, deliverables and other elements defined by the

Project Management Body of Knowledge (PMBOK). Start your DataOps Transformation with at least the first project clearly defined.

Figure 3.1 shows a Strategic Roadmap template intended to represent your entire plan in one page by documenting milestones or outcomes across three phases and three tracks. There will be more time in later months to add additional details, but at the beginning it is important to have a holistic one-page plan that contains the critical points to help articulate the plan and reach agreement across all key stakeholders. See chapter **9 DataOps To-Do List** for ideas to start completing the roadmap.

The main value of the Strategic Roadmap is to quickly develop the plan; a few days or one week may be sufficient. Roadmaps may be created in a four-hour workshop. The components of the roadmap are:

- **Program Owner**: The person responsible for ensuring that the roadmap details are completed
- **Program Sponsor**: Senior staff supporting the resource needs
- **Roadmap milestones** ordered by tracks and phases:
  - **Tracks** defining at least three dimensions:
    - People and Organization
    - Process and Policy
    - Technology and Infrastructure
  - **Phases**
    - For a 3-year roadmap – Phases are Year 1, 2 and 3
    - For a 1-year roadmap – Phases are Quarter 1, 2, 3 and 4
    - For a 3-month roadmap – Phases are Month 1, 2 and 3

Figure 3.1 A DataOps strategic roadmap template.

The complete blueprint may take a few months to fully define and gain agreement across key stakeholders. The Transformation strategy will take longer than the Evolution approach to get started, but it will establish a foundation to scale the DataOps capability and sustain it to become adopted company-wide.

The next steps for transformation are ongoing activities to grow and sustain your DataOps capability:

- **Execute and Advertise/Market**: The Program Owners develop detailed project plans, secure the resources, and then make it happen. Keep stakeholders and the data community up to date with progress. Make specific efforts to highlight successes and measurable outcomes.
- **Periodically Assess and Renew your Plan**: Do a periodic review with the Executive Sponsor and leadership team; depending on the pace and speed of your Transformation, you should do a review every month, quarter or year. At least once a year you should review and possibly reshape the plan, especially if company strategies have evolved or significant technologies or other best practices are now possible.
- **Reinforce the DataOps Culture**: This is an ongoing process and is a deep enough topic that we will expand it in Chapter 4.

In summary, start DataOps with minimum investment. A large investment may happen as the team or capability grows and becomes adopted across the enterprise, but when that happens, the payback will be obvious, and the investment will be justified.

3.2 Overcoming DataOps Resistance

It would be a mistake to underestimate the difficulty in leading DataOps change in a large enterprise. Some of the really difficult challenges include:

- The "not invented here" syndrome and similar behaviors of people simply resisting anything that comes from "the outside";
- Project funding by fine-grained silos that don't have the money for and aren't motivated to solve the "big picture";

- Tactical short-term investment emphasis that doesn't appear to leave any room for strategic infrastructure investments;
- Concessions and trade-offs needed by tactical pressures that get in the way of "the right thing" in the long term;
- Autonomous operating groups in distributed geographies that will not accept guidance from a central group; and
- Fear of change and vested interests in the status quo.

The term "challenges" may be too polite when referring to the above list; these seem a lot more like immovable barriers. As insurmountable as these hurdles may appear to be, they are not unique to a DataOps implementation and have been conquered in the past. While there is no simple "silver bullet" solution, there are a number of key concepts which have been proven over and over to be effective. Here are seven of the best:

1. **Think strategically and act tactically:** Have a clear vision of the future but be prepared to get there one step at a time. It is good to keep in mind that *there is no end-state*. In other words, things are always changing. For example, if you miss a window of opportunity to establish a new architectural standard on the latest project, don't worry. Another project will come along. If you are in it for the long run, individual projects, even big ones, are just blips on the radar screen.
2. **Credibility through delivery:** In order to be perceived as a leader by others in the enterprise, you need their trust and respect. It is not just about being open, honest and trustworthy; but do people trust that you will actually get the job done? In the final analysis it comes down to your ability to execute. To organize your work, set appropriate priorities, assign the appropriate resources to the task and maintain good communications with your customers. Above all, keep your promises.
3. **Sidestep resource issues:** In this global economy of outsourcing, offshoring and contracting, you should always be able to find the resources to get a particular job done. If you want to create a reputation as a "can do" customer service–oriented team, there should never be a time when you need

to say "No" to a service request due to lack of resources. (There may be other reasons to say no.)

4. **Choose your battles:** Whenever you have the choice between a carrot and stick approach, always use the carrot. You can, and should, carry a stick in terms of having the support of senior executives for any mandated processes or standards, but you should use the power as infrequently as possible. Sometimes this might even mean deviating from enterprise standards. One way to help you choose your battles is try this exercise. Write down your DataOps principles on a piece of paper and cross out one at a time starting with the ones that you would be willing to compromise if pushed into a corner until you only have one left. That is the principle that you should use your stick for.
5. **Take out the garbage:** Accept responsibility for work that no one else wants. An interesting lesson learned is that many of the jobs that no one really wants are those that don't serve a specific function but end up being ideal data initiatives. Sometimes these also end up being really difficult challenges, but generally they are recognized by management as such, which opens the door to asking for top-level support when needed.
6. **Leverage knowledge:** There is a well-known truism that states that "knowledge is power". In a DataOps team you are ideally positioned to talk with just about anyone in the organization. By asking a lot of questions and being a good listener, you can gain a lot of knowledge about the organization that narrowly focused project teams or groups don't have. This knowledge can come in very handy in terms of which projects are getting approved and where you shouldn't spend your time, where next year's budget will land, which groups are hiring and which aren't, etc.
7. **Take it outside:** Another aspect of leadership is active participation in the broader community; specifically, participation in standards bodies and professional organizations. The external activities can be useful for both getting new ideas and insights, and for polishing your own ideas through discussion and debate with others. These activities make you stronger as an individual, which can help you play a leadership role inside your enterprise.

3.3 DataOps Organization

DataOps is about bringing a high degree of automation and monitoring into operationalizing the movement and transformation of data. However, DataOps doesn't exist in a vacuum, it serves a business function. It fits into the context of the larger organizational structure and technical architecture, and it enables downstream users/consumers of data. Success in achieving a state of DataOps nirvana means not just creating a platform for DataOps but creating a culture within project teams that allows for a focus on outcomes rather than just declaring success on a particular technology function.

To think about creating an effective team at the highest level there are two primary factors at play:

1. Who participates in teams delivering value via DataOps.
2. How do these teams operate, centrally or decentralized.

The IT workforce is often structured around a highly specialized set of competencies. For example, a Database administrator as the sole arbitrator of any and all Database-oriented decisions. An ETL Engineer responsible for creating pipelines to move data into Data Warehouses. A program/project manager setting the overall schedule of how the project progresses.

Enterprises have realized that fast-evolving technology landscapes and dynamic business objectives have necessitated a far more agile and cross-functional approach to delivering value. Because of the advances of modern technologies (especially on cloud), users can do a lot more and work at much higher levels of abstractions because the underlying services are expected to take care of the details. As an example, that ETL Engineer job could now be a higher order DataOps Engineer who is able to not just create and automate data movement/transformation jobs, but can create an environment where hundreds of business users can self-serve data themselves without needing to understand any of the details of the underlying infrastructure.

A pattern that has worked well in many successful DataOps organizations is to have small cross-functional teams that behave as a Product Development organization. Let's say a health insurance company wants to be able to provide its customers with a fitness monitoring device. The company is doing this not only to provide customers a valuable new service that's deeply integrated with the overall healthcare experience, but also to set the next level of insights into their customers and let them use aggregated data patterns to improve products and services.

A traditional approach to managing a project of this magnitude would be to set up a potentially large business unit to start from the top and build out a multiyear plan to execute it. A start-up time of a year or two to get the unit producing, and a five- to ten-year lifecycle would be a reasonable assumption. However, competitive dynamics may not allow for long start-up times; technologies relevant today may be gone tomorrow and the value derived from the endeavour may need to quickly evolve based on overall market and economic drivers. A leaner approach would be to rapidly test product-market fit with small percentages of the targeted customer segments, fail fast and adapt where required, and incrementally build out the business based on evolving project success factors. A small highly cross-functional team should focus on proving out ideas in small incremental and demonstrable steps.

Typically a Product Manager owns and drives the overall vision. Data Scientists or Analysts set the stage for how to derive value from data. Technology and Data of course play an outsized role in such a project. There are tens or hundreds of options for technology and methods available to solve a problem. Being able to choose one, test it out with the lowest impact possible, and iterate quickly to change is critical to achieving success. DataOps Architects and Engineers set the stage for choosing and enabling technologies providing the highest value and the lowest amount of overhead (economic, functional and operational)—in other words technologies should be best in class within the budget constraints they need to operate on, provide the best set of capabilities for current and future work, and require the least amount of human capital for start-up and ongoing work.

"Centralized vs. Decentralized" has been an evergreen debate in every enterprise since the early days of corporate entities. It has certainly been a critical decision for all IT organizations ever since they came into existence. In the 21st century, with the ever-increasing velocity of changing business objectives and underlying technologies, enterprises often need to recalibrate this debate every few years. Systems that were managed centrally and on premise by a single IT group could move to a SaaS environment that doesn't necessitate administration by operations staff and could potentially be managed by decentralized teams.

One may ask then, what is the right answer to the Centralized vs. Decentralized question for DataOps? Brace yourself… there is no right answer, and the answer almost always is "It depends". What DataOps encourages is to apply a framework to answer this question for every organization and every problem.

- A popular framework for this debate championed by McKinsey in 2011 was to ask the following three questions:

- Is it mandated? (Do external stakeholders or laws require it?)
- Does it add significant value? (Does it add 10% market cap or profits?)
- Are the risks low? (Does it avoid risks of bureaucracy, business rigidity, reduced motivation or distraction?)

The decision to centralize typically required a "Yes" to any one of these questions. A "No" to all of these questions resulted in a decision to not centralize.

These questions continue to be the right ones for any enterprise; however, modern technology and business landscapes require an updated set of secondary criteria to consider in that debate.

- Is it mandated?
  - In recent years, regulations such as Privacy, Data Sovereignty, Data Residency have become de facto drivers of architectural decisions. Storage, processing and access to data are subject to these new laws, and enterprises often need to require compliance to even do business. A decision to centralize or not will have to account for and provide mitigating strategies for all of these factors and more.
- Does it add significant value?
  - While monetary value is certainly a powerful motivator, business strategies can't always provide clarity about the sources of underlying value. Technology or team management are often far removed but critical enablers for getting to increased business value. For example, a decision to move from an onprem data warehouse solution that required a lot of maintenance and elaborate change management procedures, to a SaaS data warehouse solution that any organization within the enterprise can quickly get productive with for their own projects—and where change management is a contained local problem—can greatly accelerate individual project teams to become successful and move the needle for the larger enterprise.

Successful DataOps cultures often adopt a strategy where a central DataOps platform team is the clearing house for system/data architecture, best practices and evangelism throughout the organization. This team has DataOps Architects & Engineers, and Data Scientists, among others, and works under the auspices of the Chief Data Officer. This group may lay

out broad-based guidelines of technology use and data models based on enterprise needs. When designed for agility, these guidelines and technology choices allow downstream project groups to rapidly iterate and build products and solutions quickly. They may choose to innovate and pick a particular technology that works well for their project but will do so while bringing in DataOps sensibilities of continuous governance, operations and data. Effective central DataOps groups take on the role of mentorship rather than one of gatekeeping.

In our healthcare fitness example, a central DataOps team would perhaps work out contracts with various cloud vendors and publish a service catalog of permitted technologies project teams could use. They might create central data models and access controlled API's for anyone in the organization to consume. They'd also publish and evangelize best practices, guides and reference architectures for all. Meanwhile, the project team that's trying to get the project off the ground would build an architecture drawing from published data models and permitted technology components. After a couple of sprints they might realize that a particular technology they had chosen, for example, a Hadoop cluster, just doesn't do what they need it to do or has high operational overhead. They could just swap that component out and try a cloud native object store and processing framework instead. Loosely coupled technologies and API's give that project team the highest amount of flexibility to focus on what's most important, i.e., outcomes, rather than getting stuck in solving technology problems that don't further the cause of the overall project. Of course, true success in the DataOps culture happens when these project teams are able to evangelize their success and learnings back into the broader organization and repeatable and reusable design patterns emerge.

Data too should ideally be widely socialized. For example, if our project team designed some analytical models that predicted customer behaviors, the model, dataset and learnings could show up in a company-wide data marketplace that other departments had previously not considered workable or useful. Emergent data, practices and patterns are the surest sign of a vibrant DataOps culture.

3.4 DataOps: Name It and Claim It

At some point in the roadmap execution, you will formalize the team and align it structurally with the organization. The general suggestion is to develop a name or "brand" of the DataOps team early, but keep it focused on specific needs and responsibilities. In short, don't start with a wide

scope; instead let the size and scope of responsibilities grow as the team demonstrates success and the demand for their services increase.

In addition to identifying the initial team, enterprises should also create space for individuals and change agents. They will be doing disruptive and hard work; the core team is the leading edge of the "spear" that the rest of the company can learn from and adopt as the practice executes.

In terms of the DataOps name or brand, you may be able to build on a competency center or center of excellence that already exists in the enterprise, such as a Security Competency Center, Integration COE, BI COE, Network Operation Centers (NOC), and so on. One option is to structure it as the Chief Data Office (CDO) or Data Operations Center (DOC) that serves as a center of excellence across the enterprise as it is influencing and directing the practice being developed across the enterprise.

4

Institutionalizing the DataOps Culture

The most mature phase of DataOps capability is maintained not as a project or program, but as a shared set of values, goals, and conventions that are viewed simply as *the way the enterprise does things*. It takes time to achieve this level of maturity, but once you have realized DataOps as a "culture", you don't need to do anything special since it is the normal way people perform. In time, effective aspects of DataOps will become routine and generally accepted; they are simply acknowledged as "why would you do it any other way?"

To start, what does it look like when you have a DataOps culture? The result is that dataflows in your enterprise are clear, fast and safe. Similar to air traffic control for aircraft, you know where all your data is flowing and when it gets to the target! And if there are any data drift issues, like weather problems affecting air traffic flow, they are handled automatically by systems or the staff as needed. And if a new data source is needed or appears, it is added to the enterprise data flows in a matter of days or hours. In short, the key capability of your enterprise data mastery and data value is controlling not the creation of data and the related software, hardware and infrastructure – rather, the center is the efficient and rapid flow of data delivered to the right target where it can be used as needed. DataOps is a way of travel, rather than a destination.

The key practices to make DataOps consistent and repeatable, and maintained for an ongoing basis even with enterprise reorganizations and replacement in key leaders, include:

- Define a Shared Vision
- Define Policies, Metrics and Goals
- Incentivize Good Behavior
- Enable Self-training and Shared Practices
- Automate Everything
- Continually Improve

As data collection and data sharing become routine, and analysis and big data become common practice in the enterprise, it is important that all management and staff, not just data professionals, become competent

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (35–38)

in understanding, creating and communicating data as information. The practices outlined here contribute to improving data literacy and enabling digital transformation initiatives in the business.

4.1 Define a Shared Vision

Data Integration has been a practice that has seen many iterations over the decades. Any enterprise, large or small, old or new, will have practitioners who bring their respective biases and practices to the game. When modernizing data practices and establishing a new culture of DataOps, leaders in the organization must rally the troops by laying out a vision that outlines the desired end-state in this new world.

The end-state vision could include highly automated data flows, resiliency to drift (in all its forms), self-service data to any function that needs it, dependability, and speed. The enterprise business end-states should result in a data-driven company that is more valuable for its staff, customers and all stakeholders.

4.2 Automate Everything

Automate everything is a lofty goal, but in reality, not everything can be automated; organizations should chart out end-to-end processes and make conscious decisions to automate where possible and acknowledge human steps in the loop actions where appropriate. It is important to automate not just physical data flows, but also related management information processes. For example, standardizing data security both through the adoption of tools and system for security operations, but also for security governance and policies.

It is worth noting that automation is vulnerable to data drift and can mask true failures if not done right. The correct way of automation is to balance it with the help of instrumentation and monitoring; to create a feedback loop that observes the data processes for their intended outcomes and raises awareness where such outcomes do not occur.

4.3 Define Policies, Metrics and Goals

After a shared vision is defined, it is important to align the policies, metrics and goals of data processes that drive toward that vision. Once a

policy is defined and formalized, it helps inform related projects, investments, processes and operations. For example, you could formalize a policy that:

> All internal research and product-related data domains are "enterprise" resources not "owned" by the function that created the data. As a result of this, the DataOps COE has the ability to access and extract data from source systems and deliver it to the data lake where it can be accessed by employees.

See Case Study 2 at Appendix A for a real-life example of this policy. The policies that are relevant for each business may be different, but in any event, they help to enforce the culture.

Metrics and goals, for DataOps specifically, also steer the culture. The term "Measure Everything" is a powerful call to action to get people in the organization to take a scientific approach. DataOps teams should establish early on primary and secondary metrics to measure. Primary metrics are high-level metrics that track the overall success or health of the project. They should be business outcome-focused and a small digestible set of indicators that executives and key stakeholders will use. Secondary metrics are more detailed parameters that the operational teams will use to track day-to-day health and well-being of the project.

However, metrics are only effective if they are also visible by everyone and generally available. For example, to make data quality a widely adopted capability that all management and staff care about, you need to define it, measure it on a persistent basis, and publicize the results. One organization that did this published trend charts and posted them near lunch areas, call centers and hallways. This was seen as a key driver to influence everyone in the business (not just the data team) to focus on data quality, and it had a large impact on the business results.

With a DataOps culture, visibility of metrics and goals becomes a key enabler in helping with cross-functional alignment and achieving operational efficiencies in a manner that benefits the enterprise.

4.4 Incentivize Good Behavior

As mentioned in the prior paragraph, simply making metrics and goals very visible communicates the importance to management and staff. In addition, you should add DataOps policies to the job responsibilities of most staff and factor DataOps metrics into compensation and bonuses for managers. Furthermore, lead managers and directors can include DataOps

goals to the agenda of routine staff meetings or business reviews. As some have said, "If it's important to my boss, it's fascinating to me!"

4.5 Enable Self-Training and Shared Practices

Training and onboarding should also include a top-down view of how the various business processes map to the data infrastructure and where their contributions can help ensure collective success. A business capability framework is a useful tool to support learning and also helps fuel data literacy across the enterprise.

Whether or not democratizing data is a stated objective in your organization, democratizing methods, tools and patterns should be a goal. Highly successful DataOps organizations are ones that allow for teams or individual contributors to develop solutions without much guidance and allow the rest of the organization to learn from and leverage their success.

Evangelism is the key to spreading the learnings—positive or negative; and allows for further organic growth. Institutionalizing channels for people to share their personal experiences and knowledge is a powerful way to reinforce both the DataOps culture and best practices. This could be as simple as a quarterly meeting for data analysts/scientists to meet and share recent successful results with others. More formally, someone in the DataOps COE can be appointed the point person promoting best practices.

4.6 Continually Improve

An effective way to institute a culture is to have front-line staff define improvements and create the DataOps practices. As described later in A3 Problem Solving in Section 6.3 Continuous Governance, once individuals invest effort into solving a problem, proposing a solution, getting management support to implement it, and realizing positive results, they are clearly bought in. As a result, front line staff "own" the DataOps capability and will work to market it and gain adoption by others. In other words, they will help make DataOps consistent and repeatable.

5

DataOps Functions

5.1 The DataOps "Big Picture"

Many people have asked "Can you send me a picture of DataOps?" It's a complex question because there are many perspectives, including its purpose, functions and processes, team structure, people and roles, technologies, systems and so on. And if it includes the transformation and change perspectives there are multiple views of a starting phase, target phase and multiple transitional phases, not to mention views for business leaders, IT staff and data professionals. A conceptual view is shown below.

The conceptual graphic shows a few concepts of DataOps, but no specifics. The Big Picture graphic that follows is a more substantive view of DataOps.

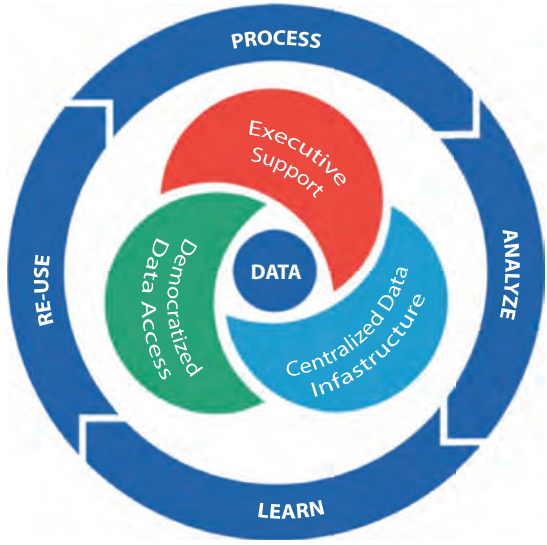

Figure 5.1 Conceptual DataOps graphic.

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (39–46)

Figure 5.2 DataOps "Big Picture".

The Big Picture starts at the top, which shows the DataOps purpose as enabling Digital Transformation by accelerating delivery for new business demands for data. The ultimate sponsor for this capability is the CEO and the senior leadership of the enterprise.

The next section of the picture shows that the transformation is in turn executed through a series of business initiatives led by organizational leaders. They leverage modern analytics to discover new insights and program managers to make the plans a reality.

Continuing down the picture we arrive at the picture where DataOps actually does the work to enable all these capability. Here we see the key practice areas of Continuous Design, Continuous Governance and Continuous Operations working together to deliver Continous Data. And those practices in turn are based on the DataOps Platform, which is the systems and technology to automate processes in the world of ceaseless change.

Notice the arrows connecting the Business Initiates and the DataOps Practices. The Data Projects adopt the practices from the DataOps capability below, as well as provide new learnings to the DataOps COE and develop new innovations that can be folded back into the DataOps Platform. These iterations are key to enabling continuous improvements than ensure that DataOps isn't just an effective method at a particular point in time, but can evolve as the industry evolves.

And finally, we come to the DataOps culture, which is a transformation by itself. DataOps isn't something you just install or "turn on"; it needs a program plan to fully execute changes and interact with other enterprise capabilities such as ongoing evangelization, training and organic growth to fuel its adoption and maturity.

The colors of the blue boxes are meaningful. They are the capabilities that leverage processes, methodologies and technologies to operate the DataOps Center of Excellence. These capabilities are unique to DataOps and are explained in the following sections.

Other areas, such as Organic Growth, aren't direct DataOps capabilities but rather interactive abilities from a company's existing resources. DataOps fuels Organic Growth by supporting modern analytics to identify innovations for customer experience, market penetration, market development, product development, diversification and others. The most dramatic types of growth may become Digital Transformation programs by leveraging data for new operational capabilities and new business models.

The DataOps Big Picture shows six blue items:

- Continuous Design
- Continuous Operations
- Continuous Governance
- Continuous Data
- Program Execution
- Design Operations

An overview of these functions is defined in the sections below, followed by Chapter 6 which will add more detailed operational capabilities along with principles, standards and best practices.

5.2 Continuous Design

The Continuous Design function enables delivering data solutions on an ongoing basis rather than as discrete project events. DataOps technologies

and systems are designed as a whole and not put together as individual Lego bricks that are pieced together after the fact.

Continuous Design is not the same as waterfall methods, which were key steps in projects usually following a requirements definition and preceding development or configuration processes. In DataOps, design is an approach that enables responses to changes that support continuous data availability. Key tenets of Continuous Design include:

- **Converge a top-down and bottom-up design philosophy.** Always look at the big picture and, conversely, allow details to shape and fashion the big picture.
- **Focus on outcomes.** Expect systems to abstract away complexity at every level so that the focus can always remain on solving business problems.
- **Maximize skill sets.** A developer should be able to design for edge devices or run complex processing pipelines with little to no retraining.
- **Collaborate by default.** Discover systems to allow for ease of interaction at all times and collaborate with the owners and architects.
- **Reuse and automate.** Don't repeat yourself – don't build the same thing twice, don't do things manually that machines can do automatically!
- **Test everything.** Establishing quality data should not be an afterthought and should be designed and tested from the beginning.

5.3 Continuous Operations

DataOps encourages a holistic view where operators are able to see a living map of all the pipelines working together to serve the data needs of higher-order business functions. Operations staff are able to better manage the system as a whole, and yet drill down into problem spots as required.

Continuous Monitoring is an aspect of operations that is responsible for the surveillance of all components related to DataOps capability. It demands monitoring the interconnectedness of end-to-end data flows; in short, it monitors everything and maintains metrics about data that is moving and problems that occur 24/7.

This includes monitoring the operation and performance of applications and automatically resolving many types of data drift issues, reporting any

bugs encountered and applying any patches and fixes. It also verifies and controls access to data, identifies and coordinates cross-system impacts of data changes, and measure, analyze and report data delivery operations, quality, security, and execution exceptions.

5.4 Continuous Governance

Continuous Governance is responsible for establishing a data governance framework, a methodology and standards for Enterprise Information Management. It ensures that information strategies and policies are followed and ensures the information usability and protection for the enterprise information stakeholders. It is also responsible for establishing and overseeing compliance with rules, guidelines and procedures for ensuring the security and privacy of all types of information.

As the nature of data and regulatory pressures make the security and privacy of data more complex, enterprises must evolve from securing data at rest to adopting a posture of Continuous Governance, where all data is protected regardless of whether it is at rest or in motion.

Continuous Governance is also responsible for establishing and overseeing compliance with rules, guidelines and procedures for creating and maintaining enterprise metadata. This covers all areas of metadata management including documenting data assets, logging organizational responsibility and accountability, establishment of business glossaries, tracking of data lineage, guidance on data reuse, collection and usage of operational metadata and usage of metadata for audit and governance purposes. It confirms roles and responsibilities for creation, maintenance and stewardship of metadata and provides guidance on the application and usage of related software tools.

5.5 Continuous Data

Continuous Data is responsible for sourcing data both internally generated and externally sourced, publishing data services for end users and applications, and maintaining service levels and performance. All data should be collected accurately and delivered downstream with the lowest latency possible in an end-to-end approach. This includes building data flows and data pipelines that are able to adjust to data volume requirements that change on the fly.

Allowing data to flow from source to destination as fast as possible does not only mean high throughput while data is moving. It includes thinking about the key pieces that surround data, such as infrastructure. A key aspect of implementing continuous data is adopting data flow standards that avoid reliance on specific technology vendors. Change is a constant in technology so it is important to ensure that data solutions do not contain rigid boundaries and connections. The solutions should allow for (and encourage) the ability to handle infrastructure drift. The ability to measure and monitor data is extremely important in understanding where bottlenecks are when "continuous data" is not on time. This can be achieved by requiring all data flows to follow a set of monitoring and implementation guidelines. Continuous Data is at the heart of enabling the business to be as data-driven as possible.

In the real world, data exists everywhere: on edge devices, or servers, on the cloud or within applications. A common pattern of processing data is to bring the data from the systems that generate it into central data lakes and then run analytics on it. As enterprises move to real-time analytics, computation often needs to happen as data is generated or while in motion even before it arrives at destination systems. In DataOps, data integration efforts expect to execute in any environment that is necessary—on the edge on lightweight IoT devices, within the datacenter or cloud on highly resilient containerized infrastructures, or on other modern compute platforms. The complexities of managing and operating the execution environment are abstracted away and replaced by a focus on the outcomes of successful execution of the higher-order business function.

5.6 Program Execution

Program Execution is responsible for creating the overall strategy for the DataOps Platform to achieve defined goals and objectives. Program Execution improves the timeliness and quality of enterprise data flows. It includes planning and managing programs to deliver new or enhanced data discovery.

It oversees projects for delivery and implementation of operational, systems and technology solutions, developing detailed work plans, resource plans and milestone deliverables that address cross-enterprise collaboration. It also defines parameters to deliver a program's scope based on near-term architectures. Finally, it delivers change management requirements across projects.

5.7 Design Operations

This function applies engineering efforts to projects that have an opportunity to use the DataOps Platform and tools. Since it is part of many applications for any business unit, it also reviews and implements new software releases and supplies system fixes or patches as required and maintains an accurate baseline inventory of installed data systems, data stores and integration technology.

Design Operations is also where techniques like Continuous Integration and Continuous Development (CI/CD) and Agile methods for self-organizing are applied in a way that encourages rapid and flexible response to change.

6

DataOps Practices

DataOps is a new practice, but it also leverages and aligns with many traditional methods. Figure 6.1 builds on the Big Picture in Chapter 5.

Functions	DataOps Practice Areas
Continuous Design	Automate and Reuse
	Modern Architecture
	API Platform
Continuous Operations	Continuous Monitoring
	Dataflow Operations
	Dataflow Security Operations
	Data Drift Synchronization
Continuous Governance	Policy-driven Data Security
	Metadata Management
	Governing Data Quality
	Continuous Improvement (Lean A3/VSM)
Continuous Data	Data Marketplace
	Publication Services
Program Execution	Strategy Roadmap (Blueprint)
	Solution Architecture
	Business Case Justification
	Program Management
Evangelism	Agile Methodology
Training	Business & IT Staff Training
Design Operations	Continuous Integration & Continuous Development
Roadmap Business Analysis	Business Case
Predictive Analytics	Modern Data Analytics
Descriptive Analytics	

Figure 6.1 DataOps areas, functions and operational capabilities

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (47–70)

The table shows the DataOps techniques in blue and the traditional enterprise of industry capabilities in green. The rest of this chapter explains each of the DataOps practice areas and provides guidelines, methods or standards for applying them.

6.1 Continuous Design

Modern data architectures are often highly distributed and complex. Applications and use cases built up with a number of these complex systems have a high degree of interconnectedness.

Scaling data integration is often a continual challenge; every decision or action that requires a human in the loop is inherently an impediment to scale. Continuous Design encourages examining every step in the lifecycle of the data and setting up automation with appropriate checks and balances in place. It also designs in change. In other words, change is assumed to occur constantly. Loosely coupled architectures that abstract data endpoints and strong interoperability capabilities ensure that the data continues to flow regardless of constant changes.

6.1.1 Automate and Reuse

Designers should seek out the repeating patterns in DataOps processes and define components that can be reused again and again with just minor or configuration adjustments for requirement variations. And further, they should define components that be applied by non-designers, like business or data professionals; this is another form of automation.

Building individual data flow pipelines for solving point data movement projects is a typical pattern that has emerged from the days of legacy ETL technologies. When enterprises wanted to create any given BI report, a small team would come together and develop pipelines to pull data from a few operational tables, join them and feed them into a data mart.

Depending on the size of the data, these pipelines would then run every night, and operators would check in on the status of the run in the morning.

When another reporting requirement came up or another datamart needed to be created—perhaps another group would get together and create new pipelines for a different business unit—at times reading from the same set of tables and repeat the process over again.

Over time thousands of pipelines get created, often reading and writing the same data by different teams, often without knowing of each other's existence. Deep dependencies are formed within these systems and

the slightest change in one environment ends up having catastrophic and unpredictable impacts on other environments. Companies then have to create highly complex change management processes that take a long time to deploy. There are many instances where something as simple as changing a single field in a table could easily take up to three months because every group has to sign off after checking the impact on every one of their systems. When dealing with big data, where change is the only constant, these methods just don't work.

Continuous Design is a paradigm where data integration is always performed in context. The "big picture" is kept in view and is often the starting point of the design process. Metadata is at the heart of automation and reuse. Similar to how financial assets are automated with systems for sales, accounts receivable, accounts payable, payroll and others; at the heart of these systems is a structured chart of accounts. Metadata is the equivalent for any system that executes data-related processes; as such, it is a priority during your DataOps implementation to architect and implement a broad-based and widely adopted metadata management system.

Data Architecture diagrams that architects draw up are not just images on a slide deck but the living map of what is real in the system. When change happens to the underlying systems, for instance, data sources or destinations are added or removed, that change is immediately reflected in the living map. A dataflow pipeline designed to move data from a source to destination shows up in that map and is globally visible to anyone interacting with the environment. When the next project wants to access the same data, developers look to the living map or topology and merely reuse existing pipelines.

As the architecture grows in complexity or the number of pipelines moving or processing data increases, impact analysis is highly simplified by not only simple visual analysis of the interconnected map, but searches on the metadata and lineage that are naturally emitted from the map.

Another hallmark of Continuous Design is a focus on outcomes. Because of the complexity that is inherent in any of these systems, it is often easy to get mired in deep technical details. A system built for Continuous Design abstracts unnecessary technical details away so that practitioners can focus on solving the business problem instead. Legacy ETL technologies often require developers to define mappings where every field in the source and destination are manually captured. Even a single change to a field would require a pipeline to be stopped and the mappings redefined. Instead this new paradigm allows developers to merely connect to a source and expect the pipeline to automatically understand the underlying schema and semantics of the data.

As data changes, drift occurs, and pipelines should be expected to detect and react to these changes, updating destination schemas or notifying users as required.

6.1.2 Modern, Loosely Coupled Architecture

The DataOps architecture should be loosely coupled and easy to reproduce and deploy in a wide range of infrastructure scenarios. Traditional architects use dedicated computer hardware, software, data stores and networks which were the default technology 50 years ago. Some people still prefer this today, since they are familiar with specific products, such as servers from HP, client devices from Dell, databases from Oracle, software from SAP, etc. Dedicated computers are expensive, slow to adapt, and this approach does not relate to the current reality of constant change.

The first phase of loose-coupling and easy change was virtualization which, over decades, has been applied to hardware, data, networks, programming languages and operating systems. This movement enabled major improvements in speed and flexibility, especially with the advent of cloud computing. But it's still not flexible enough to meet today's demand for instant data from anywhere to anywhere and do so quickly.

The latest evolution is containerization, which may also be thought of as operating-system-level virtualization. A container packages up software code and all its dependencies so the application runs quickly and reliably across different environments. Container management platforms facilitate the organization and virtualization of software containers. We use containers to streamline delivery and avoid the complexities of interdependent system architectures.

Modern containers are scalable and can greatly improve the performance of widely distributed applications. Docker provides a way of describing, packaging and running container, while Kubernetes has been widely adopted for container orchestration. Both are open-source tools. Containers are much more lightweight, use fewer resources than virtual machines, and can easily be initiated on in-house infrastructures or anywhere in the cloud.

6.1.3 API Platform

Interoperability is imperative when dealing with the hundreds or thousands of applications that are typically present in enterprises. A central API Platform becomes an easy way for enterprises to expose aspects of applications and data both internally and externally while maintaining a high degree of control. Such platforms allow building a bridge between legacy

systems and modern applications and greatly simplify integrating third-party offerings with the enterprises' core services. Such platforms also allow creating a unified developer experience and allow unlocking siloed systems.

6.2 Continuous Operations

6.2.1 Continuous Monitoring

You can't improve what you can't measure is an aphorism that holds true in the DataOps world. In fact, Continuous Monitoring insists that systems that don't naturally measure and expose their own performance are inherently ill-suited to the challenges of this new world. Data Sensors within systems and dataflow pipelines must self-report on how they are performing at every step of the way. These metrics are not only useful decision support tools for developers and operators to examine the health of the environment, they are fundamentally what drives intelligent systems to be self-aware and self-heal.

Large interconnected dataflows are often monitored in isolation; an operator monitoring a pipeline in one area of the application will not know how it relates to another area of the app. The cascading effects of one failure are impossible to understand when the operators' only view is a tabulated list of individual pipelines.

6.2.2 Dataflow Operations

Modern enterprise systems need to deal with a very wide variety of data; IoT sensors emit readings, web applications produce events and messages, customers or partners send binary files and every flavor of database system (relational, NoSQL, graph, time series, etc.) contains different types of data.

Real-time insights or decisions generated by operational and analytical systems are fueled by events and data from these varied systems along with historical and contextual data.

Getting data in and out of these systems often requires the ability to run on a variety of platforms and utilize modern compute engines. Force-fitting traditional data integration tooling that imposes rigid semantics of batch or streaming to these modern problems often results in poor progress at best or costly recurring spends at worst. Decisions on data apply at various phases of its journey; sometimes at the point of origin, at times while it is in motion, and at other times in conjunction with other historical data. Flexibility in choosing when, where and how to use data to make decisions is the desired option.

Data has to be continuous. Continuous Data brings together several paradigms:

- stream processing for real-time predictive and preventive analytics,
- batch data to enrich or provide training data for forward-looking analytical systems, and
- data that drives business-critical descriptive analytics.

Metadata and lineage also play an important role in the effective use and veracity of the data. With the ever-increasing volumes and types of data generated, searching, cataloging, finding emerging uses and ensuring the source of truth of this data is as important as the data and its analysis.

Enterprises must look for solutions that make data available to a wide variety of users (data engineers, data scientists, data analysts, etc.) in an easy to consume manner with a full complement of metadata easily accessible. Building multiple technology stacks to handle batch or streaming data when each needs a high degree of maintenance and varied skill sets to manage is costly to set up and is a significant cost to operate.

6.2.3 Dataflow Security Operations

This aspect of operations is to manage and control services that implement and enforce Policy-driven Data Security. At a minimum, it includes executing and monitoring the security procedures and operations that are part of the DataOps Platform and the end-to-end flow of data, including data encryption and data masking. This discipline must also collaborate with other enterprise teams to update virus and intrusion protection and manage access controls to networks, computer sites, applications and data stores.

6.2.4 Data Drift Synchronization

Back in the day, a highly manual process of impact analysis and roll out of changes to schema was inevitable and often expensive due to the sheer number of steps, individuals and applications involved. The schema and semantics of data in modern organizations is constantly evolving, and significantly automating the change management process is the only solution for enterprises to keep up.

DataOps calls for automatic detection, notification and, when systems support it, synchronization of drift. An example of such a mechanism is when reading data from an IoT device; a designer may have set up five

columns on a destination analytics store. Over time as a new firmware revision of that IoT device was rolled out into the field a new sensor may have been added. Dataflow pipelines designed for DataOps would automatically detect the new fields on the incoming stream, notify the appropriate operations staff or developers and/or automatically alter the destination store so that it is able to store the new field without requiring any downtime.

Obviously not all stores are well suited for automatic schema updates—an operational database that is used by many applications expecting a certain schema would not fare well with unannounced changes. However, big data systems with schema-on-read semantics work very well with such automation.

6.3 Continuous Governance

6.3.1 Policy-Driven Data Security

Security policies and practices of yesteryear just don't scale in the current data-driven enterprise.

Over the last few decades security has been largely focused on IT infrastructure, software, overall defense architectures and the human factor. Data security has no doubt been important; however, it was mostly a function of designating certain entities as being Personally Identifiable and specifying high-level policies of how this PII data is protected. The actual detection and protection of entities was left to individual groups or developers and more often than not, these policies were not uniformly applied across the organization.

Modern architectures are sprawling and often hybrid. Data originates, is processed or stored in any number of systems—on edge, within the datacenter or in the cloud. Data also often goes across several hops, at times being processed on the fly or within ephemeral compute systems. As a result, many enterprises have to contend with not just a duplication of effort, but more importantly an overall weakened security posture. The advent of big data has further complicated this situation, as data comes in fast and the surface area of secure information and the overall vulnerability of data has increased greatly. Ephemeral systems add to the problem as data flowing into them is not always protected as terminal destinations typically are.

DataOps calls for a framework of constant vigilance and protection. Continuous Governance is about defining a security policy and automatically enforcing it whenever data flows through the enterprise. As data evolves and regulatory pressures increase (e.g., GDPR, California Consumer Privacy Act and others), staff such as Data Steward and Data Protection

Officer (DPO) need to be able to oversee a continually changing data security landscape. DPOs can set up policies but enforcement is generally left to engineers. A DataOps-driven culture gives roles such as the DPO the ability to specify security policy and expect underlying systems to automatically enforce the detection and protection of data.

Developers may have role-based options to override or generally tweak the detection or protection of data, but the default action should be automatic detection and protection.

Audits are usually performed for systems; however, it is imperative that every action taken on data too is logged in a non-repudiable format as well.

As the nature of data and regulatory pressures make the security and privacy of data more complex, enterprises must evolve from securing data at rest to adopting a posture of Continuous Governance where all data is protected no matter whether it's at rest or in motion.

6.3.2 Metadata Management

The sheer complexity and distributed nature of modern big data makes managing metadata a critical capability. Metadata typically exists in three forms.

1. Technical Metadata on the schema and data types originating from any data source or lineage, audits and statistics typically provided by data integration tools.
2. Operational Metadata capturing the frequency, timing and volume of sources, dataflows, and destinations that aid in optimizing the DataOps platform or supporting capacity planning.
3. Descriptive Metadata provides insights into how business uses the data, typically with the aid of cataloging and business glossary tools which may also include responsibilities such as data owners and stewards.

Traditional metadata repositories simply cannot handle modern data infrastructure due to the ever changing/evolving nature of today's infrastructure. Instead enterprises find many repositories in use across islands of data processes.

A DataOps-oriented organization will recognize that the manual curation and maintenance of metadata is not a scalable solution. DataOps calls for automatic detection and inventory of metadata. It uses machine learning to apply appropriate taxonomies on the data and facilitates visualization of relationships between data that's easily accessible across the

enterprise. Operational metadata is used not just by operational teams for managing and tuning dataflows, but also as a means to highlight the health of the systems to all consumers of the data. Metadata then becomes the vehicle by which the business collaborates on data.

To ensure that metadata is adding value and is managed effectively, the following steps should be followed:

1. Define the Metadata mission in terms of its purpose and expected business benefits.
2. Assemble the current-state architectural systems and technologies; the enterprise likely has established a number of Metadata details from past efforts so it is important to know what is already available and how it can be leveraged.
3. Identify opportunities to improve the current-state to better support the mission.
4. Define the target-state Metadata architecture and processes in conjunction with the Chief Architect and the Office of the CDO.
5. Develop a migration strategy to realize the target state and initiate the early-state projects.

6.3.3 Governing Data Quality

With the proliferation of fast-changing, fast-moving, structured, semistructured and unstructured data, data quality checks need a lot more than the consistency checks that are typical in the primarily static, structured traditional data world. Not only should checks such as accuracy and completeness work on data at rest; data in motion should be continually tested and validated. Consumers of data should be able to tap into data at any place within the architecture and be guaranteed that the data they are working with is accurate and verified.

Data drift (changes in structure and semantics) is all too prevalent and rapid in this new world. Data almost always exists in multiple locations and it is hard to establish what is considered the "source of truth" for that data. DataOps calls for continuous monitoring of data from its birth through the entirety of its lifecycle, no matter how and when it's transformed.

When working with data, users should expect that the data (and the pipelines it flows through) can prove its own quality and lineage no matter where it flows or resides. Non-repudiated metadata on every change that has happened should be available to easily validate what the enterprise deems as the authentic reference data.

6.3.4 A3 Problem Solving

A core principle of DataOps is continuous improvement through experimentation and learning. There are many approaches to continuous improvement, but this book describes two that have been found to be the most practical and powerful—Lean A3 and Lean Value Stream Mapping.

In Lean A3, the general notion is that there isn't a "perfect" way to do something. Rather, seeking perfection is an ongoing process, since there are always opportunities for improvement that can be uncovered using scientific disciplines, including these steps:

- Observe and describe a phenomenon or group of phenomena.
- Formulate a hypothesis to explain the phenomena.
- Use the hypothesis to predict something—the existence of other phenomena or the results of new observations.
- Perform experiments to see if the predictions hold up.
- If the experiments bear out the hypothesis, it may be regarded as a rule.
- If the experiments do not bear out the hypothesis, it must be rejected or modified.

A3 problem solving supports this principle by providing a concise summary of the quantified problem statement, performance history, prioritized root causes, and corresponding countermeasures for the purpose of data-driven problem analysis and management.

1. **Plan**: Establish the objectives and processes necessary to deliver results in accordance with the expected output (the target or goals).
2. **Do**: Implement the plan, execute the process, and make the product. Collect data for charting and analysis in the following "Check" and "Act" steps.
3. **Check**: Study the actual results (collected in the "Do" step) and compare against the expected results (goals from the "Plan" step) to ascertain any differences. Look for deviation from the plan in the implementation, and also, look for the accuracy and completeness of the plan to enable the execution.
4. **Act**: Request corrective actions on significant differences between actual and planned results. Analyze the differences to determine their root causes.

The term "A3" refers to the paper size used to document the problem, root cause, action steps, and results and is available in a template.

For further information, refer to Best Practice Resources for details on A3 Problem Solving and Management By Fact (MBF).

6.3.5 Continuous Improvement (Lean VSM)

DataOps activities generally involve multiple teams working together in a coordinated fashion to deliver a solution or provide a service. It is common to see five to ten different functional groups involved in an end-to-end process for a typical business analytics or application effort and frequently some of the groups involved are third-party organizations or outsourced service providers. We need an agile data process across multiple teams that is fast, minimizes waste, has high quality, and continuously improves.

Value Stream Mapping (VSM) is a technique that enables teams to achieve breakthrough performance improvements by creating a better overall flow for an entire process, rather than isolated improvements to a single point in a process. The value stream mapping activity creates a blueprint for applying problem analysis tools such as A3 Problem Solving and improvement events. It brings disparate teams together to gain a common understanding of the end-to-end process and helps people to see beyond the symptoms of waste and understand the root causes so they can make substantial and sustainable improvements.

The main focus of this best practice is to provide guidance on how to apply VSM concepts in a data delivery and analytical context. VSM concepts can be applied to virtually any repetitive integration process.

Step 1: Define the product/service

The first critical step is to develop a clear picture of the value stream to be analyzed. This can at times be challenging, especially in cases where processes are immature and informal, since every handoff in the internal company value chain can be considered a customer-supplier relationship and every customer has yet another customer whom they serve.

Many organizations when first applying VSM have poorly defined or inconsistent metrics to measure productivity, throughput, or overall lead time. The recommended approach in this case is to select a recent integration project or service delivery that is representative of a typical value stream flow and perform a detailed analysis of what actually happened. Then identify key participants from all the teams involved and arrange a kickoff meeting to explain VSM concepts in a non-threatening way.

Step 2: Create the current state value stream map

A key objective of VSM is to deepen one's understanding of a value stream by drawing a map of it. In current-state mapping this is done while observing the actual value stream as work is performed. For example, rather than asking "how is the work supposed to flow" we should ask "how does the work actually flow".

Value stream maps are often drawn by hand on paper, white boards, or post-it notes. The idea is to keep the mapping process real-time, simple and iterative and not let technology get in the way.

In order to do the VSM "on the floor", the VSM analyst could walk around and meet with individual staff involved in all the steps of a product/process. Alternatively, the VSM current state map can be developed in a facilitated working session with a group of staff that represents each of the steps in the value stream. If a representative from each group isn't available for a group session, then conduct the session with as many representatives as possible and follow up with the missing teams separately. Some participants, like a project manager, may be able to represent several teams.

Step 3: Create the future state value stream map

A future state map builds from the current state view and answers the question, "what could be?" The future state represents an image of a process as if it was designed and developed from start to finish as the ideal process. It answers questions such as:

- What does a perfect process look like?
- If there were no restrictions, how would we design our process?
- What would the process look like if all waste was eliminated?

Note that there are two kinds of non-value-added activities or waste: Non-Value-Add and Non-Value-Add But Required. Work that falls into the first category should simply be stopped. Work in the second category may present opportunities for improvement either by reviewing the policy to confirm that it really is required or by changing the process so that the impact on the customer is minimized.

The goal is to build a value chain where the process is linked to the customer through continuous flow, and each step gets as close as possible to processing only what the customer needs exactly when he or she needs it.

Step 4: Develop an action plan to address opportunities and achieve the future state

Improvement opportunities can be identified in one-on-one discussions with front-line staff, or as part of a facilitated group discussion. The basic substeps are:

1. Develop a list of opportunities without judgment or qualification (use brainstorming or Go-around in a group setting).
2. Prioritize the list by assessing the expected benefit from each of the opportunities. A second level of prioritization can be added by assessing the level of effort or complexity associated with each opportunity. (Use multivoting or Go-around in a group setting.)
3. Develop a specific action plan for each improvement opportunity using the A3 Problem Solving best practice or similar method.

It is also important to communicate the VSM findings and the action plan to the people that participated in the process so that they see the results of their contribution. This feedback process is critical for organizational learning and for setting expectations for repeating the VSM process on a periodic basis to identify new improvement opportunities.

For further information, refer to Best Practice Resources for details on Value Stream Mapping instructions and training.

6.4 Continuous Data

Modern enterprise systems have to work with a very wide variety of data; IoT sensors emit readings, web and mobile applications produce events and messages, customers or partners send binary and XML files and every flavor of database system (relational, NoSQL, graph, time series, etc.) contain different types of data.

Getting data in and out of these systems often requires the ability to run on a variety of form factors and modern access engines. Force fitting traditional data integration tooling that imposes rigid semantics for Batch or Streaming to these modern problems often results in poor progress at best or costly recurring spends at worst. Decisions on the data sometimes need to be made at the point of origin, at times while it is in motion, and at other times in conjunction with other historical data. Flexibility in choosing when, where and how to use data to make decisions is the desired end-state; dogmatic adherence to traditional, rigid architectures

are meaningless technological handcuffs that greatly impede progress in an ever-evolving data landscape.

Data is needed at the point of use, at the speed of need—data has to be continuous. Continuous Data converges the paradigms of stream processing that are imperative for real-time predictive and preventive analytics, with batch data to enrich forward-looking analytical systems and drive business-critical descriptive analytics.

Building multiple technology stacks to handle Batch or Streaming data is not only redundant but each system needs a high degree of maintenance and varied skill sets to manage. It is costly to set up and remains a significant cost center to operate. Teams managing these disparate stacks cannot guarantee consistent Data SLAs around quality, performance, delivery and privacy.

Enterprises must look for solutions that make data available to a wide variety of users (data engineers, data scientists, data analysts, etc.) in a manner that's easy to build, operate and consume with a full complement of metadata easily accessible.

6.4.1 Data Marketplace

Data has become the ultimate commodity and enterprises are racing to strategically exploit its latent value. While ideas of consumption and monetization of data are plentiful in forward-looking organizations, facilitating a means for producers and consumers to connect is often a hard challenge. A data marketplace is a powerful vehicle that allows enterprises to serve the needs of not only exposing producers but also facilitate the internal and external consumption of data.

The purpose of a Data Marketplace is to simplify the way data is shared between organizations and people using standard technologies. Data Marketplaces allow for easy discovery of raw, analyzed or streaming data. Discovery across the organization leads to reuse and social awareness of emergent use cases. This Marketplace infrastructure can handle millions of simultaneous connections and provide the confidence for organizations to share data sources and connected devices for specific use cases.

Typical objectives for the Marketplace are:

- Streamline data minimization and remove unimportant elements to accelerate approval of new data for analytics
- Showcase new data uses and scenarios for internal data users and partners

- Establish a central business metadata management capability linking business data assets to technical details
- Establish metadata management describing operational data flow and transformation processes

Benefits of the Data Marketplace include:

- Automate key processes, steps and replace ad hoc excel and simple file-transfers with a controlled central repository with supported tools
- Enable business transparency for the appropriate use of data for analytics
- Legal transparency for data privacy policies and reduced risk of non-compliance with privacy laws
- Enable Metadata for enterprise-wide use

A marketplace naturally becomes an ideal container to productize and monetize data and create opportunities for new business applications.

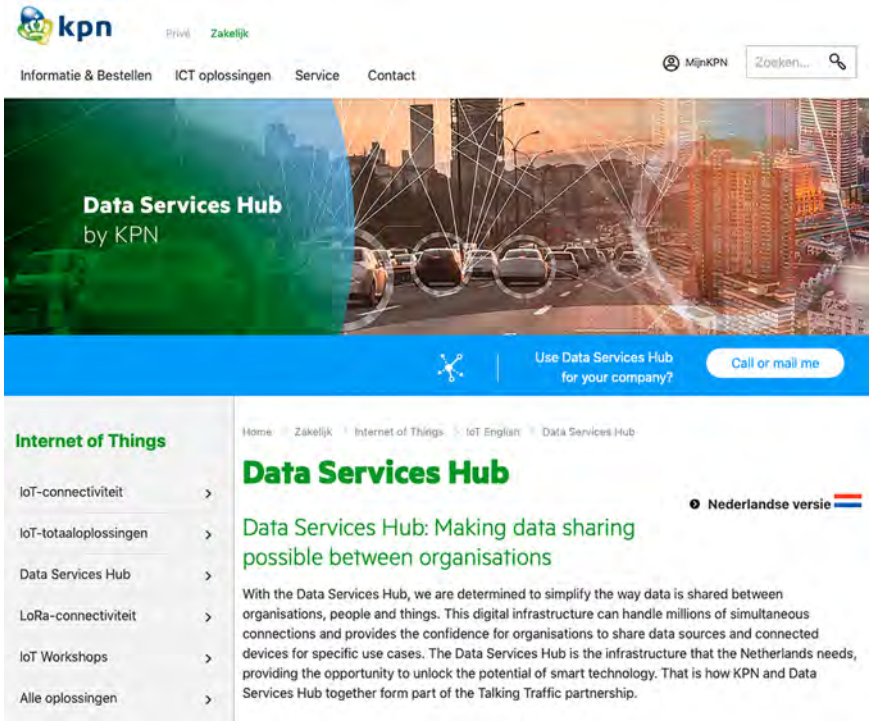

Figure 6.2 Example of a real-world data marketplace

The Data Services Hub above is an example of a Data Marketplace from a telecommunication company – in this example KPN from The Netherlands. See Appendix A for an outline of a Data Marketplace Proof of Concept.

6.4.2 Publication Services

Publication Services define the supported capabilities to finding, consuming and ultimately changing the contents of the Data Marketplace.

Historically for analysts or data-driven IT staff, access was flexible since staff could query via SQL and request exactly what was needed. However, it required users to know everything about the data being accessed. Application-driven access exposes data in fixed, well-defined models which were fine initially, but changes to data structures require application code to be updated. DataOps assumes data publications are based on the data itself where the client doesn't explicitly define their request and applications don't code the structures. Metadata in turn defines how data is detailed for publishing. If data structure changes, or if new metadata is created, it is reflected in the publication.

Use metadata and governance rules to control systems and data sources that expose data for publication. Metadata defines relationships between entities which then drives the automated processes that generate integrations.

The Publication Function is responsible for testing changes as they are deployed. Each consuming system must be able to test changes before being deployed.

6.5 Program Execution

6.5.1 Strategy Roadmap (Blueprint)

The Strategy Roadmap (sometimes referred as a blueprint) is a comprehensive approach for using model-based planning techniques to simplify and focus complex decision-making for strategic investments. (See also 3.1 Developing the DataOps Roadmap and Gaining Executive Support for additional information.)

The Strategy Roadmap capability follows defined steps to enable users to assess baseline capabilities, develop target architectures, identify transformation opportunities, and create migration roadmaps.

For further information, refer to Best Practice Resources for details on Strategic Roadmap Planning instructions and training.

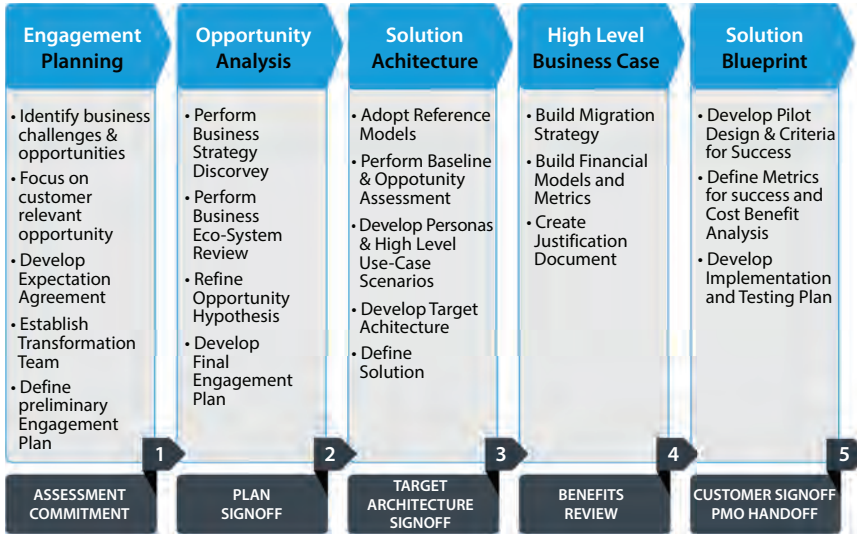

Figure 6.3 Phases for development of DataOps roadmap.

6.5.2 Solution Architecture

The Solution Architecture supports transformation program planning and systems strategy development. It does so by providing target system architectures and an associated roadmap of new and enhanced system capabilities to enable new operational capabilities and simplify and improve existing architectures. This includes creating system reference models, using these reference models for baseline assessment, conducting opportunity assessments for reducing diversification and upgrading system capabilities, creating integrated target architecture models to show future application systems, data stores, and information exchange solutions, and placing these capabilities on a migration strategy to align with related operational and technology programs. These target models are then used by planning functions to structure, organize, and govern related system programs.

A key model in support DataOps is a System Information Exchange (SIE) model which leverages an Enterprise Information Model not an Enterprise Data Model (see notice below under Model-based Business Transformation).

SIE models identify the primary components of application system and the information exchanged between those components. SIE models are decomposed into three primary groups:

- Systems – Applications that create or modify information
- Integration Systems – Connect data between systems or aggregate data
- Data Stores – Permanent information storage

SIE models capture information exchange opportunities between system modules and links them to business functions which are key inputs to data flows within the DataOps capability.

For further information, refer to Best Practice Resources for details on developing and using SIEs.

6.5.3 Business Case Justification

This capability is primarily about developing economic justifications for Digital Transformations. The reality is that data-based business cases are not just about data – they are about business processes. Here are a few real-world examples:

- Improved revenue via higher close rate comes from the use of full, unique client and partner 360-degree views with all key performance metrics around interactions linked to product IoT and customer engagement apps. Average benefit of all the assessed use cases in this category to date is $11.7M annually.
- Improved revenue from collections and billing results via a true, hierarchically linked counterparty entity risk and transaction profile with accurate reference data. Average benefit of all the assessed use cases in this category to date is $8.3M annually.
- Improved revenue through prospect identification and conversion is due to enriched and standardized record for demographic and interaction linkage. Average benefit of all the assessment use cases in this category to date is $6.4M annually.

In summary, to build a data-oriented business case to a) define the business functions or operation scenarios that are critical elements for your

strategy or objectives, b) prioritize the functions by scoping the ones that have the greatest opportunity for improvement, and c) collect data analytics and gather facts that precipitate the metrics to justify an investment in realizing the improvements. See the A2 Problem Solving capability for a template to structure a one-page decision summary.

6.5.4 Program Management

Program Management consists of three capabilities that work together to fully realize the result of DataOps platform and competency center.

1. **Funding initiatives** to facilitate business operations investments and decision-making. These investment decisions relate to processes, organizations, applications and technology infrastructure services. This capability defines the initiatives based on business strategies, operational opportunities, target architectures, and scope units from planning engagements. It prioritizes and makes funding decisions for initiatives based on their business value, scope units, and interdependencies within the constraints of financial investment guidance.
2. **Planning and governing programs** to deliver new or enhanced capabilities related to business operations, including processes, organizations, application and information systems, and technology infrastructure services. It defines projects to deliver a program's full scope based on near-term architectures from business transformation, renovation, improvement and sustained planning engagements.
3. **Program Management Execution** to oversee the delivery and implementation of operational, systems and technology projects, developing detailed work plans, resource plans and milestone deliverables that address the many interdependencies and change management requirements across projects. It is responsible for reviewing and approving all milestone deliverables and receives support from Enterprise Architecture in conducting compliance reviews to ensure that program architectures are compatible with enterprise targets. It also assures that the impacted operational functions are ready to support the new process, as part of acceptance testing and turnover.

6.6 Design Operations

Design Operations techniques are used in data-driven business projects. Once a supported DataOps Platform is in place, many, or most, projects can simply "configure" reusable components to implement business requirements. That said, there will always be times when new data sources, infrastructures, networks or protocols require some custom engineering. In those situations, Design Operations uses a host of tools that include modeling and simulation, software construction, configuration management and testing to build new capabilities.

A key aspect in support of DataOps is to engineer solutions to not only satisfy the immediate requirements, but to designbuild it in a way that the new solution components can be folded into the DataOps Platform and reused for future needs.

6.6.1 Continuous Integration/Continuous Development (CI/CD)

Since DataOps leverages the priciples of and includes the application of DevOps practices, CI/CD features and capabilities are going to be different than those commonly found in software engineering. Most of the time, when technologists say "CI/CD", they are thinking about the ability to check code into a repository that has an associated tool that triggers a build or deployment package.

DataOps calls for a Continuous Integration and Continuous Deployment model where tests are integrated into the design experience and any changes to pipelines result in automatic validation of outputs and further deployment into proceeding environments or rollbacks as necessary.

The reality of dataflows is that a failure could happen anywhere; in a nutshell it is about scoping efforts to deliver executable results on a short period, typically over a few weeks.

Networks or hardware fail, applications fail, pipelines fail and data changes, which breaks all previous assumptions. A DataOps culture prioritizes systems that are inherently self-aware of upcoming changes and are able to make autonomous decisions to self-heal. These systems aim to further simplify operations so that precious time isn't spent solving problems that may occur multiple times.

6.6.2 Agile Methods

Much has been said about Agile methods; they are about scoping efforts to deliver executable results incrementally over short periods. Even large,

complex solutions are divided into multiple small capabilities that are delivered independently in steps until the complete solution is working. Key benefits are:

a) the users are able to realize partial benefits and business value even before the solution is complete (in some cases the benefits may be months or quarters sooner);
b) the designers and engineers receive feedback early to adjust the design and validate that it is working and meeting user needs; and
c) Agile methods provide a more effective solution that delivers more benefits at lower costs compared with solutions that are built as a holistic single large project.

6.6.3 Modern Data Analytics

Modern data analytics expands beyond traditional Business Intelligence in four ways:

- **Any user:** The analytics user base extends beyond centralized BI personnel to data scientists and "citizen analysts" spread across an organization, as well as applications that make rule-based decisions.
- **Any data:** Structured on-premises databases are augmented by semistructured and unstructured data stored in flexible and cost-effective platforms such as HDFS, cloud object stores and even on smart edge devices.
- **Any method:** Analytic techniques evolve from SQL queries and data mining to data science featuring advanced analytics up to and including machine learning, deep learning and artificial intelligence techniques.
- **Any speed:** Batch transaction data is augmented with interaction data from numerous sources to identify real-time events, often delivered in microbatches or streams.

Descriptive analytics that provide summaries based on historical data, or Diagnostic analytics that answer the question "Why did this happen?" have been the mainstay of enterprises over the last few decades. While these are still critically important to run the business, Predictive and Prescriptive analytics have quickly become crucial to the future of most enterprises. Predictive analytics answers the question "What will happen?",

and Prescriptive analytics takes in a far broader and wide-ranging challenge of answering "How can we make something happen?"

Where and when analytics is done has also seen a major shift. A once common pattern of bringing the data into a data warehouse or data lake to derive knowledge has further evolved to being able to diagnose or predict at the point of use. In other words, analytics has become pervasive; it can happen anywhere—at the source, while data moves or in destinations.

Traditional technologies and practices are not designed to serve the needs of these forward-looking or pervasive analytics, and businesses looking to innovate or in some instances remain relevant need to adopt a culture shift that is inherently designed to solve for this new world.

Modern data analytics gives companies the means to innovate quickly, operate efficiently and better serve customers by leveraging data wherever and whenever possible. The Figure 6.4 architecture diagram from a healthcare enterprise illustrates this evolution. The traditional BI architecture of transactional databases and data warehouses is supplemented by a variety of data-driven applications serving a variety of business needs.

Figure 6.4 Example of business intelligence architecture from a Health Services Enterprise.

This architecture has a few notable characteristics. Data is leveraged from a variety of sources, including network-attached devices, medical equipment, or even social media feeds. These sources of data are largely unstructured, and control of the source data is fragmented across groups and companies. To support various analytics requirements, the underlying infrastructure has evolved into a complex web of fit-for-purpose components, each of which is susceptible to change at any moment. A myriad of users is supported, and these data consumers generate new requirements and produce derivative data sets that get pumped back into the architecture for further use.

7

The DataOps Platform

In an earlier section we talked about a model where a central team published a service catalog of permitted technologies and provided reference architectures to promote optimal use. Notice that it didn't mention an actual centrally managed technology stack. This is quite a departure from traditional Enterprise Architecture design patterns and is worth examining in detail.

In the traditional world, lifetime costs of software, hardware and the human costs of building and operating technology are some of the important economic factors that influenced technology decisions. In that world, smaller startup firms could focus on one or more factors to strengthen their competitive advantage. Larger organizations may have had certain latitude because of their larger budgets, but were constrained by skill availability. All in all, the multiplicity or lack thereof of technologies determined the fate of business. One couldn't easily spin up a new product venture or modernize a business process that required a completely different underlying technology because the central platform simply didn't provide it; or onboarding that technology would be a non-trivial cross-departmental adventure.

Furthermore, in the traditional world, applications and data management were managed by different groups. Applications may have been designed to run on web servers backed by operational databases. The Data Platform may have comprised of ETL systems, Data Warehouses and BI systems, Data Catalogs and Master Data systems including a means for governance. Applications and the Data Platforms were probably tied at the hip by ETL systems that read data off operational databases and fed them into the analytical warehouses. Agile software development or DevOps practices also contributed in the application space, but the teams operated in their own silos with no canonical agile practices in the Data side. The net result was that the velocity of achieving business objectives were often impeded by these organizational and technical dynamics.

Modern technologies and new consumption models such as "as a Service" have fundamentally changed the landscape of what's possible. As

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (71–76)

we've seen in earlier examples, business needs and architecture designs have also changed. Applications nowadays don't just work with operational data but utilize processed analytical data to alter app behavior. Analytical models access huge amounts of historical data that applications produced but could not use for day-to-day function; In short, applications had a huge amount of data but could not leverage it (other than for routine transactions) until an analytical process discovered insights from the data patterns which could be fed back to the app. And all of this might be happening in real time. As a result, businesses can greatly accelerate product or project delivery; applications can become smarter and analyze and utilize data in ways they weren't able to do before.

7.1 The DataOps Framework

A modern application may live in a space between two worlds: microservices applications running on containers using an event-based architecture to feed data into object stores; and dataflow pipelines training machine learning models based on incoming, historic or master data. And the application uses those models to classify or predict user behaviors in real time. The boundaries between applications and data management platforms have blurred.

Traditional data warehouses or unwieldy and complex Hadoop systems have given way to fully managed systems offered by cloud vendors. Such an amalgamation of applications and data systems that live in on-prem environments or on cloud have created a new set of requirements for how such technology stacks are architected. A modern DataOps Platform decouples logic from underlying implementations of the technology, is designed for agility, confidence in the data it operates, and indeed in some instances could even be virtual.

Let's say a health insurance company wants to be able to provide its customers with a fitness monitoring device. That company is doing this not only to provide customers a valuable new service that's deeply integrated with the overall healthcare experience the company provides, but also because it gives the company the next level of insights into their customers and lets them use aggregated data patterns to improve products and services. Their legacy data and application platforms aren't able to provide the flexibility required. Lets see how an enterprise may architect a DataOps framework to modernize and onboard such new projects.

The shared layer (DataOps Aware Control Plane) in such an architecture presents components or requirements such as collaboration, global

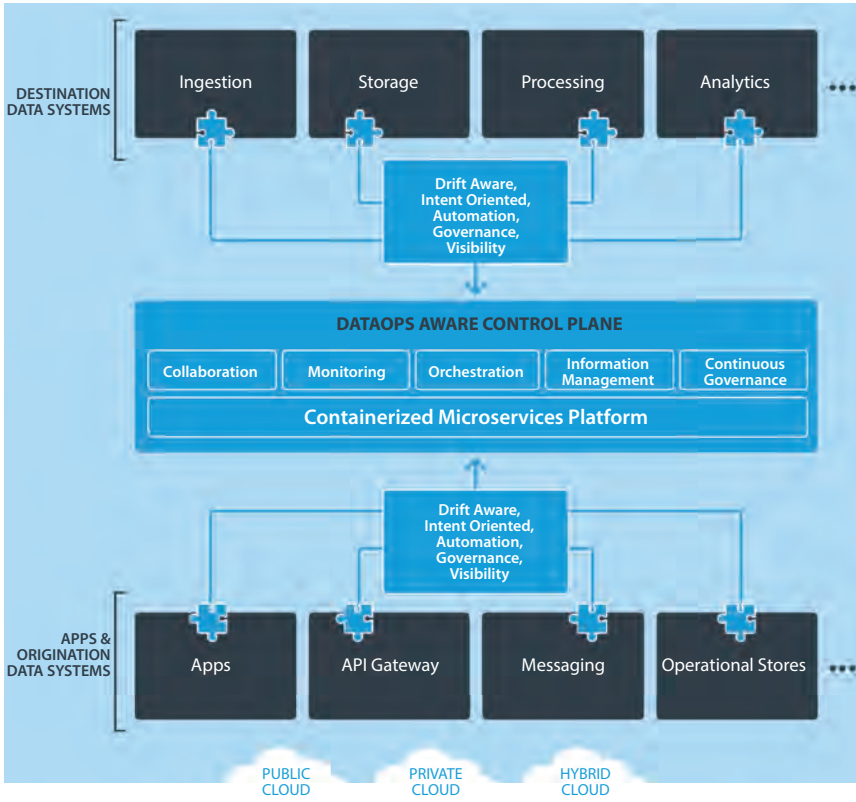

Figure 7.1 A framework with components providing one or more DataOps capabilities.

visibility into the interconnectedness of data and events flowing through the systems, detailed visibility into KPI's and operational metrics, scheduling and orchestration, information lifecycle management, data quality, governance, security and others. Also important is an infrastructure supporting containerized microservices that allow for ephemeral applications and workloads. Such a layer represents a central control plane that all other components within the architecture integrate with.

The components of the Data and Application layers represent the Data Plane. The functions performed by these components may vary, but admittance into the DataOps framework demands the following capabilities:

- **Agility and Reduced Friction:** Ability for all users based on their roles to quickly build and deliver functionality. Components should already be cleared for security, and architectural compliance. For example, Data Scientists wanting to

work with new datasets should be able to self-serve design and operationalize dataflow pipelines without needing to understand the nuances or operations of the underlying execution engine. The execution engine should come into existence when required and give up resources when not in use. A focus on agility and reduced friction is placed so as not to introduce unnecessary steps to the user during normal function.

- **Automation:** Operationalization of any activity within the platform should be highly automated. Particular focus should be given to reduce any actions that require a human in the loop. For example, if a dataflow pipeline needs to be promoted from one environment to the next, a Continuous Integration/Continuous Deployment (CI/CD) process should be used to automatically test the pipeline and deploy it into a production system without any manual intervention. Maximizing automation within these functions reduces the operational overhead of managing large and complex systems; it minimizes the human footprint and therefore human error.
- **Drift Handling:** All components should be able to detect when schema or semantic changes happen to the data and automatically take action to mitigate risks of working with erroneous information. The near constant drift in modern apps would require that dataflow pipelines don't require hard definitions at design time; instead they are automatically able to infer the schema and notify operators when unexpected changes happen. If the incoming data into an analytics pipeline changes, all the assumptions of the underlying algorithm would be incorrect, and results presented would likely be wrong. In such an environment the analytics pipeline should automatically detect drift and notify developers or end consumers where appropriate. In some instances, automatically updating downstream systems based on those changes may be possible as well. Drift handling is critical in ensuring confidence and building resiliency into the system.
- **Visibility:** Enterprises typically have thousands or tens of thousands of such dataflow pipelines at any given time. They may be pipelines that individual data scientists or analysts have created, or pipelines created by data engineers to support major business applications. Another key capability of a DataOps framework is to provide global visibility into the operation and health of these pipelines. Indeed, a mature DataOps framework

is one that is self-documenting and automatically draws out a map of all dataflows or applications; making sense of the interconnectedness of the data or events that wouldn't otherwise be possible without human intervention.

- **Governance:** Governance in DataOps deals with both the performance and health of the pipelines but also examines the data flowing through the pipelines. Deviations of previously stated SLA's of throughput or error rates automatically trigger alerts or actions where appropriate, operators and users are notified, or the system automatically adds capacity when required to compensate. These pipelines also automatically send technical metadata into catalog systems while the pipelines are running and if the schemas evolve unexpectedly. DataOps Platforms also provide a mechanism to define policies to automatically detect and protect sensitive data as it flows around the organization.

7.2 Building the DataOps Framework

Every piece of this reference architecture serves a different function and audience, and the capabilities presented above will manifest differently for each and every component. DataOps Architects and Engineers should use this framework to guide design and selection of these components to get to a state of Continuous everything in the DataOps Platform.

An ideal DataOps framework is set up as a shared service that projects can use if they don't need to reinvent that function. But in keeping with the DataOps ethos, the central team would provide guidance stating that project teams are allowed to use a different data storage or processing engine if the default one doesn't solve their use case. A project team would then be able to look through the published service catalog and choose different storage/processing technologies for their needs. For example, a project team may decide that the AI/ML capabilities provided by a different cloud vendor are better suited to their business needs than the one within the central platform.

Guidelines for technology selection for project teams should account for not only appropriate capabilities but should also consider the long-term impact of operating it. As mentioned in earlier examples, choosing a full SaaS solution removes the complexity of operations from the equation but it will need to be balanced with lifetime costs. A project team could use a global scale SaaS data warehouse that gives them the functionality they

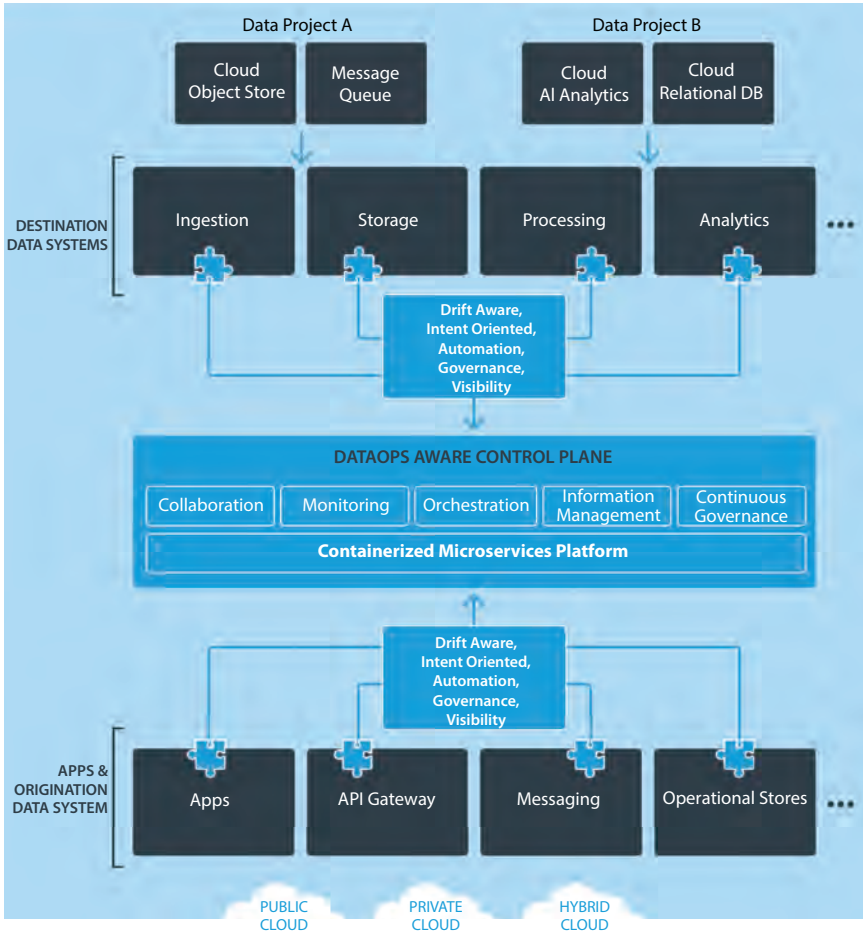

Figure 7.2 Individual projects using components better suited to their needs.

need instead of the central DataOps framework hosted Hadoop platform. Similarly by decoupling the control plane (that are typically managed centrally), from the data plane for moving and processing data (resides anywhere – on devices, in the datacenter, or on any cloud) the DataOps framework would give the project team maximum flexibility in architecting a solution that serves the best interest of that project's goals.

Every architecture decision has tradeoffs and DataOps Architects/Engineers and project teams should be given the widest latitude to capitalize on a loosely coupled DataOps Platform architecture that isolates the logical flow of data from the underlying complexity of technology and yet provides a bird's eye view into the health and operation of the platform.

8

DataOps Scorecard and Business Value

> *"What's measured improves"*
>
> **— Peter Drucker**

In the modern, constantly changing world; the only methodologies that survive are those that improve. A practice that is not expanding and getting better, is dying; something superior will come along and replace it. There are other uses for metrics such as providing transparency, rewarding staff and justifying investments, but the core purpose is to fuel improvements.

This book will not provide a "formula" for DataOps metrics, rather we define the DataOps Maturity Model, a structural approach to organize a Scorecard, a template to specify individual metrics, and a few examples to help make it real. To start, view the technology maturity in Figure 8.1 below.

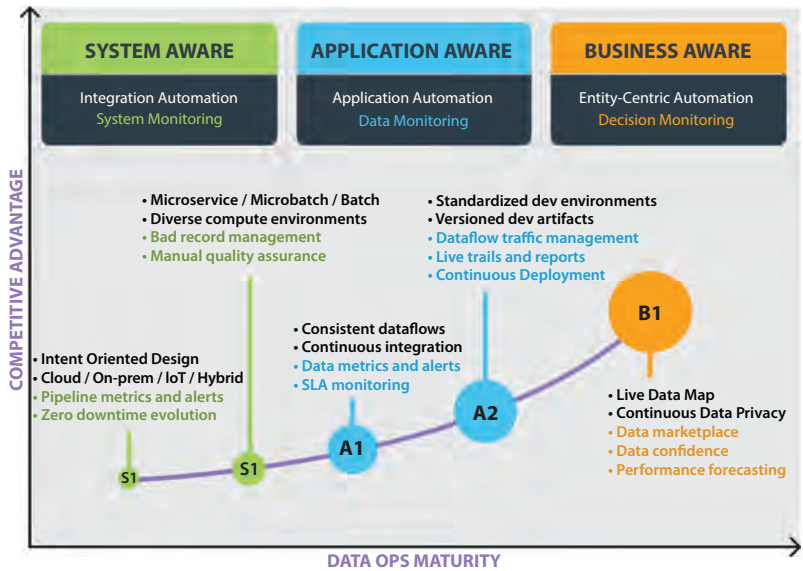

Figure 8.1 The journey to DataOps technology maturity.

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (77–90)

DataOps capabilities and processes are the predominant measure of maturity, but the technology perspective is extremely relevant because it improves over time to become more sophisticated and it streamlines and automates processes.

The first item to note in the prior figure is that there are three stages of awareness that culminate to yield competitive advantage: System Awareness, Application Awareness and Business Awareness. Each of these stages involve automation and monitoring functions, milestones, and more capabilities to clarify their differences. From a quick look, it appears that the maturity progress starts at S1 and is progressed through S2, A1, A2 and ends at B1. The reality is more nuanced since all awareness stages can approve somewhat independently.

The next figure shows some advantages of the DataOps stages enabled by the systems and technology supporting each of them. These items are potential areas to include the DataOps Scorecard.

Stage 1 **SYSTEM AWARE**	Stage 1 **APPLICATION AWARE**	Stage 1 **BUSINESS AWARE**
Data Productivity Systems • Fast acquisition of new data sources • Fast turnaround time for system integrations • Reduced complexity • Less resource investment needed for implementation • Improve application development and operational efficiency • Accelerate technology changes • Reduction in systems costs	Application Efficiency Systems • Improve operational control and visibility of applications and their dependencies • Improve certainty of data quality, availability and timeliness • Enhance application dependability and value • Agility in charging application requirements and operational environment	Business Confidence Systems • Direct line of sight between business and data operations • Control business led changes in agile manner and controlled cost • Improve business decisions concurrent with new data events • Gain valuable insight into customer's behavior • Find out what manufacturing costs are synchronized with production volumes • Gain advantage over competitors

Figure 8.2 DataOps maturity stage advantages.

8.1 The DataOps Maturity Model

In addition to the System-Application-Business progression, a more meaningful view is to consider the maturity of the DataOps capabilities, and their functions and practices, as defined in Chapters 5 and 6:

- Continuous Design
- Continuous Operations
- Continuous Governance
- Continuous Data
- Design Operations

8.1.1 DataOps Maturity Levels

We score the maturity of DataOps capabilities using a 5-level scale similar to the Capability Maturity Model (CMM). The CMM defines the degree of formality and optimization of processes and has been widely adopted, and extended, since 1980. The most widely used labels of the 5 levels are:

1. Initial (chaotic, ad hoc, individual heroics) - the starting point for use of a new or undocumented repeat process.
2. Repeatable - the process is at least documented sufficiently such that repeating the same steps may be attempted.
3. Defined - the process is defined/confirmed as a standard business process.
4. Capable - the process is quantitatively managed in accordance with agreed-upon metrics.
5. Efficient - process management includes deliberate process optimization/improvement.

There is a common variant of CMM that changes Level 4 to Managed, enabling adaptations without measurable losses of quality or deviations from specifications, and changes Level 5 to Optimizing, continually improving process performance through technological changes. This variant is closer to the DataOps maturity model, but still doesn't align with a key differentiator. Fundamentally, CMM assumes that you get to a relatively steady state where most things are consistent and there are incremental improvements; DataOps embraces continuous change, even in its own practices and processes.

For DataOps we add a "success factor" to each level to mention a key cause of its effectiveness and we introduce new labels for level 4 and 5, Flexible and Innovate, to highlight a principle way that DataOps is different from CMM variants.

1. **Initial**: The starting point for use of new or unproven approaches. The processes are ad hoc, and success is dependent on individual efforts and experience. Success is driven by competent and energized Individuals.

2. **Repeatable**: The DataOps process is documented sufficiently such that repeating the same steps may be attempted. Pockets of expertise exist across the organization and teams and individuals share their experience. Success is driven by sharing and effective collaboration across cross-functional teams.
3. **Defined**: DataOps processes and technologies are defined & confirmed as standard. Disciplines and named teams are in place that perform capabilities consistently enable cross-functional collaborations. Success is driven by consistency.
4. **Flexible**: While the DataOps processes, including the technology framework, is blessed by the enterprise and supported by expert teams, it is not enforced in a command-and-control fashion; everyone has the latitude to adopt elements that are different as long as new capabilities are structure to be folded into the Framework. Success is driven by Flexibility; building on consistency with elasticity to satisfy unique needs and evolving capabilities in new ways.
5. **Innovate**: The most mature DataOps maturity is one where the enterprise is open to emergent behavior and explore innovative non-linear solutions. It encourages an entrepreneurial approach to search for emerging processes, technologies, patterns or methods. The organization embraces emergent strategies to evolve rather than follow a predefined static state. Success is driven by changing DataOps in consequential, even radical new ways on a continuous basis.

To appreciate the difference, not that CMM's level 5 follows deliberate and known processes. Even in the Optimizing variant, it is still based on process improvements which are known strategies and linear changes; in other words, by applying rational and logical changes that have a similar structure and are not complex.

DataOps by contrast demands different types of changes which may be nonlinear and dramatic. This difference is needed for one reason – in this world of constantly changing everything, data is evolving so quickly that you can't keep up if you just apply linear methods. For DataOps to keep up with embryonic data abilities, you need an approach where progress does not simply develop smoothly from one stage to the next in a logical way. Instead, it makes sudden changes or goes in different directions at the same time. In fact, it may appear unpredictable, counterintuitive or even chaotic.

Furthermore, you need to not only continuously improve your processes, but also need a new way to improve your strategies. Traditionally, strategies are deliberate and defined for an intended target. Some strategies are effective and some not; strategies can be changed but each time they change, they have a new deliberate and defined state. In other words, traditional strategy changes are discrete management efforts controlled by top-level senior leaders/planners.

The strategic approach for DataOps is to encourage Emergent Strategies to appear from front-line teams as a bottom-up process. The strategies could be new/radical technologies, new/radical processes, or new/radical approaches to finding/delivering/visualizing data. Level 4, Flexible, sets the foundation for preparing for nonlinear changes from this statement from its definition:

> **While DataOps processes, including the technology framework, is blessed by the enterprise and supported by expert teams, it is not enforced; everyone has the latitude to adopt elements that are different as long as new capabilities are structured to be folded into the Framework.**

While Level 3 defines a complete and consistent set of DataOps capabilities and technologies and says, "do everything the same way", Level 4 says "it's OK to do whatever you want". Level 4 doesn't exactly say "do whatever you want" since there are still some defined principles that need to be followed, but we said it this way to highlight how different the DataOps maturity is from traditional methods.

To highlight the difference with Level 5, let's look at the meaning of entrepreneurial. Dictionaries describes an entrepreneur as "*a person who starts a business and is willing to risk loss in order to make money*" or "*one who organizes, manages, and assumes the risks of a business or enterprise*." For DataOps we aren't quite that literal and focus on entrepreneurial as a *mindset* and entrepreneurs as *those that think and do things differently*.

You can be entrepreneurial even if you are working for someone else and have the desire to be adaptable, flexible and think for yourself. Being entrepreneurial can mean knowing data management, integration and analytics inside out, and being able to exploit that knowledge to create new opportunities. It also means sharing ideas freely and celebrating failures as learning and growing experiences. It means thinking outside of the box and expecting the unexpected.

For an example, metadata for a large organization that captures data sources, their technical definitions, how the data flows (lineage), their relationship to business processes and security policies, can be huge and

complex. It's common to find companies that utilize millions of data elements in multiple tools that only highly technical staff can navigate. But what if one of your entrepreneurs could create a dramatic way for anyone in the company to view the universe of dataflows in an easy way to find what they need on a mobile app?

A similar challenge has been solved for the Air Transport Industry. Flightradar24 is an internet-based service that shows real-time commercial aircraft flight information on a map. It includes flight tracks, origins and destinations, flight numbers, aircraft types, positions, altitudes, headings and speeds. It can also show time-lapse replays of previous tracks and historical flight data by airline, aircraft, aircraft type, area or airport. It aggregates data from multiple sources and is available via a web page or mobile device apps.

If you haven't experienced Flightradar yet, give it a try on the internet or mobile phone. There is no cost and the experience is amazing! When I am waiting for a friend who is travelling by air to my city, I don't use the airline's website or mobile app; I use Flightradar to track the flight in real-time (which is more accurate than the airline and easier to use). An entrepreneur could develop this capability for your enterprise metadata to let business leaders or data scientists track the delivery of dataflows from their latest marketing campaign or IOT device tracking sales at new retail store, without having to create a static, universal data catalog that is out of date upon inception. The ideas are endless for people that are able to think outside of the box.

8.1.2 The DataOps Maturity Assessment Tool

In preparation for this book, we have collaborated with several DataOps professionals, including leaders from StreamSets Inc., to develop a tool to measure the maturity for a company, or a portion of an enterprise. The Framework is based on this book with emphasis on the sections describing the DataOps Functions and Practices plus the prior portions of this chapter. A graphical view of the framework is shown in Figure 8.3 below.

This tool gives organizations the ability to quantify their maturity at a point in time, use the definitions of the security levels to define a future target, and periodically measure if the maturity is improving as desired. The tool can also be used to benchmark the maturity of different companies who has assessed their capability and are willing to share their score.

Further details for completing the assessment and interpreting the results are explained at https://go.streamsets.com/dataops-assessment.html.

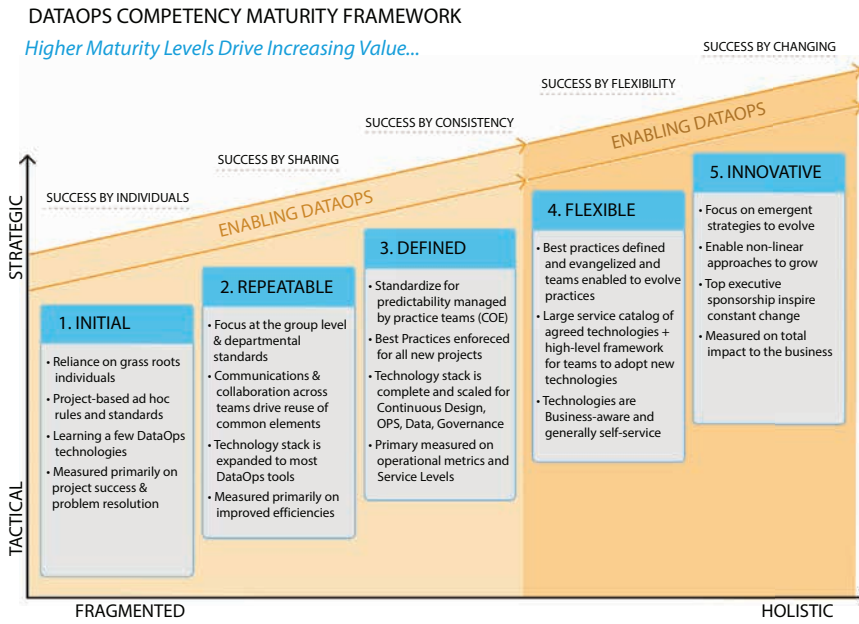

Figure 8.3 DataOps competency maturity framework.

8.2 Establishing Lean Metrics

The DataOps maturity progression is based on a number of success factors:

- Continuous improvement across the entire data value chain
- Investment in process-based structures, tooling and automation
- Data-driven processes that emphasize reuse and rapid delivery
- Cultivation of the right technical skills and practices, and a culture of accountability and agility

In order to manage and achieve these factors, broad-based agreement is required in order to drive organizational change and gain support for necessary investments and cross-functional process changes. As a result, a fact-based approach to improvement development and decision-making based on measurement and monitoring of metrics is essential.

The general process for developing and maintaining DataOps metrics is as follows:

- Define the objectives to describe the purpose behind the collection of metrics.
- Define an initial set of metrics to be considered for full implementation.
- Start recording metrics as soon as the initial set has been defined.
- Publish metrics to make the facts visible to targeted stakeholders.
- Establish an initial baseline measurement for each metric to set the stage for measuring progress.
- Define targets for process improvement.
- Refine metrics on a periodic basis to implement tactics to achieve the improvements, measure the results of the improvement initiatives, identify new or improved metrics, and repeat.
- Evaluate when metrics need to be overhauled due to a major change in DataOps strategy or when the definition of success has changed meaningfully. Measurement is critical to DataOps, but organizations must ensure the metrics themselves do not shackle them to outdated approaches or processes.

8.2.1 Define the Objectives

The overarching purpose for using data is to improve business bottom-line results, enhance operational processes, enhance customer experiences, implement new data-driven solutions or transform aspects of the business capabilities. Looking more directly at DataOps capabilities, the purpose for measuring, collecting and analyzing metrics are:

- Respond to analysts or user data needs faster and faster
- Continuously improve DataOps processes to make them more effective
- Plan for data changes and automatic response to structural, semantic and infrastructure data drift
- Automate by leveraging reusable assets, metadata-based processes and self-service
- Optimize complete end-to-end data delivery and not just individual steps

In order to support these objectives, start by clearly describing them, linking them to your corporate strategies, list the key capabilities and

processes that support them, and define the most important priorities. This is then the source for defining the actual metrics.

8.2.2 Define Initial Metrics

For a first pass of defining the list of metrics, start with the top priorities from the previous list of processes, and identify the "pain points". For example, if a high priority objective is faster response to customer demands and a top pain point is a long backlog, the following metrics should be on the list:

- Number of requests in the backlog
- Average time items stay in the backlog until they are:
  - Delivered
  - Cancelled by the customer due to resolution time
  - Deleted by the COE
- Period of time the list of backlog items is reviewed, recounted or re-prioritized

Mature DataOps teams will have a broad range of metrics that support a wide range of processes to grow the culture. An initial list will commonly include:

- Lead Time
- Cost per dataflow
- Service requests completed per period (x)
- Service requests in backlog
- Reusability percentage of new pipelines
- First time through percent
- Data processed per day or per second
- Dataflow service levels deviations
- Value-Add Ratio
- Data drift changes (and error count and error rate)
- Proportion of end user requests addressed on a self-service basis
- Analyst and business user satisfaction

In order to record metrics, publish them, establish the baseline and improvement targets, the process for each metric must be identified. Table 8.6 below shows 15 parameters to be considered, and an example following the Lead Time metric.

1. Measurement Name	Lead Time
2. Objective	Continuously reduce the time from a customer request and complete delivery
3. Information Needed	Record the date, time and method of data request, customer ID, time of start of solution, time of solution delivery, time of solution acceptance.
4. Relationship to Organizational Objectives	Reducing Lead Time is key to supporting the overall enterprise object for real-time marketing; using data reported instantaneously so marketers can make decisions based on information on what's happening in that moment. Instead of creating a marketing plan in advance and connect consumers with the product or service that they need now, in the moment.
5. Description of Proposed Measurement	The primarily metric is to measure the time from a customer request until completion of delivery. Interim starts of the process are also used to identify steps in the delivery process that could be improved.
6. Possible Outcome(s) of Measurement	Lead Time measures may be very consistent or show a high degree of variation, they might be constantly improving showing faster solution delivery, or they could be show that lead time is increasing. Each of the outcomes provide useful information to the COE staff.
7. Proposed Data Sources	Three main sources for data are 1) DataOps website for service requests, 2) Agile spring backlog and burndown chart, and 3) manual tracking by COE staff for informal requests from email, phone call or hallway discussions with stakeholders.
8. Frequency and Methods of Data Collection	Daily registration of start and interim events as they occur. Lead Time for a customer request is formally logged at the point of customer acceptance.
9. Proposed SLA or other Industry Benchmarks	Historical lead times were measured in months; 9-12 month lead times where very common. Lean Manufacturing benchmarks shows that a 90% lead-time reduction is possible within a 1-year period with a concerted effort to eliminate waste, automate routine processes and minimize non-value-added time. That would suggest a lead time target of roughly 1 month.
10. Process Description and Success Criteria	Lead time is the latency between the initiation and completion of a process. For example, the lead time between the placement of an order and delivery of a new car from a manufacturer may be anywhere from 2 weeks to 6 months. Lead time measures the time elapsed between request for a data and delivery, thus it measures your production process from your customer's perspective. Cycle time starts when the actual work begins on the unit and ends when it is ready for delivery.
11. Processes to be Created or Modified	Define existing processes requiring change.

Figure 8.4 Metrics definition template. (*Continued*)

12. Performance Target and related actions	The target goal for DataOps is that 80% of user requests for new data sources will be resolved within 2 weeks.
13. Review Process	The metric process will be reviewed for completeness and effectiveness ever 6 months
14. Publishing detail and frequency	Lead Time will be officially published on the first Monday of every month and included in the DataOps website, the regular newsletter to stakeholders, and printed on the wall near the Agile team review
15. Limitations to Data Collection	Current known limitations e.g. currently manual to be moved to automated as soon as possible.

Figure 8.4 (Continued) Metrics definition template.

8.3 Reusability Metrics

Reuse is a critical capability to enable the DataOps vision to expedite data connectivity, so it justifies being highlighted as a main section in the Scorecard chapter.

To define 'reuse', it may include templates, software assets or shared libraries. For example, a data cleansing rule to remove leading spaces in text records that is developed once as a microservice or pipeline fragment can then be used in more than one deployed solution as an example of a typical software asset.

Reusable components can be sourced from different areas

- Project – Developed by a specific project and re-used within the scope of the project, and possibly by other projects in the enterprise
- Project by Design – The project identifies the need for a series of reusable components and plans the build before the main build
- Open Source and Corporate Communities – Developed as part of a predefined project to build reusable components in advance. This may require a business case to justify the investment based on corporate funding policies and the size of the investment.
- Technology Community – Reusing another organization's component

The components can take several differing forms:

- Design Patterns – A high- or low-level design that is implemented with some difference e.g. operating system
- Templates – A set of objects prebuilt but copied into the new implementation
- Policies – A policy or rule that can be encapsulated in a pipeline stage or process and pervasively enforced
- Shared Objects – An existing set of objects that are used 'in place' e.g. called directly
- External Calls – Calls to existing components using a different technology
- Object Generators – Such as "wizards" to generate Pipeline mappings and workflows

Within the DataOps domain the following represents a list of possible sources/definitions of reusable components:

Reuse Type	Description	Forms
Pipeline Source and Target	A pipeline facilitates the flow of data from a source system to destination systems or databases. Pipelines contain stages to process extract, transform and deliver data flows and use multithreaded pipelines to handle large volumes. Pipelines can be duplicated to quickly create a similar but partly modified data flow.	It is common practice for these to be stored in the DataOps Platform controller and related tools.
Pipeline Fragments	A pipeline fragment is a component or stage used in pipelines. Use pipeline fragments to easily add the same processing logic to multiple pipelines and to ensure that the logic is used as designed.	Once fragments are published, their logic can be reused in any number of pipelines.
API	Application Programming Interface (API) to allow applications to communicate with one another. Or a publicly available web-based that returns data, likely in JSON or XML.	Public (external), private (internal) or partner (specific organizations) APIs.
Web Service	Web Service is a communication method between machines over a network for exchanging simple of complex data within pipelines. Typically using HTTP as the protocol and possibly SOAP, REST, and XML Remote Procedure Calls (RPC).	Usually external to the project as an External call. Also can be within the technology community at large.

Figure 8.5 List of reuse types. (*Continued*)

Reuse Type	Description	Forms
Third party Function	A new or existing procedure or function. This will ideally be encapsulated within a single Pipeline reusable component to allow of impact analysis. Use of an existing procedure is common to avoid redevelopment. This type of component is developed to take advantage of specific functionality in a technology platform or system to manage objects or for performance reasons.	Usually external to the project as an External call.
O/S script	A set of common corporate scripts is very common. Flat File archiving is an often-used example.	Can be a shared library where a single script is parametrized for many uses or a pattern where a script is copied and then modified

Figure 8.5 (Continued) List of reuse types.

8.4 Automation and Reuse Example

One of the best ways to demonstrate the value of DataOps is to measure its impact economically. The example below shows one type of automation by quantifying the value of reusing fragments, which are parts of a segment of pipeline, and computing the money saved per hour.

The Activity Log fragment at the top of the list is significant functionality that is useful in many dataflows. It took 40 hours for the software engineer to design and develop the first one and it was done in a way that could be used by a wide range of dataflow sources and targets. Each time it is used again for a subsequent pipeline it simply a matter of the developer copying it which only takes a few minutes. The savings therefore is 40 hours each time it needs to be copied rather than re-developed at a load time of $120/hours which translated into $48K in one month! By applying this technique to all types of reuse, you are able to show that the DataOps COE is saving millions of dollars per year. This is not a fictitious amount – this is the reality of what DataOps is truly able to deliver.

Reuse Asset Name	Fragment Description	Reuse Effort Saved	Reuse in May	Costs Saved by DataOps
Activity Log	Generate metadata for activity log	40	10	$48,000
Validator	Validation Fragment	20	7	$16,800
Kafka Destination - JSON	Parameterized Kafka Destination – JSON Format	20	5	$12,000
Kafka Original - JSON	Parameterized Kafka Origan – JSON Format	8	4	$3,840
Mongo Destination	Parameterized Mongo Destination	16	7	$13,440
Change Logs Post Processor	Keep or Prune Logs Entries (Flag-Driven)	16	2	$3,840
Retention Comparator	Compares, drop, and logs affected member IDs	16	2	$3,840
Aggregate JSON	Takes flattened JSON in, performs two nested group by operations based on parameters. Parameters: - group_by_keys_1 - Top-level group by fields – group_name_1 – First subgroup name (subgroup includes all fields)	8	2	$1,920
ECOM Action Validation	Validation	8	3	$2,880
Error Handler	This pipeline fragment is a generic error handler	8	3	$2,880
Error Handler Catch All	This pipeline fragment is a generic error handler	8	3	$2,880
Kafka Origin - JSON	Parameterized Kafka Source	8	4	$3,840
Mongo Distinct Query Executor	Groover Evaluator that queries a collection and returns the distinct values. It also generates a no-more-data event after the query has completed.	16	3	$5,760
Mongo Query Executor	Groovy Evaluator that queries a collection and returns the distinct values. It also generates a no-more-data event after the query has completed.	16	1	$1,920
Mongo Lookup	Mongo lookup	8	5	$4,800
Mongo Original	Parameterized Mongo Origin	8	3	$2,880
Policy Executor	Groover Evaluator that executes data retention policies for against Mongo	8	2	$1,920
			Total Savings by Month:	**$133,440**

Figure 8.6 Example of reuse metrics scorecard report.

This example is one of many benefits of DataOps automation. More powerful examples could be the costs avoided by automated drift-handling so that business operations don't stop when source data changes. COE management has responsibility to recognize significant results, work with impacted users to quantify the value, and communicate the results stakeholders.

9

DataOps To-Do List

The specific activities that should be done for implementing a given DataOps capability and COE varies for each organization and leadership team. The timing and speed of completing the activities is dependent on the scope of resources that are available to do the work and the amount of money that can be allocated. That said, there are a number of key items that are relevant to most implementations and should be considered.

Chapter 3 introduced the structure and process for creating the DataOps Roadmap. This chapter builds on the process by detailing key activities at various stages of implementation as shown in Figure 9.1, which shows major milestones or deliverables that are typically created in the initial, foundation and optimizing phases as companies advance their maturity. The following sections in this chapter provide a brief explanation of each item.

Roadmap	Phase 1 Initiation	Phase 2 Foundation	Phase 3 Optimizing
Organization & People	• Purpose & Goals • COE Team Members • Core Training • Extended Team and Stakeholders • Roles, responsibilities & service interactions	• Metadata Management Office • Communication Program • DataOps Roadmap to Stakeholders	• Business data literacy maturity • Business owners share data quality
Process & Policy	• Vision and Mission for DataOps COE • Quick-win Projects • Communication Portal • Business Value Metrics	• Dataflow Reuse • Best Practices • Website for eCommerce • Metrics Tracking and Reporting	• Metadata to External Community • Continuous Improvement Process • Total impact to the business
Technology & Infrastructure	• Dataflow Tools • DataOps Platform • Metadata Current State Architecture	• Metadata Target State Architecture • Templates, Fragments and Tools Architecture • Security Framework	• Performance Tuning & Troubleshooting • Advanced Metadata

Figure 9.1 Milestones and to-do items for DataOps implementation.

John G. Schmidt and Kirit Basu. DataOps: The Authoritative Edition, (91–100)

9.1 Implement a DataOps Roadmap

9.1.1 Clarify Purpose & Goals

Leaders should clarify the purpose and business outcomes in terms of objectives and measurable outcomes. The DataOps strategy should also define current challenges in terms of "pain points" and frustrations.

9.1.2 Define a Reference Architecture

The Reference Architecture defines the terminology and labels to be used for describing each of the identified capabilities and the associated best practice guiding principles and characteristics. The architecture framework is used to define Business, Operational, System and Technology view: note that architectures are not just about technology and should be applied to all aspects of the enterprise. The framework is even more helpful if it is used to describe current-state, interim-state and target-state models as components are planned to change in the roadmap.

9.1.3 Migration Strategy Planning

Define Programs and related Projects that are aligned with the Target Architectures and scheduled over a sequence of phases. Completing a specific transformation may require multiple projects over several years.

9.1.4 Transformation Roadmap

Action plan defining a high-level map for moving from the current state to the target state of business, operations and technology. The documented roadmap includes milestones as outlined in Chapter 3.1.

9.2 People & Organization To-Do List

9.2.1 Document COE Team Members

Capture the list of staff directly forming the DataOps services, including their name, role and contact information. This also may include the direct or indirect management reporting lines and, if appropriate, also a target-state view of the team structure if significant changes are planned for a future date.

9.2.2 Define Extended Team and Stakeholders

List of teams and individuals that support or have significant dependence on DataOps including name, title and organization. See Appendix B, Glossary of DataOps Dependent Capability Functions, for potential teams or stakeholders. Also, potential services provided by extended team members or other functions are shown in Figure 9.2.

Service Category	Specific Enterprise Service to DataOps Team
Information Security	Approve Data Flow Security Practices
	Define Security Rules & Standards for Data Flows
	Define Data Protection & Privacy Regulations Rules
Data Management Operations	Maintain Data Stores (Archive Non-active data, Access Controls, Data security Monitoring, etc.)
	Set up new data stores
Metadata Management	Define Metadata strategy and operational Practices
	Define Metadata Architecture
	Enforce Data Catalog Record Keeping
Architecture Management	Maintain Enterprise Data Model
	Enforce Enterprise Architecture for DataOps Capabilities & Solutions
BI, Analysis & Performance Management	Define data delivery needs for BI
	Monitor and deliver Key Performance Indicators
Data Governance & Data Quality, Reference Data, Master Data	Maintain Common Reference Data
	Enforce DQ Rules for Data Flows
	Provision Business Metadata
	Support for Insightful Use of Information Resources
	Resolution of Common Master Data
Biz Ops Program Management	Governance of Program & Project Plans
	Forecast of Potential Programs
	Identify Data Stakeholders for individual programs

Figure 9.2 Potential services provided to the DataOps team.

9.2.3 Define Roles, Responsibilities & Service Interactions

Formalize the responsibilities or job descriptions of each COE role. The DataOps services could include:

- Name of service offering
- Description or narrative if not self-descriptive
- Who is the buyer: Focus on the actual consumer of the service and not activities making up the service
- Value proposition: Described from the perspective of the consumer
- Cost of service: What is the actual cost and how would the cost get recovered
- Ordering mechanism and delivery process: How would users request the service including any specifics about the process.

Potential services by the DataOps COE to the enterprise are shown in Figure 9.3 below. This list should be rationalized with Figure 9.2 above as the list of services from or to the COE may change depending on where the enterprise assigns responsibility.

Service Category	Specific DataOps Service to Enterprise
Information Security	Employ Rules & Standards for Data Flow
	Define Security Processes for Data Flow
Data Management Operations	Resolve Client Reported Data Issues
	Define needs for Data Stores
	Registration of Data Lineage
	Registration of Operational & Technical Metadata
Architecture Management	Apply with EA Standards
	Define Technology & System Architectures for DataOps Solutions
Infrastructure, Applications, & Platform Operations	DataOps Services needed by IT Infrastructure Support
	Applications DataOps Services
	Coordination of Application Information Exchanges

Figure 9.3 Potential services provided by the DataOps Team. (*Continued*)

Service Category	Specific DataOps Service to Enterprise
System Solution Delivery	Design of Data Associated with Business Events
	Data Store and Data Integration Design
BI, Analysis & Performance Management	Support for Insightful use of data Resources
	Provide BI Support Services
Data Governance & Data Quality, Reference Data, Master Data	Apply DQ Management Practices
	Enforcement of Business Intelligence Management Practices
	Data Delivery Services
Biz Ops Program Management	Development of Data Architectures for Programs Biz Ops Investment Prioritization
	Data Flow Fulfillment of Biz Ops Programs

Figure 9.3 (Continued) Potential services provided by the DataOps Team.

9.2.4 Communicate DataOps Roadmap to Stakeholders

Develop a presentation or wiki to communicate the DataOps Roadmap to supporters and stakeholders. To have the DataOps capabilities be used and adopted, data consumers and users need to know about it. Start with the most important stakeholders first and roll it across the enterprise and partners as needed.

9.2.5 Define a Communication Program

In addition to the communication events in the prior section, develop an ongoing series of updates either by newsletters, videos or promotional website.

9.2.6 Establish a Metadata Management Office

This could be a centralized team responsible for Metadata Management or set of distributed teams collaborating on various aspects of business, operational and technical metadata. Regardless of which structure is used, metadata is an essential capability needed for a mature DataOps capability.

9.3 Process and Policy To-Do List

9.3.1 Define Vision and Mission for DataOps COE

Make sure the DataOps vision aligns with or supports the enterprise vision. To collect input, meet with senior executives to collect their perspective and review the most recent company annual report and similar materials. Define a concise mission statement for DataOps using this template:

- **Our mission** is to [purpose]
  - **by doing** [high-level initiatives]
    - **to achieve** [business benefits]

9.3.2 Complete Quick-Win Projects

Identify a few projects (or just one) for early implementation. The projects may be simple or be more complex, but in any event they should implementable in a relatively short period. The success of early projects will fuel support to continue adoption and growth.

9.3.3 Define Metrics and Maturity Tracking

Easy metrics are measuring work items and deliverables by the COE. Things like the number of new data sources added to the data lake by month and the trend over time; or the number of dataflow pipelines in production and the volume of data they move on a daily/weekly process. Another important metric is the maturity of the DataOps process, which could include the number of steps in end-to-end flows that are automated and the degree of adoption across the enterprise ecosystem.

9.3.4 Define Dataflow Templates for Reuse

Reuse is a key aspect of enhancing the speed and quality of data delivery. From the perspective of process steps, one of the first steps to enable reuse is to create a set of templates or forms; these may be simple steps using Spreadsheets, PowerPoint or Word templates.

9.3.5 Document Best Practices

Define processes that work best in the enterprise for developing dataflows and make them operational by activating them. Once processes are

formalized, look for opportunities to automate them and continuously improve.

9.3.6 Define Business Value Metrics

Define measures of interest to business leaders. For example, in addition to tracking how many new data sources are added per month and the number of dataflows, measure how they are helping to increase sales, improve customer experience, reduce operational costs, lower enterprise risks, etc.

9.3.7 DataOps Wiki for eCommerce (accept service requests)

Build a website to explain the services that the DataOps COE provides, and as the team matures, use the website to assist consumers to initiate service requests and allow them to serve themselves.

9.3.8 Report Tracking for Metrics (business, roadmap, reuse)

Determine how to communicate metrics (paper, website, charts posted to walls, newsletter, etc.), the frequency, and target stakeholders (management, front-line staff, external consumers, etc.).

9.3.9 Extend Metadata to External Community

This builds on the earlier task to establish the Metadata Management Office to focus on methods and tools to make the business, technical and operational metadata available to support analysts, data scientists, auditors, data engineers, and others. It is common to use a range of metadata tools and repositories to satisfy a broad range of users.

9.3.10 Perform Continuous Improvement Process

This is an ongoing task for many staff both directly within the COE and dependent stakeholders. If there is a one-time task for the DataOps leader, it is to a) define processes and tools for identifying improvement opportunities, b) teach staff to apply the techniques, c) team with senior management to define methods to empower and encourage staff to raise opportunities on an ongoing basis, and d) lead the effort to initiate a specific improvement process and use it as an example to others.

9.4 Technology and Infrastructure To-Do List

9.4.1 Establish the DataOps Platform

This task is an obvious one. However, a less obvious item is whether to acquire the DataOps platform from one or more vendors, or to build the systems with internal software engineers. The "buy" option is quicker to get started and provides capabilities that are proven. One the other hand, the "build" option will provide a more tailored capability, but it might take some time to complete the implementation, and it demands a dependency on the skills and experience of internal sources. Nonetheless, an executable set of systems is fundamentally necessary to achieve a highly effective and automated DataOps capability.

9.4.2 Launch Communications Portal

This technology milestone is for enabling collaboration between internal collaborators and also with partners and external data professionals. The intent is to coordinate collaboration activities by leveraging communication capabilities such as video, voice and messaging, workspace, etc. It also concludes collaboration sessions by generating minutes and surveys, and by utilizing communication and content management capabilities to deliver recordings, meeting content, etc.

9.4.3 Define Metadata Current State Architecture

Every enterprise has some historical information and tools to support "data about data," which may be based on spreadsheets created and maintained by individuals, or scripts capturing data on a mainframe computer or a set of in-house application services, or data models created by data architects. The first task is to collect an inventory on what metadata already exists and what tools and processes are currently in place.

9.4.4 Define Metadata Target State Architecture

Once the current state metadata sources and tools is captured, the next step is to define a future state architecture and the services that will be useful for the typical range of data consumers.

9.4.5 Define Templates, Fragments and Tools Architecture

This task builds on the activity to Document Best Practices from 9.3 Process and Policy To-do List above. The earlier section defines what the DataOps processes are (in terms of words and visual process models) while this task implements tools, templates and systems to automate the processes.

9.4.6 Implement Security Framework

This to-do list is about capturing the range of security rules and standards defined by the enterprise and implementing them into usable software components or code. The best solution is not a 100-page document defining security, but rather a software framework that automatically ensures that teams that use it will automatically apply the rules correctly.

9.4.7 Define Performance Tuning & Troubleshooting

DataOps needs to maintain the usability, integrity and privacy of the contents in the data storage, archiving capabilities and other data systems within the IT Infrastructure to facilitate troubleshooting and system enhancements. The Performance capability establishes and maintains performance measures and targets (the "Score Card") for all critical data capabilities. Furthermore, it visualizes, locates and operates all virtual and physical data for cloud service performance, availability and cost. It monitors the performance of infrastructure and platform assets by measuring parameters such as network tomography and route analytics, as well as any system metrics such as CPU, memory and disk usage.

Appendix A Case Studies

A.1 DataOps Capability at Umbrella, an Online Global Marketplace

This case study is based on an online global marketplace which we renamed as Umbrella Corp to protect the identity of the actual company.

A.1.1 Executive Summary

The legacy approach to data sharing facing Umbrella was ad hoc, slow and inefficient, leading to a tangled web of data pipelines. The goal of the effort was to increase efficiency, lower costs, and make actionable data easily accessible for sophisticated real-time analytics.

To ingest their large and varied data sources to the data lake, they initially planned to build a custom solution. They soon came to realize that a hand-coded approach was unsustainable and was preventing them from meeting the goals of the data lake. New data sources were taking weeks to onboard and the backlog of jobs was growing to the point where it was unsustainable. In addition, the reality of changing schemas, known as data drift, was leading to endless maintenance of sources already feeding the lake.

To create a sustainable solution to the data lake ingestion problem, the data team launched an exhaustive across-the-board survey of available ingestion solutions. They chose StreamSets to create a real-time self-service data exchange that ingests data from all their sources, including social media, SMTP servers, JSON, XML, unstructured and binary data, into a new unified data lake, available to all business users in real time.

After implementing their DataOps capabilities based on the StreamSets solution, Umbrella was able to fulfill over a year of backlogged data ingestion requests in less than a month. They now use an automated form-based process to make new data sources and streams universally available as soon

as a request arrives; the need for backlog was eliminated. In addition, data drift is automatically handled with schema changes propagated into the data lake without manual intervention. Business partners can now leverage all of the company's data to both innovate and improve operational effectiveness.

A.1.2 Challenges

The business challenge for Umbrella was to speed innovation by unifying access to data from its separate divisions and trading partners. Each entity has its own systems, schemas, data centers and staff. Each independently ingests and processes multiple real-time data streams and databases. Their starting point was a "tangled web" of logical data flows between business units which I characterize as an Integration Hairball.

Data movement between companies was historically custom-created and managed by scarce and expensive engineering talent, usually to meet a very specific need. The aggregate result of these ad hoc pipelines was a spider's web of data movement pipelines that was not only inefficient to build but also difficult to track, expensive to maintain and next to impossible to govern. Data was being moved to numerous locations and in some cases making unintentional round trips. In one example of data flowing hither and yon within the Umbrella ecosystem, OSCORP gave data to ACME, who then gave it back to OSCORP, who then moved it to Nakatomi Trading, who then gave it back to ACME, who then gave it over to Tyrell to market the product.

The customized pipelines also could not handle the data drift or schema evolution inherent in the company's data sources due to the constantly changing nature of its businesses and its decentralized structure. Any kind of data drift in the upstream schema would cause pipelines to fail and required urgent maintenance work to get pipelines back online and maintain a functional data operation.

Also, the complexity of the data sets was daunting. Across the company ecosystem, there are many source types: RDBMSs like Oracle, Netezza, MySQL, DB2 as well as APIs, flat files, Kafka topics and more. And every entity has more than one of those implementations. And each source had multiple schemas. For example, at ACME, their Oracle RDBMS contains one schema that has roughly 1,600 tables. It was clear that the traditional approach—hand coding—was not going to scale gracefully.

In short, data was not democratized, not real-time, not consumption-ready and not reliable due to data drift. Umbrella could not integrate or leverage the latent data power that existed in its 25 walled-off data silos.

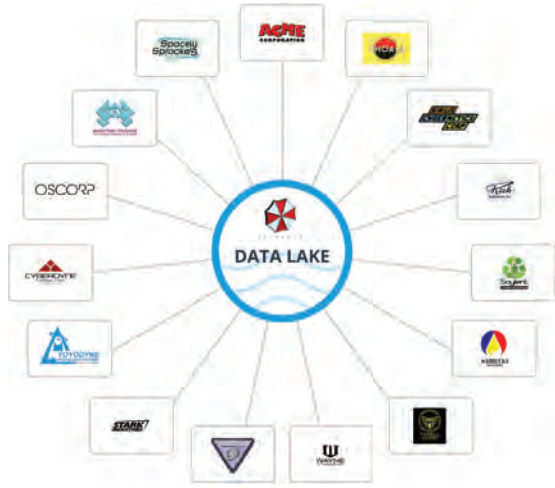

Figure A.1 Exchanging data at scale with independent data silos.

A.1.3 Solution

To achieve the ambitious goal of unifying all data systems, Umbrella decided to create a central data lake which would contain everything, so any team member from any company could access and analyze timely and trustworthy data from any of its peer companies.

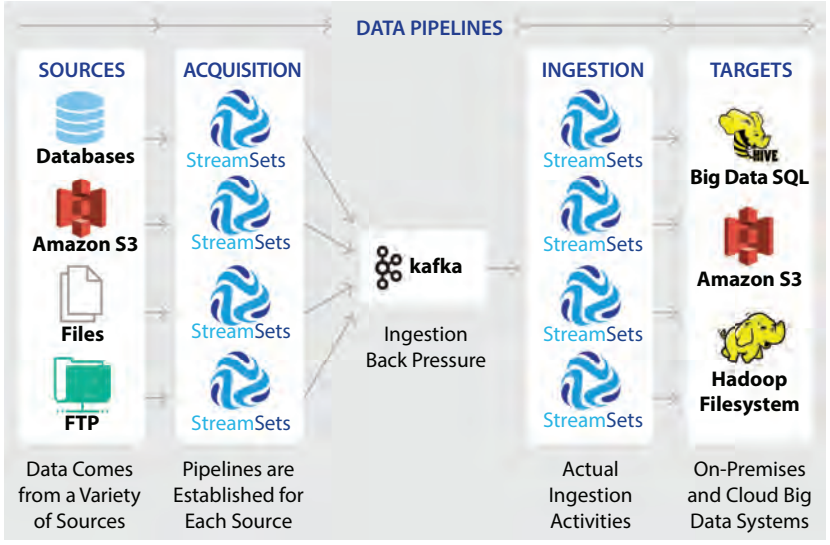

Figure A.2 The umbrella DataOps StreamSets architecture.

The Umbrella ingestion architecture leveraged StreamSets DataOps technology to create a heavily automated self-service ingestion platform. A key architectural concept was the decoupling of data acquisition from data ingestion.

Pipelines were created to move data from sources such as relational databases, files, FTP servers and cloud environments into a shared-service, on-premise/cloud hybrid big data system. A load-balanced tier of pipelines fronts an Apache Kafka message bus to ensure scalability and security. It also handles both impedance mismatches from different sources and provides back pressure to the dataflows. A separate set of pipelines was used to consume data from Kafka and send it to various destinations including an on-premises HDFS/Hive data warehouse and Amazon S3.

Key attributes of the DataOps technology solution were:

- Acquisition pipelines are decoupled from ingestion pipelines with Kafka acting as the interceding message bus.
- Acquisition pipelines are source-specific, retrieving data from a particular source, but all acquisition pipelines send data to the same destination.
- Ingestion pipelines are completely generic and are used to route data coming from all acquisition pipelines.
- New acquisition pipelines can be brought online without any changes to the ingest pipeline tier.
- Data moving between data centers is encrypted in transit using TLS.
- Errors are handled dynamically.
- Data standards are applied to the data in motion including compression, file formats, partitioning schemes, row-level watermarks and time stamping.
- Auto-creation and ongoing management of Hive schemas during data flows:
  - New tables and partitions are created automatically
  - Upstream schema changes/drift are synchronized with the downstream Hive warehouse
- Ingestion for new sources is completely automated and any field changes are dynamically and automatically reflected, eliminating pipeline breakage and the maintenance cycles that had plagued Umbrella's custom-coded processes.
- All data is fingerprinted with hash records. Avro schemas are created automatically on the fly, a massive productivity improvement over the legacy model of manually mapping every field in every source.

Use of StreamSets has allowed Umbrella to massively streamline data ingestion. Rather than each ingestion job becoming a new IT project in the backlog, now a new job is automatically triggered by completing a form that specifies the source and clicking "Build." The pipeline is automatically built in StreamSets and, leveraging its REST API, machines are deployed automatically. This all occurs within minutes of the initial request.

StreamSets handles data drift dynamically so that when a database changes its schema, for example by adding and removing columns, StreamSets deals with this seamlessly and automatically.

A.1.4 Results

The impact of the DataOps implementation was immediate! Before DataOps, the data ingestion team had been processing in the range of 25 to 50 jobs on a typical month. Once the DataOps tools and methods were applied, the throughput increased to **350 jobs in a single month**; an order of magnitude or 1,000% improvement in productivity! The massive backlog of requests from the business units—some more than a year old—was cleared in a couple of weeks, freeing up the backlog with no increase in staff.

Furthermore, the data availability latency—how quickly a new data source can be made available in the data lake—**fell from 21 days to four hours**. User demands for data can now be immediately satisfied as fast as they arrive!

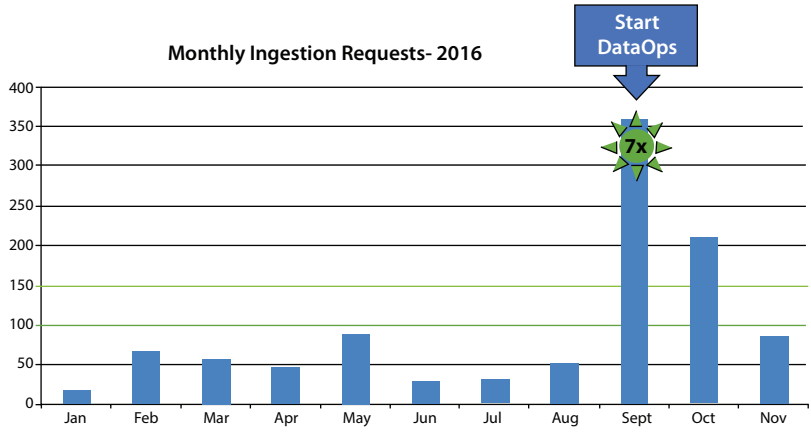

Figure A.3 Data throughput – almost an order-of-magnitude improvement via DataOps methods.

A.1.5 Long-Term Benefits

Instead of experienced high-value engineers, interns can now manage the process, no coding required!

Access to data is centralized, federated and democratized. Data is freed from fragile, custom-coded batch processes. *Anyone* from any Umbrella company can access the data.

Data democratization enables business innovation. Prior to StreamSets, Umbrella business units were highly selective with their requests for new data sources. They knew that the data teams were overloaded and would not be able to service them for weeks or perhaps months. With the StreamSets COE, these limiting barriers have been torn down; any data source can be made available to the entire corporation *immediately*.

Now that Umbrella had unified their data, they can move onto new innovations.

A.1.6 Summary

The problem Umbrella faced would be easily recognizable to data professionals in many large companies – the chaos created by organizational complexity, data variety and the fragmented ownership and governance of the data.

DataOps and StreamSets technology provided a unique path to the automated onboarding of new data sources and reliable continual data ingestion. The key features that Umbrella was able to take advantage of included:

- Any-to-any ingestion infrastructure to eliminate hand-coding.
- Data drift handling to reduce pipeline maintenance due to schema changes.
- REST APIs allowing deployment automation using Ansible, Puppet, Chef and other tools.

The challenge Umbrella faced could be applied to any industry as companies endeavor to unlock the value of their data. Improving how enterprises make data universally accessible and immediately consumable should be an early priority since so much depends on it in today's data-driven world.

A.2 DataOps in R&D at a Health, Pharmacy and Biotech Company

This case study is based on a global Health, Pharmaceuticals, and Biotech company which we have named as INGEN (International Genetic Technologies) to protect the identity of the actual company.

The Health, Pharma and Biotech industry is enormous, including segments such as Diagnostic Laboratories, Doctors and Health Care Practitioners, Hospitals, Medical Devices, Medical Supplies and Equipment, Outpatient Care Centers and Personal Health Care Products. R&D is an extensive process which requires recording, storing, and analyzing massive amounts of test, experimental and clinical data. This case study applies to any healthcare enterprise that has an R&D organization regardless of whether they are involved in new drugs, vaccines, medical devices, clinical trial research, health execution, or healthcare products.

A.2.1 Executive Summary

INGEN has three global businesses that research, develop and manufacture innovative pharmaceutical medicines, vaccines and consumer healthcare products. Its goal is to be one of the world's most innovative, best-performing and trusted healthcare companies.

At the time of this writing, the enterprise is three years through the DataOps transformation and has a very mature practice. During the journey, they have made a series of policy and process changes, invested in hardware and software to build DataOps systems, and assigned and acquired dozens of staff. The DataOps team is now at the point where they are recognized at the heart of the enterprise strategy and adding tremendous business value that far exceeds the investment to date.

At INGEN, they implemented the hybrid model with the creation of the R&D Data Center of Excellence. The team is comprised of employees focused on building the information platform, data movement, data curation, data standardization, enablement, data science, and overall program management. The DataOps COE leads the construction of the platform to provide analytics-ready data across R&D and works with research scientists regarding the use of data to answer specific questions.

A.2.2 Starting DataOps Center of Excellence

The journey for INGEN started with an executive memo three years ago to initiate the DataOps capability. The memo was extremely significant since it:

- Reinforced the support from an executive sponsor and the entire senior management team, and
- Announced the key vision and charter summarized as combining *all* data sources in a centralized shared infrastructure.

This change had a tremendous impact because previously each independent team was creating and maintaining their own data, making it difficult to share enterprise-wide knowledge.

Memo: Data Source Load approval

As you are aware, the vision of the Data Centre of Excellence is for INGEN to be able to better utilize internal/external data to transform our business by providing better insight and improve decision making. A key step towards this vision is being able to copy all of INGEN's data from multiple source systems we have onto the single IIP (INGEN Informational Platform). From this platform, users will be able to conduct exploratory data analysis to answer specific business questions.

To ensure that data integrity, privacy, confidentiality and transparent disclosure is respected, on behalf of the IMT (Management Team) we have set up a "Data Access and Security Committee" which includes senior leaders from across INGEN, including Paul Spencer in his Chief Strategy Officer role and other IMT members. Their remit is to:

- *Review and approve access to data sources across INGEN to enable data to be copied onto the IIP,*
- *Determine the level of access for users where confidentiality or regulatory controls may apply, and*
- *Provide guidance on linkage of data from various sources which may have different regulatory and security requirements.*

At the first meeting in April, we discussed and reviewed the request to begin copying INGEN data onto the platform including one of the most significant types of data we possess, product R&D

quality data. After some good discussion the committee approved and agreed that ALL INGEN data sources could be copied onto the IIP. INGEN DataOps COE and relevant I.T. team members will have access to this data to support the loading from source systems.

At a subsequent meeting on June 13, the committee granted permission for access to a limited number of individuals for innovation development. This memo serves as the approval, on behalf of GMT, for the loading of data from the source systems and access approval for these limited individuals (included in appendix).

Please note that discussions on the process for user access, and "who" should have access continues. The committee meets bi-monthly and will seek input from their constituencies where needed to be able to approve decisions on access. Any further decisions will be communicated as appropriate, but feel free to contact me if you have any questions.

Best wishes,
John Smith on behalf of the INGEN Data Access and Security Committee

The memo allowed functional owners to officially approve and gain access to share enterprise-wide data. The value of breaking down their functional barriers and sharing information is now estimated to create billions of dollars of profit for the enterprise.

Sharing data across independent teams may or may not be the strategic driver for *your* enterprise, but it is critical for an organization to explicitly articulate the strategic needs for data that will drive DataOps results. An appropriate memo by an executive is a simple example, but in this case it was a powerful milestone.

A.2.3 Challenge

One of the major challenges was the development of the steps required to share data across the organization. The DataOps COE created a brand-new Hadoop-based solution as the information platform. One of their main focus areas at the time was centered upon the movement of the data, and one of the major goals was to load at least 90% of the structured data within a six-month period. The only way to achieve this goal was through automation. They selected a technology which would allow them to quickly move data, have access to source data from a variety of structured sources, and move the data onto the platform.

For data acquisition, the COE selected StreamSets to move data from the source systems to the Hadoop platform. A key benefit of the StreamSets technology is the ability to create pipelines with additional automation. Rather than manually create each of the pipelines, the team implemented the capability to dynamically create and execute all the pipelines directly from the inventory of data sources. Using this approach allowed them to automate the data ingestion process across thousands of structured data sources leveraging hundreds of thousands of pipelines to load 6 petabytes with 4 petabytes refreshed every week.

A.2.4 DataOps COE Advice

The DataOps COE was structured in four major areas. First, they established an information platform to be the single location for integrated data.

The second part was to enable an extended multidisciplinary team. Some team members understand the science in discovering a new medicine, some understand R&D work, while others understand the clinical trials needed in developing a new medicine to assess efficacy and safety. With their deep understanding of the data, they act as a translator between the scientific need and the data and analytics environment. The team correlates data with business needs and attempts to consolidate it.

The third component of the team handles data science and analytics. It is very important that the R&D organization defines and derives value from the data. But in many cases, the R&D department requires assistance in understanding the tools needed. For this reason, INGEN created a small data science and analytics team to be a catalyst in assisting the needs of the R&D scientists.

The fourth part of the team focuses on overall program management and operation. Building a well-architected, production-level data and analytics environment can be complex. INGEN established a program management office (PMO) early to assure the proper level of project management and to coordinate project plans across the subteams. In effect, they have a project management COE inside the DataOps COE. In addition to the PMO, the team also provides internal communication, training, finance, contract management, and leverages the common user experience program.

Finally, as part of the extended COE team, there is also a solution development subteam which acts as a type of packaging area. It is important to understand the core team does not develop software; however, as the business requires a solution to provide access to data, the solution development team leads the effort in producing the dashboards, queries, or analyses by leveraging the technologies available on the platform.

A.2.5 The Solution

Establishing an information platform to support the data and analytics needs of a large organization requires the integration of several technologies. The solution is not a single technology from just one company, but rather a best-in-class ecosystem that delivers a production-level, large-scale platform. The foundation for the INGEN platform is the Cloudera Enterprise Data Platform, which provides Hadoop and additional components, including security, Spark, Hive, Kafka, search, and platform management.

The DataOps COE addressed their data ingestion challenge using StreamSets. For data curation, they used Tamr (www.tamr.com), which uses machine-learning to rationalize data elements and align data to industry data models. There were a lot of data sources, and a lot of similar data sources due to data fragmentation. By using machine learning, they could understand and make sound decisions by utilizing the data itself, rather than having people sitting in a room arguing over data attributes. INGEN use analytics on the data to improve the traditional Extract-Transform-Load process.

To complete the solution, the COE also used Trifacta to enable business user data wrangling, Waterline Data as the metadata repository, ZoomData to provide dashboards and data visualization, and Kinetica to enable GPU database capabilities. The power of a big data and analytics platform is in the collaboration across the ecosystem of technologies and service providers, not in a single component.

A.2.6 Lessons Learned

One of the more important lessons learned at INGEN is that it requires integration of several technologies to create a large-scale, successful data analytics platform; and this is not easily achieved. Many new technologies have so much capability they can be difficult to integrate and form a best-in-class ecosystem. It is important to understand the level of work in bringing all of this together to deliver a full production-level solution.

Another key lesson is the significance of developing use cases. Rather than selecting a single use case to use for the start of the program, a better approach is to select a portfolio of use cases from across the business to serve as a base for the program.Addressing 10+ use cases, rather than just one, drives very different decision-making related to the approach and dimensions of both the environment and processes.

Rather than "painting yourself into a corner" with a single use case, the portfolio approach drives a model that enables an easier progression to additional use cases, and production-level items from the start.

Appendix B: Data Marketplace Proof of Concept

Following is a list of scenarios that a Data Marketplace could be expected to support in an enterprise POC:

1. Search for Data Asset (DA)
    a. Leverage meta info to find DA and use it to answer questions or create new insights
    b. Search for Meta Data
        i. Data Models
        ii. Data entities and source systems
2. Create New Data Asset
    a. Self-service Remix (recombine existing DA)
    b. Onboard new DA that has been enabled for self-service: e.g., weather data from web or other internal/external sources
    c. Onboard new DA that requires a one-time enablement for self-service
    d. Onboard new DA that requires a one-time custom integration development effort
    e. Publish DA for users: Report, Definitions, etc.
3. Distribute/Share Data Asset for System Use
    a. Self-service replication/copy of DA; ad hoc or scheduled
    b. Make DA available via API, Virtual Table
4. Manage Data Asset Security/Compliance
    a. Publish/Change/Maintain Rules
    b. Enforce Policies (Geographical access, masking, archival/purging,…)
    c. Monitor Security

5. Capture/Maintain Metadata
    a. Connect to a new source of metadata and capture it in the catalog
        i. Data about data
        ii. Data models
        iii. Business definitions
        iv. Responsibilities & accountabilities
        v. Policies (Privacy/Regulatory/Security/other)
        vi. Data categories & annotations
        vii. Data Lineage (OOTB and Custom sources)
        viii. Custom resources and connections (i.e., IT assets, service portfolio)
        ix. API

There are a number of common use cases for metadata. Some are available out of the box (OOTB) and may require configuration and administration activities, but no significant development effort. Other use cases can be implemented using metadata but require some custom development to extract the metadata and automate its use.

OOTB Use Cases

6. Search for Data Assets
7. Relationship Discovery
8. Lineage and Impact Analysis
9. Change Management
10. Incident Management
11. Business Glossary Management
12. Data Profiling & Analysis
13. Security & Privacy Policy Management

Use Cases Requiring Custom Development

1. Factory Development
2. Integration Development
3. Digital Transformation Planning

OOTB Metadata Use Case	Description
Search for Data Assets	Find data assets using a Google-like semantic search to view general information about the asset such as the asset type, parent asset, resource that contains the asset, and date that the asset was last updated in the catalog. Asset types could include Data Category, Columns or Rows, CSV File or Field, Data Domain, Data Source or Connection, Database, Dimension Table or Fact Table, Field, File System, Glossary, Namespace, Policy, Project, Rule, Schema, Stored Procedure, Table, Tableau Column, Table or Workbook, Business Term, User, View, View Column, Worksheet, XML Field or File.
Relationship Discovery	The Relationships view describes the relationship between a selected asset and other assets in the catalog. The related assets that you view for a selected asset depends on the asset type. For example, if the selected asset is a column, the Relationships view shows all the data domains, similar columns, business term, and users that are related to the column. If the selected asset is a data domain, the Relationships view shows all the columns, business term, users, and data domain groups that are related to the data domain.
Lineage and Impact Analysis	Lineage and impact describe the end-to-end data flow of data for an asset. The data flow for an asset has two components, the lineage and the impact. Lineage describes the flow of data from the origins to an asset. Lineage shows you where the data for an asset comes from and which assets affect the asset that you are studying. When you view an asset in a lineage and impact diagram, the lineage includes the asset that you are viewing and all of the upstream assets in the data flow. Impact describes the flow of data from an asset to the destinations. Impact shows you where the data is used, and which assets might be affected if you change the asset that you are studying. When you view an asset in a lineage and impact diagram, the impact includes the asset that you are viewing and all of the downstream assets in the data flow.

OOTB Metadata Use Case	Description
Change Management	This is a variant of the impact analysis use case. Identify impacted systems for a given change to ensure that all appropriate systems have been validated and that stakeholders have been informed before the change is made to production systems.
Incident Management	This is a variant of data lineage and impact analysis. Use metadata lineage and impact views to assist with production outage root cause analysis, upstream and downstream impacts of an incident, and identify necessary recovery steps.
Business Glossary Management	Create, update or view business glossary assets such as business terms and data categories in the catalog. The business glossary assets in the catalog come from the business glossaries in the Analyst tool. Following are the most common types of business glossary assets: Business terms: Words or phrases that use business language to define relevant concepts for business users in an organization. When you view the details for a business term, the catalog displays the term properties such as name, description, and usage. Categories: Descriptive classifications of business terms and policies that define a structure for a business glossary. For example, an Analyst tool user might create a category called "Financial Statements" and assign the terms related to financial statements to this category. When you view the details for a category, the catalog displays the terms in the category.

OOTB Metadata Use Case	Description
Data Profiling & Analysis	View and analyze profile results for selected data assets such as tables and fields including: • The Value Distribution showing the percentage and frequency of null, distinct, and non-distinct values in the column or field. • The patterns for the column or field values along with the percentage and frequency in which the patterns appear. • The list of inferred data types for the column or field. This provides support for rapid analysis of source data during requirements specification or at design time and aids in identifying needs for data harmonization or mapping to target data model.
Security & Privacy Policy Management	Create, modify or view the business purposes, processes, or protocols that govern business practices and data security and privacy requirements that are related to business terms. For example, an Analyst tool user might create a policy called "US GAAP" to represent a framework of accounting standards and apply the policy to specific business terms. When you view the details for a policy, the catalog displays the related terms and categories.

Metadata Use Cases requiring custom development	Description
Factory Development	Provide transparency via a dashboard regarding the process of data changes, such as what new BI/Analytics reports or data ingestions are under development, what their status and release dates are, what issues are outstanding and who is working on them.

Metadata Use Cases requiring custom development	**Description**
Integration Development	Use of technical and business metadata in support of project development and ongoing maintenance activities. Extract selected source metadata or business metadata and use it as input to an automated process or code generator.
Digital Transformation Planning	Use the BOST methodology and framework to prioritize and plan transformation initiatives. Create, modify and view the key business, operational, systems and technology elements that are involved in a given initiative, and define the relationships between them. Typical elements include: • Business Strategies • Objectives & Goals • Success or Progress Metrics • Milestones • Business Capability Functions (SF's) or Operational Capabilities (OC's) in scope including a score of the function's importance based on its strategic relevant and the opportunity for improvement • Functional pain points or problems to be addressed by the program • Requirements for specific OC's • Reference Systems in scope • Data Stores and other data assets associated with the program • Actual production systems in scope • Technology dependencies for the systems

Appendix C: Glossary of DataOps Dependent Capability Functions

Implementing DataOps demands a collaboration across a range of capabilities across the enterprise. This section describes a core collection of 48 basic service functions that would be involved in a mature DataOps practice. That said, when you are in the starting stage of implementation, you may only have a handful of service interactions that are strategic and relevant. The teamwork will grow as your practice matures.

It is also important to note that these 48 functional areas don't mean there are 48 organization units or team. In reality, these functions may be implemented in a distributed manner across a global enterprise and it could be that multiple organization units have responsibilities for different aspects of the same functional area.

That said, these functional entities are described in a way that they apply to virtually any company, in any industry, and any country on the globe. The reality is that data is a critical capability for every enterprise and data touches every part of it, so to optimize it requires a broad-based collaboration involving teamwork across many functions.

C.1 Enterprise Information

Information Security Governance: Responsible for establishing and overseeing compliance with rules, guidelines and procedures for ensuring the security and privacy of all types of information over the course of its existence in accordance with information security policies. This includes all aspects of information protection including access controls, virus and intrusion protection, and information classification, usage, duplication, disclosure, disruption, modification, perusal, recording and destruction. It assigns the roles and responsibilities for information

security operations and also provides guidance on the application and usage of related software tools to the involved users.

Information Security Operations: Manages, and controls security services associated with the IT infrastructure and its supported base of application and data management systems. This includes the monitoring of security procedures in operations (including updating of virus and intrusion protection), the management of access controls to networks, computer sites, applications and data stores, as well as the analysis of cyber intelligence to guard against threats and vulnerabilities affecting service and information integrity.

Data Governance: Responsible for establishing data governance framework, methodology, and standards for Enterprise Information Management. It ensures that Information Strategies and Policies are followed. It also ensures the information usability and protection for the enterprise information stakeholders and is responsible for establishing the relationship between Information owners and custodians.

Data Integration Operations: Responsible for all data movement & exchanges between application systems and data stores. Responsibilities include effective operations of Data Delivery systems within the Systems Families Reference Architecture, identifying and coordinating cross-system impacts of data changes, and for measuring, analyzing and reporting data delivery operations, quality, security, and execution exceptions.

IT Infrastructure, Platform & Datacenter Operations: Responsible for operating the complete set of technology services that provide the development and run-time environments for application systems, their related data stores and integration systems, as well as the collaboration and communication services. This includes set-up, ongoing monitoring, performance management, repair and restoration, and usage accounting for these technology services. This function also verifies and controls access to IT Infrastructure, Platform & Datacenter.

Information Quality Management: Responsible for establishing and overseeing compliance with guidelines, standards, and best practices for the measurement, analysis and reporting of the quality of information. This includes information accuracy, validity, reliability, relevance, conformity, consistency and timeliness. It assigns the roles and responsibilities for information quality management to the associated information stewards. It is also responsible for overseeing the identification of failure points and administration of corrective actions.

Data Management Operations: Responsible for maintaining the usability, integrity and privacy of the contents of all electronic data storage

and archiving capabilities within the IT Infrastructure. It defines rules and guidelines for data operations and assigns roles and responsibilities to data operations personnel. This includes all types of electronically stored information, including transactional data, published data, reference data, and test data. It does initial loads of data sets, performs the specified data back-ups, restores corrupted or lost data, and manages transfers of data to maintain the specified service levels and performance. It is also responsible for archiving and purging electronically stored information in accordance with the established rules and guidelines for information retention and lifecycle management.

Data Architecture Management: Responsible for establishing rules and standards in support of information policies that govern how data is organized, stored, integrated, and made accessible to application systems. It is also responsible for maintaining an Enterprise Data Model which is a canonical (or common) form of data elements and entities with attributes and interrelationships. It also designs specific data models to guide the development of new data stores, new data exchanges between system components and manual data management systems such as forms and tables.

Applications Operations: Responsible for the operation of all application systems running on the Information Technology (IT) Infrastructure. This includes setting up those applications in accordance with allocated capacity, monitoring the operation and performance of those applications, reporting any bugs encountered and applying any patches and fixes, to operational systems, provided to address those bugs. This function also verifies and controls access to Applications.

Metadata Management: Responsible for establishing and overseeing compliance with rules, guidelines and procedures for creating and maintaining enterprise information that is considered meta data, as defined in the metadata architecture and in accordance with information policies. This covers all areas of metadata management including documenting data assets and related organizational responsibility and accountability, establishment of business glossaries, tracking of data lineage, guidance on data reuse, collection and usage of operational meta data, and usage of meta data for audit and governance purposes. It assigns the roles and responsibilities for creation, maintenance and stewardship of all meta data and also provides guidance on the application and usage of related software tools to the involved users.

Source Code Management: Responsible for maintaining the library of source code throughout its development cycle and for providing software configuration management of the various modules that comprise the

software products. This includes source code searching to reveal reusable lines of source code for applying common fixes or enhancements.

Software Development Project Management: Responsible for managing the production and maintenance of all software modules and their integration into software products in accordance with the provided system designs to deliver the specified software features. This includes creating, updating and tracking the status of software development project plans with required work activities, resource allocations and schedules. These plans encompass the code development activities, various levels of testing, rework and debugging, software development documentation and acceptance reviews/sign-offs, including any reported software defect resolution. This function also processes any change requests to the software feature specifications and updates the plans accordingly. It also establishes software development and testing standards and addresses opportunities to improve software production quality.

Operational Architecture Management: Supports transformation program planning and operational strategy development by providing target operational architectures and associated roadmap of new and enhanced operational capabilities to enable new business models and improve operational effectiveness. This includes creating service function and information reference models; using these reference models for baseline assessment; conducting opportunity assessments for improving operations; creating integrated target architecture models to show future business process requirements, organizational structures with accountabilities, and resource requirements; and placing these capabilities on a migration strategy to align with related business and systems programs. These target models are used by operational planning and delivery functions to structure, organize, and govern related operational transformation programs.

Business Architecture Management: Supports transformation program planning and business strategy development by providing target business architectures and associated roadmap of new and enhanced business capabilities to enable the execution of those business strategies. This includes creating various business reference models, using these reference models for baseline assessment, conducting opportunity assessments for improving business performance, and creating integrated target architecture models to show future market positioning, product offering capabilities, enterprise structures, proposed business partner relationships, resource requirements, and placing these capabilities on a migration strategy to align with desired delivery timelines. These target

models are used by business planning functions to structure, organize, and govern related business transformation programs.

Systems Architecture Management: Supports transformation program planning and systems strategy development by providing target system architectures and associated roadmap of new and enhanced system capabilities to enable new operational capabilities and simplify/improve existing architectures. This includes creating systems reference models, using these reference models for baseline assessment, conducting opportunity assessments for reducing diversification and upgrading system capabilities, creating integrated target architecture models to show future application systems, data stores, and information exchange solutions, and placing these capabilities on a migration strategy to align with related operational and technology programs. These target models are used by systems planning functions to structure, organize, and govern related systems programs.

Technology Architecture Management: Supports transformation program planning and technology strategy development by providing target technology architectures and associated roadmap of new and enhanced technology capabilities to enable new systems capabilities and simplify/improve existing architectures. This includes creating technology reference models, using these reference models for baseline assessment, conducting opportunity assessments for adopting technology standards and upgrading technology capabilities, creating integrated target technology architecture models, and placing these technology capabilities on a migration strategy to align with related systems programs. These target standards, architectures, and vendor products are used by technology planning functions to structure, organize, and govern related IT Infrastructure programs. Additionally, this function is responsible for evaluating new technology product releases for inclusion in the technology strategy.

Business Intelligence Governance: Responsible for establishing and overseeing compliance with rules, guidelines and procedures for creating and maintaining enterprise information that is considered business intelligence (BI) related to ongoing operations and analytical information for investigative purposes, as defined in the enterprise data model and in accordance with information policies. This covers all areas of enterprise operations involved in analytical activities to derive business insights, such as strategy development functions, performance assessment, market analysis, competitive assessment and opportunity assessment, and may involve cross-relationships with other analytical data. It assigns the roles and responsibilities for maintenance and stewardship of BI information

and analytical data and also provides guidance on the application and usage of related software tools to the involved analysts and planners.

Master Data Governance: Responsible for establishing and overseeing compliance with rules, guidelines and procedures for creating and maintaining enterprise information that is considered master data as defined in the enterprise data model and in accordance with information policies. This covers all areas of enterprise operations involved in data management-based services where that information relates to key elements of the business model and is therefore highly shared across multiple functions, such as products, customers, suppliers, partners, and workers, and may involve cross-relationships with other master data. It assigns the roles and responsibilities for maintenance and stewardship of master data across the involved functions. It also resolves any conflicts across these various master data domains to ensure effective utilization of key enterprise information resources.

Reference Information Management: Responsible for establishing and overseeing compliance with rules, guidelines and procedures for creating and maintaining enterprise information that is considered referential in nature as defined in the enterprise data model and in accordance with information policies. This covers both externally sourced and internally generated reference information, such as geographical areas, postal codes, identification codes, organization codes, sales territories, and product codes, many of which are organized as hierarchies and involve cross-relationships. It is also responsible for maintaining the usability, integrity and timeliness of reference data and hierarchies that are commonly shared within the enterprise and also with external organizations. It also assigns the roles and responsibilities for maintenance and stewardship of enterprise reference information.

C.2 Analysis and Assessment Functions

Large enterprises typically have a number of Analysis and Assessment capabilities, each focusing on a particular major functional area. This section identifies eight common areas under the oversight of Business Intelligence Governance.

Each of the functions described in this section are also responsible for discovery and communication of new meaningful patterns in data and for communicating the results and business insights to relevant planning functions as defined by Business Intelligence Governance. This includes responsibility for implementation and oversight of BI policy, standards

and procedures, resolving analytics and reporting issues, building ad hoc reports, oversight of sand-box environments, and initiating production BI capabilities Including watermark reports.

Business Performance Assessment: Tracks the achievement of the enterprise in meeting the established performance targets. It extracts and analyzes information from financial results, product sales, and cost analyzes to produce business results related to key performance measures. It also assesses these results against targets and accounts for variances and influences on results achievement.

Financial Analysis: Provides assessments and consulting on key financial analyses and decisions pertaining to enterprise plans and programs such as mergers and acquisitions and other business cases and cost analyses associated with enterprise transformation proposals, including buy vs. make, buy vs. lease and similar financial trade-offs.

Market Opportunity Analysis: Provides an assessment of market growth and expansion potential in current and prospective market areas for targeted market segments. These assessments support new market entry, new product introduction, brand awareness, and channel suitability.

Market Segment Preference Analysis: Provides an analysis of product and channel preferences by market segment for targeted market areas. These analyses are used in establishing product portfolios and product assortments for various channels and locations.

Market Competitive Analysis: Provides an analysis of competitors and how they are performing in selected market areas and segments. It produces penetration analyses to track performance against competition.

Customer Analysis & Insights: Responsible for identifying actionable customer insights that can become leads and opportunities. It aggregates and enriches customer transaction data, customer network data, customer software lifecycle data, customer profile data, partner profile data, and product profile data to deliver the insights and analytics. It defines, applies and governs rules for insight identification and analytics and their corresponding quality and effectiveness. It manages access and subscription to the actionable insights and delivers them to authorized subscribers.

Workforce Analysis & Planning: Creates and updates the workforce plan to meet the projected needs for human capital taking into account planned business growth, current workforce, projected retirements, attrition, acquisitions and any downsizing and in accordance with the human capital strategies. Based on this analysis, it also determines the

need for hiring new personnel by category of employee or contingent worker and by work locations. This is used to drive recruitment and supplier sourcing.

Financial Operational Analysis: Responsible for recording non-accounting financial measures, such as pricing and sales order bookings. It is also responsible for analyzing accounting and non-accounting financial measures, including pricing, sales order bookings, revenue and cost information to provide insights into leading and lagging measures of financial performance. It provides analysis of financial forecast deviations and uncovers opportunities for improved financial performance.

C.3 Master Data Management Functions

Large enterprises typically have a number of Master Data Management capabilities, each focusing on a particular major functional area or data domain. This section identifies eight common areas under the oversight of Master Data Governance.

Each of the functions described in this section are also responsible for identification and resolution of commonality between master data entities as defined by Master Data Governance. This includes using matching, merging, hierarchy, and relationship rules and following the policies for ongoing reconciliation.

Customer Information Management: Responsible for capturing and maintaining information related to the registry of parties with whom the enterprise has selling relationships, including organizations, locations, contacts, and the network of hierarchical relationships between them. It maintains customer accounts and associated addresses, as defined by the Enterprise, the Customer and/or the Customer's legal structure, for specific contexts including billing, shipping, servicing, entitling and other purposes. It includes Customers supported directly by the Enterprise as well as by Channel Partners. The relationships of customer parties to each other include affiliate-of, subsidiary-of, employee-of or contact-of. The function also maintains statuses, classifications and preferences of the Customer.

Contract Information Management: Responsible for consolidating and maintaining information pertaining to individual contracts with customers for the provision of financial and insurance services over the life cycle of those contracts. This includes managing contract information from the original set up activities, any changes to those contracts,

generating renewal notifications, updating premiums, registering claims and payments, and maintaining contract status such as active, cancelled or expired.

Workforce Information Management: Responsible for creating and maintaining information about the enterprise's employees, contingent workers, and past workers. This responsibility begins at the time of making employment offers to recruits or the onboarding of contingent workers and continues throughout the period of their engagement with the enterprise and extends to postemployment periods for tracking any ongoing relationships. This function gathers information about all transactions that will affect the worker profile and makes that profile information available to the workforce, the organizations within the enterprise, and specific other service functions that require profile information. It also ensures the security and privacy of this information and validates entitlements for access.

Partner Information Management: Responsible for capturing and maintaining information relating to specific partners of the enterprise over the course of the business relationships with that partner and extending into any periods of ongoing obligations (for service, warranties, settlement etc.). This information includes basic profile data, contact information, sales history, and certifications achieved. It also tracks current partner entitlements based on partner agreements and contracts.

Vendor Information Management: Responsible for capturing and maintaining information related to the registry of vendors with whom the enterprise has procurement relationships, including organizations, locations, and contacts.

Supplier Information Management: Responsible for capturing and maintaining information for both direct and indirect suppliers over the course of the business relationship with that supplier. This information includes supplier profile, payment terms, pricing and offering information. It also includes business license, tax information, banking information, contractual information, non-disclosure agreements, background investigation results, code of ethics, environmental health and safety certifications.

Risk Information Management: Responsible for consolidating different risk data and exposure information and providing the tracking and management reporting capabilities to enable the user to monitor and control the overall risk exposure for better decision support. It supports expert advice and cost-effective data management solution around key processes like Risk identification and assessment, Risk Control and Risk Financing.

Reference Information Management: Responsible for establishing and overseeing compliance with rules, guidelines and procedures for creating and maintaining enterprise information that is considered referential in nature as defined in the enterprise data model and in accordance with information policies. This covers both externally sourced and internally generated reference information, such as geographical areas, postal codes, identification codes, organization codes, sales territories, and product codes, many of which are organized as hierarchies and involve cross-relationships. It is also responsible for maintaining the usability, integrity and timeliness of reference data and hierarchies that are commonly shared within the enterprise and also with external organizations. It also assigns the roles and responsibilities for maintenance and stewardship of enterprise reference information.

C.4 Biz Ops Planning

Operational Strategy & Policy Planning: Responsible for creating the overall strategy for enterprise operations to achieve business goals and objectives through increasing operational effectiveness and efficiencies. It requests operational architectural assessment of improvement opportunities. It also creates and maintains policies related to business operations in accordance with current and planned operating requirements.

Systems Strategy & Policy Planning: Responsible for creating the overall strategy for systems applications with a view to contributing to the achievement of business goals and objectives through enabling operational efficiencies with effective system solutions. It requests systems architectural assessment of change opportunities. It also creates and maintains policies related to systems applications in accordance with current and planned system requirements.

Technology Strategy & Policy Planning: Responsible for creating the overall strategy for deploying information technologies across the enterprise to enable information management and systems applications through effective use of computing, storage, networking, and end user technologies. It requests systems architectural assessment of change opportunities. It also creates and maintains policies related to technology utilization in accordance with current and planned requirements.

Information Strategy & Policy Planning: Responsible for creating the overall strategy for managing information with a view to contributing to the achievement of business goals and objectives through enabling

operational efficiencies with effective data management solutions. It provides strategic guidance to operational, systems and technology planning to address data management opportunities. It also creates and maintains policies related to data management, security and privacy in accordance with current and planned operational, system and technology requirements.

Biz Ops Program Planning and Governance: Responsible for planning and governing programs to deliver new or enhanced capabilities related to business operations, including processes, organizations, application and information systems, and technology infrastructure services. It defines projects to deliver a program's scope based on near-term architectures from business transformation, renovation, improvement and sustaining planning engagements. It is responsible for allocating funding for approved projects, maintaining the portfolio of projects in support of programs, and tracking project delivery milestones in compliance with the program scope, budget, and timeline.

Biz Ops Portfolio Management: Responsible for creating and governing initiatives to facilitate business operations investment prioritization and decision-making. These investment decisions relate to processes, organizations, applications and information systems, and technology infrastructure services. It constructs these initiatives based on business strategies, operational opportunities, target architectures, and scope units from planning engagements. It prioritizes and makes funding decisions for initiatives based on their business value, scope units, and interdependencies within the constraints of financial investment guidance.

C.5 Biz Ops Program Delivery

Biz Ops Program Management: Oversees the delivery and implementation of all operational, systems and technology programs, developing detailed workplans, resource plans and milestone deliverables that address the many interdependencies and change management requirements across these programs. It is responsible for reviewing and approving all milestone deliverables and receives support from the Enterprise Architecture Management functions in conducting compliance reviews to ensure that program architectures are compatible with enterprise target architectures for the operational, systems and technology views. It also assures that the impacted operational functions are ready to support the new process, systems, and technology capabilities as part of acceptance testing and turnover.

Business Process Solutions Delivery: Establishes and manages process reengineering activities to design and develop new or improved business processes in accordance with operational development programs and related operational and systems architectures. These new process designs build on planned organizational designs and new systems capabilities to ensure synchronization. This function also develops process change implementation plans and supporting documentation for new or improved operational procedures.

Organizational Design Solutions Delivery: Creates and maintains the detailed organizational design for the enterprise based on the target organizational model and related RASCI accountabilities to BCFs. It provides the complete accountability structure of organizational units, their reporting structures, their assigned functional responsibilities and any geographic, product line, market segment or other delineations of roles and responsibilities. This also includes developing and updating skill and role descriptions for the various positions within the organization design. A key role of this BCF is to develop implementation plans to address the change management requirements associated with the organizational change.

Systems Solutions Delivery: Designs develops, acquires, tests and implements new and improved information systems for the enterprise in accordance with systems development programs, related architectures, approved system vendors and compatibility with technology standards. It also responds to performance issues or IS incidents that are related to system design or implementation problems; plans review and implements new software releases and supplies system fixes or patches as required. It is also responsible for maintaining an accurate baseline inventory of installed application systems, data stores and integration systems.

Technology Solutions Delivery: Designs, develops, acquires, tests and implements new and improved information technology and infrastructure services for the enterprise in accordance with technology development programs, related architectures, established technology vendors and compatibility with technology standards. It also responds to performance issues or IT incidents that are related to technology design or implementation problems and supplies technology fixes or patches as required. It is also responsible for maintaining an accurate baseline inventory of installed information technologies, platforms and networks.

Operational Control & Audit: Is responsible for conducting regular and ad hoc audits of operational processes to ensure compliance with process

performance standards. This includes publishing and submitting operational audit reports that may be required by regulatory or certification bodies. It also includes tracking the status of any remedial actions that are required as a result of the audit.

Data Architecture Management: Is responsible for establishing rules and standards in support of information policies that govern how data is organized, stored, integrated, and made accessible to application systems. It is also responsible for maintaining an Enterprise Data Model which is a common form of data elements and entities with attributes and interrelationships. It also designs specific data models to guide the development of new data stores, new data exchanges between system components and manual data management systems such as forms and tables.

About the Authors

John Schmidt

John Schmidt's data management career began over 40 years ago at Digital Equipment Corporation followed by American Management Systems, Best Buy, Bank of America, Wells Fargo Bank and Informatica.

Over the years he worked as a technician, software engineer, project manager, professional services manager, program manager, enterprise architect and management consultant. He has practiced and honed his expertise in half a dozen industries (banking, retail, telecommunications, education, government, and utilities) in 20 countries. John holds a master's degree in Business Administration from the Carlson School of Management at the University of Minnesota.

John's current role is Business Architect at Proact Digital Transformation. He helps organizations accelerate their speed and modernization of data by adopting a deep understanding of technology, automation and business architecture to innovate. He accelerates strategies and business objectives by applying repeatable structures, principles and practices to guide organizations through the transformation journey.

He has published hundreds of articles on Systems Integration, Enterprise Architecture, and Program Management and is a frequent speaker at industry conferences. John wrote the first book about ICCs in 2005, *Integration Competency Center: An Implementation Methodology*, followed it up in 2010 with *Lean Integration: An Integration Factory Approach to Business Agility* and continued in 2019 with *DataOps: The Authoritative Edition.*

Kirit Basu

Kirit Basu has over 20 years of experience as a technologist at startups performing disruptive innovation in industries such as healthcare, logistics, edtech and big data.

He has worked in engineering, product management and entrepreneurial roles across all aspects of the technology stack, from hardware and embedded systems to enterprise applications and analytics systems. Most recently, Kirit has focused on building products that help companies solve for data integration in the world of constantly changing data, business requirements, and technological advancements.

Kirit's current role is Vice President of Product at StreamSets, Inc. At StreamSets, he helps define and drive the vision for the industry's first DataOps platform.

About StreamSets

StreamSets built the industry's first multi-cloud DataOps platform for modern data processing and integration, helping enterprises to continuously flow big, streaming and traditional data to their data science and data analytics applications. The platform uniquely handles data drift, those frequent and unexpected changes to upstream data that break pipelines and damage data integrity. StreamSets allows for execution of any-to-any pipelines, ETL processing and machine learning with a cloud-native operations portal for the continuous automation and monitoring of complex multi-pipeline topologies.

Founded in 2014, StreamSets is backed by top-tier Silicon Valley venture capital firms, including Battery Ventures, New Enterprise Associates (NEA), and Accel Partners. For more information, visit www.streamsets.com.

"This book delivers a complete perspective on DataOps from data cataloging to pipeline management, bringing sanity to out of control data proliferation."

Alex Gorelik, Founder and CTO at Waterline Data

"DataOps is transforming the delivery of data for analytics by using a continuous improvement approach to acquire, aligning, rationalizing and evolving the data to meet the needs of the consumer. For anyone focused on harnessing data as a strategic asset, DataOps is a must, and this is the definitive guidebook to building that practice."

Mark Ramsey PhD., Chief Data & Analytics Officer

"Modern data infrastructures are materially different than what they were 10 years ago. The level of chaos and complexity inherent in it is overwhelming, posing new challenges that were never felt before. The DataOps Authoritative Edition presents the methodology to solve such challenges in a sustainable and cost-effective manner for the long term."

Arvind Prabhakar, CTO at StreamSets

"DataOps builds on the infonomics concept of managing information as an actual asset. The differences between typical 'hairball' data architectures and the approach Schmidt and Basu lay out are stark yet achievable—and imperative for any organization that wants to thrive in the data economy."

Doug Laney, Data Strategy Practice Lead at Caserta
Best-selling Author of "Infonomics"

"DataOps bridges the gap between how users think of data (in business terms) and how data is documented (cryptic, missing or misleading technical names assigned to data by developers). Once data is understood, data can be used more strategically, and this book by Schmidt and Basu is a great resource to make it happen."

Joe Hellerstein, Trifacta's Chief Strategy Officer and Co-founder

---

Visit go.streamsets.com/dataops-assessment.html for additional resources, more case studies, best practices, templates, software demos, and the DataOps Maturity self-assessment tool.