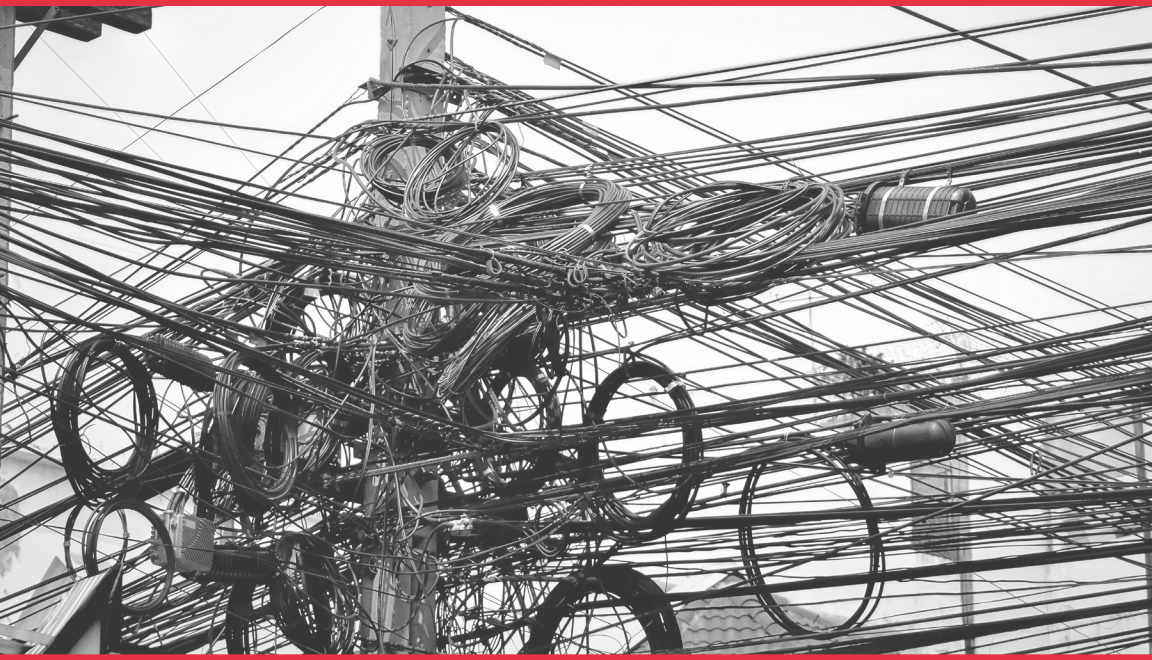
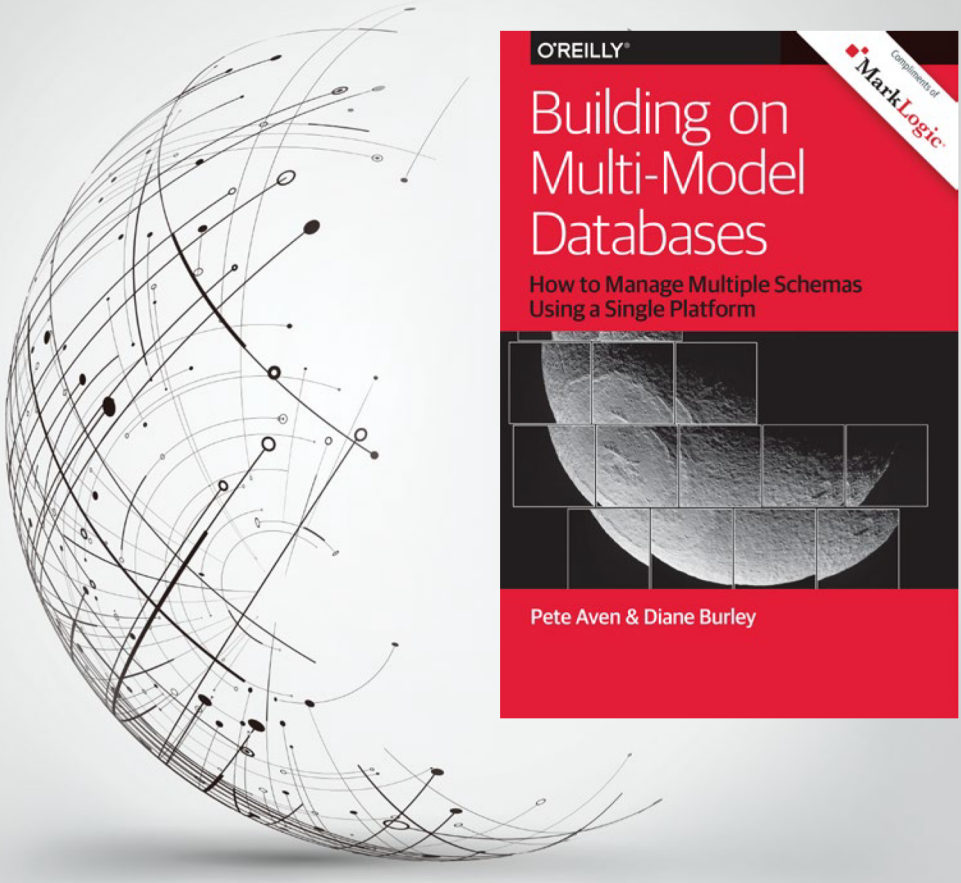
O'REILLY®

Compliments of MarkLogic®

Cleaning Up the Data Lake with an Operational Data Hub



Gerhard Ungerer



YOUR DATA DESERVES BETTER

The Evolution of Data Integration

Each year billions of dollars and countless hours are spent integrating data from silos across organizations. Legacy tools aren't agile enough to handle today's heterogeneous data. Find out how you can use a multi-model database to reduce complexity and risk, save money, and shorten time to value.

Download your free eBook, compliments of MarkLogic.

MARKLOGIC.COM/MULTIMODEL

MarkLogic

Cleaning Up the Data Lake with an Operational Data Hub

Gerhard Ungerer

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

Cleaning Up the Data Lake with an Operational Data Hub
by Gerhard Ungerer

Copyright © 2018 O'Reilly Media, Inc. All rights reserved.

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com/safari*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Editor: Jeff Bleiel
Production Editor: Nicholas Adams
Copyeditor: Octal Publishing, Inc.
Interior Designer: David Futato
Cover Designer: Randy Comer
Illustrator: Rebecca Demarest

January 2018: First Edition

Revision History for the First Edition
2017-01-10: First Release

The O'Reilly logo is a registered trademark of O'Reilly Media, Inc. *Cleaning Up the Data Lake with an Operational Data Hub*, the cover image, and related trade dress are trademarks of O'Reilly Media, Inc.

While the publisher and the author have used good faith efforts to ensure that the information and instructions contained in this work are accurate, the publisher and the author disclaim all responsibility for errors or omissions, including without limitation responsibility for damages resulting from the use of or reliance on this work. Use of the information and instructions contained in this work is at your own risk. If any code samples or other technology this work contains or describes is subject to open source licenses or the intellectual property rights of others, it is your responsibility to ensure that your use thereof complies with such licenses and/or rights.

This work is part of a collaboration between O'Reilly and MarkLogic. See our *statement of editorial independence*.

978-1-492-02735-5

[LSI]

Table of Contents

Cleaning Up the Data Lake with an Operational Data Hub

Introduction

Organizations of all sizes are attempting to gain control over the data that resides in multiple formats and locations across the enterprise. Each business unit maintains data in text files, spreadsheets, databases (both custom built and commercially available), applications (both modern and legacy), and other media. The ability to consolidate all of this data throughout the enterprise will provide insight into business outcomes (including trends in customer, staff, service, product, and revenue data) as business processes or annual budgets are adjusted.

Traditionally, collection and integration of data was performed on relational database technologies to form data warehouses. Extract-Transform-Load (ETL) processes implemented the transformations between each source system and the data warehouse. This approach, however, required a tremendous amount of data modeling of the data warehouse and mappings from each source system before processing can begin. Any change in format at the source system or data warehouse required impact assessment, data modeling, and updates to code. The result was a fragile processing environment with huge project cost overruns and failure rates of 70 to 80 percent.

It was not until the "big data" era that technologies emerged that can process data without concern to data structure and protracted data modeling. Search engines and social media sites need the ability to quickly ingest tremendous amounts of data in whichever format it is found and present it to consumers. Organizations took note of these

capabilities and used big data technologies such as NoSQL and Hadoop to collate data across the enterprise into a single location; referred to as the *data lake*. The initial results of the data lake were promising, but the ease of integration and the lack of governance soon led to a systematic loss of quality. Data lakes that reach this condition are called a *swamp* (see the section "What Is a Data Swamp?" on page 12). The result is a failure rate of up to 88 percent of big data analytics projects[1]—despite the investment of millions of dollars and months (if not years) of development time.

As NoSQL technologies matured and enterprise capabilities were added to avoid the data swamp condition, a new pattern emerged. This pattern is called an *Operational Data Hub* (ODH), as illustrated in Figure 1-1. The ODH allows organizations to collect, store, index, cleanse, harmonize, and master data of all shapes and formats. The ODH also supports transactional integrity so that the hub can serve as integration point for enterprise applications.

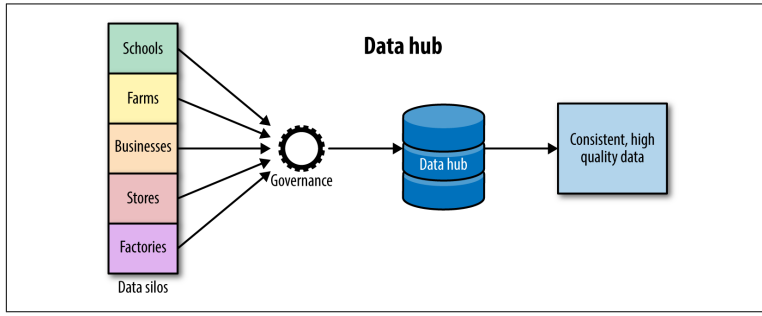


Figure 1-1. The Operational Data Hub pattern

The benefit of these new capabilities, and the ODH pattern, is that organizations can take advantage of the investment already made in its enterprise integration efforts to remedy the data swamp. But for us to avoid swapping a data swamp implemented in one technology for a data swamp implemented in another, some initial strategic planning (including technology selection) is in order. After this initial planning is completed, and the objectives of the data hub are defined, the data in the swamp can be ingested, processed, and provisioned according to its purpose.

1 "Inflexible Data, Analytics Fueling Failures, Survey Finds - Datanami"

The End Goal: Enterprise Data Integration

Let's take a step back.

Most large organizations have several departments and business units, each with defined objectives and responsibilities. Because IT systems are developed to streamline and automate business processes, it stands to reason that a typical large organization has hundreds, if not thousands, of IT systems implemented in a gamut of technologies over the past few decades. Organizational, political, and technology barriers often impede collaboration and access to data that might be useful across the enterprise. These islands of operation are called *data silos*. Data silos obviously negatively affect organizational efficiency and data quality.

Consider the scenario in Figure 1-2. A typical health insurance organization has several business units, focused on client referral, enrollment, claims processing, and payment processing. Client data is entered into the referral system and then sent to the enrollment system. At this point, client information can be updated before making it available to provider systems and the claims processing system. Client data across these systems quickly lose synchronization and point-to-point integrations (data exchanges directly between systems) need to be updated when client data changes.

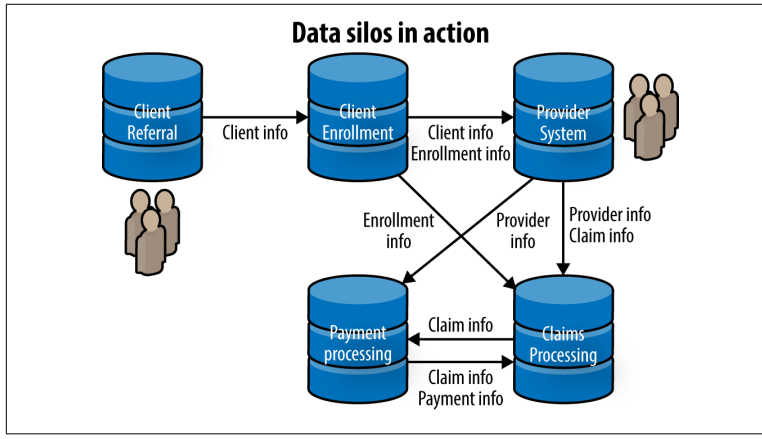


Figure 1-2. Data silos in action with point-to-point integrations

To alleviate some of the issues associated with point-to-point integration represented in Figure 1-2, organizations turned to enterprise data integration efforts to collect data across data silos into a single

location (let's call it the enterprise data store for now) where it could be further processed for a single view across the enterprise. Figure 1-3 shows the updated architecture. Client data is created by the referral system and submitted to the enterprise data store. The enterprise data store sends new client information to the enrollment system where business unit workers can complete enrollment workflow. During this process, additional client information is collected and submitted to the enterprise data store. The enterprise data store adds the additional client data and makes the information available to the client referral system, provider system, and claims processing system.

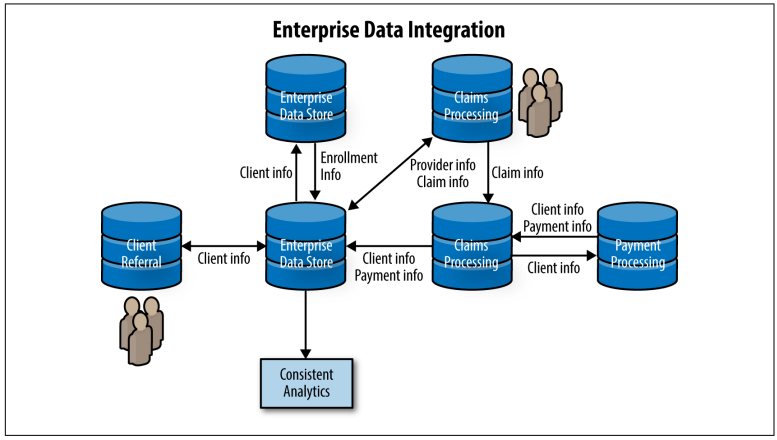


Figure 1-3. Enterprise data integration with an enterprise data store

We expand on enterprise data integration patterns and challenges that implement the enterprise data store later in this report.

Marketing literature typically predicts increased profit, streamlined service delivery, and regulatory compliance as natural outcomes of enterprise data integration efforts. But does that mean that enterprise data integration is useful only for profit-driven organizations? Of course not. Enterprise data integration is equally applicable to organizations of all sizes in both the public sector and the private sector. The following list includes, but is not limited to, enterprise data integration objectives that exists in both the public and private sector:

360° view of a person
: It is clear that a comprehensive view of a customer across lines of business will offer the opportunity to sell additional products

or services. However, in social services, it will also enable the caseworker to refer clients to programs for which they might be eligible. Also, in law enforcement, an officer or caseworker might be able to reference all known information about a citizen, including license status, outstanding warrants, etc.

Enterprise view of business processes
: In the previous example, the insurance company would be able to analyze the process from referral to completion of enrollment and gain efficiencies that would reduce client abandonment rates. The result is higher revenue and lower cost; increased profit. In social services, sharing information between Child Support, Medicaid, and the various Public Assistance systems would enable integrated eligibility processing for persons needing temporary assistance with less delay and fewer mistakes due to incomplete data.

Regulatory compliance
: Much effort is expended in regulatory reporting (e.g., Sarbanes-Oxley and Dodd-Frank) for financial institutions. Reducing this effort to an automated report from consistent, integrated data will reduce cost and apply resources to revenue-generating purposes. In the public sector, where policy compliance drives federal funding in areas such as Child Support (OCSE-157, OCSE-34a) and Medicaid reimbursement, a single source of truth is paramount.

What Are the Challenges Faced by Enterprise Data Integration Efforts?

Enterprise data integration initiatives have encountered the following business and technology challenges when interrogating data across business and information silos:

Large amounts of data
: Depending on the data strategy of the organization, and the size of the organization, an integration initiative must contend with a tremendous amount of data during the initial data load and possibly also during incremental updates.

Multiple technologies
: Although some organizations are fortunate enough to have a leading-edge technology platform (often of a single technology

stack) that offer built-in capability for integration, most large enterprises have a vastly diverse technology platform that spans several decades of technology and software engineering maturity.

Multiple data formats
: Integration efforts in a highly diverse environment offer the challenge of contending with multiple data formats (database tables, delimited files, structured files, EBCDIC, ASCII, XML, JSON, etc.). When integrating with content management systems, we might need to consider additional formats such as PDF, PNG, JPG, and sound files.

Multiple data schemas
: Beyond the data formats, data might be presented in multiple schemas with the same element represented by a different name across systems. For instance, a client's first name could be represented by CLNT_FRST_NAME in the referral system and CLIENT_FNAME in the enrollment system. The ability to progressively translate elements ensures that a common view takes shape over time. We expand on this topic in the section "What Is an ODH?" on page 13.

Rapid changes in data schemas
: Business objectives and technology solutions will continue to evolve across the enterprise while the enterprise data integration effort progressively takes shape. The enterprise data integration process will need to easily adapt to changes in data feed schema.

Complexity of legacy data
: Legacy data is notorious for unexpected use of fields over time. Often, the same field in the database, for instance the client data element CLNT_SUFFIX (meant to capture the suffix for the name) might have been used to contain gender code, population group, age group, or veteran status over time. This is a commonly encountered, but very tough to contend with, condition.

Data quality challenges
: Data validation routines have evolved over time as technology capabilities matured. Dates that are stored in text fields are notorious for containing just about any text, if data validation in not enforced. Date formats are also often inconsistent across systems. Address elements are notorious for containing invalid

data. Routines can be implemented to validate address information and indicate where invalid entries are encountered.

Organizations of all sizes have used traditional (relational) database technologies for decades to build operational data stores,[2] data warehouses,[3] and data marts in an attempt at enterprise data integration. Data warehousing technologies have been around in various formats since the mid-1970s, gaining maturity in the 1980s and reaching a tremendous amount of popularity in the 1990s.

Enterprise data integration efforts (see Figure 1-4) built on traditional data warehouses have been notorious for huge cost overruns and failing to meet its intended operational objectives. Industry surveys consistently indicate that between 70 and 80 percent of data warehousing projects fail.

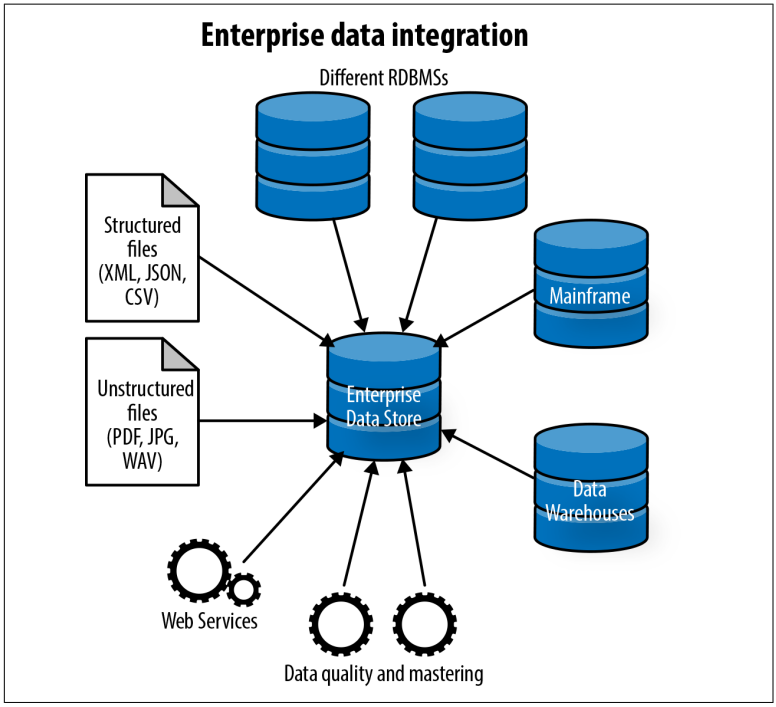


Figure 1-4. Enterprise data integration patterns

2 "Operational Data Store - ODS - Gartner Tech Definitions"

3 "Data Warehouse - Gartner"

These solutions are typically purpose-built for the desired outcomes; implying that business and technical resources need a comprehensive understanding of the solution and the data schemas across the enterprise. Data integrations are performed according to a predetermined schema at the source and destination. When business or regulatory requirements change, typically a tremendous amount of time and effort is required.

The restrictive nature of traditional technologies renders it ineffective as an enterprise integration pattern in the modern era.

Big Data for Enterprise Data Integration

Gartner defines *big data* as a "high-volume, high-velocity, and/or high-variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation."[4]

Informatica offers a definition that is most fitting in the business context as "the 21st-century phenomenon of exponential growth of business data, and the challenges that come with it, including holistic collection, storage, management, and analysis of all the data that a business owns or uses."[5]

However you term it, big data technologies were developed to ingest and process large amounts of data with different data structures and formats. Data from relational databases (rows and columns), semi-structured data (CSV, XML, JSON, etc.), unstructured data (emails, documents, PDFs), and even binary data (images, audio, video) can be ingested into a centralized data store, as demonstrated in Figure 1-5.

4 "What Is Big Data? - Gartner IT Glossary - Big Data"

5 "What is Big Data: Definition - Informatica US"

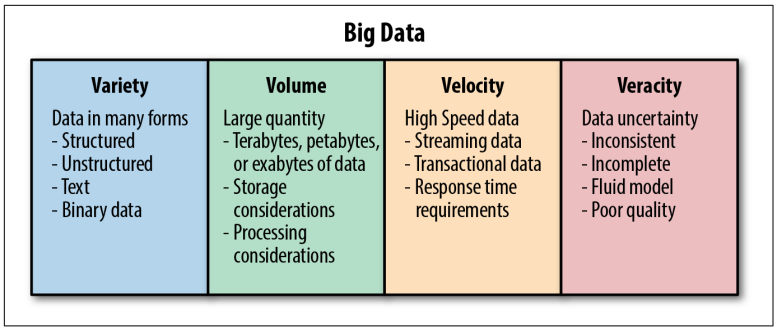


Figure 1-5. The four Vs of big data

Big data solutions are facilitated by NoSQL database technologies. NoSQL databases offer the flexibility to manage both structured and unstructured data on a super large scale; providing the ability to process data in a distributed fashion that relational databases cannot handle. Sites such as Facebook, Amazon, Twitter, and Google use NoSQL technologies to store, index, process, and retrieve the vast amounts of data that is central to their respective business models.

NoSQL technologies provide the ability to integrate, store, and process masses of data in multiple formats (colocated in a single technology platform) without the need for a specified structure. For this reason, organizations are increasingly looking at NoSQL technologies to replace traditional data warehouses[6] to implement enterprise data integration solutions; running software on a multitude of parallel servers.

There are four main types of NoSQL databases, each with a proposed use case:[7]

Key–value
: Use this type to store and lookup data associated with a key. This is usually used in transient data applications such as caching.

6 "It's the End of the Data Warehouse as We Know It - TDWI"

7 "The NoSQL Generation: Embracing the Document Model - MarkLogic"

Column–family
: This is an extension of the key–value database. A row key associates a column–family, which is a number of column–key and column–value entries.

Document
: Use this to store and lookup data in a structured format such as XML or JSON. Usually used in areas where data is denormalized and associated elements stored together. Records (called documents) do not all have to comply to the same structure in a single collection. This is the type of database used for enterprise data hubs.[8] The ODH pattern is discussed later in this article.

Graph
: Use this type when you want to focus on the relationships between records. This is often used in environments that measure associations between people, such as LinkedIn, Facebook, and Twitter. A triple-store is an implementation of a graph database

Hadoop

For many organizations, big data and the Hadoop technology are synonymous. The Hadoop Framework, or ecosystem, is *not* a database but a means of massively parallel computing that features MapReduce, which spreads computations across commodity computers. Distributing processes across the Hadoop framework significantly reduces computing time.

Although the Hadoop ecosystem does offer corresponding databases (HBase, etc.), the challenge to date has been Hadoop's lack of operational, transactional (ACID), and enterprise robustness, and therefore it is not fit for all big data solutions. It is important to note though, that there are *many* NoSQL database technologies that implement big data solutions.[9] Cassandra, CouchBase, MongoDB, MarkLogic, and RavenDB are just a few examples of technologies that are often used.

8 "Enterprises hedge their bets with NoSQL databases - PwC blogs"

9 "NOSQL Databases"

What Is a Data Lake?

Gartner defines a data lake as follows:

> [a] collection of storage instances of various data assets additional to the originating data sources. These assets are stored in a near-exact, or even exact, copy of the source format. The purpose of a data lake is to present an unrefined view of data to only the most highly skilled analysts, to help them explore their data refinement and analysis techniques independent of any of the system-of-record compromises that may exist in a traditional analytic data store (such as a data mart or data warehouse).[10]

That is quite a mouthful. Figure 1-6 illustrates the concept. Data lakes are formed when disparate and seemingly unrelated raw data is ingested in its native format typically into a NoSQL database or Hadoop[11] and stored for undetermined later use.

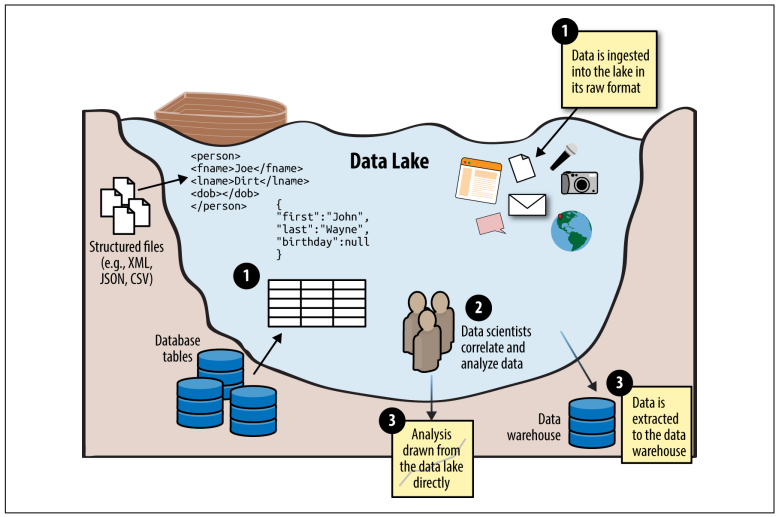


Figure 1-6. The data lake pattern

Because data is available but the intended use is unknown, organizations begin ingesting any data that it can find just in case it can be used later. The fact that no data modeling is required is a further benefit for quickly gathering data.

10 "Data Lake - Gartner IT Glossary - Gartner Inc"

11 "The Hadoop Ecosystem: HDFS, Yarn, Hive, Pig, HBase and growing"

Data scientists and data analysts can query data across data structures and apply mappings to form potential correlations that drive predictive (and historical) analysis. Analysis is typically run via MapReduce batch jobs that can also be offloaded to data warehouses and data marts.

The hype surrounding big data indicates that the data lake is available to anyone across the enterprise for query and analysis. However, it requires a specialized skillset to query and process data from the lake. We can overcome this by feeding data from the lake to a data warehouse or data mart where the typical user can perform queries, but not without first performing a great deal of transformation to get the data into the data warehouse or mart. Furthermore, this means that organizations are highly dependent upon a handful of resources to process data that will drive business decisions and operational insight.

What Is a Data Swamp?

The ease by which we can add data to the data lake is noteworthy. This is one of the primary benefits of NoSQL data lakes. However, this also means that contributors can add tremendous amounts of data at such a rate that organizations can easily lose control of their data lake. A data lake that has deteriorated into this situation is called a data swamp, as illustrated in Figure 1-7.

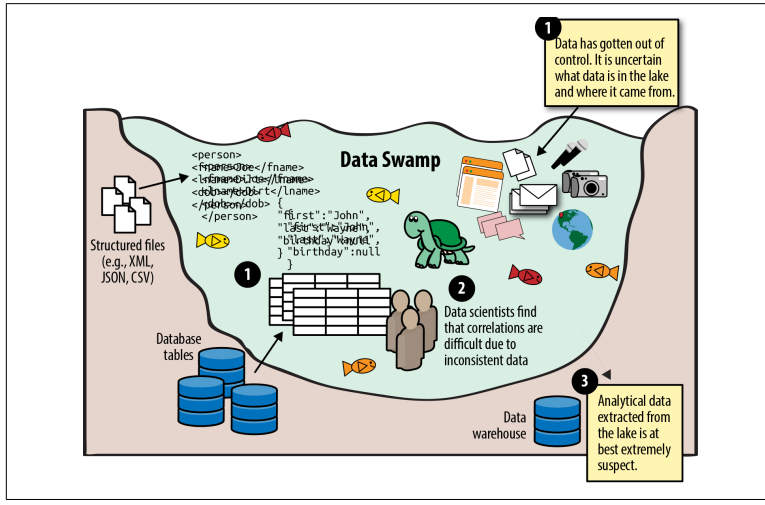


Figure 1-7. The data lake turned into a data swamp

A data swamp has so much data from so many different data sources, that it is uncertain what data is available and what purpose it serves. The ways in which data is related to one another is also unknown, making it increasingly difficult to process and to analyze results. Data scientists spend increasing effort to assess, correlate, and validate data.

Collective data quality reduces (or become more suspect) as more data sources are added. This is acceptable to use for trends and postulations that will be validated independently, but less unacceptable for master data. Also, as collective data quality decreases, so typically does the confidence and usefulness that business operations place on the data lake. This is a large contributing factor why big data analytics has not realized the promised potential[12] and up to 88 percent of big data analytics projects fail to go beyond pilot implementation.[13]

What Is an ODH?

The ODH pattern is an extension of the data lake. The typical data lake pattern focuses on moving data in its native form to a central data store (typically a Hadoop system or a NoSQL database). If you define a "data hub" as a central location for data that could be used for many purposes, several technologies could suffice. Some hubs, for example, are built on top of Hadoop, which provides the benefit of cheap storage and can accept any form of data. However, to realize the full benefit of a data hub within an enterprise environment, you need a solid database with capabilities to support:

- *Governance* to structure data processing and maintain data provenance
- *Security* to provide access control for data assets
- *Indexing* to make the data queryable in an efficient way
- *Transactional integrity* to update data in an operational environment that ensures data doesn't become corrupted or lost

12 "The age of analytics: Competing in a data-driven world - McKinsey"

13 "Inflexible Data, Analytics Fueling Failures, Survey Finds - Datanami"

The ODH pattern goes beyond moving data to a central location; it also focuses on categorizing the data and making it available for easy and fast retrieval. Adapters allow for ingest of data into the hub, where it is harmonized, cleansed and mastered, and made available across the enterprise, including source systems.

The ODH supports a rigorous governance model (see Figure 1-8) with data processing capabilities that can be used to ingest and process data according to preconfigured patterns. The Key Process Areas (KPAs) of an ODH governance model include the following:

Access
: Define the approach to access data from structured files, unstructured files, database tables, web service payloads, etc. These predefined patterns and mechanisms to determine incremental changes will form the catalyst for accessing data from new sources.

Ingest
: Bring data in different formats and structures into the ODH. Depending on the type of NoSQL database, ingesting also converts the data to a structured format (XML or JSON for document databases) and associates records with one another (via graph data). The ODH indexes the data upon ingestion for efficient retrieval and processing.

Harmonize
: As data scientists and data owners learn more about the data in the lake and how it contributes to business outcomes, they can lightly process raw data by using the envelope pattern to a harmonized structure. This makes it possible for data to be easily queried across data sources and data collections simply by using a common structure that is progressively built over time.

Materialize
: Mastered records and analytical data can be created according to a common framework and set of technologies. Mature NoSQL technologies offer scripting language support for data query and processing.

Provision
: Cleansed, harmonized, and mastered, this can be made available across the enterprise via web services, and file extracts. Provision data can be sent to source systems from a central location

so that point-to-point exchanges directly between systems can be retired.

Consume
: Web services, batch processes, analytics platforms, web applications, and other applications can consume data that is consistent across the enterprise.

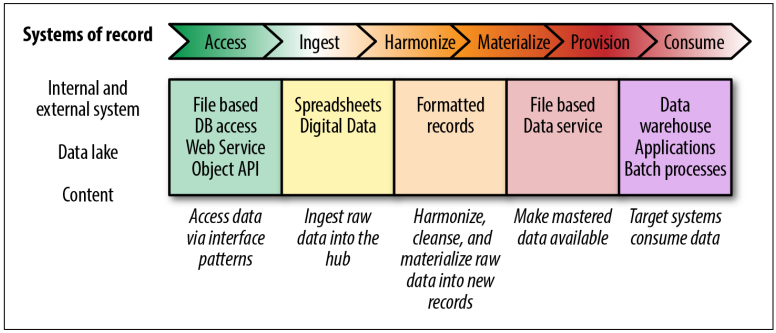


Figure 1-8. Data governance in the ODH

Progressive Transformation

After data has been ingested as raw data into the ODH, it needs to be made useful, and the ideal way to do that is to begin organizing it into business entities and retrieving these entities in flexible ways. To do so, indexing is key. You should map the most important data fields to canonical forms early, leaving the bulk of the data in raw form. Over time, more and more data is transformed and indexed, in a progressive way. In addition, data must be transformed on read into a variety of formats and schemas. Therefore, both a flexible indexing capability and a powerful transform engine is needed to provide data in an agile way.

Now let's reconsider the previous healthcare insurance example in the context of the ODH and the associated governance model (see Figure 1-9). Client data is accessed across the enterprise and ingested into the ODH in its raw format. From here, client data, enrollment data, provider data, and claim data are processed into business entities that can be used across the enterprise. After it has been cleansed and mastered, the referral data can be shared with the enrollment system to complete health plan enrollment. After it has been updated, ingested, harmonized, and materialized, the enroll-

ment data can be shared with back-office workers and/or providers for prior authorization of services or adjudication of claims. All constituents across the enterprise have the latest data available to drive effective business processing and information sharing.

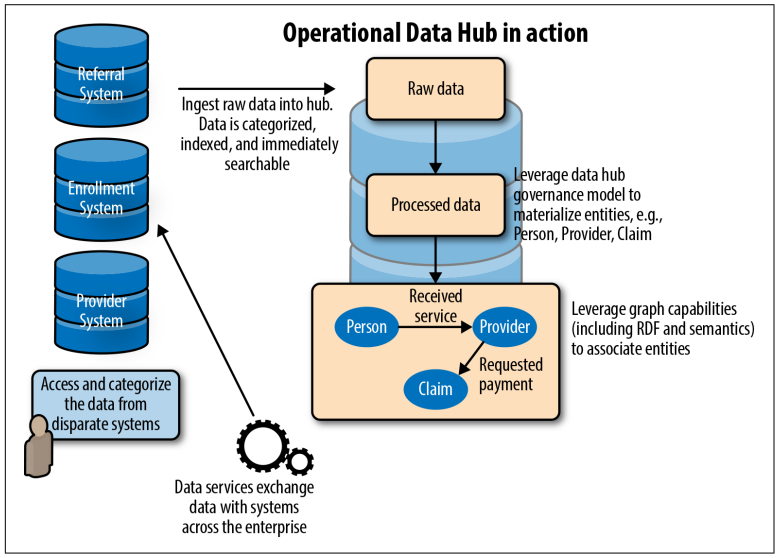


Figure 1-9. ODH in action

The Benefits of an ODH

Most organizations begin experimenting with open source NoSQL technologies such as Hadoop and MongoDB and see tremendous potential during pilot and experimentation as the data lake takes shape. However, when organizations attempt to roll out the data lake to the enterprise, things take a turn for the worse. A recent Capgemini report indicated that more than 70 percent of big data projects eventually fail due to lack of enterprise capabilities and support.

The ODH pattern overcomes the following shortcomings of data lake technologies:

Flexible storage models
: Data lakes are typically stored as files and managed by Hadoop, column–family databases, or key–value databases.

Indexing support
: Data lake technologies typically do *not* provide automated indexing support. To implement the ODH pattern, however,

real-time query and fast indexing is a key capability. Therefore, technologies such as document databases, which index document structure without a schema, make the ODH pattern possible. New NoSQL technologies that categorize and index data upon ingest and add metadata describing the data in the hub are ideal. For unstructured text content, each word in the document would ideally be indexed to support quick access. For structured data (such as XML and JSON), both data elements and the data values are indexed. In short: the more indexing the better to support query and flexibility.

Query capabilities
: Both data lake technologies and ODH technologies allow for data to be queried from its respective databases. Data lake technologies typically query data in a batch-oriented fashion but cannot be easily used to query data or serve master data results via a web service interface such as SOAP or REST services. ODH technologies typically allow additional features such as cross-document and cross-collection queries, interactive queries, and expose data via a web service interface.

Scripting and automation support
: ODH technologies support scripting through the use of scripting languages. Some ODH technologies also allow for triggers and alerts to be implemented, by which data movement and data processing are automated.

Multiple interface capabilities
: The ability to interface with multiple development technologies allows the ODH to interact with applications across the enterprise.

Transactional integrity
: Data lake technologies typically do not support transactional integrity. For the NoSQL database to be useful in the operational context, transactions need to be ACID compliant.[14] ACID (Atomicity, Consistency, Isolation, Durability) is a set of properties of database transactions that ensures the validity of data changes (insert, update, delete) that are committed to the database. An ACID-compliant transaction assures a user (or data-

14 "ACID properties - MSDN - Microsoft"

trading partner) that submitted data will be saved as expected, a characteristic that is critical for mastering data.

Enterprise security
: For domains in which regulatory requirements (such as HIPAA, SAS70, and Sarbanes Oxley) require access control of sensitive data and audit trails of updates at the element level, data lakes struggle.[15] Because the ODH will contain the most critical, most useful, and most sensitive data across the enterprise, it also needs to be properly secured[16] in order to be trusted.

Governance framework
: ODH technologies can integrate data from many sources, transform it to a standard form, and serve it in many ways, but all of these operations must be governed. Analytical needs depend on well-defined data traced to reliable sources. Some data must be reviewed by humans to resolve quality or duplicate data issues. This review process requires workflow and policy about changing data. Combining and transforming data to a common form relates to data mastering. And most of all, data must be secured. All of these processes—traceability, workflow, review policy, mastering, and security—comprise data governance.

Choosing the right NoSQL technology for the intended purpose is paramount.[17] Technology selection is based on the individual use case, deployment model, and budget of the organization; hence, a recommendation would be inappropriate here. However, it is crucial that each organization create an assessment model suited to the most important technology characteristics to measure potential technologies.

Although it is possible to implement an ODH with other types of NoSQL databases, document databases and triple-store databases (e.g., MarkLogic, Microsoft Azure CosmosDB, and MongoDB) are best suited for an ODH.[18]

15 "Securing NoSQL applications: Best practises for big data security"

16 "3 security pain points NoSQL must remedy - InfoWorld"

17 "The Forrester Wave™: Big Data NoSQL, Q3 2016"

18 "Enterprises hedge their bets with NoSQL databases - PwC blogs"

Planning to Clear the Data Swamp

Your data lake, after much investment of time and money, has turned into a swamp and has lost its value to the organization. Many organizations are considering the tremendous benefit of ODH technologies, but are hesitant to throw out the data lake and start all over. Fear not, all is not lost. You can consider the data lake as yet another data source to the ODH. But it is not as easy as just installing software and starting the processing; it is important to have a clear goal in mind and a plan on how to accomplish it. Lewis Carroll wrote: "If you don't know where you are going, any road will get you there."

We want to avoid exchanging a swamp in one technology with a swamp in another. The first step is to begin with proper strategic planning, which includes the following activities:

Agree on the objective(s)
: Clear objectives such as "We want to reach an integrated, mastered, view of a client profile" or "We want to have a clear idea of which datasets we have in our data lake, who uses them, and how they are used" will drive project focus and identify data needs along the way. These objectives can be adjusted as objectives are reached. It is alright if multiple objectives exist because ODH technologies mostly offer multitenant development environments. However, too many concurrent activities will require increasing cross-project coordination.

Identify data goals
: Most organizations that embark on big data projects, begin with a "collect-everything, keep-everything approach." This is typically what leads to the data swamp condition that we are trying to resolve. Start with the data that is directly tied to the goals we are trying to achieve. After the hub begins taking shape, other data sources can be added to expand on analysis capabilities such as click-streams. If the prospect of letting go of data is too much to bear, it might be best to keep a separate "keep-everything" database and take advantage of the ODH technology capabilities to categorize and index unused raw data for later search and retrieval.

Define intermediate goals

With a goal such as "develop enterprise data integration point to serve the needs of trading partners across all business operations," it is important to define intermediate goals. In healthcare operations, intermediate goals might include first trading client data, then claims adjudication, then claims payment, and then, finally, fraud detection analytics. Defining intermediate goals contributes to both managing expectations and ensuring that the planned objectives are met.

Use engineering patterns

As established earlier, a good data governance framework allows for data to be accessed, processed, and provisioned according to defined patterns. However, because we are using the capabilities of a schema-agnostic technology to implement progressive transformation, we do not necessarily need to conduct a comprehensive analysis of engineering patterns prior to starting our transformation. Working closely with the enterprise architecture group, the ODH team can identify, define, and utilize engineering patterns while the project executes. Using engineering patterns ensures a consistent approach to data ingestion, dealing with reference data, data mastering, and conflict resolution.

Assign ownership

Data quality and mastering routines typically identify conflicts and/or discrepancies between data sources. In an enterprise integration environment, this is a near certainty. Assigning data ownership for categories of data (such as client identifiers, client demographics, and product categories) ensures that automated conflict resolution is implemented consistently. Incorrect but consistent mappings can be addressed easily.

Focus on data quality

Because the quality of data in the swamp has deteriorated to the point where it adds little value to the enterprise, it is important to overcome that stigma from the outset. Reference data such as gender codes, ethnicity, race, population group, and age group are prevalent in the health and human services domain. However, these categories (e.g., age group) are often implemented by domain. Progressively agreeing on reference data mapping is imperative to meeting enterprise analytic objectives. However, project teams often miss opportunities to address data quality issues with immutable elements such as addresses. Ensuring

that accurate contact information for a client is often one of the primary goals for a consolidated view of a client. In this instance, an organization might decide to use validation routines (many vendors exist in this space) to indicate whether an address is valid and then to implement an interface in the client portal to clarify and select valid addresses.

Add immediate value
: With well-defined goals, an iterative approach to reach said goals, and engineering patterns to expand the ODH in a progressive fashion with good quality data, it is possible to reinstate big data confidence across the enterprise.

Involve data trading partners
: Data in the hub is only as useful as the value it holds for trading partners. For instance, if a client validates the correct address to use for correspondence, it poses a valuable event across the enterprise. Trading partners might want to query the latest data via a web service or to receive updates as they occur, either through a publish–subscribe model or more traditional data feeds. Data trading partners (internally and externally to the organization) will assist in identifying data quality challenges and objectives.

Get help
: NoSQL databases require a different frame of reference than relational databases. The same is true for the difference between any of the NoSQL databases. Document databases, graph databases, and multimodel databases require a different approach than column–family databases, such as Hadoop. Even if most of the work will be done internally, it is important to engage the services of consultants and obtain training to enable staff.

Transforming the Data Swamp into a Hub

Now that the high-level strategic planning activities and tool selection have been completed and the appropriate resources are available, we are ready to take advantage of the capabilities of the ODH technology to process data from the data lake and make it available according to the data governance model (Access, Ingest, Harmonize, Materialize, Provision, Consume).

The first step is to assess and categorize the data in the swamp. Information to capture includes the following:

Data source
: What system or process produced the data?

Purpose
: What purpose does the data serve? Is it a log file or a view of client demographics?

Business entities
: What data entities are represented in the data? A dataset might contain client demographics, address and contact details. These all form part of the client business entity.

The analysis of the swamp data (Figure 1-10) can either be performed in the swamp, or it can be ingested into the ODH, where it will be indexed and offer a searchable interface to make the process of categorization easier to accomplish.

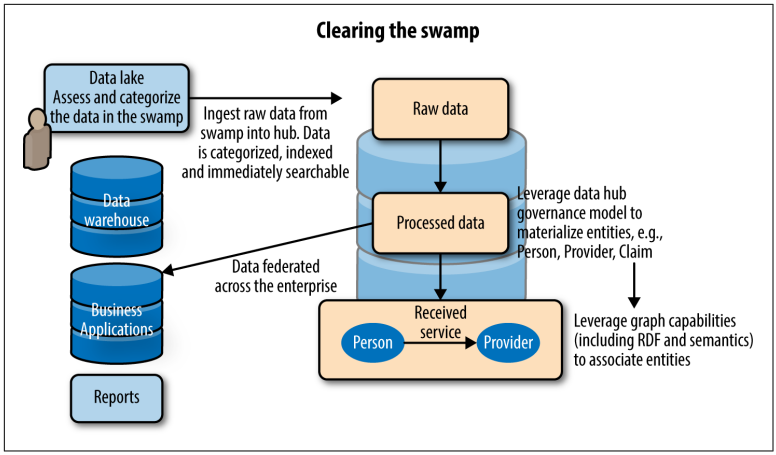


Figure 1-10. Clearing the data swamp

After business entities have been defined, you again use the data governance framework to harmonize, cleanse, master, and materialize data according to purpose. Person, Provider, Claim, etc. are possible business entities from the earlier example. This process does not need to be a comprehensive modeling exercise, but rather a progressive transformation as more is known about the raw data that exists in the hub.

In a multimodel database, you can now use graph capabilities (including RDF and semantic implementations) to associate business entities with one another so that cross-domain queries can be completed.

The process of actually clearing the swamp and progressively maturing the ODH might seem like a bit of anticlimax with all the hype around ODHs, but this is exactly what we want. With the advent of ODH technologies, we can use the capabilities of the tools and process data according to a well-defined governance model as opposed to the tremendous amount of data modeling and ETL processing that is required by traditional, relational, technologies.

After the data in the swamp has been cleared, the data lake can serve as yet another data source to the ODH and used for cases in which complex file processing is required before feeding a subset of data to the hub (for whatever reason).

Summary

Organizations of all sizes have struggled for decades to obtain an enterprise-view of its data across the many applications, databases, spreadsheets, and other data silos. Big data technologies enable organizations to ingest and process data in several formats and schemas to facilitate a consolidated view across the enterprise. However, many of these technologies are mostly focused on the data consolidation to form data lakes; a central store where we can extract data mainly via batch jobs for analytical purposes. With the tremendous amount of data that flows into these data lakes, often with questionable provenance and quality, the promise of an enterprise view of data quickly deteriorates into a data swamp. Although the data swamp still contains valuable data, the overall quality and confidence deteriorates to the point where its value is questioned.

NoSQL technologies have been developed with capabilities to index, process, govern, and secure data with transactional integrity of operations (insert, update, delete). These technologies allow organizations to implement an ODH that integrates, cleanses, masters, and serves data across the enterprise in an orderly fashion.

The ODH can retain the investment in the data lake through rigorous governance and progressive transformation to serve high-quality data to internal and external data trading partners.

About the Author

Gerhard Ungerer is the chief technology officer and cofounder of Random Bit, LLC, which provides enterprise architecture, software design, and governance services. He currently serves as enterprise architect and subject matter expert on a large-scale enterprise data integration project in the government sector.

Gerhard holds a bachelor of science degree in computer science and applied mathematics from the University of Stellenbosch (South Africa). He has worked in the software field for more than 25 years, including defense, insurance, finance, health, engineering, and human services.

Across sectors, he encountered the same basic challenge: to integrate data across organizational, functional, and technology silos. He proposes a data lake, with a well-formed and well-governed ODH, as an architectural solution to this challenge.