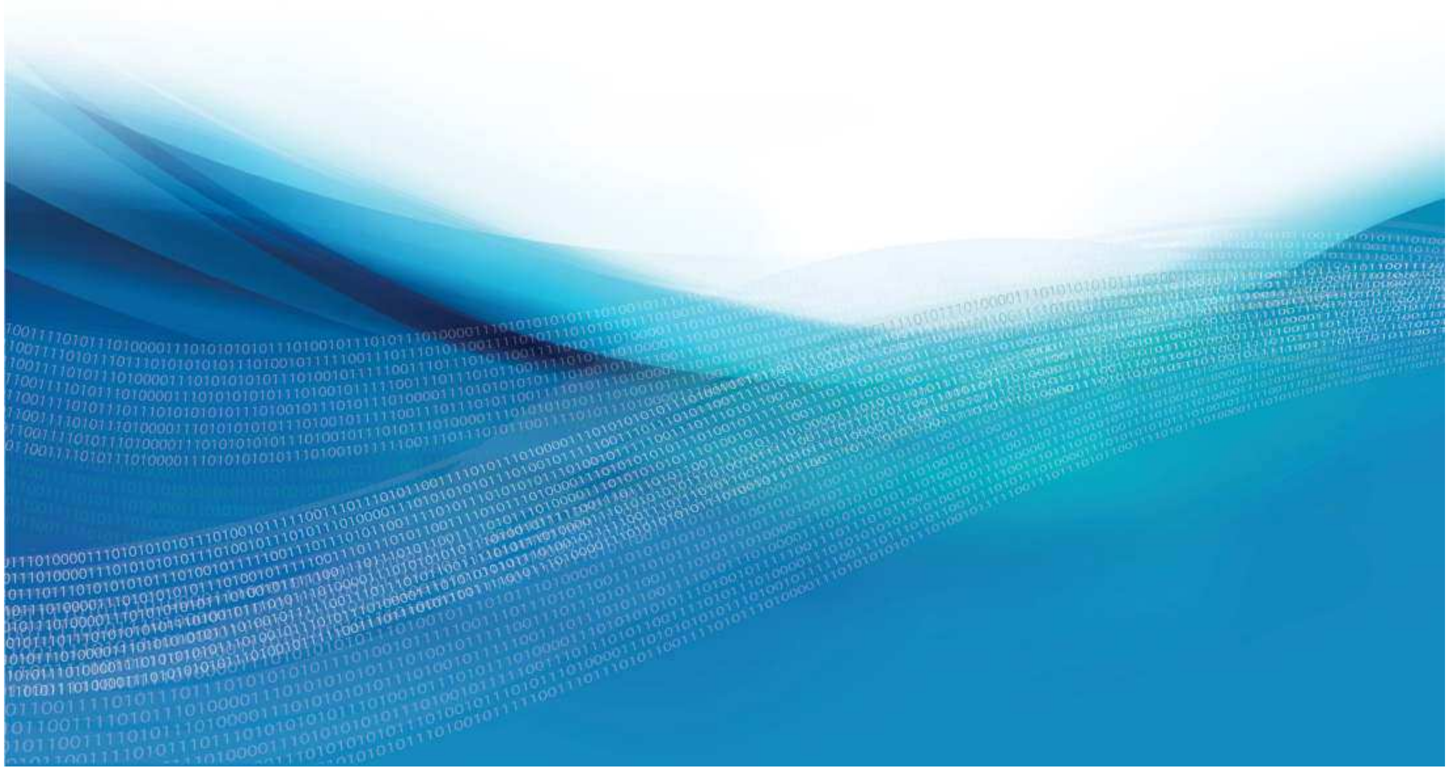
White Paper

Taming Data Drift

The Silent Killer of Data Integrity



StreamSets

Introduction

The emergence of big data has created tremendous opportunities for businesses to gain real-time insights and make more informed decisions by leveraging data from the exploding number of digital systems they have in use. However, as often is the case with disruptive technologies, the innovations behind big data have created a critical problem - one that we call *data drift*. Data drift creates serious challenges for businesses looking to fully harness the insights available from big data. In this paper we will discuss the causes of data drift, its technical and business implications, and how StreamSets Data Collector provides an open source solution to address data drift.

We define data drift as:

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

Today, many sources of important data, such as mobile interactions, sensor logs and web clickstreams, are constantly changing as those systems are tweaked, updated or even re-platformed by their owners. These changes to data content, structure, behavior and meaning are similar to genetic mutations, they are unpredictable, unannounced, and unending. We call this new fact of life *data drift* and it wreaks havoc with data processing and analytics systems and operations.

Data drift erodes data fidelity, data operations reliability and ultimately the productivity of your data scientists and engineers. It increases your costs, delays your time to analysis, decreases the productivity and agility of your data engineers and leads to poor decision-making by data scientists and the line of business.

From an operational point of view, data drift breaks the prevalent model of data processing which relies on schema-driven ETL (extract, transform and load). This approach is based on several assumptions that big data either violates or makes obsolete:

How Big Data Creates Data Drift		
	Traditional Data	**Big Data**
What	Data sources have a well-defined and stable schema. Schema changes are infrequent, predictable, publicly shared and opted into by the consuming organization.	Data often is treated as "exhaust" by the source system and arrives in a semi-structured or unstructured state so that the usefulness of schema as a defining criteria is limited. Changes occur often and without notice.

Where	The data technology stack is mature, stable, physically static and managed by a dedicated group in IT. Changes are adopted primarily through acts of IT governance.	Data is processed by a heterogeneous ecosystem of independent software components which are often virtualized or cloud-based. Changes are implemented as “shadow IT” outside of established IT governance.
Who	Data sources are controlled and managed by named entities in the organization.	Numerous data sources are managed – often poorly – by independent internal departments and third-parties.
How	Data processing and analysis is performed in batch mode.	A combination of batch, event-based and streaming operations must be accommodated.

In a world driven by big data, change is a constant and any processing architecture based on an assumption of stability and control is doomed. Due to data drift, legacy ETL processes are being constantly patched in an ad hoc fashion, making them brittle (they fail easily) and opaque (anomalous data can’t be detected while the data is in motion).

Data Drift Types and Their Challenges

There are three types of data drift: schematic, semantic and infrastructure.

Schematic drift: a change to the structure of the data at the source. Examples include addition, deletion, type changes and order changes to data fields.

Semantic drift: a change to the meaning of the data, even if the structure is unchanged. For instance, a change from imperial to metric measurement, from IPv4 to IPv6 addressing in a string or changes made to the meaning of coded field values by an upstream producer of data.

Infrastructure drift: a change to the data processing software that creates incompatibilities, such as an upgrade or addition to one of the many software components involved in data production, including virtualization or data center and cloud migration.

The challenges that data drift creates fall into two major categories: deterioration of data fidelity and damage to the reliability and productivity of the end-to-end data operations.

Real-World Examples of Data Drift

Unreliable Data

A bank adds leading characters to its text-based account numbers to support a larger customer base. Existing downstream systems - unaware of the change - start to conflate the data for multiple accounts (e.g. account 0023456 and account 0123456) leading to bad information in customer service systems.

False Insights

A digital advertising company sees a spike in its revenue. Upon further investigation it is determined that the measurement was a false positive driven by a change in an upstream data field from IPv4 to IPv6 address format, which led to misinterpretation by analytic systems.

Loss of Agility

A price prediction service regularly adds new external data sources to improve its predictive model and expand its market reach. These 3rd-parties often fail to consistently conform to data schema guidelines in which case data engineers must write custom code for each data set. This slows the onboarding of new sources, ties up valuable data personnel and makes the ingest operation unreliable and difficult to diagnose when it fails.

Loss of Data Fidelity

A serious negative outcome of data drift is a loss of fidelity in your data. Data fidelity issues take three forms: data corrosion, data loss and data squandering.

- *Data corrosion* occurs when data that has drifted passes into data stores for use by consuming systems, triggering errors or misinterpretations.
- *Data loss* occurs when information is removed from the data stream because data drift has caused it to run afoul of data rules.
- *Data squandering* is when new data fields have been added upstream but, due to a rigid adherence to schema, are discarded by downstream systems.

These effects are especially pernicious because they often go undetected for long periods of time, polluting data stores and subsequent analyses with incomplete, inconsistent and inaccurate data. Until detected, the use of this low-fidelity data can lead to false findings and damage the business. When the fidelity problem is finally detected, it is usually fixed through manual data cleanup and preparation by data scientists, which adds hard costs, opportunity costs and delays to data analysis.

Damage to End-to-End Data Operations

Data drift also degrades the capability of data operations. Reliance on well-defined schema means that custom-coded logic is required to keep ETL processes afloat in a world of data drift. This makes things brittle — these ETL jobs break easily. Also, since these solutions are blind to the characteristics of the underlying data as it mutates, they can't detect drift conditions in order to predict and prevent breakdowns. Nor are they instrumented to diagnose and fix drift-related problems. Brittleness and opaqueness cause the operational reliability of the data ingest process to deteriorate.

A byproduct of these operational issues is that valuable data engineers that are placed in fire-fighting services to code new data sources, maintain fragile pipelines and cleanse corroded data. As with the data fidelity problem, these operational issues create not just hard costs but also opportunity costs since there are fewer resources available to innovate or respond to new business requirements.

The Serious Business Impacts of Data Drift

As a result of these issues, data drift can greatly harm the business in the following ways:

Decisions Based on Bad Data

As they say, "garbage in, garbage out." Polluted, incomplete or misinterpreted data – low-fidelity data – leads to false insights and missed insights, which drives improper and potentially harmful business decisions. In addition, the delays imposed by having to prepare data for consumption delays time to analysis, which is deadly for real-time responsiveness.

Loss of Trust in Your Big Data

To paraphrase Warren Buffett, "trust takes a lifetime to build and only a moment to lose." Once the data has come into doubt, the reputation of your big data initiative is damaged and strategic role and funding is put at risk.

Loss of Agility and Effectiveness

It has been said that "the biggest expense is opportunity cost." The brittleness of ETL processes – whether hand-coded or tool-based – in the face of data drift makes it difficult to respond to new business requirements since even minor changes might break the operation. Plus, with your valuable data personnel devoting their time to "keeping the lights on" by patching pipelines or sanitizing polluted data, higher value tasks are pushed to the back-burner to the detriment to the business.

How StreamSets Battles Drift

StreamSets Data Collector is an open source continuous ingest tool designed expressly to arrest the insidious effects of data drift. Specifically, it allows you to:

- specify ingest pipelines based on intent rather than schema, which gives your data operations increased flexibility and resilience;
- sanitize your data in-stream so that it lands ready for consumption. In a drifty environment this means that you can specify automated processing steps for handling drift and;

- monitor the data contents as it flows through the pipeline so you can detect and route data drift conditions for exception processing.

Let’s look at how StreamSets Data Collector handles the three types of data drift.

Structural Drift
StreamSets Data Collector does not require the schema to be specified unless and until it is required. Structural changes that are irrelevant to the processing intent are simply passed through and changes to them do not break the ingest operation. We call this being *intent-driven* and it simplifies building pipelines and increases their operational reliability.

Semantic Drift
A semantic change is usually detectable as values that diverge from historical ranges. StreamSets Data Collector handles semantic drift in two ways. First, it intelligently monitors both the data and metadata for the data stream. You can apply data rules at any stage in a pipeline for immediate alerts about data drift. These anomalies can be handled by either human inspection or automated exception processing. Second, it provides data sanitization so you can use built-in processors to automatically transform the exceptional data and reintroduce it to your standard processing flow.

Infrastructure Drift
StreamSets Data Collector helps manage the constantly changing world of both source data infrastructure and data processing systems. For instance, it tames data drift in your virtualized environment. As new instances spawn, you can embed StreamSets Data Collector to ensure that data ingestion, cleansing and monitoring continues unabated. Also, when you upgrade or replace one of your data store or message queue technologies, the StreamSets Data Collector minimizes downtime to the ingest process by containerizing each component in the data pipeline. You simply branch a copy of your data flow through the new component, test the flow to ensure it operates as intended, and then connect the new component into your production flow.

How to Thrive in a “Drifty” World

As the world shifts from well-curated traditional data stored in RDBMS and enterprise applications towards the chaotic, real-time and ever-changing world of big data, data drift is becoming a fact of life for enterprise data operations. In this new era, chief data officers must build new assumptions about data drift into their “data operating systems” to ensure that they ingest efficiently, in real-time and with high fidelity. StreamSets Data Collector is an innovative ingest infrastructure that helps to meet these needs and tame data drift.

Download the StreamSets Data Collector
Click here to download the open source StreamSets Data Collector.