

WHITEPAPER | CASE STUDY

# Enterprise Test Data Management

## The Fundamental Handbook

Entity-based test data management tools give DevOps teams the ability to generate relevant and accurate test data – easily, quickly, and securely – to enhance business agility.

# The DevOps Data Problem

Despite the agile methods and automation you've implemented, if testing remains a major constraint, you're not alone.

A recent GitLab survey indicated that testing is "the area most likely to cause delays". Why? Because slow and partial provisioning processes create test data bottlenecks.

Shift-left testing is making the test data problem even worse. As testing moves upstream to earlier stages in the development process, and additional teams become involved, the need for fast test data provisioning is on the rise.

## Introduction

Test data has become a DevOps problem. But, it doesn't have to be.

DevOps and agility have become requirements in today's business environment – and this makes sense. The promise of reduced development times

and improved customer experience is a priority for every organization. To achieve these goals, testing is required to keep pace with continuous integration, continuous delivery (CI/CD) pipelines.

Further delivery acceleration and quality improvements are driven by shift-left testing, where testing is included in early stages of software development. The idea is to test early, and often, to catch bugs at the beginning of the pipeline, where they can be fixed more quickly and at lower cost.

The combination of agile development and shift-left testing is a difficult challenge. Even with the adoption of testing automation solutions, provisioning test data is still a slow, complex, and error-prone process.

The image shows a screenshot of a 'Tasks' table and a circular dashboard titled 'Create and Process Micro-Databases'. The table lists various tasks with columns for Task ID, Task Title, Source Environment Name, Target Environment Name, Business Entity Name, Task Type, and Operation. The dashboard includes a profile picture of a man and a list of operations with progress bars: Ingest, Unify, Enrich, Transform, Mask, Secure, and Compress.

Task ID	Task Title	Source Environment Name	Target Environment Name	Business Entity Name	Task Type	Operation
56	cleanCust	TAR	TAR	Customer	DELETE	Delete entity
57	yyy	SRC	SRC	Customer	EXTRACT	Extract entity
51	copyEntity	SRC	TAR	Customer	LOAD	Load entity false
50	ReserveEntities		TAR	Customer	RESERVE	Reserve entity true
49	testReserveEntityWarning	SRC	TAR	Customer	LOAD	Delete and load entity true false Entities
48	testLoadAndReserve	SRC	TAR	Customer	LOAD	Load entity true false Entities
43	testCustomLogicLoad	SRC	TAR	Customer	LOAD	Delete and load entity true false Entities
42	testCustomLogicExt	SRC	SRC	Customer	EXTRACT	Extract entity false false Entities
41	testParams2	SRC	TAR	Customer	LOAD	Load entity false false Entities
40	testParamsView	SRC	TAR	Customer	LOAD	Load entity true false Entities

According to SDLC Partners, data testing and data quality professionals are spending **over 30% of their testing time** dealing with defective test data.

Most spend 5 to 15 hours per week setting up, or addressing issues with, test data. That equates to at least one day per week wasted on resolving test data challenges.

## Challenges facing Test Data Management

Existing test data methods, like cloning production datasets or creating ad hoc samples, are too slow to keep up with modern DevOps and shift-left testing demands.

A continuous stream of application development needs faster turnaround times. Developers need to test applications and release new services in days. They can't afford to wait days or weeks just to get data.

Test data management is a costly component of the process because, even with current levels of automation, it's still time consuming and labor intensive. Challenges with data quality and security increase cost and also risk, making testing less effective.



### Test data accuracy

It's difficult to attain test data that accurately reflects the real-life use cases of the applications being tested.



### Complex extraction

Multiple production systems and data silos make it complicated to extract valid test data.



### Data corruption

Data can be corrupted as a result of multi-team testing processes.



### Time consumption

It takes a long time (days) to match, extract, scramble, and load test data.



### Data functionality

There is no seamless way to move data from production to target environments.



### Security and privacy

Duplicating production data increases the risks of security breaches and non-compliance with privacy laws.

---

## Better Test Data in Less Time

The key to better test data management is the ability to easily, quickly, and securely generate relevant and accurate test data. It's how an enterprise can reduce production cycles and lower costs.

DevOps and quality assurance (QA) teams need to be able to test against production-like data provided in minutes (as opposed to days).

This ensures they're able to more accurately test how an application will behave and perform – well before it's ready for deployment.

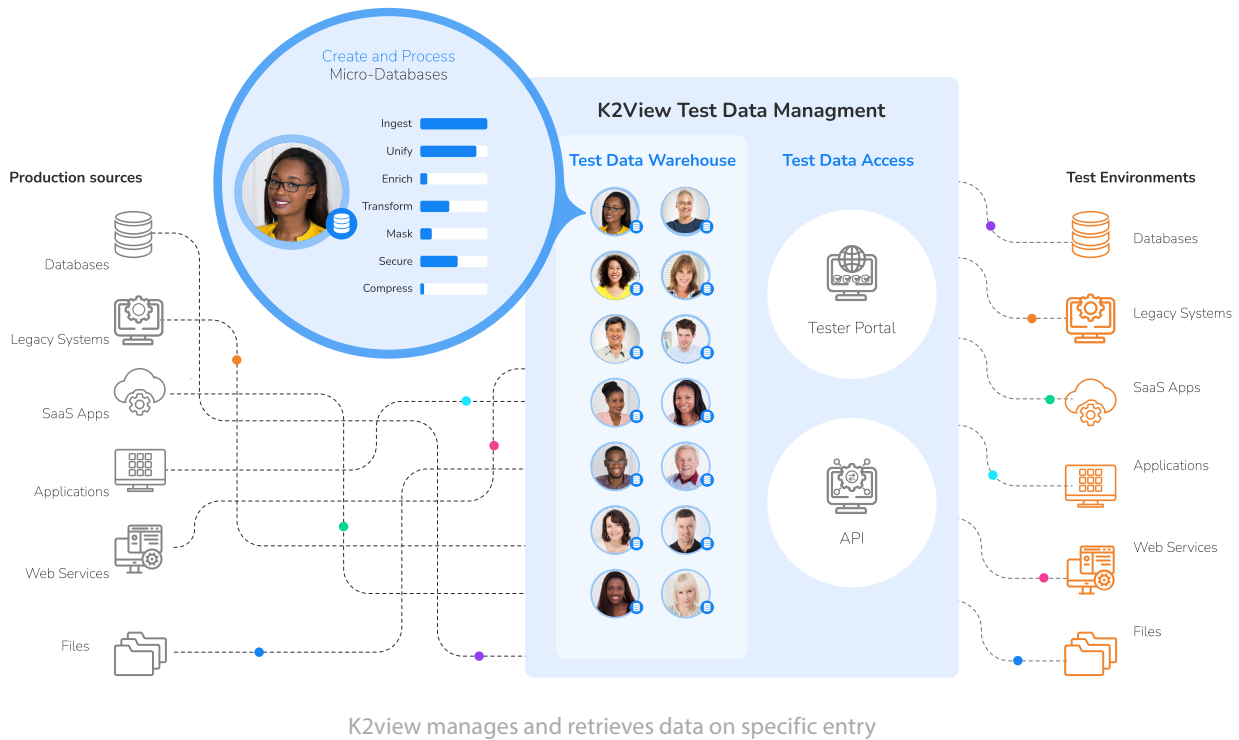
Most data management systems store data based on type (e.g., customers, locations, devices, etc.), translating into massive tables, that must be queried using complex joins, every time a user needs access to business data. It's cumbersome and time-consuming.

# Business Focus Test Data Management

Instead of storing data based on type, consider organizing data into business entity schemas. The data schema becomes a single, complete data source for everything an enterprise knows about a specific business entity.

## A new approach to Test Data Management

K2view Test Data Management enables faster application delivery, at a fraction of the cost, because our unique solution makes it easier and more efficient to take data from any source (without impact) or to create synthetic test data, and provide it to any type of downstream environment.



Our customers report tangible **benefits** that include:



**Substantial reduction**  
in cost  
for testing  
support



**Elimination of redundant**  
testing  
environments,  
and corresponding  
hardware and  
software costs



**Improved SPI**  
-compliant data  
protection



**Zero impact**  
on current  
systems and  
operations



**Dramatically reduced**  
test duration,  
and improved  
time-to-market

## What is a business entity?

A product, customer, facility, credit card – in fact, anything that’s important to the business. This unique approach allows for data to be delivered based on company needs, rather than some pre-defined structure.

For example, a “customer” entity might include three tables from the CRM system running on MySQL and five tables from the billing system residing on Oracle. This schema is used every time data needs to be copied or synthetically generated in business-focused test data management. Using embedded migration (ETL) capabilities, the data is processed, masked, stored, and distributed simply and quickly.

Managing data entities that are compressed, encrypted mini-databases improves performance, security, high availability and customizable data synchronization.

Unlike other solutions that automate an antiquated process, we create an “end state” of the data that provides the power, flexibility and simplicity needed to enable testing that keeps pace with your business. We then wrap it with a complete set of capabilities, from the continuous sourcing of the data, through the masking, modification and generation of the test data, to simple and secure access to the data for testing.

Instant Subsetting



## K2view covers every phase of the test data management lifecycle:

- Defining and sourcing:** Relevant data for test data management is done through a simple GUI and can be enhanced at any time.
- Refreshing and synchronizing:** Sync strategies and refresh rates for the test data are particular to each and every element of the business entity, allowing for full control over the test data.
- Cloning and sub-setting:** The business entity approach enables on-the-fly correction of data inconsistencies, even when the data is coming from multiple sources.
- Masking and securing:** Data is masked centrally, so even data from multiple sources is consistent, and the most complex rules can be implemented simply and efficiently. Each business entity is encrypted with a different key, for extra data protection.
- Creating synthetic data:** When defining a business entity schema, you also define a pathway to synthetic data creation. Generating synthetic data is as easy as generating synthetic business entities, whose definitions can be enhanced to comply with any requirement.
- Provisioning:** Test data management hinges on its ability to move data from many sources, to many target systems. Because K2view test data management is executed in memory, in a distributed environment, provisioning test data is quick and efficient.

## Putting K2view to the test: K2view Test Data Management

1. Makes it easy to set up, administer, and manage test data with user-friendly controls and an intuitive self-service portal
2. Speeds up the testing process (and ultimately, the delivery of applications) by retrieving data in real time
3. Automates cumbersome legacy processes, eliminates manual workarounds, and lowers IT costs
4. Accesses any data source, such as a traditional RDBMS, big data environments, flat files, web services, and cloud applications
5. Allows for data correction and/or modification, as the data is consolidated into a schema, and before it can impact testing
6. Supports SQL in a distributed data layer, allowing test data to be accessed without having to copy it to test environments
7. Simplifies sub-setting, due to the business entity's granularity, even for the most complex and fragmented data
8. Enables 100% linear scalability, without having to process massive amounts of data in bulk
9. Processes data in memory, and encrypts it when it is stored, making large data breaches impossible
10. Speeds up migrations significantly, with no system downtime during deployments

## Let's re-think test data management

### Traditional TDM

#### Waterfall, bulk provision

- Many handoffs to request and provision data
- Weeks to provision; days to refresh
- Inaccurate, not what the user needed
- Lengthy masking process



### K2view TDM

#### Agile, data as a service

- Self-service data for Development, QA and Dev Ops
- Populate and refresh with the right data **in 80% less time**
- Centrally mask data in minutes, not weeks
- Native SQL support
- Substantial reduction in manual processes

#### Siloed, system-based

- Isolating bugs or data issues cross applications difficult
- Finger-pointing among teams due to lack of visibility
- Defects found late in QA cycle



#### Visible, business entity-based

- End-to-end visibility of data across applications
- Real-time identification of root cause
- Faster problem resolution

#### Unsecure, complex, expensive

- Masking is seldom consistent, rendering data not usable
- Breaches expose large amount of data
- Expensive HW and Licensing



#### Secure, simple, savings

- Consistently and intelligently masked
- Secured at each logical unit, eliminating mass data breach
- Runs on commodity hardware
- Virtualized, compressed and distributed **performance at 40% of cost**



## Key challenges

Communications firms, including wireless carriers, cable, and media companies, have enormous amounts of data on their customers, networks, devices, plans, and more. Providing access to the right data at the right time – and to the right person – can transform the customer experience.

That's true whether it is via the web, or mobile apps, or simply providing the support team the insights it needs to provide fast, knowledgeable service.

That's what DevOps is all about – enabling companies to quickly develop, test, and deploy customer and operational systems that keep customers happy.

Traditional Test Data Management (TDM) involves manual requests, multiple teams, and time-consuming database backups and restores. This means provisioning test data alone can take days or even

weeks. Despite the agility a DevOps pipeline promises, this prolongs each test cycle and forces a time-to-market cycle from three to six months—at best.

In addition to negatively impacting time-to-market, traditional TDM's dependence on manual processes means data experts are tasked with repetitive busy-work instead of higher value-added work. That means the cost to support testing – not to mention masking data to keep it secure – continues to rise over time.



Testing these apps is where most DevOps and Agile development pipelines slow to a crawl. That's because the data needed to adequately test and re-test each app is trapped in siloed systems and databases.

## Solution and results

### Unique approach to test data management delivers measurable business outcomes

What AT&T needed was the ability to rapidly access the data it needed for testing, then organize it, and deliver on demand. With K2view TDM, AT&T can now provision fresh data, scrub and mask it in-flight, reducing test data creation to a self-service task that takes just minutes, not days or weeks. K2view TDM also eliminates redundant test environments, saving costs on software, hardware and administration, while having zero impact on current operational systems.

### Provisioning test data by business entities

K2view TDM takes a business entity approach to provisioning test data. It enables test teams to model 'digital entities' - corresponding to business entities, such as customer, household, location, device - to provision trusted test data with referential integrity. Testing teams can access the data on demand, via a

web portal or APIs, cutting test data preparation efforts significantly. Continuous data sourcing from ANY source K2view TDM's Micro-Database design allows any data source—traditional RDBMS, legacy systems, big data warehouses/lake, flat files, web services, cloud apps and more—to be integrated, organized into business entities, and continuously updated to the test data warehouse.

### Central, dynamic masking, modification, and generation of data

K2view TDM can access and automatically subset data without massive bulk copying, and perform in-flight data masking, transformations, and validation. That means test data is always current, compliant, and in the structure needed for the test systems. Finally, test data automation is achieved by integrating the test data provisioning process with the CI/CD pipeline through APIs.



## About K2view

**At K2view, we believe that every enterprise should be able to use its data to be as disruptive and agile as the best companies in its industry.**

**We make this possible by enabling data teams to transform fragmented data into complete and compliant data products – in real time.**

Data products could be customers, products, suppliers, orders – or anything else that's important to your business. We manage every individual data product in its own secure Micro-Database™, continuously syncing it with all source systems, and making it instantly accessible to authorized data consumers

This is all made possible by our Data Product Platform, which delivers a trusted, real-time view of any business entity. The platform deploys in weeks, scales linearly, and adapts to change on the fly.

It supports modern data architectures, such as data mesh, data hub, and multi-domain MDM – in on-premise, cloud, or hybrid environments.

This one platform drives many use cases, including application modernization, cloud migration, customer 360, data privacy, data testing, and more – to deliver business outcomes in less than half the time, and at half the cost, of any other alternative.