



Virtualizing Data with the Delphix DevOps Data Platform

Enterprises are focused on initiatives that accelerate application development to enhance customer experience, out-innovate the competition and win market share.

Projects under these initiatives often require production-quality data to build applications at a faster rate, with more features, and at a higher quality. The Delphix DevOps Data Platform helps enterprises consistently deliver production-quality data to these initiatives' stakeholders across all phases of application development including testing, release and production fix. The platform ingests data from heterogeneous data sources, virtualizes that data and delivers it to key stakeholders as lightweight virtual data copies.

In this whitepaper, we will discuss how Delphix ingests, virtualizes and delivers data.

The reader should be aware of the following Delphix terminology:

-
- » **Blocks or Data Blocks:** The set of storage blocks mapped to the entire data set, which can be a production database. Blocks can be individually addressed, refreshed, compressed, and accessed.
-
- » **Data Source:** The source, typically a database or file, that is ingesting data to the platform, and from which virtual data copies are derived. A data source can also be an application, a VDB, or a set of unstructured files.
-
- » **DevOps Data Platform:** Combines data compliance with API-first data delivery.
-
- » **dSource:** An object that is an internal compressed representation of a database, data file or binary created by the platform. As a virtualized representation, it cannot be managed, manipulated, or examined by database too.
-
- » **dxToolkit:** A set of pre-compiled Perl executables performing calls, which are available for both Windows and Unix platforms.
-
- » **Replication:** The ability to replicate objects (dSources, VDBs) between source and target.
-
- » **Source Database:** The original (sometimes physical) database that is usually a production database, although it could be any database that the database administrator designates as a source. The platform creates a dSource from the source database.

- » **Source Environment:** A server/host from which the Delphix Engine can capture data, which usually resides in the production environment.

- » **Target Environment:** A server/host or cluster to which the Delphix Engine will provision VDBs.

- » **TimeFlow:** A rolling record of file and log changes of source database or virtual databases retained by a policy in the DevOps data platform.

- » **Virtual to Physical (V2P):** The process of creating a physical DB from a virtual database or dSource, for example for performance testing or disaster recovery.

- » **VDB (Virtual Database):** A database provisioned from either a dSource or another VDB, which is a full read/write copy of the source data. Each VDB is created and managed by the DevOps data platform and has its own TimeFlow.

- » **Virtualization SDK:** A software development kit used to create and integrate extensible data source integrations with DevOps data platform.

In addition to the glossary above, the reader should familiarize themselves with figure 1 which shows how DevOps data platform works in the data center on-premises or in the cloud.

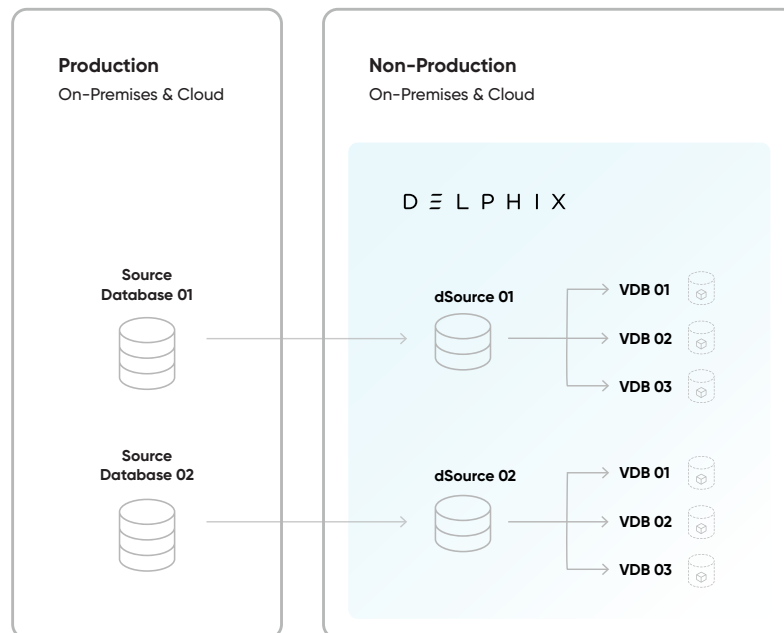


Figure 1. DevOps data platform deployment

Introduction

The Delphix DevOps Data Platform runs on-premises and in the public cloud. Delphix ingests heterogeneous data sources, remains in sync with changes from those sources and creates virtual, fully read/writeable, lightweight copies known as virtual database (VDB) for the user. As shown in figure 2 the platform ingests data from data sources such as Oracle, SQL Server, MySQL, AWS RDS and distributes it to the users in different teams such as software development, testing, and analytics.



Figure 2. Production data from different applications and data sources delivered to users

Connect with Data Sources

The DevOps data platform ingests data from application databases in a production environment or from the enterprise backup copies. The platform links to an external database and copies the entire file and log blocks of source databases to it. The platform then creates a compressed representation of a database called a dSource. After the initial loading, the platform keeps each dSource synchronized with source databases based on pre-defined or customized policy. For example, once daily, or within seconds of the last transaction. Once linked, Delphix maintains a TimeFlow of the source database — a rolling record of file and log changes retained according to a policy (for example, „keep for two weeks.“). Connection to a data source is:

- » **Non-disruptive:** No changes to the source database workflow are required or made during the setup or later.
- » **Minimal Touch:** The DevOps data platform uses standard protocols and APIs to pull changes from the source database and is configured to run according to policies that minimize impact to the source database. Data can also be ingested from backup systems when the organization does not want to touch their production data sources.
- » **Incremental and Continuous Sync:** Only the changed data is sent after the initial setup and one-time full data ingestion is completed. The engine only requests and captures the changed data blocks. Synchronization to the production data source resumes from the previous point, maintaining a continuous history of changes, even if the operation is interrupted or if the servers become temporarily unavailable.
- » **Organizational Policy Driven:** TimeFlow parameters can be configured to the organization's data governance and retention policies for each data source. Multiple retention policies can be put in place based on the organizational requirements for each data source. This provides an easy and straightforward alignment with enterprise data management policies.

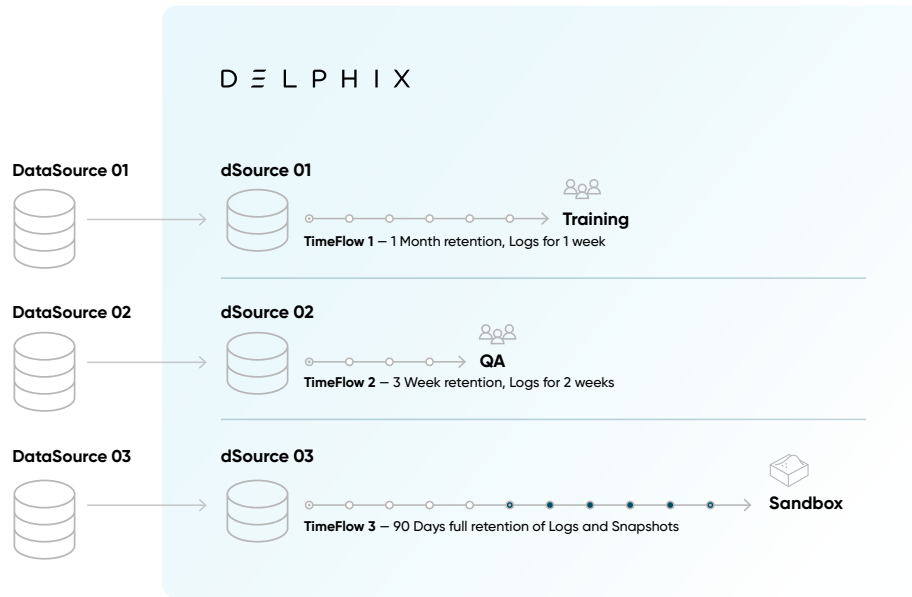


Figure 3. TimeFlow setup according to organizational data retention policies

Figure 3 shows how data from three heterogeneous data sources have their own use defined TimeFlows. For example, dSource 1 is configured to retain data for 1 month and logs for 1 week to meet a 5 week sprint cycle for application development. Similarly, TimeFlow 3 is configured to keep snapshots and logs for 90 days for quarter-end reporting.

Heterogeneous Data Source Support

Delphix provides both out-of-box support and extensible data source integrations for data ingestion. Delphix supports leading databases such as Oracle, SQL Server, SAP, DB2, SAP and ASE out of box. Extensible data source integrations based on the Delphix Virtualization SDK can ingest data from heterogeneous data sources, including SQL, NoSQL, MongoDB, PostgreSQL, AWS RDS, and more. With Virtualization SDK, users can create their own integrations for ingesting data from data sources of their choice and need into the platform.

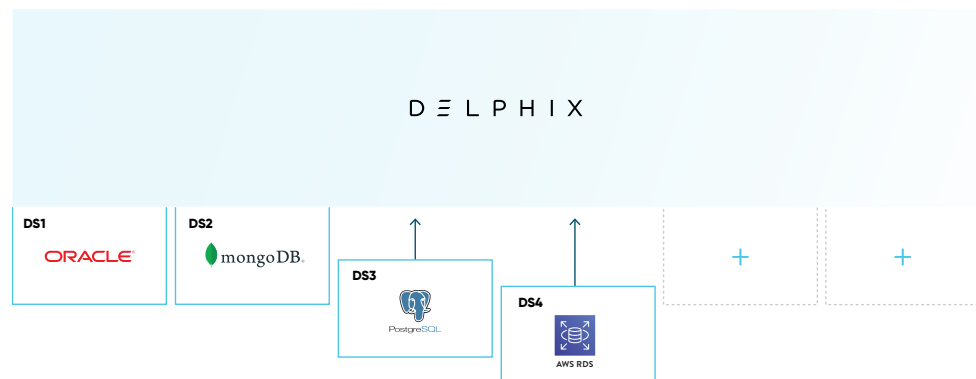


Figure 4. Extensible Data Source Integrations

Virtualize Data

The Delphix DevOps Data Platform virtualizes data to provide complete, fully-functional virtual databases in a fraction of the space, with improved agility and manageability, and API data controls.

Delphix combines traditional snapshot and cloning underlying storage technologies with database awareness and intelligence, abstracting key features — such as integrated log management and synchronization across copies — to streamline database management. The following features focus on delivering improved performance for data storage and retrieval:

Compression

Delphix leverages the ZFS filesystem and compresses incoming datasets at the block level, maintaining the ability to access data block-by-block. Compression is applied to the copy of the database files and transaction/redo log files when ingesting into the platform. When changes are made to a VDB, Delphix compresses new blocks associated with that VDB before writing them to disk.

The DevOps Data Platform uses copy on write technology to allow child VDBs to share compressed blocks from their sources. The platform caches the blocks in compressed format to maximize the memory allocated.

The Delphix Data Platform virtualizes data to provide complete, fully-functional virtual databases in a fraction of the space, with improved agility and manageability, and API data controls.

Lightweight Virtual Databases

The DevOps Data Platform creates lightweight virtual database copies combining changes to the dSource over the time and associated transaction log management done by the platform.

As seen in the figure 5 below, the platform does a one-time full data ingestion of the data source, creating a dSource and starting a TimeFlow for that dSource. Over the time, Delphix keeps track of incremental changes and updates the dSource on Delphix as a series of snapshots, in addition to maintaining a transaction log.

When a VDB is created between snapshots like VDB 1 and VDB 2 below, the platform uses transaction logs to create the virtual database. Virtual databases take up very little storage space. The amount of space consumed is directly proportional to the amount of data changed. This makes delivering copies a matter of sharing a virtualized view of the data the platform has already collected, rather than making and moving new data blocks.

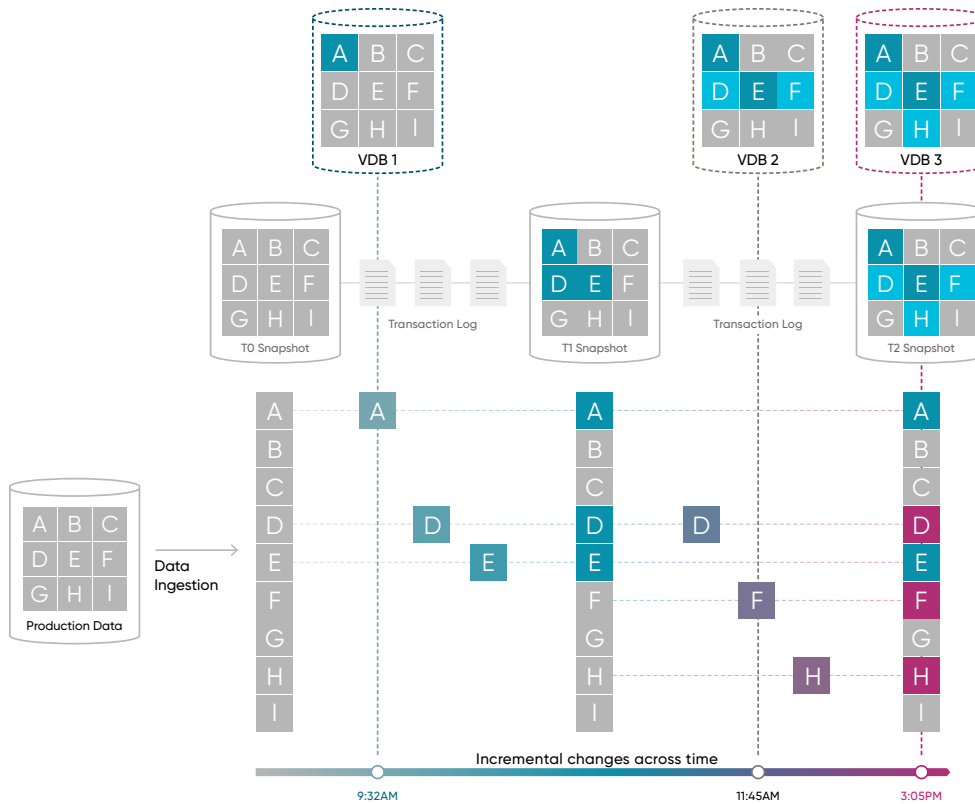


Figure 5. Creating VDB with Delphix

Replication

The Delphix DevOps Platform virtual instance can replicate data to another Delphix virtual instance. As shown in the figure 6, data can be replicated from the production environment to a non-production environment, either on-premises or in the cloud, with two instances of the virtual instances.

After an initial one-time sync, only changed data is sent to the target, dramatically reducing network load and associated cost. If a source instance fails, the enterprise can use the target instance to re-provision real database from virtual environments. In addition, the user can provision VDBs from replicated copies, allowing for the geographical distribution of data and remote provisioning.

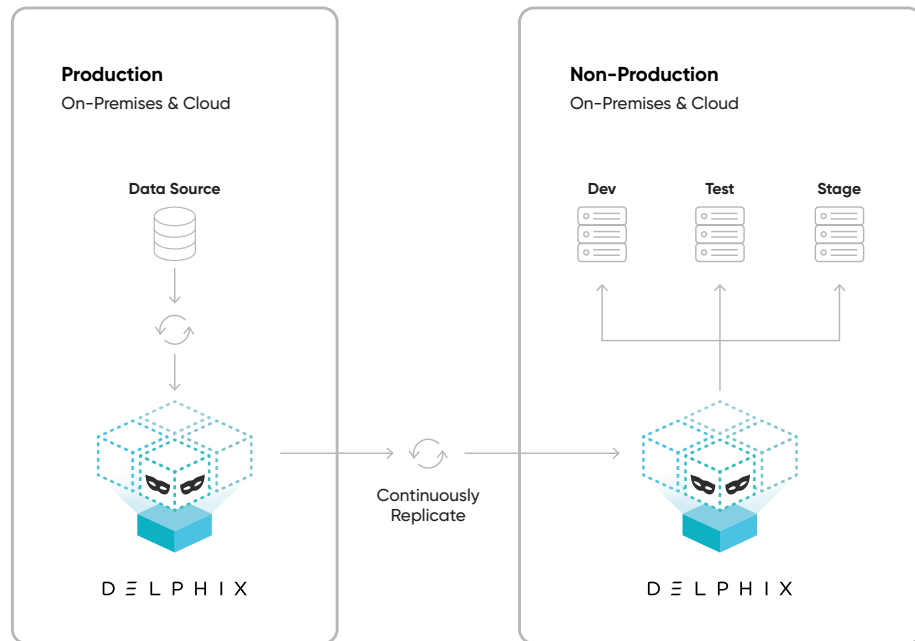


Figure 6. Replicating data across data centers with Delphix

Reminder

- » **dSource** is virtualization representation of a database. It cannot be managed, manipulated or examined by database tools. As a virtualized representation, it cannot be managed, manipulated, or examined by database too.
- » **VDB** is a virtual database full read/write copy of the source data.
- » **TimeFlow** is rolling record of file and log changes of source database or virtual database retained by a policy.

Deliver Data to Users

Delphix delivers data to the end user as a Virtual Database. A VDB is a lightweight read/write database copy created from the TimeFlow of a dSource or another VDB. It is configured, set up, provisioned, started, stopped, migrated, and managed by Delphix. VDB provides the same data related tasks and workflow experience as a physical database to the user. A VDB has the following benefits:

Scale

Multiple VDBs can be provisioned from any point in time through TimeFlow of any dSource or VDB. Once provisioned, a VDB is an independent, read/write database, and any changes made to the VDB by users or applications are unique to the user.

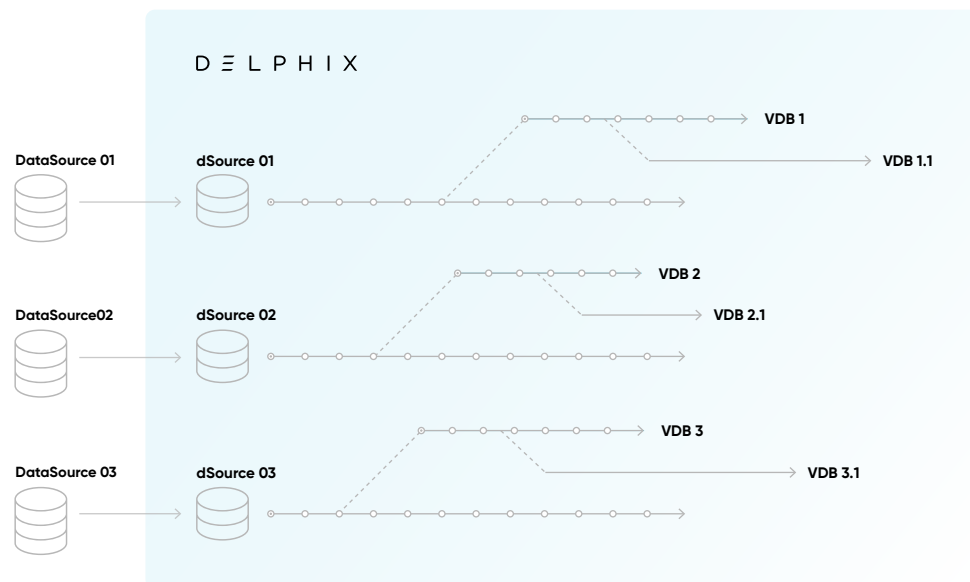


Figure 7. Scaling VDBs

When multiple VDBs are created, as shown in figure 7, each VDB has its own TimeFlow. VDB 1.1, VDB 2.1, VDB 3 TimeFlows in figure 6 will have the changes from their parent sources which are dSource 1 and VDB 1, dSource 2 and VDB 2, dSource 3 here respectively.

In application development, multiple VDBs from same parent VDB or dSource can be used for subprojects or quick sprints. This allows multiple developers and testers to work on different parts of a project that require data from different points of time. For example, VDB1.1 in figure 7 can be created by the developer who is using VDB 1 to share his personal data environment with his testing team to start a QA cycle for a new feature. Similar TimeFlows can be used for multiple data sources from other VDBs for different projects such as analytics, ad-hoc queries, modeling scenarios, etc. Changes unique to an individual VDB cannot be merged back to original branch.

Fast Provisioning

VDB provisioning from the Delphix DevOps Data Platform is fast and can be automated with user-defined workflows. Since newly-provisioned VDBs point to existing data blocks in the platform, they take up minimal incremental space and can be created and delivered in minutes.

Same Data Management Tasks Across Data Sources

VDBs support a variety of operations like physical databases such as refresh, rollback and snapshot, providing the user the identical data environment to a physical database without production database access restrictions. The platform records change data, allowing VDBs to be provisioned from any point in a TimeFlow, refreshed to current state, a specific point in time, or a previous state — as shown in figure 8 below. Users can directly perform data management tasks in minutes. Enterprise policy-driven daily rollbacks for patch and upgrade test environments can be done by users instead of IT.

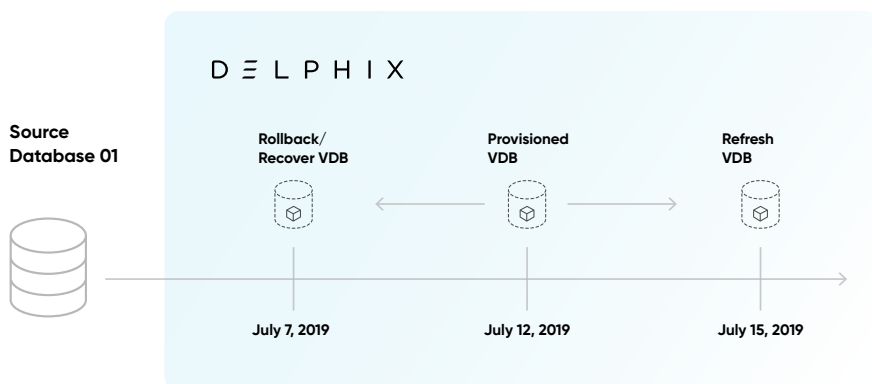


Figure 8. Rollback, Recover and Refresh with Delphix

Integrated Synchronization, Provisioning, Rollback and Refresh

VDBs can synchronize virtual data copies from heterogeneous data sources in near real time without affecting the corresponding data sources, therefore providing the user with the latest production-quality data.

Users can also selectively synchronize a VDB to a specific point in time to analyze a problem or test a feature as figure 9. For packaged applications and certain business projects (such as reporting, business intelligence, master data management) data management tasks have to be synchronized across sources. Generally data synchronization or snapshot specific requests for a particular timestamp are often challenging because different business lines own and manage the data sources. Even if all owners provide concurrent access, extracting the same time slice of data or refreshing to a new point in time across sources is challenging. Normal refreshes can take days, and federated refreshes only add complexity. Invariably, wait times for data access and technical challenges with synchronizing data extracts reduce data quality and consume test time, increasing the risk of project failures.

Delphix can connect to multiple sources and enable integrated, API-first data delivery. Independent access to production data (databases, config files, binaries) eliminates reliance on production teams. The elimination of organizational dependencies, process overhead, and hours of labor around these data integration projects can speed up project schedules by cutting days of wait time, increasing time for testing, and improving data quality, thereby increasing project success rates.

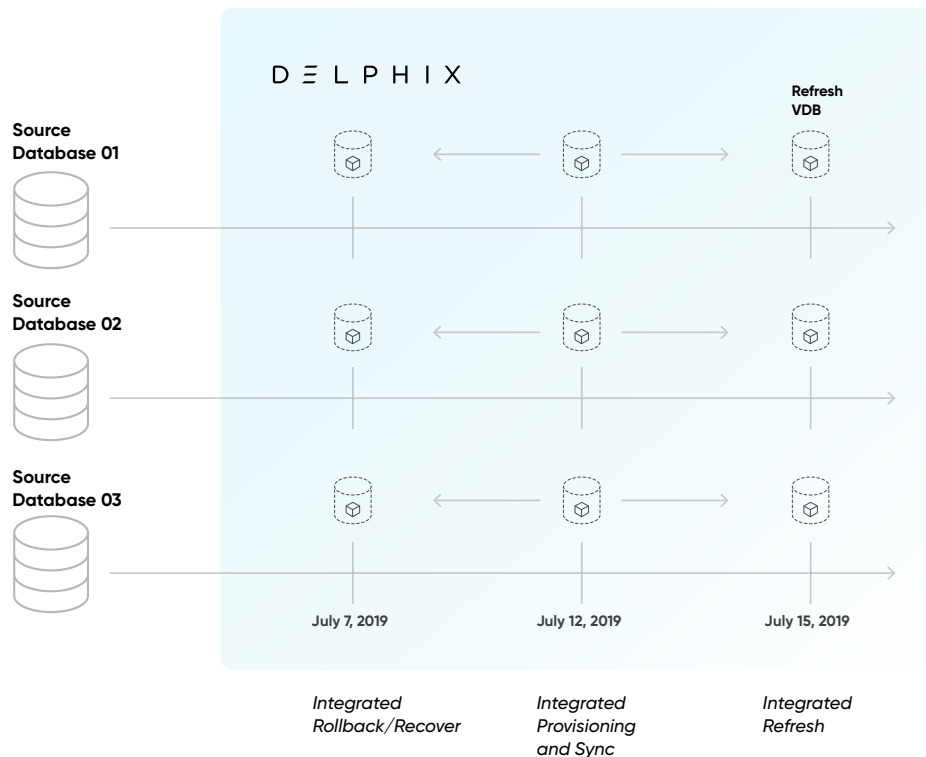


Figure 9. Integrated Provisioning, Synchronization, Refresh and Rollback with Delphix

Virtual to Physical (V2P)

Delphix can also convert virtual data back to a physical state for UAT or production scenarios. V2P can also be used in the event of a production data source failure. V2P can be used to recover a working physical production copy from the last known good copy. Interim production workloads can then be replayed on the new physical instance, minimizing downtime to users.

APIs and Toolkits

Delphix provides a robust set of RESTful APIs and CLI for users to integrate the platform features and capabilities as part of their workflows for infrastructure monitoring, software development, test data management, and analytic tools. Delphix also provides DxToolkits, a set of pre-written scripts for performing Delphix API calls in UNIX executables. DxToolkit allows users to programmatically work with Delphix without writing their own extensive API calls. DxToolkits can be used to get underlying Delphix infrastructure performance information such as CPU utilization, number of VDBs, storage consumption, network bandwidth, etc.



Summary

Delphix provides enterprises with the unmatched benefits of API-driven data provisioning at scale for their business initiatives. By shrinking the storage requirements and speeding up the replication of data, the Delphix DevOps Data Platform enables accelerated deployment of resources, enhanced cost savings, and faster transformation in enterprise digital initiatives. API-first data controls give different user groups access to the data they need while individual TimeFlows allow for navigating business changes with greater agility.

D E L P H I X

Delphix is the industry leading data company for DevOps.

Data is critical for testing application releases, modernization, cloud adoption, and AI/ML programs. We provide an automated DevOps data platform for all enterprise applications. Delphix masks data for privacy compliance, secures data from ransomware, and delivers efficient, virtualized data for CI/CD.

Our platform includes essential DevOps APIs for data provisioning, refresh, rewind, integration, and version control. Leading companies, including UKG, Choice Hotels, J.B. Hunt, and Fannie Mae, use Delphix to accelerate digital transformation. For more information, visit www.delphix.com or follow us on [LinkedIn](#), [Twitter](#), and [Facebook](#).
©2021 Delphix