TDWI RESEARCH

TDWI CHECKLIST REPORT

Data Synchronization

By Philip Russom

Sponsored by

GOLDENGATE™

www.tdwi.org

tdwi
THE DATA WAREHOUSING INSTITUTE

TDWI CHECKLIST REPORT

APRIL 2009

Data Synchronization

By Philip Russom

TABLE OF CONTENTS

FOREWORD

The amount of operational and transactional data being integrated and synchronized across enterprise applications continues to grow. As a consequence, tools and techniques for data synchronization are used today at an unprecedented level. The practice of data synchronization—or simply “data sync”—is driven up by leading trends in data-driven business activities.

For example, many corporations create a 360-degree view of the customer by synchronizing customer data across multiple customer-facing applications; other firms apply data sync to building a complete view of products, employees, suppliers, and other business entities.

Since data sync can operate in real time—or any speed that’s appropriate—it’s ideal for distributing up-to-date information in support of popular time-sensitive business practices, such as just-in-time inventory, customer service, and operational business intelligence.

As the economy becomes ever more global, mission-critical applications must operate 24/7. Data sync is a tried-and-true strategy for database high availability, and it can handle the bidirectional active-active configurations that are becoming the standard architecture for database high availability.

These trends and use cases demonstrate that data synchronization is an amazingly versatile technology and practice that has many valuable applications across an enterprise. This TDWI Checklist Report celebrates the versatility of data synchronization by showcasing many of its valuable capabilities and popular use cases.

☑ NUMBER ONE

USE DATA SYNC FOR 360-DEGREE VIEWS, REAL-TIME DATA, AND HIGH AVAILABILITY

There are many successful use cases for data synchronization, but most boil down to three popular categories:

- **Assembling a complete, 360-degree view of key business entities.** When IT systems share data in common—typically about business entities such as customers, products, or financials—it may be necessary to synchronize data across the redundant systems so the view of these business entities is the same from each application and its database. For example, data synchronization regularly syncs customer data across multiple CRM and CDI solutions, and it syncs a wide range of operational data across ERP applications and instances.
- **Delivering up-to-date data in real time.** One of the strengths of data synchronization is its ability to operate in real time, as well as other speeds. Synchronizing operational data has a noticeable and positive impact on time-sensitive business processes like customer service, material management, fraud detection, cross-selling, and up-selling. It may also boost employee productivity and improve the quality of decision making at the operational level.
- **Providing database high availability.** By definition, a mission-critical information system needs to operate continuously. Data sync is a well-known and trusted method for providing database high availability, whether with a live standby, active-active, or multi-master configuration. Also, data sync regularly provides high availability for a database to avoid downtime as it’s being migrated, consolidated, or upgraded.

As another sign of data sync’s versatility, a single implementation can support all the mentioned solution types. On occasion, two or more of these solution types combine. For example, information can’t be delivered in real time unless the database it comes from is highly available. Complete views of business entities achieve greater value when they’re updated continuously in real time.

NUMBER TWO

DEFINE DATA SYNC BY ITS ADVANCED CAPABILITIES

Data synchronization is the process of making multiple data sources agree in terms of content (data values) and structure (data models or schema). A data sync process may run continuously, infrequently, or only once. The data may flow one way from a primary data source to secondary targets, or it may flow bidirectionally or multidirectionally among all data sources and targets involved in synchronization.

Another definition of data synchronization is that it's an advanced configuration of replication. But this is a bit misleading. While it's true that many users' data synchronization solutions are built with replication technology, other technologies can also implement or contribute to data sync, including enterprise application integration (EAI), extract, transform, and load (ETL), and hand coding.

There are prominent data sync project requirements that you should keep in mind, because they tend to differentiate basic configurations from more advanced ones:

- **Direction of data flow.** Although most replication configurations move data one way (typical of high availability or disaster recovery), data sync is inherently bidirectional or multidirectional. By definition, it moves data two or more directions among multiple databases, files, applications, and so on.
- **Conflict resolution.** The multidirectional nature of data synchronization may give it the added burden of resolving conflicting data values, depending on the use case. After all, if data sources and targets are being updated regularly, it's possible that some data values will conflict when they are compared during synchronization. Note that the development of a data sync solution usually entails defining rules for resolving data conflicts.
- **Transactional data integrity.** Much of the data handled by a data sync solution originates as database transactions. When transactional data is synchronized across transactional systems, the data integrity of the target system must be assured by syncing data through a transaction processing mechanism.
- **Heterogeneous sources and targets.** Data sync solutions regularly involve interfacing with a diverse collection of data sources and targets. This is especially common when synchronizing customer data across CRM applications or operational data across ERP applications.
- **Data transformation.** Basic replication configurations only need to copy data unaltered from a data source to a target. However, heterogeneous data environments demand data transformation capabilities, if for no reason other than normalizing and merging complex data coming from diverse schema.

Hence, if you're contemplating an advanced application of data sync, you'll need support for multidirectional data flow, conflict resolution, transactional data integrity, heterogeneous data sources and targets, and data transformation.

NUMBER THREE

SYNCHRONIZE DATA WHEN MULTIPLE SYSTEMS SHARE COMMON INFORMATION

A prominent goal of data synchronization is to build a complete view of key business entities—a view that application users in multiple departments can access and use.

Synchronize customer data across customer-facing applications. The most common example of data synchronization involves customer data. Many organizations have multiple applications for customer relationship management (CRM) or similar customer-facing functions, such as sales force automation (SFA), call center, order entry, billing, shipping, and so on. All of these share common information about the organization's customers, and business units are increasingly under pressure to have as complete a view as possible of all customer activities across an enterprise. Hence, many data sync solutions synchronize customer data (whether for operational or analytic purposes) across multiple CRM and CRM-like applications and their databases.

Synchronize operational data across related applications. Similar to the situation with CRM applications, many firms have ERP applications from multiple vendors or multiple instances of one vendor's application. Again, data sync helps end users of individual applications see a more complete view of business processes that reach across multiple ERP applications. In cases with multiple instances, data sync can make all the instances look like one global application instance.

Synchronize analytic and operational databases. Fresh, up-to-date operational data is a critical success factor for relatively new analytic practices, such as performance dashboards and operational business intelligence. Data sync can move data from operational applications to a data warehouse or mart, thus enabling analysis and reporting for time-sensitive business activities like inventory management, yield management in manufacturing, sales monitoring in e-commerce, and so on.

Synchronize a variety of data domains. Other data sync examples include synchronizing master data across multiple applications, synchronizing mobile devices with enterprise databases, and synchronizing primary and secondary databases for the purpose of high availability or scalability. In these cases, a wide variety of data domains (from operations to financials) require synchronization across instances.

NUMBER FOUR

DELIVER SYNCHRONIZED DATA IN REAL TIME OR AT THE RIGHT TIME

Some data synchronization technologies are capable of moving data in less than a second, assuming the data's processing is straightforward. Therefore, you should depend on data sync for data that requires near-real-time delivery, especially since there aren't many other technologies that operate so quickly with transactional data integrity.

Yet not all business processes need such speedy data delivery. In fact, most business processes tolerate latency in the flow of enterprise data. From monthly to microsecond, data sync should be configured to move data at a speed that's appropriate to a given use case:

Continuous, real-time data feeds. To most people, real time means that changed data is synchronized with sub-second performance. Indeed, many business and technology situations require fast, nonstop data synchronization of this type. For example, in a database high availability solution, real-time or sub-second latency is critical for minimizing or eliminating downtime. And fraud detection scores should be synchronized continuously with payment processing applications to stop fraud as soon as it's detected.

Frequent, intraday data updates. Some business tasks need data updated a few times a day, but not continuously. For example, many manufacturers use operational reports to monitor production yield on the shop floor. The reports are usually refreshed three times a day; hence, data from shop floor applications should be synchronized with the data warehouse three times daily. Likewise, just-in-time inventory management may tolerate data that's a few hours old, so a few intraday data updates can be used instead of continuous data feeds.

Latent batches and queue management. Even when changed data is available to the data sync solution, the target systems may not be available until a specific time. For example, many operational applications or data warehouses are updated at an off-peak time or in a specified batch window. In other cases, the data may not be time sensitive, or it may be more efficient to process data in bulk instead of per record or transaction. For these use cases, the data sync solution can sort change data from its queue into batches and deliver the batches on a scheduled basis.

NUMBER FIVE

APPLY DATA SYNC TO DATABASE HIGH AVAILABILITY CONFIGURATIONS

By definition, a mission-critical application must be available continuously throughout the business day. Making a mission-critical application highly available means (among other things) that its database management system (DBMS) must have a high availability strategy. Various configurations of data replication and synchronization commonly enable database high availability.

Live standby. In a live standby configuration, an application stack includes a DBMS and data set that are considered the primary (or source) database. Yet, there's another instance of the DBMS located elsewhere, called the secondary (or target) database. As data operations are applied to the primary database, these are captured and applied to the secondary one. The point is that the secondary database—or live standby—is a backup that can step in immediately and keep the application running in the event of an unplanned outage of the primary database.

Most live standby configurations rely on unidirectional replication to keep the secondary database synchronized with the primary one. Although we usually think of data sync as bidirectional, a simple live standby configuration supporting database high availability may only involve one-way data movement.

Active-active. Another way to look at the live standby configuration is that the primary database is active (taking database operations directly from the application), whereas the secondary one is passive (receiving application updates indirectly from the primary database). For many users, it's difficult to rationalize a passive database on a cost-to-benefit basis, even when the passive database handles read-only operations for reporting and backup. So, the trend is to deploy an active-active configuration (sometimes called multi-master) where two or more databases are synchronized (bidirectionally and continuously), although each is also actively accepting transactions and other write operations.

Database high availability aside, an active-active database configuration is also useful for integrating multiple ERP instances, synchronizing customer data, or publishing master data. In addition to synchronization, an active-active configuration can load balance database servers to achieve greater scalability and performance. In these use cases, an active-active configuration may include many active databases, which are part of many applications.

NUMBER SIX

APPLY DATA SYNC TO DATABASE MIGRATION AND MAINTENANCE PROJECTS

It's important to keep mission-critical applications and data highly available, even as those systems are migrated to a new platform, consolidated with others, upgraded to a new version, or treated to other maintenance activities. When a database maintenance project demands database high availability or the synchronization of old and new systems during upgrades and migrations, consider using data sync to:

Make a database highly available for zero-downtime maintenance. We all know from experience that planned downtime for database maintenance can easily turn into unplanned downtime, due to human error or an unforeseen data condition. A simple database upgrade, reorganization, or partitioning suddenly turns into a fire drill. When data sync provides high availability for a database, it helps avoid and recover from problems incurred during database maintenance.

Synchronize old and new databases during migration or consolidation. With migrations and consolidations, it's common that the old platform being retired will continue to run simultaneously with the new one while users, applications, and data are moved and tested. Multiple phases are recommended for migrations and consolidations to reduce the risk of a "big-bang project." Bidirectional data sync is also recommended to keep the data of old and newly migrated systems synchronized and to reduce downtime in the switch-over process.

☑ NUMBER SEVEN

RESOLVE CONFLICTING DATA VALUES VIA RULE-BASED SYNCHRONIZATION

The bidirectional or multidirectional nature of data synchronization gives it the added burden of resolving conflicting data values, in some use cases. After all, when applications are writing to all the databases in an active-active or multi-master configuration, it's possible that conflicts will arise in data values or data structures.

Detect, identify, and resolve data conflicts. To cope with these situations as automatically as possible, a data sync solution should include features that detect a conflict, identify the type of conflict, and apply an appropriate resolution. These features are somewhat weak in the replication utilities built into DBMSs, so in general, conflict resolution is best done with a data sync tool from an independent vendor.

Control conflict processing with rules. Detection, identification, and resolution should be controlled through rules defined by the developer. The rules should be flexible, such that they're automatically applied globally or per data structure. And the rules should tap into available information, such as data values and filters or database and application error messages.

Augment conflict resolution with coded approaches. Although it's best to build your rule-driven data sync solution atop a vendor tool, you should expect to augment it with SQL routines, DBMS-specific stored procedures, and custom code. This helps you deal with unique situations or reuse legacy hand coding.

Plan to process exceptions. A data sync solution with good rules for conflict resolution will minimize the number of exceptions. But you still need to provide ways to process exceptions manually or via another software tool.

☑ NUMBER EIGHT

MAINTAIN TRANSACTIONAL INTEGRITY WHILE SYNCHRONIZING DATA

Much of the data handled by a data synchronization solution originates as transactions. More to the point, when transactional data is synchronized across transactional systems, the data integrity of the target system must be assured by syncing data through a transaction processing mechanism. As an industry-wide practice, most transactional applications and databases maintain transactional integrity by following the so-called ACID process:

- **Atomicity.** Commit the whole transaction or none of it.
- **Consistency.** Adhere to the commitment rules of the application and DBMS.
- **Isolation.** Lock data during the intermediate state of a transaction.
- **Durability.** Don't roll back a transaction once it's fully committed.

The ACID properties were developed as guidelines for transactional applications and DBMSs, but they also apply to many data synchronization solutions:

Synchronized financial and healthcare data must be perfect. There is no tolerance for error with transactions that involve money or a patient's health information. The careful commitment of the original transaction must be followed as data is synchronized.

Database high availability demands transactional integrity. Otherwise, when the secondary database jumps in to replace a failed primary database, transactions may be lost or misrepresented. Also, reports and backups populated from the secondary database are inaccurate without transactional and referential integrity.

Customer data changes relentlessly, which is a data integrity challenge. When a new customer is entered into one of many customer-facing applications, the customer record should be synchronized with other customer-facing applications. Likewise, when an existing customer changes, those changes must be captured and propagated. This is a challenge, because some applications allow data to enter at the database level, while others accept only data that comes through application logic. Data sync solutions must accommodate each application's approach while maintaining transaction integrity.

NUMBER NINE

SYNCHRONIZE HETEROGENEOUS DATA SOURCES AND TARGETS

Heterogeneity is the norm in IT because the average enterprise has many different databases, applications, operating systems, legacy platforms, and so on. Connecting diverse IT systems presents a number of challenges for data synchronization:

Diverse database management systems (DBMSs). Most brands of DBMSs support mature and robust replication capabilities. Yet, these are mostly intended to move data among databases managed by the same DBMS brand (e.g., from SQL Server to SQL Server). The DBMSs have become more open in recent years. For example, Oracle's replication supports its own databases, plus SAP ERP (a fairly ubiquitous platform nowadays). Even so, DBMS-based replication is still best applied in a homogeneous or lightly heterogeneous database environment.

Users contemplating replication or data synchronization in a heterogeneous data environment should consider an independent tool that's open to a wide variety of source and target databases. Before committing to a tool, be sure it can access and load a variety of relational, open source, open systems, and legacy databases on a variety of platforms. Furthermore, flat files and indexed files may be involved.

Diverse data sync configurations. IT environments change continuously as new applications are added, and data movement requirements change to support business operations. A data synchronization configuration should be flexible, to add and remove systems easily and to change to (and from) bidirectional data movement, if needed. For example, a new billing system should be added to the list of databases containing customer data, so it gets synchronized for maintaining a single view of the customer. And the synchronization configuration may need to support many-to-many or cascading architectures for effective data sharing. For example, the customer data that is collected from several OLTP systems may be feeding a data mart that is synchronized with a data warehouse.

NUMBER TEN

TRANSFORM AND IMPROVE DATA AS YOU SYNCHRONIZE IT

If you take the word *replication* too literally, you might think that replication and synchronization technologies simply copy data without altering it. This is true for some applications of replication and synchronization, especially when database high availability is the goal. But many other use cases require that data be transformed to meet the needs of target systems.

Diverse data models demand data transformation. Given the diverse databases discussed here, it's inevitable that their data models differ from one another. A data sync solution must deal with this diversity by providing data transformation and mapping capabilities that move data from one data model to another—not just one database to another. For example, data is often replicated and synchronized one record at a time, so you may need to reorder the fields of a record and recalculate some fields to make the record fit the target system. If data sync is many-to-one, merging data (like joining tables) invariably involves some form of data transformation, plus mappings from the source models to the target model.

Diverse data sources demand data standardization. Transformation functions can perform basic data quality tasks, such as standardization and validation. This is useful when you need to transform synchronized data to comply with enterprise data standards. For maximum flexibility, look for data sync tools that can apply data transformations and mappings in a hub or push them into a target database.

Data demands improvement, not just management. The altruistic goal of all data management professionals should be to add value to data and improve it, not just manage it. By nature, data sync adds value by enhancing the complete view of business entities as seen from individual applications. And it increases data's value by making it current and continuously available. These benefits are built into most data sync solutions automatically without much effort from the developer. Yet, developers should go the extra mile and look for other ways to preserve and extend data's value through capabilities associated with data sync, such as data transformation, conflict resolution, and transactional integrity.

NUMBER ELEVEN

TURN TO INDEPENDENT SOFTWARE TOOLS FOR DATA SYNCHRONIZATION

There are good reasons why users planning a data sync solution should consider building it atop an independent or third-party tool:

Advanced uses cases of data sync require an advanced tool. It's true that most DBMSs (and some operating systems and applications) have replication capabilities built in, which can support basic configurations of data sync. But the use cases for data synchronization described in this report demand solutions that require advanced features from a tool to be successful. For this reason alone, data synchronization is best done with a third-party or independent tool. This is especially true for any data sync solution that requires many-to-many data movement, transactional data integrity, rule-based data-value conflict resolution, heterogeneous data sources and targets, and data transformation and mapping.

A data synchronization solution should be like Switzerland. It should be neutral and not unduly biased toward certain brands or types of data sources and targets. Similarly, the norm in data integration (a broad category that includes data sync) is to deploy a single, central solution that's open enough to access and load data for just about any source or target, as well as handle data transformation, mapping, and integrity. A neutral and central solution fosters a number of desirable outcomes, like organized architectures for integration, enterprise data standards, reuse across solutions, and an enterprise view of data as a global asset.

All this helps you come closer to the noblest goal of data synchronization, which is to complete the view of important business entities (like customers, products, and financials) with up-to-date and highly available data.

ABOUT OUR SPONSOR

GoldenGate™

GoldenGate Software Inc. is a leading provider of high availability and real-time data integration solutions for improving the availability, accessibility, and performance of critical data across heterogeneous enterprise IT environments. Its software platform moves transactional data between heterogeneous systems with sub-second latency. More than 500 customers worldwide, including Visa, Bank of America, US Bank, UBS, Sabre Holdings, DIRECTV, Comcast, MGM Mirage, Chase Paymentech, AMD, Mayo Foundation, Retail Decisions, and Overstock.com, rely on GoldenGate solutions. The company broadens its global market reach through strategic relationships with leading technology vendors including ACI Worldwide, Amdocs, Business Objects, Cerner, Fujitsu, GE Healthcare, HP, IBM, Ingres, Microsoft, Oracle, and Teradata.

www.goldengate.com

ABOUT THE AUTHOR

Philip Russom is the senior manager of TDWI Research at The Data Warehousing Institute (TDWI), where he oversees many of TDWI's research-oriented publications, services, and events. He's been an industry analyst at Forrester Research, Giga Information Group, and Hurwitz Group, where he researched, wrote, spoke, and consulted about BI issues. Before that, Russom worked in technical and marketing positions for various database vendors. You can reach him at prussom@tdwi.org.

ABOUT TDWI RESEARCH

TDWI Research provides research and advice for BI professionals worldwide. TDWI Research focuses exclusively on BI/DW issues and teams up with industry practitioners to deliver both broad and deep understanding of the business and technical issues surrounding the deployment of business intelligence and data warehousing solutions. TDWI Research offers reports, commentary, and inquiry services via a worldwide Membership program and provides custom research, benchmarking, and strategic planning services to user and vendor organizations.

ABOUT THE TDWI CHECKLIST REPORT SERIES

TDWI Checklist Reports provide an overview of success factors for a specific project in business intelligence, data warehousing, or a related data management discipline. Companies may use this overview to get organized before beginning a project or to identify goals and areas of improvement for current projects.

tdwi
THE DATA WAREHOUSING INSTITUTE

1201 Monster Road SW
Suite 250
Renton, WA 98057

T 425.277.9126
F 425.687.2842
E info@tdwi.org

www.tdwi.org