Top 5 Considerations When Implementing a Logical Oracle® Database Replication System

A Dbvisit Guide
Getting Your Data Flowing With Real Time Logical Replication

You can get your data flowing by implementing a real-time logical replication solution to replicate data between databases. Replication can solve many data challenges within your organization, and bring a number of business benefits, including:

1. **Offload Reporting.** Providing a solution to create and support dedicated reporting environments. These databases can then be used to cater for all your reporting requirements without any impact to the operational users on your primary system.
2. **Zero downtime migrations.** Use logical replication to upgrade or migrate an Oracle database from one version to the next version with little or no downtime. It can also be used to migrate Oracle from one Operating System to another or from one data center to another.
3. **Data Distribution.** Improve Workflow processing by replicating data to different system to facilitate business processes.
4. **Cloud Integration.** Provide Replication between on-premise systems and the cloud and back.
5. **Data Warehouse.** Capture all the latest changes to feed the Data Warehouse system (CDC – Change Data Capture).

Considerations

As with any IT project there are a number of considerations when implementing new solutions and real-time logical replication solution is no exception. However, these considerations are relatively minimal when compared to implementing other solutions. Five key considerations include:

1. **Data instantiation**

   Seldom do we start with a clean slate with no historical data. Data replication is usually implemented on a running system that already has historical data, but when the replication is started, only the data that has changed after this point will be replicated. So the historical data accumulated before the replication started needs to be loaded into the target database. Careful attention needs to be paid to this to ensure that there are no gaps between the historical data and the point at which the replication began.

   In Oracle this can achieved by using the Oracle SCN number. So the replication can start with a particular SCN number and all data before that SCN number can be loaded into the target database. The Oracle datapump tool, which is available in all Oracle editions, is an excellent candidate to achieve this functionality. It has a “flashback_scn” option that can be used to export and import data prior to a specific SCN. Dbvisit Replicate automatically creates the datapump script with the correct flashback_scn added to ensure there are no gaps between the historical data and the new replicated data.

2. **Primary keys and unique data**

   It is not necessary to have primary keys specified for the tables you are replicating, but at a minimum each row should be uniquely identifiable – otherwise you will run into data conflict issues. So what is needed, in the absence of primary keys, is that the data in each row of the replicated table is effectively a unique entry. This can be done by having a unique index specified on a combination of columns in the table, or if there is no unique index, then all the columns in the table will be
used by Oracle to ensure uniqueness. Note that different replication tools may have different requirements relating to uniquely identifiable data. For an explanation on why the data must be uniquely identifiable please see this article.

3. **Plan conflicts**

With logical based replication, data is being replicated between two separate databases that are both open for normal read-write transactions. This means that other processes not associated with the replication can actually change the target data. If the target data has been altered independently, then there is a chance of a conflict. A conflict is a warning sign that the two datasets (source and target) are, or are going to be, out of sync, so conflicts are to be avoided as much as possible.

However, it is a fact of life that there is always a potential for a conflict to occur with logical replication (even more so in 2-way replication) and so they need to be planned for. Planning for a conflict involves specifying what the rules are when a conflict occurs. Even though the DBA may be the person that is in charge of managing conflicts, it is usually the data administrator or business owner who is responsible for deciding what to do when the conflict occurs because it is their data. A key consideration then, for example, is deciding where the master data will reside: should this be on the source, or on the target? So if an independent update has occurred on the target, will this update be overwritten with the information from the source? In this case the update will actually be lost, and is this desirable?

These are some of the questions that need to be planned for. Most replication tools allow conflict handlers to be set so that conflicts are automatically dealt with. Typically conflict handlers can be set independently for Insert, Update, Delete and DDL statements as well as Oracle errors (such as constraint errors).

4. **Monitor and set the correct thresholds**

Most people prefer that their systems are monitored so that they do not have to remember to check if everything is ok. They also do not want to be inundated with notifications every minute to say everything is still running. We can become immune to these messages and then do not see the important messages anymore. Generally we only like to get notified when things go wrong, or have a potential to go wrong in future.

As with any automated system, replication needs to be monitored to ensure the replication is successful and continues to be successful. Every replication is different, so ensure the right level of monitoring and thresholds are set to get a good balance between not getting too many false positives and being notified when there is an error, or a potential for an issue to occur. Some of the key statistics to monitor in a replication partnership are:

1. Time difference between source and target data. For example, if this is more than 300 seconds then generate an alert.
2. Number of conflicts. If automated conflict handlers are set but there are still conflicts occurring that are not caught by the automatic handlers then this can be used to notify.
3. Oracle SCN difference between source and target data. If the Oracle SCN difference is greater than say 1000, then notify.
If the notification only sends out emails when things go wrong, it is also important to send out daily heartbeat notifications to say that the replication and monitoring is still working successfully. This notification should come at a specified time everyday so that you can take action if it does not come.

5. **Not all data needs to be replicated**

Depending on the use case of the replication, it may not be necessary to replicate all data. For example, when replicating to keep a reporting database up to date, so that report processing can be offloaded from the primary database, it may not be necessary to replicate all columns in the table. This is especially so if there are BLOB or CLOB columns, as these are probably not going to be needed in reporting so they do not have to be included in the replication. This makes the replication more efficient. Most replication solutions including [Dbvisit Replicate](https://www.dbvisit.com/software/dbvisit-replicate) have the option to select only a subset of data to be replicated.
About the author

Arjen Visser is a technical entrepreneur with over 20 years of experience in the IT industry and is the founder, CEO and CTO of Dbvisit Software Limited. He is most passionate about database technology and especially Oracle database technology.

He is driven to create complementary software solutions that are highly effective, simple to use, and in great demand. He is a well-known figure in the world of Oracle database experts and a frequent speaker at Oracle conferences.

Arjen Visser
Founder, CEO and CTO

Dbvisit Software Limited

www.dbvisit.com
info@dbvisit.com
USA: 1-800-933-8007
Rest of the world: +64 9 950 3301
Fax: +64 9 950 3302

Dbvisit Software Limited
P.O. Box 48180, Blockhouse Bay
Auckland 0604, New Zealand

Copyright © 2013, Dbvisit Software Limited.. All Rights Reserved.