1 Executive Summary

Informatica MDM Hub supports a number of different approaches to providing a highly available solution. Our use of the term "highly available" refers to the degree Informatica MDM Hub can continue to operate without any need for downtime.

In most circumstances, availability is measured with respect to unplanned downtime – that is, downtime that occurs as the result of a failure or set of failures in the Hub or its supporting infrastructure. However, Informatica MDM Hub supports the concept of continuous operation, even in the event of planned downtime.

This document summarizes Informatica’s approach to availability.

2 Informatica MDM High Availability Solutions

Informatica MDM supports several approaches to build operational deployments of the Informatica MDM Hub that satisfy the business availability requirements. These approaches trade availability against the associated implementation complexity and related costs.

There are three basic approaches supported by Informatica MDM:

- Non-Distributed High Availability
- Distributed High Availability
- Zero Downtime

3 Non-Distributed High Availability

Non-Distributed High Availability supports continuous operations in the event of failure of any single component in the solution, but does not handle system wide failure such as the failure of the LAN on which the Hub operates. The solution is shown below:
In this solution, redundant enterprise class load balancers are used both to load balance between the application servers and also to handle failover. The application server tier in this case is implemented as an application server farm or as a full J2EE cluster, and the database tier is implemented in a clustered environment, such as VERITAS DB cluster or Oracle RAC. Note that this architecture also assumes redundant network connections and redundant SAN connections, and high availability implemented within the SAN.

Failures at the application server and database tiers are handled by failover to another node in the associated cluster. Network connectivity issues are handled at the load balancers, and SAN failures (both SAN node failures and SAN connectivity failures) are handled within the SAN.

Because this configuration resides within a single system context, system-wide failures (e.g., power failures, network domain controller failures, etc.) are not in scope and may still result in system outage.

This approach to High Availability is by far the most prevalent amongst our MDM customers. It provides a balanced approach to risk versus cost. For example, a Pharmaceutical MDM customer recently implemented this approach to MDM after first insisting that Distributed High Availability (see below) was required. However, the cost difference between the two approaches caused this customer to evaluate the actual requirements. In doing so, they realized that the facility in which MDM was hosted had not failed in over 10 years, so the risk of such a failure was considered to be minor and the customer decided to implement a Non-Distributed High Availability solution.

4 Distributed High Availability

Distributed High Availability supports continuous operations in the event of failure of any single component in the solution, including system wide failures. The solution is shown below:
In this solution, redundant enterprise class load balancers are used both to load balance between two independent application stacks as well as between the application servers in an individual application stack. The picture above shows an Active-Passive configuration, but Active-Active configurations may also be employed.

Within each application stack, the application server tier is implemented as an application server farm or as a full J2EE cluster, and the database tier is implemented in a clustered environment, such as VERITAS DB cluster or Oracle RAC. Note that this architecture also assumes redundant network connections and redundant SAN connections, and high availability implemented within the SAN.

Database replication technology is used to replicate data between the two application stacks. In an active passive configuration, this may be done using one way database replication. However, if one-way replication is used, no fail-back capability is required. If Active-Active configuration is chosen, the database replication must be two-way.

Failures at the application server and database tiers are handled by failover to another node in the associated cluster. Network connectivity issues are handled at the load balancers, and SAN failures (both SAN node failures and SAN connectivity failures) are handled within the SAN.

At the system level, system-wide failures are addressed by redirecting all requests to the passive environment or, in an Active-Active configuration, to the active environment which has not experienced a failure.

This approach to High Availability is used by customers that cannot deal with any unscheduled downtime—typically companies that deal directly with consumers and use the MDM Hub as an integral part of that interaction. This approach is significantly more expensive than the Non-Distributed High Availability but allows continuous interaction with consumers. For example, an Electronic Media MDM customer recently implemented this approach to MDM in order to ensure that consumers (as well as internal entities such as the call center) would be able to interact with the MDM Hub unless there was scheduled and pre-
announced downtime. It was deemed that failure of the system would frustrate consumers who may never return, and the revenue risk from this was determined to be higher than cost of the facility redundancy required supporting this. The Hub was configured as Active-Passive, as described above, but the planned architecture allows the system to become Active-Active in case there is a significant up-tick in activity.

5 Geographic Distribution

Geographic distribution supports locating application stacks in physically distinct locations in order to achieve geographic load balancing or in order to handle the failure of a facility (for example, due to extraordinary weather conditions such as a flood). This is provided by using the Distributed High Availability solution described in Section 4, but locating each distinct application stack in a physically separate location:

6 Zero Downtime

Zero downtime supports continuous operations of the Hub during planned downtime activities, such as the upgrade or modification of the Hub infrastructure, including hardware, software and/or data models. The requirement a zero downtime solution is to ensure that there is no disruption of services for the end users of the applications built on top of the Informatica MDM Hub.
In this solution, we start with the Distributed High Availability solution described in section 4. However, in this case a specific type of database replication is applied; namely, real-time, logical replication technology instead of the more common log replication. This choice is necessary to deal with the volume of data required to be synchronized during a system upgrade, along with the need to introduce some special processing during the upgrade to support real-time, heterogeneous (i.e. source and target DBs need not be the same format) data replication.

A typical Zero Downtime upgrade has five phases during a Hub Release (any Hub upgrade activities such as changes to the infrastructure, changes to the database structure, or changes to the Hub data):

- **Pre Release:** The hub is at steady state, with the active environment being replicated into the passive environment.
- **Apply release to passive environment.** Replication is halted to passive environment (all actions on active environment are captured and queued). The release is applied to the passive environment. The passive environment is brought into synchronization with the active environment.
- **Swing between environments.** The formerly passive environment is made active and the formerly active environment is made passive.
- **Apply release to formerly active environment.**
- **Post Release:** The hub is again at steady state, with the active environment being replicated into the passive environment.

Zero downtime is a very expensive proposition. Primary drivers in determining whether it is required are

1. Is the system truly mission-critical such that it cannot be taken offline for a few hours/year?
2. Are all associated components (database, source systems, consuming system) also configured for zero downtime?
3. Is the risk of loss during the scheduled outage so severe that it warrants the costs associated with solution?

The vast majority of MDM customers find the answer to one or more of these to be in the negative. However, some companies, and in particular financial institutions, tend to have large number of high value transactions 24x7x365 thus justifying this solution. One financial institution that implemented the Zero Downtime architecture determined that they would lose between $100,000 and $1,000,000 each time an
eight hour planned outage occurred, so the expense associated with this solution was significantly less than the risk associated with not having the solution.

7 Summary

As discussed in this document, Informatica MDM Hub supports three different approaches to providing a highly available solution. It is important to note that not all applications require high availability – many require a more simple resilience model. While High Availability aims to continue operations in the event of a single failure, resilience is a process used to recover in case of a failure.

For example, if a resilient web services framework loses the connection, it attempts to reconnect for the amount of time configured for the retry period. If no connection is established in that period of time, the framework fails. By comparison, a High Availability solution will provide a failover capability and a redundant network connection, and in the event of a single failure, will automatically failover to the redundant connection and then retry jobs that were in flight when the failure occurred, thus incurring no downtime nor potential loss of operation. However, a resilient solution is much less expensive than a HA redundant solution, so proper trade-off of cost versus benefit should be performed before embarking on a High Availability solution.

Should it be determined that High Availability is appropriate for a given situation, MDM provides three different models. The following describes those models and some high level thoughts on how to choose the right HA model for a given implementation.

1. **Non-distributed High Availability**: This is the least expensive of the high availability solutions and provides continuous operation of the MDM hub in the event of unscheduled downtime (failure) of any single component in the MDM system. However, system-wide failures (for example, loss of facility power, loss of LAN, etc.) are not handled in this architecture. In general, the intent of Non-Distributed High Availability is to eliminate unscheduled downtime for the vast majority of situations, but accept the risk that a facility failure will cause a failure of the MDM solution. For most applications, Non-distributed High Availability is sufficient to meet the business needs.

2. **Distributed High Availability**: As discussed above, Non-distributed High Availability is sufficient to meet the business needs for most applications. However, if the system cannot accept any unscheduled downtime, then a solution that provided facility redundancy is important. Distributed High Availability is an expensive proposition, because it requires at least one replication of a Non-Distributed High Availability instance. As such, Distributed high Availability systems are generally two or three times more expensive (on a recurring basis) than the Non-Distributed approach. Therefore, the risks associated with a facility failure and the likelihood of such a failure needs to be balanced against the cost of such an infrastructure.

3. **Zero Downtime**: The first two solutions describe solutions for unscheduled downtime. However, some applications need to continuously operate, even during scheduled maintenance periods for the software. Zero Downtime is an extremely expensive proposition which could cost 10 to 20 times the cost of Distributed High Availability so it should be reserved for mission critical applications where continuous operation is an absolute imperative.

Ultimately, which solution gets chosen is the result of a trade-off between the benefits of the solution and the costs. Since the cost implications are quite large, plan to spend sufficient time identifying and challenging the requirements to ensure that an expensive solution is not deployed where a less expensive solution might adequately address the requirements.