

Inside Citrix chapter nine – The one with the Central Site database

The XenDesktop Central Site Database holds all Site wide static (policies, configured Catalogs and Delivery Groups, Host Connections, Zones and so on) as well as dynamic (run-time) information (who is logged on to which VDA, what resources are currently in use etc.) needed during the user logon, authentication and resource enumeration process as well as the actual resource launch sequence (load balancing).

Needless to say, it is an important part of your infrastructure: when it is down your users won't be able to connect and/or launch resources and IT will not be able to make any configuration changes to the Site itself. Because of this you'll probably want to implement some kind of high-availability mechanism keeping your database up and running at all times, or at least to try and keep downtime to a minimum. Of course, Connection Leasing, which we will discuss in more detail shortly, also helps in keeping certain resources available even when the Central Site database is offline. *2018 update: I've added a link to more detailed information on the LHC as well.*

First things first

Especially coming from XenApp 6.5 and earlier versions, these types of changes took some time to get used to. Unlike the 6.5 XenApp servers (Data Collectors), the XenDesktop 7.x Virtual Delivery Agents only communicate with the Site Delivery Controller(s) and do not need to access the Site database directly. Having said that, XenApp 6.5 Session Host only servers offer the same sort of benefit. Because they only host user sessions and will (or can) never be 'elected' as a Data Collector for their Zone, hence the 'Session Host only' part, they won't get all the IMA store (database) information pushed into their LHC (Local Host Cache), enhancing overall performance – Will be discussed in more detail later on.

While Delivery Controllers are comparable to XenApp Data Collectors, there are some distinct differences. Sure, they both handle user authentication, are involved during the resource enumeration process, and are in control of load balancing, for example, but in very different ways.

Data Collectors are part of Zones with each Zone having its own (there can be only one per Zone) Data Collector. Having multiple Data Collectors in your Farm basically means having multiple Zones. Also, Data Collectors need to be able to communicate with each other. With Delivery Controllers this works differently.

Note: Make sure to have a look at the comparison table, comparing the Delivery Controller to the Data Collector as well.

Although, as of XenDesktop 7.7, we are able to create Zones (again), when compared to the Zones in the IMA there are still some architectural differences between the two. To name a few, Delivery Controllers do not communicate with each other, Session Hosts (VDAs) do not communicate directly with the Central Site database, and of course we now have Connection Leasing instead of the LHC, although it was never meant as replacement.

Also, if you look at the characteristics of a XenDesktop 7.8 Zone, at least today, you will notice that you will also have to place and configure at least one Delivery Controller and StoreFront server per Zone, but more on this later on. In addition to the ‘Zones’ section on page 145, here are some of the main differences between Zones in the IMA and the FMA, see next page.

IMA Zones vs. FMA Zones

Zones in IMA (6.5)	Zones in FMA (7.x)
Data Collector	Delivery Controller
Keeping resources and users close together, offering load balancing and HA when needed	Simplified administration of multiple locations and keeping users close to their resources
Max one primary DC per Zone	Unlimited DCs per Zone. Minimum of one is required
Secondary (backup) DCs configured. Election takes place when main Data Collector goes down	All DCs within the same Zone are active
LHC on each DC. Speeds up application enumerations and keeps Farm fully functional when the database is down	Connection Leasing. Enables the launch of certain (assigned) resources when the database is down
DC contacts the IMA database every 30 minutes by default	DC has a direct and continues connection with the FMA database
DC communicates and updates the Session Hosts within its Zone	DC communicates with VDAs in its Zone but primarily for brokering purposes
Load balancing gets applied based on information from the LHC	Load balancing gets applied based on information from DB
DCs exchange (Zone related) information	DCs do not communicate with each other
Zone based policies: Zone Preference and Load Balancing	No specific Zone based policies
No registration	VDAs register themselves

Max of 5 Zones (best practice)	Max of 10 Zones (best practice)
Zones for XenApp only	Zones for both XenApp and XenDesktop

Local Host Cache

In a XenApp 6.5 Farm a Data Collector, including all other XenApp servers with the exception of Session Host only servers, have something called a Local Host Cache (LHC) in which a copy of the central IMA database is cached. And while it primarily functions as a fallback configuration for whenever the IMA database becomes unavailable, it also helps speed up the user authentication and application enumeration process.

Data Collectors also hold and collect dynamic live runtime data used for making load-balance decisions, which it will store in its LHC as well. Delivery Controllers, on the other hand, do not have an LHC, so if they need to authenticate a user or enumerate resources, for example, they will (always) need to contact the Central Site database to get this information. The same goes for load-balancing (dynamic runtime) information: it doesn't get stored locally.

If you have multiple Controllers configured within your Site, but on different physical (geographically separated) locations, keep in mind that they will all need to communicate with the same central database when a user logs in, starts an application or to get load balance information etc., even with one or multiple Zones configured. Consider your bandwidth requirements and make sure to involve the networking team (s) during the design as well as implementation phases.

Also, in a XenApp environment all servers, including the Data Collectors, will contact the IMA database (often referred to as the IMA store) every 30 minutes to update their LHC (except for Session Host only servers, which need to be configured explicitly). In XenDesktop there's no need for this since the Delivery Controllers don't have an LHC and get all their information directly from the Central Site database, live runtime data included. Although the above differences do raise some questions as far as Farm vs. Site designs go, XenDesktop has been doing it this way for a few years and the FMA has proven to be very flexible and robust in many ways.

Minimizing downtime

Now that we've established the importance of the Site database and the need to keep it online as close to 100% as possible, let's see what options we have in accomplishing this. To start, it is (highly) recommended to back up your database on a regular (daily) basis so it can be restored if necessary when the database server fails (or the database itself). In addition, there are several high-availability solutions to consider.

When using SQL Server 2012 (Enterprise Edition) or later, Citrix recommends AlwaysOn Availability Groups: a high-availability and disaster recovery solution first introduced in SQL Server 2012 to enable you to maximize availability for one or more user databases. AlwaysOn Availability Groups require that the SQL Server instances reside on Windows Server Failover Clustering (WSFC) nodes.

The following SQL Server versions are supported:

- SQL Server 2014, Express, Standard, and Enterprise Editions.
- SQL Server 2012 SP1 and SP2, Express, Standard, and Enterprise Editions. By default, SQL Server 2012 SP2 Express is installed when installing the Controller if an existing supported SQL Server installation is not detected.
- SQL Server 2008 R2 SP2, Express, Standard, Enterprise, and Datacenter Editions.

When SQL AlwaysOn HA isn't possible, or optional Citrix recommends implementing SQL mirroring, with SQL Clustering coming in second. Mirroring the Database ensures that, should you lose the active database server, the automatic failover process happens in a matter of seconds, so that users are generally unaffected. Be aware that full SQL Server licenses are required on each database server, but this goes for some of the other solutions as well. Note that SQL Server Express edition can't be mirrored, and clustering isn't supported as well.

Microsoft's SQL Clustering technology can be used to automatically allow one server to take over the tasks and responsibilities of another server that has failed. However, setting up a clustered solution is more complicated, and the automatic failover process is typically slower than with alternatives such as SQL Mirroring.

Again, as mentioned, SQL Server Express doesn't support clustering, at least not out of the box, but with a little creativity, it is possible: Google is your friend (although not recommended for production environments).

Using one of the above methods, in combination with regular (daily) backups, will ensure that your database will (almost) always be online. If anything, it will at least narrow your chances of running into any issues. Make sure to talk to your SQL DBA when needed and spend some time on putting together a SQL Maintenance Plan. SQL Maintenance Plans help you put together various workflows required to make sure that your database is optimized, regularly backed up, and free of inconsistencies.

What about some of the alternatives?

1. VMware HA. Although VMware HA is a great feature in itself, it has no clue with regard to what's running on the VM it's protecting. I mean, your VM will stay online without too much trouble, even if one of the underlying physical hosts goes down, but HA doesn't know when your SQL database goes down or stops responding due to updates gone bad, drives filled up, services stopped or something similar. You could say that VMware HA, and I guess the same goes for Hyper-V or XenServer as well, simply is not application-oriented.

2. VMware App HA. As part of the Enterprise Plus edition, VMware offers Application High Availability. App HA is more intelligent; it can restart failed application components or use the Application Awareness API through VMware HA to reset the VM if needed (no OS reboot). VMware App HA provides support for SQL, Tomcat, Apache, TC Server and IIS. For now, no other ‘big’ applications, like Exchange, SharePoint, ShareFile and Oracle are supported. But since it’s SQL we’re interested in, it could work. VMware App HA is relatively complex to set up: it requires multiple VMs and points of management. Also, as an addition to the applications already mentioned, no custom or generic applications are supported, and as far as I could tell it’s only supported on vSphere 5.5. On the other hand, if you are already using VMware combined with Enterprise Plus licenses you’ll have not only App HA, but also a whole bunch of other cool features and technologies as well. And if it’s SQL you’re supporting, then this is probably the most logical and cost-effective step to take, especially for smaller and mid-sized companies. However, when using ‘lighter’ editions of VMware or when real enterprise functionality is needed (more on this in a bit), then this probably won’t be your first choice.
3. VMware Fault Tolerance. One of my favorites. As of vCenter Server 6.0 (and vSphere 6.0) a fault-tolerant machine can have up to 4 vCPUs (this used to be only one). This comes from the VMware website: Fault Tolerance (FT) provides continuous availability for applications in the event of server failures by creating a live shadow instance of a virtual machine that is always up to date with the primary virtual machine. In the event of a hardware outage, vSphere FT automatically triggers failover, ensuring zero downtime and preventing data loss. After failover, vSphere FT automatically creates a new, secondary virtual machine to deliver continuous protection for the application.
4. Symantec ApplicationHA. Although not known by many, this is probably the most Enterprise product of them all. It supports all major applications, including custom and/or generic applications, over 23 tier 1 applications in total. It’s based on Symantec Cluster Server, powered by Veritas agents to monitor and control applications. It can restart failed application components, restart the underlying Operating System, or restart the VM itself. And if all else fails, it can use the last known good backup for restore purposes. The Symantec HA console communicates with VMware through a vCenter plugin providing centralized management for all protected VMs. Of course, Symantec App HA would be a separate purchase, and if you already have VMware running you will probably need a strong business case to get the funds. If it’s more than SQL that needs to be supported, if you’re going ‘Greenfield’ or you may have a specific need for one, or multiple, of the Enterprise class features that the Symantec product has to offer, like the number of supported applications or VMs / agents (think big, one of my former colleagues always used to say), then Symantec Application High Availability might just fit the bill.

It is not just the Central Site database

Besides the Central Site database XenDesktop also has a Configuration-Logging database and a Monitoring database. The Configuration-Logging database stores information about all Site configuration changes taking place, including other administrative activities. This database is only used when the Configuration-Logging feature is enabled, which it is by default.

The Monitoring database stores all information used by Director, like session and connection information. Both databases will be discussed in more detail.

Database sizing numbers

When properly sizing your FMA Central Site database you need to keep your eye on two files in particular: the database file itself, which will contain all Site information, the data and objects like stored procedures, tables, views and so on, and the so-called Transaction-Log file. The later contains a record of all transactions, including any database modifications that might have been made by a transaction. If there is a system failure and the current live Site database becomes corrupt and/or unusable in any other way, the Transaction Log can be used (replayed) to re-create the database and bring it back to a consistent state. However, this does depend on how you configure the Transaction-Log to handle data. You have the following options:

- Simple Recovery mode: In this mode no log backups are required, meaning that no transaction log data will be saved. If the database fails, all changes made to the database since the last full back up must be redone.
- Full Recovery mode: This mode does require backup logs. If the database fails, no work is lost. All data or any specific point in time can be recovered. Full Recovery mode is needed for database mirroring.
- Bulk-Logged Recovery mode: This model is an adjunct of the full recovery model that permits high-performance bulk copy operations. It is typically not used for Citrix databases.

If you back-up your Site database on a daily basis, or at least multiple times per week, simple recovery mode will probably be sufficient. However, it depends. If Site configuration changes are constant, multiple times per week or daily even, then Full Recovery mode might be desirable. Always make sure that high performance storage is used for your SQL infrastructure, SSDs preferably.

Site database

Typically, the size of your Central Site database will depend on multiple factors. Since the information stored is both static as well as dynamic, its size can vary during the day. The following factors need to be taken into consideration: the number of configured and registered VDAs, connected and active sessions, the number of transactions taking place during logon and logoff, general logon and logoff behavior, the physical size of your Site, and a few more. To give you a ballpark indication of its possible size, see the table below. Numbers are in MB.

Site database sizing numbers

Users	Apps	Type	Max size
1000	50	RDSH	30
10,000	100	RDSH	60
100,000	200	RDSH	330
1000	N/A	VDI	30
10,000	N/A	VDI	115
40,000	N/A	VDI	390

Note that real-world values may vary. These numbers are for indication purposes only and belong to Citrix Consulting Services.

These same factors also play a role in determining the size of your Transaction Log. Just make sure to regularly back up your Transaction Logs (which will shrink them) by configuring and scheduling SQL maintenance plans on an ongoing basis.

Monitoring database

This one is expected to grow the largest. It contains historical information and, as such, its size depends on multiple factors, which will include but not be limited to: The number of Session Hosts within your Site, including the number of total sessions and connections, and even more specifically, how long data will be stored, which will also depend on the type of license used.

FMA fact: Non-Platinum-licensed customers can keep and store data for up to 7 days, while a Platinum license allows you to store all data for up to a year, with the default being 90 days.

The following table should give you an indication of how the Monitoring database might grow. Again, these numbers are based on Citrix Consulting Services scale testing, assuming a 5-day working week and one connection and 1 session per user. Numbers are in MB. See next page.

Monitoring database sizing numbers

Users	Type	1 week	1 month	3 months
1000	RDSH	20	70	230
10,000	RDSH	160	600	1950
100,000	RDSH	1500	5900	19000
1000	VDI	15	55	670
10,000	VDI	120	440	5500
40,000	VDI	464	1700	21500

To help you size your databases Citrix released the Database Sizing Tool for XenDesktop 7.x. After filling in certain information like, the total number of users, sessions, applications, the type of deployment, RDSH and/or VDI etc. it will produce an overview of database sizes at various points in time. This will include day 0, 1, the first week, months, quarter and year. Have a look at the following CTX article for more detailed information: CTX139508.

Configuration Logging database

Overall (without MCS, see below) this one will be relatively small compared to the others. Its size will depend on the number of administrative tasks taking place on a daily basis using tools like Studio, Director and/or PowerShell scripting.

Another important factor impacting the Configuration Logging database size is the use of MCS for provisioning desktops and servers. Without MCS a typical Configuration Logging database tends to be somewhere around 30 to 40 MB in size, while with MCS, depending on the number of machines provisioned, it can easily grow beyond a few hundred MBs.

Connection Leasing

Although assumed by some, Connection Leasing was never meant as a replacement for the Local Host Cache. The Connection Leasing feature (as of XenDesktop 7.6) supplements the SQL Server high-availability best practices highlighted earlier, enabling users to connect and reconnect to some of their most recently used resources, even when the Central Site database is unavailable.

FMA fact: Connection Leasing is meant to supplement SQL High Availability set-ups.

Whenever a user successfully launches a resource, the Delivery Controller will collect information specific to that connection (icons, enumeration and launch information etc.) and will send this to the Central Site database where it will be stored first. Then, in ten-second intervals, this information will be synchronized to all Delivery Controllers known within the same Site where it will be saved on their local hard drives in an XML file format. This is what we refer to as the actual lease information.

Again, I would like to highlight that each Controller has a direct connection to the database; Delivery Controllers do not communicate among each other.

Since the number of files can be significant, I'll explain why in a minute, the creation of these files is done in timed batches to distribute the required disk I/O, which will, or might, be needed during creation. By default, each controller will synchronize at a rate of up to 1000 leases every 10 seconds until all leases have been synchronized, although this behavior can be changed through Registry or GPO.

On average, 2 to 4 GB of additional disk space should be sufficient, depending on the number of users, published resources and/or connections. While for smaller and mid-sized companies the impact of Connection Leasing will probably be minimal, it doesn't hurt to check on the estimated resources needed with regard to CPU and IOPS or disk activity. Citrix has written an

extensive article (white paper) on this particular subject, including multiple calculations and examples with regard to the number and total size of lease files generated under certain circumstances, including the expected load on the underlying storage subsystem and Delivery Controllers.

The Connection Leasing white paper can be found here:

http://support.citrix.com/content/dam/supportWS/kA460000000CktXCAS/Connection_Leasing_XenDesktop_7.6.pdf

For each successful lease connection, multiple folders and XML files will be generated and stored within the Central Site database and on each individual Delivery Controller. Lease files are relatively small in size, around 1 KB and, by default, are stored in subdirectories of:

%programdata%\Citrix\Broker\Cache.

- Apps.
- Desktops.
- Icons.
- Leases\Enumeration.
- Leases\Launch.
- Workers.

These folders combined will store various (XML) files whenever a resource is successfully launched.

- The Apps directory will contain a single file per application per Delivery Group.
- The Desktop folder contains an entry for every user-assigned VDA, meaning one for every user assigned to a VDI-based VDA and one for every RDSH-based VDA (not per user). So in the case of a VDI-based architecture, this folder will hold a lot more files when compared to an RDSH deployment, where multiple users share a single VDA.
- The Icons directory holds a file for every unique published resource, one for each application, and one for all desktops to share.
- The Leases\Enumeration folder contains one file per user, which lists all available resources on a per user basis.
- The Leases\Launch directory contains an entry for each successfully launched resource. One for each desktop a user is entitled to launch, and one shared by all applications available to that same user. Here it is assumed that sessions sharing will direct the user to the same host. These launch files contain specific information on the resource launched, including the machine the resource was started on.
- Finally, the Worker folder contains an entry per VDA, which does not have to be assigned, so one for every VDI-based VDA and one for every RDSH-based VDA.

All this takes place during normal operations where the Central Site database is up and running and everything behaves as expected. As soon as a database failure takes place and none of the active Delivery Controllers is able to communicate with the Central database, and only then will Connection Leasing become active.

When Connection Leasing kicks in, by default there will be a two-minute period where no connections will be brokered; after that, the locally cached leases will become active. At that same time any registered VDAs will start to deregister, and again register once the database comes back online and is reachable.

Once Connection Leasing is active it will use the information stored within the various XML files to present resources to users. More specifically, it will use the information available within the Launch files to actually launch resources. It does this by contacting the machine where the resource was last successfully started, as highlighted earlier.

While all this may sound perfect on paper, it is (very) important to note that all the information stored as part of the Connection Leasing process (the generated XML files stored locally on each Delivery Controller) will be valid / usable for a maximum period of two weeks by default, something which is also configurable through either the Registry or GPO. When configured through GPO the configuration details will be stored in:

HKLM\Software\Policies\Citrix\DesktopServer\ConnectionLeasing

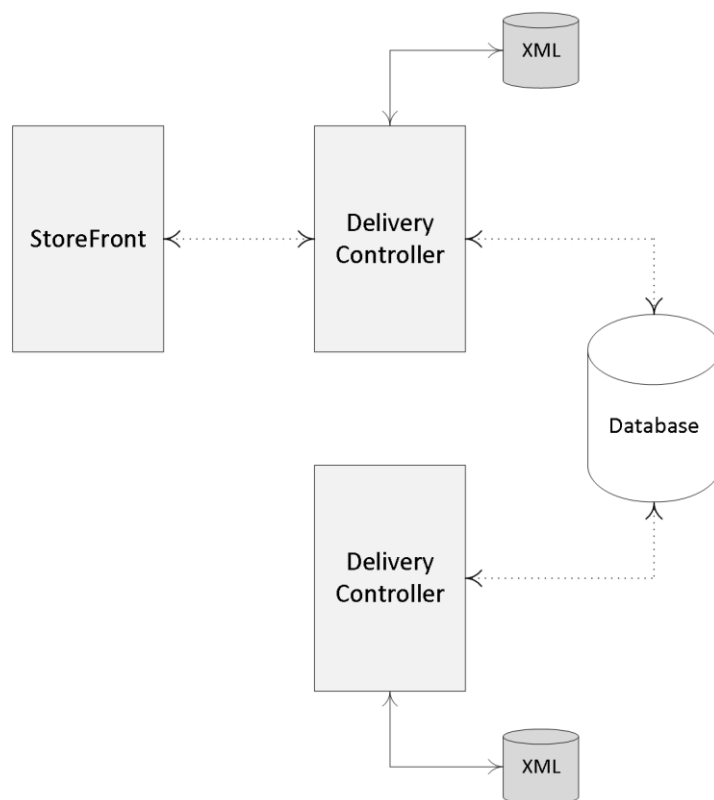
When editing the Registry directly it will need to be done at:

HKLM\Software\Citrix\DesktopServer\ConnectionLeasing

This means that, if we go with the default of two weeks at the time Connection Leasing becomes active, resources that have not been successfully launched during those last two weeks will not be available. Also, if a Launch file contains a machine, which might be down for maintenance, shutdown completely, or no longer operational, your users will not be able to launch that particular resource.

Another important point to mention is that Connection Leasing only works for assigned resources; it does not work for pooled desktops. Below is an overview of the limitations while Connection Leasing is active:

- Desktop Studio and Desktop Director operations are unavailable.
- Citrix PowerShell cmdlets requiring database access will not work.
- No VDA load balancing will occur.
- Users can only connect to the last host they connected to when the site database was available.
- There is a small window (2 minutes) during which no sessions will be brokered when the site database becomes unavailable or is restored. This is to allow for environments with SQL HA enabled to fail over.
- Users must have logged onto the resources within the default 14-day period. This can be configured via a registry setting or GPO.
- Connection Leasing does not support anonymous users.



Connection Leasing overview

2018 update: The Local Host Cache has been renewed and re-introduced – as off version 7.12. It is now also part of the FlexCast Management Architecture. Go [here](#) to read more about how it works in detail.

Key takeaways

- As of XenDesktop 7.x, only Microsoft SQL is supported for the Central Site database.
- It contains all static as well as dynamic Site-wide information.
- Make sure you understand the differences between the IMA and FMA when it comes to your Controllers and the Central Site database.
- If your Site is spread over multiple geographically separated locations, or you have multiple Zones configured, your Central Site database should always be in the Primary Zone, or the main datacenter.
- Even with multiple Zones configured it is still one central database.
- Make sure to implement some form of HA solution for your Site databases, since Connection Leasing is only meant as a supplement.
- Your Delivery Controllers and the Site database are constantly communicating: for this, Windows authentication is required between the Delivery Controller and the database.
- When the database fails (even without Connection Leasing) existing connections will continue to work. New sessions cannot be established and Site-wide configuration changes are also not possible.
- It is not recommended to install SQL on the same machine as a Delivery Controller.
- Try to keep your database server physically close to your Delivery Controllers in the data center.
- SQL software, server or Express, must be installed and configured before creating a XenDesktop Site after its initial installation.
- While SQL Express is primarily used for PoC and testing purposes it could be used for smaller production environments as well. No HA capabilities though. It's up to you.
- You must be a local administrator and a domain user to create and initialize the databases (or change the database location).
- To be able to create an empty database, to add Delivery Controllers to it, create and apply schema updates and so on, you will need to have the following server and database roles: dbcreator, securityadmin and db_owner.
- When using Citrix Studio to perform these operations, the user account must be a member of the sysadmin server role.
- As of version 7.12 of XenApp and XenDesktop the Local Host Cache (LHC) got renewed and re-introduced.