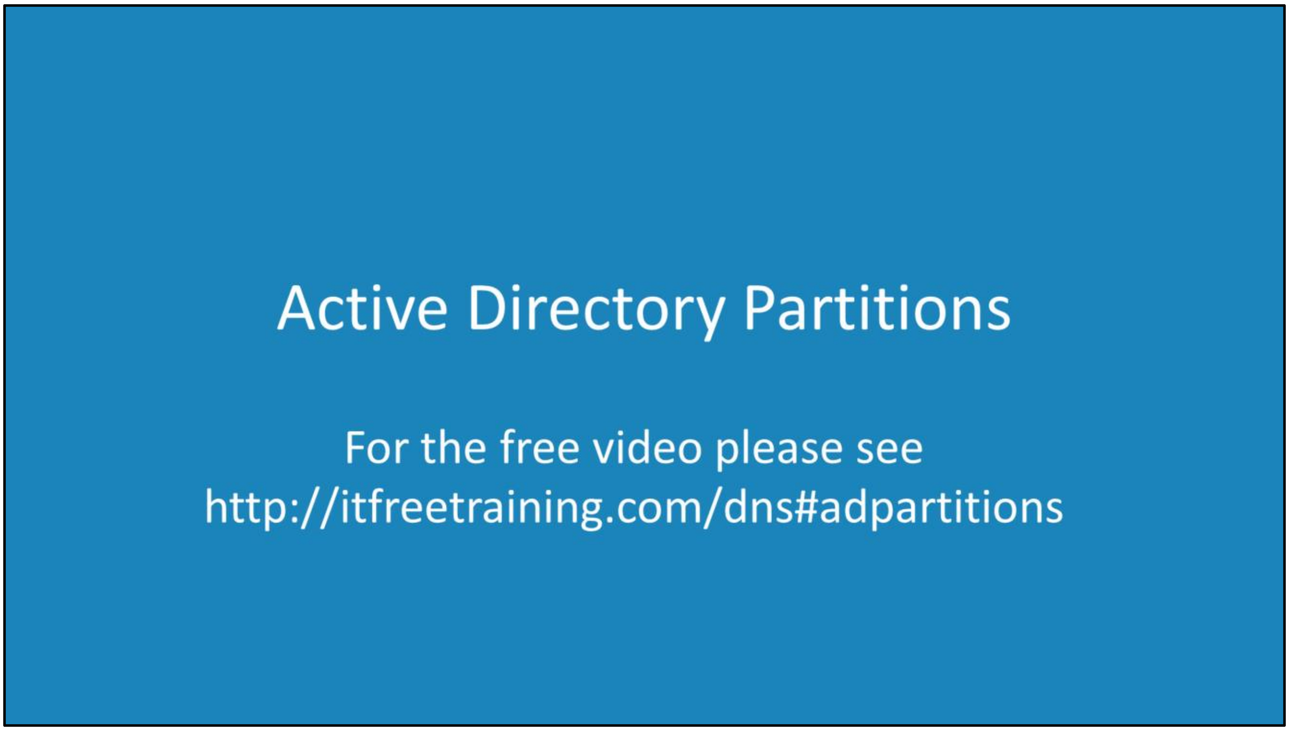
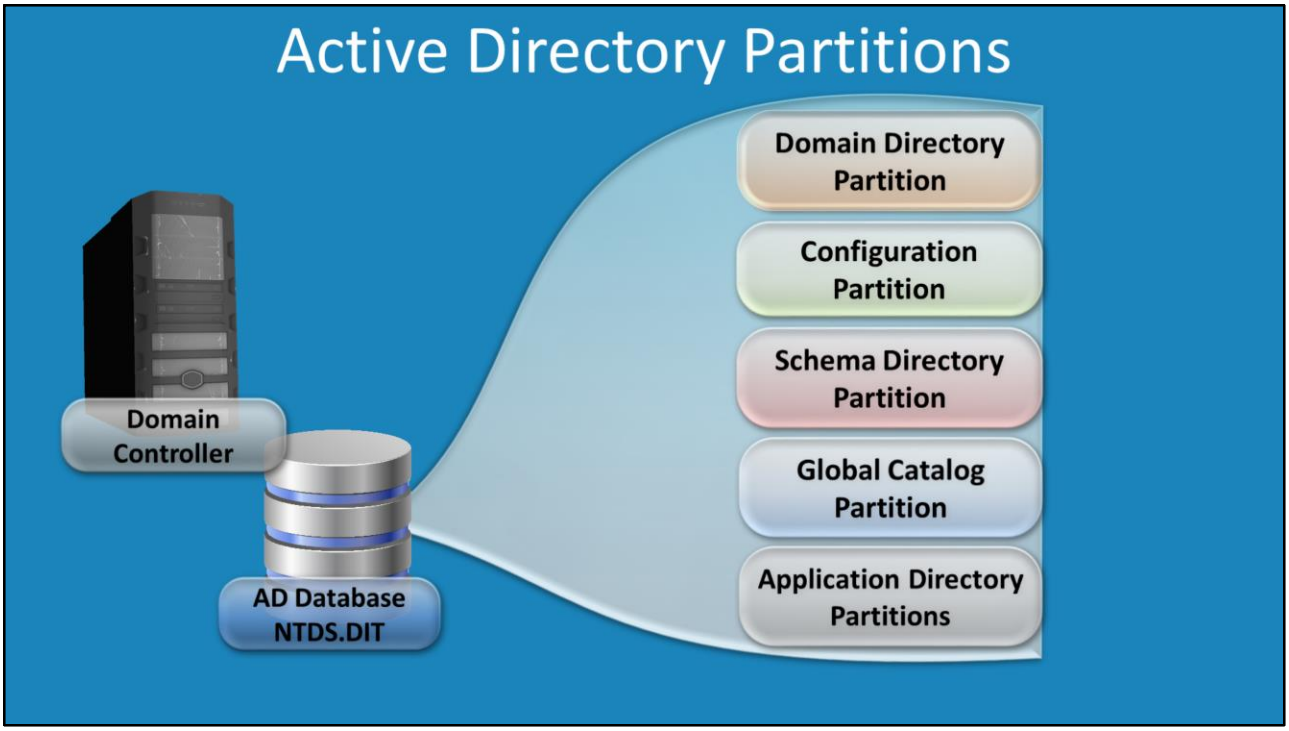


Active Directory Partitions

For the free video please see
http://itfreetraining.com/dns#adpartitions

This video looks at how DNS data is stored in Active Directory Integrated zones and how it is replicated about the domain or forest. Once you have finished watching this video you will understand how this DNS data is stored in Active Directory and how you can configure the replication of this data at the domain and forest level.



Active Directory Partitions

Domain Controller

AD Database
NTDS.DIT

Domain Directory Partition

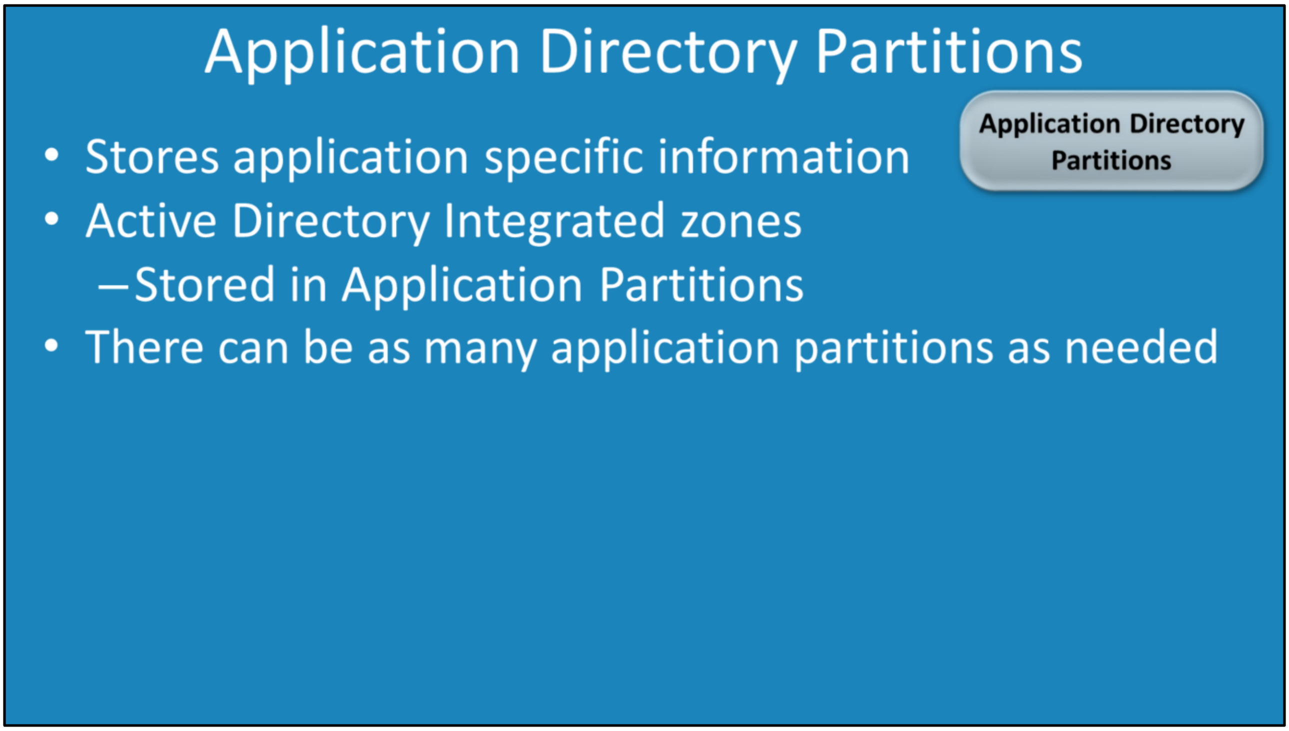
Configuration Partition

Schema Directory Partition

Global Catalog Partition

Application Directory Partitions

Active Directory Partitions
The Active Directory database may be hierarchy in nature but essentially it is a database stored in a single file name NTDS.DIT. Like a drive that may be divided in multiple parts, the Active Directory database is divided in multiple partitions. This is done for organization and replications needs. For example, certain partitions are configured to be replicated at the domain level while other partitions are configured to be replicated at the forest level.



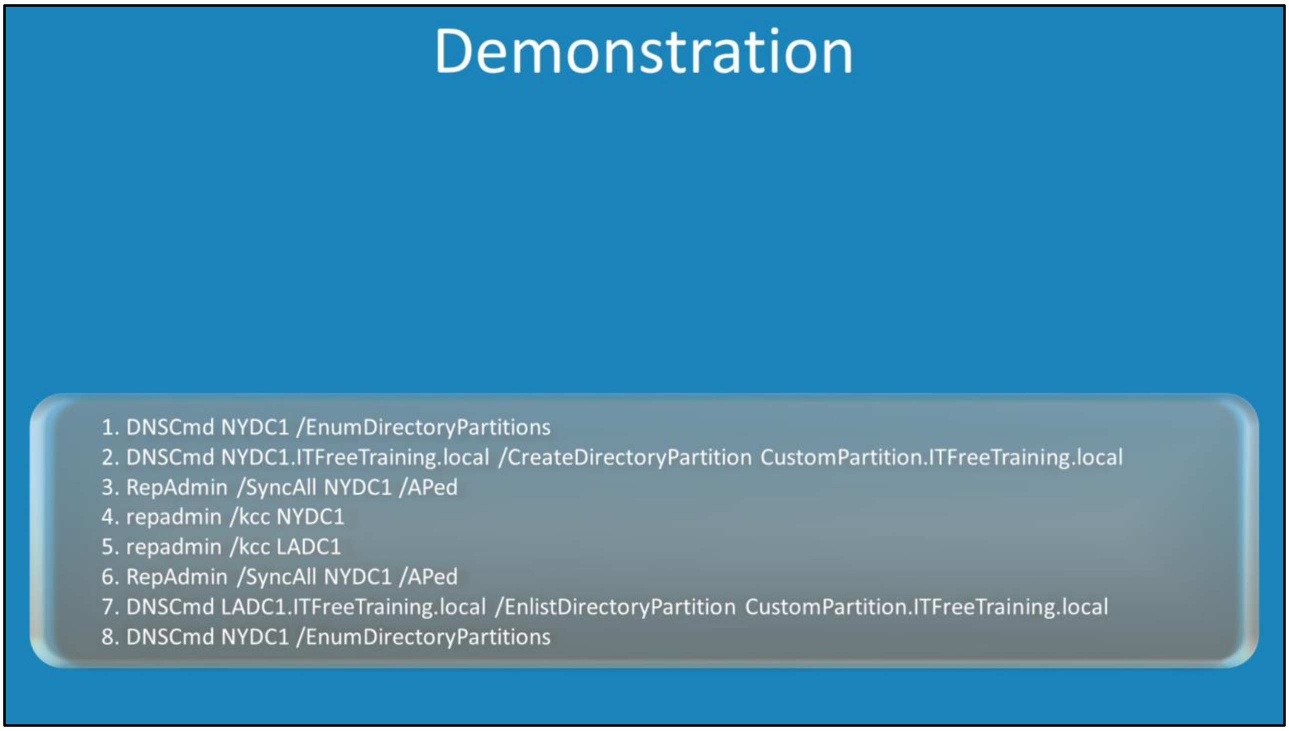
Application Directory Partitions

- Stores application specific information
- Active Directory Integrated zones
  - Stored in Application Partitions
- There can be as many application partitions as needed

Application Directory Partitions

Application Directory Partitions

This partition is used to store data from applications. This is different from the other partitions as there can be as many or few as required. Since Microsoft DNS server has data that needs to be stored and replicated around the domain or forest, this is a good choice for an application partition. DNS uses application partitions to store the data from Active Directory Integrated zones. Once stored in the application partition, like any other partition in the Active Directory database it is replicated to the required Domain Controllers using the Active Directory replication system.



Demonstration

```
1. DNSCmd NYDC1 /EnumDirectoryPartitions
2. DNSCmd NYDC1.ITFreeTraining.local /CreateDirectoryPartition CustomPartition.ITFreeTraining.local
3. RepAdmin /SyncAll NYDC1 /APed
4. repadmin /kcc NYDC1
5. repadmin /kcc LADC1
6. RepAdmin /SyncAll NYDC1 /APed
7. DNSCmd LADC1.ITFreeTraining.local /EnlistDirectoryPartition CustomPartition.ITFreeTraining.local
8. DNSCmd NYDC1 /EnumDirectoryPartitions
```

Demonstration

Active Directory Service Interfaces Editor (ADSI Edit) is a Lightweight Directory Access Protocol (LDAP) editor that allows modification for the objects and attributes in Active Directory.

The ADSIEdit tool is a tool that allows the administrator to see the data stored in the Active Directory database. Unlike other Active Directory tools, this provides a raw method and thus if you use this tool to make changes, changes should be made very carefully. ADSIEdit is found in Administrative Tools under the start menu or in the control panel. If you are using a client operating system like Windows 8, you will need to install remote server administration tools (RAST).

To connect to an Active Directory database, right click ADSIEdit and select the option connect to. There are a lot of different options in here. Under the option “select a well known naming context” you can select common partitions, for example Default naming context, configuration, RootDSE and schema. In this particular case I will connect to the configuration partition that contains forest wide configuration information for Active Directory. This partition itself does not contain any DNS data, however it does list the application partitions that are currently being used and thus gives us a better understanding of DNS stored data in Active Directory.

To see what partitions are currently configured in Active Directory, expand down through configuration down to partitions.

In the partition container there should be entries for the standard partition types Enterprise Configuration, Enterprise Schema and a domain partitions which will be named after your domain. If you have multiple domains, there will be one partition for each domain. Unlike the other partitions which will have a friendly name associated with it, each application partition will have a unique id associated with it in the form of a random appearing string of characters.

In the column Directory Partition Name will be a friendly name for the application partition. In this you should have the name ForestDnsZones and DomainDnsZones.

To connect to a particular application partition, right click ADSI Edit and select connect to. Under the option "Select or type a Distinguished Name or Naming Context" and then enter in the distinguished name for the application partition. In this example the forest DNS zone partition is "DC=ForestDNSZones,DC=ITFreeTraining,DC=Local".

To see the DNS data, expand down through ForestDNSZones and then to MicrosoftDNS. Any data that is stored in DNS and configured to replicate at the forest level will be found in here.

To connect to DNS Domain application partition, right click ADSI Edit and select connect to. Under the option "Select or type a Distinguished Name or Naming Context" enter in the distinguished name for the Domain application partition. In this example the forest DNS zone partition is "DC=DomainDNSZones,DC=ITFreeTraining,DC=Local".

Like the forest application partitions, if you expand down to MicrosoftDNS you can see all the DNS data that is stored for that domain in Active Directory. If you were to open a particular container that contained a primary zone stored in Active Directory, you would be able to see all the DNS records that are stored in that zone. It is also possible to edit the records here, however it is recommended to use the DNS Manager to perform changes as this has additional error checking. Changing data using low level tools like ADSIEdit has no additional error checking.

If you open DNS Manager from Administrative tools and expand down through the zone you should be able to see the same data that is in ADSIEdit in a different friendly format.

If you want to configure a particular DNS partition to be replicated to the domain or forest level, open the properties for the zone from DNS Manager. On the general tab, press the button change on the right hand side of replication (3 button down). In this you have 3 options. The first two select if the partition is replicated to all Domain Controllers in the forest or limited to only Domain Controllers in this domain.

Once you make changes to the replication for a zone in DNS Manager, if you look inside ADSIEdit the data should have moved from the application partition DomainDNSZones to ForestDNSZones or vice versa depending on whether you choose Forest or Domain replication.

Custom DNS Application Partitions

To create custom application partitions this needs to be done from the command line. The commands that I run here are available in a script. The download location

for this script is shown below. Remember these commands are configured to run on the particular Domain Controller used in the example, if you plan to run any of the commands on your network you will need to make additional changes.
http://itfreetraining.com/handouts/dns/dnscreatpartition.zip

DNSCmd is a command line utility to perform DNS administration.

```
DNSCmd NYDC1 /EnumDirectoryPartitions
```

This command shows all the DNS application directory partitions in the forest and domains.

```
DNSCmd NYDC1.ITFreeTraining.local /CreateDirectoryPartition
CustomPartition.ITFreeTraining.local
```

The command creates an additional application Partition. The first parameter is a Domain Controller that you will connect to in order to create the partition. The next parameter tells DNSCmd that you want to create an Application partition. The last parameter is the name of the partition that you want the new partition to be called.

```
RepAdmin /SyncAll NYDC1 /APed
```

The command RepAdmin is used to control Active Directory replication. The parameter /SyncAll tells the command that the Domain Controller should replicate changes to all its partners. The next parameter is the Domain Controller that you want to run the command from. The last parameter is switches you can use in the command, the available switches are listed below. Multiple switches can be combined together. Source http://technet.microsoft.com/en-us/library/cc835086.aspx It should also be remembered that this performs a background replication so depending how much data you have will determine how long the synchronization will take to complete.
/a Aborts, if any server is unavailable.
/A Synchronizes all naming contexts that are held on the home server.
/d Identifies servers by distinguished name in messages.
/e Synchronizes domain controllers across all sites in the enterprise. By default, this command does not synchronize domain controllers in other sites.
/h Displays Help.
/i Iterates indefinitely.
/I Runs the repadmin /showrepl command on each server pair in the path instead of synchronizing.
/j Synchronizes adjacent servers only.
/p Pauses after every message to allow the user to abort the command.

/P Pushes changes outward from the specified domain controller.
/q Runs in quiet mode, which suppresses call back messages.
/Q Runs in very quiet mode, which reports fatal errors only.
/s Does not synchronize.
/S Skips the initial server response check.

```
repadmin /kcc NYDC1
```

The parameter KCC forces the Knowledge Consistency Checker to run on the Domain Controller specified in this case NYDC1. The KCC is responsible for making connections between domain controllers. Depending on which partitions exist will determine which connections are required between different Domain Controllers. For example, an application partition may be configured to replicate between domains in the same forest and therefor there must a replication path between the two Domain Controllers which have a connection between them. One in each domain. If this connection does not exist, the KCC will pick this up and create the connection. The important point to remember is that the KCC uses the Active Directory Database to make these decisions. If you find the connections are missing and partitions have not been replicated, force a replication for the Domain Controllers so the data is up to date on the Domain Controller. Once this occurs, force the KCC to run. Once the data is the same, the Domain Controllers will make the same decisions about which connections are required and will make the same connections.

```
repadmin /kcc LADC1
```

This does the same as the command above but for the Domain Controller LADC1. It should be remembered that the KCC will automatically run locally on each Domain Controller given enough time. Thus changes to Active Directory, assuming data has being replicated, will automatically create the required connections.

```
RepAdmin /SyncAll NYDC1 /APed
```

Since the KCC has run, another sync was performed as the connection between the Domain Controllers may have changed. In a large production environment it will take time for replication and the KCC to run so it should not run as quickly as it was run here. Assuming replication is working between your Domain Controllers, given time the KCC should automatically run and create any required or missing links.

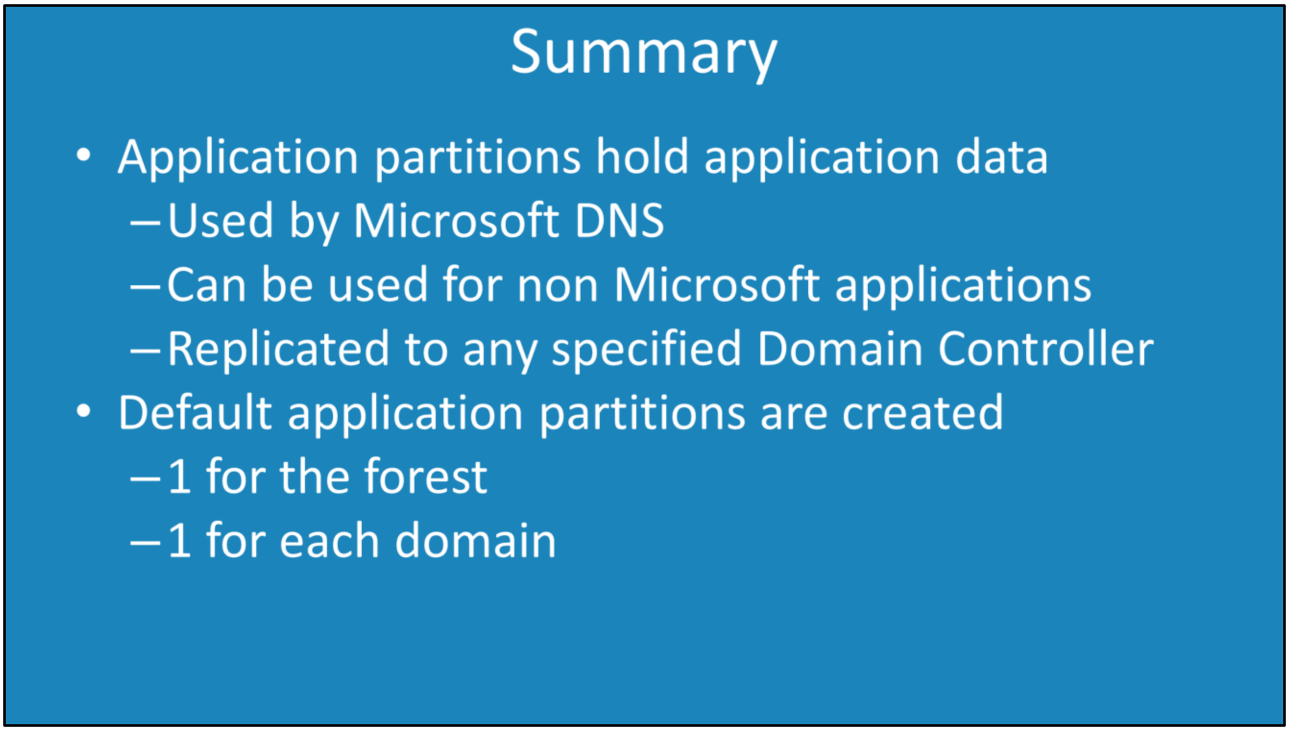
```
DNSCmd LADC1.ITFreeTraining.local /EnlistDirectoryPartition
CustomPartition.ITFreeTraining.local
```

The second command that was run created an Application Partition, however this partition is not replicated with any Domain Controllers. This command uses the Enlist Parameter to add the Domain Controller LADC1 to the replication list so that it will receive any changes to the data stored in this partition.

```
DNSCmd NYDC1 /EnumDirectoryPartitions
```

This command shows the Application Partitions. After you create your Application partitions it should appear in here.

When you create a custom application partition, in the replication options for a zone you will have the option available “To all domain controllers in the scope of this directory partition” which was previously not available.



Summary

- Application partitions hold application data
  - Used by Microsoft DNS
  - Can be used for non Microsoft applications
  - Replicated to any specified Domain Controller
- Default application partitions are created
  - 1 for the forest
  - 1 for each domain

Summary
This video has looked at the Application partition in particular its use in DNS. Unlike the other partition types in the Active Directory database, there can be many or only few required. Even though this video has looked at just storing data in an Application Partition for Microsoft DNS, it is possible for any application to create its own application partition and store the data in this partition. By default, when a forest is created there is one application partition created to store DNS data in. There is also one application partition created for DNS data created for each domain on the forest. Even though it is possible for other applications to store data in these partitions, it is recommended that the application create their own partition to store their own data in. Once an application partition is created, it can be replicated to as many or as few Domain Controllers as required.

See http://YouTube.com/ITFreeTraining or http://itfreetraining.com for our always free training videos. This is only one video from the many free courses available on YouTube.

References
"How to create and apply a custom application directory partition on an Active Directory integrated DNS zone in Windows Server 2003" http://support.microsoft.com/kb/884116
"Repadmin /syncall" http://technet.microsoft.com/en-us/library/cc835086(v=ws.10).aspx
http://itfreetraining.com/handouts/dns/dnscreatpartition.zip