# CNT 4603: System Administration
# Spring 2011

## Introduction To Active Directory – Part 2


Instructor : Dr. Mark Llewellyn
markl@cs.ucf.edu
HEC 236, 4078-823-2790
http://www.cs.ucf.edu/courses/cnt4603/spr2011

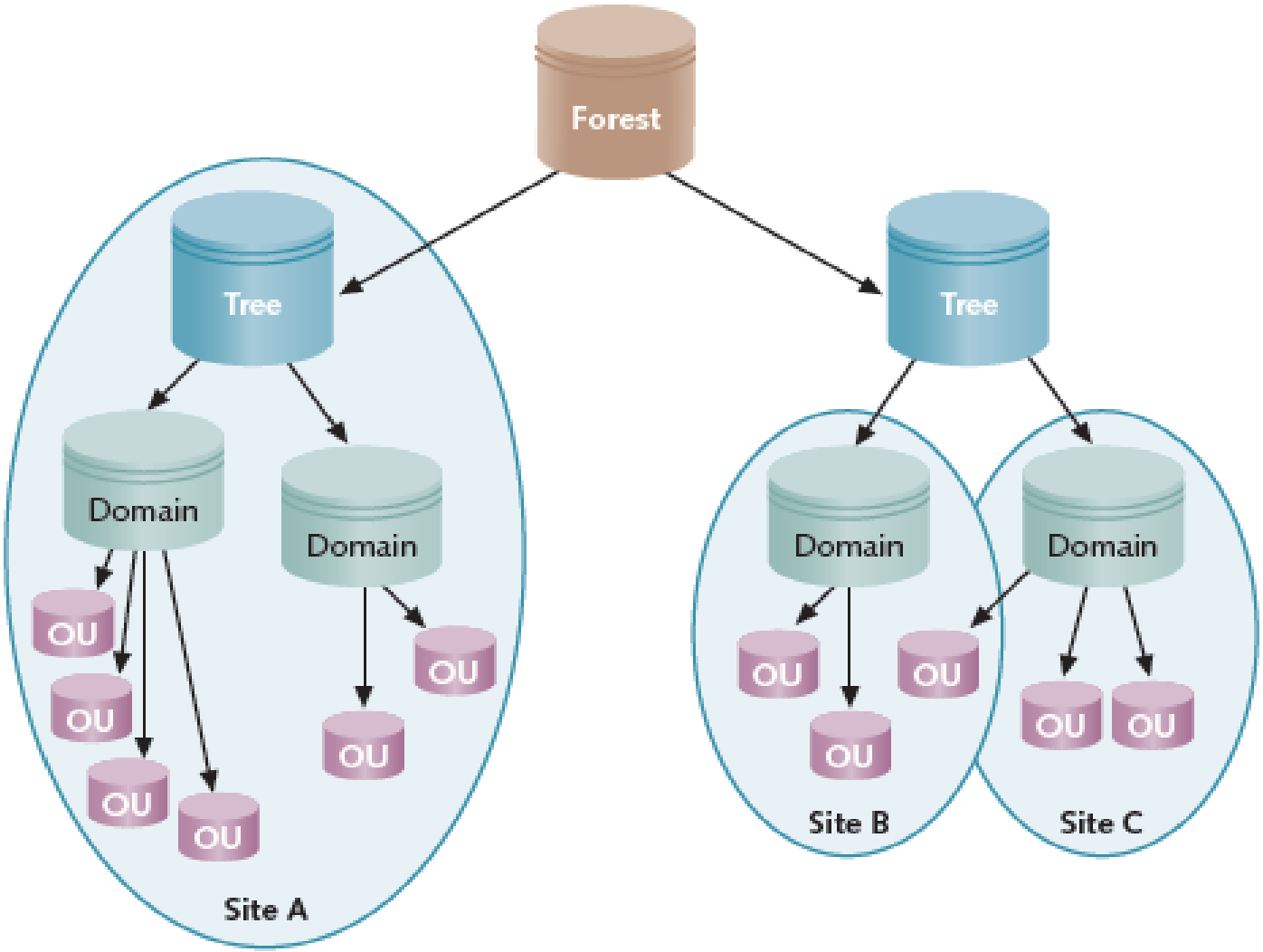
Department of Electrical Engineering and Computer Science
University of Central Florida


# Containers In Active Directory

- We saw in the previous section of notes that AD has a treelike structure which is based on the X.500 standard for directory structures.
- In a normal directory structures, folders contain subfolders and within subfolders there can be more subfolders to an arbitrary depth.
- Just as files are the basic element that are grouped in a hierarchy of folders and subfolders, objects are the basic elements of AD and are grouped into a hierarchy of containers.
- Containers in AD include forests, trees, domains, OUs, and sites.



# Containers In Active Directory






# Active Directory - Forests

- At the highest level in an AD design is the forest.
- A forest consists of one or more AD trees that are in a common relationship and that have the following characteristics.
  - The trees can use a disjointed namespace.
  - All trees use the same schema.
  - All trees use the same global catalog.
  - Domains enable administration of commonly associated objects, such as accounts and other resources, within a forest.
  - Two-way transitive trusts (resources that are equally shared) are automatically configured between domains within a single forest.



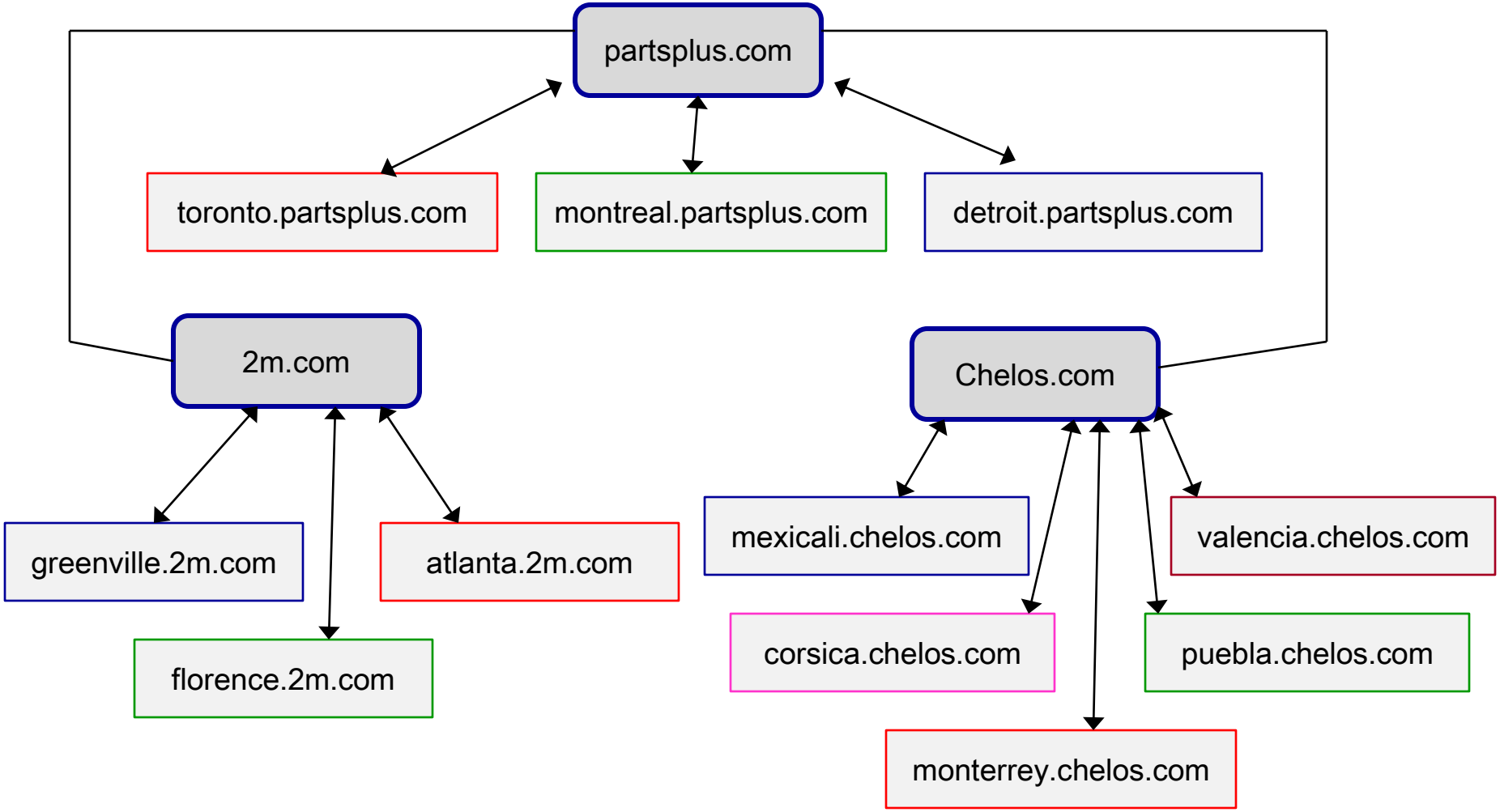
# Active Directory - Forests

- A forest provides a means to relate trees that use a contiguous namespace in domains within each tree but that have disjointed namespaces in relationship to each other.
- Consider the following scenario: an international automotive parts company that is really a conglomerate of three different companies, each with a different brand name. The parent company is PartsPlus, located in Toronto. PartsPlus manufactures alternators, coils and other electrical parts at plants in Toronto, Montreal, and Detroit, and has a tree structure for domains that are part of partsplus.com.
- Another company owned by PartsPlus is Marty & Mikes (2m.com) makes radiators in two South Carolina cities, Florence and Greenville, and radiator fluid in Atlanta.

# Active Directory - Forests

- A third member company, Chelos (chelos.com) makes engine parts and starter motors in Mexico City, Corsica, Monterrey, and Puebla, all in Mexico – and also has a manufacturing site in Valencia, Venezuela.
- In this situation, it makes sense to have a contiguous tree structure for each of the three related companies and to join the trees in a forest of disjointed name spaces.
- This is illustrated in the figure on page 7.




partsplus.com
toronto.partsplus.com
montreal.partsplus.com
detroit.partsplus.com
2m.com
Chelos.com
greenville.2m.com
atlanta.2m.com
florence.2m.com
mexicali.chelos.com
valencia.chelos.com
corsica.chelos.com
puebla.chelos.com
monterrey.chelos.com

# A Forest



# Active Directory - Forests

- The advantage of joining trees into a forest is that all domains share the same schema and global catalog.
- A schema is set up at the root domain (which is partsplus.com in the previous example), and the root domain is home to the master schema server.
- At least one DC (domain controller) functions as a global catalog server. However, in the previous example it would be likely that you would plan to have a global catalog server located at each geographic location (domain).



# Active Directory - Forests

- Windows Server 2008 AD recognizes three types of forest functional levels.
- The forest functional level refers to the AD functions supported forest-wide.
- The functional levels are:
- *Windows 2000 native forest functional level* – provides AD functions compatible with a network that has a combination of Windows 2000 Server, Windows 2003 Server, and Windows 2008 Server domain controllers.
- *Windows Server 2003 forest functional level* – intended for Windows Server 2003 and 2008 domain controllers only and enables more forest management functions, such as more options for creating trust relationships between forests, domain renaming, Read-Only Domain Controllers, cross-forest authentication of users, and enhanced replication of AD.



# Active Directory - Forests

- *Windows Server 2008 forest functional level* – contains only Windows Server 2008 domain controllers. Currently this level has no more functional features that in the Windows Server 2003 forest functional level, although there is room for new features that can be added later. This level is also included for compatibility with the domain functional levels we'll see later.



# Active Directory - Trees

- A tree contains one or more domains that are in a common relationship, and has the following characteristics.
  - Domains are represented in a contiguous namespace and can be in a hierarchy.
  - Two-way trust relationships exist between parent domains and child domains, essentially creating a trust path.
  - All domains in a single tree use the same schema for all types of common objects.
  - All domains use the same global catalog.

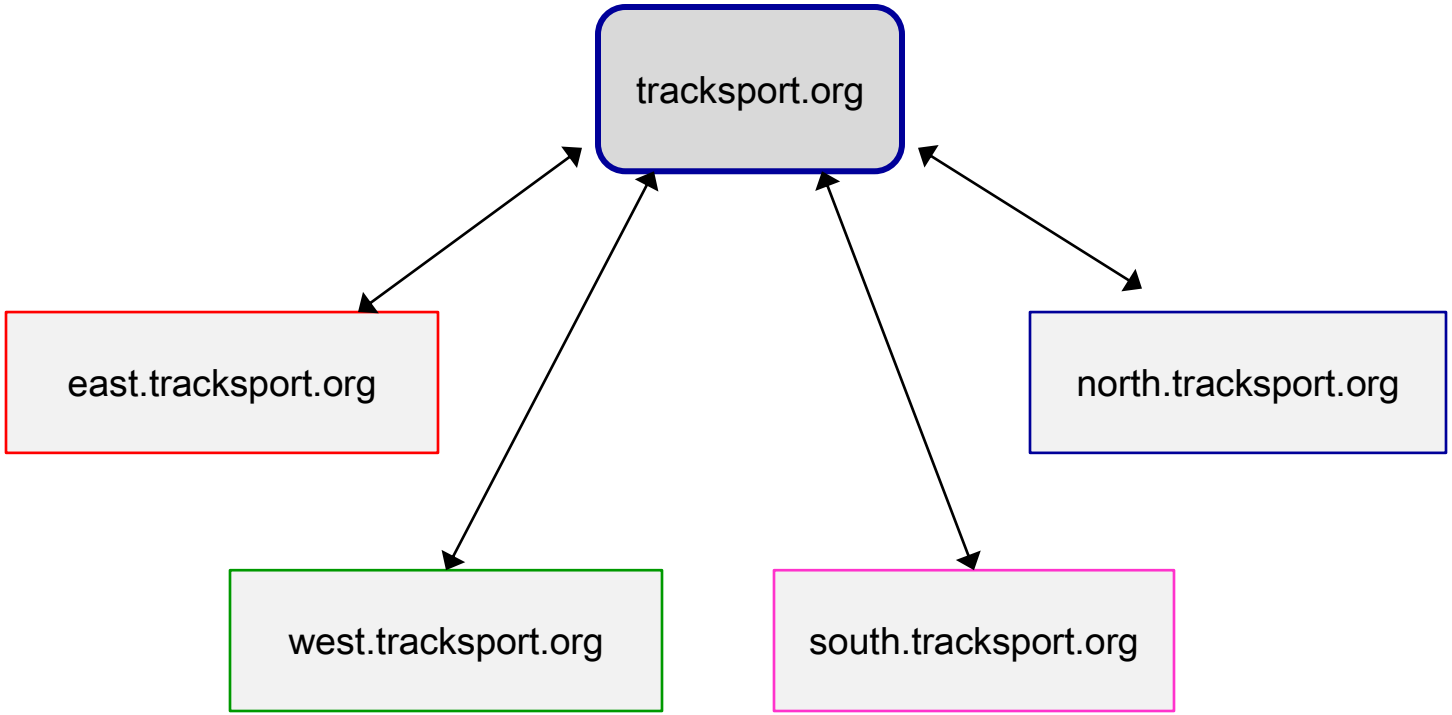


# Active Directory - Trees

- The domains in a tree typically have a hierarchical structure, such as a root domain at the top and other domains under the root (similar to a parent-child relationship).
- Consider the following example: tracksport.org might be the root domain and have four domains under the root to form one tree: east.tracksport.org, west.tracksport.org, north.tracksport.org, and south.tracksport.org, as shown on page 13.
- These domains use the contiguous namespace format in that the child domains each inherit a portion of their namespace from the parent domain.



# Active Directory - Trees




A Tree



# Active Directory - Trees

- The domains within a tree are in what is called a Kerberos transitive trust relationship, which consists of two-way trusts between parent domains and child domains.
- A transitive trust means that if A and B have a trust and B and C have a trust, the A and C automatically have a trust as well.
  - This transitive property comes from first-order predicate logic and inference axioms. The transitive axiom states that if A implies B and B implies C, then A also implies C. Example: if CNT 4603 meets on Monday, and Monday is today, then by implication, CNT 4603 meets today.



# Active Directory - Trees

- A trusted domain is one that is granted access to resources, whereas a trusting domain is the one granting the access.
- In a two-way trust, members of each domain can have access to the resources of the other. In other words, either domain can play the role of both a trusted domain and also that of a trusting domain.

Windows Server 2008 (and Server 2003) also have a forest trust. In a forest trust, a Kerberos transitive trust relationship exists between the root domains in Windows Server 2008 forests, resulting in trust relationships between all domains in the forest.

# Active Directory - Trees

- Because of the trust relationship between parent and child domains, any one domain can have access to the resources of all the others.
- The security in the two-way trust relationships is based on Kerberos techniques, using a combination of protocol-based and encryption-based security techniques between clients and servers.
- A new domain joining a tree has an instant trust relationship with all the other member domains through the trust relationship that is established with its parent domain, which makes all objects in the other domains available to the new one.



# Active Directory - Trees

- All domains within a single tree (as well as all trees in a single forest) share the same schema defining all the object types that can be stored within AD.
- Further, all domains in a tree also share the same global catalog and a portion of their namespace.
- In addition, a child domain contains part of the namespace of the parent domain.



# Active Directory - Domain

- Microsoft views a domain as a logical partition within an AD forest.
- A domain is a grouping of objects that typically exist as a primary container within AD.
- The basic functions of a domain are as follows:
  - To provide an AD "partition" in which to house objects, such as accounts and groups, that have a common relationship, particularly in terms of management and security.
  - To establish a set of information to be replicated from one DC to another.
  - To expedite management of a set of objects.

# Active Directory - Domain

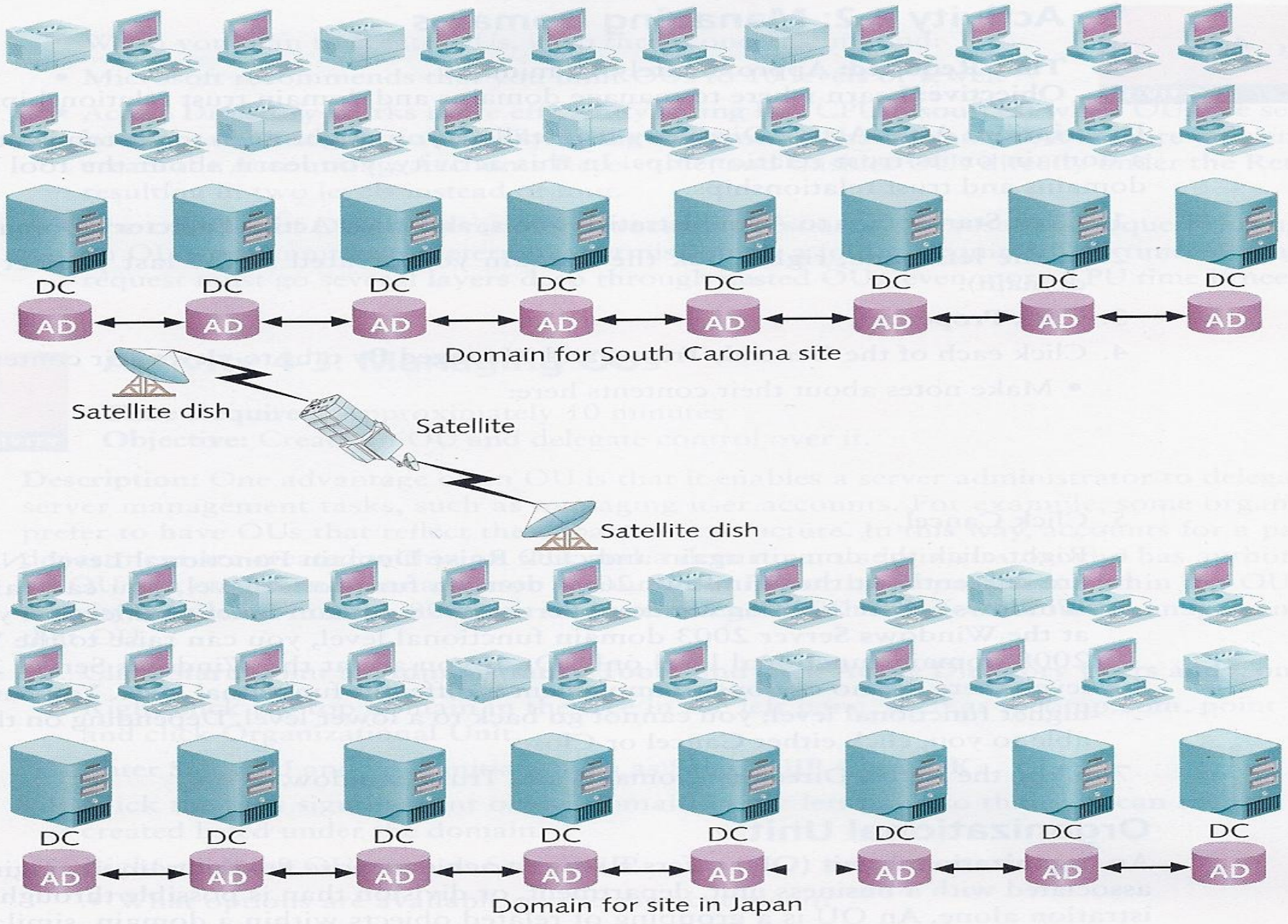
- When you use a server-based networking model to verify who to log on to the network, there is at least one domain.
  - For example, if you are planning an AD for a small organization of 34 employees who have workstations connected to a network that has one or two Windows Server 2008 server, then one domain is sufficient for that organization.
- The domain functions as a partition within which to group all of the network objects consisting of servers, user account, shared printers, and shared folders and files.



# Active Directory - Domain

- In a midsized or larger organization, you might use more than one domain. This would be especially true if the business units are separated by large distances and you want to limit the amount of DC replication over expensive WAN links or to manage the objects differently between locations, such as through different account or security policies.
- Consider the following example: a company builds tractors in South Carolina and has a parts manufacturing division in Japan. Each site has a large enterprise network of Windows Server 2008 servers, and the sites are linked together in a WAN by an expensive satellite connection. When you calculate the cost of replicating DCs over the satellite link, you cannot justify it in terms of the increased traffic that will delay other virtual daily business communications. In this situation it makes sense to create two separate domains, one for each site, as shown on the next page.




DC
AD
Domain for South Carolina site
Satellite dish
Satellite
Satellite dish
Domain for site in Japan

# Active Directory - Domain

- Windows Server 2008 AD recognizes three domain functional levels, which refers to the Windows Server OS on domain controllers and the domain-specific functions they support.
- The domain functional levels are:
- *Windows 2000 domain functional level* – provides AD functions compatible with a network that has a combination of Windows 2000 Server, Windows 2003 Server, and Windows 2008 Server domain controllers. This level supports universal groups, which were not previously supported in Windows NT Server, converting types of groups, and nesting groups.
- *Windows Server 2003 domain functional level* – intended for Windows Server 2003 and 2008 domain controllers only and enables more domain management functions, such as delegating management of AD object, time stamps for logons, use of Authorization Manager policies in AD, and other features not available in Windows Server 2000 domain controllers.



# Active Directory - Domain

- *Windows Server 2008 domain functional level* – contains only Windows Server 2008 domain controllers, and offers new features such as default incorporation of the Distributed File System (DFS), with better security, enhanced security for Kerberos authentication, Advanced Encryption Standard (AES) encryption servers, and enhanced user account password policies, including fine-grained password policies.



# Active Directory – Organizational Unit

- An organizational unit (OU) offers a way to achieve more flexibility in managing the resources associated with a business unit, department, or division than is possible through domain administrations alone.
- An OU is a grouping of related objects within a domain, similar to the idea of having subfolders within a folder.
- OUs can be used to reflect the structure of the organization without having to completely restructure the domain(s) when that structure changes.
- OUs allow the grouping of objects so that they can be administered using the same group policies, such as security and desktop setup.

# Active Directory – Organizational Unit

- OUs also make it possible for server administration to be delegated or decentralized.
  - For example, in a software development company in which the employees are divided into 15 project teams, the user accounts, shared files, shared printers, and other shared resources of each team can be defined as objects in separate OUs. There would be 1 domain for the entire company and 15 OUs within that domain, all defined in AD.
  - With this arrangement, file and folder objects can be defined to specific OUs for security, and the management of user accounts can be delegated to each group leader (OU administrator).



# Active Directory – Organizational Unit

- OUs can also be nested within OUs, as subfolders are nested in subfolders, so that you can create them several layers deep.
  - For example, you might have one OU under the Retail Sales OU for the Accounting Department, and OU under the Accounting Department for the Accounts Receivable Group, and an OU under Accounts Receivable for the Cashiers, thus creating four layers of OUs.
- The problem with this approach is that creating OUs many layers deep can get as confusing as creating subfolders several layers deep. It is confusing for the server administrator to track layered OUs, and it is laborious for AD to search through each layer.



# Active Directory – Organizational Unit

- When you plan to create OUs, keep three things in mind:
  1. Microsoft recommends that you limit OUs to 10 levels or fewer.
  2. AD works more efficiently (using less CPU resources) when OUs are set up horizontally instead of vertically. For example, it is more efficient to create the Accounting, Accounts Receivable, and Cashier OUs directly under the Retail OU, resulting in two levels instead of four.
  3. The creation of OUs, involves more processing resources because each request through an OU (for example, to determine the permission of a folder) requires CPU time. When that request must go several layers deep through nested OUs, even more CPU time is needed.



# Active Directory – Site

- A site is a TCP/IP –based concept (container) within AD that is linked to IP subnets and has the following functions:
  - Reflects one or more interconnected subnets, usually having good network connectivity.
  - Reflects the physical aspect of the network.
  - Is used for DC replication.
  - Is used to enable a client to access the DC that is physically closest.
  - Is composed of only two types of objects, servers and configuration objects.



# Active Directory – Site

- Sites are based on connectivity and replication functions.
- You might think of sites as a way of grouping AD objects by physical location so AD can identify the fastest communication paths between clients and servers and between DCs.
- The physical representation of the network to AD is accomplished by defining subnets that are interconnected. For this reason, one site may be contained within a single OU or a single domain, or a site may span multiple OUs and domains, depending on how subnets are setup.
- The most typical boundary for a site consists of the LAN topology and subnet boundaries rather than the OU and domain boundaries.

# Active Directory – Site

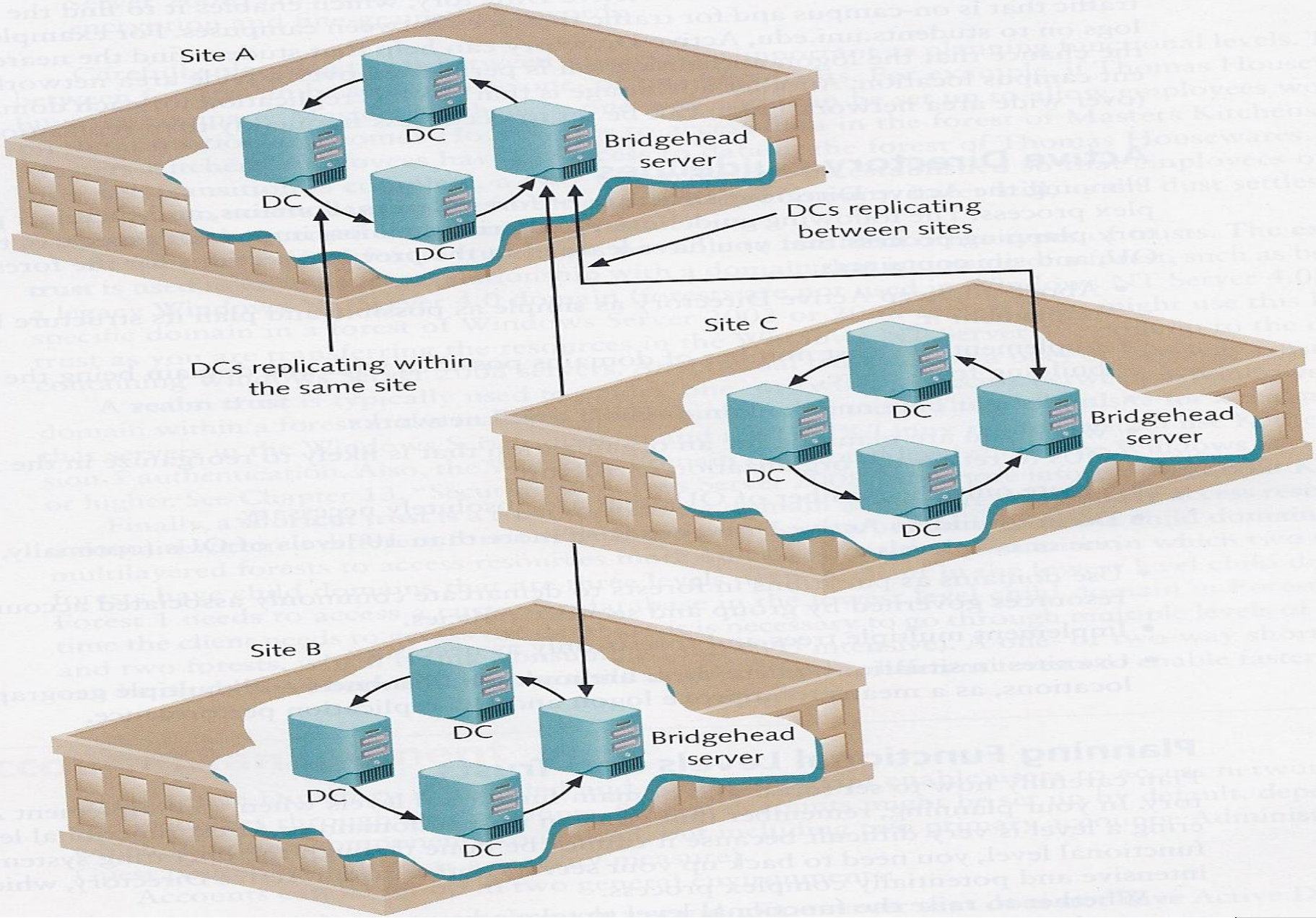
- There are two important reasons to define a site.
- First, by defining site locations based on IP subnets, you enable a client to access network servers using the most efficient route.
- In the partsplus.com example (see pages 5-7), it is faster for a client in Toronto to be authenticated by a Toronto global catalog server than for a client to go through Detroit or Mexico City.
- Second, DC replication is the most efficient when AD has information about which DCs are in which locations.
- Within a site, each DC replicates forest, tree, domain, and OU naming structures, configuration naming elements, such as computers and printers, and schema information.



# Active Directory – Site

- One advantage of creating a site is that it sets up redundant paths between DCs so that if one path is down, there is a second path that can be used for replication.
- This redundancy is in a logical ring format, which means that replication goes from DC to DC around a ring until each DC is replicated.
- If a DC is down along the main route, the AD uses site information to send replication information in the opposite direction around the ring.
- Whenever a new DC is added or an old one removed, AD reconfigures the ring to make sure there are two replication paths available from each DC.
- Between sites, replication is coordinated through one server, called a bridgehead server, located at each site. See next page.




Site A
DC
DC
DC
Bridgehead server
DCs replicating between sites
DCs replicating within the same site
Site C
DC
DC
DC
Bridgehead server
Site B
DC
DC
DC
Bridgehead server



# Active Directory – Site

- When you replicate between sites, the replication occurs only between two bridgehead servers.
- The bridgehead server is a DC that is designated to have the role of exchanging replication information.
- Only one bridgehead server is set up per site, so the network traffic per site is kept to a minimum. Otherwise, having multiple DCs replicating with partners across sites could take up considerable bandwidth.



# Active Directory Guidelines

- Planning the AD structure of forests, trees, domains, and OUs is a potentially complex process. The following guidelines summarize the most important aspects of the AD planning process.
- Above all, keep AD as simple as possible and plan its structure before you implement it.
- Implement the least number of domains possible, with one domain being the ideal and building from there.
- Implement only one domain on most small networks.
- When you are planning for an organization that is likely to reorganize in the future, use OUs to reflect the organization's structure.

# Active Directory Guidelines

- Create only the number of OUs that are absolutely necessary.
- Do not build an AD with more than 10 levels of OUs (optimally, no more than one or two levels).
- Use domains as partitions in forests to demarcate commonly associated accounts and resources governed by group and security policies.
- Implement multiple trees and forests only as necessary.
- Use sites in situations where there are multiple IP subnets and multiple geographic locations, as a means to improve logon and DC replication performance.