ASYMMETRIC STORAGE VIRTUALISATION

In contrast to symmetric virtualisation, in asymmetric virtualisation the data flow is separated from the control flow. This is achieved by moving all mapping operations from

logical to physical drives to a metadata controller outside the data path (Figure 5.17). The metadata controller now only has to look after the administrative and control tasks of virtualisation, the flow of data takes place directly from the application servers to the storage devices. As a result, this approach is also called out-band virtualisation.

The communication between metadata controller and agents generally takes place via the LAN (out-band) but can also be realised in-band via the storage network. Hence, in our opinion the terms ‘in-band virtualisation’ and ‘out-band virtualisation’ are a little misleading. Therefore, we use instead the terms ‘symmetric virtualisation’ and ‘asymmetric virtualisation’ to refer to the two network-based virtualisation approaches. Like the symmetric approach, the metadata controller is logically structured in two

layers (Figure 5.18). The volume management layer has the same duties as in the symmetric approach. The second layer is the control layer, which is responsible for the communication with an agent software that runs on the servers.

The agent is required in order to enable direct access to the physical storage resources. It is made up of a data access layer with the same tasks as in symmetric virtualisation and a control layer (Figure 5.18). Via the latter it loads the appropriate location and access information about the physical storage from the metadata controller when the virtual storage is accessed by the operating system or an application. In this manner, access control to the physical resources is still centrally managed by the metadata controller. An agent need not necessarily run in the memory of the server. It can also be integrated into a host bus adapter. This has the advantage that the server can be freed from the processes necessary for virtualisation.

In asymmetric storage virtualisation – as is also the case for symmetric storage virtualisation – advanced storage functions such as snapshots, mirroring or data migration can be realised. The asymmetric model is, however, not so easy to realise as the symmetric one, but performance bottlenecks as a result of an additional device in the data path do not occur here. If we want to increase performance by the use of caching for both application as well as metadata, this caching must be implemented locally on every application server. The caching algorithm to be used becomes very complex since it is a distributed environment, in which every agent holds its own cache (Section 4.3). Data inconsistencies as a result of different cache contents for the same underlying physical storage contents must be avoided and error situations prevented in which an application crashes, that still has data in the cache. Therefore, additional mechanisms are necessary to guarantee the consistency of the distributed cache. Alternatively, the installation of a dedicated cache server in the storage network that devotes itself exclusively to the caching of the data flow would also be possible. Unfortunately, such products are not currently (2009) available on the market.

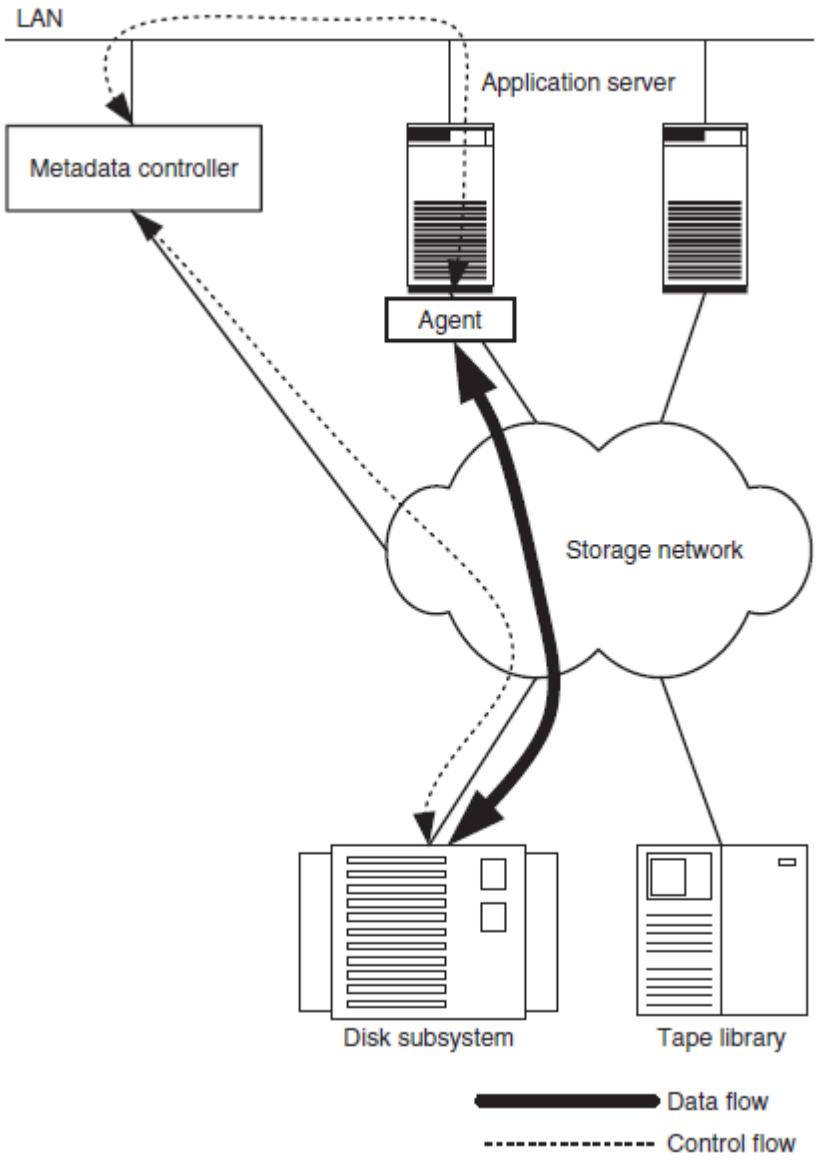

Figure 6.17 In contrast to symmetric virtualisation, in asymmetric virtualisation the data flow is separated from the control flow. The abstraction of physical to logical storage thus takes place outside the data flow.

Metadata controllers can also be constructed as clusters for the load distribution of the control flow and to increase fault-tolerance. The implementation is, however, easier with the asymmetric approach than it is with the symmetric since only the control flow has to be divided over several computers. In contrast to the symmetric approach, the splitting of

the data flow is dispensed with.

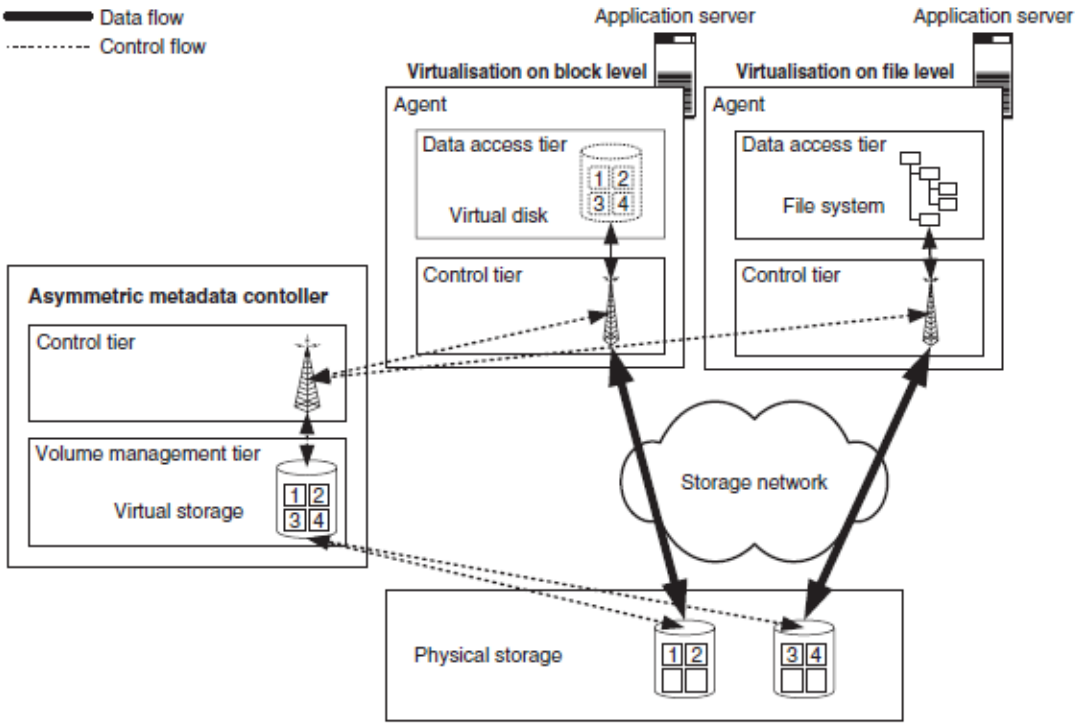

Figure 5.18 In asymmetric virtualisation the metadata controller takes on only the administrative control tasks for the virtualisation. Access to the physical storage is realised by means of an agent software.

The following advantages of asymmetric virtualisation can be established:

• Complete control of storage resources by an absolutely centralised management on the metadata controller.

• Maximum throughput between servers and storage devices by the separation of the control flow from the data flow, thus avoiding additional devices in the data path.

• In comparison to the development and administration of a fully functional volume manager on every server, the porting of the agent software is associated with a low cost.

• As in the symmetric approach, advanced storage functions such as snapshots or mirroring can be used on storage devices that do not themselves support these functions.

• To improve fault-tolerance, several metadata controllers can be brought together to form a cluster. This is easier than in the symmetric approach, since no physical connection from the servers to the metadata controllers is necessary for the data flow.

The disadvantages of asymmetric virtualisation are:

• A special agent software is required on the servers or the host bus adapters. This can make it more difficult to use this approach in heterogeneous environments, since such software or a suitable host bus adapter must be present for every platform. Incompatibilities between the agent software and existing applications may sometimes make the use of asymmetric virtualisation impossible.

• The agent software must be absolutely stable in order to avoid errors in storage accesses.

In situations where there are many different platforms to be supported, this is a very complex development and testing task.

• The development cost increases further if the agent software and the metadata controller

are also to permit access on file level in addition to access on block level.

• A performance bottleneck can arise as a result of the frequent communication between

agent software and metadata controller. These performance bottlenecks can be remedied

by the caching of the physical storage information.

• Caching to increase performance requires an ingenious distributed caching algorithm to avoid data inconsistencies. A further option would be the installation of a dedicated cache server in the storage network.

• In asymmetric virtualisation there is always the risk of a server with no agent software

being connected to the storage network. In certain cases it may be possible for this server to access resources that are already being used by a different server and to accidentally destroy these. Such a situation is called a rogue host condition.

Source : http://elearningatria.files.wordpress.com/2013/10/cse-viii-storage-area-networks-06cs833-notes.pdf