

HP Enterprise Data Warehouse Appliance architecture overview and performance guide

Introduction to Business Intelligence architectures

Technical white paper

Table of contents

Executive summary.....	2
ETL tier – Extract, Transformation and Load processes	3
ETL tier and the HP Enterprise Data Warehouse	3
Data warehouse tier – ROLAP	3
The data warehouse tier and the HP Enterprise Data Warehouse	4
Data mart tier– ROLAP	5
OLAP cube tier – MOLAP Multi-dimensional Online Analytical Processing	5
End users – portals, data access and analysis tools.....	5
Optimization of hardware for data warehousing or OLTP	6
HP Enterprise Data Warehouse architecture.....	7
EDW data rack	9
Storage node	9
Database server node.....	10
Compute nodes and high availability	11
EDW control rack	11
Backup server node.....	11
Management server nodes – EDW domain controllers	13
Landing zone server	13
Control server nodes (active/passive) cluster	14
EDW – data rack network	15
How EDW achieves high throughput.....	16
Why I/O throughput is important for data warehouses and data marts	16
Traditional database designs vs. EDW and Fast Track.....	17
EDW and Ultra-Shared Nothing technology	21
Other PDW software benefits	24
How loading data can affect EDW/PDW performance	25
EDW performance metrics	26
Summary	27
For more information.....	28

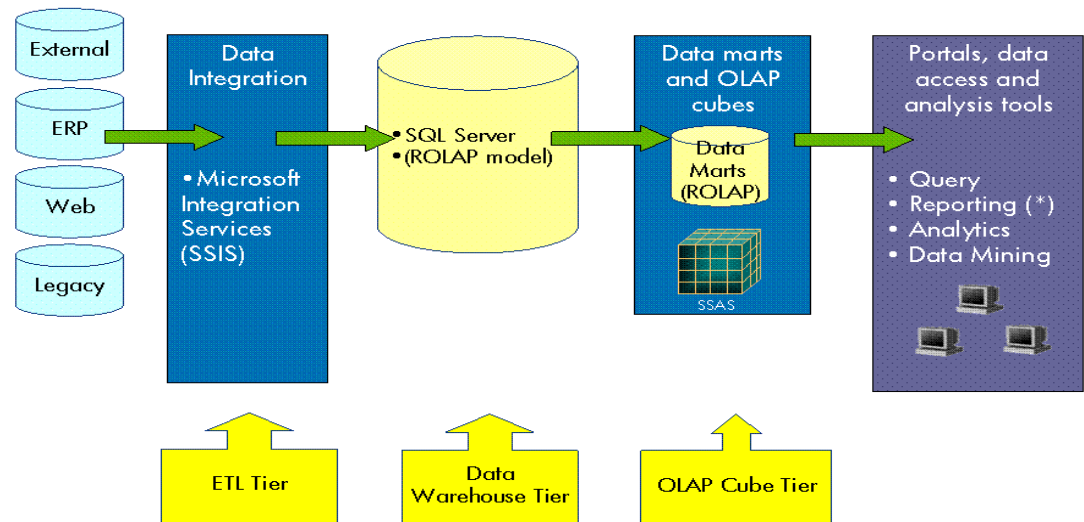


Executive summary

Business Intelligence (BI) applications are systems designed to help an organization make intelligent business decisions based upon the results of analyzing massive amounts of data.

Companies may architect their business intelligence environment in various ways. However, most BI implementations contain three main functional tiers. These include an Extract, Transformation and Load tier (ETL) using Microsoft® SQL Server Integration Services (SSIS), relational data warehouse or data mart tier (ROLAP), and a “Multi-dimensional Online Analytical Processing” (MOLAP) cube tier using Microsoft SQL Server Analysis Services or front end relational data mart. End user queries may access the ROLAP database directly on the Enterprise Data Warehouse (EDW) Appliance or on a ROLAP data mart using SQL Server Fast Track Data Warehouse (commonly referred to as “Fast Track”). It is also common for users to query MOLAP cubes which may provide “memory resident” access speeds. Figure 1 shows the relationship of the three major tiers of the BI environment.

Figure 1. Three tiers of BI environments



It is also common for an organizations operations staff to require high availability and provide the ability to backup and then restore data in the event of a failure.

NOTE:

The reader of this document will find the acronym “EDW”, which usually refers to the HP Enterprise Data Warehouse Appliance. But, in some cases EDW may refer to the generally accepted industry term “Enterprise Data Warehouse”. Therefore the reader should take note of the context in which this acronym is used.

NOTE:

The acronym “PDW” is short for SQL Server 2008 Parallel Data Warehouse R2 software. The HP EDW Appliance has been optimized for execution of PDW software.

ETL tier – Extract, Transformation and Load processes

The ETL tier uses SQL Server Integration Services to support three main data migration functions.

- Extract – Typically, the extract process reads data from an OLTP database or Operational Data Store (ODS) containing transactional data.
- Transformation – This process converts the extracted, operational data from its previous, normalized, OLTP structure into a form that uses a fact table(s) with conformed dimensions. The ETL software does this by remapping columns, using rules, using look-up tables and sometimes combining the data with external sources (such as demographic and spatial data) to generate easier-to-use, meaningful information. Meta data is also commonly maintained for documentation purposes and to ensure that all users of the data warehouse will have a consistent view not only of the information but also of how it may have been manipulated or derived in the multi-dimensional data warehouse structure.
- Load – Load is the process of loading, inserting or updating the transformed data into the target data warehouse or data mart. The majority of time during the load is typically spent loading or inserting new data into the fact table(s). Occasionally dimension tables are updated, but, the volume of changes to those tables are usually very low.

ETL tier and the HP Enterprise Data Warehouse

The HP Enterprise Data Warehouse (EDW) Appliance has been configured with a “Landing Zone” server. This server will do one of the following:

- Accept data directly from the customer’s OLTP systems and run ETL software (SQL Server Integration Services). This data may then be placed in a staging database to be bulk loaded into the destination data warehouse tables.
- Accept data from an external ETL server, which may have already remapped, transformed and cleansed the data to be loaded into the data warehouse.

The EDW Landing Zone, PDW “dataloader” software, and staging tables work together to load data into the data warehouse in an attempt to store data sequentially on disk. The sequential storage of data tends to minimize disk seek time and allow for faster table scans. This, in turn, allows the EDW Appliance to provide users with optimal I/O throughput when servicing queries.

It should be noted that dataloader also executes loads in parallel, across multiple servers. This will provide customers with shorter load windows than traditional SMP systems.

Data warehouse tier – ROLAP

The data warehouse is a data store designed to support the management decision making process. Transaction processing systems traditionally contain real-time operational data to support the day-to-day operations of the business. The data on these servers is usually short lived. In addition, OLTP data structures are usually stored in 3rd normal form, which may not be optimal for servicing business intelligence queries from users. In order for management to make good strategic decisions, this operational data is sometimes stored in an Operational Data Store or, more commonly, directly loaded into a data warehouse or data mart from the transaction processing servers.

Traditionally, the ETL tier manipulates and transforms operational, demographic and other data to be stored in a multi-dimensional, star or snowflake schema in the warehouse. These database structures, when coupled with meta-data and conformed dimensions, present a coherent picture of business conditions over time.

End users may query the EDW Appliance directly or, in some architectures, end users may query business unit or functionally specific front end data marts to address an organizations needs.

The data warehouse tier and the HP Enterprise Data Warehouse

The HP Enterprise Data Warehouse Appliance has been optimized to support the Microsoft SQL Server Parallel Data Warehouse (PDW) software. This Massively Parallel Processing (MPP) hardware and software combination will provide end users with high levels of I/O throughput by executing queries in parallel across 10 – 40 multi-core servers which are physically located within one to four data racks.

In addition to the data rack(s), each HP EDW Appliance must have one control rack. The control rack’s functionality will be addressed later in this document.

Table 1 shows some of the major differences between MPP and SMP systems.

Table 1. Major differences between MPP and SMP systems

Massively Parallel processing (MPP) HP EDW/PDW	Symmetric Multiprocessing (SMP)
Horizontally scalable hardware and software is used to design medium to large data warehouses/marts.	Vertically scalable hardware and software is used to design small to medium data warehouses/marts.
An MPP system uses multiple servers, giving the appearance of a single Appliance. As data grows, an MPP data warehouse scales by adding more hardware nodes vs. replacing a server with a larger server.	An SMP system uses one hardware server; as a result, to scale a data warehouse, a company must buy bigger hardware, which creates long-term costs.
The HP EDW Appliance data rack contains 10 – 40 active compute nodes. Each compute node physically has two processors i.e. 12 cores. Therefore, EDW can scale to 480 physical cores.	SMP systems range from 2 to 64 (or more) cores.
MPP systems contain redundant components. In addition, high availability is achieved as PDW uses Microsoft High Performance Computing (HPC) software to protect compute nodes in the EDW/PDW Appliance. Therefore, high availability may be achieved by using only 1 extra server per data rack (N+1).	Clusters of two or more SMP systems can be used to provide high availability (Microsoft Cluster Service – MSCS).
Each hardware node has its own CPUs, memory, and disks. Therefore, queries face less competition for resources.	Queries compete for common hardware resources (CPU, memory and I/O).
Parallel processing across multiple servers (nodes) AND across multiple cores within each server	Parallel processing only across cores within a single server

The bottom line is that MPP systems provide higher levels of availability and scalability than SMP systems.

Data mart tier– ROLAP

Data marts and data warehouses are similar in that they both store historical operational data in a multi-dimensional, star or snowflake schema. The primary difference is that data marts tend to address the business needs of a smaller niche of business users. Some data marts may be very large; however, when compared to a data warehouse, the data mart's vertical business niche is generally its defining trait rather than its size. On the other hand, data warehouses tend to be more enterprise-wide because they contain data from multiple business units within an organization. These data warehouses are frequently referred to as Enterprise Data Warehouses.

The HP EDW Appliance has been optimized to function as a large data mart, or as an Enterprise Data Warehouse, depending upon the organization's BI architecture.

OLAP cube tier – MOLAP Multi-dimensional Online Analytical Processing

OLAP cubes are very similar to data marts. However, rather than being implemented by using flat SQL tables in a multi-dimensional, star or snowflake schema, the OLAP cube is an n-dimensional structure which, ideally, is stored in memory.

OLAP cubes generally give end users significantly faster response times because the data is summarized and aggregated. They perform best when the frequently accessed cells of the cube are memory-resident. However, note that this more rapid response time does consume hardware resources during cube build and aggregate creation time.

The OLAP cube tier uses SQL Server Analysis Services, which have data mining and analytical enhancements, making it easier for the business analyst to use the data to make decisions. Some of these operations include: slice, dice, drill-up, drill-down, roll-up, pivot, etc. OLAP cubes also allow for hierarchies, which are more difficult to describe using traditional, ROLAP, and SQL tables.

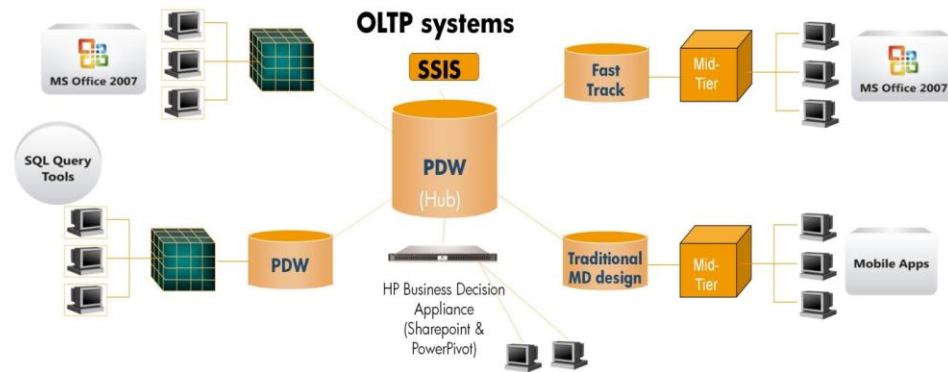
The EDW Appliance also provides easy connectivity between SQL Server Analysis Services and the Appliance.

End users – portals, data access and analysis tools

Business analysts typically have access to graphical, user friendly tools to perform analytics, data mining, or simply generate reports and execute queries. Executive dashboards are also commonplace to provide flexible visual access to the information in the data warehouse, data mart or OLAP cube. Examples of end user tools which may be used to query data are: Nexus, Microsoft PerformancePoint, Excel, and Reporting Services, among others.

It should be noted that customers can deploy their own Microsoft SharePoint and PowerPivot environment or HP may make SharePoint and PowerPivot implementations simpler by offering customers the HP Business Decision Appliance. The Business Decision Appliance is pre-configured, tuned and can connect to EDW, Fast Track or traditional data marts. Figure 2 shows the relationship of the Business Decision Appliance running SharePoint and PowerPivot.

Figure 2. HP Business Decision Appliance



Optimization of hardware for data warehousing or OLTP

Unlike transaction processing, where transactions are well defined and system demand is quantifiable because of a pre-determined number of I/Os per transaction, business intelligence queries are at the opposite end of the spectrum. Table 2 shows some of the major differences between OLTP and data warehouse/data mart workloads.

Table 2. Data warehouse / data mart versus OLTP workload characteristics

Characteristic	Typical BI (DW's & DM's)	OLTP (Operational Database)
Database design	Typically multi-dimensional, star or snowflake schema, although some EDW databases may have some degree of normalization	3 rd or 4 th normal form
Data activity	Large reads (disjoint sequential scans, joins, sorts, group-by, aggregation, etc.) Large writes (new data appends) Indexed reads and writes Large scale hashing	Small transactions Constant small number of index reads, writes, and updates
Database sweet spot size	100s of gigabytes to hundreds of terabytes (need medium to large storage farms)	Gigabytes (require smaller to medium sized storage farms) Small databases
Time period	Historical (contributes to large data volumes)	Current
Queries	Largely unpredictable (ad-hoc)	Predictable
I/O throughput requirement	Sustained throughput of 10s of GB/sec (12-15GB/data rack)	IOPs is more important than sustained throughput

It is difficult to design hardware which will support both BI and OLTP workloads. Users who try to execute these dissimilar workloads (with very different user service level agreements) on "one size fits all" hardware are likely to find that short, fast running OLTP transactions will suffer when competing

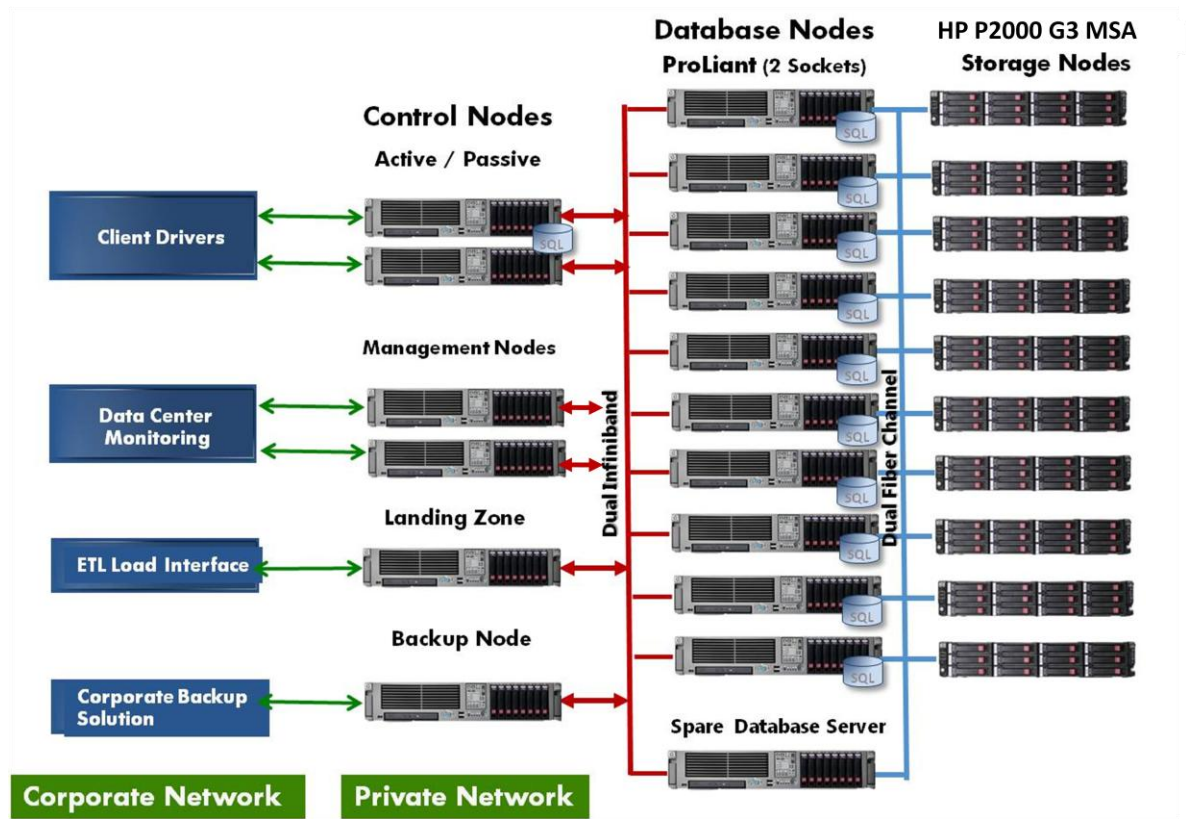
for resources against long-running BI queries that stream large blocks of data off the disks as quickly as possible.

It is physically simpler to manage, and run OLTP and BI workloads on different servers as opposed to burdening DBAs and operations staff with trying to fine tune and isolate OLTP and BI mixed workloads on a system with shared servers and a shared I/O subsystem.

The EDW Appliance has been specifically optimized, balanced and tuned for medium to large scale data warehouses or data marts. In addition, the hardware has been balanced using a shared nothing (MPP) architecture which utilizes the EDW resources optimally when executing BI queries, load and backup/restore operations.

HP Enterprise Data Warehouse architecture

Figure 3. HP EDW optimized for SQL server 2008 Parallel Data Warehouse



While canned reports may be generated by executing predefined queries, it is frequently observed that upper management, business analysts and users do not want to look through stacks of paper or online PDF files to make business decisions. In recent years, business analysts demand their data warehouses and data marts to be able to process ad-hoc queries quickly, with more uniform and consistent response times.

A company's IT department may size and configure a system to handle a certain query workload. But, due to the changing nature of the business environment, end user query workload characteristics and volume change over time. Yet, the IT department needs to maintain their service level agreements.

While traditional database designs using SMP hardware allow users to scale-up to a certain point, more IT departments have come to realize that they want systems which can scale to much higher levels than a single SMP system can offer. Therefore, HP and Microsoft have engineered the HP EDW Appliance to be optimized for SQL Server 2008 PDW software. Figure 3 shows the architecture for optimization.

The HP EDW Appliance allows customers to optimize their valuable data center's floor space by using highly dense servers and storage. As stated above, the HP EDW can scale from 10 to 40 multi-core servers using footprint of only 1-4 data racks.

In an effort to simplify sizing and the customer's purchase experience, the SQL Server 2008 PDW software is included and pre-installed on the HP EDW Appliance. Customization options have been intentionally minimized by limiting selections to disk size and power connectivity issues.

NOTE:

The EDW hardware described in this document are valid as of the publication date. Please check the latest EDW Quickspecs for the most recent description of EDW components.

<http://h10010.www1.hp.com/wwpc/pscmisc/vac/us/en/sm/solutions/enterprise-overview.html>

In summary, the available options are:

- Size and number of disk drives in each data rack
 - 240* 300GB SFF (small form factor)
 - 110* 300GB LFF (large form factor)
 - 110* 1TB LFF (large form factor) disks

The table below estimates the expected amount of available storage per data rack assuming a 3.5x compression ratio:

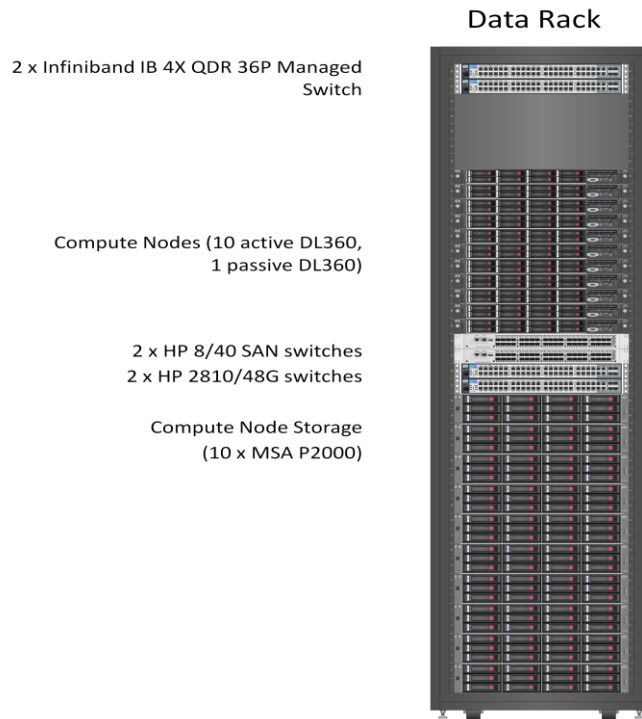
User Data Capacity	Number of Data Racks Ordered			
	1	2	3	4
300GB SFF Disks	76TB	152TB	228TB	304TB
300GB LFF Disks	38TB	76TB	114TB	152TB
1TB LFF Disks	127TB	254TB	381TB	508TB

- Number of data racks: Customers may choose one up to four data racks
- PDU type (single phase/non fault tolerant , triple phase/fault tolerant)
- Adding a stand-alone Test/Dev system (only orderable by customers that have purchased a full EDW system)

In order to understand the EDW Appliance's massively parallel architecture we need to define the functions of each server in the data rack and control rack. An EDW system will always have a single control rack and a total of 1 to 4 data racks. See Figure 4.

EDW data rack

Figure 4. HP EDW data rack



The EDW data rack (HP Universal Rack 10642 G2 Shock Rack) is where the physical data for the data warehouse or data mart is stored. Each component performs specific functions.

Storage node

The storage options per data rack consist of fact table(s), dimension tables, etc. The data warehouse or data mart is stored on these P2000/MSA arrays. By balancing the data across the MSAs, the PDW software is able to access the data in parallel in order to achieve high levels of I/O throughput.

As shown in Table 3, all MSAs and disk sizes will be the same in the data racks.

Table 3. Data rack MSA and disk sizes

Disk size	MSA
240 * HP 300GB 6G SAS 10K 2.5in DP ENT HDD (22 * RAID1 disks + 2 spare per MSA)	10 * HP P2000 G3 MSA FC Dual Cntrl SFF Array
110 * HP P2000 300GB 6G SAS 15K 3.5in ENT HDD (10 * RAID1 disks + 1 spare per MSA)	10 * HP P2000 G3 MSA FC dual Cntrl LFF Array
110 * HP P2000 1TB 6G SAS 7.2K 3.5in MDL HDD (10 * RAID1 disks + 1 spare per MSA)	10 * HP P2000 G3 MSA FC dual Cntrl LFF Array

Database server node

Each MSA is logically associated with a database server node. See Figure 3. Therefore, each data rack contains 10 ProLiant DL360 G7s plus 1 spare DL360 G7. Therefore, there are a total of 11 * DL360 G7s per data rack.

Figure 5. DL360 G7



DL360 G7 (2x X5670) * 11 per data rack

- 96GB RAM (12x HP 8GB 2Rx4 PC3-10600R-9)
- HP DL360 SFF HD Backplane
- HP 512MB P-Series BBWC Upgrade
- 8x HP 300GB 6G SAS 10K 2.5in DP ENT HDD
- HP InfiniBand (IB) 4X DDR PCI-e DUAL PORT 0 Memory HCA
- 2x HP 460W HE 12V Hotplug Power Supplies
- HP iLO Advance license

Tempdb performance

It is also important to note that each database server node has 6 disks dedicated to tempdb operations and 2 disks for the Microsoft Windows® operating system. Therefore, each data rack has 60 active tempdb disks to provide excellent tempdb performance.

EDW network traffic (data rack)

The PDW software architecture not only allows for simultaneous query execution, but, more importantly, each individual query also has the capability to execute on all the servers and access all the MSAs in each data rack in parallel. It is this massively parallel architecture which gives the EDW/PDW its speed to provide user queries with significantly higher levels of I/O throughput than that which may be achieved on a scale-up SMP system.

The servers communicate to each other using these redundant switches.

- Two * HP Switch 2810-48 G
- Two * IB 4X QDR 36P Managed Switch

The storage nodes and database server nodes are connected via

- Two * HP 8/40 SAN Switch 8Gb 8-port Upgr LTU

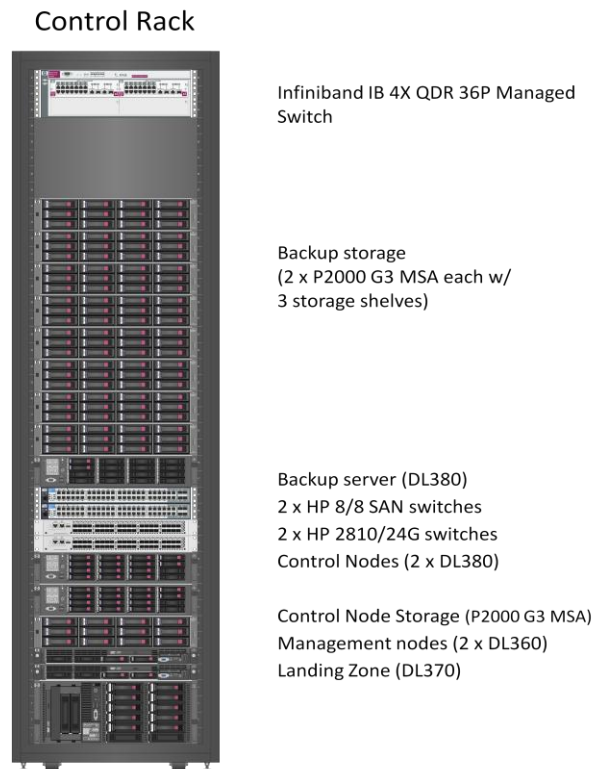
Compute nodes and high availability

The combination of a database server node and its logically associated storage node is called a compute node. Each database server node is configured using RAID1 (mirrored) arrays. In addition, the node has either one Large Form Factor (LFF) spare disk or two Small Form Factor (SFF) spare disks per MSA in the event of failure.

The database server nodes are configured using Microsoft High Performance Computing (HPC) cluster software which provides high availability by using an “N+1” failover technology. By configuring each data rack with 11 physical database servers and running with 10 active servers, the EDW provides the data rack with high availability features.

EDW control rack

Figure 6. HP EDW control rack



Each EDW Appliance contains one control rack (HP Universal Rack 10642 G2 Shock Rack) with multiple servers which are responsible for various operations.

Backup server node

As the name implies, the backup Server node is used to back up the data in the data warehouse. See Figure 7.

Figure 7. Backup server (ProLiant DL380 G7)



The backup server contains the following components.

DL380 G7 (2x E5620)

- 24GB RAM (6x HP 4GB 2Rx4 PC3-10600R-9)
- HP 256MB P-Series cache upgrade
- 2x HP 146GB 6G SAS 10K 2.5in DP ENT HDD – 146GB RAID1
- HP 82E 8Gb Dual- Port PCI-E FC HBA
- HP IB 4X DDR PCI-E Dual Port 0 Memory HCA
- 2x HP 460W HE 12V Hotplug power supplies

The backup server storage system size is determined by the size of disks and the number of data racks in the EDW Appliance.

Backup storage (supports 131TB compressed backup)

- 2 * HP P2000 G3 MSA FC Dual control LFF array
- 2 * HP P2000 Dual I/O LFF drive enclosure
- 48 * HP P2000 1TB 6G SAS 7.2K 3.5in MDL HDD

Backup storage (supports 262TB compressed backup)

- 2 * HP P2000 G3 MSA FC Dual control LFF array
- 6 * HP P2000 Dual I/O LFF drive enclosure
- 96 * HP P2000 1TB 6G SAS 7.2K 3.5in MDL HDD

Backup storage (supports 523TB compressed backup)

- 2 * HP P2000 G3 MSA FC Dual control LFF array
- 6 * HP P2000 Dual I/O LFF drive enclosure
- 96 * HP P2000 2TB 6G SAS 7.2K 3.5in MDL HDD

Management server nodes -- EDW domain controllers

The management server is the way that DBAs or data center operations access and manage the EDW Appliance. In addition, it acts internally as the domain controllers for all of the servers in the EDW Appliance.

Figure 8. The management server (ProLiant DL360 G7)



Each management server contains the following components.

- DL360 G7 (2 x E5620)
- 36GB RAM (9x HP 4GB 2Rx4 PC3-10600R-9)
- 2x 300GB 6G SAS 10K RPM 2.5" HDD – 300GB RAID1
- 256MB P-series Cache Upgrade with 650mAh battery
- HP IB 4x DDR PCI-e Dual Port 0 mem HCA
- 2x 460W HE 12V Hotplug Power Supplies
- HP iLO Advance license

Landing zone server

When loading data into a data warehouse or data mart, it is common to have the cleansed data stored and ready for loading. The EDW Appliance has a dedicated landing zone server to perform these operations. See Figure 9. Some customers may have an external server running ETL software and transfer the cleaned files to the landing zone to be loaded into the PDW database.

Other customers may want to eliminate the need for an external server running ETL software. This may be accomplished by having their OLTP systems transmit data directly to ETL software (such as SQL Server Integration Services) running on the landing zone server. At this point, SQL Server Integration Services may then load the cleansed data directly into the EDW database or write to staging tables for bulk load.

The landing zone is the only node which is configured with an optical drive because it is the only node where customers are permitted to install and run non-PDW software.

Figure 9. The landing zone server



The landing zone contains the following components.

DL370 G6 (2x X5690)

- 36GB RAM (6x 2GB 2Rx8 PC3-10600R-9, 6x HP 4GB 2Rx4 PC3-10600R-9)
- HP ML/DL370G6 6 LFF backplane
- HP SAS expander card
- 10x HP 1TB 6G SAS 7.2K LFF Dual Port Midline HDD - 9TB RAID5
- 2x HP 160GB 3G SATA 7.2K LFF Entry HDD – 160GB RAID1
- HP 512MB P-Series BBWC upgrade
- HP Slim 12.7mm SATA DVDRW optical kit

Control server nodes (active/passive) cluster

The primary function of the control node is to accept queries from users. These queries may enter the EDW Appliance directly (i.e. via Excel, PowerPivot, etc.) or, at other times, there may be an application tier or OLAP cube (SQL Server Analysis Services) which submits queries on behalf of users. The control server runs as an active/passive MSCS cluster in order to provide users with a high availability environment. See Figure 10.

Figure 10. Control server (ProLiant DL380 G7)



Each control node contains the following components.

DL380 G7 (2x X5680)

- 96GB RAM (12x HP 8GB 2Rx4 PC3-10600R-9)
- HP SAS expander card
- HP 8 SFF cage
- 14x 300GB 6G SAS 10K 2.5in DP HDD – 300GB RAID1 & 3300GB RAID5
- HP 512MB P-Series BBWC upgrade
- HP 82E 8Gb Dual-port PCI-E FC HBA
- 2x 750W CS HE power supplies

In addition, the control node has shared storage, consisting of:

- HP P2000 G3 MSA
- 5 * 450GB 6G SAS 15K 3.5in HDD – 1800GB RAID5

EDW -- data rack network

The EDW Appliance uses both InfiniBand and Ethernet for inter-nodal communication. In addition, Ethernet ports are also available for external connectivity to the EDW Appliance. The customer needs to provide seven external cables to connect EDW to the customer's network.

The servers communicate to each other using these redundant switches:

- 2 * HP Switch 2810-48 G
- 2 * IB 4X QDR 36P Managed Switch

Storage nodes and database server nodes are connected via two HP 8/40 SAN Switches.

Parallel Database Export – Hub and spoke support

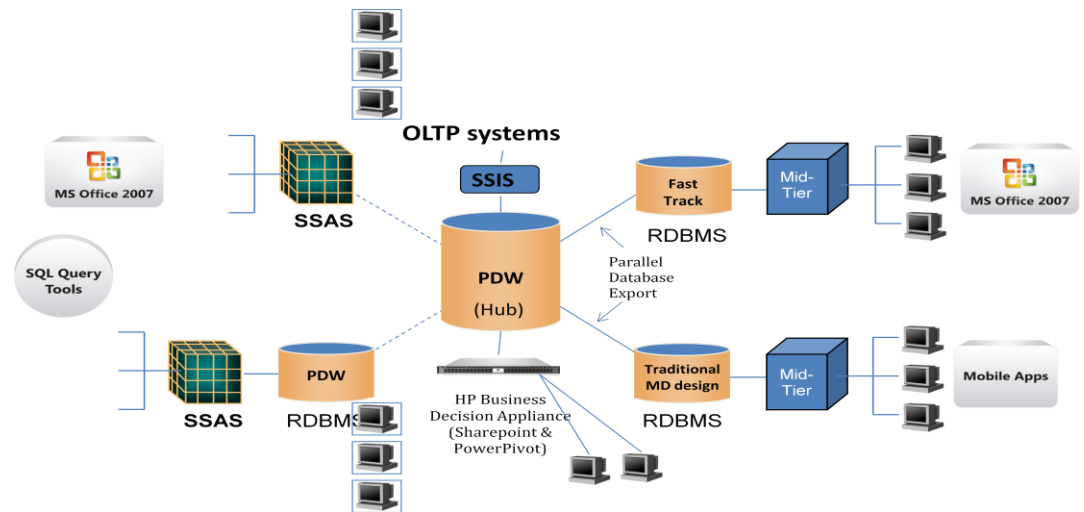
The EDW Appliance has a useful feature which allows for high speed data transfer to external servers and databases (spokes) in a business intelligence environment where a “hub and spoke” architecture is deployed. This is accomplished by allowing external servers to connect to the EDW's InfiniBand network while running Parallel Database Export software. See Figure 11.

Many customers find that a hub and spoke infrastructure is desirable if users want to have their own data marts which extract data from the HP EDW Appliance. In addition, these external servers are sometimes used for complex data mining operations.

Data marts are commonly implemented to support different business functions, geographic locations, classes of users, etc.

The EDW Appliance may effectively support “hub” functionality or may serve as a large “spoke” data mart.

Figure 11. Hub and spoke architecture



How EDW achieves high throughput

This section of the technical white paper will discuss how hardware, software and a database design work together as a triad to provide the HP EDW Appliance with highly optimized levels of I/O throughput.

Why I/O throughput is important for data warehouses and data marts

In the “Optimization of hardware for data warehousing or OLTP” section of this paper, we discussed the differences in workload characteristics between OLTP and BI databases. If your organization runs canned reports, at first glance it may appear that the use of indexes will improve performance by trying to directly access specific groups of rows which are required to generate the report.

However, users today realize that they do not want to look through paper reports or large PDF files to find their answer or make a business decision. They prefer to make “ad-hoc” queries which will allow the data warehouse or data mart to return the specific result set they want to answer their question. In addition, the nature of the user and management questions will change weekly, daily or hourly.

In this dynamic environment, indexes may present a double edged sword.

Advantages of using indexes

- Tends to reduce the number of I/Os to service a set of known queries/reports
- If you are lucky, an ad-hoc query may find the data it needs in an index

Disadvantages of using indexes

- Users will get inconsistent response times if indexes are used. If you have an index-heavy design, a query may be fortunate; however, if the optimizer does not recognize that an index may be used, then tables will need to be scanned. Scan rates tend to be slower on the index-heavy system than on index-lite designs because of disk fragmentation and extra seek time required.
- Slower load times if indexes are maintained during load/insert operations
- Longer batch ETL windows if indexes need to be rebuilt
- Extra disk space used (indexes may require 4 – 10 times more space)
- More DBAs may be required to spend time analyzing frequently executed queries/reports and to constantly create new indexes. They also tend to be wary of deleting indexes because trying to understand their effectiveness is time-consuming and may be difficult. For these reasons, index – heavy designs tend to get heavier over time.
- Indexes tend to force random physical disk I/O (extra seek time). Seek time slows down disk I/O throughput, resulting in slower scan rates.

Note

Since seek time is the slowest disk operation, the EDW Appliance tries to avoid disk seek time whenever possible. The goal is to minimize or eliminate disk seek time, allowing the EDW Appliance to support very high scan (throughput) rates. Index-lite designs coupled with PDW best practice loading techniques allow the EDW Appliance to provide customers with excellent performance, especially for ad-hoc queries.

Traditional database designs vs. EDW and Fast Track

This section of the technical white paper will discuss why EDW and Fast Track “Index-lite” database designs tend to provide users with rapid and more consistent response times than traditional database designs. This section will also briefly address how Fast Track and EDW are related.

Traditional (index-heavy) database designs

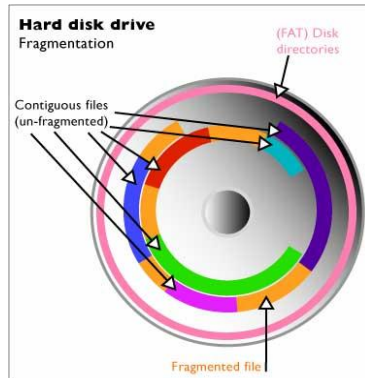
Traditional data warehouse/data mart database designs use fact tables, dimension tables, summary tables and *indexes* in an effort to minimize I/Os when servicing queries.

When loading data, database tables are typically striped and balanced across one or more LUNs and file groups. In turn, these filegroups are spread across multiple physical disks. Unfortunately, little attention is paid to the physical location of where the data is loaded and where indexes are maintained.

Traditional database load procedures

Typically, most customers try and load multiple fact tables and dimension tables simultaneously. Simultaneous load execution, dynamic index updates or index rebuilds tend to result in data being physically fragmented on the disk. See Figure 12.

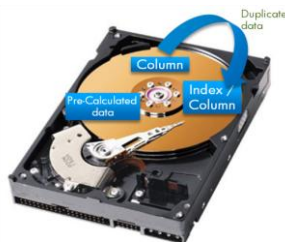
Figure 12. Hard drive fragmentation



Traditional (Index-heavy) issues

- ETL batch windows may take a long time to complete because dropping and re-building indexes can be resource and time intensive
- If you insert/update indexes as data is being loaded vs. performing a drop/rebuild, then the time to load data while maintaining indexes will be slow. In either case, index-heavy design load times will affect batch load and/or trickle update ETL windows.
- Indexes (duplicate data) consume a large amount of disk space (4x – 10x more space)
- DBAs spend a lot of time managing and tuning indexes
- Indexes may reduce the number of disk I/Os to service a query, although at a cost of slower disk service times due to extra disk head movement.

Figure 13. Index heavy design encourages disk head movement which slows down throughput



This extra disk head movement is typically due to inefficient data and index placement on the physical disk media. But, more importantly, index usage forces disk head movement that will slow down table scan operations that typically occur for ad-hoc queries.

Excessive disk head movement (seek times) can result in at least 2 – 4 times longer disk access times than expected. Therefore, significantly more disk drives will need to be purchased to support a database with tables and indexes that were loaded in a sub-optimal manner.

Traditional database design conclusions

Traditional database designs, load procedures and maintenance issues are likely to provide slower query response times (scan rates) and index-heavy designs will be more difficult for DBAs to manage.

In addition, there is likely to be a wide variance between canned query throughputs, which were tuned by managing indexes versus slow scan rates which may be common for ad-hoc queries.

EDW and Fast Track “index-lite” database designs

This section of the technical white paper will discuss how EDW and Fast Track design philosophies are similar. In addition, this section will address the advantages of EDW and Fast Track “index-lite” designs versus “index-heavy” designs.

EDW and Fast Track Index-lite design

EDW, Fast Track and traditional data warehouse/data mart database designs typically use multi-dimensional, star-schema or snow-flake schema designs. These are all variations on a similar database design philosophy that include the use fact tables and dimension tables. Sometimes Enterprise Data Warehouse and Operational Data Store (ODS) implementations normalize data structures to a higher degree. The EDW Appliance supports both multi-dimensional and more normalized schemas.

EDW and Fast Track load procedures

As mentioned earlier, most executives, business analysts and BI users today do not necessarily know what questions (queries) they will ask ahead of time. Users also tend to get frustrated if they find that query response times are wildly inconsistent. EDW and Fast Track are both optimized to relatively consistent performance by optimizing the system to maintain extremely high scan rates.

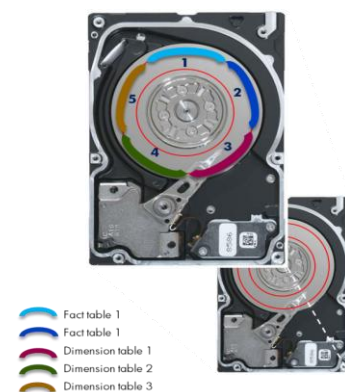
EDW and Fast Track are efficient “scan rate engines” which provide excellent price for value performance. This is accomplished by using some of the following best practices techniques.

Initial load best practices

- Create filegroups and “pre-define” filegroup storage space
- Allocate extents sequentially in the filegroups to store data sequentially on disk by allocating extents sequentially during the initial load

Since the goal of both EDW and Fast Track is to physically store data sequentially, data may be streamed at high rates of speed off the disks because there is little or no disk head movement to seek and find data.

Figure 14. Example of how data may be stored sequentially for optimal performance



Subsequent loads

- Subsequent loads will tend to have data physically clustered on disk in their respective filegroups

How data can become fragmented over time

- Loading data into multiple fact tables may cause a slight increase in disk latency and/or seek time
- Changes to dimension tables after the initial load may also cause some fragmentation. However, in most data warehouses and data marts, dimension tables are relatively static.

Figure 15. Example of how data may be fragmented over time



Note

Similar colors represent the same table. Notice how all similar colors are clustered in large blocks, even though they are not sequential. For example, this type of large block fragmentation may be clustered date ranges; therefore query performance should not be significantly affected because most queries have a date range as part of the predicate.

- This clustering of fact table data across the disk is typically not a problem because most queries access the most recent data which tends to be clustered in groups (e.g., users may query what happened last week).
- If a query accesses multiple date ranges, each date range will likely be clustered together on disk. Therefore, disk head movement will be minimal. Even though seek time will not be completely eliminated; scan rates will still be excellent.

How to make data sequential again (defragment) for optimal scan rates

- To re-org, perform a CTAS (CREATE TABLE as SELECT) operation
OR
- Backup and restore

EDW and Fast Track (Index-lite) for optimal performance

When designing a data warehouse or data mart for EDW or Fast Track, a best practice is to try and eliminate as many indexes as possible.

There are three main reasons why index-lite improves query performance and I/O throughput rates.

- Index-lite minimizes random I/O and extra seek time on the physical disks
- Reducing or eliminating indexes uses disk space more efficiently. In addition, seeks times are improved significantly.

When SQL Server uses the index, it tries to minimize disk I/O. But, those I/Os tend to force random I/O. Which, in turn, causes extra seek time. This extra seek time slows down the I/O throughput rate on the disk drive. Therefore, minimizing indexes will improve EDW performance.

- Index-heavy designs trade off fewer I/Os, index maintenance and creation time versus Fast Track and EDW index-lite designs, which encourage ultra-fast scan rates coupled with an Ultra-shared nothing implementation. In addition, EDW and Fast Track will provide users with more consistent response times which tend to make for less frustrated users when they execute ad-hoc queries that may vary as frequently as the business environment evolves.

EDW and Fast Track index-lite design conclusions

Try to eliminate all indexes from EDW and Fast Track database designs. The concept is to stream data as fast as possible because fast table scans will service most queries faster than performing fewer, slower random disk I/Os.

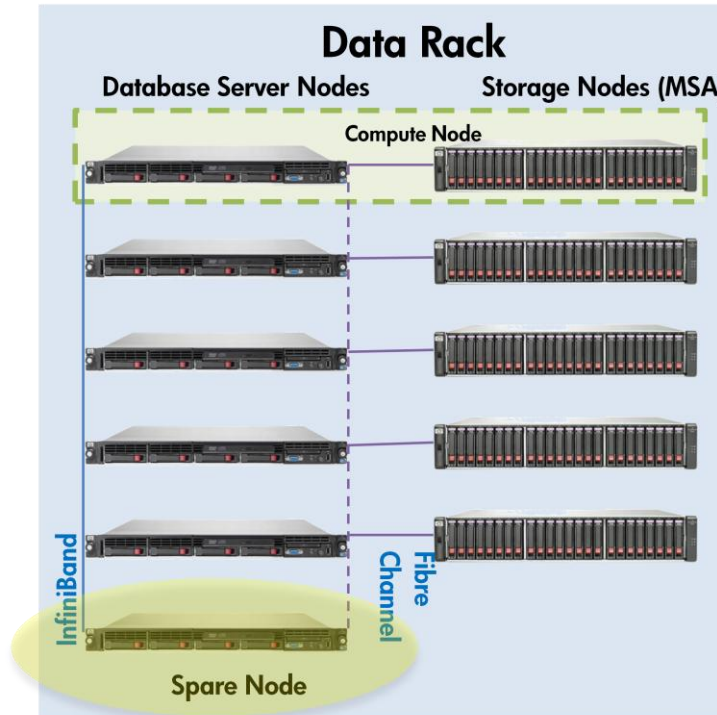
EDW and Ultra-Shared Nothing technology

Some parallel database implementations on the market share storage. The EDW Appliance has been intentionally designed as a true, InfiniBand-based, loosely-coupled architecture which does not share storage. Ethernet is used for EDW connection to the customer's network and for some internal command and control functionality.

From a performance point of view, it is important to note that shared nothing MPP implementations are more efficient than shared storage architectures. A major reason for the added efficiency is because all of the disks in a shared storage environment get bombarded with requests from all the database servers requesting data. The more database servers in the database layer, the more pressure is exerted on the shared storage subsystem. In addition, all of these simultaneous requests result in the disks performing more random I/O. Hence, shared disk architectures tend to not stream data as fast as systems which do not share storage when multiple queries are executing simultaneously.

Loosely-coupled, shared nothing (MPP) architectures like the EDW Appliance have storage subsystems that are separated and isolated. These isolated units are called compute nodes. Each compute nodes contains a database node and a storage node. See Figure 16. This loosely coupled, compute node concept allows EDW to stream data off of disk faster due to less disk contention and less disk head movement than shared storage systems can provide.

Figure 16. EDW Appliance is a loosely coupled, shared nothing, MPP architecture



Note

Each compute node does more than simple predicate evaluation. Each node has a fully functional copy of SQL Server 2008, which allows the compute node to execute most SQL operations as close to the physical data as possible. In addition, each compute node is aligned with its own storage. This non-shared storage design avoids the contention that shared storage subsystems encounter.

Database servers send data to each other, via InfiniBand, only when needed.

The loosely coupled architecture of the EDW Appliance allows it to support and encourage the concept of “Ultra-Shared Nothing.” Thus, the following benefits are integrated into the process:

- Control node decomposes the SQL query to execute in parallel (using Distributed SQL – “DSQL”)
 - Multiple physical instances of tables
 - Replicate small tables
 - Distribute large tables
 - Data Movement Service (DMS) - Redistributes rows “on-the-fly” for loads or queries
- Fault tolerance
 - Hardware components have redundancy
 - CPUs, disks, networks, power, and storage processors
 - Control node uses failover clustering
 - Management nodes and compute nodes are part of a single cluster

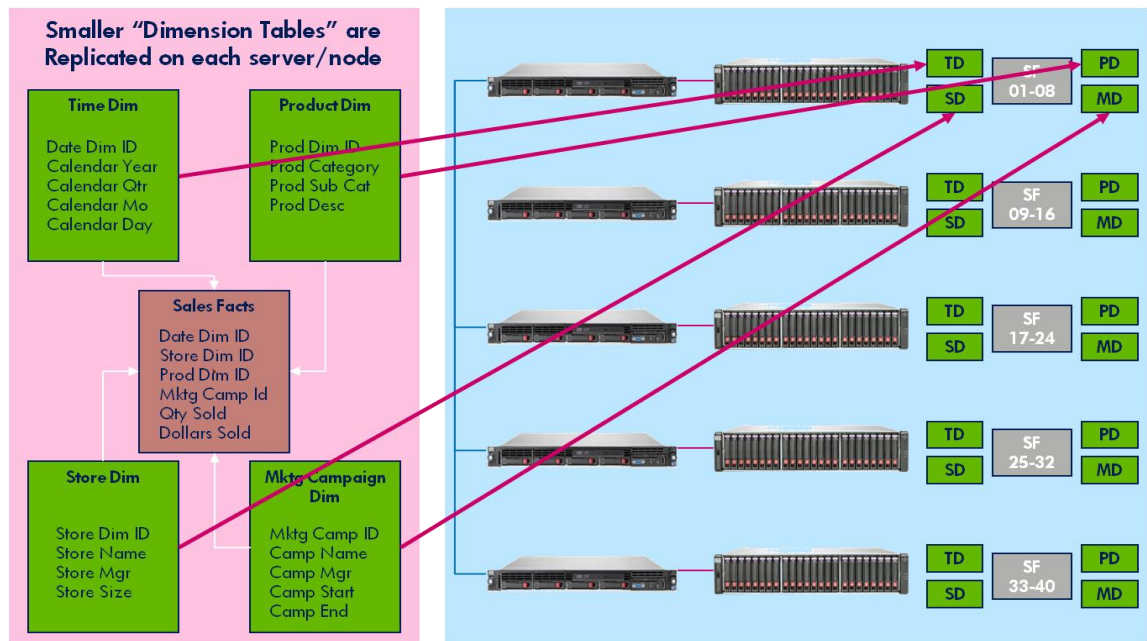
The most important thing to understand about Ultra-Shared Nothing technology is when to use distributed or replicated tables. The concept of distributed or replicated tables tends to be easiest to explain using a multi-dimensional database designs. The concepts work equally as well when implementing a more normalized schema which is sometimes found in operational data stores or in some EDW.

Ultra-Shared Nothing – replicated tables

In order to minimize inter-nodal traffic for join operations or dimension table look-ups, a best practice for EDW Appliance is to define small tables (typically dimension tables), to be replicated on each compute node. In other words, each compute node has its own copy of the replicated table on disk. It is also likely that frequently accessed replicated tables are cached for rapid response time.

This replication process is transaction-protected and automated, so the operations staff or DBAs do not have to manually “replicate” tables across 10-40 servers. The PDW software automatically assumes that tables will be replicated (default) unless the table was specifically defined as a distributed table.

Figure 17. The replication process



NOTE: Replication results in less “inter-nodal” traffic for joins, etc.

Ultra-Shared Nothing – distributed tables

As shown in Figure 17, replicated tables allow for very efficient join operations, especially star joins, because the dimension tables will always be local on each compute node, thus minimizing or eliminating dimension table inter-nodal traffic. Ultra-Shared Nothing replicated tables are not available in “shared disk” architecture. Hence EDW Ultra-Shared Nothing design techniques will place less of a demand on the InfiniBand network and many join operations will be more efficient.

On the other hand, fact tables can contain billions of rows, so they are not practical to replicate. Therefore, fact tables are typically “distributed.”

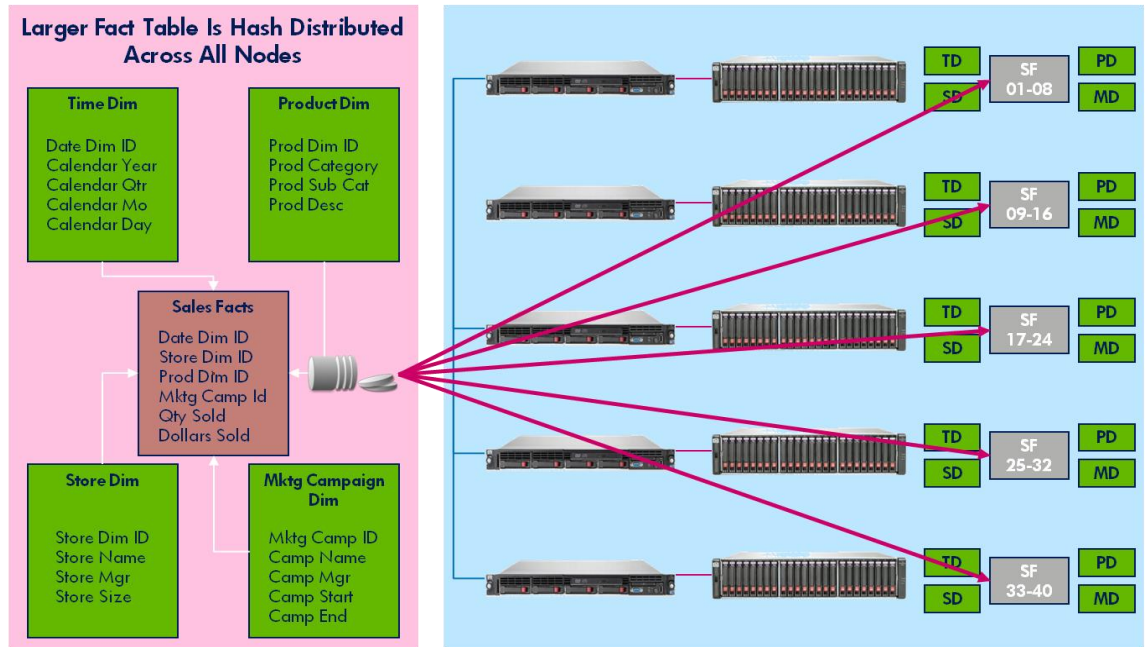
Distribution is accomplished by defining a “distribute” key column. This key column subsequently hashed so the large (fact table) data will be distributed across all of the compute nodes in the EDW.

The goal of the distribution (hash) column is to evenly distribute data across all the compute nodes in the data racks. See Figure 18.

Note

One data rack has 10 active compute nodes and four data racks contain 40 active compute nodes.

Figure 18. Distributed table



Distributing data evenly allows table scans to execute efficiently, in parallel, in order to provide users with rapid response times.

Other PDW software benefits

Partitioning distributed data

In addition to distributing data, PDW software also allows for “partitioning” and “clustering.” These features allow DBAs and operations to manage large quantities of data more effectively.

A good example would be to partition data by date range. As the data warehouse grows, archiving old historical partitions which may be dropped after it is archived.

Loading data into the EDW

Data loaded into the EDW Appliance must be stored on the Landing Zone node before it gets loaded into the PDW database. Once data is on the landing zone, it may be loaded using “dwloader” which loads data in parallel for maximum speed.

It is also possible to have application software, such as SQL Server Integration Services (SSIS) perform direct insert/update or delete operations to the PDW data warehouse/mart. These insert/update or delete operations may also be transaction protected.

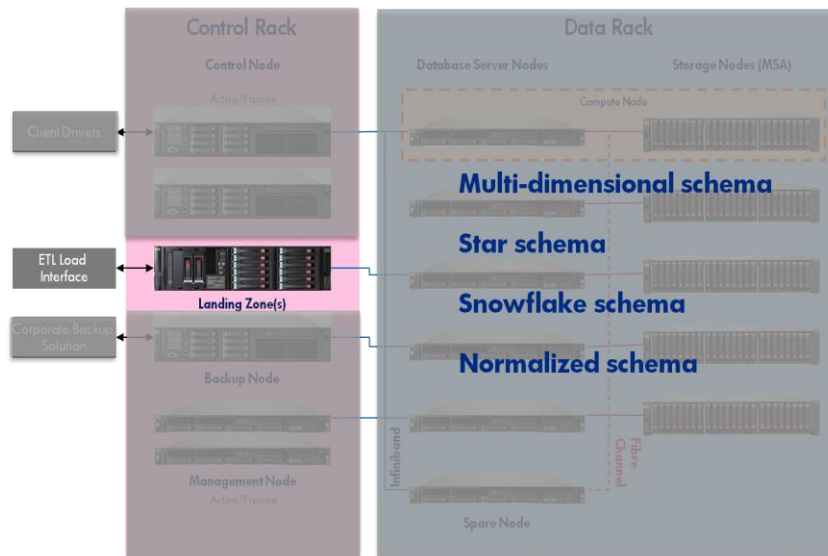
External ETL tier

Some customers may already have an existing BI infrastructure which already has ETL software on a dedicated server. Therefore, input data may have been already remapped, cleansed and formatted by the ETL software running on the external ETL server. In this case, it is likely that the landing zone functions as a true “landing zone” which houses data to be loaded into the EDW Appliance compute nodes.

EDW/PDW internal ETL tier

Customers also have the option to have their OLTP systems or other external feeds store raw data directly on the Landing Zone. Then, SQL Server Integration Services (ETL software) may remap, cleanse and reformat data to be loaded into the EDW compute nodes.

Figure 19. Landing zone is used to load data into various schemas



The landing zone is physically a ProLiant DL370 G6 (2x X5690) with 36 GB of memory and 10 * 1 TB disks (RAID5), which may be used for storage. If your ETL workload requires more resources than what is provided on the EDW Landing Zone, an external ETL tier would be desirable to run ETL software.

Note

The landing zone is the only server on which the customer is allowed to install application code. The server has a CD/DVD and USB for software installation.

How loading data can affect EDW/PDW performance

Earlier in this document we discussed how PDW table replication and distribution features are leveraged to optimize query response time by performing more efficient join operations and executing faster (parallel) scan rates.

In addition, PDW software also provides the option of using “staging tables” to automatically optimize how data is physically stored on disk.

When loading data you simply have to specify the “staging” database in the load command and “dataloader” will automatically optimize load operations by insuring that data in the target PDW data warehouse is stored sequentially on disk.

Sequential storage of your data warehouse data on disk and efficiently organized SQL Server block structures allow replicated and distributed tables to perform more efficiently. Scan operations are faster and more data is retrieved per data block due to these load optimization features. In addition, PDW data is automatically compressed to further enhance the system’s effective throughput. Compression allows more rows in each block to be retrieved from the physical disk.

Bear in mind that the default for PDW is to compress rows in the database. In addition, PDW’s column compression algorithms allow:

- Data to be compressed during bulk load OR during trickle insert/update/delete operations. This means that EDW performance is maintained in real time operational data stores or data warehouses. Queries can also execute while loads, inserts, updates or deletes are taking place.
- All the rows in a block to be decompressed once the block is read in memory. This is an advantage of compression algorithms which may require multiple compression groups to be read to reassemble rows.
- Row level compression to use CPU resources efficiently, freeing up CPU time for other useful work.

Reorganizing EDW/PDW tables

Occasionally, data may become fragmented. If this occurs, it may be remedied by performing a CTAS (Create Table As Select) operation. This task will restructure the target table to allow for more efficient sequential scan performance.

EDW performance metrics

HP and Microsoft have run various performance tests internally and with customers. The following are a few reasonable performance metrics which may be used when evaluating EDW Appliance throughput.

Since the EDW is a loosely coupled MPP architecture, system scan rates are expected to perform in a linear fashion. Therefore, it should be expected that a 4 rack EDW can scan at about 66GB/second.

Table 4 shows what is considered to be reasonable performance metrics for each data rack.

Table 4. One data rack performance metrics

EDW Performance Metrics	Small Form Factor disks per data rack	Large Form Factor disks (15K) per data rack
Load rates	1.7 TB/Hr	1.1 TB/Hr
Scan rates compressed (raw disk, per rack)	16.5 GB/sec	12.8 GB/sec
Backup rates (to disk)	5.4 TB/Hr	5.1 TB/Hr

Summary

This technical white paper discusses the HP Enterprise Data Warehouse Appliance architecture, functional components, PDW database design differentiators and expected performance metrics. Please feel free to contact HP or Microsoft to provide more in depth insight relating to EDW best practices to implement a successful and efficient foundation for your business intelligence environment.

For more information

EDW solutions page: http://h71028.www7.hp.com/enterprise/us/en/partners/microsoft-enterprise-data-warehouse-solution.html?jumpid=ex_r2858_us/en/large/tsg/microsoft_edw

EDW product page: <http://h10010.www1.hp.com/wwpc/pscmisc/vac/us/en/sm/solutions/enterprise-overview.html>

HP Business Intelligence Solutions for Microsoft SQL Server:
<http://www.hp.com/solutions/microsoft/sqlbi>

HP Business Decision Appliance Overview (Useful for a data mart spoke, in a hub and spoke BI architecture or a standalone data mart):
<http://h10010.www1.hp.com/wwpc/pscmisc/vac/us/en/sm/solutions/business-overview.html>

ActiveAnswers page: www.hp.com/solutions/activeanswers/microsoft/sql

To help us improve our documents, please provide feedback at
http://h20219.www2.hp.com/ActiveAnswers/us/en/solutions/technical_tools_feedback.html.



© Copyright 2011 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft and Windows are U.S. registered trademarks of Microsoft Corporation in the U.S. and other countries.

4AA3-5625ENW, Created July 2011

