Architecture Considerations for Production Environments Incorporating IBM *e*server® pSeries® and System p5™ Servers

March 3, 2006

Table of Contents

Table of Figures

Introduction

Abstract

Because of the breadth and depth of the IBM @server® pSeries® and System p5™ hardware product lines and related software products offered by IBM and other vendors, clients often face a daunting task in identifying the right set of System p5 hardware and software to meet their business needs. As a result, clients can experience uptime challenges with their System p5 environments (e.g. maintenance (applying fixes non-disruptively, knowing which fixes are appropriate/safe) and configuration for greatest uptime. This document describes a sample architecture for a pSeries production environment. While cost control must (as always) be considered, the assumption made here is that the audience for this document is a larger customer who is basing their IT infrastructure on pSeries. Resiliency of the environment is important but stops short of being characterized as highly available. In some quarters, this is thought of as "I'm not running a bank"-level availability. For example, this document will not discuss the use of high availability clustering such as the IBM HACMP™ product. Instead, it will illustrate how redundancy and virtualization in the server, network and storage layers help protect against failures in single components that lead to a significant business-wide outage.

This document will build the architecture in steps, starting with a high-level view, then drilling down to examine the specifics of System p5 in the context of the overall environment.

A separate document is available that discusses high availability sample architectures based on pSeries or System p5 servers.

Useful Background Information

This document does not specify the amount of memory (RAM) or disk storage for each step. Memory and disk storage sizes must be determined by planning for the workload that will be run and examining capacity requirements. In general, more memory or disk storage is better than less; also, keep in mind that the amount needed tends to increase over time. The book cited below talks about this topic in great detail.

- Redbook: IBM eServer pSeries Sizing and Capacity Planning: A Practical Guide http://www.redbooks.ibm.com/abstracts/sg247071.html?Open

The IBM System p5 hardware range supports the optional Advanced POWER Virtualization feature, a suite of hardware and software technologies that provide access to the following components:

- Micro-Partitioning technology (LPAR creation and management where individual LPARs are given less than one physical CPU)
- Virtual I/O Server (virtual SCSI, virtual Ethernet, and Integrated Virtualization Manager)
- Partition Load Manager

Further information about virtualization on System p5 can be found at:

- Advanced POWER Virtualization overview
  http://www-03.ibm.com/servers/eserver/pseries/ondemand/ve/

- Redbook: Advanced POWER Virtualization on system System p5
  http://www.redbooks.ibm.com/redpieces/pdfs/sg247940.pdf

Capacity on Demand (CoD) is an important optional feature of the System p5 range. It gives the ability to build in reserve CPU and memory at the factory that remains inactive until needed. Additional information on CoD is available at:

- Capacity on Demand overview
  http://www-03.ibm.com/servers/eserver/pseries/ondemand/cod/

- IBM eServer System p5 550, System p5 570, System p5 590, System p5 595: Working With Capacity on Demand
  http://publib.boulder.ibm.com/infocenter/eserver/v1r3s/topic/ipha2/ipha2.pdf

This document does not attempt to cover IP or storage networking in any detail and instead illustrates these concepts at the block diagram level. Detailed information regarding networking and storage area network (SAN) technologies and practices can be found at:

- Redbook: Introduction to Storage Area Networks
  http://www.redbooks.ibm.com/redbooks/pdfs/sg245470.pdf

- Redbook: IBM TotalStorage: SAN Product, Design, and Optimization Guide
  http://www.redbooks.ibm.com/redbooks/pdfs/sg246384.pdf

- Redbook: IP Network Design Guide
  http://www.redbooks.ibm.com/redbooks/pdfs/sg242580.pdf

- Redbook: Extending Network Management Through Firewalls
  http://www.redbooks.ibm.com/redbooks/pdfs/sg246229.pdf

Detailed information regarding the IT Information Library (ITIL®):

- IBM ITIL White paper
  http://www-1.ibm.com/services/us/imc/pdf/g510-5072-information-technology-infrastructure-library.pdf

- IBM Global Services ITIL Website
  http://www-1.ibm.com/services/us/index.wss/offerfamily/its/a1000429

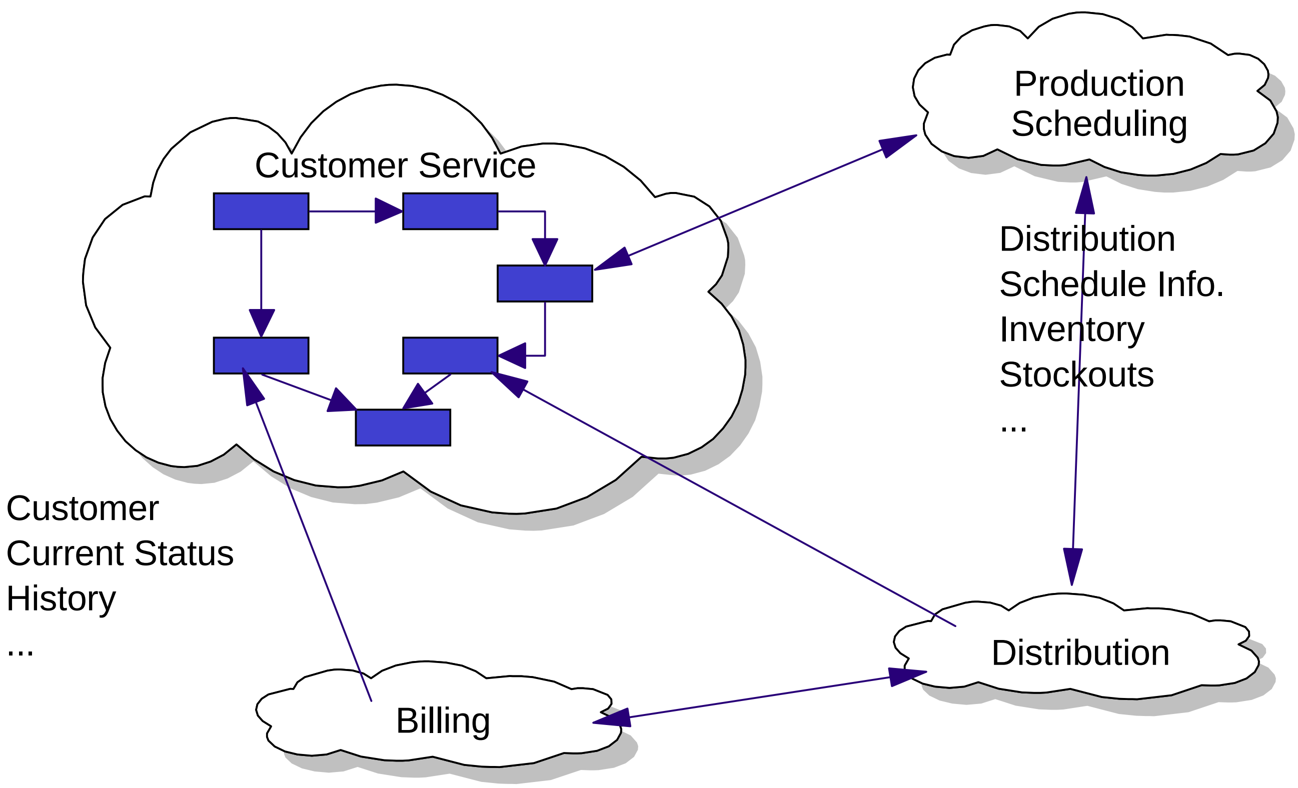
Production Environment Description

Overall Infrastructure Considerations

In practice, when designing an IT infrastructure, you would start with a summary of the business context and functional plus non-functional requirements. Though this document is not intended to be a treatise on how to design a modern end-to-end IT infrastructure, the examples shown here are gleaned from best practices and lessons learned from IBM and its varied customers. IBM also brings the necessary design skills to the table when working with a client to deliver a new or updated infrastructure.

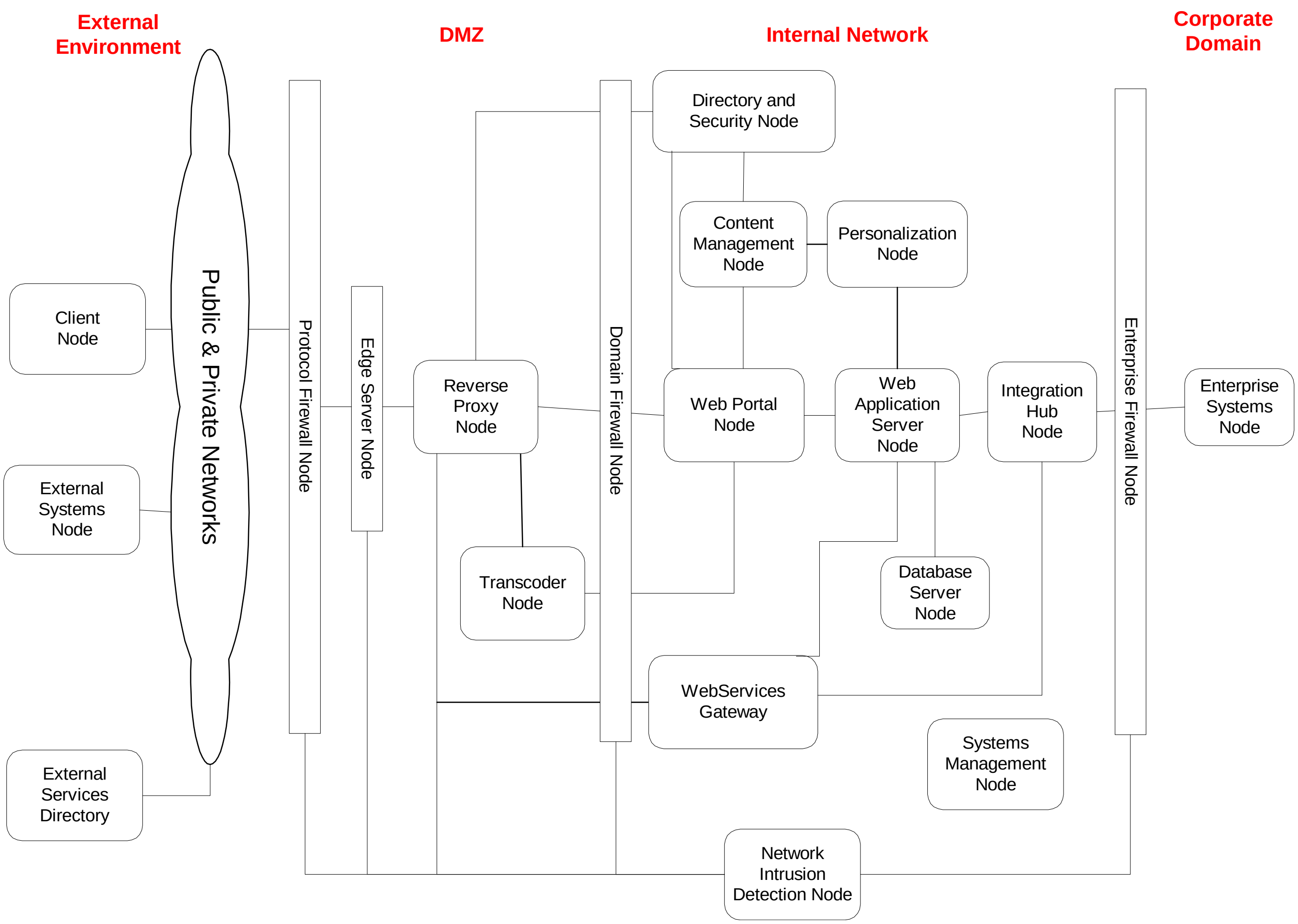
For the purposes of this document, it is useful to quickly illustrate the different levels of thought that go into designing an IT infrastructure. As with any type of project, defining the requirements and goals is a critical and sometimes difficult first step. One technique used to get a handle on this often daunting task is to create a business context diagram, an example of which is shown below.

Figure 1 - Sample Business Context Diagram



The business context diagram, along with the requirements, plus use cases, would in turn be used to direct the design of a high-level topology diagram. The diagram below illustrates an infrastructure for a business that conducts transactions over the internet, certainly a common model. Keep in mind that this diagram is fairly abstract; the term "node" does not refer to a single server but instead is used here to indicate an IT function within the overall infrastructure. This modular approach makes it easier to keep track of the different functions required in the overall infrastructure.

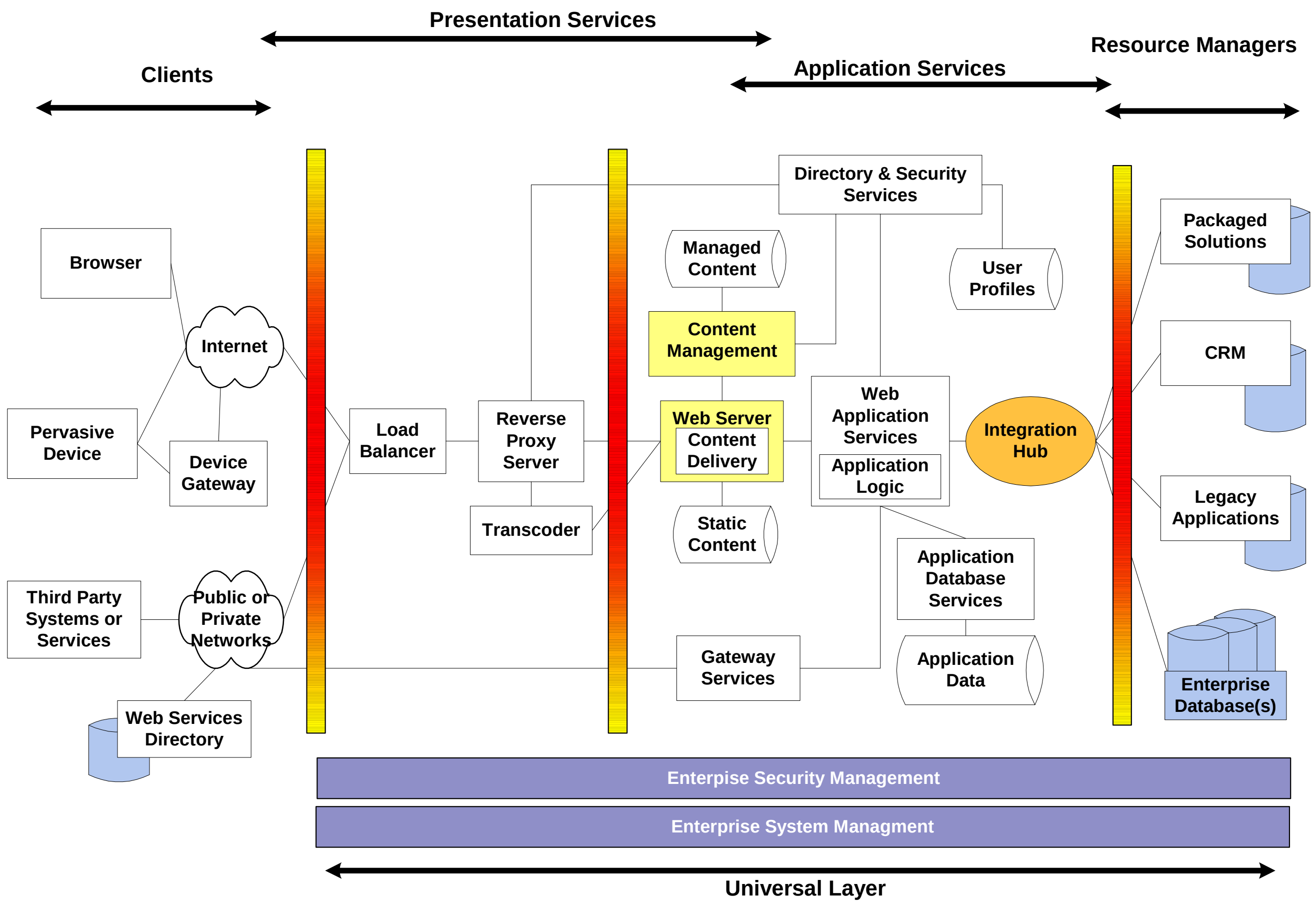
Figure 2 - Conceptual-level System Topology Diagram



A detailed examination of each node in this diagram is outside the purpose of this document, but its inclusion here serves as a useful reminder of the many functions that must be considered when designing a modern IT infrastructure. Note, for example, the inclusion of three levels of firewalls as well as the "directory and security" and "network intrusion detection" nodes. The modern business climate demands a focus on security as indicated in this diagram.

Clearly, the level of detail is insufficient to be used directly as a physical representation of the infrastructure. From the diagram shown above, an IT architect would then produce an architectural overview diagram similar to the example below.

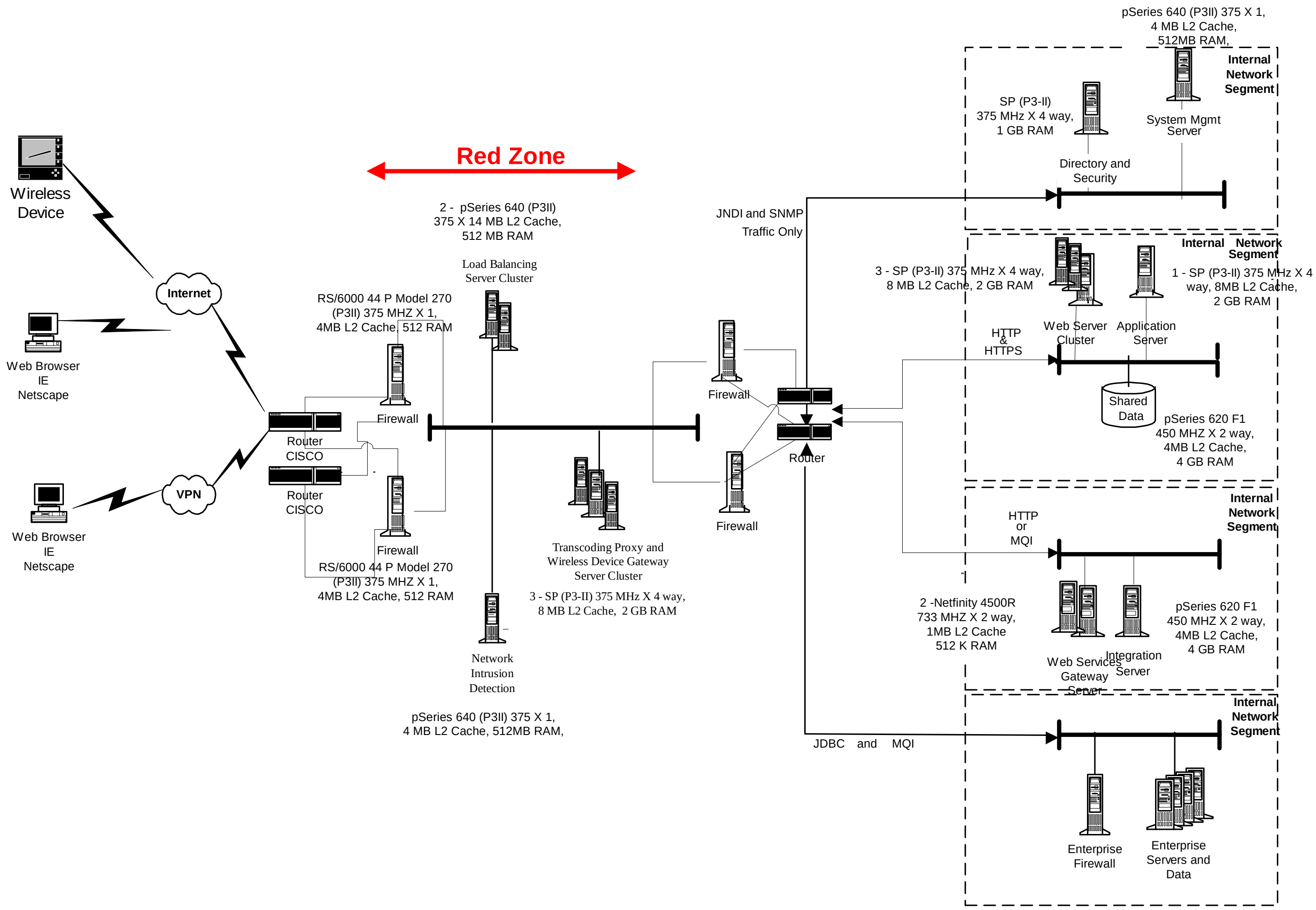
Figure 3 - Sample Architecture Overview Diagram



You can see how the "nodes" shown in the topology diagram are now replaced with indications of potential servers, workloads, and storage, and the interconnections between the components suggest how the IP and storage networks might be designed.

From the architecture overview diagram, the IT architect and technical leaders begin to sketch out a physical layout, similar to the example shown below. Don't worry about the specific callouts in this diagram – it is meant only to illustrate what a typical physical layout diagram might look like. You can see how the abstract components in the architectural overview diagram above are now translated into a more physical representation, with example servers, storage, and networking components being illustrated.

Figure 4 - Sample Physical Layout Diagram



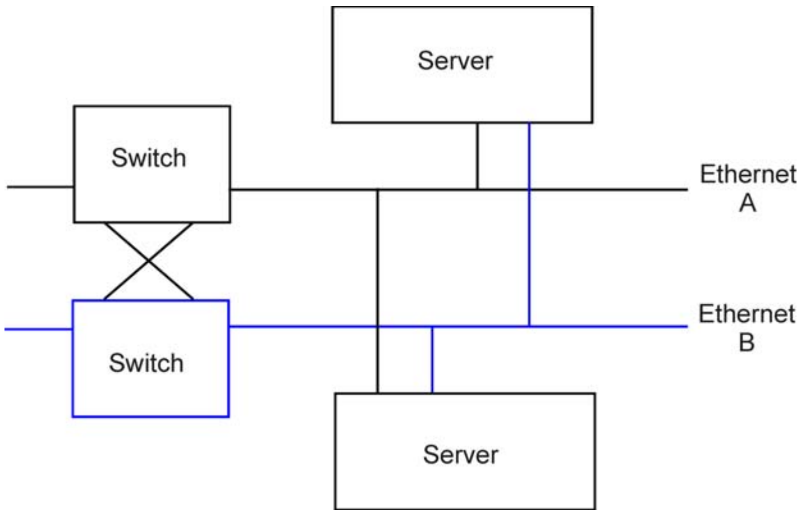
The point of this section is to illustrate the disciplined, step-by-step process that is employed when an IT infrastructure is designed. In particular, when IBM IT architects are engaged by a client's IBM sales team to build a new infrastructure (or to update a client's existing infrastructure), they base their efforts on this process and have access to industry-specific reference architectures on which they build a particular client's solution.

Network Considerations

The sample physical layout diagram shown above indicates a segmented network approach, which is typical for an infrastructure of this scope. What the diagram doesn't illustrate is the need for redundancy throughout the IP network topology. Even if you are not looking to build a highly available infrastructure (in which redundancy at every level is coupled with HA clustering software to create systems which can react automatically in the event of individual component failures), it is still very important to build in network redundancy and segmentation. By doing so, you give yourself the ability to recover more quickly from a network component failure.

Typically, two IP network topologies should be built in parallel, with redundant network hardware (e.g., switches, routers, firewalls) connecting each segment. Further, the network components should be cross-connected so that traffic being handled by one component can be automatically handled by the second in case the first one fails. Each server on the network should have an Ethernet port connected to each Ethernet network. The following diagram illustrates this concept.

Figure 5 - Dual Ethernet Paths



A further note about the server Ethernet connections: many Ethernet adapters now offer dual ports on a single adapter. Keeping the example diagram in mind, it is tempting to use one port on a single adapter for Ethernet A, while the second port is connected to Ethernet B. Be aware that this approach makes the Ethernet adapter a single point of failure. If the Ethernet adapter in the server fails, both of the ports it provides will be unavailable, effectively cutting the server off from either Ethernet network. To avoid this issue, install an Ethernet adapter for each network connection required by a given server.

Storage Considerations

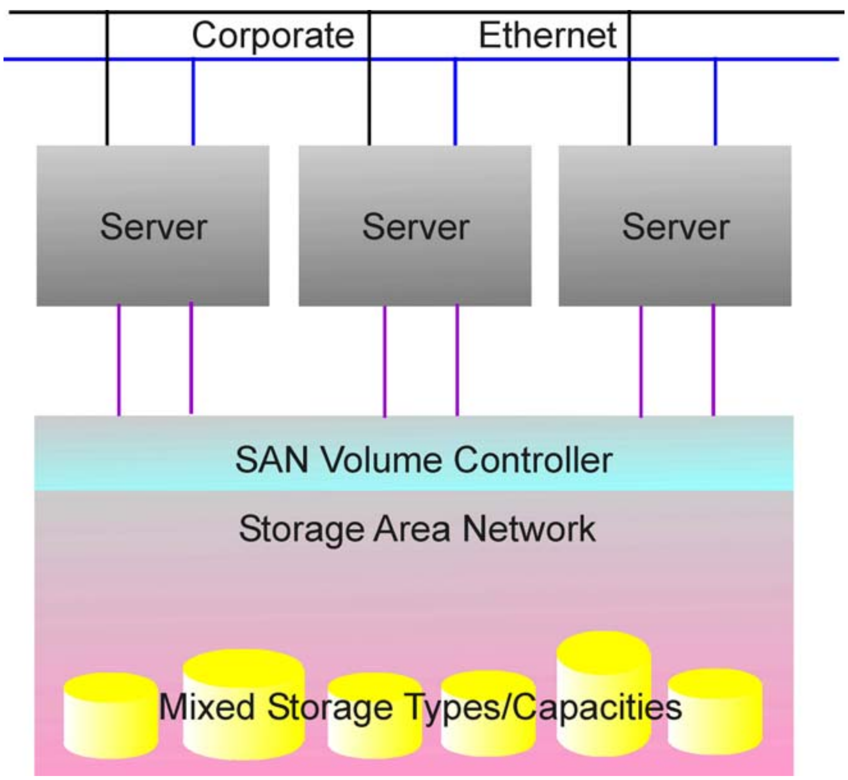
The attachment of storage to servers in the infrastructure brings with it the same concerns about redundancy in terms of access to business data. In years past, external storage devices were directly attached to the server which needed to read and write data on that device. Administrators quickly learned that redundant connections (e.g. SCSI, Fibre Channel (FC)) were required so that the server didn't immediately lose access to its data if a single storage connection failed. This still did not address the problem of providing quick access to the data in the event of a server failure. The workload on the server could be brought up elsewhere, but the storage would have to be cabled to the replacement server, lengthening the duration of the outage.

There is also the issue of protecting against storage device failures. Today, storage devices, especially at the higher end, have significant redundancy built in, but even so, one must plan for the remote possibility that the entire storage device could fail, or could require a disruptive firmware upgrade. While that particular storage device is down, how can its data be accessed so that the servers depending on the data can continue working?

Enter the Storage Area Network (SAN) – in gross terms, applying the concept of networking to the storage world. SANs are well-documented and now pretty common, so this document will not spend any time describing SAN concepts. Further reading on the subject is cited in the Useful Background Information section above. As with Ethernet networks, it is possible and indeed strongly recommended that SANs are designed with redundancy at every level (storage connections, switches, directors) so that individual component failures do not cripple the SAN. In the diagram below, note the redundant connections between each server and the SAN fabric. Avoid the temptation to use a single dual-port adapter to achieve the dual connections, as the adapter itself becomes a single point of failure – if it malfunctions, both storage connections will be unavailable. Instead, install a dedicated I/O adapter for each storage connection.

In the traditional SAN model, care must be taken to map different classes of storage devices to different storage needs. Often, administrators must deal with cases where some storage devices are underutilized, while others are running out of available space. Storage virtualization is an interesting way to address this problem. Conceptually the idea is to decouple servers from the storage they are using.

Figure 6 - Storage and IP Networking Block Diagram



In pSeries lingo, all the server sees is an hdisk – it doesn't know (or care) what physical storage device is providing that disk resource. IBM's TotalStorage SAN Volume Controller (SVC) is a hardware/software product that adds this storage virtualization capability. While it is optional when building an IT infrastructure, it should be strongly considered when there is a need for an adaptable, resilient storage infrastructure that can maximize the use of existing storage capacity across a mix of storage devices.

Despite the advantages offered by advanced storage technologies in the areas of manageability, resiliency, and virtualization, consideration must be given to the protection of critical data. In addition to the mirroring and backup of business data (e.g. database contents), consider also protecting critical software programs from the operating system up through the middleware/application stack. Modern storage systems offer mirroring capabilities, and in the case of AIX 5L™ on pSeries, the operating system can provide options for data mirroring and backup as well.

IT Process Considerations

You can't achieve high levels of availability without discipline in the infrastructure design and implementation activities, nor will strong availability characteristics be possible if there are single points of failure in the IP or storage networks. However, even if you do those things correctly, high levels of availability will remain out of reach if the IT processes governing the operation and maintenance of the infrastructure are ill-defined or not well-documented. Additionally, if the IT processes are well defined and documented, high levels of availability will still be unattainable if the IT staff is not well-trained on those processes and is not maintaining the IT processes over time.

Often, businesses will suffer outages due to deficiencies in the design of the infrastructure, or due to single points of failure in the network environments, or due to immaturity in the business's IT processes. It is tempting to throw more technology at the problem, such as purchasing an HA clustering software package and hurriedly implementing HA clustering as the cure-all against further outages. Unfortunately, taking this approach without first addressing any underlying infrastructure or operational issues actually tends to make the problem worse, because you've now added another layer of software into the mix that must be managed along with the rest of the infrastructure.

Here are a few basic IT process lessons learned:

- IT processes must be clearly defined and documented. Trying to run even a small IT environment "by feel" is doomed to fail.
- IT staff must be trained on the IT processes, and the processes must be maintained as the environment changes over time. Too often, a business will take the initiative to get control of their processes and go through the hard work of defining and documenting everything. Copies are printed, bound, and handed out. The copies either disappear under other paperwork or get put on a bookshelf. In a surprisingly short period of time, the actual processes deviate from the written processes.
- Institute regularly scheduled process tests, and use the test results to identify and correct inaccuracies or weaknesses in the process. This also forces the operations staff to review and touch the processes, increasing their process familiarity.
- IT roles and responsibilities must be clearly defined. For example, consider defining an availability manager role. An availability manager becomes the "traffic cop" for all changes to the environment and the focal point for resolution in the event of an outage.
- A central database is needed to document all aspects of the environment (e.g. standard software catalog, individual machine firmware and operating system levels), so that everyone has access to the same, up-to-date information. Too

often, everyone keeps their own copy of some piece of data about the environment, which leads to contradictory views of the state of the environment.

The IT Information Library (ITIL) is a standard framework for use as a base in defining IT processes and using them to manage an IT infrastructure. While at first glance the totality of the ITIL framework can seem overwhelming, best of breed IT shops invest the time and effort in embracing ITIL and basing their own IT processes on it. Sources for further reading about ITIL are cited in the Useful Background Information section in the Introduction.

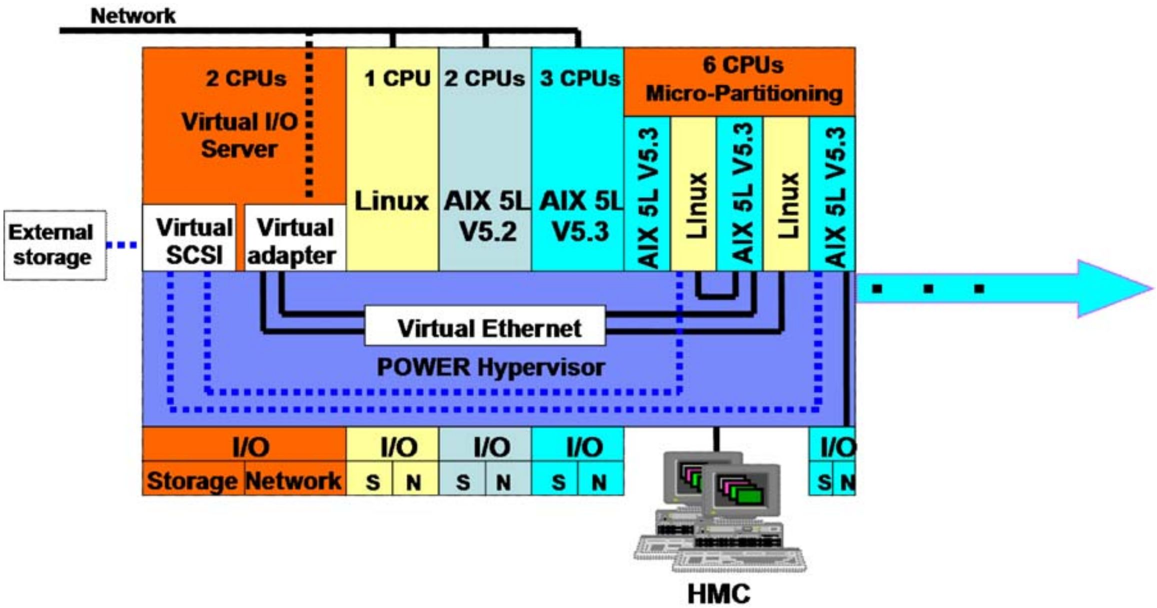
System p5 Server Specifics

Up to this point, we've discussed at a high level general considerations that one must take into account when designing and building out any production-level IT infrastructure. An infrastructure incorporating the practices and technologies discussed so far typically has different server types deployed (Intel® processor-based, commercial UNIX® operating system such as AIX 5L on System p5, and often in large environments, mainframes on the backend). For the purposes of this document, we will now focus on how System p5 servers fit in this type of infrastructure, pointing out how System p5's virtualization and capacity management features bring additional value and flexibility. Let's quickly recap these features here:

- Logical Partitioning (LPAR)/Dynamic LPAR – the ability to subdivide a System p5 server's total CPU and memory capacity into discrete partitions (LPARs), each of which can run an operating system instance. Resources can be reallocated across the LPARs without rebooting the machine or partitions.
- Capacity on Demand (CoD) – a capacity and cost management feature which allows reserve CPU and memory resources to be installed at the factory in an inactive state. When conditions require it (e.g. due to a workload peak or when manually activated), the reserve resources are activated and assigned to either new or existing LPARs. After the exceptional condition has passed, the resources can then be deactivated and put in reserve again. The customer pays only for the extra resources used during the exceptional condition.
- Micro-Partitioning – part of the optional Advanced POWER Virtualization feature, Micro-Partitioning technology gives the administrator the ability to configure LPARs that receive less than one full CPU's worth of processing power (down to as little as 10% of 1 CPU). This technology gives increased flexibility to the administrator and helps optimize and maximize the use of the server's resources.
- Virtual I/O Server (VIOS) – part of the Advanced POWER Virtualization feature, the VIOS provides the ability to share physical Ethernet and disk connections across LPARs by providing each LPAR virtual Ethernet and disk links and mapping those to existing physical hardware resources. Inter-LPAR communications can take place entirely within the System p5 Hypervisor layer, reducing external IP and storage network traffic and reducing the number of physical I/O adapters required.

- Partition Load Manager – Management software which the administrator uses to balance LPAR resource needs

Figure 7 - System p5 Advanced POWER Virtualization Diagram

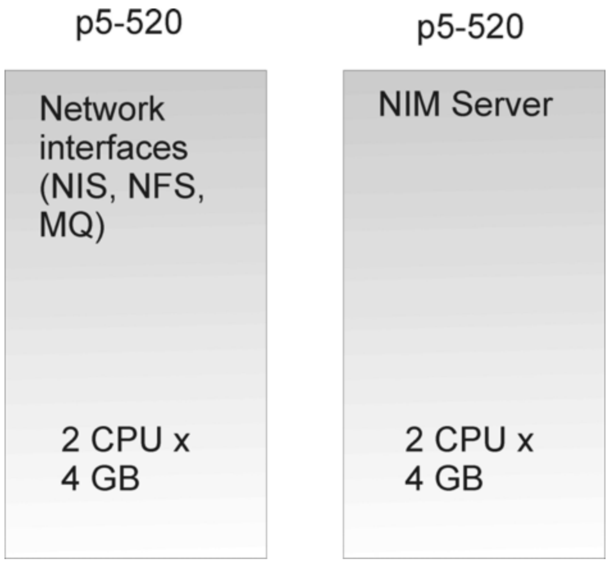


Aside from the thumbnail descriptions given above, this document assumes the reader is familiar with System p5's unique virtualization and capacity management features. Further reading is cited in the Useful Background Information section of the Introduction.

A popular System p5 choice for implementation in a medium to large IT infrastructure such as has been discussed so far, is the System p5 570. This particular machine brings outstanding price/performance, expandability (2-16 CPU's) and a modular packaging approach to mid-size environments. Its big brothers, the System p5 590 and System p5 595 are high-end servers with up to 32 (590) or 64 (595) 64-bit POWER5™ CPUs. Customers also deploy some of the lower-end System p5 servers such as the System p5 520 and System p5 550 for smaller, standalone workloads.

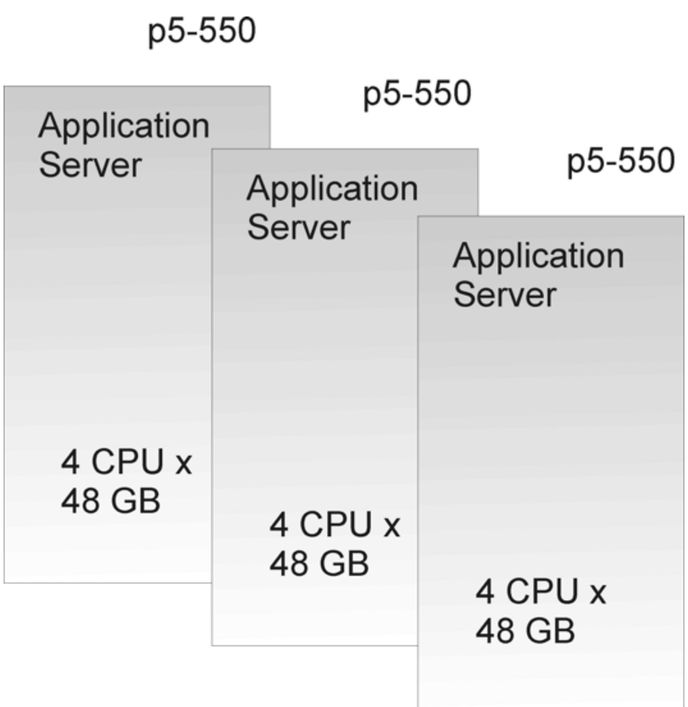
Now, let's take a look at how System p5 fits in this type of environment. Thanks to its LPAR features, System p5 is an attractive way to consolidate different workloads in a single footprint, reducing raised floor space and costs. For clarity, the following diagrams do not show external IP or storage network connections. Assume, unless otherwise noted, that each LPAR has redundant Ethernet and storage network links.

Figure 8 - System p5 520's for Specialized Workloads



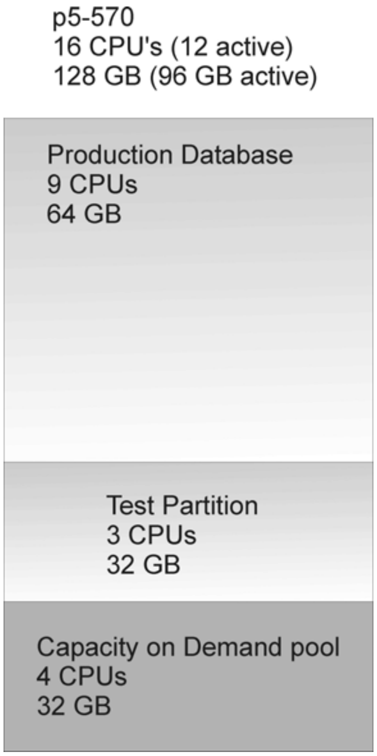
For these specialized workloads, lower-end machines serve well: they are relatively inexpensive and simple to set up and manage. Here, it is not necessary to exploit System p5's advanced virtualization capabilities. In practice, it would be advisable to configure two servers for use as network interfaces, so that a failure on a single network interface server does not cripple the IP network.

Figure 9 - System p5 550's for Application Servers



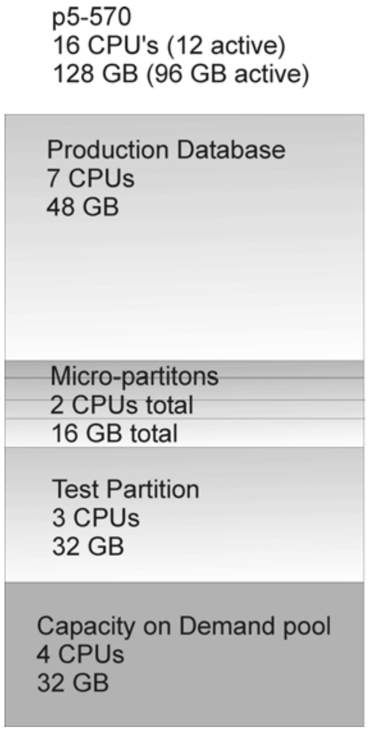
Next, a number of System p5 550 servers are specified for use as application servers. This gives a greater number of CPU's per machine, which reflects the greater power needed for application serving. As with the System p5 520's above, the System p5 550's can be purchased in 4U rack mount packaging, keeping the raised-floor space requirements under control. While three application servers are shown in the diagram, that is just an example and the number used would vary according to a particular customer's needs.

Figure 10 - System p5 570 as Production Database Server



Here, we show how a System p5 570 can be used as a production database server. Note that the bulk of the CPU and memory on the machine is allocated to the database LPAR. A relatively small test LPAR is also specified and could be used for testing of another workload or possibly as a test instance of the database. 4 CPUs and 32GB of memory are inactive but can be activated via Capacity on Demand to address utilization peaks in the database LPAR, or perhaps expanded capacity for the test LPAR. Additionally, if a workload on another server failed and needed a place to run while the failure was being addressed, the test LPAR on this server could be reconfigured (possibly including the addition of resources from the inactive pool) so that the other workload could run on this server.

Figure 11 - Production Server with Micro-Partitions



While the System p5 570 example above shows a small number of LPARs, it is not difficult to imagine a System p5 570 configuration where a number of smaller workloads are all configured in LPARs, possibly even using micro-partitions to divide the system resources very granularly. A potential configuration is shown here. Note that four micro-partitions are specified, with a total of 2 CPUs and 16 GB of memory allocated across the micro-partitions. These partitions can be individually grown or shrunk as needed.

The System p5 590 and System p5 595 deliver the same virtualization and capacity management capabilities but provide more total horsepower, either for running a few very resource-intensive workloads or many smaller workloads.

Next Step: Consider HA

Up to this point, we have built upon the hardware capabilities of the System p5 range and the features of the optional Advanced POWER Virtualization and optional Capacity on Demand technologies to create flexible, efficient, and available configurations for use in medium to large IT environments. Failures in one LPAR do not affect other workloads running in other LPARs. By using dynamic LPAR reconfiguration and CoD, it is possible to restart the failed workload quickly in a different LPAR so that diagnosis can take place in the failed LPAR. Testing needs can be addressed when needed without always having to purchase separate servers.

Still, one must ask: what happens if an entire server fails, whether due to a fluke hardware failure or the need to apply a firmware update that requires a machine restart in order to take effect. In the System p5 570 example above, what happens to the database if the entire server must be taken down? Most likely, another server in the infrastructure would have been set up with lower priority workloads that could be shut down temporarily, allowing for the database to be run there instead. Still, these processes, in the context of the environment discussed in this document, are manual in nature.

Also, failures in single workloads running within an LPAR require manual intervention to correct, whether the solution is simply to restart the workload, reboot the LPAR, or move the workload to a different LPAR. Human intervention takes time and brings with it the risk of a mistake being made when attempting to recover from the failure. Can your business tolerate this risk?

The two preceding paragraphs suggest scenarios and questions that generally lead an organization to consider high availability. If the answer to the question “can some part of your business afford to be down while the server is rebooted?” is “no,” then it becomes necessary to consider methods that will help protect against such possibilities. In short, you must start looking at options for high availability.

While even a high-level treatment of high availability is beyond the scope of this document, there is one concept that is the starting point when considering high availability: eliminating single points of failure (SPOF). In this document, we discussed the advantages of introducing redundancy in the IP and storage network layers. In a similar vein, if a single server is running critical workloads, and it is determined that the cost of an unplanned server reboot is prohibitively high in terms of lost revenue and productivity, it is then reasonable to consider adding a second, redundant server. A more detailed discussion of highly available pSeries environments is presented in a separate document.

http://www.ibm.com/servers/eserver/pseries.