fast SBP™

High availability search

Enterprise search is mission critical for many enterprises - for example, an eCommerce seller or a criminal investigator cannot afford for their search boxes to be out of order. In face of inevitable hardware failures, search systems use carefully architected redundancy for constant availability. Search providers should differentiate between availability on content ingestion side and on user query side.

A technical white paper

by FAST Search Best Practices

5 things you should know about high availability

1. Computers fail for many reasons: hardware, software, power, connectivity, etc.
2. Service degradation can be abrupt or graceful
3. High availability is ensured with redundant systems
4. Different parts of a system can have varying degrees of fault tolerance
5. Downtime costs money, but so does redundancy!

Critical IT systems are often described as fault-tolerant, redundant, or displaying high availability. That is, should something go wrong (for example, the office cleaners pulling the plug on the servers by mistake) the systems have been designed to maintain a certain level of service.

The standard solution is to purchase more hardware in order to mirror vital elements of the system. The downside of this simple solution is the price of the equipment and the internal cost of managing a duplicate system. Consequently, the extra expense of redundancy is tallied against lost revenue from downtime and the odds of certain fatal errors occurring in the different parts of the system.

The extent of built-in redundancy required needs to be measured against the opportunity cost of a failure.

For example, an e-commerce site will know its sales distribution for one day, and can estimate the potential sales losses for downtime during peak hours. In the same way, a pay-per-click search service can measure the loss of revenue associated with downtime. In addition, the search provider must evaluate the damage to user loyalty from an unsuccessful search experience, and the cost of rebuilding the index.

Equally important are content indexing and searching. In an e-commerce application, delaying the addition of new products is less costly than the loss of querying. But for a military intelligence or financial trading application, having out-of-date information may be worse than having no information at all.

Q: My e-commerce site generates revenues of thousands of dollars per minute. How much would it cost to guarantee 100% uptime?

A: Unfortunately, no amount of money can guarantee absolute 100% uptime. Having three fully independent data centers, each with 99.5% SLAs (that is, fully mirrored redundant systems), along with highly experienced systems administrators to repair systems as soon as they break down, is your best guarantee against downtime.

This paper will tackle the different ways of creating a highly available search system, and the different scenarios that search providers must cater to.

Tackling failures to enhance search

Backbone failures

It's important to decide which eventualities to plan for when specifying a redundant search infrastructure.

Network outages require two separate ISPs (Internet service providers) with independent cabling for correct fault tolerance. Short power failures can be avoided with UPS (Uninterruptible power supply); longer ones with back-up power generators. Users can also pull in power from different parts of the power grid for extra redundancy. Dependent on the location and sensitivity of the system, physical attacks can be protected against with various security measures (locked doors, armed guards, etc.) and robustness in building construction can be considered to protect against natural phenomena such as earthquakes.

Q: My archiving solution application is replicated for redundancy. How do I ensure the same level of protection for search?

A: When there is a pre-existing multi-server distribution of data within an application, the simplest method is to replicate that structure. For example, if each archiving box stores 50 million documents, each should have a search node of the same size. If it's feasible, this can be installed on the same server as the data. In this case, the redundancy model will automatically conform to the data fault tolerance levels.

When considering independent data centers, it's important to weigh the administrative and maintenance effort of running two separate systems (an administration team in each location, large amounts of data-traffic between locations, for example). Two inexpensive data centers in distinct locations may be cheaper than one fully redundant and secure one. On the downside, because the two installations would be fully independent, data synchronization is not guaranteed, which can affect some applications.

Component and service failures

Companies typically have a corporate or service-wide policy regarding power supply and data center security, and search will likely conform to this policy. But search services will also have specific high-availability considerations, requiring hardware redundancy combined with intelligent recovery operations to ensure search uptime.

Search applications to consider are:

Content aggregation and processing. This refers to the crawling or capture of data from source repositories and any pre-indexing transformation. Failure would interrupt the continual flow of new or updated information into the search index.

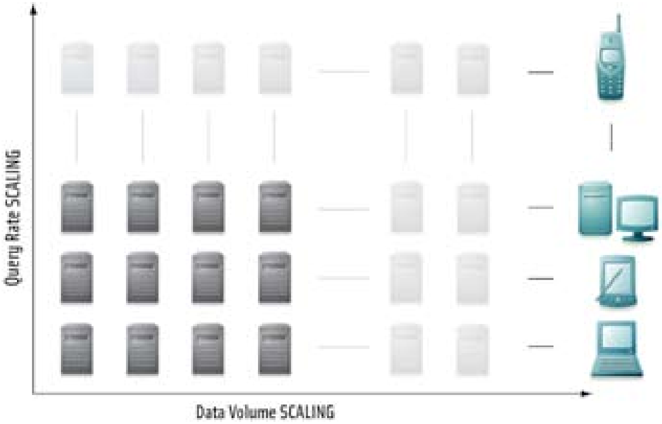
Search engine. The search engine is where the actual queries are assessed and documents returned. No search is possible without it; although in a system distributed over multiple nodes, a partial service may be available if some nodes are still alive (searches will of course be against an incomplete index).

Search broker. This is where the merging of content from multiple search nodes and any query or results processing are performed, as well as the federating of queries. No search is possible without this.

User interface. The application with which users interact with the search engine. No searches are possible without it. The presentation layer infrastructure (Web servers, Java/.NET environment, etc.) is typically independent of the search technology provider and in certain situations it is acceptable for the user interface to be available while search is offline.

Administration and management tools. Search services will run without such tools, but the ability to monitor for fatal failures will be reduced

Elements of High Availability of Search



Factors that affect high availability

Before implementing a search solution, we recommend that users investigate::

Whether a particular component should have some fault tolerance

What procedures should be in place for rapid restoration of the system

Whether the service should have a live and failover node, or a redundant system with graceful degradation.

Content aggregation and processing: A second set of all relevant components is necessary to offer high availability of content ingestion into the search engine. For the content distributors and document processors, parallel installations can either run continuously or in hot fail-over mode. Connectors, which contain state information, typically deliver redundancy with one installation in active mode and a second in passive mode. To achieve this, the state of the connector (for instance, which documents have been indexed, when the last full batch was completed, etc.) can be stored in a shared environment displaying high fault tolerance, such as a SAN (storage area network) or database.

Importantly, although connectors may display high availability, both set-ups will run against the same data source. Therefore, guaranteeing the uptime of data feeds requires administrators to adopt policies and redundancy to ensure the same level of service from the underlying repository.

It is also considered best practice to use separate network infrastructure for the content processing and search engine.

Search engine: Fault tolerance is guaranteed by duplicating a row. Typically an actual index in a large system will be shared across multiple machines in a grid configuration, where each segment of the content is contained within one column of servers. Smaller systems may have only one column. To increase query performance or fault tolerance, more rows are added, containing duplicates of the index. These spare nodes can be set up as backup servers in an active/passive configuration. They can also all be active, allowing a graceful degradation in the maximum query throughput that the system supports.

Two Dimensions of Scalable Search

Generally, it is best to share the load across all servers and not to use active/passive failover. This improves availability by reducing the average load on one server and thus improving mean time between failures. Also, a server may sometimes fail when it experiences large changes in load, which makes active/passive configurations somewhat dangerous. However, an active/passive set-up is appropriate where the spare machines are used for other purposes – for example, if one server hosts the backup for three live servers.

Fault tolerance of a search engine is guaranteed by duplicating the search rows.

In the case of a distributed grid configuration, a failure could result in an incomplete index. A cluster of 200 million documents may consist of five columns, each with 40 million entries. If one full column fails (that is, all the servers in that column), queries are performed on a subset of 160 million possible hits. If this is acceptable, the service may well stay up. If it's unacceptable, search should be taken off-line until at least one node from each column is back up.

With regard to large deployments, it is important to understand that two actions are performed by a search engine: indexing (converting the text data into a searchable index) and searching, where the binary files created by the indexing process are used to serve queries. Some options when setting up the search engine for fault tolerance using multiple rows therefore include: 1) run indexing and search on every row, which will provide fault tolerance for both indexing and search but will hurt query performance since machine resources are used during the indexing stage; 2) run with a single indexer and multiple search rows, providing fault tolerance for search but not indexing, but allowing the search performance to scale better since each search row is 100% dedicated to search; and 3) a hybrid approach.

An important factor when designing mirrored systems is the number of co-dependencies. For higher fault tolerance, each row calls for its own network switch and power layout. In addition, each server should have built-in robustness to protect against hard-disk failures, which can be achieved with RAID (Redundant Array of Independent Disks) disk configurations. For machines with raw data, such as the crawlers, fully mirrored disks should be used. For search servers, the extra disks may be better used for increasing performance with disk striping rather than redundancy, or by choosing the RAID 5 level (striping with a parity disk) since the index can most often be rebuilt from the content in the case of disk failure.

The use of a SAN (or NAS) is another option for improving resistance to disk failures, since networked disk systems can be designed for high performance with redundancy.

Search broker: High availability of search necessitates redundant search brokers. Traditionally, multiple instances must be installed with a third-party load balancer which will distribute queries and accommodate for failed servers. The load balancer can be either hardware- or software-based.

Realistic design of high-availability systems

There is a basic trade-off between hardware and maintenance costs and service uptime when designing a high-availability system. Correct and diligent assessment will make the system design a cost-neutral and well-justified business decision rather than a gamble on the chances of a system outage.

Depending on your uptime requirements, some essential metrics and costs to take into account are mean time between failures (MTBF), mean time to recovery (MTTR), the cost of hardware and maintenance, and the costs of indexing and search downtime. These are important when reviewing the SLA (service level agreements) of data centers, hardware suppliers, and support contracts.

Correctly and diligently assessing the probability of failure makes system design a cost-neutral and well-justified business decision, rather than a gamble on the unforeseen.

Best practice is also to compare premium hardware service contracts (e.g., four-hour onsite repairs from Dell, IBM, Cisco, etc.) versus anticipatory purchasing of spare hardware such as network switches. In reality, hard-disk failures are the most common cause of server outages. Assuming the search index has redundancy, these malfunctions can be easily guarded against with spare disks.

Consideration must be given to the procedures to recover from failures – replacing faulty hardware and rebuilding the servers. Frequently the most robust setups are not necessarily those with the most costly redundant hardware; rather, they are the setups that were planned for and tested for failures. It's critical to have a recovery plan for every kind of disaster and to test the plan before an actual event occurs. The plan and the tests should include everything from relatively simple recovery procedures such as recovering and re-synchronizing a single row to restoring an entire index from backup and setting up basic search on a totally new set of servers.

Q: Is it necessary for a corporate intranet search to have full failover?

A: Not if there is no immediate financial loss incurred by an outage; the cost of equipment duplication will be prohibitive. It pays to look at the time to rebuild an index. If this is longer than the accepted downtime, we recommend either having a regular back-up of the index or at the very least a passive back-up on a spare machine.

Different companies, different solutions

In the case of a mid-sized company using search across a knowledge management system or an intranet, search is non-critical to the company's business continuity. The downtime cost is tied to the extra time users must spend looking for information. A redundant search node will protect against failure of the core index without the extra complexity of load balancers and a fully distributed system.

On the other hand, search is critical for an e-commerce player since revenue generation and brand strength rely on provision of quality service. The application requires a data center with power generators, a secure server room, a highly available grid of servers, load-balancers and so on if it is to provide search redundancy. In this instance, there is redundancy of the search service, with graceful degradation of performance when nodes fail, but no indexing redundancy.

For cases where uptime is vital, such as mission-critical applications, infrastructure management tools can be used to monitor the health of the hardware and software. In addition, applications such as military intelligence may consider a fully robust system with independent secure data centers, on-standby passive search nodes, and fault-tolerant hardware.

Mini case study

100% uptime for over two years on a 200 million-document index

Who
Major online science journal index

Challenge
To support 100% of queries against an agreed SLA even during power outages.

Solution
Three mirrored nodes, each sized to handle the projected loads alone, and each with independent network switch and power layouts. There are dual network feeds to the data center as well as multiple front end and query servers. Four-hour service contracts or spare hardware are available for all servers and networking gear..

The fundamental steps to improve search

Designers of fault-tolerant systems often make two common mistakes. First, they fail to address fault tolerance until after designing the initial system, making an architecture rework prohibitive. Second, some systems are over-engineered to deal with unrealistic threat levels.

It is important to know the place and value of search within your system. For example, a search engine uses an index that can be rebuilt from the original data, so it's crucial to assess the true cost of downtime (for example lost revenue, or IT staff being taken off projects to rebuild systems) to determine the main concerns.

The size of the main system will have cost effects on the failover system. So it's vital to size the system carefully.

Simple mirroring of the search nodes will provide high availability to the level required for most applications. Then the decision must be taken whether each complete node should be designed to handle all expected traffic, or whether there is a degradation below the required service level (query speed, average response time, etc.) during a partial failure. Systems like these typically run for more than two years without fatal outages. When search is mission-critical or downtime comes at a very high financial cost, service providers should consider a complete second data center or higher level service agreements with hardware support providers.

Lastly, a system is deemed "available" if the end-to-end application is online. Therefore, the investment in enabling highly available search must be made within the context of a robust IT infrastructure.

Frequently asked questions

Q: What's a fail-soft system?

A:When system components fail, a fail-soft system continues to operate, but with reduced functionality. Such systems are also often said to provide "graceful degradation."

Q: What's a fail-stop system?

A: A fail-stop system will not provide any functionality if system components fail. It may return false results – a situation often referred to as a "Byzantine failure."

Q: What's a SAN or a NAS?

A: The acronyms refer to "storage area network" and "network attached storage". The servers used are remote high-performance drives shared across multiple machines, and often connected with a fiber channel. SAN uses a disk controller and acts as a local disk, communicating via disk-access protocols, whereas NAS uses network protocols to communicate.

Q: What is RAID?

A: It is short for "redundant array of independent disks" – a configuration of multiple drives used to provide fault tolerance (via mirroring or parity checking) or higher performance (via striping). RAID 5, with striping and parity checking, is frequently used to increase search- engine performance and provide a certain level of redundancy.

Q: What is MTBF?

A: It is the mean time between failures - the total elapsed time subtracted by downtime divided by the number of failures of the component.

About FAST SBP™ (Search Best Practices)

SBP consulting is a highly focused transfer of search knowledge and experience from FAST to its prospects and customers. SBP workshops aim to help enterprises realize the full potential of search, by creating optimal strategic, functional and technical roadmaps, delivered in the form of business model, solution and architecture designs.

Fast Search & Transfer
www.fastsearch.com
info@fastsearch.com

Regional Headquarters

The Americas
+1 781 304 2400

Europe, Middle East & Africa (EMEA)
+47 23 01 12 00

Japan
+81 3 5511 4343

Asia Pacific
+612 9929 7725