Monitoring FAST Search Server 2010 for SharePoint

Table of Contents

Table of Figures

Chapter

1

Monitoring the Microsoft FAST Search Server 2010 for SharePoint

Microsoft FAST Search Server 2010 for SharePoint offers a new choice in enterprise search. Combining the power of FAST with the simplicity of SharePoint, FAST Search Server 2010 for SharePoint delivers an exceptional intranet and people search experience and a platform for building custom search-driven applications.

The efficiency of any search engine is typically judged by how quickly the engine processes search queries and outputs the search results. The performance of the FAST Search Server 2010 for SharePoint too is measured by the same parameters. Users are typically intolerant towards slowdowns in searches and even switch to other search engines if such slowdowns recur too often or persist. To avoid this, administrators will have to continuously monitor the performance of the FAST Search Server 2010 for SharePoint, proactively detect processing bottlenecks (if any), and accurately point you to where the delay occurred and what is causing it.

eG Enterprise embeds a specialized *Microsoft Fast Search* monitoring model for monitoring the performance of the FAST Search Server 2010 for SharePoint.
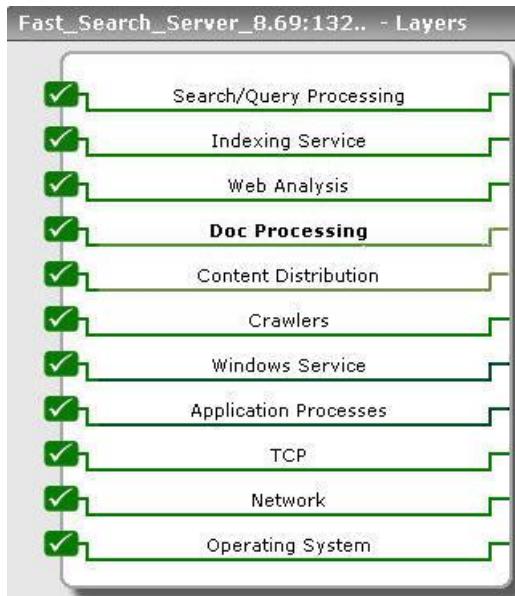
Figure 1: The layer model of the Microsoft Fast Search server

Each layer of Figure 1 above is mapped to a wide variety of tests that collect a wealth of performance information related to the FAST Search server, Using these metrics, administrators can find quick and accurate answers to the following performance queries:

➢ How much time elapsed between item initialization on the server and the completion of indexing? Was too much time spent? If so, where did the slowdown occur? - did it occur before initialized items were placed for submission? did it occur after submitting batches for processing - if so, was it at the first attempt to submit? or was it when waiting for an acknowledgement for submission?

➢ Did too many batches fail even before initializing?

➢ Which crawl collection is currently crawling too many web sites/web links?

➢ Is any content distributor taking too much time to process a batch? If so, which one is it?

➢ Are any document processors invalid? If so, which content distributor are they registered with?

➢ Which content distributor is not utilizing its document processors effectively?

➢ What is the current state of each of the document processors on the server?

➢ Is the workload of the web analyzer evenly distributed across all its link processing components/lookup databases?

➢ Are there adequate sites and URIs in the lookup database to service link information requests?

➢ Is any API queue overloaded with pending operations? Which indexer is such a queue attached to?

➢ What is the load on each indexer partition? How well does every indexer partition index the items on it? Is any indexer partition experience processing delays?

➢ Has a handle leak been detected in any query dispatcher?

> ➤ Have too many queries to any query dispatcher been enqueued? If so, which dispatcher is it?

> ➤ Have any queries to a QR server failed?

> ➤ Is any QR server experiencing latencies when processing queries?

> ➤ Did any query timeouts occur in the indexer?

> ➤ How much time does the indexer service take to process a single query? Is it too high?

The answers to these queries will enable administrators to proactively detect performance bottlenecks (if any).

Since the five layers at the bottom of Figure 1 have already been dealt with in the *Monitoring Unix and Windows servers* document, the sections that follow will discuss the top six layers of Figure 1 alone in detail.

1.1 The Crawlers Layer

Using the tests mapped to this layer, you can quickly detect issues in the operations of a crawl collection and diagnose the root-cause of slowdowns in content crawling

Figure 2: The tests mapped to the Crawlers layer

1.1.1 FAST Content Plugin Test

When users complain that their search queries are taking too long to execute and produce results, you can use the metrics of this test to figure out where the bottleneck is - is it because the FAST Search 2010 server is taking too long to perform indexing? if so, where is time being spent - while waiting for placing the items in a batch? while submitting batches of items for processing? while awaiting a success/failure acknowledgement for a batch submission? On the other hand, if indexing is not the bottleneck, then, could the slowdown have occurred because of too many batch failures? or because the server was overloaded with too many batch processing requests? or could the problem be owing to the slow crawling rate of the FAST Content SSA?

Purpose	When users complain that their search queries are taking too long to execute and produce results, you can use the metrics of this test to figure out where the bottleneck is - is the FAST Search 2010 server taking too long to perform indexing?

	if so, where is time being spent - while waiting for placing the items in a batch? while submitting batches of items for processing? while awaiting a success/failure acknowledgement for a batch submission? If indexing is not the bottleneck, then, could the slowdown have occurred because of too many batch failures? or could it be because the server was overloaded with too many batch processing requests? or could the problem be owing to the slow crawling rate of the FAST Content SSA?
Target of the test	FAST Search Server 2010 for SharePoint
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.
Outputs of the test	One set of results for each crawler collection on the FAST Search Server 2010 for SharePoint farm

Measurements made by the test	Measurement	Measurement Unit	Interpretation
	Avg callback time: Indicates the average time between acknowledgement of a batch submission and the receipt of a final success or failure callback in the FAST Search server.	Millisec	An abnormally high value for the *Avg end-to-end time* is an indication of a slowdown in the search engine. When this happens, you can compare the values of all the other measures to isolate where the slowdown occurred - did it occur before initialized items were placed for submission? did it occur after submitting batches for processing - if so, was it at the first attempt to submit? or was it when waiting for an acknowledgement for submission?
	Avg end-to-end time: Indicates the average time taken from item initialization to completion of indexing.	Number	
	Avg init time: Indicates the average time taken from an item initialization until the item has been placed into a batch for submission.	Millisec	
	Avg submit time: Indicates the average time taken between submitting an item's batch in the first attempt to receiving an acknowledgment for the batch that has successfully been received for processing.	Millisec	
	Avg wait time: Indicates the average time from when an item is placed into a batch until the first attempt to submit.	Millisec	
	Batches failed: Indicates the number of batches that failed prior to indexing.	Number	Ideally, the value of this measure should be zero.

	Currently processing batches: Indicates the number of batches that are currently being processed.	Number	
	Batches ready: Indicates the number of batches received from the content sources that are queued up and are ready to be submitted to the Content Distributor.	Number	If the value of this measure is 0, it indicates that the FAST Search Server 2010 for SharePoint farm back-end is processing content faster than the Content SSA can crawl. This means that you can further increase your crawling rate. If the number is higher than zero, consider adding more document processors, or see whether you have added an external item processor that slows down the speed of certain items flowing through the indexing pipeline.
	Batches submitted: Indicates the number of batches that were sent to the FAST Search server but the callback is still pending.	Number	If the value of this measure is _0_, it indicates that the batches are not sent to the Fast Search Server Farm backend for processing.
	Successfully indexed batches: Indicates the number of items that were indexed successfully.	Number	
	Total batches submitted for processing: Indicates the total number of items that were sent for processing through the Content SSA since the last service restart.	Number	
	Currently processing items: Indicates the number of items that were currently being processed.	Number	
	Successfully indexed items: Indicates the number of items that were indexed successfully.	Number	

	Total documents submitted for processing: Indicates the total number of items that were sent for processing through the Content SSA since the last service restart.	Number	

1.1.2 Web Crawler Collection Test

The FAST Search Web crawler collects content from a set of defined Web sites, which can be internal or external.

The FAST Search Web crawler works, in many ways, like a Web browser downloading content from Web servers. But unlike a Web browser that responds only to user input via mouse clicks or keyboard, the FAST Search Web crawler works from a set of configurable rules it must follow when it requests Web items. This includes, for example, how long to wait between requests for items, and how long to wait before checking for new or updated items.

The main configuration concept in the FAST Search Web crawler is a "collection". Each crawl collection contains the configuration applicable to the particular collection, such as which start addresses and crawl rules to apply. A typical solution might have crawl collections such as *Extranet* or *Blogs*. The FAST Search Web crawler starts by comparing the start URL list against include and exclude rules specified in parameters in the XML file containing the configuration of a *crawl collection*. The start URL list is specified with either the **start_uris** or **start_uri_files** setting, and the rules via the **include_domains** and **exclude_domains** setting. Valid URLs are then requested from their Web servers at a rate determined by the request rate that is configured in the **delay** setting.

If fetched successfully, the Web item is parsed for hyperlinks and other meta-information, usually by a HTML parser built into the FAST Search Web crawler. The Web item's meta-information is stored in the FAST Search Web crawler meta-database, and the Web item content (the HTML body) is stored in the FAST Search Web crawler store. The hyperlinks are filtered against the crawl rules, and used as the next set of URLs to be downloaded. This process continues until all reachable content has been gathered, until the refresh interval (**refresh** setting) is complete or until another configuration parameter limiting the scope of the crawl is reached.

To determine how efficiently the Web crawler functions, you need to understand the current load generated by each *crawl collection* in terms of the number and size of documents that are crawled per collection and the speed with which these documents are downloaded by the crawler. The **Web Crawler Collection** test provides you with these useful insights and helps assess the Web Crawler's efficiency.

Purpose	Helps determine how efficiently the Web crawler functions, by reporting the current load generated by each *crawl collection* in terms of the number and size of documents that are crawled per collection and the speed with which these documents are downloaded
Target of the	FAST Search Server 2010 for SharePoint

test	
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.
Outputs of the test	One set of results for every crawl collection configured in the FAST Search Server 2010 for SharePoint farm

Measurements made by the test	Measurement	Measurement Unit	Interpretation
	Active sites crawled: Indicates the number of websites or web links that are currently crawled with this crawl collection.	Number	The sum of the value of this measure across collections will serve as a good indicator of the current workload of the crawler. If the number of Web sites or the total number of Web items to be crawled is large, the FAST Search Web crawler can be scaled by distributing it across multiple servers. Compare the value of this measure across crawlers to know which collection is generating the highest load.
	Current document download rate: Indicates the rate at which the documents are downloaded with this crawl collection.	Downloads/ min	The crawler's overall download rate depends on the number of active sites that are busy.
	Average document size: Indicates the average size of the documents downloaded with this crawl collection.	MB	This is another good measure of the current load on the crawler.
	Documents in web crawler store: Indicates the number of documents downloaded with this crawl collection that are currently stored in the Web Crawler store.	Number	The FAST Search Web crawler stores crawled content locally on disk during crawling. The content is divided into two types; Web item content and meta data.

	Documents deleted from document store: Indicates the number of documents downloaded with this crawl collection that are currently deleted from the Web Crawler store.	Number	
	Documents downloaded: Indicates the number of documents that are currently downloaded with this crawl collection.	Number	
	Documents stored that were modified: Indicates the number of stored documents that were currently modified with this crawl collection.	Number	The crawler periodically looks for changes in the Web Sites/Web pages configured for crawling and writes these changes to the crawled items that pre-exists in the store. The *Documents stored that were modified* measure reports the number of items in the store that were currently updated with changes. The *Documents writes to web crawler store* measure on the other hand reveals how many such changes were written to the store.
	Documents writes to web crawler store: Indicates the number of current document writes to the FAST Search Web Crawler store.	Number	

1.2 The Content Distribution Layer

A content distributor is a stateless component that receives content in batches from the connectors and distributes batches to the item processing component i.e., the document processor, which subsequently sends the processed batch for indexing.

The test mapped to this layer measures the efficiency of each content distributor on the Microsoft FAST Search server.

Figure 3: The tests mapped to the Content Distribution layer

1.2.1 Content Distributor Test

A content distributor is a stateless component that receives content in batches from the connectors and distributes batches to the item processing component i.e., the document processor, which subsequently sends the processed batch for indexing.

Each FAST Search Server 2010 for SharePoint farm has one or more content distributors. You can ensure good performance by verifying that the following conditions are true:

➢ Item processing components are effectively utilized

➢ Incoming content batches are quickly distributed for processing

The **Content Distributor** test enables you to do just that. This test auto-discovers the content distributors in a Fast Search server 2010 for SharePoint farm and reports the following for each content distributor:

➢ The current load on every content distributor, thus revealing the overloaded distributors;

➢ Delays (if any) in content processing, and the probable reasons for the same - is it because the content distributor took too long to send the batches for processing? is it because the document processor took too long to send the processed batches for indexing? is it because of the absence of adequate 'valid' document processors to handle the load? is it because of ineffective usage of the available document processors?

Purpose	Auto-discovers the content distributors in a Fast Search server 2010 for SharePoint farm and reports the following for each content distributor:
	➢ The current load on every content distributor, thus revealing the overloaded distributors;
	➢ Delays (if any) in content processing, and the probable reasons for the same - is it because the content distributor took too long to send the batches for processing? is it because the document processor took too long to send the processed batches for indexing? is it because of the absence of adequate 'valid' document processors to handle the load? is it because of ineffective usage of the available document processors?
Target of the	FAST Search Server 2010 for SharePoint

test	
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.
Outputs of the test	One set of results for every content distributor in the FAST Search Server 2010 for SharePoint farm

Measurements made by the test	Measurement	Measurement Unit	Interpretation
	Average dispatch time: Indicates the average time taken by this content distributor to send a batch to the document processor since the last measurement period.	Millisec	A document processor converts the items in a batch to a unified internal format. Ideally, the value of this measure should be below 10 millisec. A value higher than 10 millisec indicates a network congestion or overload.
	Average processing time: Indicates the average time taken by the document processors associated with this content distributor to process a batch.	Millisec	The value of this measure would vary based on the type of the batch content and the size of the batch. Typically the value of this measure must be less than 60ms.
	Clients creating feeding sessions: Indicates the number of times the feeding sessions have been created by clients on this content distributor.	Number	The FAST Search Connector (often referred to as the FAST Content SSA) is where you can add content sources, and it is responsible for content acquisition. After content has been acquired by a connector, it is fed from the FAST Content SSA to one of the registered Content Distributors in FS4SP. This measure reports the number of sessions the Content SSA has created on the Content Distributor for feeding content.
	Document processors: Indicates the number of document processors registered with this content distributor.	Number	When a server consists of multiple content distributors, then the document processor would be evenly distributed across all the content distributors.

	Measure	Type	Description
	Document processors busy: Indicates the number of document processors registered with this content distributor that are currently processing a content batch.	Number	The value of this measure must be equal to that of the *Document processors* measure under maximum load.
	Document processors invalid: Indicates the number of document processors registered with this content distributor that are currently invalid.	Number	Ideally, the value of this measure should be zero. A high value could indicate that many of the document processors registered with a content distributor are currently invalid, and hence, unusable. This will compel the content distributor to impose more load on the 'valid' document processors, thus causing a performance slowdown.
	Document processors unused: Indicates the number of document processors registered with this content distributor that are currently unused.	Number	Too many unused document processors indicates ineffective utilization of the available processors. This could cause serious processing delays, especially when the load is high.
	Operation sets submitted: Indicates the number of operation sets submitted to the document processors by this content distributor.	Number	
	Operation received: Indicates the number of operations received by this content distributor.	Number	
	Number of operations: Indicates the number of operations in this content distributor since the last measurement period.	Number	

	Operations rate: Indicates the rate at which the operations were carried out by this content distributor since the last measurement period.	Operations/Sec	A very low value for this measure indicates a processing bottleneck on the content distributor.
	Processing load: Indicates the current processing load on this content dispatcher.	Number	A high value is indicative of a heavy load on the content distributor. Compare the value of this measure across distributors to figure out which distributor is currently handling the maximum load and is probably overloaded.

1.3 Doc Processing Layer

The document processor takes the content that the crawler has collected and sent via the content distributor and organizes it into searchable text by sending the content through the indexing pipeline.

This layer monitors the load on and the processing capability of each document processor configured on the Microsoft FAST Search server.

Figure 4: The test mapped to the Document Processors layer

1.3.1 Document Processors Test

The document processor takes the content that the crawler has collected and sent via the content distributor and organizes it into searchable text by sending the content through the indexing pipeline. As such, the document processor is the mechanism that pushes items through the pipeline in which different processors perform different, specific tasks on the content. Document processors can perform many tasks, such as extracting text from files, breaking the text into words, identifying languages, and extracting entities. By the end of the indexing pipeline, several unique processors are passed and the document is in a FiXML format

that can be stored in the index and made searchable. The Document Processors send the converted document to the Indexing Dispatchers which in turn sends it to the Indexer that generates a searchable Index.

This test reports the current status of each of the document processors configured on the FAST Search 2010 server for SharePoint. This way, you can quickly figure out which document processor is doing what. Additionally, this test reports the number of times the Document Processor is initialized and the number of times the reassignment of the content distributor dispatcher has been completed/started.

Purpose	Reports the current status of each of the document processors configured on the FAST Search 2010 server for SharePoint
Target of the test	FAST Search Server 2010 for SharePoint
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.
Outputs of the test	One set of results for each document processor on the FAST Search Server 2010 for SharePoint farm

Measurements made by the test	Measurement	Measurement Unit	Interpretation
	Indexing dispatcher input queue state: Indicates the current state of the Indexing Dispatcher input queue i.e., the queue through which the converted document is sent to the Indexing Dispatcher by the Document Processor.		This measure reports the value *Full* if there are one/more documents in the Indexing dispatcher input queue and *Not Full* if no documents are there in the queue. The values reported by this measure and their numeric equivalents are available in the table below:

Measure Value	Numeric Value
Full	1
Not Full	0

Note:

This measure reports the **Measure Value** listed in the table above to indicate the state of the Indexing dispatcher input queue. However, in the graph of this measure, the queue state is indicated using only the **Numeric Value**s listed in the above table. |

	Is initializing?: Indicates whether/not this document processor currently initializing.	Number	The values reported by this measure and their numeric equivalents are available in the table below: | **Measure Value**	**Numeric Value**
Yes	1
No	0 | **Note:** This measure reports the **Measure Value**s listed in the table above to indicate whether/not the Document Processor is initializing. However, in the graph of this measure, the initializing state of the Document Processor is indicated using only the **Numeric Value**s listed in the above table.
	Is processing a set of operations?: Indicates whether/not this Document Processor is currently processing a set of operations.		The values reported by this measure and their numeric equivalents are available in the table below: | **Measure Value**	**Numeric Value**
Yes	1
No	0 | **Note:** This measure reports the **Measure Value**s listed in the table above to indicate whether/not the Document Processor is processing a set of operations. However, in the graph of this measure, the same is indicated using only the **Numeric Value**s listed in the above table.

	Is submitting a set of operations to the indexing dispatcher?: Indicates whether/not the set of operations were submitted to the Index dispatcher by this Document Processor.		The values reported by this measure and their numeric equivalents are available in the table below: | **Measure Value**	**Numeric Value**
Yes	1
No	0 | **Note:** This measure reports the **Measure Value**s listed in the table above to indicate whether/not the Document Processor is submitting a set of operations to the index dispatcher. However, in the graph of this measure, the same is indicated using only the **Numeric Value**s listed in the above table.
---	---	---	---
	Is sharePoint pipeline initialized?: Indicates whether/not the sharepoint pipeline is initialized.		The values reported by this measure and their numeric equivalents are available in the table below: | **Measure Value**	**Numeric Value**
Yes	1
No	0 | **Note:** This measure reports the **Measure Value**s listed in the table above to indicate whether/not the sharepoint pipeline is initialized. However, in the graph of this measure, the same is indicated using only the **Numeric Value**s listed in the above table.
	Total number of initializations: Indicates the number of times this document processor was initialized.	Number	

	Reassignments of the contentdistributor dispatcher has been completed: Indicates the number of times the reassignment of the content distributor dispatcher is completed.	Number	
	Nameserver resolve of contentdistributor dispatcher scope: Indicates the total number of invocations of the nameserver resolve of the contentdistributor dispatcher.	Number	
	Nameserver resolve of contentdistributor master scope: Indicates the total number of invocations of the nameserver resolve of the primary contentdistributor.	Number	

1.4 Web Analysis Layer

The Web Analyzer has two main functions: It analyzes *search clickthrough* logs and hyperlink structures. Both contribute to better ranked search results.

With the help of the **Web Analyzer** test, the **Web Analysis** layer reports how well each of the link processing components of the Web Analyzer handle the workload.

Figure 5: The test mapped to the Web Analysis layer

1.4.1 Web Analyzer Test

The Web Analyzer has two main functions: It analyzes *search clickthrough* logs and hyperlink structures. Both contribute to better ranked search results.

Items that show many clicks in the search *clickthrough* log are popular and therefore receive better rank scores than less-viewed items. Items that are linked to from many other items are also perceived to be more relevant for the user and therefore receive better rank scores.

The Web Analyzer improves search relevancy by analyzing the link graph and adding anchor texts and a query independent rank boost based on link cardinality to the items in the index. Anchor texts describe the items they refer to and will improve recall and relevancy when a query term matches the anchor text. Items with many links pointing to them will be ranked higher

The Web Analyzer may scale out to many nodes to reduce the total time that is needed for the analysis. This is done by adding dedicated lookup database components and link processing components that are used during the link analysis.

The link processing component receives tasks from the Web Analyzer during link processing. Large scale installations use multiple link processors.

The lookup database component represents a key/value lookup server that retrieves the link information generated by the link processing. The item processing looks up the link information for an item using the URL as key. Large scale installations use multiple lookup database components.

This test helps administrators analyze the workload of each of the link processing components, and gauge how well each of the lookup/anchor database components will be able to provide link information by reporting the number of sites and URIs that are stored in the database.

Purpose	Helps administrators analyze the workload of each of the link processing components, and gauge how well each of the lookup/anchor database components will be able to provide link information by reporting the number of sites and URIs that are stored in the database
Target of the test	FAST Search Server 2010 for SharePoint
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.
Outputs of the test	One set of results for each link processing component/lookup database configured on the FAST Search Server 2010 for SharePoint farm
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

test	**Contributing links:** Indicates the number of contributing links i.e., hyperlinks involved in the link analysis that is performed by this link processing component.	Number	This is a good indicator of the workload of a link processing component. Compare the value of this measure across components to accurately identify the component with the maximum load. If one/more components are analyzing an abnormally large number of links, it could indicate irregularities in load balancing across the link processing components.
	Sites in the anchor database: Indicates the number of sites in this anchor database of the FAST Search Server for SharePoint 2010 farm.	Number	A high value is typically desired for this measure. If more number of sites and URIs are available in the lookup database, then it increases the probability of a large number of link information requests being served by the database.
	URIs in the anchor database: Indicates the number of URIs (Uniform Resource Identifiers) in this anchor database of the FAST Search Server for SharePoint 2010 farm.	Number	

1.5 The Indexing Service Layer

Using the tests mapped to this layer, administrators can do the following:

➢ Analyze the load to each indexer on the Microsoft FAST Search server and report processing bottlenecks;

➢ Evaluate the processing ability of each indexing dispatcher and indexer partition on the server;

Figure 6: The tests mapped to the Indexing Service layer

1.5.1 Indexer Test

Indexers manage the content to be included in an index, and propagate that content to query servers where they are stored in index files.

Each item that a connector submits for indexing passes through the indexing pipeline, in which a predefined set of modules manipulates the items one by one. At the end of this, the FIXML needed for indexing is produced and handed over to the indexer by the Indexing Dispatcher. The Indexer receives the item and creates binary indexes in the data folder on disk. These binary indexes are then sent to the query matching component and are used during query evaluation during searching.

If too many feeding operations/documents are sent to an indexer for processing, that indexer may be forced to push many of these operations into the API queue, where they will be held until they are taken up for processing. Administrators need to monitor the length of this queue at frequent intervals to determine the processing power of an indexer. Using the **Indexer** test, administrators can keep an eye on the number and nature of documents that each indexer processes, track the growth of the API queues, monitor the FiXML fill rate of every indexer, and thus promptly capture potential overloads, processing delays, processing failures, and other inefficiencies in the functioning of every indexer.

Purpose	Helps administrators keep an eye on the number and nature of documents that each indexer processes, track the growth of the API queues, monitor the FiXML fill rate of every indexer, and thus promptly capture potential overloads, processing delays, processing failures, and other inefficiencies in the functioning of every indexer.

Target of the test	FAST Search Server 2010 for SharePoint		
Agent deploying the test	An internal agent		
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.		
Outputs of the test	One set of results for every indexer configured on the FAST Search Server 2010 for SharePoint farm		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Total number of feed operation: Indicates the total number of feed operations processed by this indexer.	Number	The measure indicates the total number of items that are indexed and updated.
	API operations processed: Indicates the rate at which the feed operations are processed by this indexer.	Operations/Sec	A consistent decrease in the value of this measure could indicate a processing bottleneck.
	API partial update operations: Indicates the number of feed operations processed by this indexer that are partial update operations.	Number	
	API queue load: Indicates the current load on the API queue of this indexer.	Bytes	The API queue is used by the indexer to hold pending operations waiting to be persisted. The value of this measure is obtained by the ratio of the current size of the API queue to the maximum size of the API queue. A low value is desired for this measure.

	API queue size: Indicates the number of batches that are currently in the API queue of this indexer.	Number	Indexers queue incoming work under high load. This is common, especially for partial updates. If API queues never (intermittently) reach zero, the indexer is the bottleneck. The crawler will pause when the queue reaches 256MB in one of the indexers. This can occur if the storage subsystem is not sufficiently powerful. It will also occur during large redistribution of content between partitions, which temporarily blocks more content from being indexed.
	API remove operations: Indicates the number of feed operations processed by this indexer that are remove operations.	Number	
	API status update operations: Indicates the number of feed operations processed by this indexer that are status updates.	Number	
	API remove collection operations: Indicates the number of feed operations processed by this indexer that are remove collection operations.	Number	
	Documents in indexer: Indicates the total number of valid items that are currently contained in the docindex of this indexer.	Number	The docindex is a mapping to the persisted items on disk, so this translates to the number of valid items persisted on disk (FiXML).
	API update operations: Indicates the number of feed operations processed by this indexer that are add/update operations.	Number	
	API failed operations: Indicates the number of feed operations that were not successfully processed by this indexer.	Number	

	Fixml fill rate: Indicates the ratio of the number of valid items in a persisted file to the total number of persisted items.	Percent	FiXML files (internal item storage in the indexers) are compacted regularly, by default between 3am and 5am every night. Low FiXML fill rate (<70%) will lead to inefficient operation.
	Last batch size: Indicates the size of the previous batch that was processed by this indexer.	Bytes	
	Maximum batch size: Indicates the size of the largest batch that was processed by this indexer during the last measurement period.	Bytes	
	Maximum documents per sec: Indicates the maximum number of items that were indexed by this indexer in an index job per second.	Indexed/Sec	By closely observing variations to these measures over time, you can proactively detect potential processing latencies.
	Minimum documents per sec: Indicates the minimum number of items that were indexed in an index job per second.	Number	
	Number of file receivers currently registered: Indicates the total number of file receivers that are currently registered on this indexer.	Number	The file receivers can either be backup indexers or search controllers.
	Number of search controllers currently registered: Indicates the total number of search controllers that are currently registered on this indexer.	Number	The registered search controllers get new indices from over the indexer.

	Search controller stubs currently allocated by the indexer: Indicates the total number of search controller stubs that are currently allocated by this indexer.	Number	The search controller stub is used to communicate with a remote search controller. This measure is a good indicator of the number of search controllers that are communicating with the indexer.
	Search controller allocated count: Indicates the number of times a search controller has been allocated on this indexer.	Number	
	Size of documents: Indicates the size of all the valid items that are persisted on the disk.	Bytes	

1.5.2 Indexing Dispatcher Test

The Indexing Dispatchers handle flow control protocol by routing the processed items to Index servers. A dispatcher is required because a FAST Search for SharePoint 2010 installation may distribute the index over several servers. The index dispatcher needs to know which server to send which items to. If an indexing dispatcher server fails, the outstanding item batches associated with this indexing dispatcher will fail too. If you have redundant indexing dispatchers, the flow control protocol will ensure that the indexing connector resubmits the subset of item batches associated with the failing indexing dispatcher.

This test reports the number of batches processed by each indexing dispatcher.

Purpose	Reports the number of batches processed by each indexing dispatcher
Target of the test	FAST Search Server 2010 for SharePoint
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.
Outputs of the test	One set of results for every index dispatcher configured on the FAST Search Server 2010 for SharePoint farm
Measurements made by the	**Measurement**	**Measurement Unit**	**Interpretation**

test	**Total number of batches processed:** Indicates the total number of batches processed by this Indexing dispatcher.	Number	

1.5.3 Indexer Partition Test

The overall index created by an Indexer has two levels of partitioning: index columns and index partitions.

When the complete index is too large to reside on one server, you can split it into multiple disjoint index columns. Within each index column the indexer uses a partitioning of the index in order to handle large number of indexed items with low indexing and query latency. Index partitioning enables administrators to spread the load for queries across multiple servers. This is achieved by creating subsets of an index, and propagating individual subsets to different index servers. This partitioning is dynamic and handled internally on each index server. When the query matching component evaluates a query, each partition runs in a separate thread. The default number of partitions is 5. In order to handle more than 15 million items per server (column), you must configure a larger number of partitions.

To check whether sufficient indexer partitions are up and running to serve the query load effectively, you need to do the following:

> ➢ Run periodic checks on the state of the individual partitions to identify the unavailable partitions;

> ➢ Continuously monitor the load on the partitions and how quickly each partition processes the load, so that probable overload conditions and processing bottlenecks can be identified and fixed early.

The **Indexer Partition** test helps achieve all of the above.

Purpose	➢ Runs periodic checks on the state of the individual partitions to identify the unavailable partitions; ➢ Continuously monitors the load on the partitions and how quickly each partition processes the load, so that probable overload conditions and processing bottlenecks can be identified and fixed early.
Target of the test	FAST Search Server 2010 for SharePoint
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.

Outputs of the test	One set of results for every indexer partition configured on each query matching server in the FAST Search Server 2010 for SharePoint farm		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Number of active items: Indicates the number of active items on this index partition.	Number	The Index partitions 0 and 1 should have less than 1 million items each, preferably even less in order to keep indexing latency low. In periods with high item throughput, indexing latency will be reduced and these partitions will be larger, as this is more optimal for overall throughput. Items will although automatically be rearranged into the higher numbered partitions during periods with lighter load.
	Number of items indexed: Indicates the number of items that are indexed per second for this partition.	Number	A consistent decrease in the value of this measure could indicate a processing bottleneck.
	Pending exclusion: Indicates the number of items that are to be excluded from indexing in this index partition.	Number	This translates to items that are in the index but that should not be searchable (deleted, present in other partitions, etc).
	Current state of partition: Indicates the current state of this partition.	Number	
	Total number of items: Indicates the total number of items in this partition including the items on the excluded lists.	Number	This measure is a good indicator of the total load on a partition. Compare the value of this measure across partitions to isolate the overloaded partitions. If one/more partitions are found to be handling abnormally high load than the rest, it could indicate that load is unevenly distributed across the partitions. You may hence have to configure more partitions to handle the load.

1.6 Search/Query Processing

Query processing is the stage of the search where the query is received from (for example, the Search Center) and is processed to apply certain rules or expansion before being matched to the content of the index.

To assess the query processing ability of the Microsoft FAST Search server and capture related anomalies, use the tests mapped to this layer.

Figure 7: The tests mapped to the Search/Query Processing layer

1.6.1 FDispatch Test

Query processing is the stage of the search where the query is received from (for example, the Search Center) and is processed to apply certain rules or expansion before being matched to the content of the index. Query processing in FAST Search for SharePoint 2010 is a set of components (QR Proxy, QR Server, and Fdispatch) that receive the query from the FAST Query SSA (FAST Search Query module), process it, and pass it onto the Query Matching component. The query dispatching part (Fdispatch) passes the query to the index columns and eventually to any index partitions. There is also a query dispatcher (Fdispatch) located on each query matching server, distributing queries across index partitions.

If the query load to the dispatchers increase, query queues build up, thereby delaying query processing significantly. Using this test, you can continuously monitor the query load on each of the query dispatchers configured on the query matching servers, observe variations in the length of the query queues, and proactively isolate the following:

➢ a probable overload condition; you can even accurately identify which query dispatcher is handling the maximum load

➢ a potential handle leak and the query dispatcher responsible for the same;

➢ latencies in query processing

Purpose	Continuously monitor the query load on each of the query dispatchers configured on the query matching servers, observe variations in the length of the query queues, and proactively isolate the following: ➢ a probable overload condition; you can even accurately identify which query dispatcher is handling the maximum load ➢ a potential handle leak and the query dispatcher responsible for the same;

	➢ latencies in query processing
Target of the test	FAST Search Server 2010 for SharePoint
Agent deploying the test	An internal agent
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.
Outputs of the test	One set of results for every query dispatcher configured on each query matching server in the FAST Search Server 2010 for SharePoint farm

Measurements made by the test	Measurement	Measurement Unit	Interpretation
	Active running queries: Indicates the number of queries that are currently active on this query dispatcher.	Number	This is a good indicator of the current load on the dispatcher. Compare the value of this measure across dispatchers to identify the dispatcher that is handling maximum load currently.
	Total number of queries dispatched: Indicates the number of queries that have been dispatched by this query dispatcher since the last index set change.	Number	
	Number of open file handles: Indicates the number of file handles that are currently open in this query dispatcher.	Number	The maximum number of open file handles for fdispatch is 1014. If the value of *Number of open file handles* exceeds this limit, it is a cause for concern, as it could indicate a handle leak.
	Max number of open file handles: Indicates the maximum number of open file handles in this query dispatcher.	Number	

	Number of queued queries: Indicates the number of queries to this query dispatcher that have been temporarily put on hold by adding them to a wait queue.	Number	Ideally, the value of this measure should be low. A high value or a consistent increase in this value is indicative of latencies in query processing or an overload condition.

1.6.2 QRServer Test

The Query & Results Server (QR Server) is the component on the search server that is responsible for executing searches in the FAST Search for SharePoint 2010 farm. The QR Server also provides its own web-based interface that is mainly intended for testing FAST Query Language (FQL) expressions and debugging any potential issues with the FAST Search for SharePoint 2012 farm.

QR Server is responsible for preparing queries to be sent to the Search Nodes and refines the results before they are returned to the calling Search Front End Server (SFE). Query transformation includes spell checking, query-side lemmatization, query-side synonym expansion, anti-phrasing, stop work removal. It is applied to ensure that the best possible query is submitted. Some of this processing can be controlled by providing parameters with the query.

This test measures the query processing ability of the QR Server. The test monitors the queries to the server, swiftly captures query failures, and promptly reports latencies in query processing.

| Purpose	Measures the query processing ability of the QR Server		
Target of the test	FAST Search Server 2010 for SharePoint		
Agent deploying the test	An internal agent		
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.		
Outputs of the test	One set of results for the FAST Search Server 2010 for SharePoint farm		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	System failed queries: Indicates the rate at which the system queries failed.	Queries/Sec	Ideally, the value of this measure should be 0.

	Total failed queries: Indicates the rate at which the queries failed.	Queries/Sec	Ideally, the value of this measure should be *0*.
	User failed queries: Indicates the rate at which the user queries failed.	Queries/Sec	Ideally, the value of this measure should be *0*.
	Number of queries: Indicates the number of queries handled by this QR Server per second.	Queries/Sec	A consistent decrease in the value of this measure is a cause for concern as it could indicate a processing bottleneck on the QR Server.
	Number of requests: Indicates the current number of requests received by this QR Server per second.	Queries/Sec	In addition to the query load i.e., query requests, one internal request is received every second to check whether the QRserver is active to execute the search.
	Avg document sum latency: Indicates the average document sum latency of this QR Server.	Millisec	A low value is desired for this measure.
	Average latency: Indicates the average latency of this QR Server.	Millisec	A high value or a steady increase in the value of this measure is indicative of a query processing bottleneck on the QR server.
	Average latency of the last queries: Indicates the average latency of the last query i.e., the average time the search requests take to respond to the requesting client.	Millisec	A high value is indicative of a performance bottleneck or a slowdown of the QR Server.
	Average search latency: Indicates the average search latency of this QR Server.	Millisec	A low value is desired for this measure.
	Number of highest registered queries: Indicates the highest number of queries registered for search in this QR Server per second.	Queries/Sec	

1.6.3 Dataset Test

This test reports the time taken by the indexer to service search queries, and thus sheds light
on bottlenecks in query processing in the indexer. In the process, the test also reports the
number of active search nodes and active index partitions.

Purpose	Reports the time taken by the indexer to service search queries, and thus sheds light on bottlenecks in query processing in the indexer		
Target of the test	FAST Search Server 2010 for SharePoint		
Agent deploying the test	An internal agent		
Configurable parameters for the test	1. **TEST PERIOD** – How often should the test be executed 2. **HOST** – The host for which the test is to be configured 3. **PORT –** Refers to the port used by the Citrix Provisioning server. By default, this is 54321.		
Outputs of the test	One set of results for each index partition on the FAST Search Server 2010 for SharePoint farm		
Measurements made by the test	**Measurement**	**Measurement Unit**	**Interpretation**
	Active nodes: Indicates the number of active search nodes i.e., the number of active query matching processes (fsearch processes).	Number	Search Nodes are the processes that perform queries against Index Nodes. There will always be at least one Search Node for every Index Node. The Search Node will only search the Index Node that it is assigned to. We will go into the details of topology of Index and Search nodes when we discuss scaling FAST. Within a Search Node, there is a process called *fSearch* which is created for each index partition within the Index Node. The *fSearch* process searches the index partition for matching documents. Then a single process within the Search Node called FDispatch takes all the results from each index partition and merges them into a single result that is ranked and sorted appropriately based on the rules specified in the Index Profile.
	Active partitions: Indicates the number of active index partitions.	Number	

	Average searches rate: Indicates the average search rate of the documents in the Indexer.	Searches/min	A low processing rate could be indicative of a potential slowdown.
	Average seconds per search: Indicates the average time taken to complete the last analysis.	Secs	A consistent rise in the value of this measure could indicate a processing bottleneck.
	Average uncached search time: Indicates the average time taken by a query to perform a search operation without looking into the query cache.	Secs	A consistent rise in the value of this measure could indicate a processing bottleneck.
	Timeout percentage: Indicates the percentage of query timeouts that had occurred in the Indexer.	Percent	The timeouts may have occurred due to the documents that were skipped or due to connection timeout.
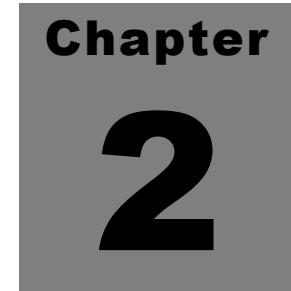
Chapter

2

Conclusion

This document has described in detail the monitoring paradigm used and the measurement capabilities of the eG Enterprise suite of products with respect to **FAST Search for SharePoint 2010**. For details of how to administer and use the eG Enterprise suite of products, refer to the user manuals.

We will be adding new measurement capabilities into the future versions of the eG Enterprise suite. If you can identify new capabilities that you would like us to incorporate in the eG Enterprise suite of products, please contact support@eginnovations.com. We look forward to your support and cooperation. Any feedback regarding this manual or any other aspects of the eG Enterprise suite can be forwarded to feedback@eginnovations.com.