# Knowledge Article

Configuring FTS for performance in a server group

|  |  |  |
| --- | --- | --- |
| Manage Document (Fix It) | [Manage Document (Fix It)](https://kb.bmc.com/InfoManager/WebObjects/InfoManager.woa?params=8F3A46A403B062ACE368DDEB8274A48A8815B00D5F63A832FA98923C72ACF55145A38427E036D86075B5461B0E6D640D610C22F79B34D3DCA3CC7C1DFAF12EA5FE96AFCC51674A871BFBC2E4D4DE40F040F5E6ED2988841F9AAA086956C95B8CAD48C79D29A58C2C5CCD733BBFA7D9F51E696CF0F79076B763F39C9BD3F0C46536FBACD1A0F1A5A80BEFE345C187E4D6B148C74777D2155295178C317CD19BC0D298C78422864B509512C45F7B3208D1C8B406C5F630B6CE&action=MANAGE&localecode=en_US&contentid=09012205de31d7201352f7153ad0067f0) |   |
| **Knowledge Article ID:** |    | KA363429 |
| **Version:** |    | 2.1 |
| **Available To:** |    | External |
| **Status:** |    | Unpublished |
| **Updated:** |    | 05/24/2012 |
|  |  |  |  |

### Problem

Configuring FTS for performance in a server group

### Solution

This document will describe an alternative configuration for FTS in a server group from that published in the AR documentation.  It has been determined that a considerable performance increase can be had by following this alternative configuration.

This document will discuss: Basic information about how FTS within AR works, How to configure for better performance, and how to manage failover situations.

# How FTS works

FTS within AR is made up of three primary components.  There is the code that exists within the AR server, which manages the FTS functionality within AR.  There is an AR Java plug-in, referred to as the FTS plug-in, which is the core index engine, and then there is the index.

As events occur within AR that that cause data to need to be indexed, AR sends that data, along with appropriate instructions, to the FTS plug-in which then adds, deletes, or updates data within the index.  Other events request a search of the index and AR will then send the search request to the FTS plug-in, which will perform the search, and return the results to AR, which will then be able to deal with the data accordingly.

In a properly configured server group, only the FT Indexer server (as defined in the “AR System Server Group Operation Ranking” form) is designated as the writer instance in the group.  All server instances can read the index.  The events that cause data to need to be written to the index result in data being put into the AR database as a queue of items to index.  Only the FT Indexer server looks to this database table and processes these, sending the indexing instructions to its corresponding FTS plug-in.  This ensures index integrity.

To further ensure integrity of the system, when all instances of the FTS plug-in are initialized they are all first initialized as read only instances.  Once the FT Indexer server in the group has identified itself then it re-initializes its instance of the FTS plug-in as a writer instance which can perform writing tasks to the index.  None of the other servers, or their corresponding FTS plug-ins, can perform any writing tasks.

Any instance of AR can send a search request to its corresponding FTS plug-in.

In the documented configuration for FTS in a server group, all instances of the FTS plug-ins reside local to their instance of AR, and they all point to a central location on the network where the index resides.  All instances of the FTS plug-in have to communicate with the index files over the network.  This has been identified as the performance bottleneck.  It has been found that performance is greatly increased when all instances of the FTS plug-in and the index files are local to each other.

AR is designed so that it can communicate with an AR plug-in sitting in any location on the network.  They do not need to be local to one another.  AR will make Remote Procedure Calls (RPC) to whatever plug-in it is configured to communicate with.

In order to achieve optimal performance, all instances of the FTS plug-in are moved to a location where the FTS plug-ins and the index files are local to each other. This allows the index intensive activity to happen local rather than over a network and then allows AR servers to make RPC calls to the appropriate FTS plug-in over the network, which activity is far less intensive and is not dependant on disk activity.

# Configuration

This section will describe several means of configuration to achieve this.  The first method represents the most commonly used configuration.  The alternative configurations have been less tested at this point in time but should work.  Each organization needs to test and evaluate any configurations based on their needs.

## Common configuration

This configuration puts all instances of the FTS plug-in onto the FT Indexer server (Primary) in the group, and puts the FTS index local on that FT Indexer server’s local drive.

### Primary FT Indexer server configuration

1.       Establish a location on the primary FT Indexer server for the index files.  On a default Windows install of AR this is: C:\Program Files\BMC Software\ARSystem\ftsconfiguration\collection.

1.1.    Set the chosen location in the AR Server Information form, on the FTS tab (AR System Administration Console, System, General, Server Information, FTS) in the FTS Collection Directory field.  The path given should be a local path, not a UNC.

2.       Create multiple instances of the FTS plug-in on the server.

2.1.    The current FTS plug-in resides at the following location in a default Windows install of AR (adjust accordingly) C:\Program Files\BMC Software\ARSystem\pluginsvr\fts

2.2.    The current “fts” folder will be the instance of the FTS plugin for the primary machine.  Copy this “fts” folder and create sibling folders for every other instance of AR in the group.  Name them something like fts2, fts3, etc. Or you can name them with something that will identify the AR server in the group that this instance of the plug-in will service.

2.2.1.  Edit the file “pluginsvr\_config.xml” that exists in each of these “fts” folders.  The following settings that need to be changed are found at the bottom of the file.

2.2.1.1.              The primary FTS plug-in will have <port> setting of: 9998.  Leave this one as is for the primary server, but for each additional fts plugin folder that was created, change this port setting to be something new, and **that is not in use on the current server**.  For example, you could simply make each one a successive number. 10000, 10001, etc.  Each instance of FTS being configured on this server needs to have its own port number, so each pluginsvr\_config.xml file that is edited will have a different port number configured in it.

2.2.1.2.              Make sure that the <ftCollectionDir> setting is pointing to the collection directory as defined in step 1 and 1.1.

2.2.2. Edit the file log4j\_pluginsvr.xml found in the same directory as previous file.

2.2.2.1.              Change the filename for the items: <param          name="File" value="C:/Program Files/BMC Software/ARSystem/ARServer/Db/arftsplugin.log" /> and <param        name="File" value="C:/Program Files/BMC Software/ARSystem/ARServer/Db/arfts.log" /> so that each instance of FTS plug-in will have unique log file names.  Give them names that will help identify each server that they will be associated with.  This will help to indentify which log is which in the log directory once the system is running.

2.2.3. Edit the armonitor.cfg file to tell AR to start these other instance of FTS.

2.2.3.1.              In a default Windows install of AR this file is found in: C:\Program Files\BMC Software\ARSystem\Conf.

2.2.3.2.              Locate the line that is something like: "C:\Program Files (x86)\Java\jre6\bin\java" –Xmx1024m -classpath "C:\Program Files\BMC Software\ARSystem\pluginsvr\fts;C:\Program Files\BMC Software\ARSystem\pluginsvr;C:\Program Files\BMC Software\ARSystem\pluginsvr\arpluginsvr7604\_build002.jar" com.bmc.arsys.pluginsvr.ARPluginServerMain --unicode -x WIN-1V9GUMROM18 -i "C:\Program Files\BMC Software\ARSystem" –m”

2.2.3.2.1.                     Copy/Paste this line as many times into the file as there are actual fts plugins being setup on this server.

2.2.3.2.2.                     Edit the value for the –Xmx parameter as needed.  The suggested values are for the primary instance of FTS to have a value of –Xmx2048m, and for all other instances to have –Xmx1024m.  Make sure the server has this much memory to dedicate to these plugins.  The reason the primary instance is larger is that this is the fts plugin instance that will be doing indexing and the extra memory can be of value to that task.  If enough memory is not available on the server to dedicate this amount, then cut the values in half.

2.2.3.2.3.                     Edit the value following the –classpath parameter to have the path to the corresponding folders.  Ie, one would be …\pluginsvr\fts, another \pluginsvr\fts2, \pluginsvr\fts3, etc.  (with fts2 and 3 being the actual names you named the folders in step 2.2.  And then save the file.

### Secondary server(s) configuration

1.       For each of the secondary servers in the group, edit the file “armonitor.cfg” that resides on the actual secondary server(s) (as opposed to the one that was edited on the primary server) found on each respective server.

1.1.    Find the line in reference to FTS (you can search for fts) and place a # at the beginning of the line to comment that line out.  Save the file.  This will prevent AR from starting the local instance of the FTS plug-in on this server.

2.       For each of the secondary servers in the group, edit the file ar.cfg found in the same directory as armonitor.cfg.

2.1.    Locate the line that is similar to Server-Plugin-Alias: ARSYS.ARF.FTS ARSYS.ARF.FTS WIN-1V9GUMROM18.localdomain:9998.  This is the configuration line that tells AR where to communicate with the fts plug-in.  Change the value that corresponds to this sample WIN-1V9GUMROM18.localdomain:9998 to have the name of the primary server (red) as AR will now be communicating with the corresponding fts plug-in now on the primary server.  And change the port to be the same as one of the fts plug-ins created on the primary server (blue).  In the end, each secondary server will be pointing to its own instance of the fts plugin on the primary server.  No two servers will be going to the same port.

3.       For each of the secondary servers in the group go to AR System Administration Console, System, General, Server Information, FTS and in the collection directory field put the same value as you put in the primary machine.  In step 1 of setting up the primary server.

Now shut down all instances of AR in the group and bring up the primary server and then bring up the secondary servers.  The primary server will communicate with the primary plug-in that exists on the same server.  Each of the secondary servers will communicate with one of the secondary plug-ins created on the primary server.

# Alternative configuration

Rather than configuring an FTS plug-in instance on the primary server for each server in the group, only two FTS plug-ins are configured on the primary server, one is the writer instance and the other the searching instance (although the writing instance can search too).  Just as in the previous configuration, the primary server will have its own instance of the FTS plug-in.  Then only a second FTS plug-in is configured on its own port and all other instances of AR are configured to use the port of this second FTS plug-in.  In other words, all other instances of AR will share this one searcher instance of the FTS plug-in.   Configure this in the same manner you would any of the secondary FTS plug-ins as described previously.  It is advised that in the pluginsvr\_config.xml file associated with the “searcher” FTS plug-in that the number of threads be increased to handle the greater load from all of the servers.

A variation of this configuration, when a large number of AR servers in the group exist, then a “search” FTS plug-in can be configured for every two or three servers in the group.  So in a system with 9 servers in a group, one is the primary and will have its own FTS plug-in that is the writing instance and then three other FTS plug-ins can be configured that service three servers each.  Each corresponding secondary AR server will need to be configured to go to one of the FTS plug-ins configured.

# Dedicated FTS server alternative

A variation that can be made to either of the above configurations is to dedicate a server solely to FTS. Rather than placing the FTS plug-ins and the index on the primary machine, these can all be setup on a server that does not have an instance of AR and is dedicated to processing FTS requests against the index.  This configuration might be considered where it is known that excessive stress will be placed on the FTS index due to large number of servers in the group and heavy indexing needs.

See the AR documentation for how to run a plug-in server manually when setting up a dedicated server for this purpose.

# Failover

At this moment in time this configuration does not provide a clean failover scenario when configured this way.  But it is felt the benefits far outweigh the risk.  However, this is not much different than with how things worked using the old FTS or RKM (pre 7.6) which used SearchServer.  The failover issues are much the same.  The following describes methods for managing failover.

## Configuring

         Configure the secondary server, which will take over the duties of the FT Indexer server, to have a corresponding set of plugins configured, but either idle (because none of the other servers are configured to communicate with it) or with the plugins commented out in the armonitor.cfg file so that they can be started when needed.

o   Optional - It may be desired to not provide a ranking in the “AR System Server Group Operation Ranking” form for any of the secondary servers.  Since the failover requires manual intervention to reconfigure things. By not providing any failover server will ensure that all FTS processes stop on the failure of the primary machine.  Once things are reconfigured to the failover machine then things can be brought up again.

         Make a copy of the index to this secondary machine on a nightly basis (or have some other backup of it which can be quickly restored).

## Switching over

When a failover situation occurs, do the following manual processes:

         If the “AR System Server Group Operation Ranking” form only has the original FT Indexer server defined in the rankings, and no others (as per the optional item in the configuring section), then this will need to be changed to reflect the failover server as the FT Indexer server.

         If only the AR server has failed, but the physical server is still functional, copy the index from the primary machine to the new failover FT Indexer machine so that it is up to date and ready to be used.

o   An alternative to this, rather than copying the index from the old primary to the failover FT Indexer is to just to leave the index in place and have the failover configurations point to this index.  This will now have the system in a configuration where the FTS plug-ins are now dealing with the index over a network, and so a slower performance may occur, but if the outage is expected to be short lived it may be that this is a more viable option than copying the index to the new primary machine if the index is large.

         If the physical machine is down and the original index is not available, then you will utilize the nightly copy of the index that is on the secondary machine (or restore a nightly backup from wherever it is done.

         Reconfigure the remaining servers to communicate with the fts plugins on the failover FT Indexer machine and restart AR to pickup this change.

         If the nightly backup of the index had to be used instead of the current one, then a rebuild of the index should be issued after coming up in the new configuration.  The index that is out of date as per the last backup will be able to be used as AR rebuilds the index and brings it up to date.  But note that some performance loss may be experienced while the index is being rebuilt.