

Think twice before permission change



Filip Břuska

Senior Software Developer

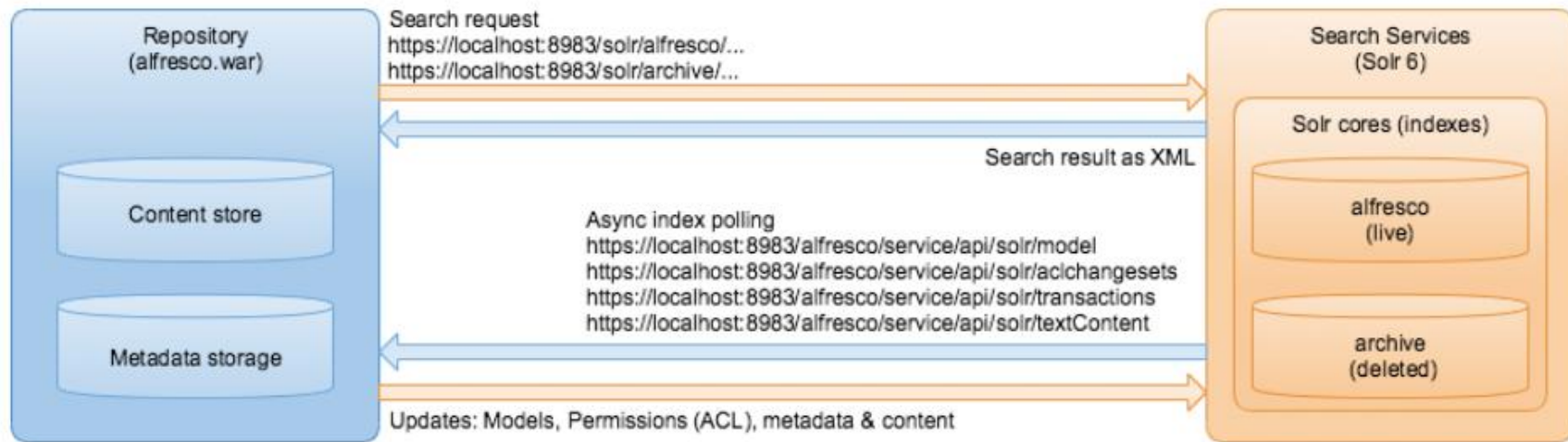
Agenda

- About me
- Transactions and Solr
- Transactions created in vanilla Alfresco
- Bulk Import tool
- Permission change
- Prevention
- Fixing
- Transaction Splitter
- Asynchronous permission inheritance

Disclaimer

- In this talk, I will use SELECT DB queries in order to see what is going on in Alfresco.
- **Do NOT change anything directly in Alfresco DB**, use prepared services such as NodeService or REST API.

Transactions and Solr



- **transaction_id** column is used for identifying nodes in same transaction
- Picture is used from [Alfresco docs](#)

Transactions in vanilla

- **SELECT transaction_id**, count(transaction_id) FROM alf_node group by transaction_id having count(transaction_id) > 100 order by 2 desc

ID: "6", Count: "565" -- *Default folder structure (Company home, Data Dictionary, ...)*


ID: "11", Count: "143" -- *Site swsdp (Sample: Web Site Design Project)*

ID: "15", Count: "130" -- *act:actionparameter
(/app:company_home/app:dictionary/app:rendering_actions/ – can be seen
using Node Browser)*

- More info about nodes in one transaction:
 - **SELECT *** FROM alf_node_properties where node_id IN (SELECT id FROM alf_node where transaction_id = "6")

Bulk Import

- Useful tool for creating testing data in Alfresco.
- Almost 20K was imported (10K folders and 9.9K files) in 3.5 mins
- In-place import is **Enterprise** feature

 Bulk Filesystem In-place Import Tool

Alfresco Enterprise v5.2.5 (re0f49095-b17)

Store-relative import directory path :

Content Store :

Target repository path :

Disable rules: ☐ (unchecked means rules are enabled during the import)

Existing file mode:

When a file being imported already exists in the repository

- ☒ skip (don't import) the file
- ☐ replace the file in the repository with the imported file
- ☐ create a new version of the file with the imported content

Batch Size:

Number of Threads:

 Bulk Filesystem Import Tool Status
Alfresco Enterprise v5.2.5 (re0f49095-b17)

General Statistics				
Current status:	Idle			
Successful:	Yes			
Batch Size:	1,000			
Number of threads:	4			
Source Directory:	/opt/alfresco/alfresco-5.2.5/alf_data/contentstore/import			
Target Space:	/Company Home/Import			
Start Date:	2020-02-13 09:33:07.677AM			
End Date:	2020-02-13 09:36:35.182AM			
Duration:	0d 0h 3m 27s 504.966ms			
Number of Completed Batches:	22			
Source (read) Statistics				
Scanned:	Folders	Files	Unreadable	
	10,000	9,901	0	
Read:	Content	Metadata	Content Versions	Metadata Versions
	9,901 (613.864B)	0 (0B)	0 (0B)	0 (0B)
Throughput:	96 entries scanned / sec 47 files read / sec 2.96kB / sec			
Target (write) Statistics				
Space Nodes:	# Created	# Replaced	# Skipped	# Properties
	9,999	0	0	79,992
Content Nodes:	# Created	# Replaced	# Skipped	Data Written # Properties
	9,901	0	0	0B 89,109
Content Versions:	# Created	Data Written	# Properties	
	0	0B	0	
Throughput (write):	96 nodes / sec 0B / sec			

Initiate another import

Initiate another in-place import

- <https://docs.alfresco.com/6.2/concepts/bulk-import-importing.html>

Transactions after Bulk Import

- You can see many transactions with size of 1000 (corresponding to batch count)
 - So far so good
- You can see how cm:content is defined if BulkFS In-place import is used
 - Import folder as top folder

Query Editor Query History

```

1 SELECT transaction_id, count(transaction_id) FROM alf_node
2 group by transaction_id having count(transaction_id) > 1000 order by 2 desc
3 --SELECT * FROM alf_node_properties where node_id IN (SELECT id FROM alf_node
4

```

Data Output Explain Messages Notifications

	transaction_id bigint	count bigint	
1	49	1000	
2	50	1000	
3	51	1000	
4	34	1000	
5	35	1000	
6	36	1000	
7	37	1000	
8	38	1000	
9	39	1000	
10	40	1000	
11	41	1000	
12	42	1000	
13	52	1000	
14	44	1000	
15	45	1000	
16	46	1000	
17	47	1000	
18	48	1000	
19	53	900	
20	43	900	
21	6	564	
22	11	142	
23	15	130	

Name	Type	Value
cm:created	d:datetime	04 Aug 2020 10:00:13 GMT-0400 (EDT)
cm:title	d:mltext	ll.txt
cm:creator	d:text	admin
sys:node-uuid	d:text	1329fee1-534c-41d0-af26-daba78bffb6c
cm:name	d:text	ll.txt
sys:store-protocol	d:text	workspace
cm:content	d:content	contentUrl=store://Import/7/77/7756/ll.txt mimetype=text/plain size=62 encoding=UTF-8 locale=en_US_ id=1269
sys:store-identifier	d:text	SpacesStore
sys:node-dbid	d:long	11884
sys:locale	d:locale	en_US
cm:modifier	d:text	admin
cm:modified	d:datetime	04 Aug 2020 10:00:13 GMT-0400 (EDT)

Permission change

- Till now Import folder is readable by everyone
 - Default permissions (Everyone is consumer)
- Let's change permissions (Add Alice or Mike)
 - Share gets stuck on Save button
 - Load to DB, indexing is running for all nodes under Import folder
 - Let's see transactions -> **All nodes are in one transaction**

Query Editor

Query History

1

2

3

SELECT transaction_id, count(transaction_id) FROM alf_node group by transaction_id having count(transaction_id) > 100 order by 2 desc

Data Output

Explain

Messages

Notifications

transaction_id

bigint

count

bigint

1

49

19901

2

6

564

3

11

143

4

15

130

Permission change

- Under the hood all nodes need to be updated with new ACLs (acl_id column in alf_node table)
 - Import (parent) folder has so called DEFINING ACL
 - All remaining nodes has SHARED ACL
 - Following query returns info about used ACL
 - `select * from alf_access_control_list where id in (select acl_id from alf_node where uuid = 'XYZ')`
 - More info: <https://docs.alfresco.com/6.2/concepts/secur-acl.html>

DEMO 1

When this problem happens?

- Changing permissions either by adding or removing person or group
- Changing permission inheritance
- Changing site visibility, f.e. from public site to private site

Problems caused by big transactions

- Out of Memory error in Solr
 - <https://issues.alfresco.com/jira/browse/MNT-18308>
 - Problems with full reindex
- Significant delay in indexing newly created content
 - At first big transaction needs to be processed
 - In case of folder structure with more than 1M nodes, it can take several hours to complete indexing
 - If you rely on search functionality in your customizations, it can cause big issues

Prevention

- Use pre-created site groups
 - Instead of ACL on folder level, use site groups as much as possible
- Update ACL of top folder before migration
 - If not using site, good approach is to disable permission inheritance and add groups (Consumer, Contributor, ...) responsible for specific data (f.e. SAP_Consumer for SAP)
 - Only first ACL change will cause problem with transactions
- Define permissions during node creation
 - <https://api-explorer.alfresco.com/api-explorer/#!/nodes/createNode>

Fixing - Intro

- Idea is to fix permissions **from bottom to top** on **folder** level
- Recommendation would be to use this fix on folders with more than 100K nodes
- Solution contains 2 webscripts and several shell scripts:
 - At first, let's run webscript that will go through folder structure and create report of folders with level of depth
 - Several shell scripts that will process input files for another webscript
 - Webscript that will fix folder's permission one by one

Fixing - Report

- Report webscript – [Github](#):
 - org.alfresco.repo.jscrip.ScriptLogger=INFO

```
1  var topFolder = companyhome.childByNamePath("Import").getNodeRef();
2  logger.info("ACL Report ||| START - " + topFolder);
3  var level = 0;
4
5  findSubFolders(topFolder, level);
6
7  function findSubFolders(nodeRef, level) {
8      var node = search.findNode(nodeRef);
9      logger.info("ACL Report ||| " + node.getNodeRef() + " | " + node.displayPath + "/" + node.getName() + " | " + level++);
10     var kids = node.childFileFolders(false, true);
11
12     for (var i = 0; i < kids.length; i++) {
13         findSubFolders(kids[i].getNodeRef(), level);
14     }
15 }
16
17 logger.info("ACL Report ||| END - " + topFolder);
```

Fixing - Permission change

- Use report and call "fix" webscript – [Github](#):
 - In order to keep props such as cm:modifier, behaviors are disabled

```
1  var ctx = Packages.org.springframework.web.context.ContextLoader.getCurrentWebApplicationContext();
2  var filter = ctx.getBean('policyBehaviourFilter', org.alfresco.repo.policy.BehaviourFilter);
3  filter.disableAllBehaviours();
4  try {
5      var node = search.findNode(args.noderef);
6      if (node.inheritsPermissions() && ("ALLOWED;GROUP_EVERYONE;Consumer".equals(node.getPermissions())) {
7          node.setInheritsPermissions(false);
8          node.setPermission("Coordinator", "GROUP_ImportCoordinator");
9          node.setPermission("Consumer", "GROUP_ImportConsumer");
10         node.save();
11     }
12 } catch (error) {
13     logger.error("FIX ACL error - noderef = " + args.noderef + " ||| error = " + error.message);
14 }
15 filter.enableAllBehaviours();
```

Fixing - Shell scripts

- Several shell scripts connect the result from report and use that input for fixing nodes - [Github](#):
 - prepareFiles.sh
 - Parse alfresco log to get folders and their level
 - generateTicket.sh
 - Helper script to get Alfresco ticket
 - After some investigation, I realized it shouldn't be needed because ticket validation is extended with each request
 - runAclFix.sh
 - Reading input from prepare files and fixing nodes from lowest level (greatest number) to top level (smallest number)

DEMO 2

Transaction Splitter

- Alfresco module to reduce the number of nodes in a single transaction
 - <https://bitbucket.org/xenit/transactionsplitter>
 - <https://bitbucket.org/xenit/transactionsplitter/src/master/src/eu/xenit/alfresco/tools/transactionSplitter/TransactionSplitter.java>
 - <https://github.com/y-melo/Alfresco-transaction-splitter>
- You need to provide list of NodeRefs and number of nodes per transaction
- Implementation adds and removes dummy aspect

Transaction Splitter - How to use



- Create a folder under repository and apply the below rule on that folder
- Select "**splitTransaction**" from the Perform Action drop down
- Number of nodes per transaction: 100
- Collect the NodeRefs from the DB in a text file and upload the same file under rule bounded folder created by you.
- It will take some time to split the transactions. From my testing 1000 NodeRefs should not take more than a minute.
- **Don't collect more than 100,000 NodeRefs to split at a time.** It may cause java out of memory issue.

Repository >  Splitter


Splitter: Rules


✓ Inherit Rules

New Rule

Run Rules...

Rules from Splitter

 Drag rules and click Save to reorder.

1	✓ 	Transaction Splitter
---	---	----------------------

Save

Reset

Edit

Delete

Transaction Splitter

Description:

- ✓ Active
- ⊗ Run in background
- ⊗ Rule applied to subfolders

When:

Items are updated



If all criteria are met:

All Items



Perform Action:

splitTransaction List of NodeRefs: Number of nodes per transaction: 100

Async permission inheritance

- Alfresco contains method that can be used to disable permission inheritance asynchronously
 - Implementation can be found on GitHub [here](#) and [here](#)
 - More info and description how to use that is in Jira <https://issues.alfresco.com/jira/browse/MNT-15368>
 - Results were not the best so I used my custom "bottom-up" approach
 - Possible PICNIC error ("problem in chair, not in computer") ☺

Lessons learned

- Think of permission changes in advance especially before data ingestion
- Use groups (f.e. predefined site groups) if possible
- Review of big transactions on regular basis
 - Good to check before full reindexing

tieto *EVRY*