Improving the Capabilities of FutureLens

Joshua Strange
June 29th, 2012

Motivation

- Add new features
- Allow for larger data sets to be processed
- Decrease the amount of time to process data

Background of FutureLens

- Created by two of Dr. Berry's Students
- A visualization tool for data mining
- Written in Java
- Uses the Standard Widget Toolkit (SWT)
- Last updated in 2010

General Obstacles

- Two obstacles to resolve before FutureLens could be improved
- Problem 1: How to get FutureLens working again
- 32-bit application on a 64-bit machine
- Temporary solution: JVM flag
- Finding the permanent solution
- Permanent solution: Upgrading SWT

General Obstacles (cont.)

- Problem 2: The system created menu no longer worked
- Problem only existed on Mac
- The options in this menu are important
- Change in system API
- Solution to problem: ArmListner

Added Features

- External Stop List
- Custom User Dictionary

External Stop List

- A stop list is a list of terms that are ignored during the processing of data
- Behavior in the Original Version of FutureLens
- Not guaranteed to save
- Importing and exporting of stop list

External Stop List (cont.)

- Demonstration of external stop list now included in FutureLens

Custom User Dictionary

- Allows the user to customize the dictionary generated by FutureLens
- Created in 3 different ways
- Helps the user in multiple ways
- Demonstration

How data was created

- Two data sets used
- Psych Abstracts with an average size of 4 KB per file
- Patent Documents with an average size of 45 KB per file
- JVM with starting memory of 1.5 GB and maximum memory of 2 GB

Data Capability of Original Version of FutureLens

- 75,000 Psych abstracts
- 5,000 Patent Documents
- Limit caused by two different structures
- Hash tables
- Strings

Size of Top Two Memory Users in the Original Version of FutureLens

	Total size in bytes of hash table entries	Total size in bytes of strings
500 – 2.1 MB	1897728	5061104
1k – 4.2 MB	3576512	10065200
2k – 8.2 MB	8213184	22906024
3k – 12.5 MB	12611328	35620576
4k – 16.4 MB	15723904	43210736
5k – 20.9 MB	18934688	53425800
10k – 41 MB	38432160	107248408
25k – 104.4 MB	89101952	257423216
50k – 209.3 MB	164210112	497489328
75k – 313.9 MB	232510272	715471040

Generated Using Psych Abstracts

How Data Capability was Increased

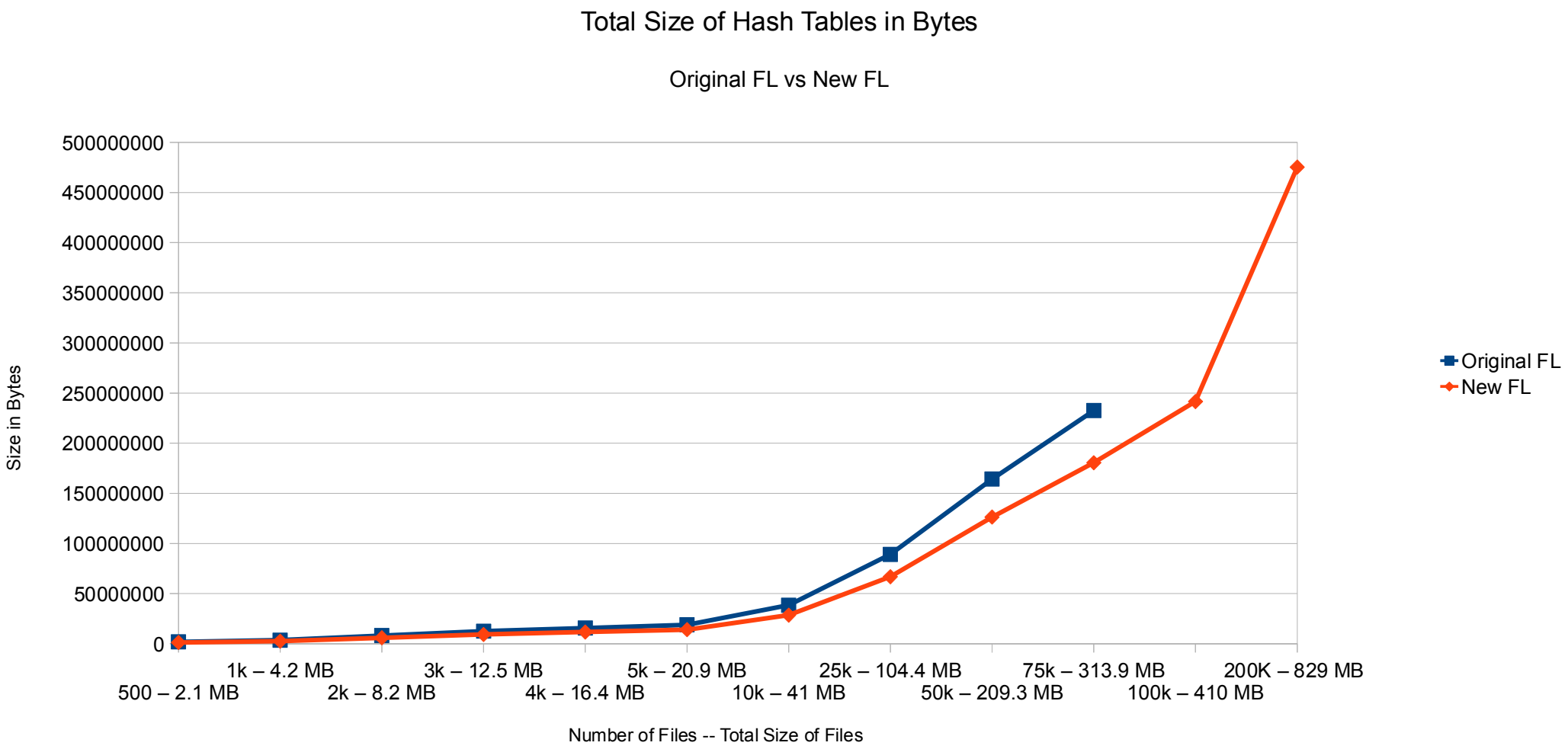
- Investigation into hash tables
- All hash tables have their key as a string
- Relationship between top two memory users
- Hash tables removed in multiple places
- In total 5 hash tables removed
- Limit on data set size now doubled

Size of Top Two Memory Users in the New Version of FutureLens

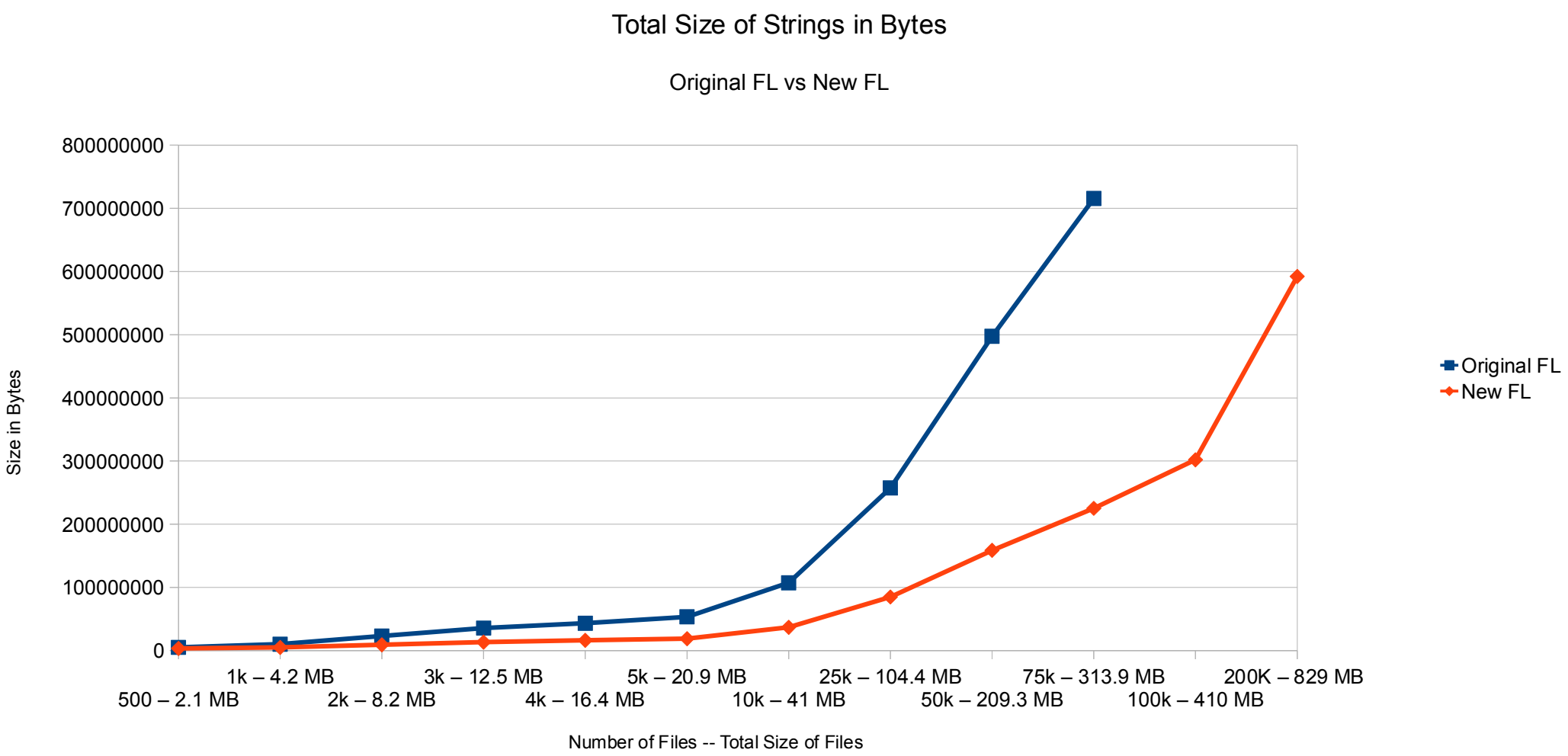
	Total size in bytes of hash table entries	Total size in bytes of strings
500 – 2.1 MB	1345120	3339496
1k – 4.2 MB	2596896	4929800
2k – 8.2 MB	6023552	9130040
3k – 12.5 MB	9312608	13240560
4k – 16.4 MB	11674368	16168048
5k – 20.9 MB	14055360	18838344
10k – 41 MB	28547008	36796640
25k – 104.4 MB	66867616	84901720
50k – 209.3 MB	126339648	158654776
75k – 313.9 MB	180497152	225158688
100k – 410 MB	241508064	301931832
200K – 829 MB	475282624	592066328

Generated Using Psych Abstracts

Data Capability Compared



Data Capability Compared (cont.)



Data Processing Time

- The average time for the original version of FutureLens to process each data set

	Average Time
500 – 2.1 MB	1.1595837
1k – 4.2 MB	1.5336145
2k – 8.2 MB	2.6333681
3k – 12.5 MB	4.4070547
4k – 16.4 MB	5.4321064
5k – 20.9 MB	6.302783
10k – 41 MB	13.2505653
25k – 104.4 MB	41.8450884
50k – 209.3 MB	129.678864
75k – 313.9 MB	259.3683904

Psych Abstract Data Set

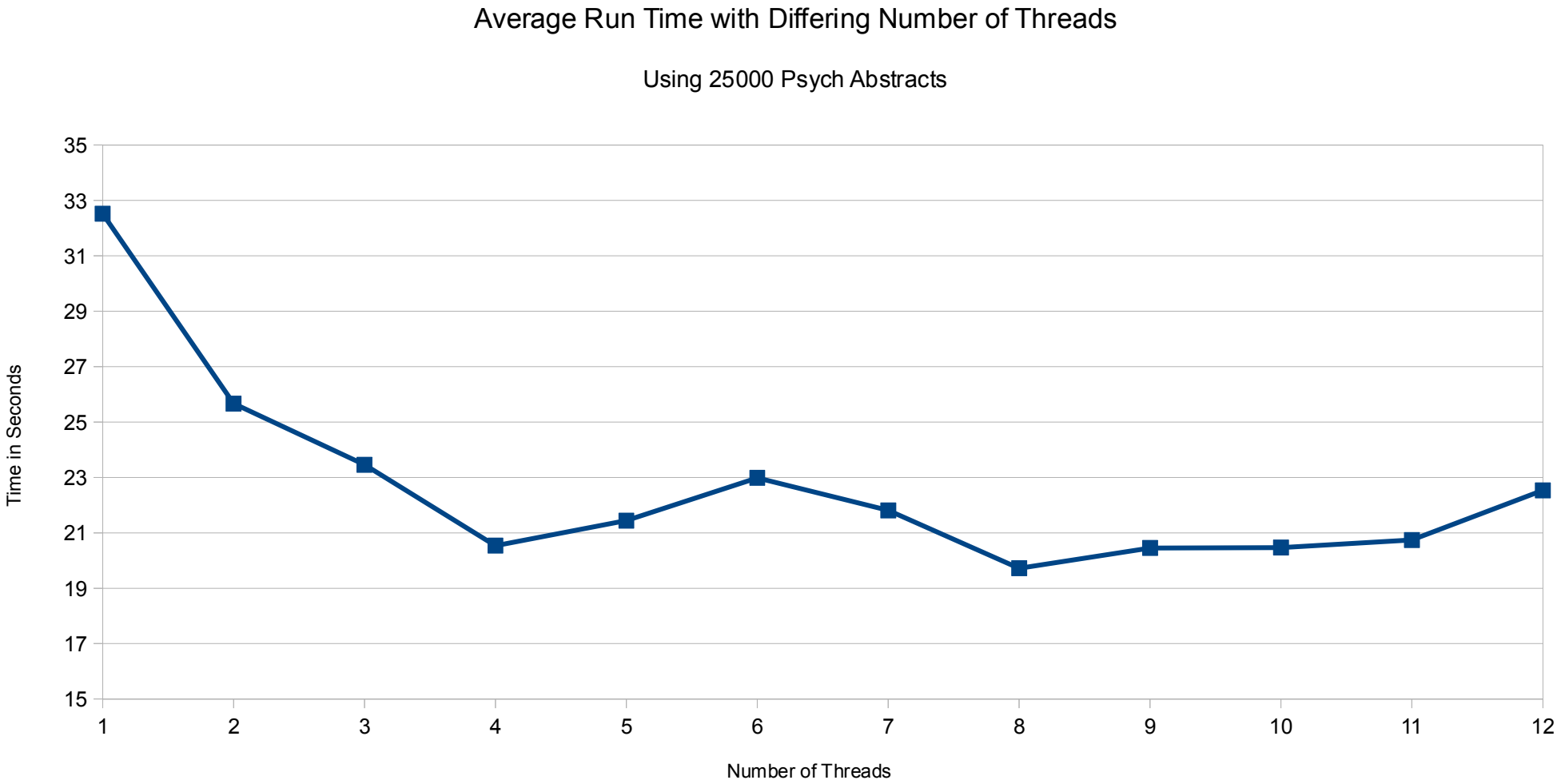
	Average Time
500 – 19.3 MB	6.5952379
1k – 40.7 MB	14.6556083
2k – 87.8 MB	38.3563106
3k – 131 MB	66.3107084
4k – 178 MB	100.9725318
5k – 218.3 MB	161.5056246

Patent Document Data Set

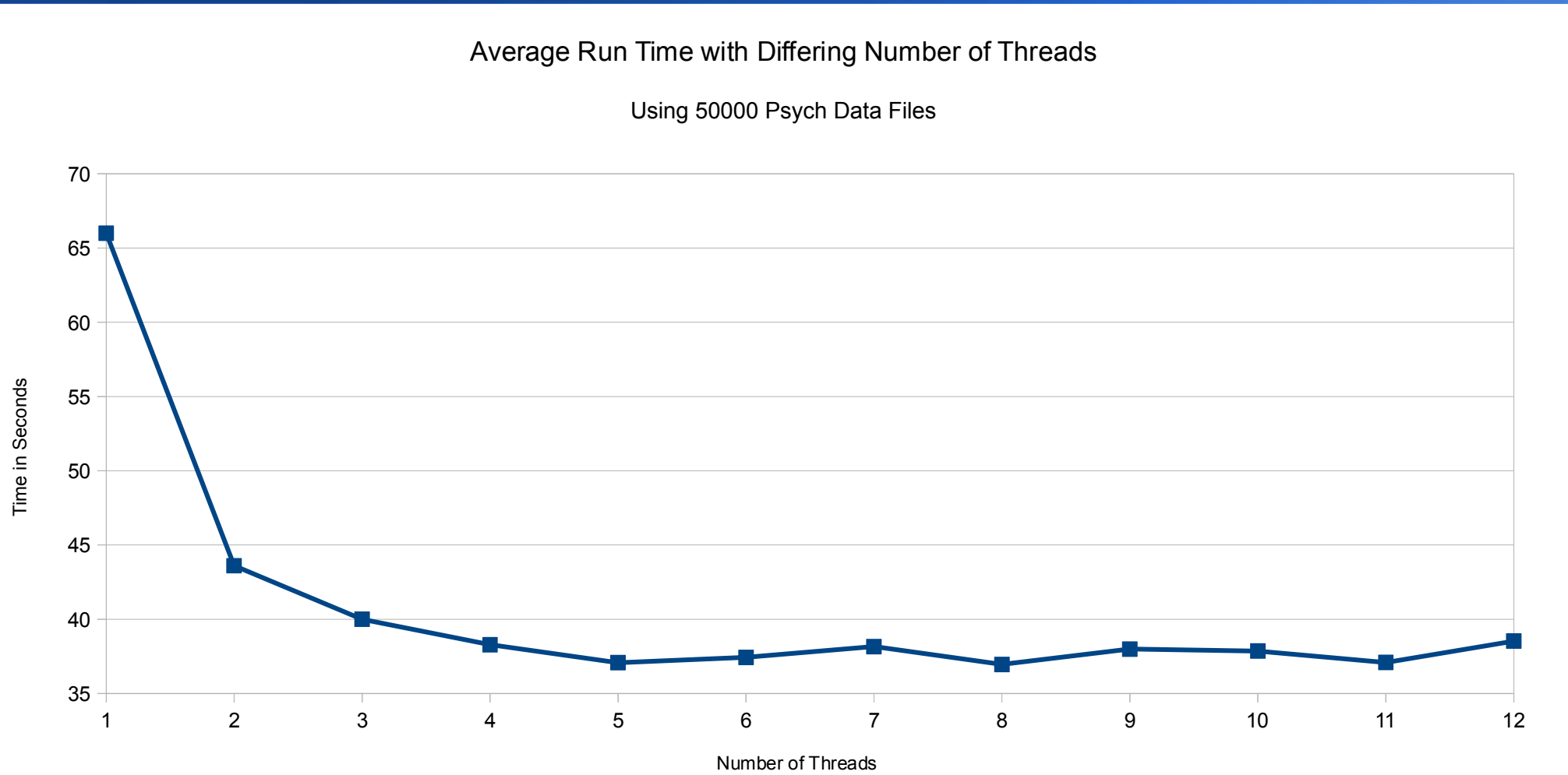
How Data Processing Time was Decreased

- Threads
- Unfamiliar with Java threads
- First implementation: My own way of threading
- Better implementation: ExecutorService
- Stop race conditions between threads
- Determine optimal number of threads

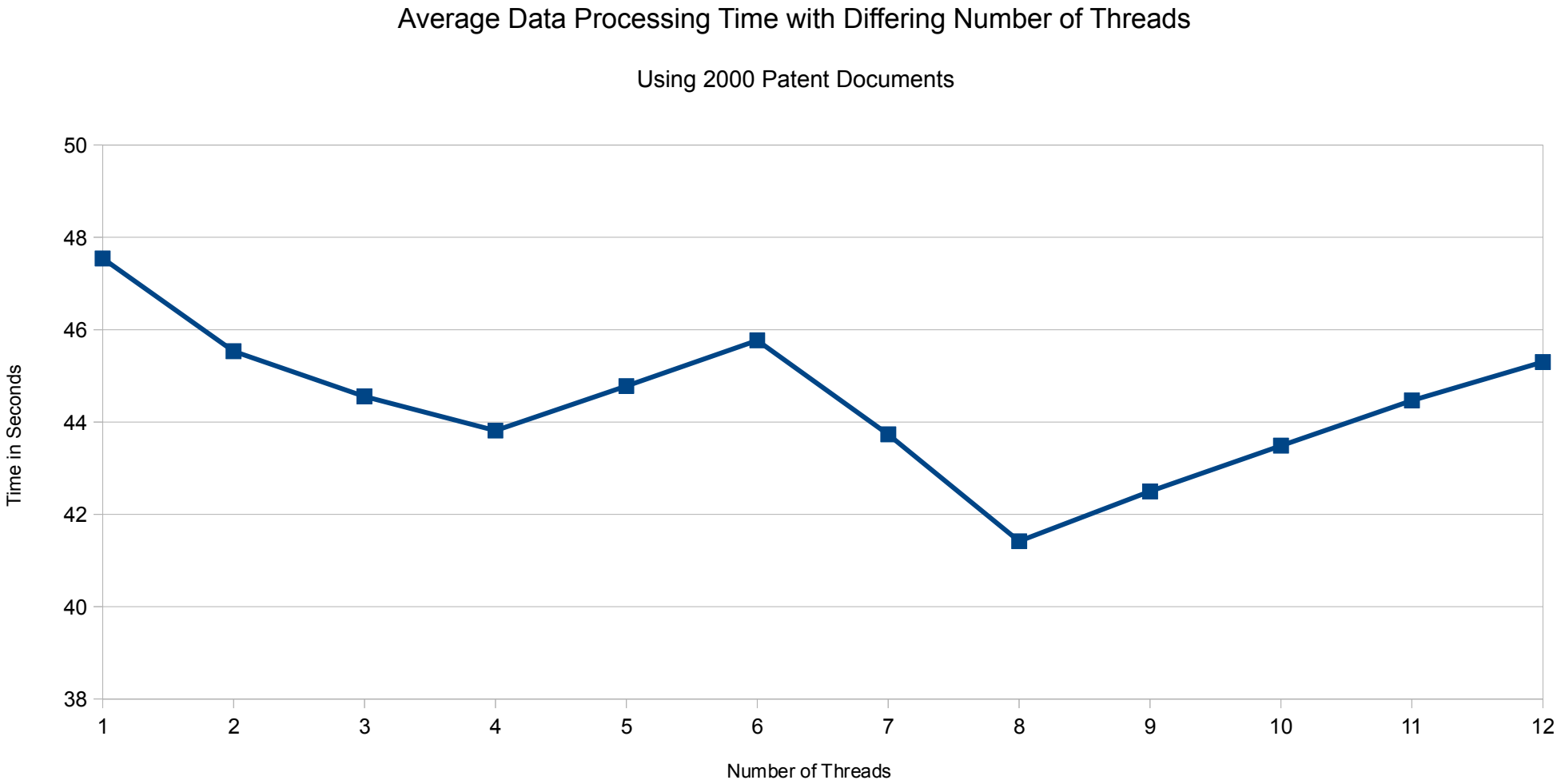
Optimal Number of Threads



Optimal Number of Threads (cont.)



Optimal Number of Threads (cont.)



Optimal Number of Threads (cont.)

- 8 threads appears to be optimal
- Number of Processors * 2

Data Processing Time

- The average time for the new version of FutureLens to process each data set

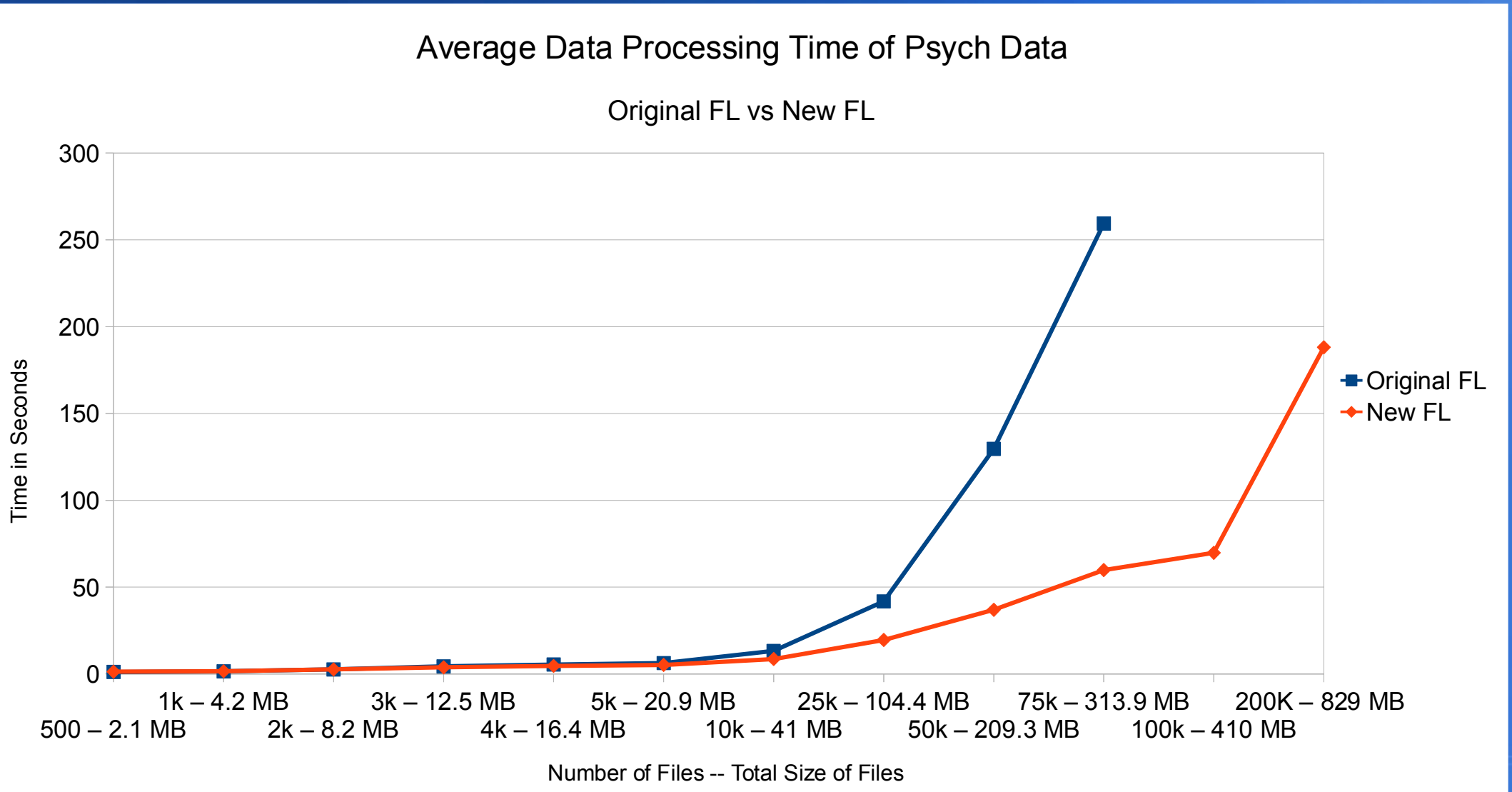
	Average Time
500 – 2.1 MB	1.3215038
1k – 4.2 MB	1.5722328
2k – 8.2 MB	2.6714057
3k – 12.5 MB	3.8262265
4k – 16.4 MB	4.6330552
5k – 20.9 MB	5.2030857
10k – 41 MB	8.6284829
25k – 104.4 MB	19.5933614
50k – 209.3 MB	36.9599291
75k – 313.9 MB	59.8540463
100k – 410 MB	69.7749802
200K – 829 MB	188.1068154

Psych Abstract Data Set

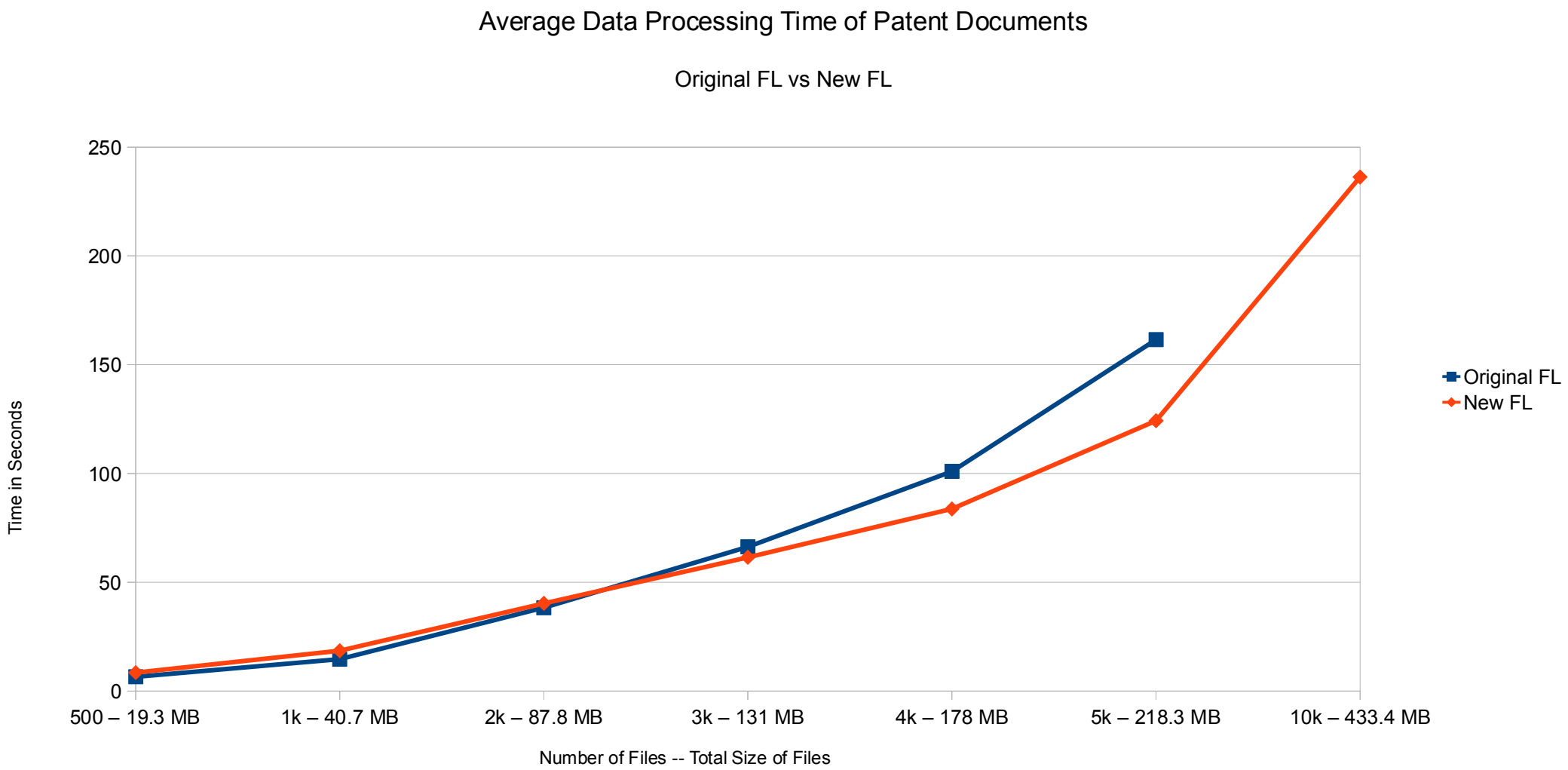
	Average Time
500 – 19.3 MB	8.499209
1k – 40.7 MB	18.5364414
2k – 87.8 MB	40.346865
3k – 131 MB	61.4941004
4k – 178 MB	83.7137996
5k – 218.3 MB	124.2625535
10k – 433.4 MB	236.2620091

Patent Document Data Set

Comparison of Data Processing Times



Comparison of Data Processing Times (cont.)



Future Work

- Addition of a database
- Rewriting the application in another language

References

Gregory Shutt, Andrey Puretskiy, Michael W. Berry, “FutureLens”, Department of Electrical Engineering and Computer Science, The University Of Tennessee, November 20, 2008.

"Eclipse documentation: ArmListner", http://help.eclipse.org/indigo/index.jsp?topic=%2Forg.eclipse.platform.doc.isv%2Freference%2Fapi%2Forg%2Feclipse%2Fswt%2Fevents%2FArmListener.html, Visited February 2012.

VisualVm 1.3.4, http://visualvm.java.net/

"Executor Interfaces", http://docs.oracle.com/javase/tutorial/essential/concurrency/exinter.html, Visited April 2012

Questions?