CLUSTER-SLACK RETENTION CHARACTERISTICS: A CASE STUDY OF THE NTFS FILESYSTEM

Zak Blacher – June 2010

Agenda

- Purpose / Motivation
- FIVES
- Terminology / Visualizations
  - Clusters
  - Binary Units
  - MFT
- Digital Forensics
- Tail Slack
- Microfragments
- Experimental Outline
- Theoretical Formulas
- Fixed Size Experiment
- Fixed Size Results
- Analysis
- Other Results
- Conclusion

Purpose / Motivation

Demonstrate the efficacy of Microfragment analysis as a forensic tool as part of the FIVES utility chain

The FIVES Initiative

Forensic Image & Video Examination Support

Karlstad University in partnership with:

Korps landelijke politiediensten	NL
Netclean Technologies AB	SE
Institute of Information Technologies at the Bulgarian Academy of Sciences	BG
German Research Center for Artificial Intelligence GmbH	DE
Federal Computer Crime Unit of Federal Police	BE

FIVES

Objectives:

- Speed up process of handling large amounts of digital evidence by using efficient file and fragment matching
- Efficiently evaluate large amounts of material through optimization techniques
- Improve capability of linking new material to existing sets of similar data

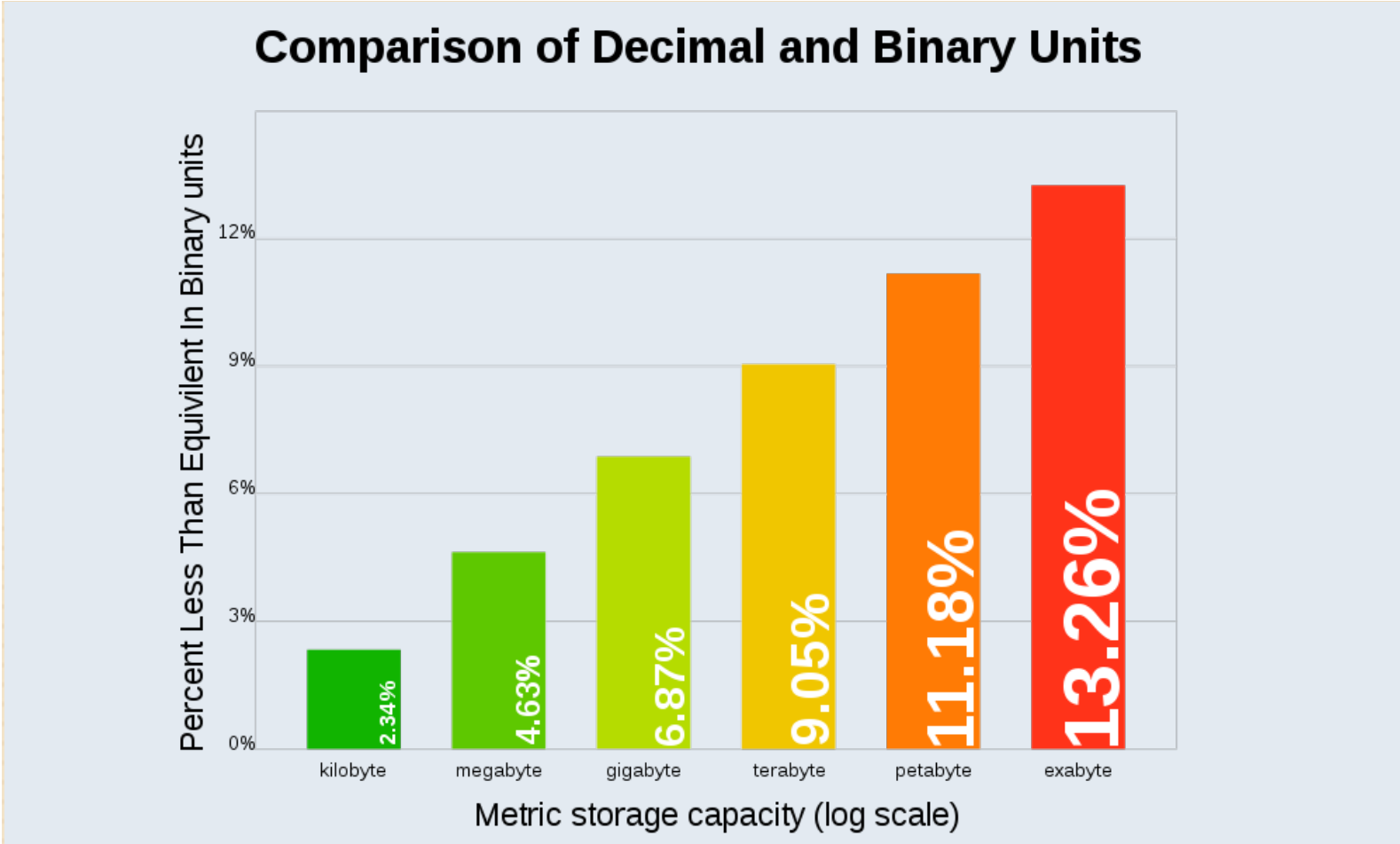
File Fragments (Microfragments)

?

Terminology / Visualizations

- Bit
- Byte
- Block
- Cluster
  - File can occupy multiple clusters

- Actual File Size < Size on Disk
  - Clusters not shared



Decimal versus Binary unit losses

Source: Wikipedia (Retrieved June 8th 2010)

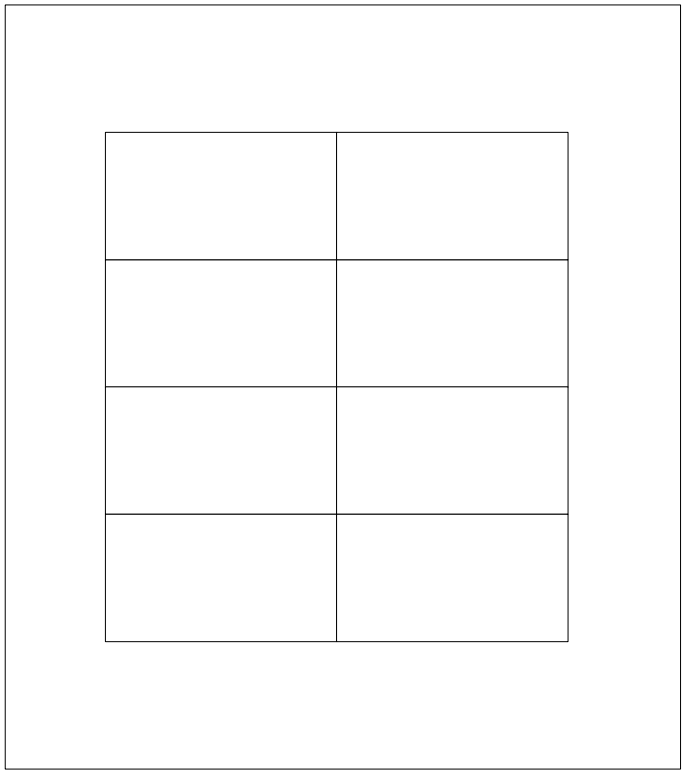
Why have clusters?

- Bytes & Blocks are too small to be addressed meaningfully and logically
- Easier to count and remap
- Can be different sizes of blocks depending
- NTFS Maximum Volume size: $(2^{32} - 1)$ * cluster size
- NTFS Maximum File count: (device size / cluster size)
- Larger clusters = potentially more slack blocks

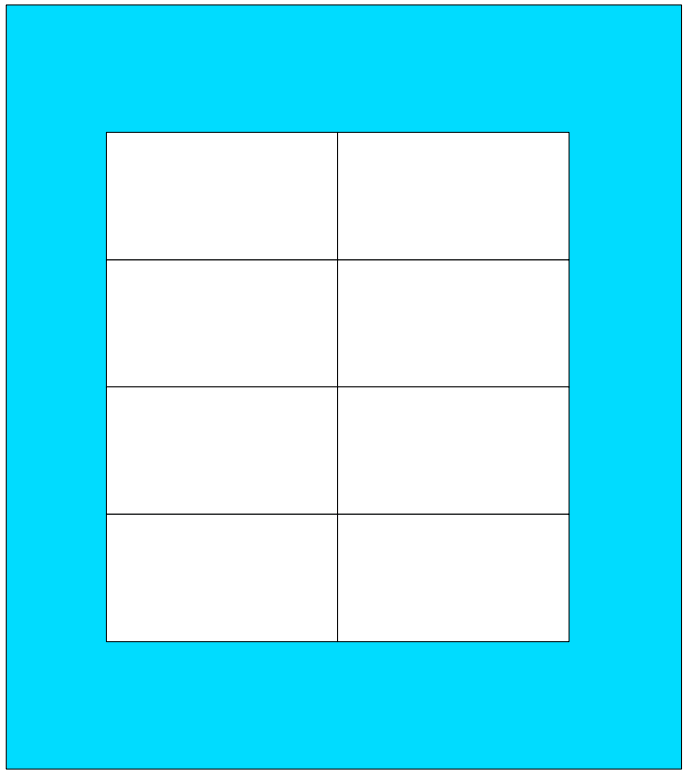
Master File Table

- Reserves ~12.5% of partition clusters
  - Grows/Shrinks as needed
- Contains Filesystem metadata
- Filesystem directory index
- Will be represented out-of-band

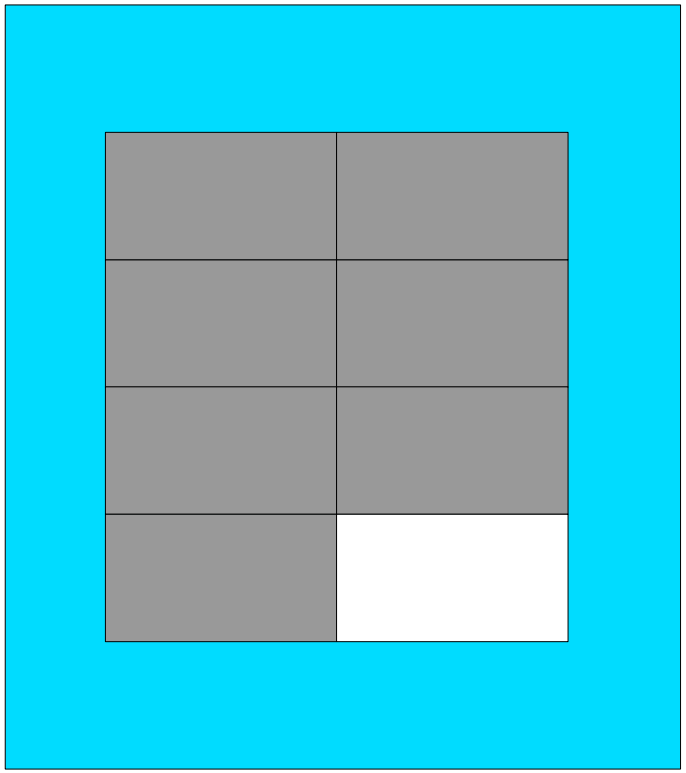
Cluster of 8 Blocks



Allocated Cluster



Allocated Cluster w/ Data



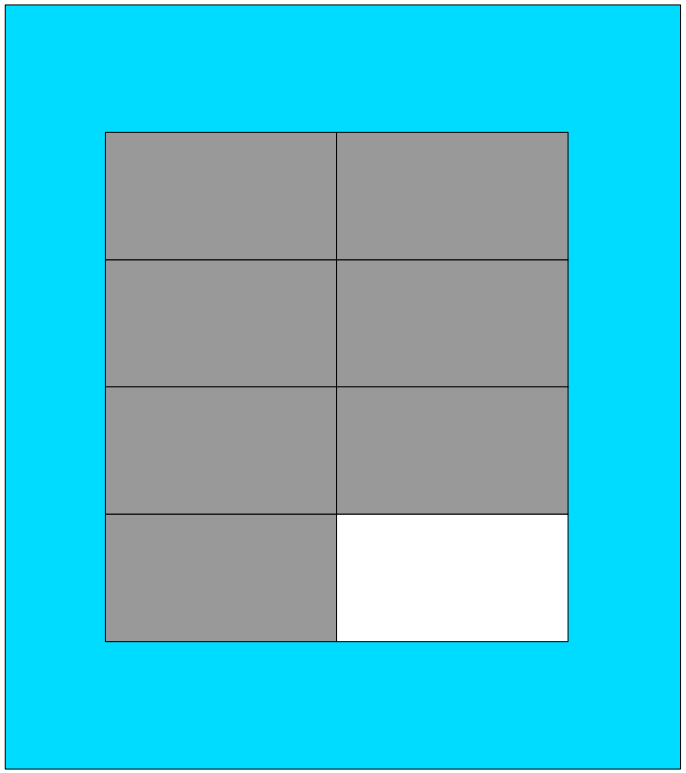
Digital Forensics

- To explain the state of the digital artifact
  - How this data came to be
- Analyze a computer for evidence
- Clear investigative trail
- Does not disturb media (static analysis)
- Focuses on finding evidence rather than explaining

Digital Forensics (cont'd)

- Traditional Approach:
  - Examine all blocks on device
  - Examine deleted sectors
- Why is this bad?
  - Slow
  - Prone to data loss – volatile state
- The space between…

Allocated Cluster w/ Data



Tail Slack

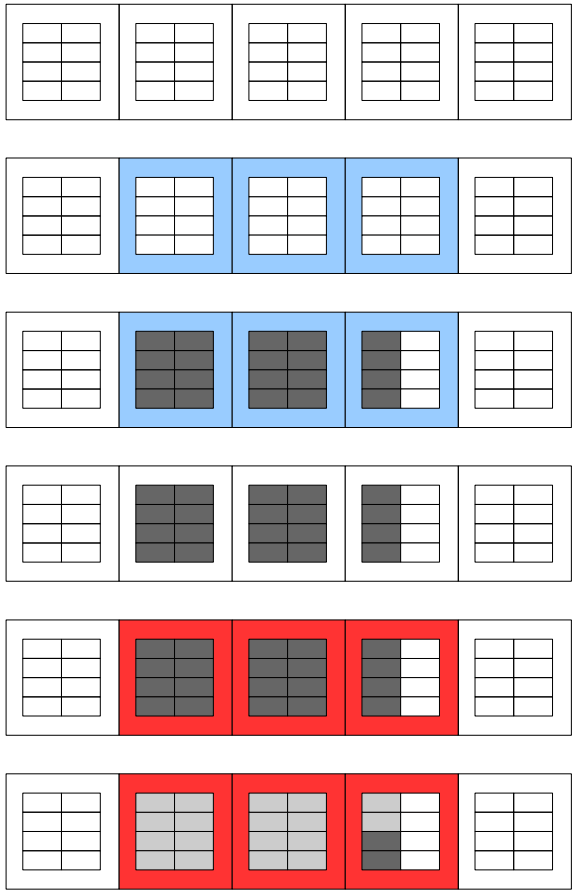
- Blocks at the end of a sector that have not been populated by new data.
- Only found in the final (or tail) cluster of an allocated group
- More protected than ‘deleted’ areas – Will not get overwritten while file remains unmodified

File Fragments (Microfragments)

Tail clusters containing one or more slack blocks.

Generation of Microfragments

In this example we write a 10KiB file, delete it, and then write an 9KiB onto the same clusters.

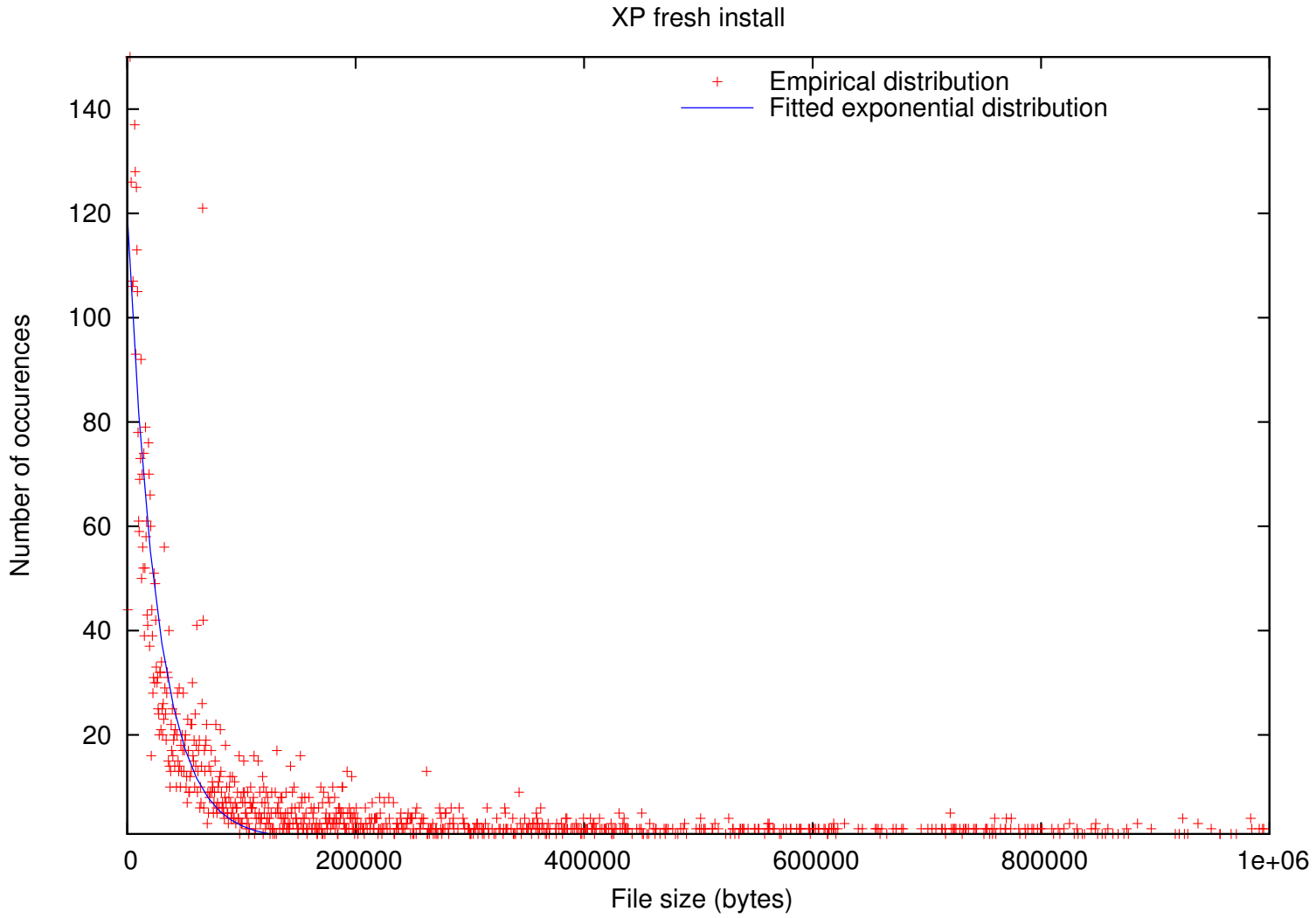


Statistical Analysis of Microfragments

This presentation will only focus on fixed size distributions due to time constraints

- Files differ in size!
- Files differ in size distribution
  - Fixed/Constant – Ripped DVDs
  - Uniform – CBR MP3s
  - Normal – JPEGs of same resolution
- The size affects the number of microfragments, and the tail difference affects the number of slack blocks

Real-World File Distribution



Experimental Outline

- 1Gb Device is formatted with given parameters
- 250x1000KiB files written to device (known content)
- Files are erased (MFT Entries deleted)
- Device is fully populated with new files conforming to a certain distribution and containing random data.
- Microfragment Analysis Performed
- Device completely overwritten with zeroes

Theoretical Formulas

- C = cluster size (in bytes),
- B = block size (in bytes),
- D = detection area (1 gibibyte),
- S = file size (bytes),
- $\hat{S}$ = average file size (bytes),
- N(S) = ceil(C/S) = number of clusters / file

Theoretical Formulas (cont'd)

Fixed Size:

$$W_R^{(c)} = \frac{D}{\left\lceil \frac{S_F}{C} \right\rceil C}$$

Uniform Distribution:

$$W_C^{(u)} = W_R^{(u)} P^{(u)} = \frac{D}{\bar{N}_C^{(u)} C} \left(1 - \frac{B}{C}\right)$$

$$\bar{N}_C^{(u)} = \frac{1}{C} \frac{1}{L_2 - L_1 + 1} \left(L_1^{(+)C} (L_1^{(+)C} - L_1 + 1) + \frac{1}{2} (L_2^{(-)C} - L_1^{(+)C}) (L2^{(-)C} + L_1^{(+)C} + C) + L_2^{(+)C} (L_2 - L_2^{(-)C}) \right)$$

$$L^{(+)C} = \left\lceil \frac{L}{C} \right\rceil C$$

$$L^{(-)C} = \left\lfloor \frac{L-1}{C} \right\rfloor C$$

Fixed Size

- Every random file has the same size
- Every Tail Sector has the same amount of retained blocks
- Easy to estimate approximate Microfragment retention

- In our tests, 63000 clusters are initially occupied by our 1000x250 Kbyte files

Fixed Size (cont'd)

Fixed 10 KiB

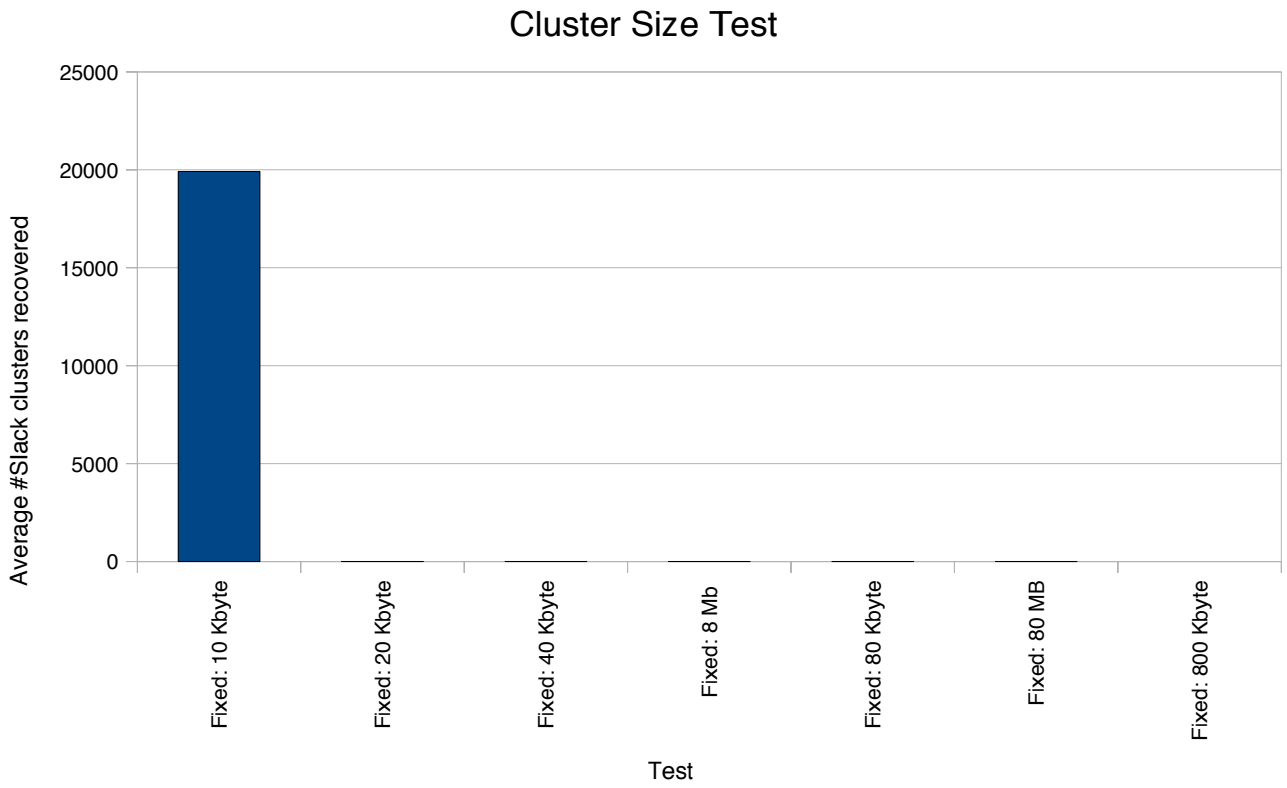
- ceil(10/4) = 3 clusters
- 10 % 4 = 2 KiB tail sl.
- 63000 * 1/3 = 21000 occ'd tail cl.
- 21000 * 2KiB = 42000 KiB slack

Fixed 20 KiB

- ceil(20/4) = 5 clusters
- 20 % 4 = 0 KiB tail sl.
- 63000 * 1/5 = 12600 occ'd tail cl.
- 12600 * 0KiB = 0KiB slack

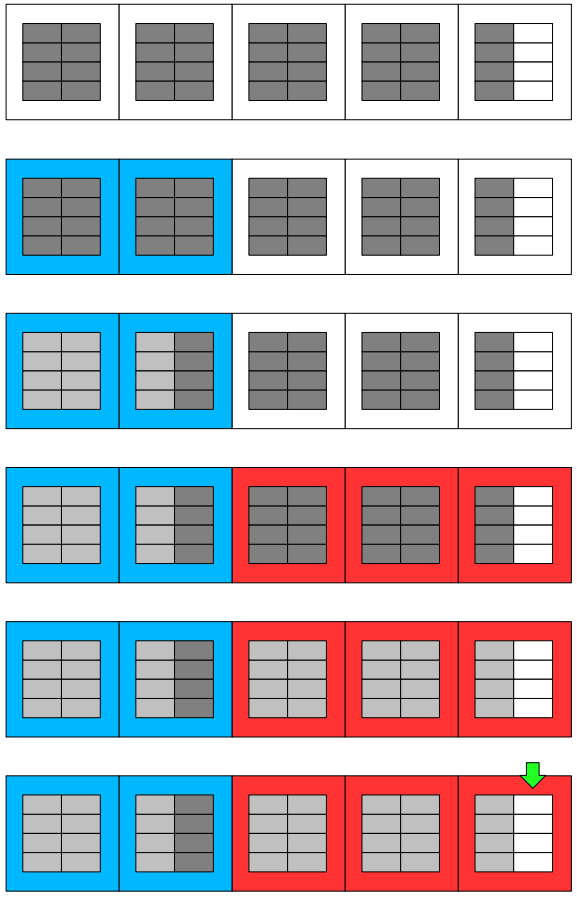
Fixed Size (Empirical Results)

Note: All tests except for Fixed 10Kbyte have random file size parameters that are integral multiples of the size of the cluster, completely overwriting cluster slack



Fixed Size: 20000 != 21000

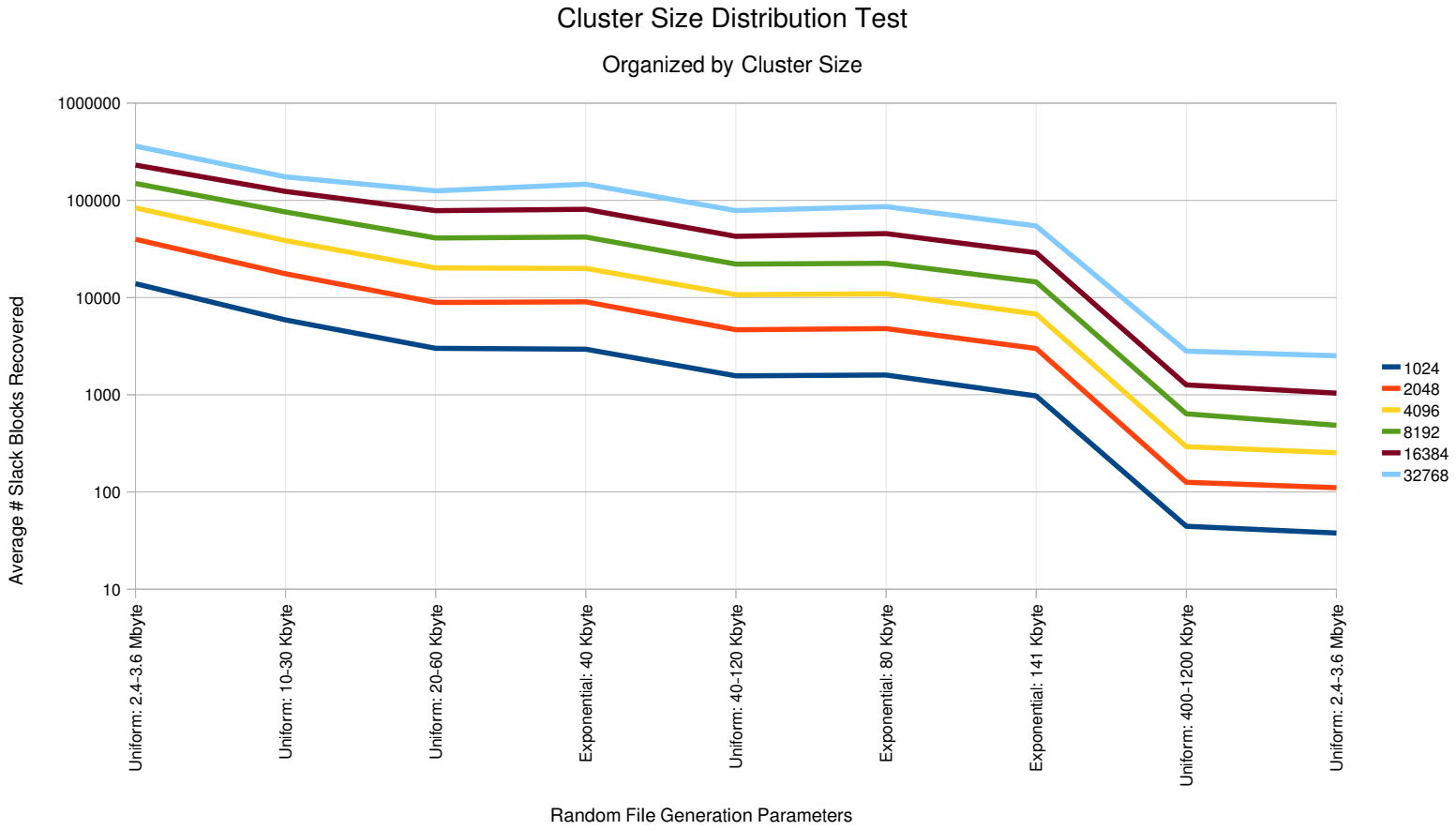
20 / 21 * 21000 = 20000



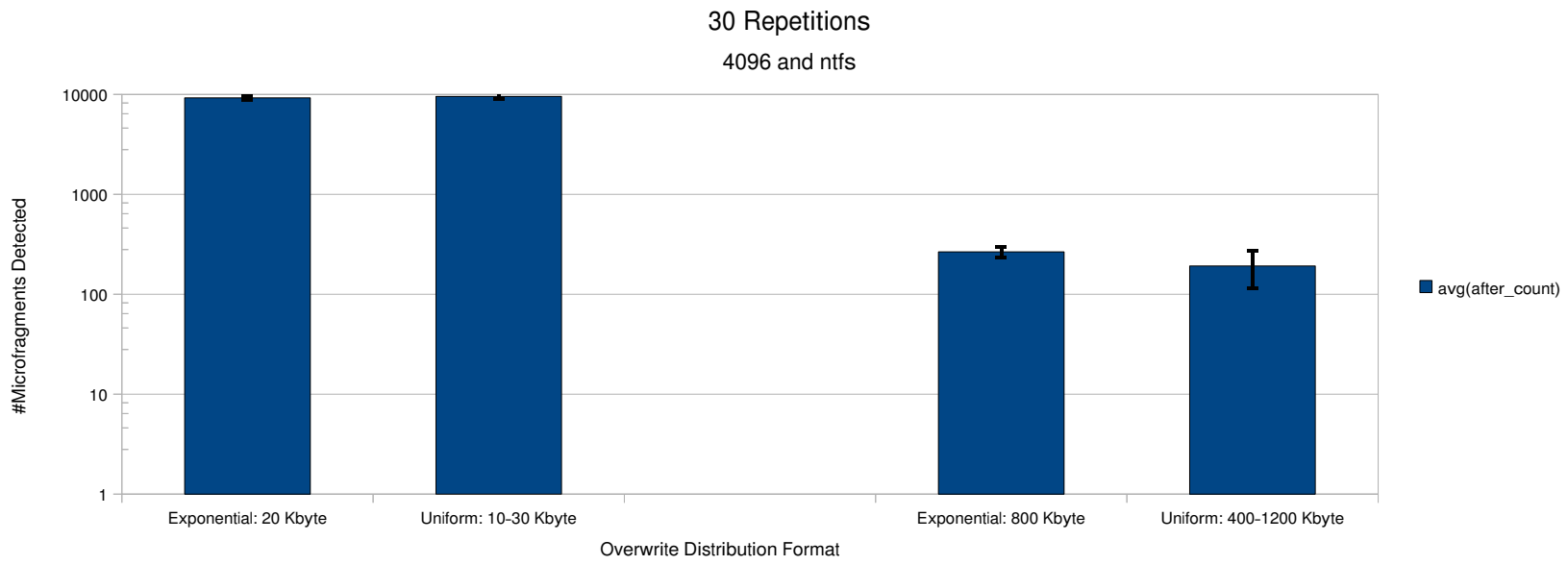
Analysis

Although this presentation only covered one case with fixed file sizes, we can see a generally ‘good’ agreement between our measured and our expected results

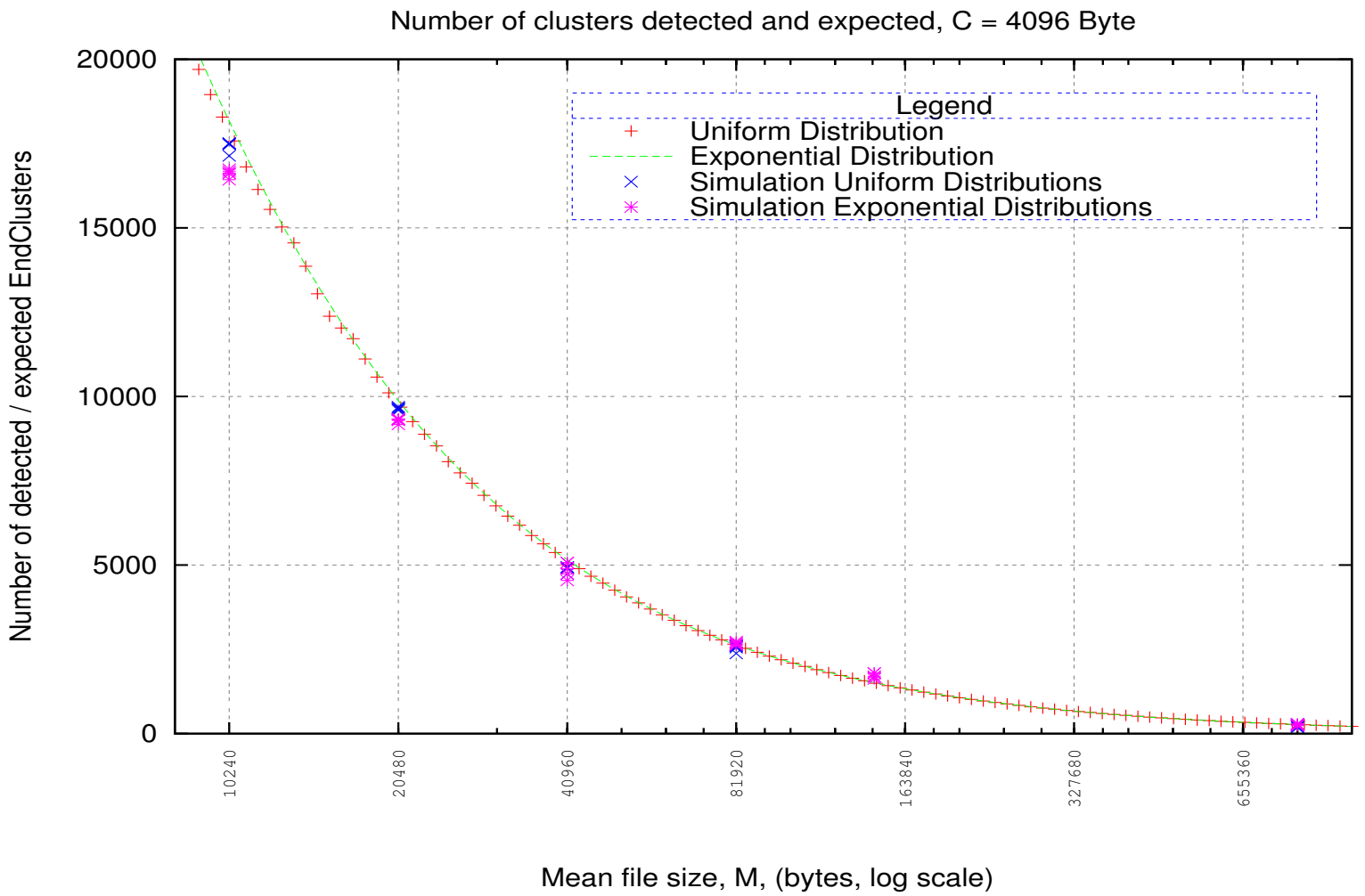
Varying Cluster Size



Repeated Tests for Accuracy



General Trend Agrees



Any Questions?

THANK YOU

Zak Blacher – June 2010