Digital Forensics Innovation: Searching A Terabyte of Data in 10 minutes

Simson L. Garfinkel
Associate Professor, Naval Postgraduate School

October 1, 2012

http://simson.net/

https://domex.nps.edu/deep/

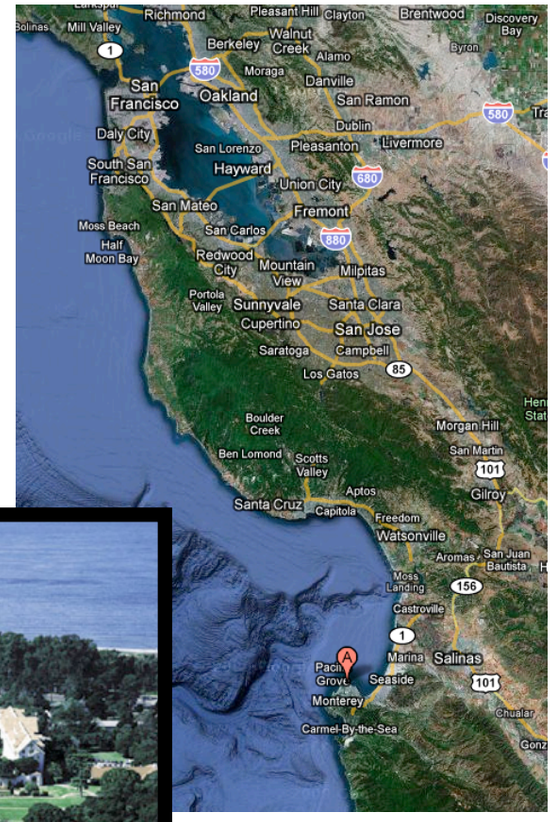
NPS is the Navy's Research University.

Monterey, CA — 1500 students

- US Military (All 5 services)
- US Civilian (Scholarship for Service & SMART)
- Foreign Military (30 countries)

Schools:

- Business & Public Policy
- Engineering & Applied Sciences
- Operational & Information Sciences
- International Graduate Studies



NCR Initiative — Arlington, VA

- 8 offices on 5th floor, Virginia Tech building
- 4 professors, 6 researchers
- ***WE ARE HIRING!***

The Digital Evaluation and Exploitation (DEEP) Group: Research in “trusted” systems and exploitation.

“Evaluation”

- Trusted hardware and software
- Cloud computing

“Exploitation”

- MEDEX — “Media” — Hard drives, camera cards, GPS devices.
- CELEX — Cell phone
- DOCEX — Documents
- DOMEX — Document & Media Exploitation

Current Partners:

- Law Enforcement (FBI & Local)
- DHS (HSARPA; Video Games & Insider Threat)
- NSF (Courseware development)
- DoD

Traditionally digital forensics was for *convictions*.

The goal: establish *possession* of *contraband information.*

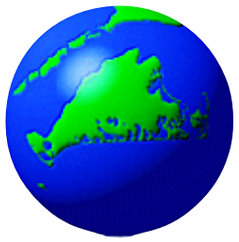
- Child Pornography
- Stolen documents
- Hacker tools

Forensics established:

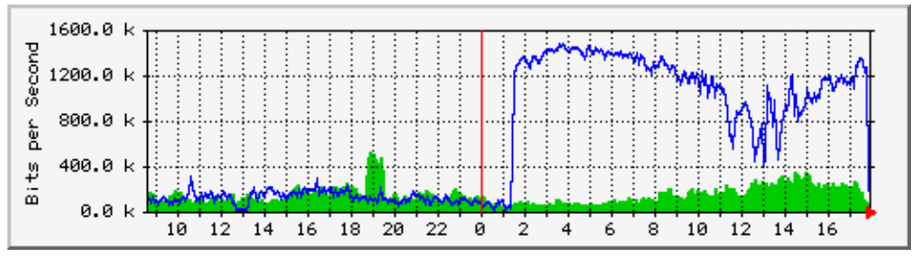
- Data *presence*
- Data *provenance* — where it came from & how it got there

I started working digital forensics in the 1990s

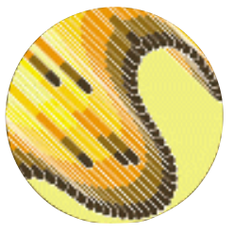
1995 — Vineyard.NET



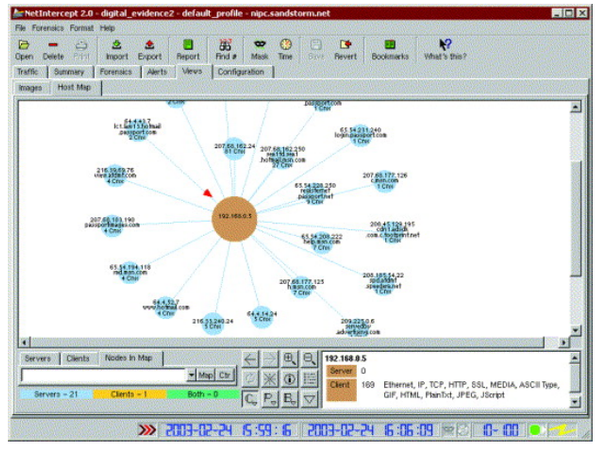
- Used forensics to investigate break-ins
- Limited forensics-related consulting
  - —*We called it "computer security"*



1998 — Sandstorm Enterprises



- Network forensics products:
  - —*PhoneSweep — Telephone scanner*
  - —*NetIntercept — Network traffic recording and analysis*



1998 —2006 The "drives" project

- Used computers purchased (for PhoneSweep) had sensitive data
- I started buying drives *for the data*.
  - —*Automated analysis techniques to find the good stuff (Garfinkel 2005)*

My goal is to use forensics for *investigation*

Data extraction — What information does the target have?

- contacts, calendar, documents

Data fusion — Putting together a unified model of the target

- When was something done?

Correlation — who has the *same information?*

- Identifies members within the organization.
- Identifying a subject's *associates*
- Automatically identifying *actionable information*

Much falls under the category "triage"

- Prioritizing analysis based on data content

Three principles underly this research:

1. Automation is essential.

- Today most forensic analysis is done manually.
- We are developing techniques & tools to allow automation.

Sencar and Memon (2009)

2. Concentrate on the invisible.

- It's *easy* to wipe a computer….
  - *but targets don't erase what they can't see.*
- So we look for:
  - *Deleted and partially overwritten files.*
  - *Fragments of memory in swap & hibernation.*
  - *Tool marks.*

3. Large amounts of data is essential.

- Most research is based on search & recognition
  - *10x the data produces 10x the false-positives*
- We are develop algorithms that work *better* with more data.

Given sufficient data, we can automatically assemble complex social network diagrams

We analyzed 2000 hard drives.

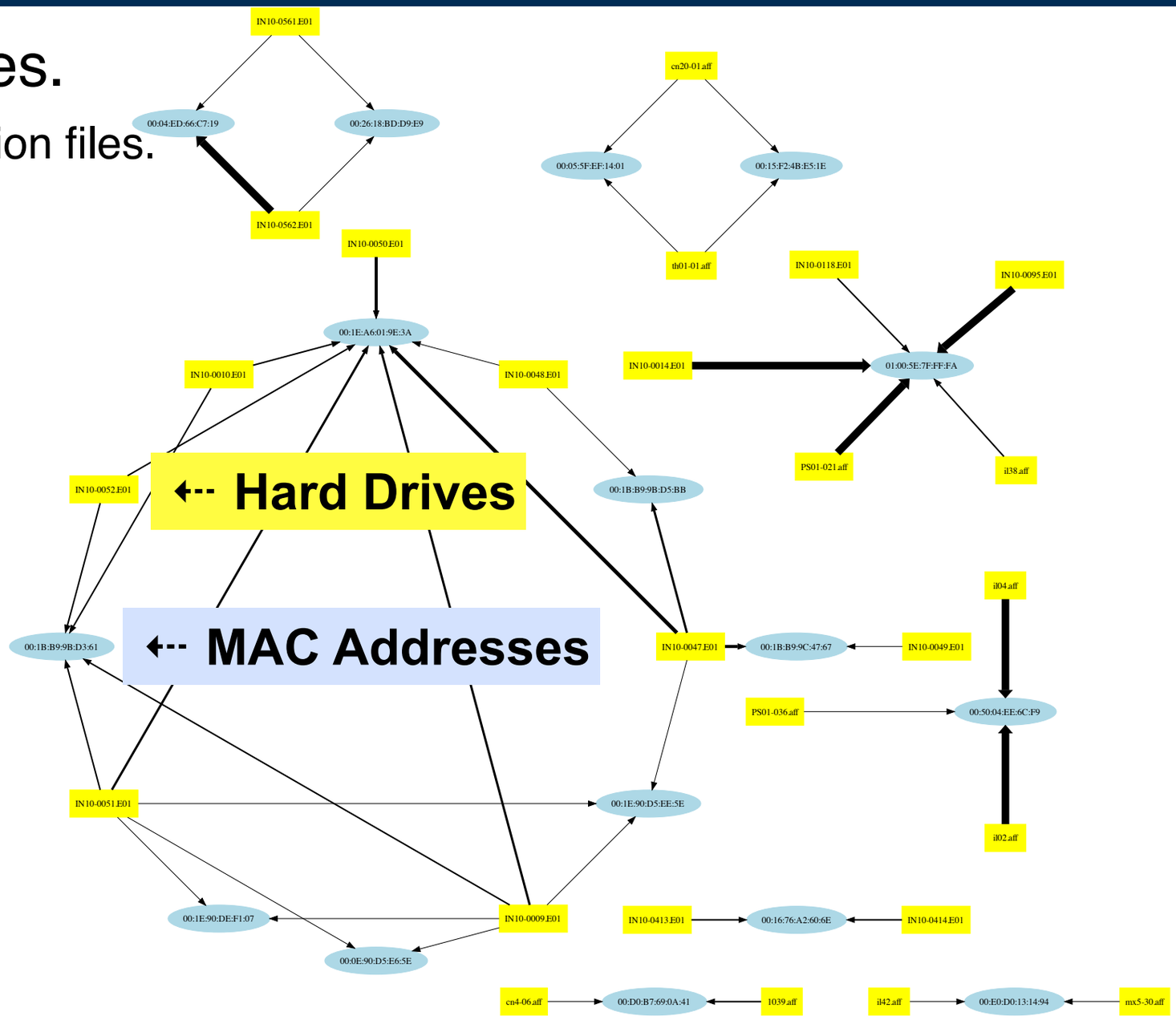
- Find IP packets in swap & hibernation files.
- Extract ethernet MAC addresses.

Post-processing identifies:

- Shared wireless routers.
- Common ethernet routers.

Validation:

- Reconstructed networks came from same organization.



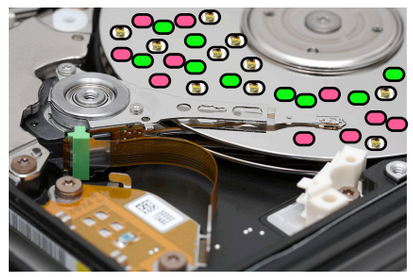
—*Forensic Carving of Network Packets and Associated Data Structures, Beverly & Garfinkel, DFRWS 2011, August 2011, New Orleans*

This talk presents today's digital forensic challenges and presents a research project that helps address them.

Introducing DEEP and Digital Forensics

Today's Digital Forensics Challenges

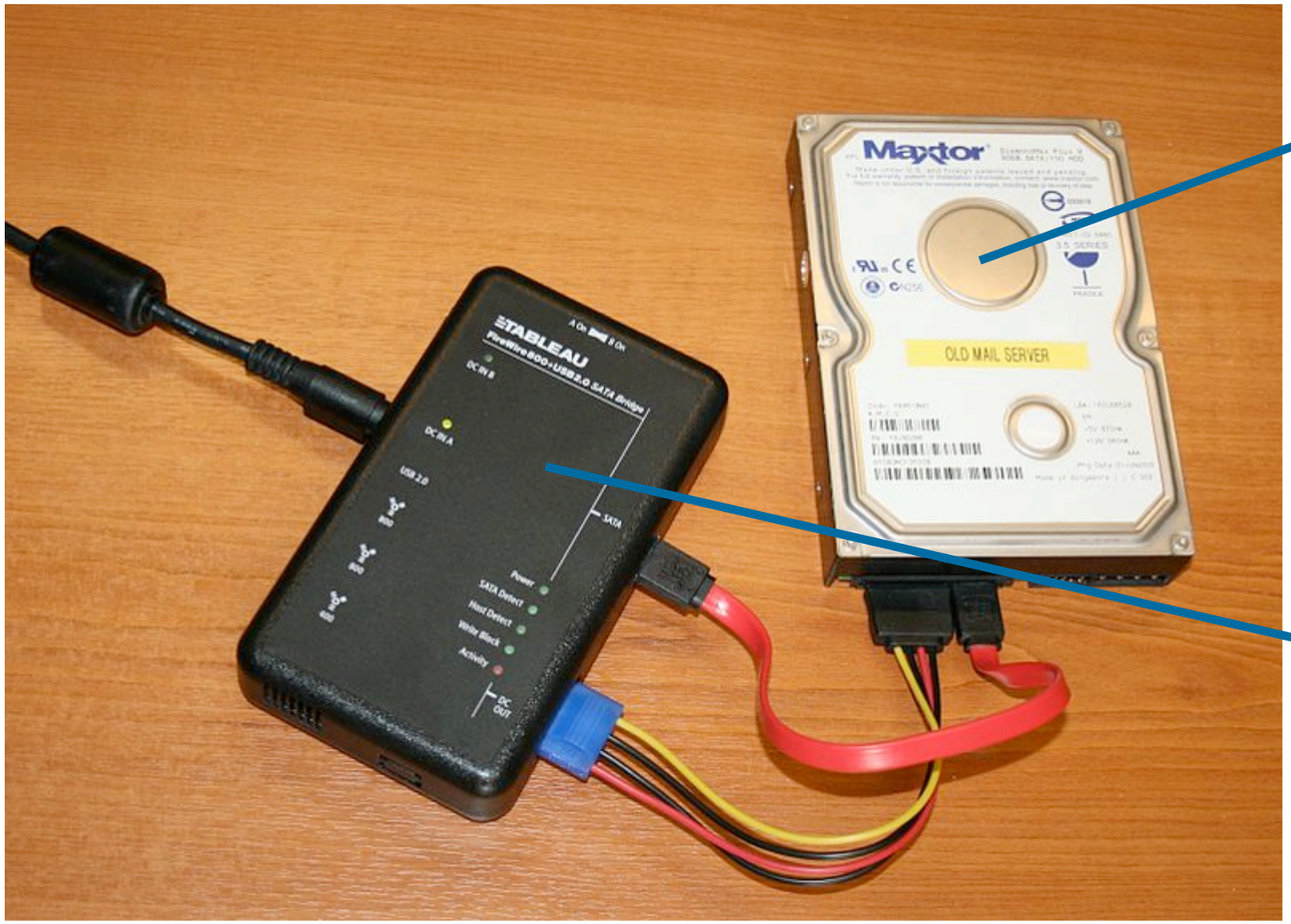
Random sampling for high speed forensics



Challenges Facing Digital Forensics

Data extraction is the first step of forensic analysis

"Imaging tools" extract the data without modification.



"Write Blocker" prevents accidental overwriting.

Original device stored in evidence locker.

Forensic copy ("disk image") stored on a storage array.

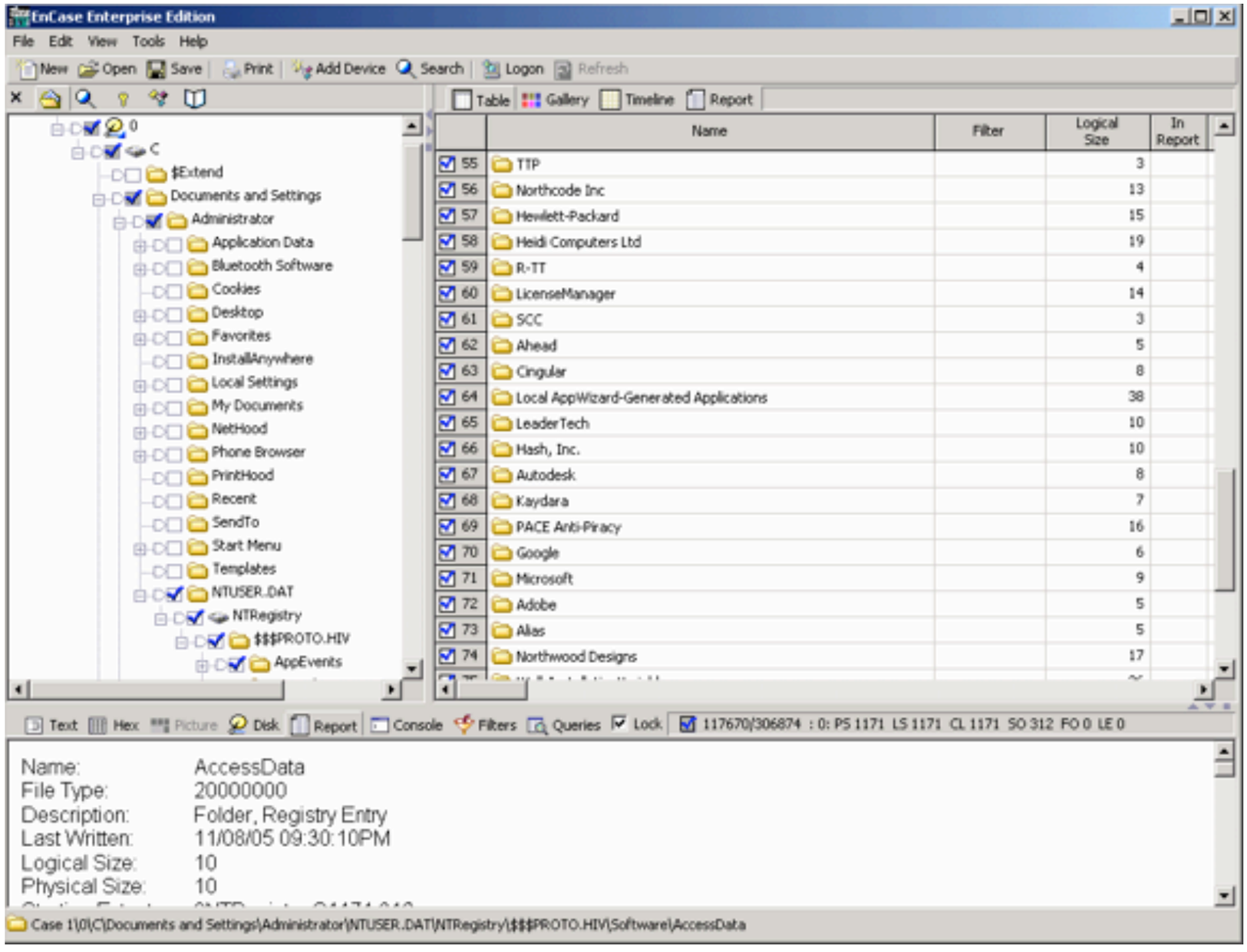
Write blockers are available for USB drives.

Phones require special handling...
most phones are not set up for easy data extraction.

Forensic tools let examiners view and search.

Today's tools allow the examiner to:

- Display of *allocated* & *deleted* files.
- String search.
- File extraction
- File “carving”
- Examining disk sectors.



EnCase Enterprise by Guidance Software

Digital forensics is fundamentally different from other kinds of scientific exploration...

There are five key challenges that we face...

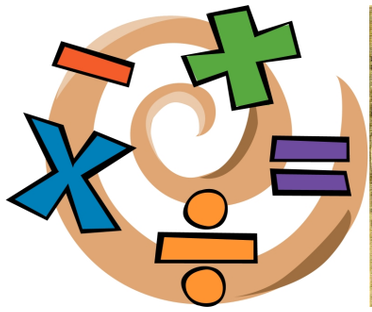
1: Diversity —of systems, file and content

Our charter:

"Analyze any data that might be found on a computer."

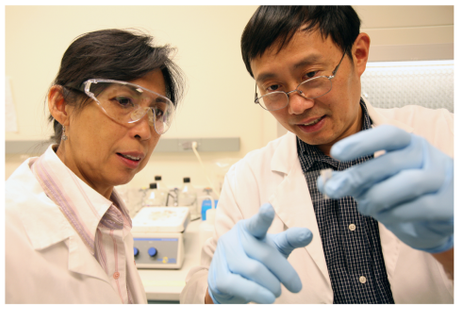
Non-DF research is typically confined to a single area:

energy



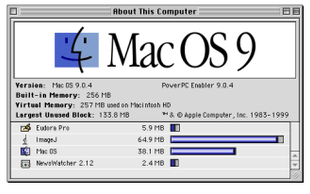
math

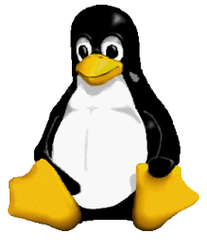
literature



chemistry

DF must analyze any OS, application, protocol, encryption, etc...





1 con't: Diversity is more than a multiplicity of file formats...

Data may be *inconsistent* or *incomplete*

- Files that are *deleted* or partially *overwritten*
- Incomplete database records
- Intentionally altered to avoid analysis

Data frequently have *no public specification*

- Hacker tools & malware
- Proprietary file formats

We need strategies for systematically addressing diversity

- Exploit similarity and correlation.
  - *Items of interest are frequently repeated.*
- Detect deliberate attempts to hide information
  - *Eliminate the truth and the improbable, and whatever remains must be impossible (and therefore falsified)*
  - *"Improbable" data should be examined for stenography.*

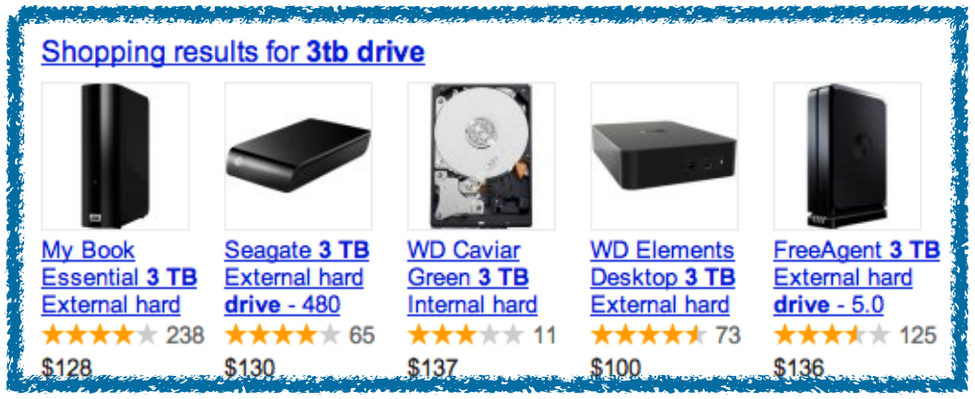
2: Data scale — a never ending problem

Scale is continually identified as a DF problem

—*DFRWS 2001:*
"The major item affecting overall performance is data volume: the amount of data collected for analysis of this type is often quite large."

Moore's law scales the targets

- We are using top-of-the-line system to analyze top-of-the-line systems
- We need to analyze in hours (or days) what a subject spent weeks, months or years assembling



∴ We will *never* outpace the performance curve.

Most "big data" solutions from other fields don't work well with DF

- They have bigger budgets per byte (CERN LHC≃1.5PB/month)
- Data diversity — Physics data is less diverse than a hard drive data
- Our data fights back — CERN data is not compressed/encrypted/fragmented/malware
  —*Data complexity dramatically increases I/O and compute requirements*

3: Temporal diversity — a never-ending upgrade cycle

Today's DF tools must process:

- Today's computers / phones / cameras
  - —*Because some criminals like to buy what's new!*
- Yesterday's computers / phones / cameras
  - —*Because criminals are using old devices too!*

Implications for DF users and developers:

- Upgrade DF software as soon as possible.
- DF software will become geometrically more complicated over time....
  - —*... or DF software will adapt on the fly to new data formats and representations.*
  - —*automated code analysis; pattern matching; hidden Markov models; etc.*

4: Human capital is bad all over – especially for DF

DF users (examiners, analysts):

- Overwhelmingly in law enforcement.
- Little or no background in CS or IS
- Deadline-driven; over-worked
- Knowledgable users tend to focus in just one particular area.
  - *Result: It takes two years to train most DF examiners.*

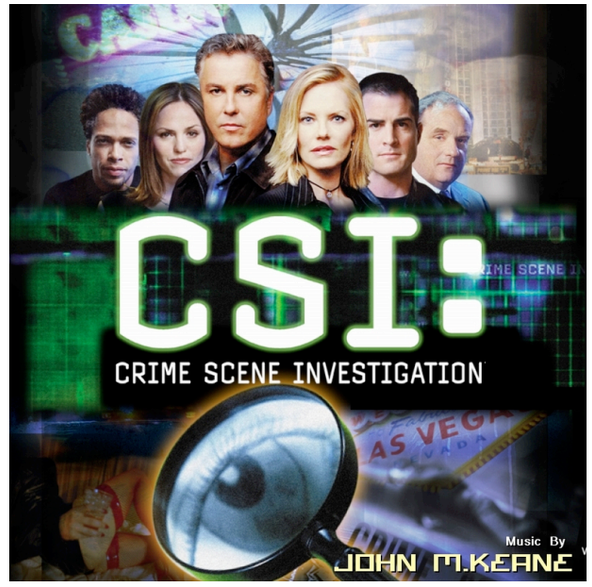
DF developers ("researchers"):

- Data diversity means developers need to know the whole stack
  - *opcodes & Unicode ⇒ OS & Apps ⇒ networking, encryption, etc.*
- Scale issues means developers need to know HPC:
  - *threading, systems engineering, supercomputing, etc.*
- Result:
  - *It's hard to find qualified developers*
  - *Developers must be generalists*

5: The “CSI Effect” — unrealistic expectations.

On TV:

- Forensics is swift.
- Forensics is certain.
- Human memory is reliable.
- Presentations are highly produced.



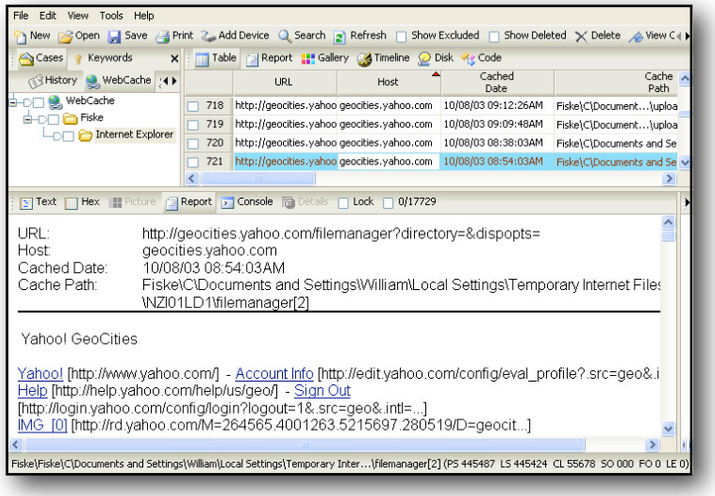
TV digital forensics:

- Every investigator is trained on every tool.
- Correlation is easy and instantaneous.
- There are no false positives.
- Overwritten data can be recovered.
- Encrypted data can usually be cracked.
- It is impossible to delete anything.

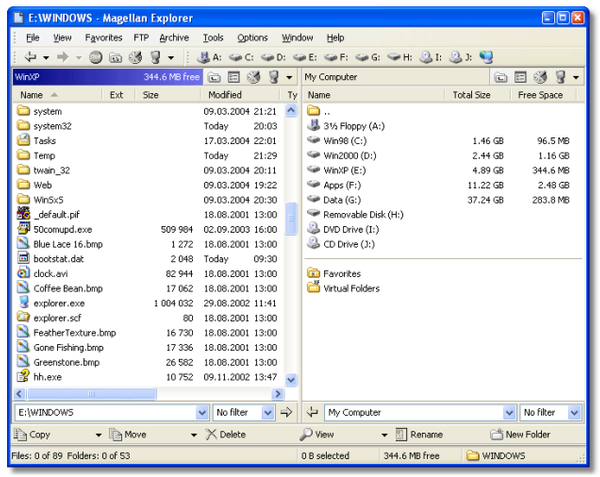
The reality of digital forensics is less exciting.

There are lots of problems:

- Data that is overwritten cannot be recovered
- Encrypted data usually can't be decrypted
- Forensics rarely answers questions or establishes guilt or provides specific information
- Tools crash a lot
- DF tools look a lot like traditional tools



EnCase



Windows Explorer

Result:

— *DF is a difficult process that looks easy*

— *This is not a good place to be*

6: Digital Forensics tools — expensive with limited market

DF tools are expensive to develop:

- Data diversity
- Security critical
- High performance computing

Limited market:

- Consulting firms (more effective tools *decreases* billable hours)
- Police departments (not known for $$)
- Defense (not known for major DF expenditures)

My personal experience:

- It's very hard to stay in business as a tool developer
- Government should have an ongoing role in funding DF research and tool development
- Open source software frequently makes the most sense
  - —*Open Source preserves investment, enables future research, empowers users.*

DF researchers must respond with new algorithms.

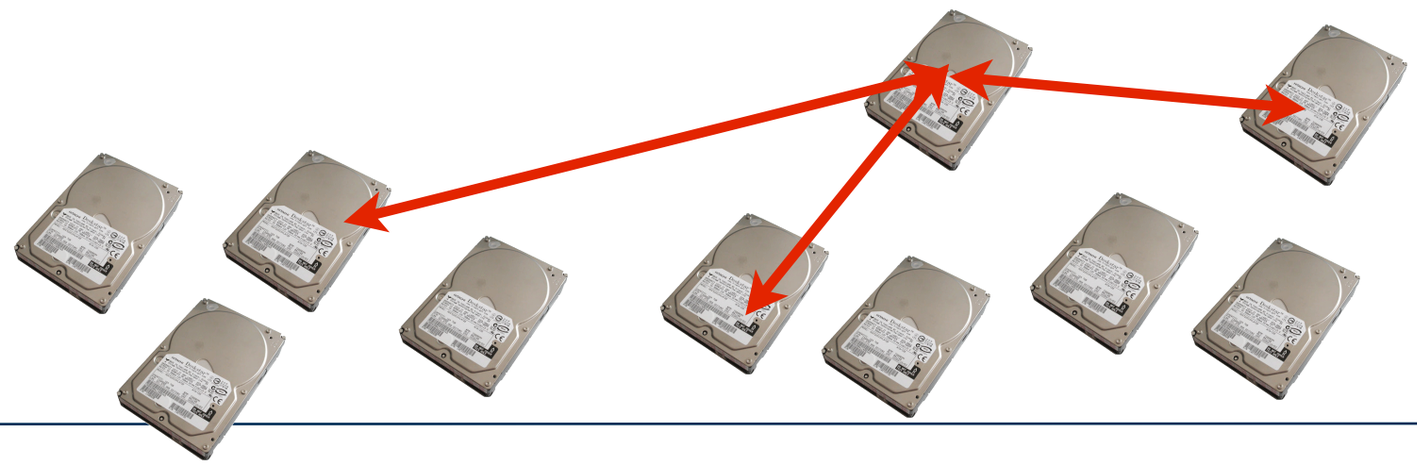
Current approaches don't scale.

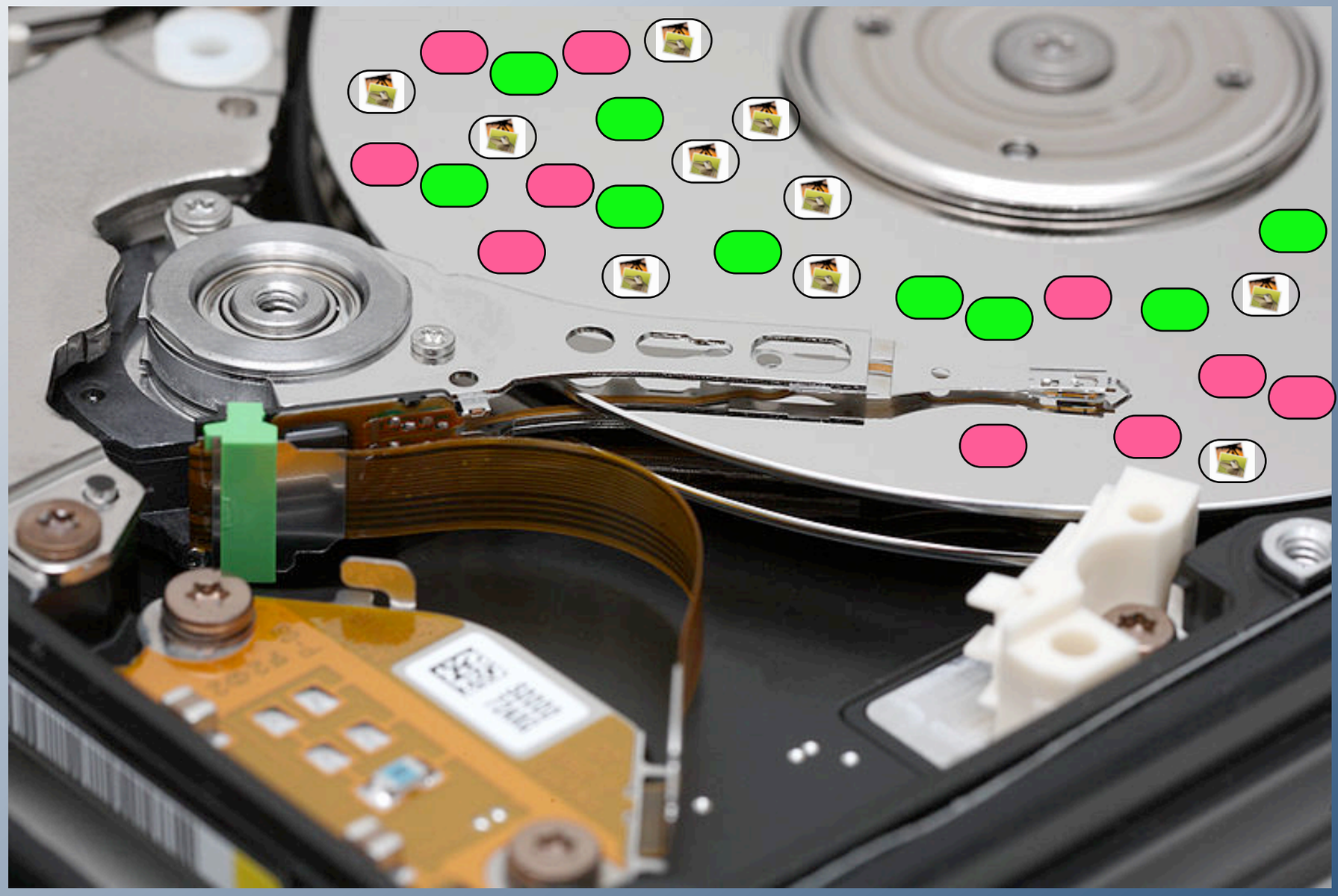
- User spent *years* assembling email, documents, etc
- Analysts have days or hours to process it
- Police analyze top-of-the-line systems
  - —*with top-of-the-line systems*
- National Labs have large-scale server farms
  - —*to analyze huge collections*

The problems:
1. Data Size
2. Mobile Devices
3. Encryption
4. Diversity
5. Time
6. No Market

Our new algorithms must:

- —*Provide incisive analysis through outlier detection and correlation*
- —*Operate autonomously on incomplete, heterogeneous datasets*
- —*Automatically calibrate; have no false positives*





High Speed Forensic Analysis with Random Sampling

Can we analyze a 1TB hard drive in five minutes?

US agents encounter hard drives at border crossings...

Searches turn up rooms filled with servers….

If it takes 3.5 hours to read a 1TB hard drive, what can you learn in 5 minutes?

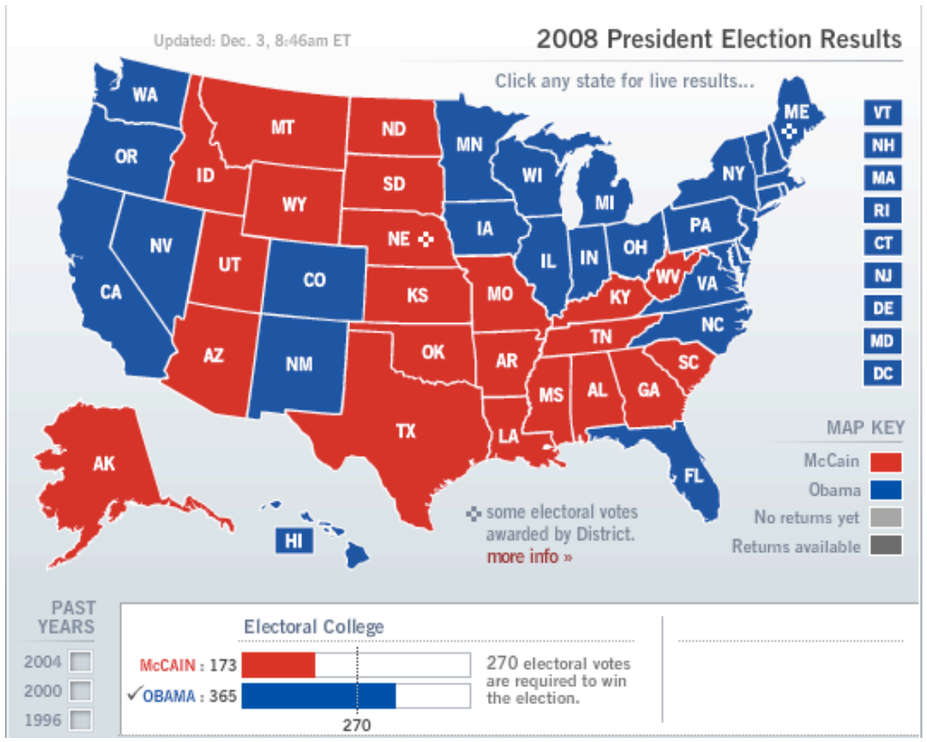
		
Minutes	208	5
Max Data	1 TB	36 GB
Max Seeks		90,000

36 GB is a lot of data!

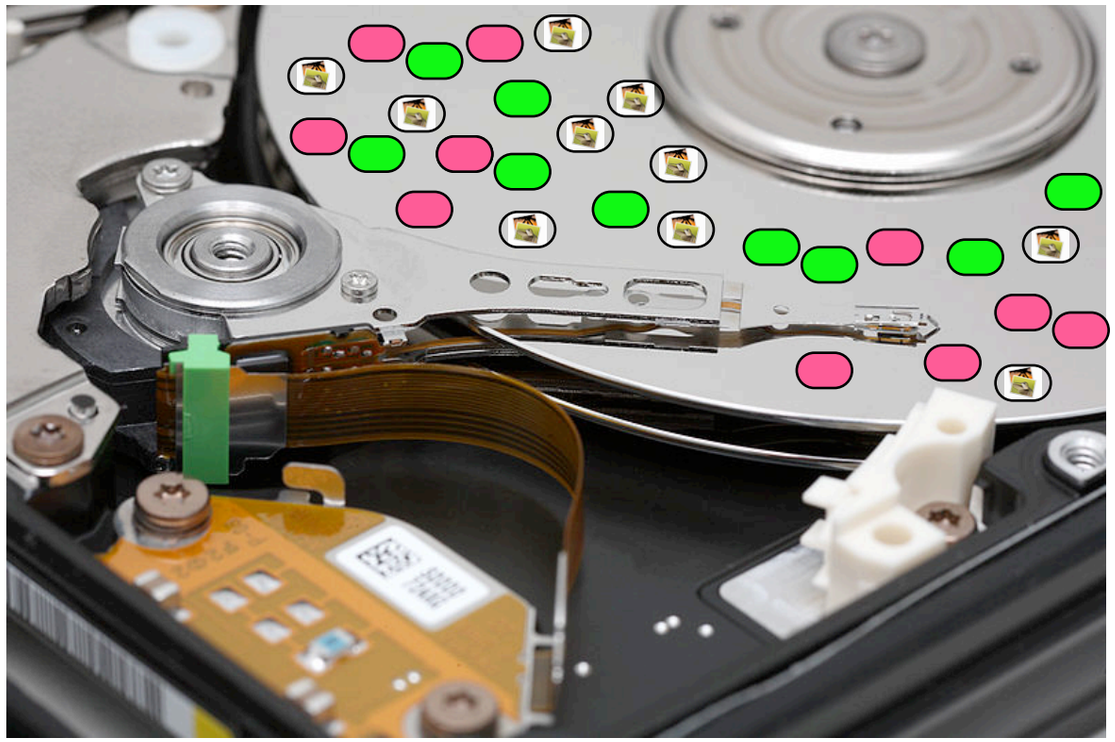
- Only ≈ 2.4% of the disk...
- But it can be a *statistically significant sample*

We can predict the statistics of a *population* by sampling a *randomly chosen sample.*

US elections can be predicted by sampling a few thousand households:



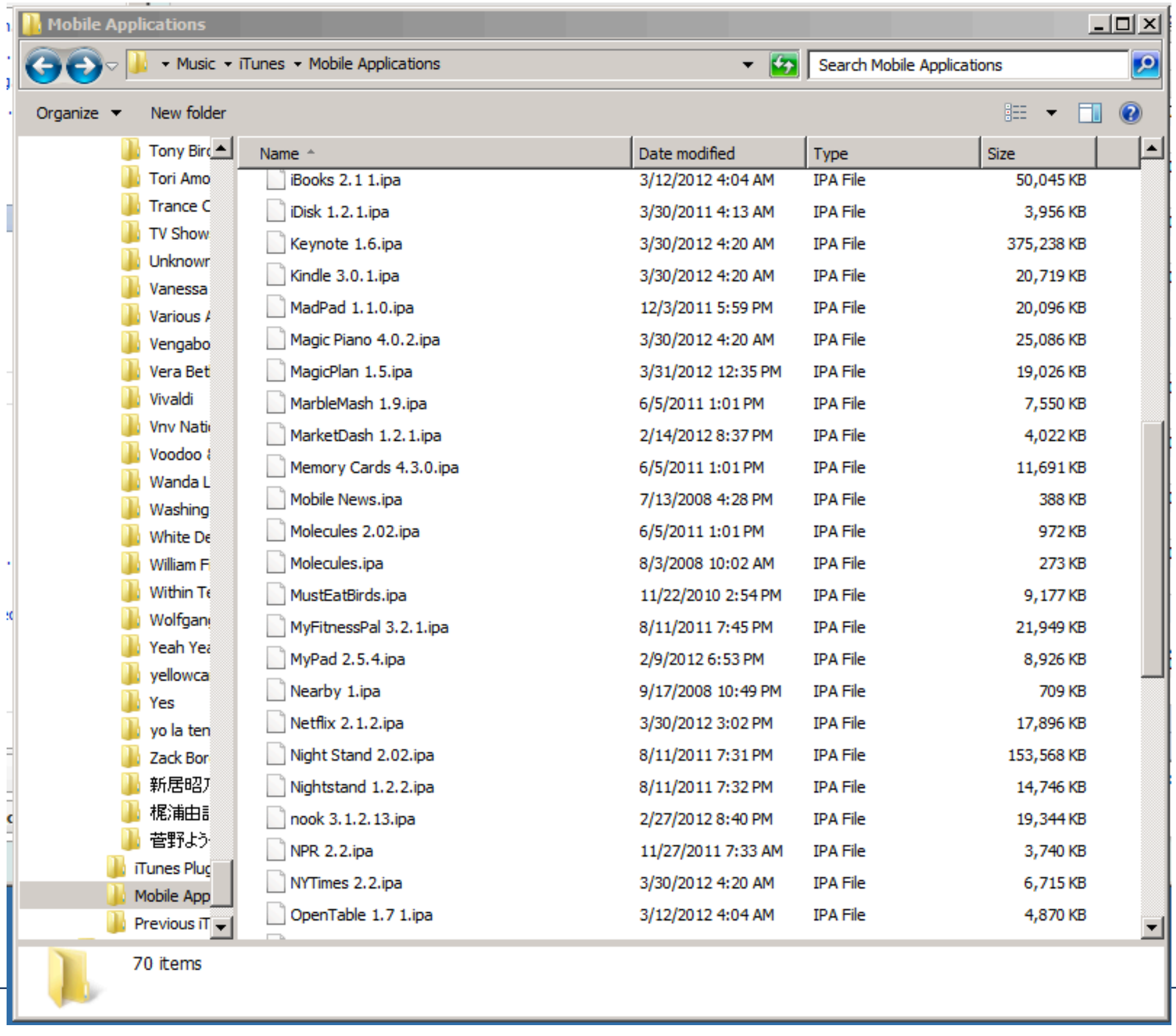
Hard drive contents can be predicted by sampling a few thousand sectors:



The challenge is identifying *likely voters.*

The challenge is *identifying the sectors* that are sampled.

We think of computers as devices with *files*.

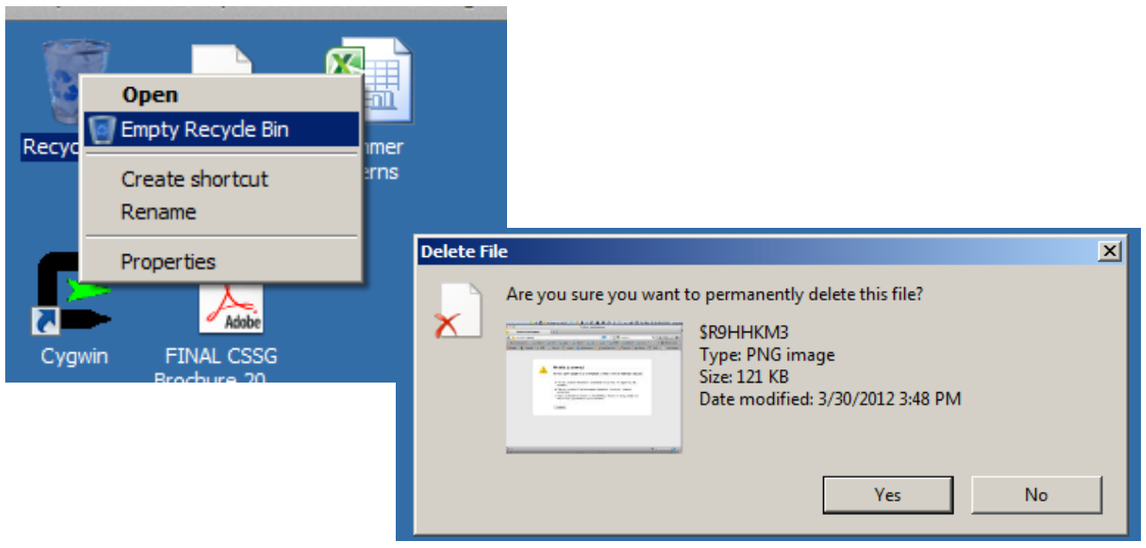


But data on computers is really in three categories:

Resident Data

user files
email messages
[temporary files]

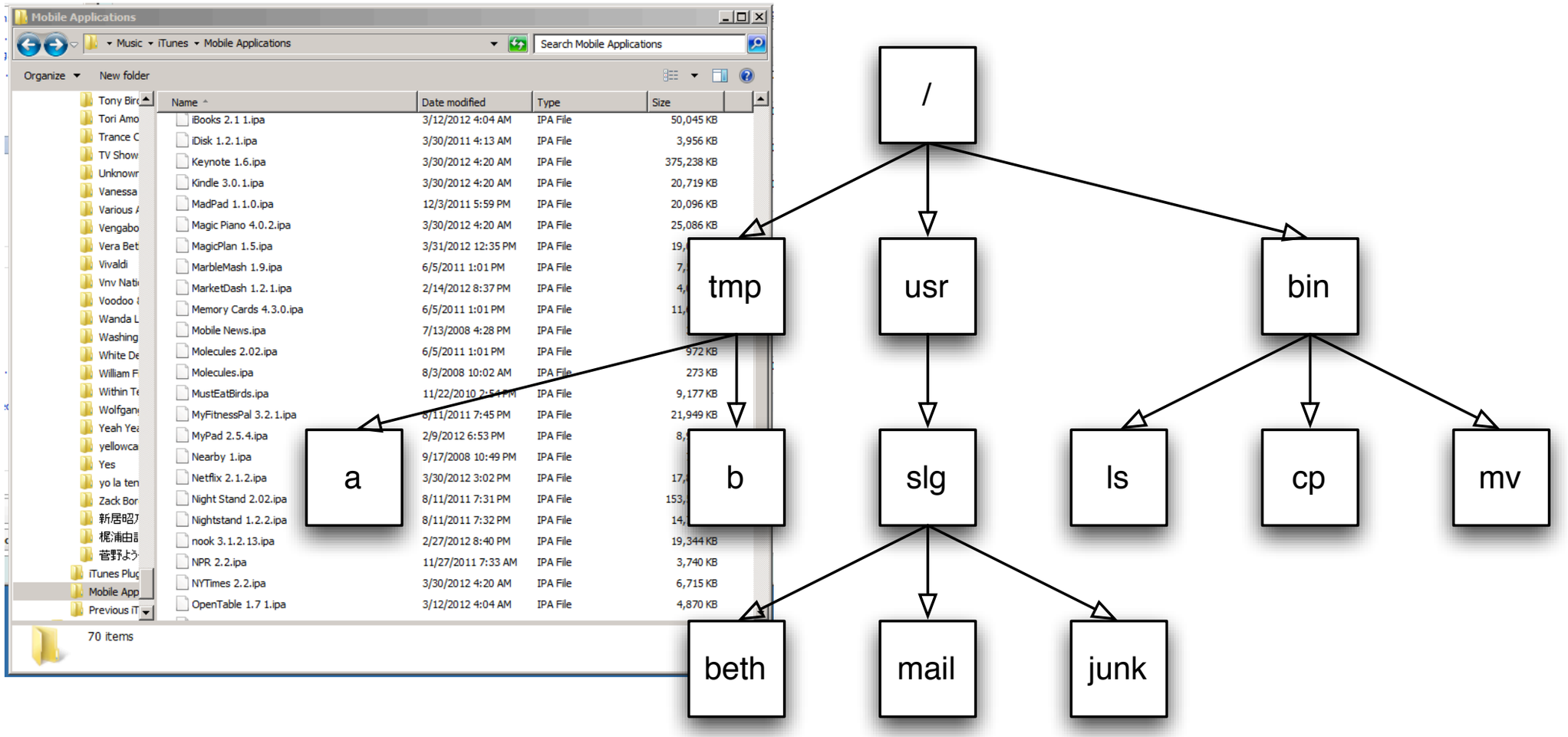
Deleted Data



No Data

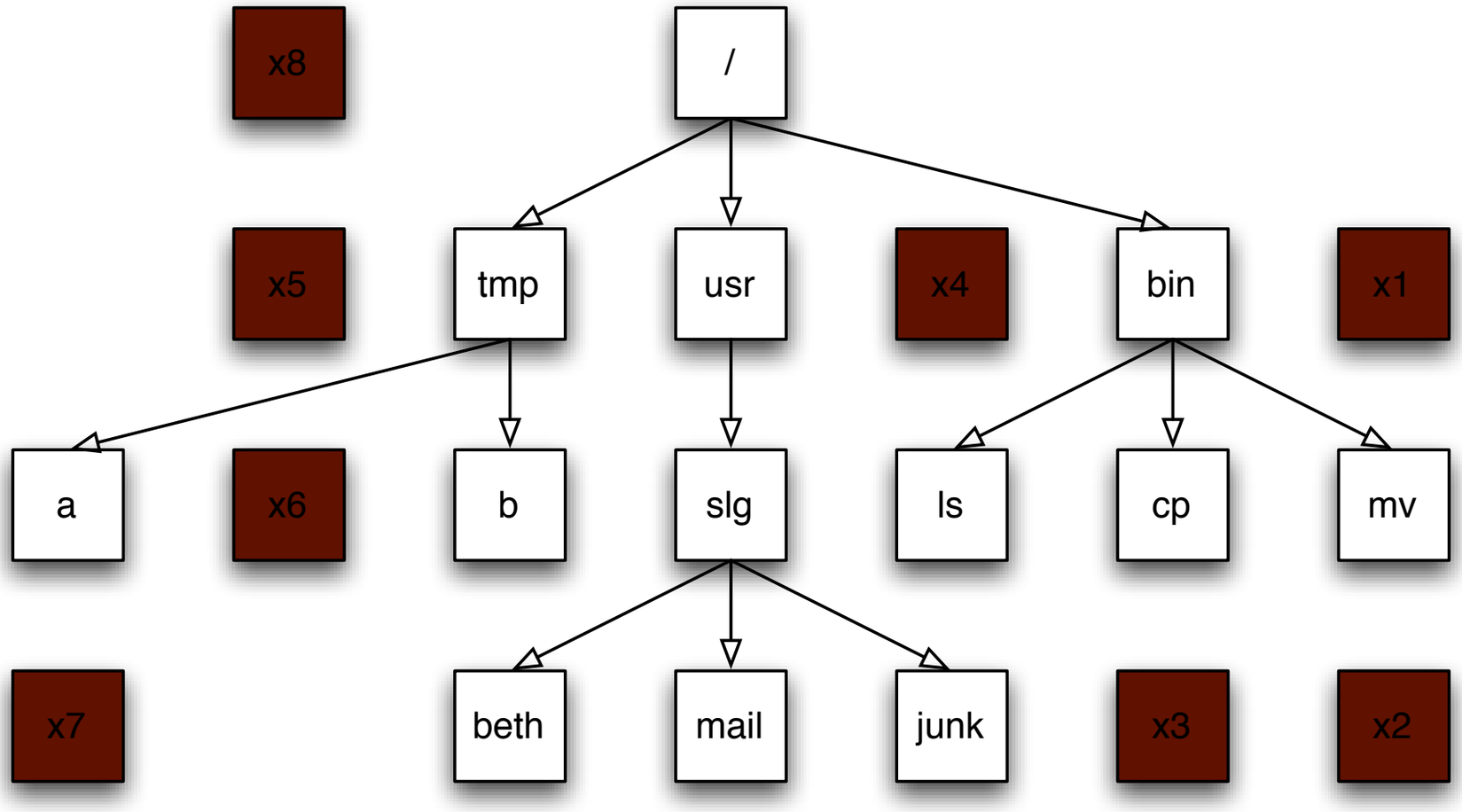
blank sectors

Resident data is the data you see from the root directory. “Allocated” files.



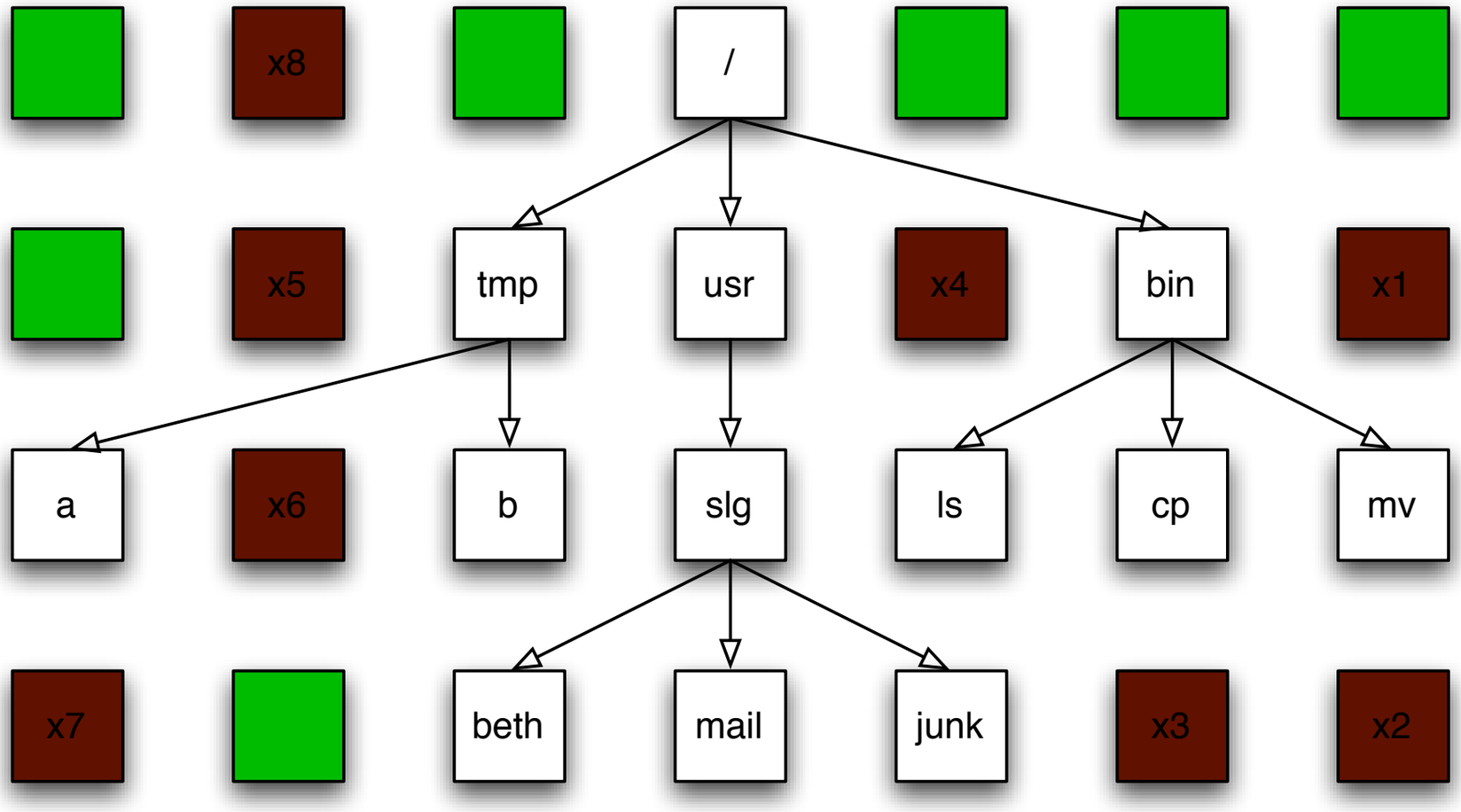
Resident Data

“Deleted data” is on the disk, but can only be recovered with forensic tools.



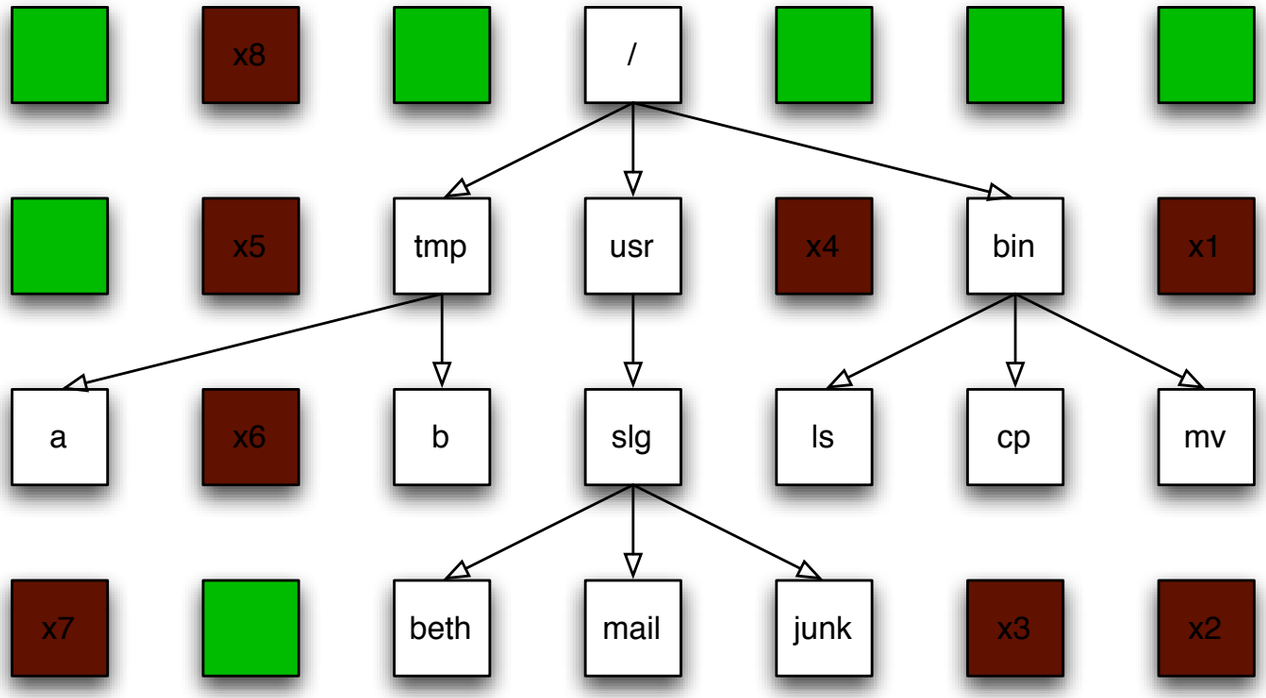
Deleted Data

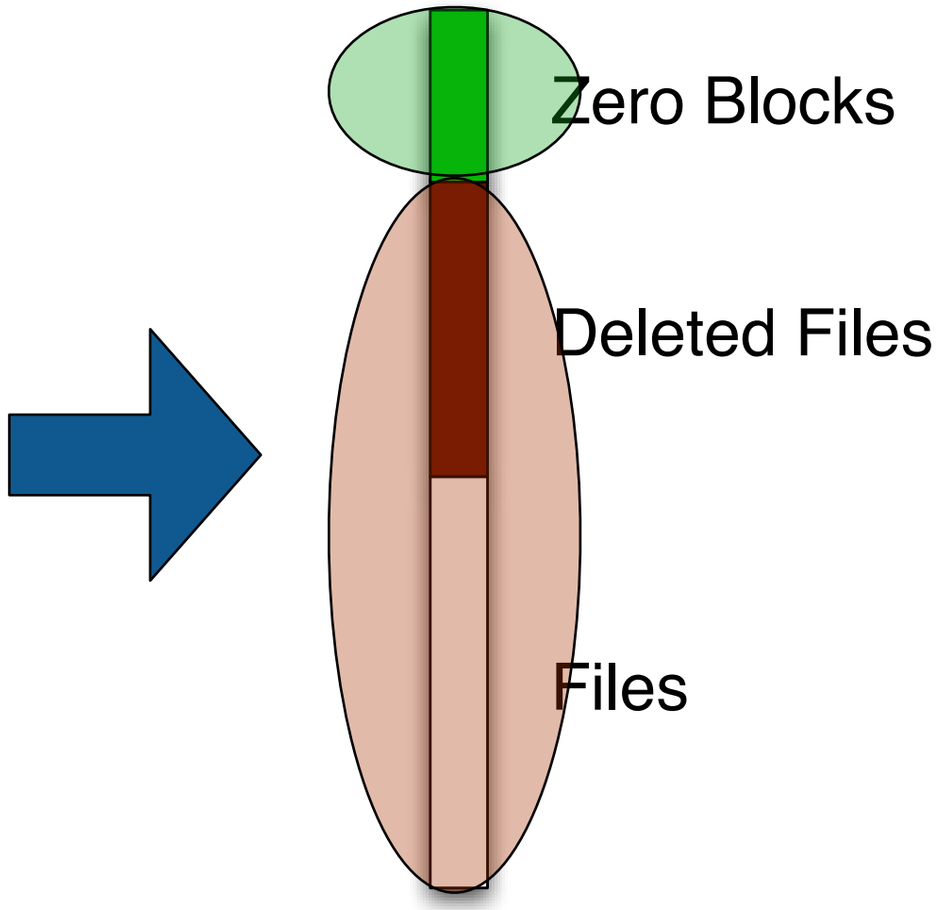
Some sectors are blank.
They have “No data.”



No Data

Sampling can't distinguish *allocated* from *deleted* data.

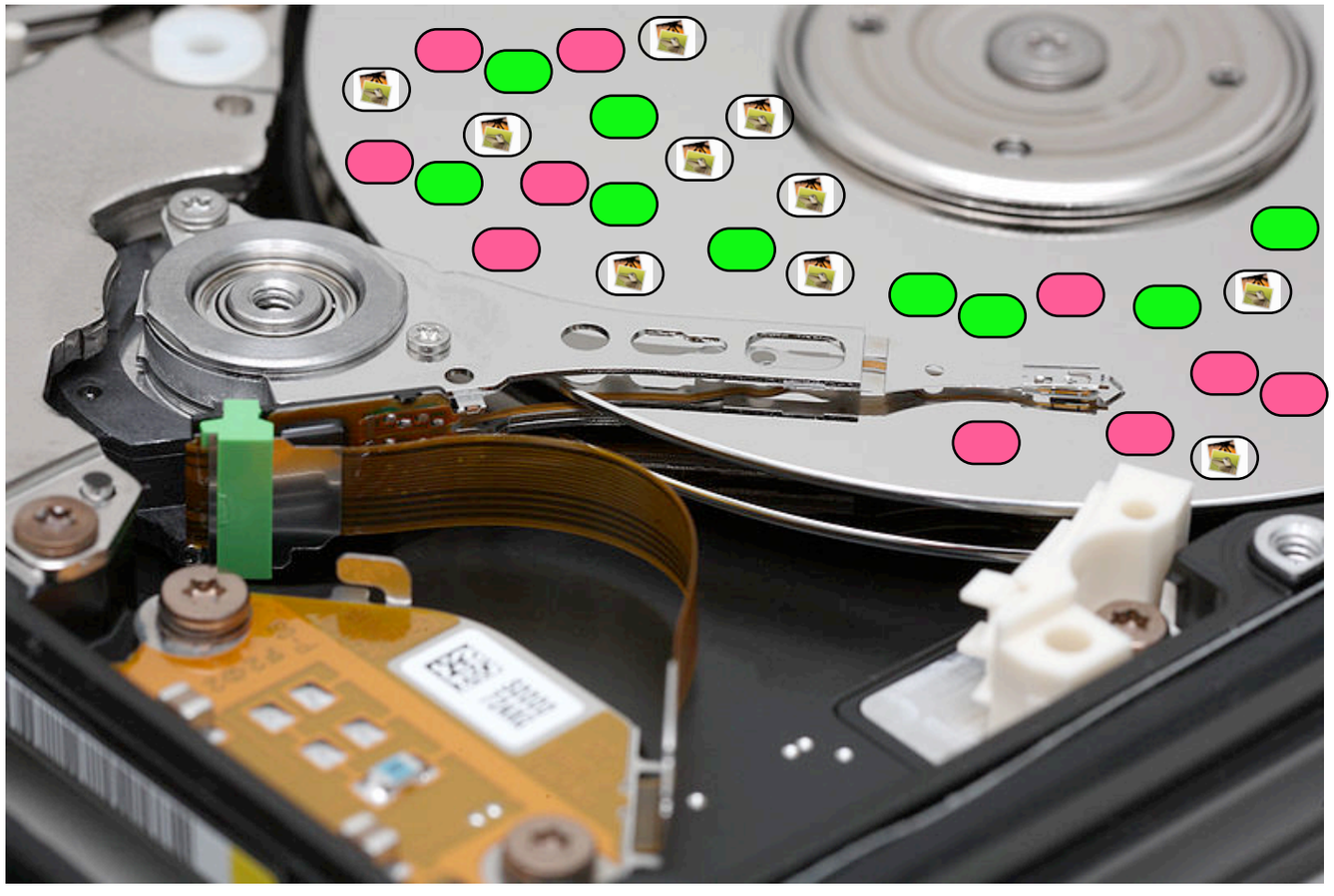




But sampling can tell us about the content of the data

Sampling can tell us the proportion of...

- *blank sectors; video; HTML files; other data types...*
- *data with distinct signatures...*



...provided we can identify it

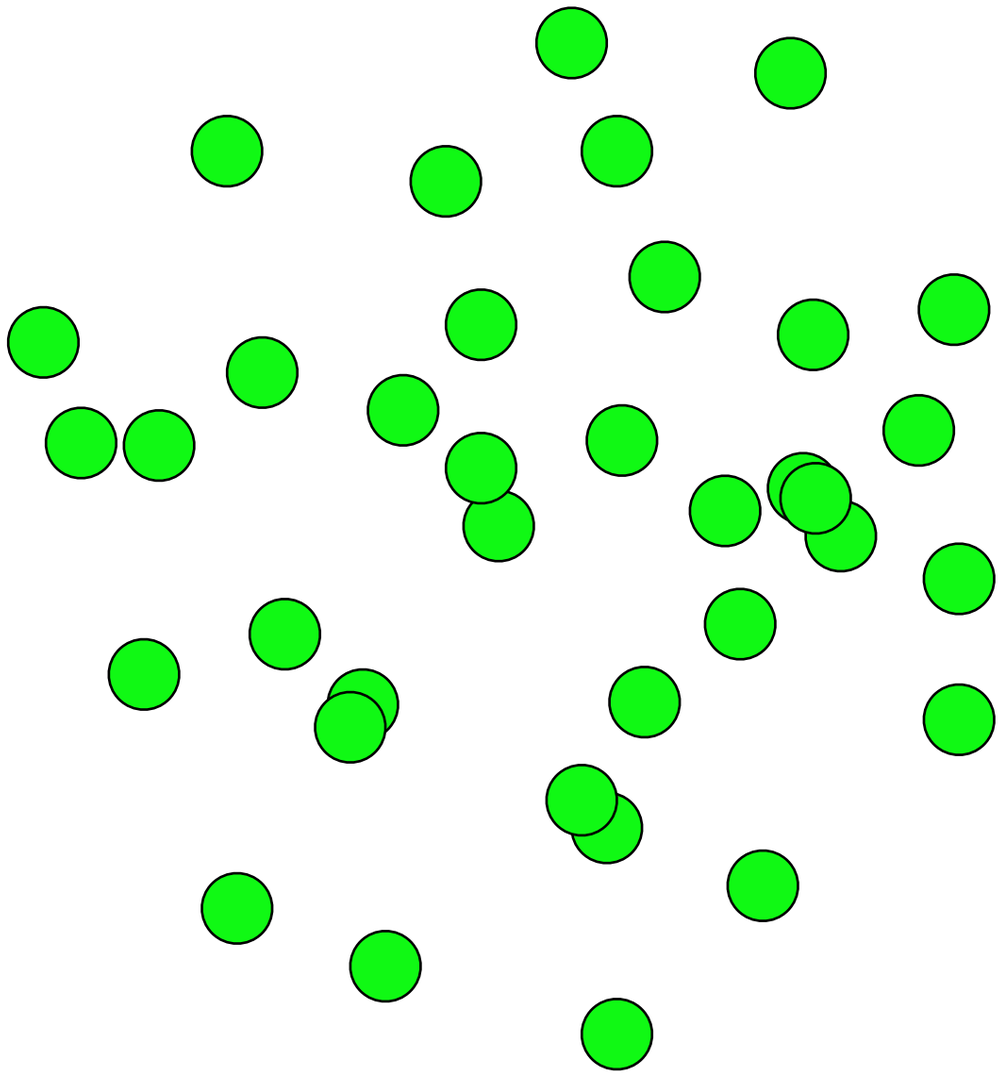
Simplify the problem.
Can we use statistical sampling to verify wiping?

Many organizations discard used computers.

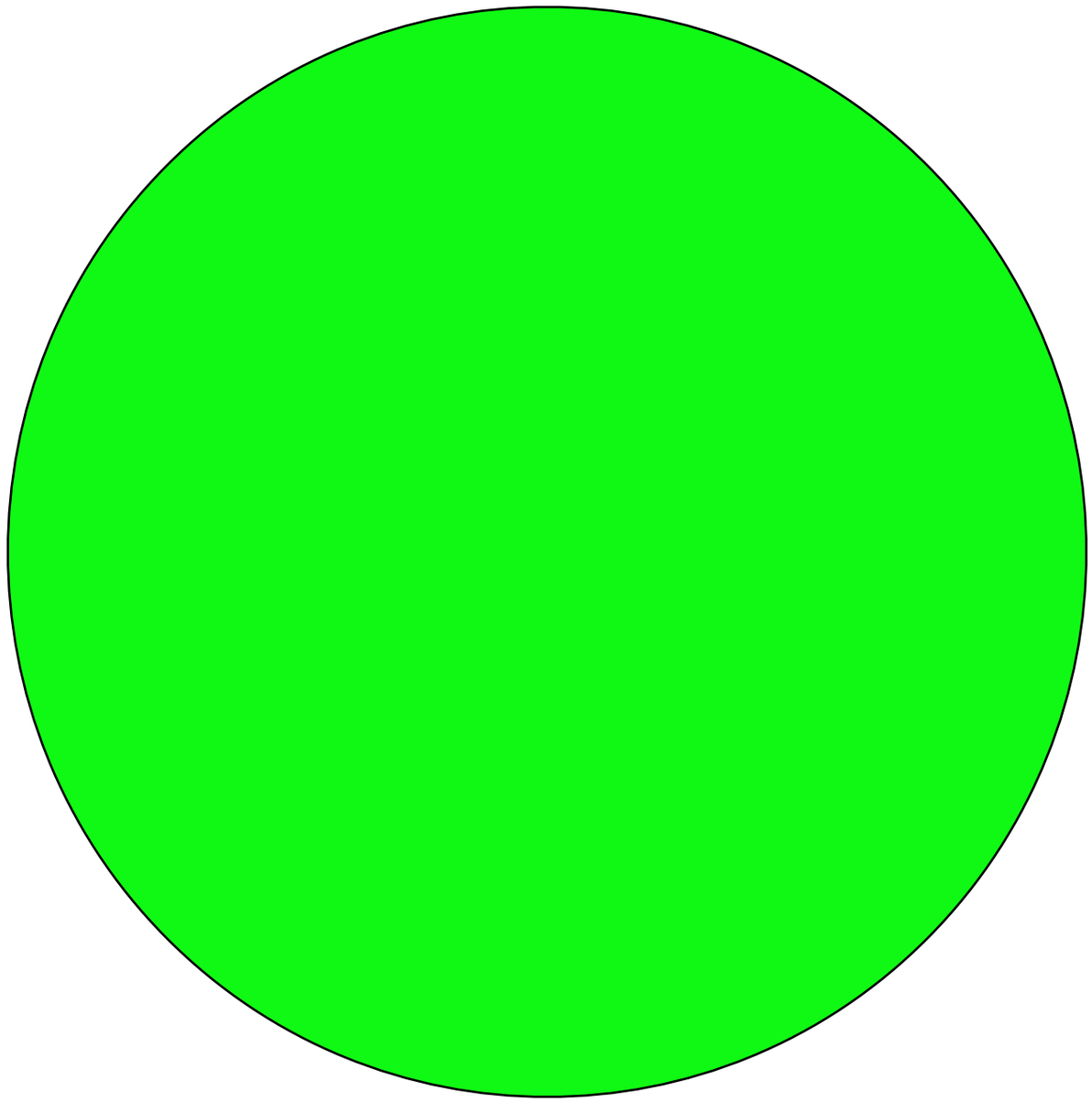
Can we verify if a disk is properly wiped in 5 minutes?

A 1TB drive has 2 billion sectors.
What if we read 10,000 and they are all blank?

A 1TB drive has 2 billion sectors.
What if we read 10,000 and they are all blank?



A 1TB drive has 2 billion sectors.
What if we read 10,000 and they are all blank?

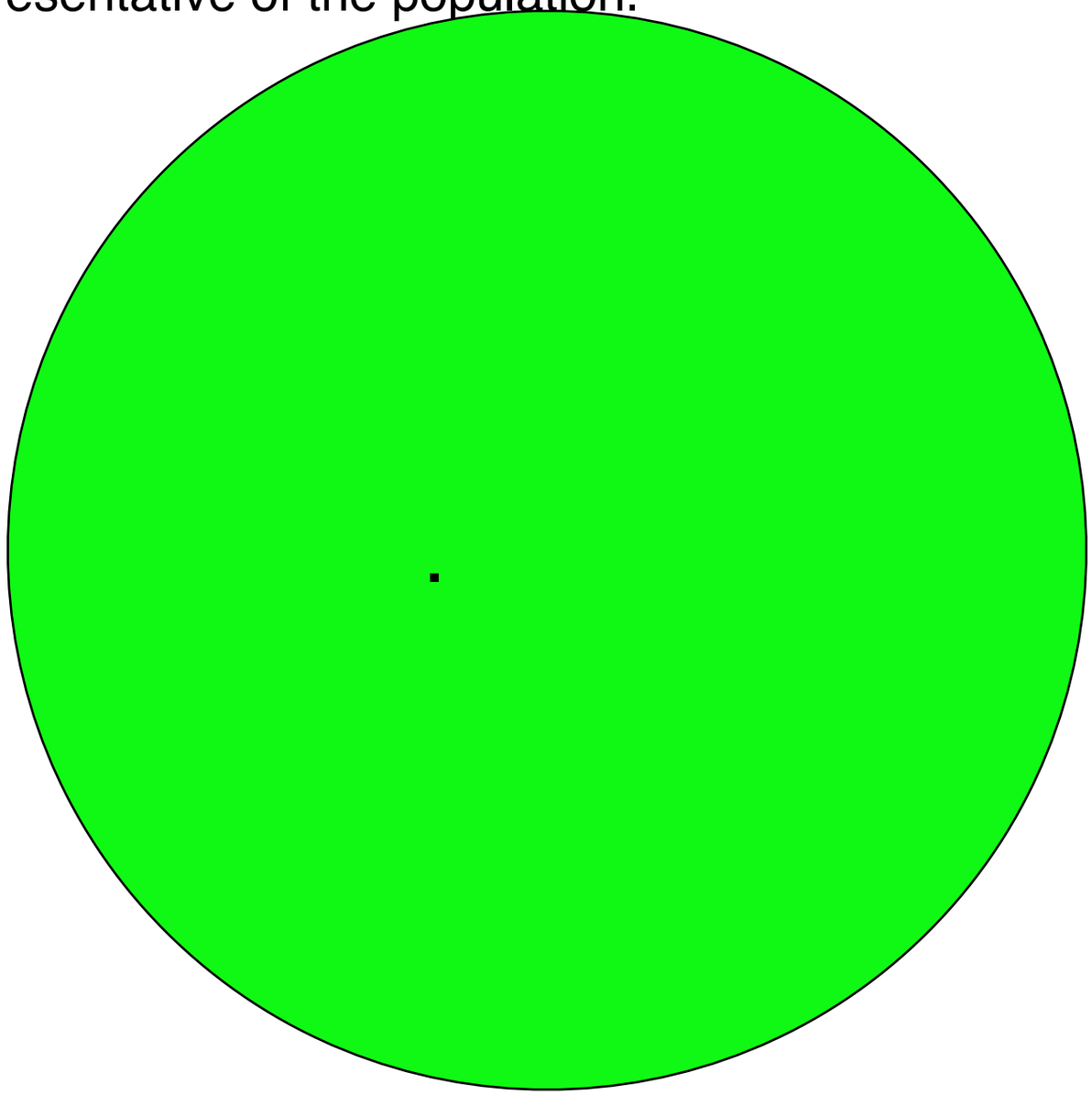


Chances are good that they are all blank.

Random sampling *won't* find a single written sector.

If the disk has 1,999,999,999 blank sectors (1 with data)

- The sample is representative of the population.

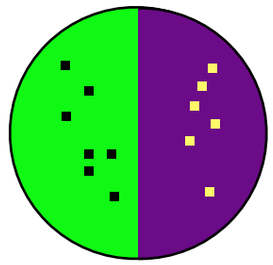


We will only find that 1 sector with exhaustive search.

What about other distributions?

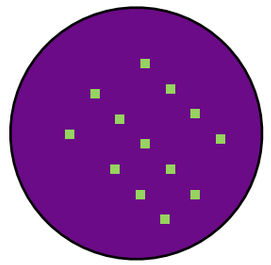
If the disk has 1,000,000,000 blank sectors (1,000,000,000 with data)

- The sampled frequency should match the distribution.
- *This is why we use random sampling.*



If the disk has 10,000 blank sectors (1,999,990,000 with data)...

- .. and we read 10,000 blank sectors
- ... and every sector we read is blank!!! ???
  - *We are incredibly unlucky.*
  - ***Somebody has hacked our random number generator!***



This is an example of the "urn" problem from statistics

Assume a 1TB disk has 10MB of data.

- 1TB = 2,000,000,000 = 2 Billion 512-byte sectors!
- 10MB = 20,000 sectors

Read just 1 sector; the odds that it is blank are:

$$\frac{2,000,000,000 - 20,000}{2,000,000,000} = .99999$$

This is an example of the "urn" problem from statistics

Assume a 1TB disk has 10MB of data.

- 1TB = 2,000,000,000 = 2 Billion 512-byte sectors!
- 10MB = 20,000 sectors

Read just 1 sector; the odds that it is blank are:

$$\frac{2,000,000,000 - 20,000}{2,000,000,000} = .99999$$

Read 2 sectors. The odds that both are blank are:

$$\left(\frac{2,000,000,000 - 20,000}{2,000,000,000}\right)\left(\frac{1,999,999,999 - 20,000}{2,000,000,000}\right) = .99998$$

first pick **second pick** **Odds we may have missed something**

The more sectors picked, the less likely we are to miss the data….

$$P(X = 0) = \prod_{i=1}^{n} \frac{((N - (i-1)) - M)}{(N - (i-1))} \qquad (5)$$

Sampled sectors	Probability of not finding data
1	0.99999
100	0.99900
1000	0.99005
10,000	0.90484
100,000	0.36787
200,000	0.13532
300,000	0.04978
400,000	0.01831
500,000	0.00673

Table 1: Probability of not finding any of 10MB of data on a 1TB hard drive for a given number of randomly sampled sectors. Smaller probabilities indicate higher accuracy.

Non-null data		Probability of not finding data
Sectors	Bytes	with 10,000 sampled sectors
20,000	10 MB	0.90484
100,000	50 MB	0.60652
200,000	100 MB	0.36786
300,000	150 MB	0.22310
400,000	200 MB	0.13531
500,000	250 MB	0.08206
600,000	300 MB	0.04976
700,000	350 MB	0.03018
1,000,000	500 MB	0.00673

Table 2: Probability of not finding various amounts of data when sampling 10,000 disk sectors randomly. Smaller probabilities indicate higher accuracy.

- *Pick 500,000 random sectors*
- *If are all NULL, the disk has p=(1-.00673) chance of having 10MB of non-NULL data*
- *The disk has a 99.3% chance of having less than 10MB of data*

In practice, we use a bigger blocks

Sample with 64K “blocks” instead of 512-byte sectors.

- It takes the same amount of time to read 65,536 bytes as 512 bytes
- Analyze 64K block with a 4K sliding window
- The statistics are much more complicated

Scan local area when interesting data is found.

- If a portion of a JPEG is found, find the front
- If a piece of an encrypted file is found, determine the extent

Update results in real-time

- Provides immediate feedback
- Catches important data faster

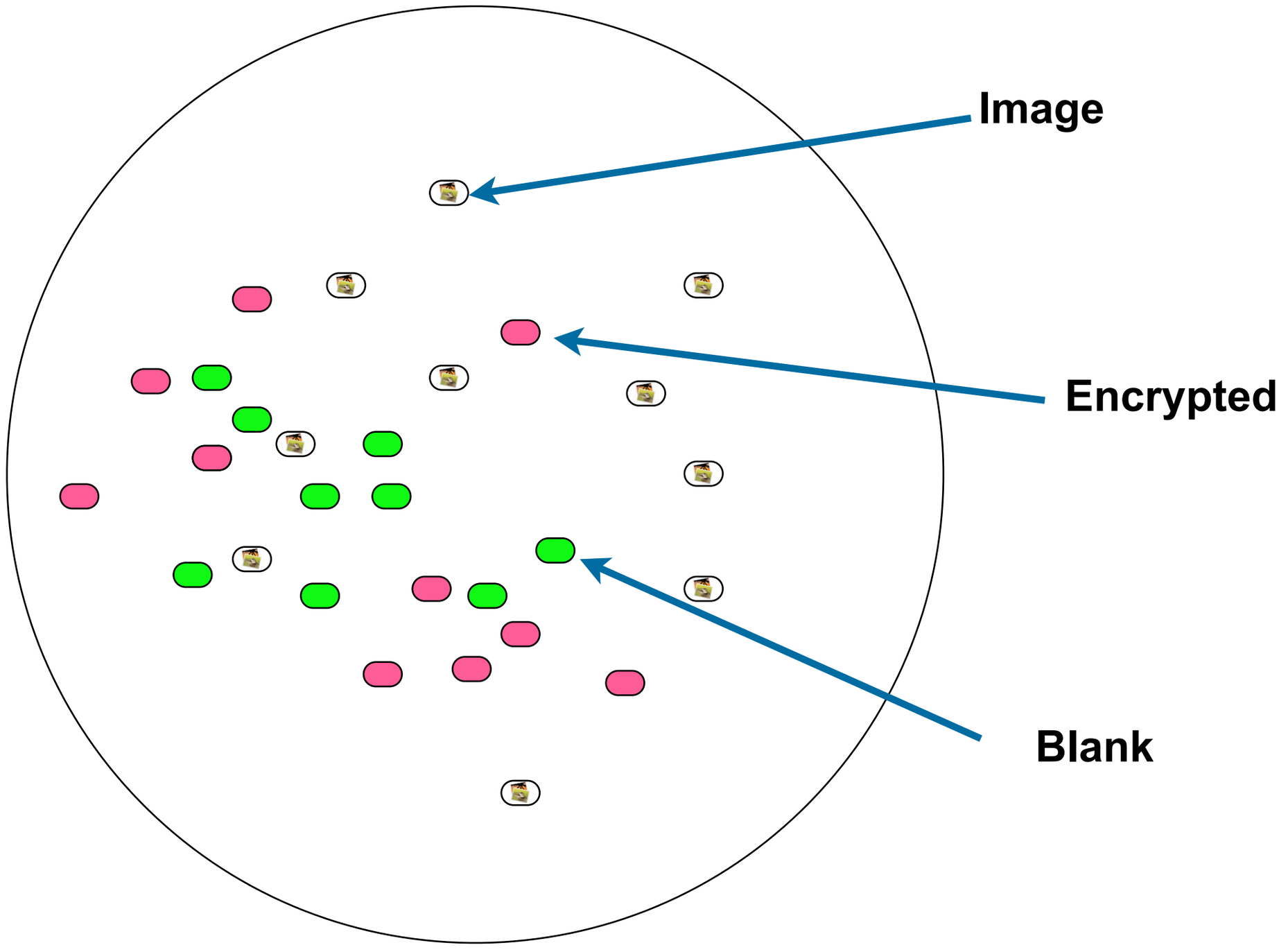
This algorithm is easy to deploy, easy to visualize, easy to explain!

We can use this technique to calculate the size of the TrueCrypt volume on this iPod.

It takes 3+ hours to read all the data on a 160GB iPod.

- Apple bought very slow hard drives.

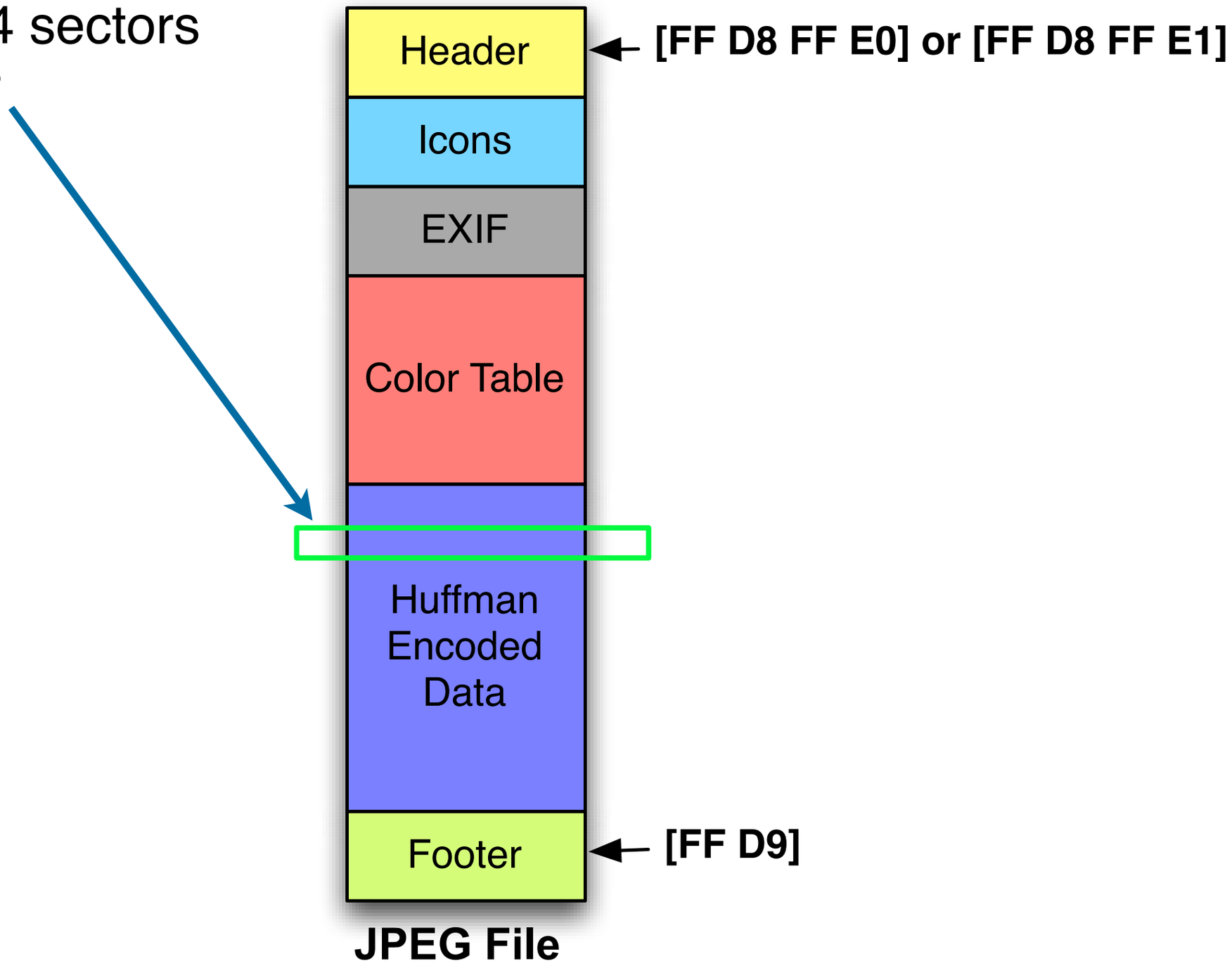
We get a statistically significant sample in two minutes.



The % of the sample will approach the % of the population.

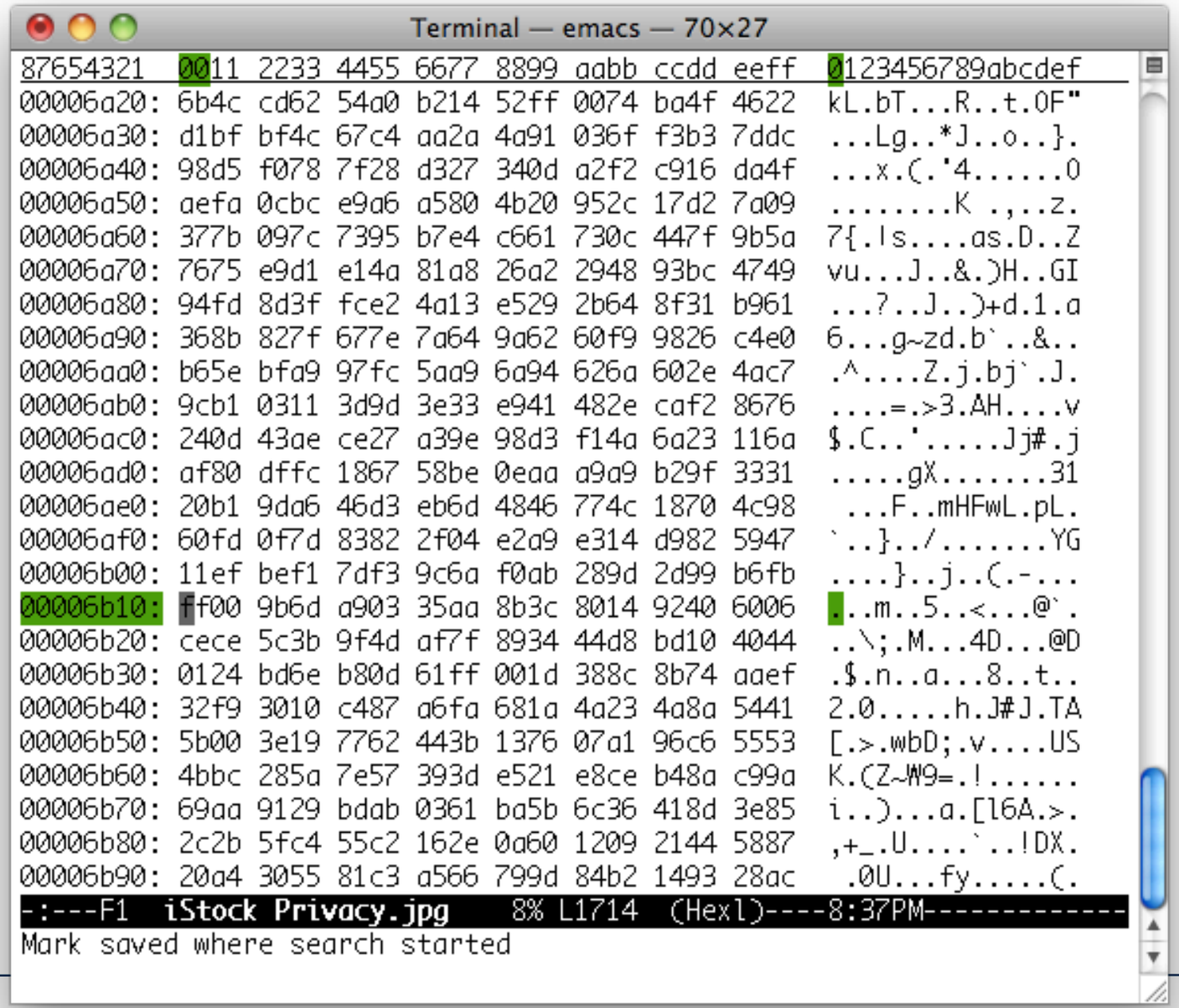
The challenge: identifying a file “type” from a fragment.

Can you identify a JPEG file from reading 4 sectors in the middle?



One approach: hand-tuned discriminators based on a close reading of the specification.

For example, the JPEG format "stuffs" FF with a 00.



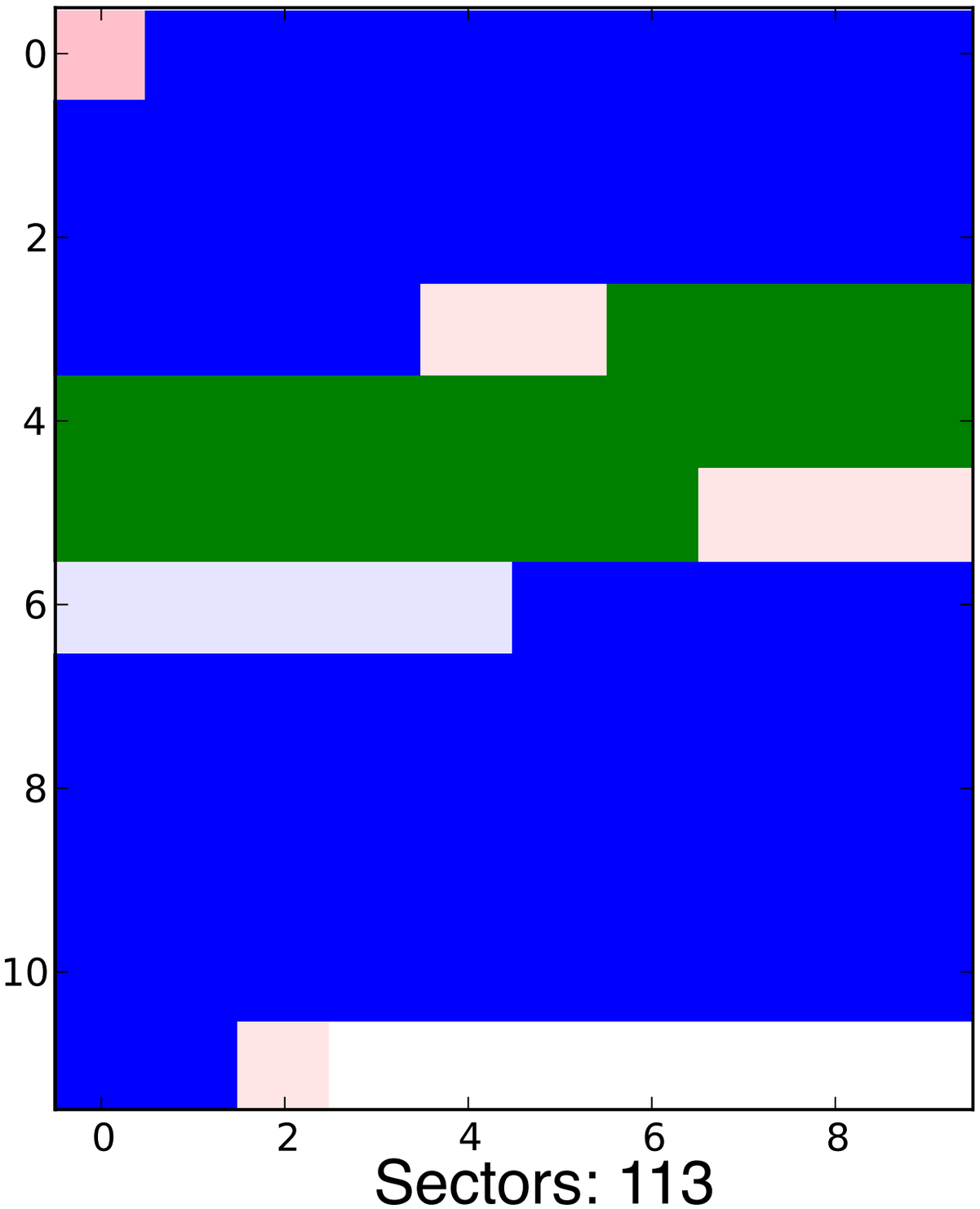
We built detectors to recognize the different parts of a JPEG file.

Bytes: 31,046

Nearly 50% of this 57K file identifies as “JPEG”

000897.jpg

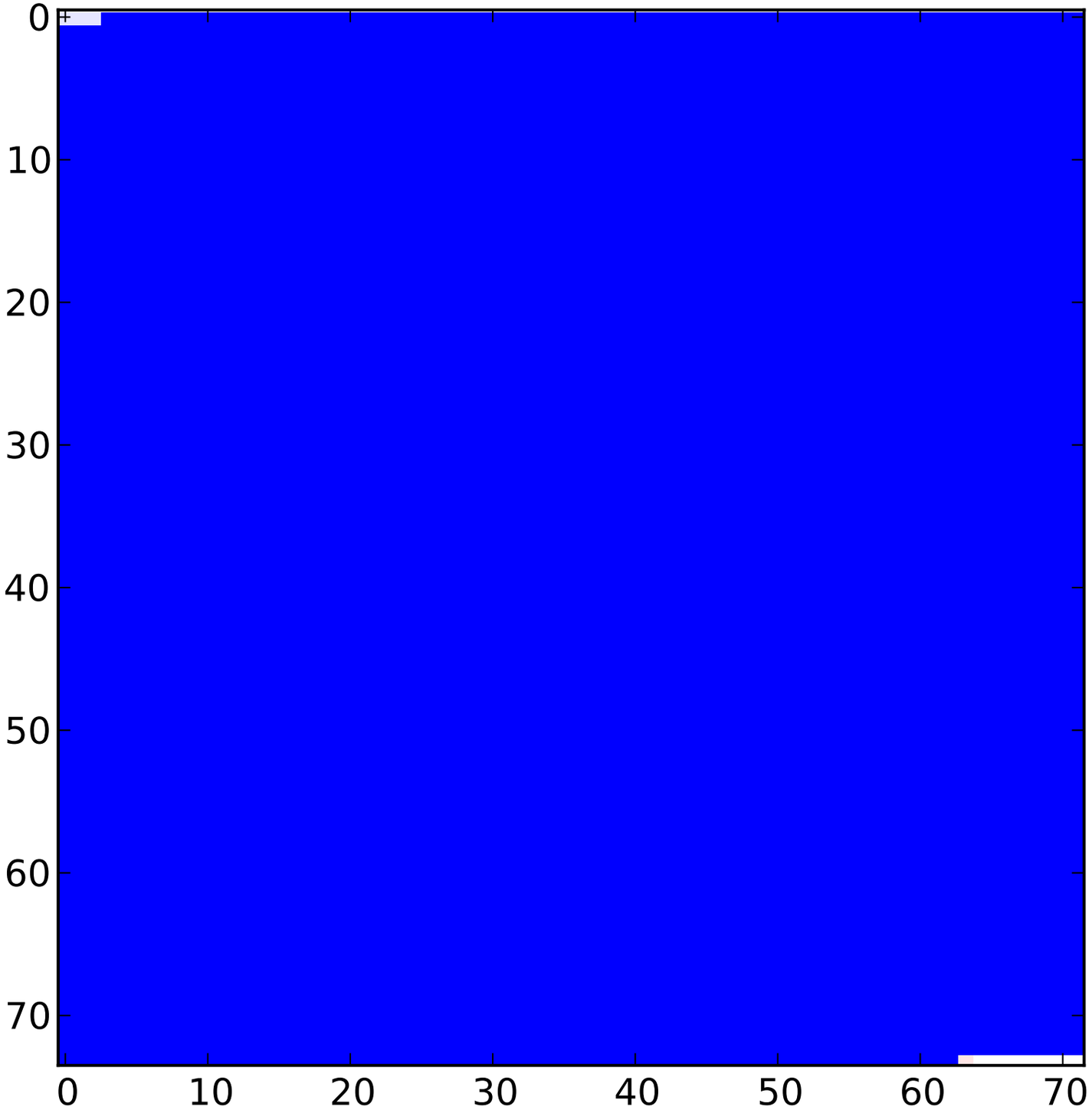
Bytes: 57596



Nearly 100% of this file identifies as “JPEG.”

000888.jpg

Bytes: 2,723,425



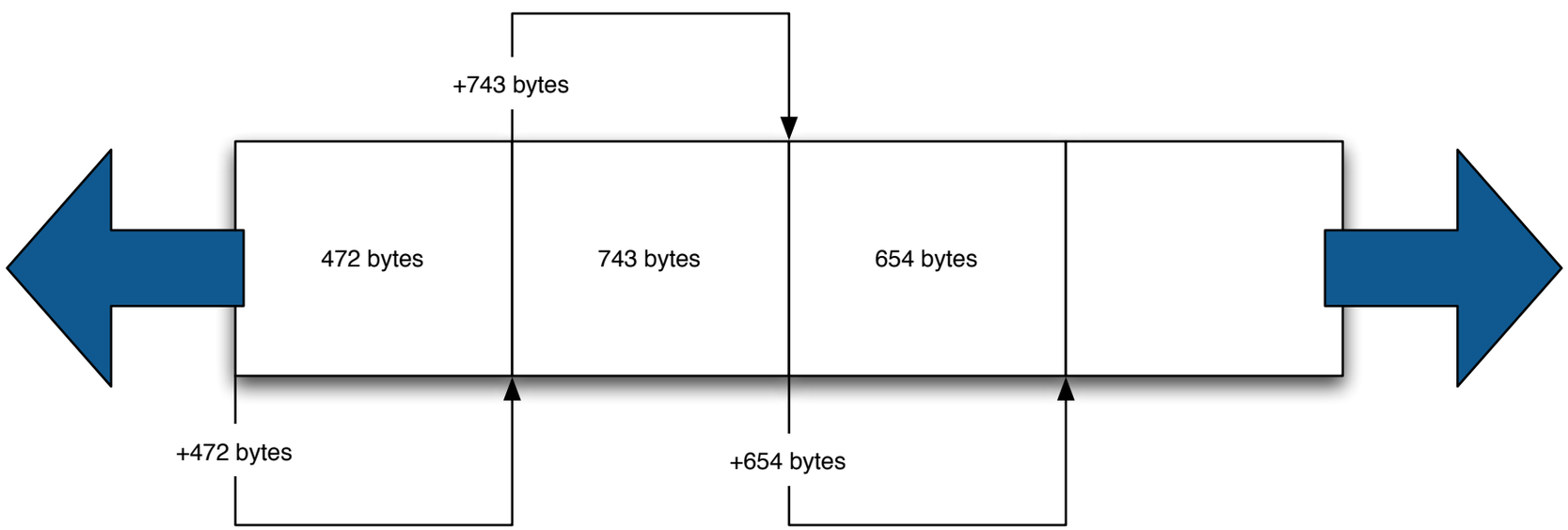
This is called the *file fragment classification problem*

HTML files can be reliably detected with HTML tags

```
<body onload="document.getElementById('quicksearch').terms.focus()">
  <div id="topBar">
    <div class="widthContainer">
          <div id="skiplinks">
            <ul>
              <li>Skip to:</li>
```

MPEG files can be readily identified through framing

- Each frame has a header and a length.
- Find a header, read the length, look for the next header.



We developed five hand-tuned sector identification tools

JPEG — Single images.

MPEG — Frames

Huffman-Coded Data — High Entropy & Autocorrelation

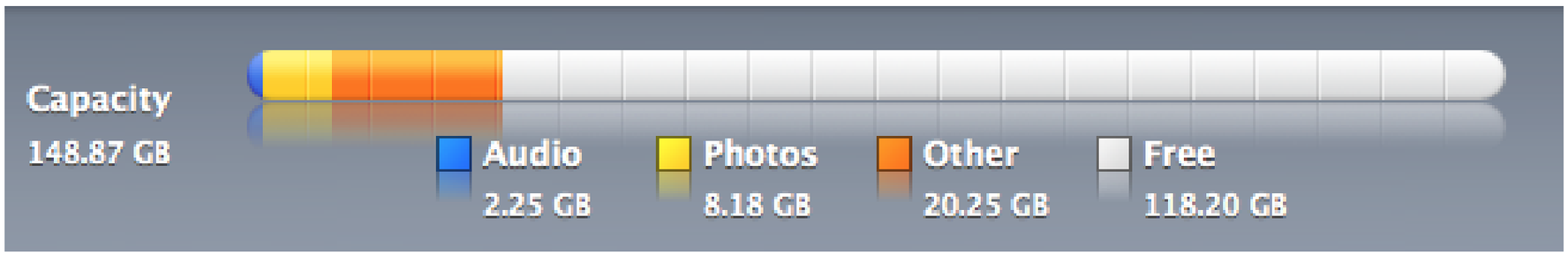
"Random" or "Encrypted" data — High Entropy & No autocorrelation

Distinct Data* — a block from an image, movie, or encrypted file.

208 distinct 4096-byte block hashes

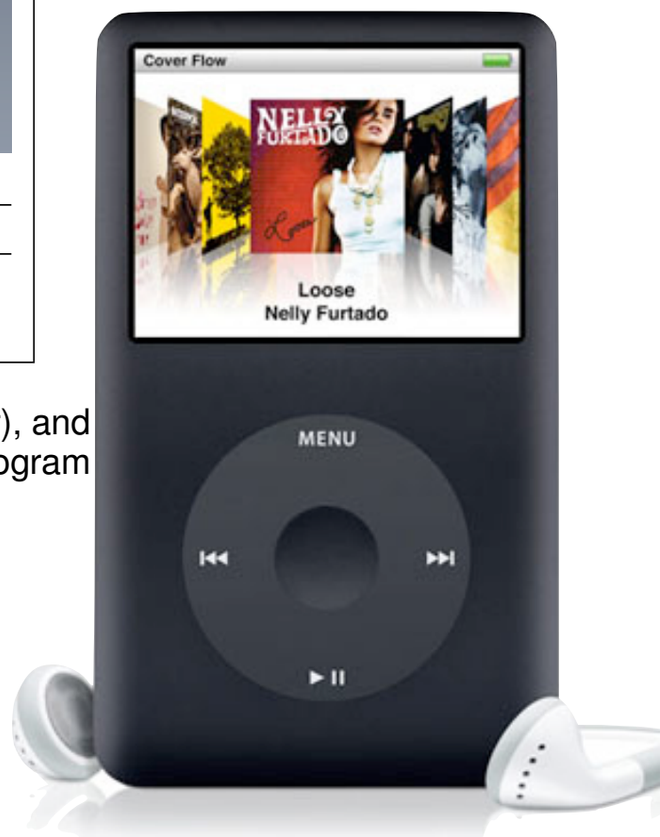
Combine random sampling with sector ID to obtain the forensic contents of a storage device.

Our numbers from sampling are similar to those reported by iTunes.



Audio Data reported by iTunes:	2.25 GiB	2.42 GB	
MP3 files reported by file system:		2.39 GB	
Estimated MP3 usage with random sampling :		2.49 GB	10,000 random samples
		2.71 GB	5,000 random samples

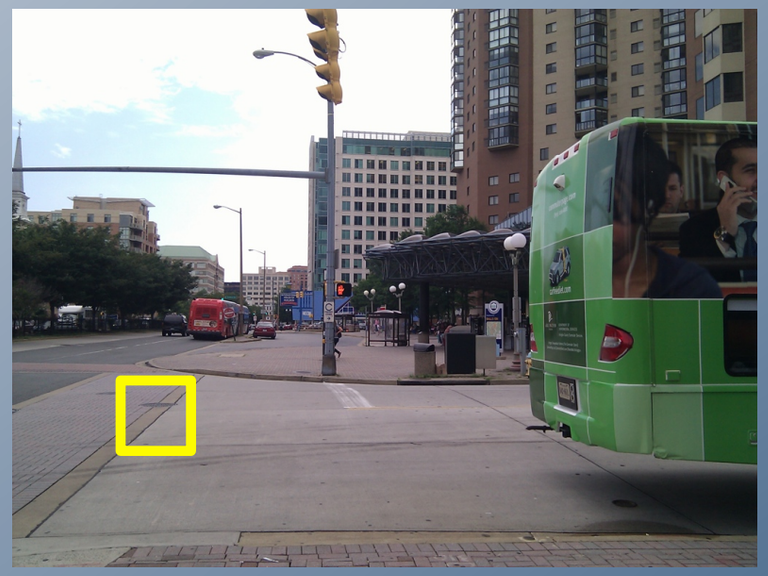
Figure 1: Usage of a 160GB iPod reported by iTunes 8.2.1 (6) (top), as reported by the file system (bottom center), and as computing with random sampling (bottom right). Note that iTunes usage actually in GiB, even though the program displays the "GB" label.



We accurately determined:

- % of free space; % JPEG; % encrypted

—*Simson Garfinkel, Vassil Roussev, Alex Nelson and Douglas White, Using purpose-built functions and block hashes to enable small block and sub-file forensics, DFRWS 2010, Portland, OR*

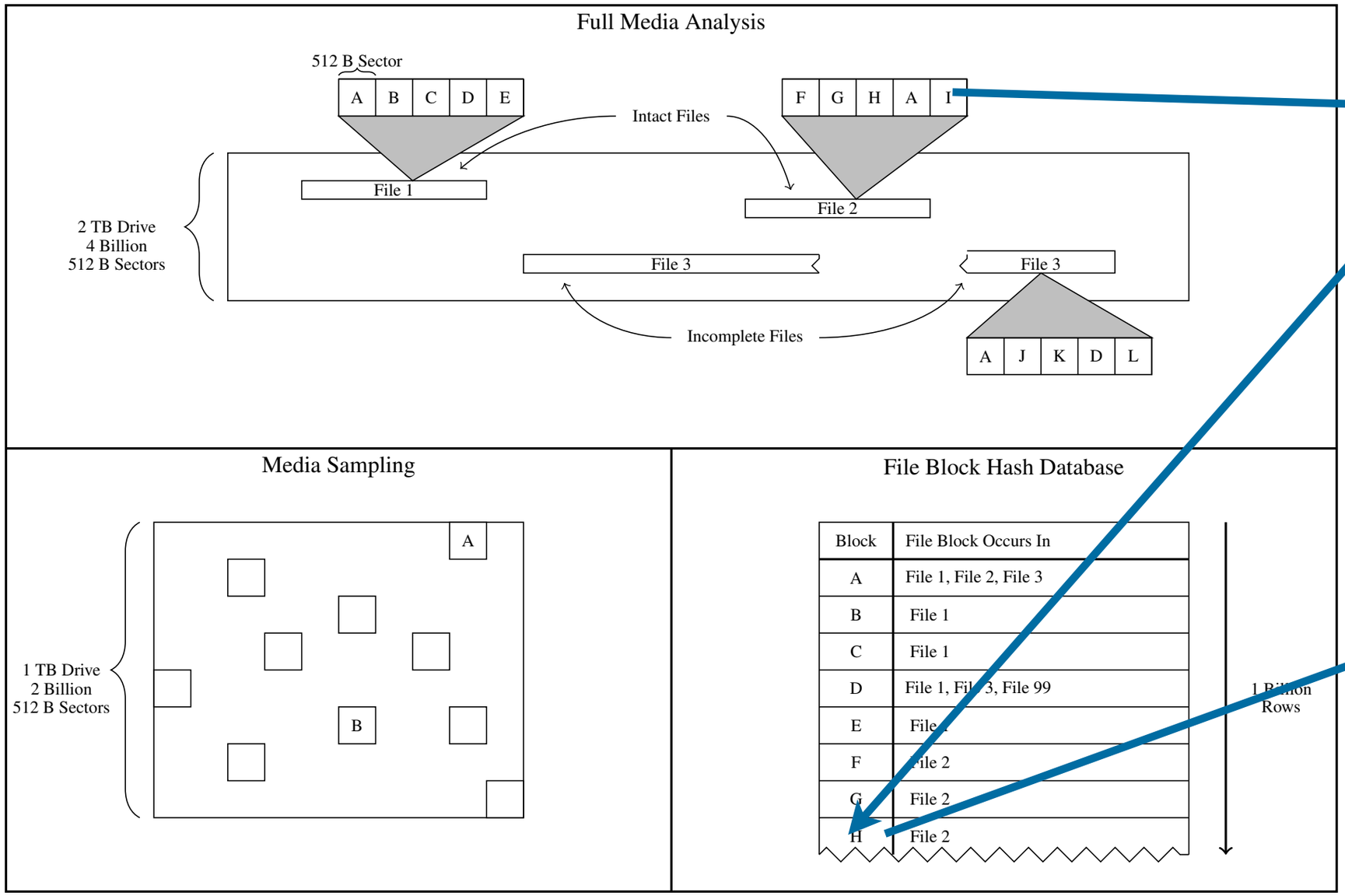


208 distinct 4096-byte block hashes

Finding Known Content with Sector Hashing...

File systems align large files on sector boundaries. We hash file blocks and identify sectors that match.

- File Blocks = file[0:512], file[512:1024], file[1024:1536], ...
- Disk Sectors = byte[512] from the media



208 distinct 4096-byte block hashes

- Known content: when hash(block)=hash(sector)

For this to work, files of interest must have “distinct” sectors.

Using distinct sectors in media sampling and full media analysis to detect presence of documents from a corpus, Kristina Foster, NPS Master’s Thesis, 2012

We measured the prevalence of distinct sectors in three publicly available corpora.

Corpora	# distinct files	Average file size
Govdocs1	974,471	506 KB
OCMalware	2,998,898	427 KB
NSRL 2009	12,236,979	240 KB

These corpora are largely made of distinct sectors.

Classification	**Definition**
Principal Classification	
Singleton	Appear only once in a corpus
Pairs	Appear exactly twice in a corpus
Common	Appear three or more times in a corpus
Secondary Classification	
Entropy	Predictability of byte values in the block (0-8)
Repeating n-gram Block	Contains a repeating n-gram
N-gram Size	The size *n* of the repeating n-gram (0-half of block size)

Table 3.1: A list of the principal and secondary classification of file blocks in the corpora.

	Govdocs1		**OCMalware**		**NSRL2009**	
Total Unique Files	974,741		2,998,898		12,236,979	
Average File Size	506 KB		427 KB		240 KB	
Block Size: 512 B						
Singletons	911.4M	(98.93%)	1,063.1M	(88.69%)	n/a	n/a
Pairs	7.1M	(.77%)	75.5M	(6.30%)	n/a	n/a
Common	2.7M	(.29%)	60.0M	(5.01%)	n/a	n/a
Block Size: 4 KiB						
Singletons	117.2M	(99.46%)	143.8M	(89.51%)	567.0M	(96.00%)
Pairs	0.5M	(.44%)	9.3M	(5.79%)	16.4M	(2.79%)
Common	0.1M	(.11%)	7.6M	(4.71%)	7.1M	(1.21%)

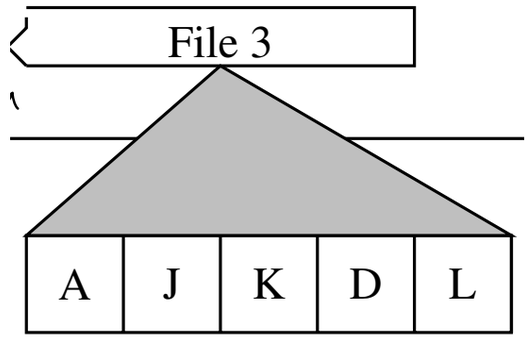
Table 5.1: Occurrences of singleton, pair and common blocks in the Govdocs1, OCMalware and NSRL2009 corpora.

This means we *can* use distinct sectors to find known content.

	Govdocs1		OCMalware		NSRL2009	
Total Unique Files	974,741		2,998,898		12,236,979	
Average File Size	506 KB		427 KB		240 KB	
Block Size: 4 KiB						
Singletons	117.2M	(99.46%)	143.8M	(89.51%)	567.0M	(96.00%)
Pairs	0.5M	(.44%)	9.3M	(5.79%)	16.4M	(2.79%)
Common	0.1M	(.11%)	7.6M	(4.71%)	7.1M	(1.21%)

Method #1 — Full media sampling

- Read & hash every disk sector.
- Lookup hash values in a database of block hashes.
- Distinct hash imply presence of files.
- Advantage: Can find a single sector of target content



Method #2 — Random sampling

- Read & hash randomly chosen sectors.
- Lookup hash values in a database of block hashes.
- Distinct hash implies presence of files.
- Advantage: Can find presence of target content *very quickly*

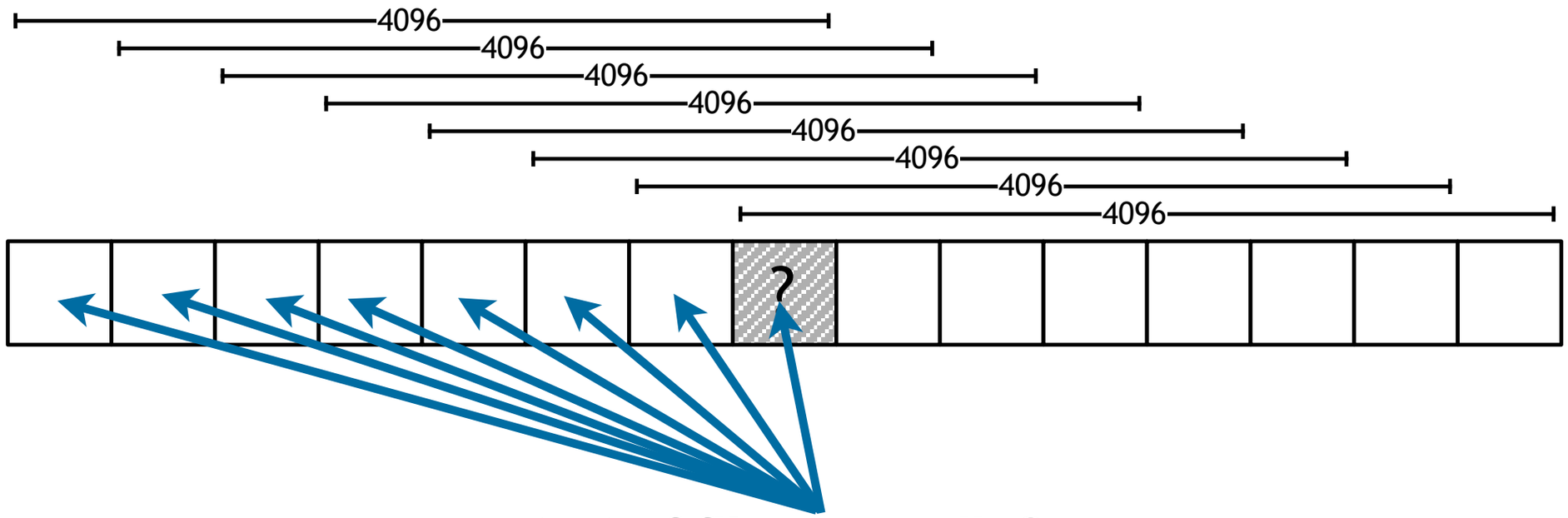
Significant hash and database requirements.

1TB data in 208 minutes

- ≃ 80 Mbyte/sec
- ≃ 150,000 512-byte sectors/sec
- ≃ 150,000 database lookups/sec

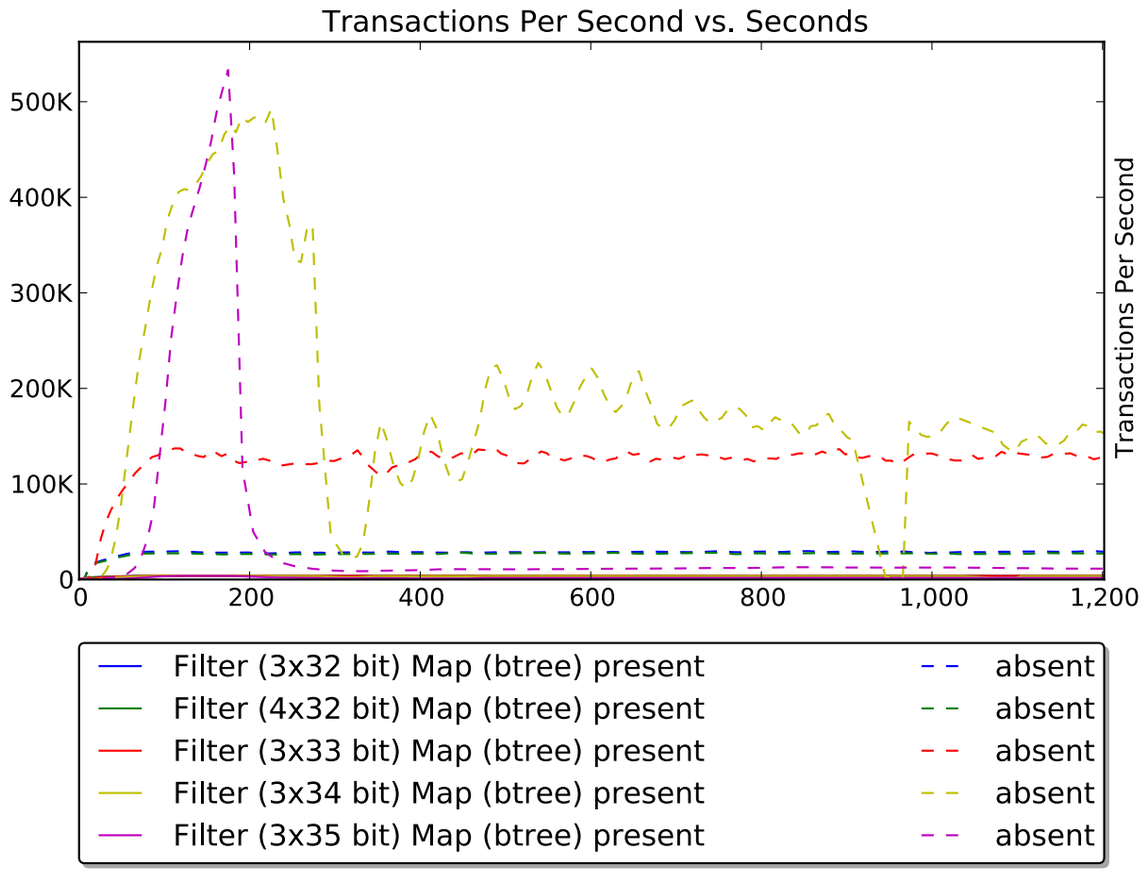
		
Minutes	208	5
Max Data	1 TB	36 GB
Max Seeks		90,000

Alignment uncertainty gives 4096-byte sectors same performance requirements:

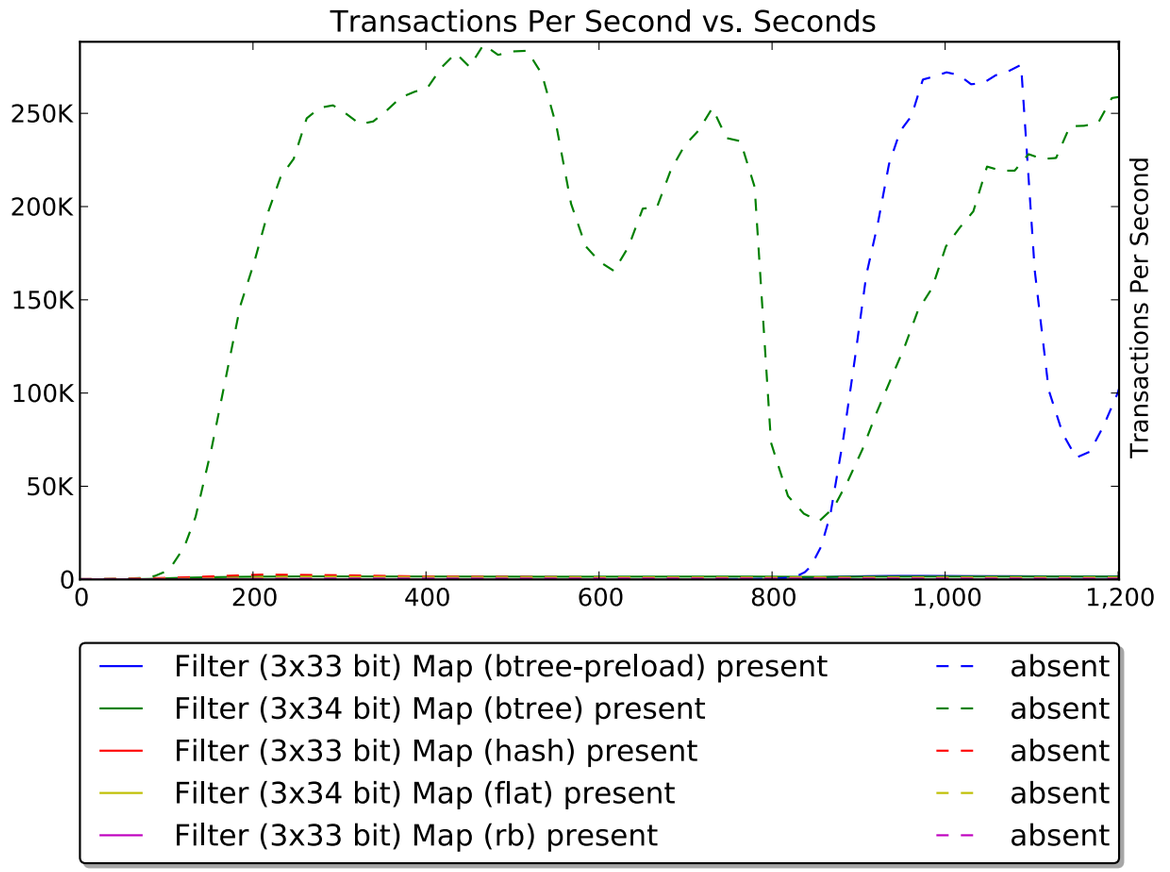


Memory-mapped B-tree with Bloom filter pre-filter.
1 billion hash values with 64 bit file id; 72GB database

1 billion hashes = 512 GB of target data



Laptop with data on ESATA SSD



Laptop with data on USB SSD

Hardware: 8GiB Laptop; 250GB external SSD.

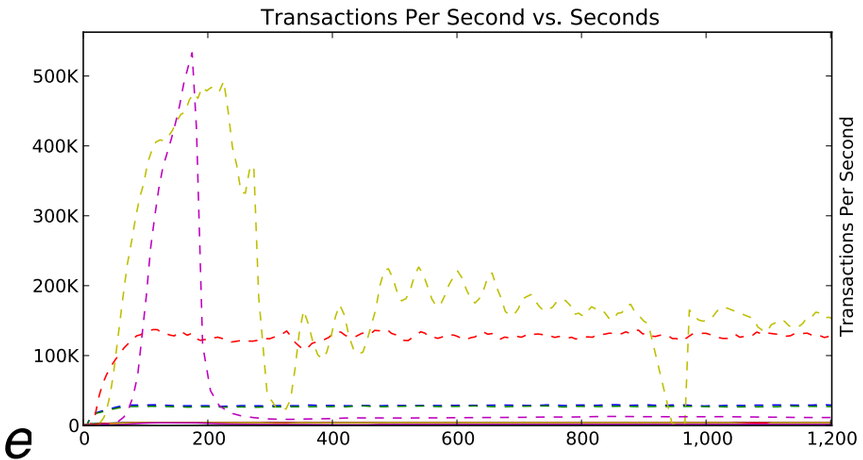
— *"Distinct sector hashes for target file detection," Young, Garfinkel, Foster & Fairbanks, to appear in IEEE Computer, 2013*

Putting it all together, we have breakthrough technology.

Field deployable on a laptop.

Rapidly search for known contraband:

- 1TB subject hard drive.
- Looking for 10MB of data from a corpus of 512GB
  - —*a10MB video segment; 10 Microsoft Word file of 1MB e*
- Determine with 95% certainty if data is present or absent
- Sectors to randomly sample: 300,000
- Samples per second: 300 (worst case)
- Seconds: 1,000 = 17 minutes
  - —*We can do dramatically better reading 64K at a time (rather than 512B)*

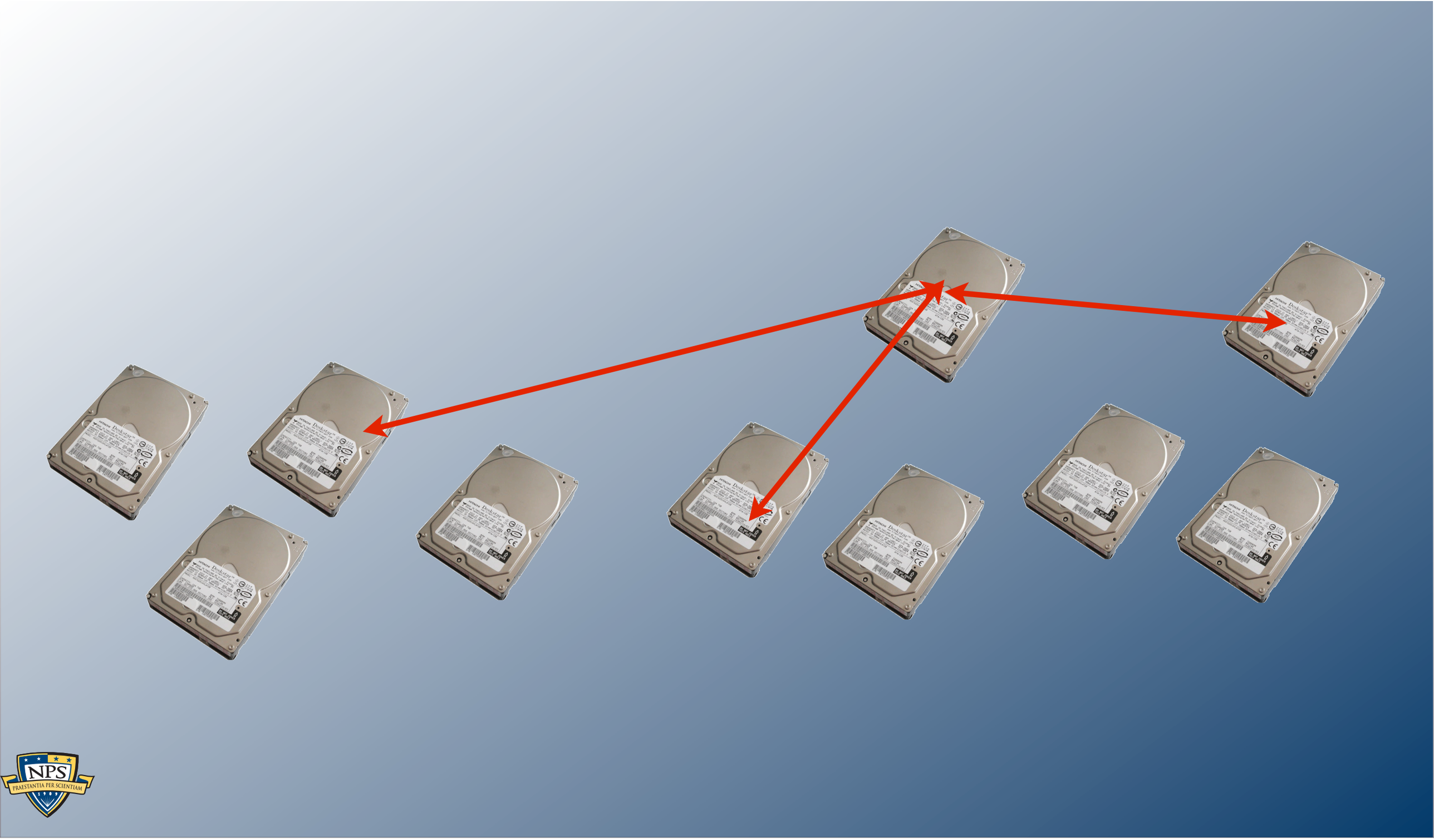


Find a single sector of known contraband:

- 1TB subject hard drive
- Time to read data & search database: 208 minutes

Technique is file type and file system agnostic

- —*JPEG; Video; MSWord; Encrypted PDFs...*
- —*provided data is not modified when copied or otherwise re-coded*



Where do we go from here?

DF has many important areas for research

Algorithm development.

- Adopting to **different kinds of data**.
- **Different resolutions**
- **Larger amounts of data (40TB—40PB)**

Software that can…

- Automatically identify outliers and inconsistencies
- Automatically present complex results in simple, straightforward reports
- Combine stored data, network data, and Internet-based information

Many of the techniques here are also applicable to:

- Social Network Analysis
- Personal Information Management
- Data mining unstructured information

—Many opportunities for funding, too!

Our challenge: innovation, reliability, scale & community

Most innovative forensic tools **fail when they are deployed**

- Production data *much larger* than test data
  - —*One drive might have 10,000 email addresses, another might have 2,000,000*
- Production data *more heterogeneous* than test data
- Analysts have less experience & time than tool developers.

How to address?

- Attention to usability & recovery
- High Performance Computing for testing
- Programming languages that are *safe* and *high-performance*
- Leverage Open Source Software and Community

Moving research results from lab to field is itself a research problem

In summary, there is an urgent need for fundamental research in automated computer forensics.

Most work to date has been data recovery and reverse engineering.

- User-level file systems
- Recovery of deleted files.

To solve tomorrow's hard problems, we need:

- Algorithms that exploit large data sets (>10TB)
- Machine learning to find *outliers* and *inconsistencies.*
- Algorithms tolerant of data that is *dirty* and *damaged.*

Questions?

Work in automated forensics is *inherently interdisciplinary.*

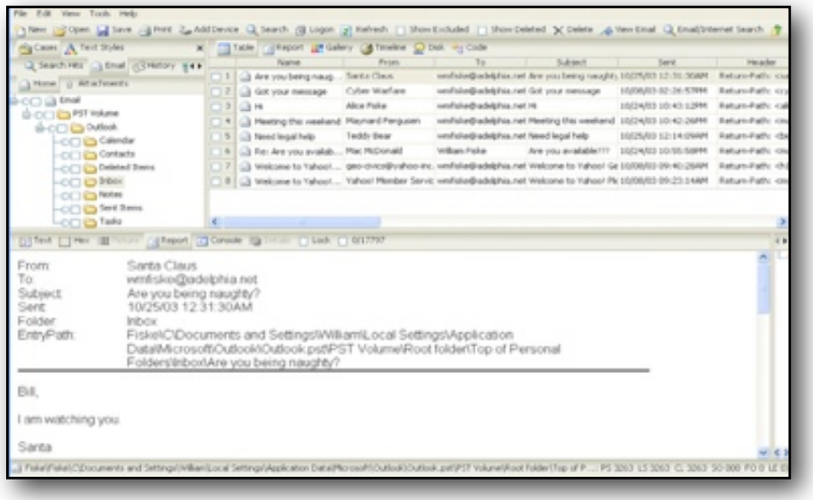
- Systems, Security, and Network Engineering
- Machine Learning
- Natural Language Processing
- Algorithms (compression, decompression, big data)
- High Performance Computing
- Human Computer Interactions

Contact Information:

Simson L. Garfinkel
slgarfin@nps.edu
http://simson.net/

Backup Slides
Building a Corpus

Digital forensics is at a turning point.
Yesterday's work was primarily *reverse engineering*.



Key technical challenges:

- Evidence preservation.
- File recovery (file system support); Undeleting files
- Encryption cracking.
- Keyword search.

Today's work is increasingly *scientific.*

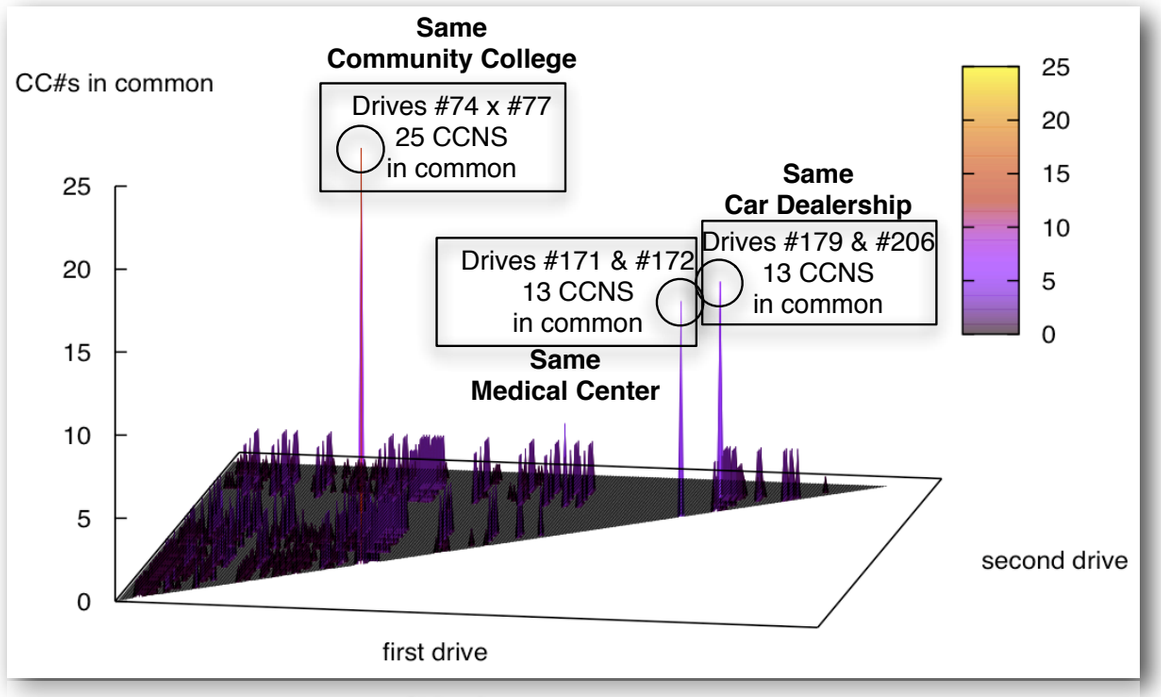
Evidence Reconstruction

- Files (fragment recovery carving)
- Timelines (visualization)

Clustering and data mining

Social network analysis

Sense-making



Science requires the *scientific process.*

Hallmarks of Science:

- Controlled and repeatable experiments.
- No privileged observers.

Why repeat some other scientist's experiment?

- Validate that an algorithm is properly implemented.
- Determine if ***your*** new algorithm is better than ***someone else's*** old one.
- (Scientific confirmation? — perhaps for venture capital firms.)

We can't do this today.

- People work with their own data
  - —*Can't sure because of copyright & privacy issues.*
- People work with "evidence"
  - —*Can't discuss due to legal sensitivities.*

We do science with “real data.”

The Real Data Corpus (30TB)

- Disks, camera cards, & cell phones purchased on the secondary market.
- Most contain data from previous users.
- Mostly acquire outside the US:
  - —*Canada, China, England, Germany, France, India, Israel, Japan, Pakistan, Palestine, etc.*
- Thousands of devices (HDs, CDs, DVDs, flash, etc.)

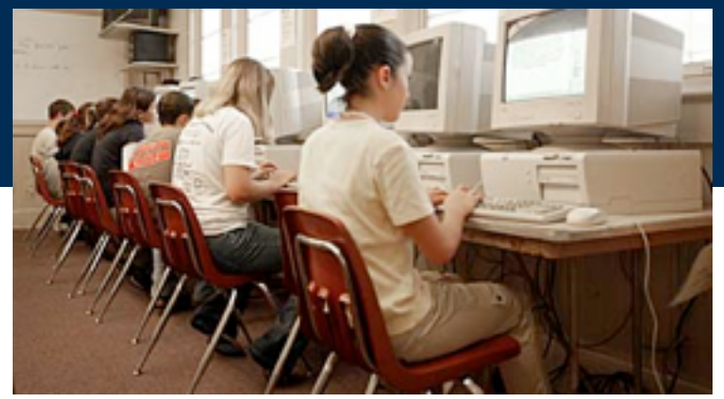
Mobile Phone Application Corpus

- Android Applications; Mobile Malware; etc.

The problems we encounter obtaining, curating and exploiting this data mirror those of national organizations

- —*Garfinkel, Farrell, Roussev and Dinolt, Bringing Science to Digital Forensics with Standardized Forensic Corpora, DFRWS 2009*
  http://digitalcorpora.org/

Digital Forensics education needs fake data!



To teach forensics, we need complex data!

- Disk images
- Memory images
- Network packets

Some teachers get used hard drives from eBay.

- Problem: you don't know what's on the disk.
  - *Ground Truth.*
  - *Potential for illegal Material — distributing porn to minors is illegal.*

Some teachers have students examine other student machines:

- Self-examination: students know what they will find
- Examining each other's machines: potential for inappropriate disclosure

We manufacture data that can be freely redistributed.

Files from US Government Web Servers (500GB)

- ≈1 million heterogeneous files
  - *Documents (Word, Excel, PDF, etc.); Images (JPEG, PNG, etc.)*
  - *Database Files; HTML files; Log files; XML*
- Freely redistributable; Many different file types
- This database was surprising difficulty to collect, curate, and distribute:
  - *Scale created data collection and management problems.*
  - *Copyright, Privacy & Provenance issues.*

Advantage over flickr & youtube: persistence & copyright




<abstract>NOAA's National Geophysical Data Center (NGDC) is building high-resolution digital elevation models (DEMs) for select U.S. coastal regions. … </abstract>

<abstract>This data set contains data for birds caught with mistnets and with other means for sampling Avian Influenza (AI)….</abstract>