



# Deep Learning is great

Ted Dunning





Deep Learning is great  
But don't forget cheap learning!

Ted Dunning



# Me, Us

- Ted Dunning, MapR Chief Application Architect, Apache Member
  - Committer PMC member Zookeeper, Drill, others
  - Mentor for Flink, Beam (nee Dataflow), Drill, Storm, Zeppelin
  - VP Incubator
  - Bought the beer at the first HUG
- MapR
  - Produces first converged platform for big and fast data
  - Includes data platform (files, streams, tables) + open source
  - Adds major technology for performance, HA, industry standard API's
- Contact

[@ted\\_dunning](mailto:@ted_dunning), [ted.dunning@gmail.com](mailto:ted.dunning@gmail.com), [tdunning@mapr.com](mailto:tdunning@mapr.com)



# Agenda

- Rationale
- Why cheap isn't the same as simple-minded
- Some techniques
- Examples



# Outline

- We have a revolution on our hands
- This leads to a green-field situation
- That implies that many important problems are easy to solve
- The limiting factor is fielding good enough solutions
  - Quickly
  - With available workforce
- Examples

*Is this *really* a  
revolutionary moment?*



# Big is the next big thing

- Data scale is exploding
- Companies are being funded
- Books are being written
- Applications sprouting up everywhere



# Why Now?

- But Moore's law has applied for a long time
- Why is data exploding now?
- Why not 10 years ago?
- Why not 20?



# Size Matters, but ...

- If it were just availability of data then existing big companies would adopt big data technology first



# Size Matters, but ...

- If it were just availability of data then existing big companies would adopt big data technology first

They didn't



# Or Maybe Cost

- If it were just a net positive value then finance companies should adopt first because they have higher opportunity value / byte



# Or Maybe Cost

- If it were just a net positive value then finance companies should adopt first because they have higher opportunity value / byte

They didn't



# Backwards adoption

- Under almost any threshold argument startups would not adopt big data technology first



# Backwards adoption

- Under almost any threshold argument startups would not adopt big data technology first

*They did*



# Everywhere at Once?

- Something very strange is happening
  - Big data is being applied at many different scales
  - At many value scales
  - By large companies and small



# Everywhere at Once?

- Something very strange is happening
  - Big data is being applied at many different scales
  - At many value scales
  - By large companies and small

*Why?*

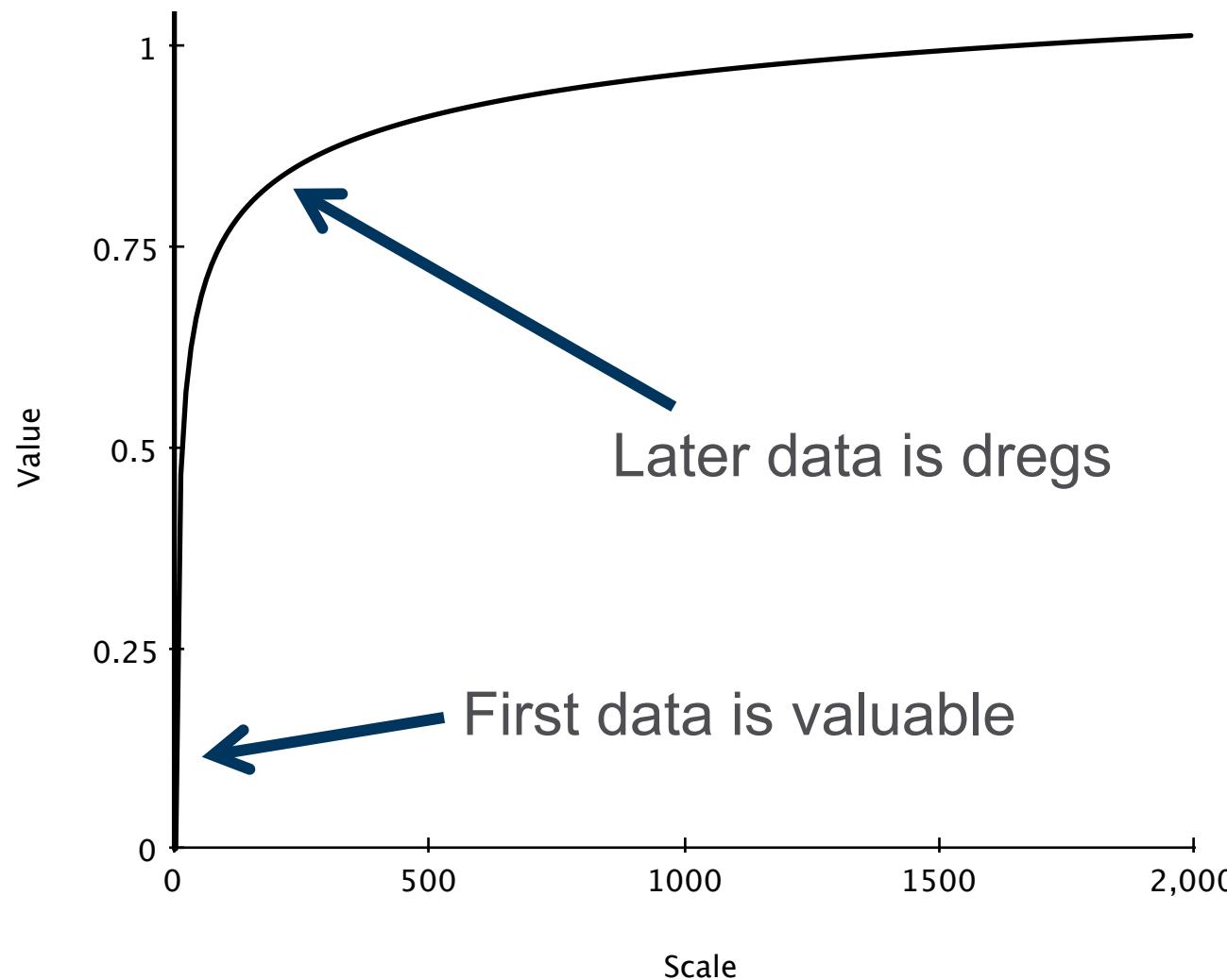


# Analytics Scaling Laws

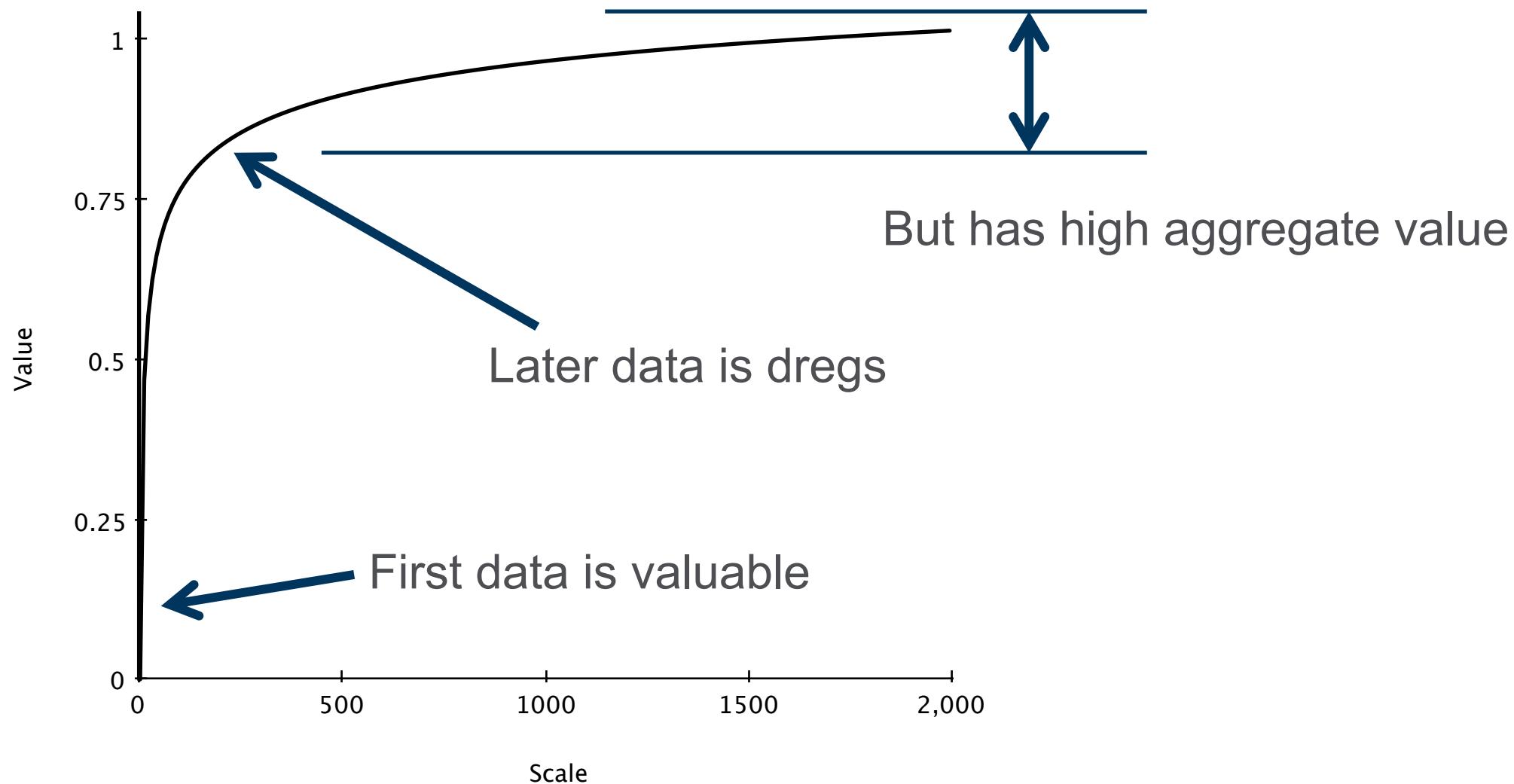
- Analytics scaling is all about the 80-20 rule
  - Big gains for little initial effort
  - Rapidly diminishing returns
- The key to net value is how costs scale
  - Old school – exponential scaling
  - Big data – linear scaling, low constant
- Cost/performance has changed radically
  - IF you can use many commodity boxes



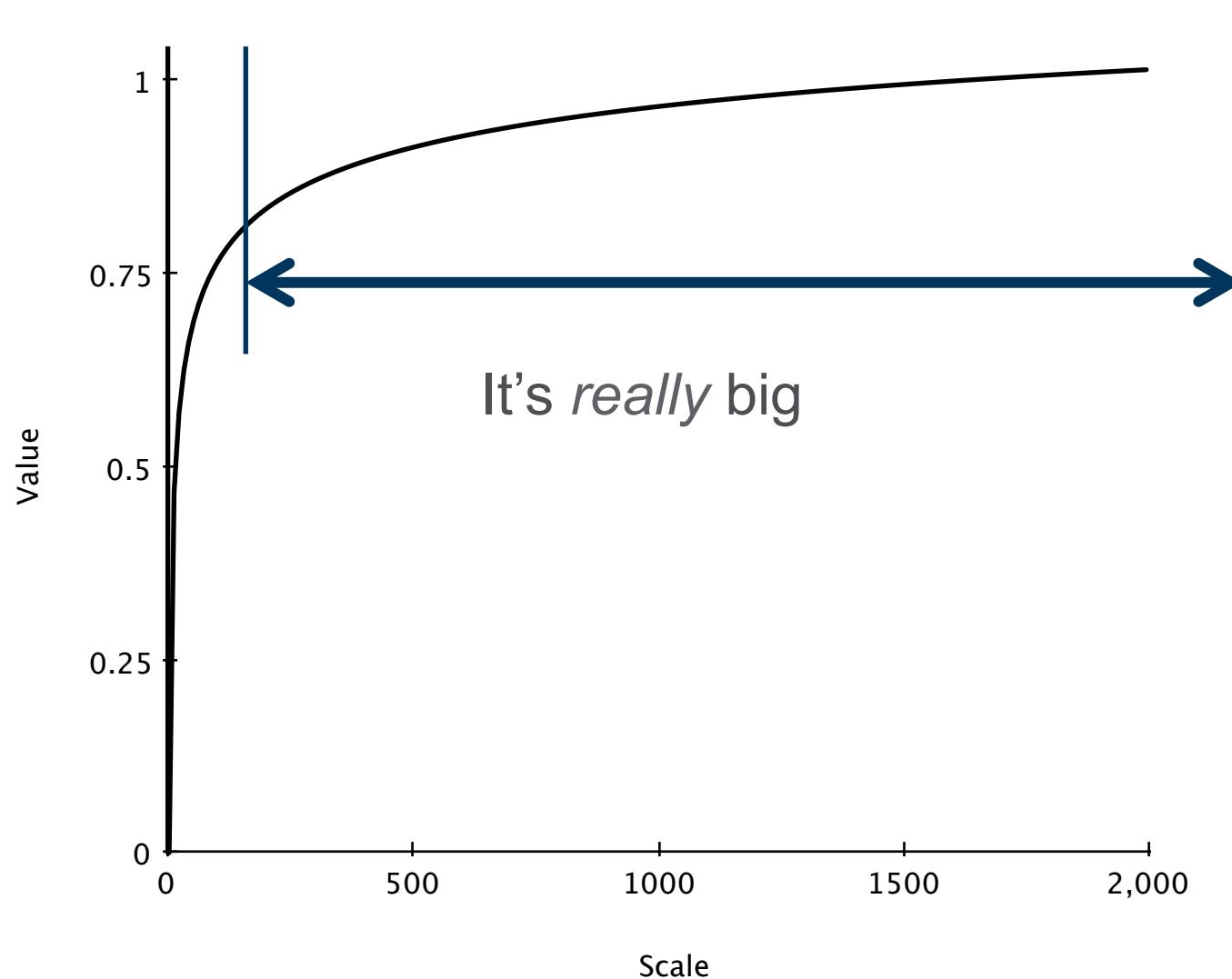
# Most data isn't worth much in isolation



# Suddenly worth processing

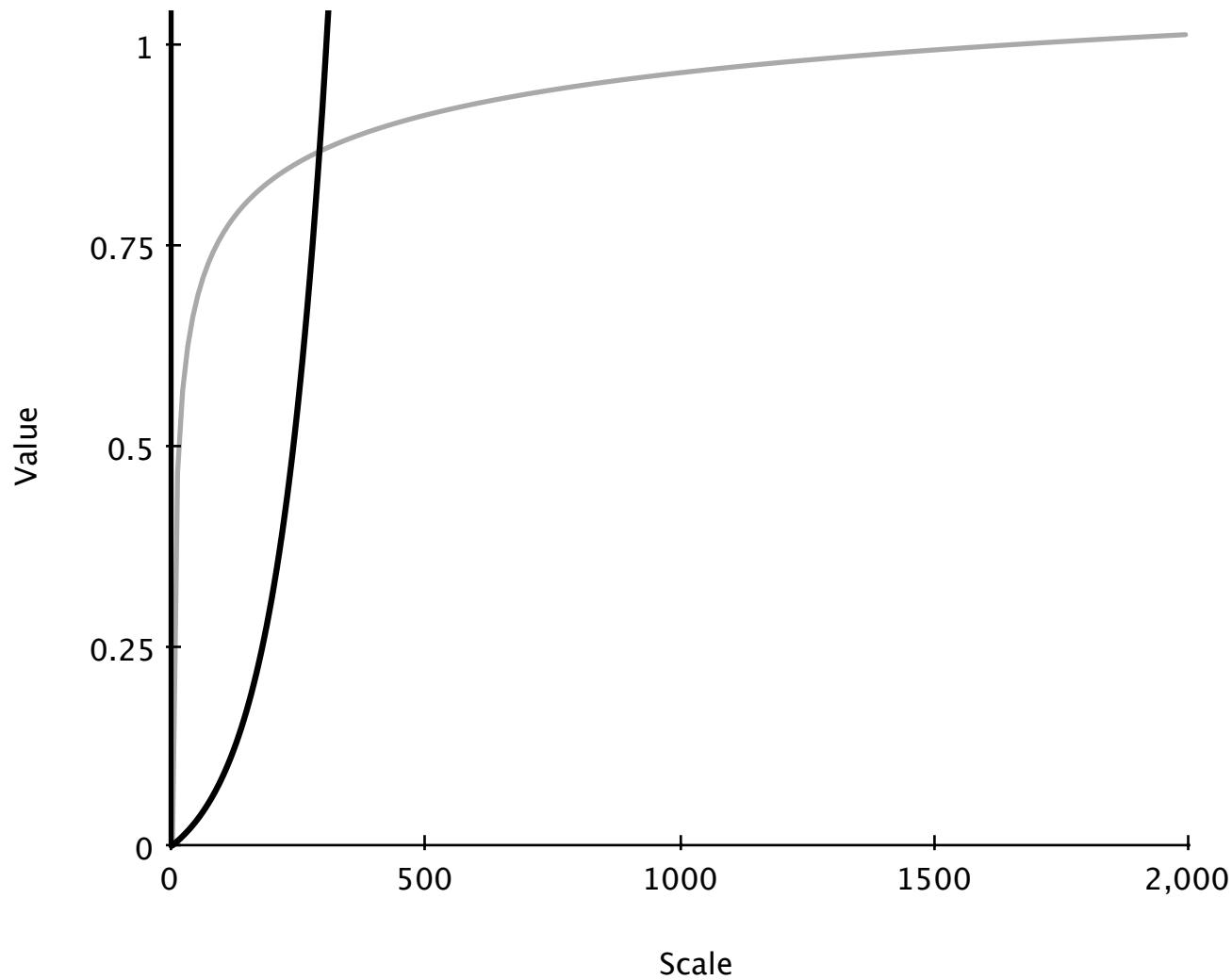


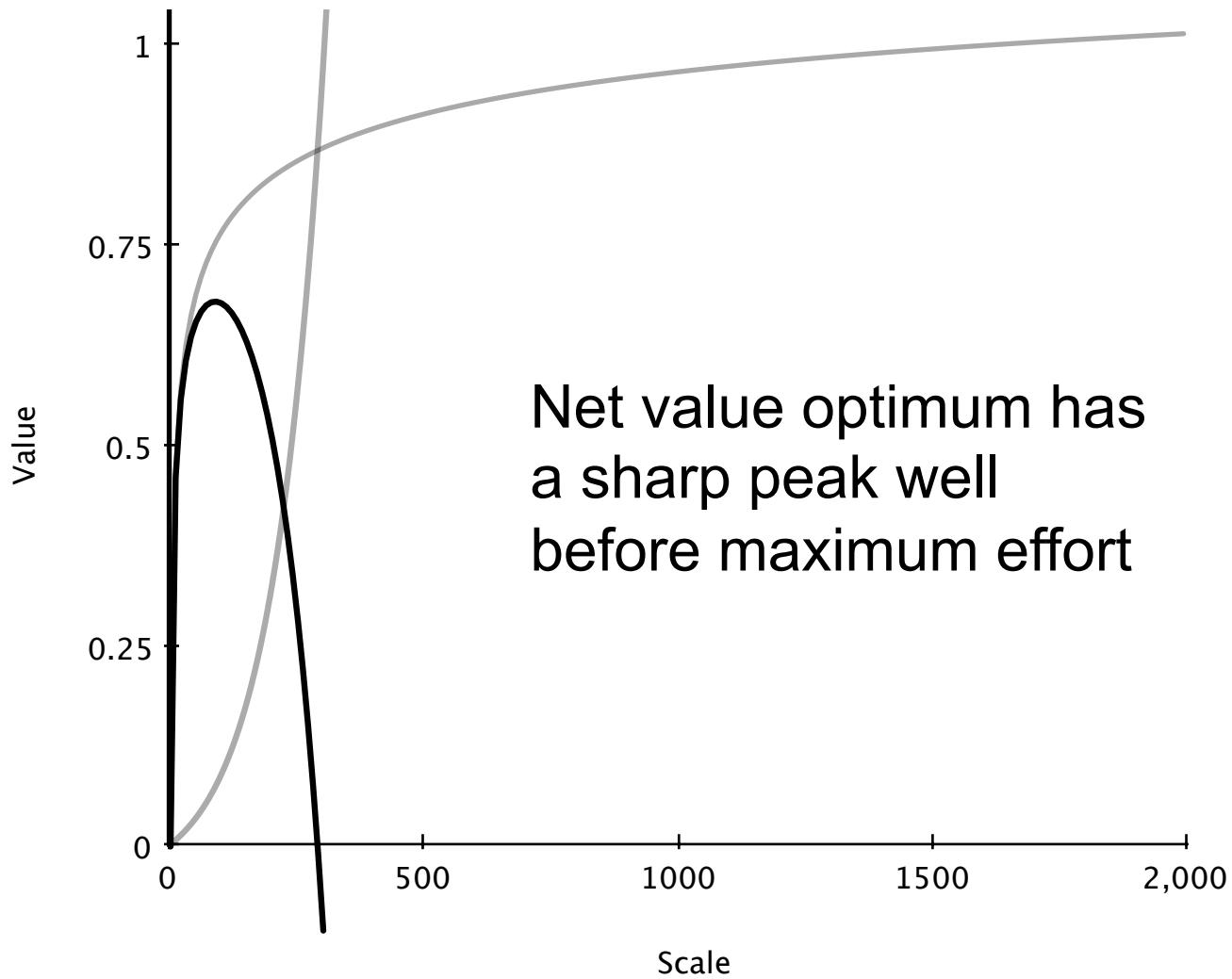
# If we can handle the scale

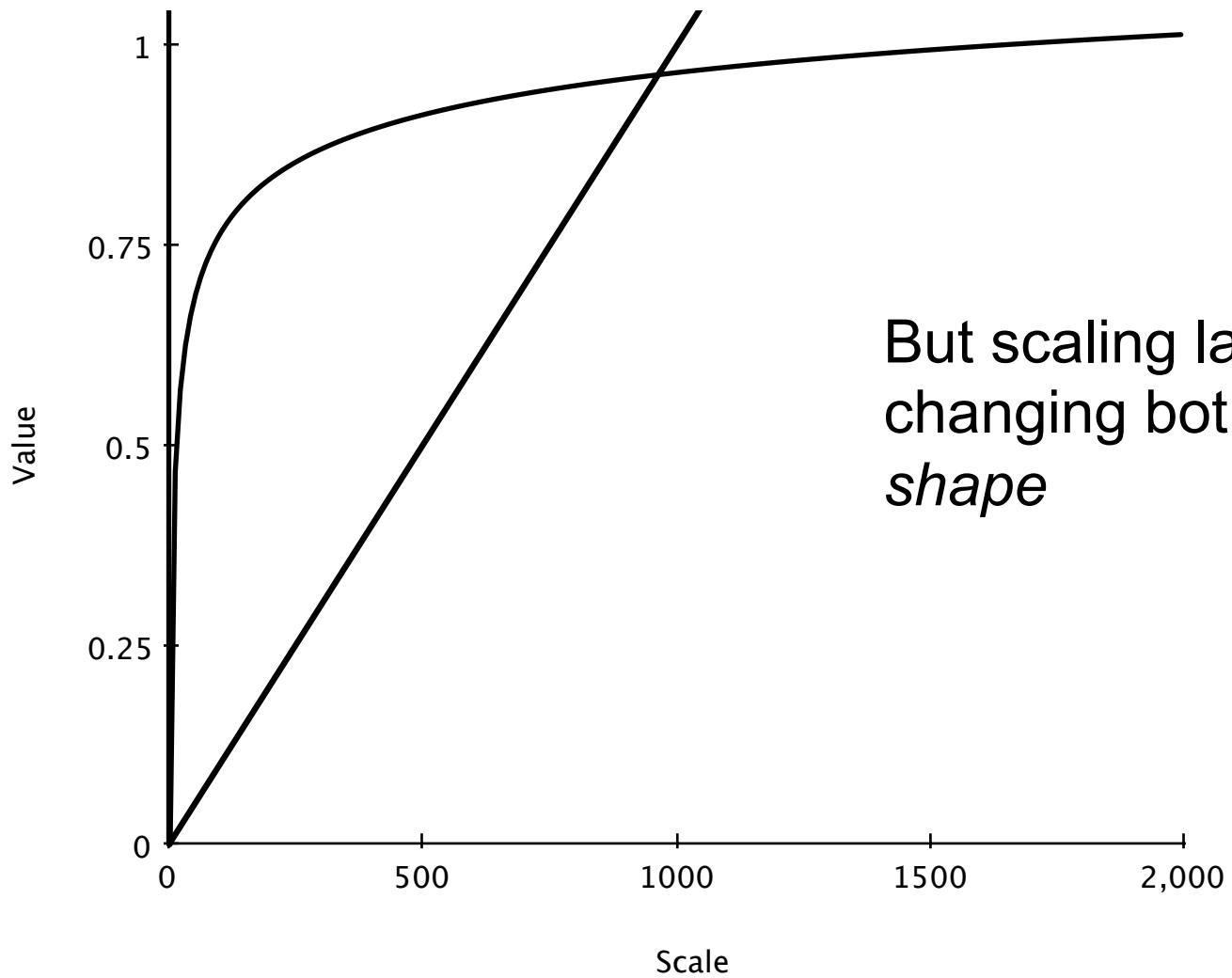


So what makes  
that possible?

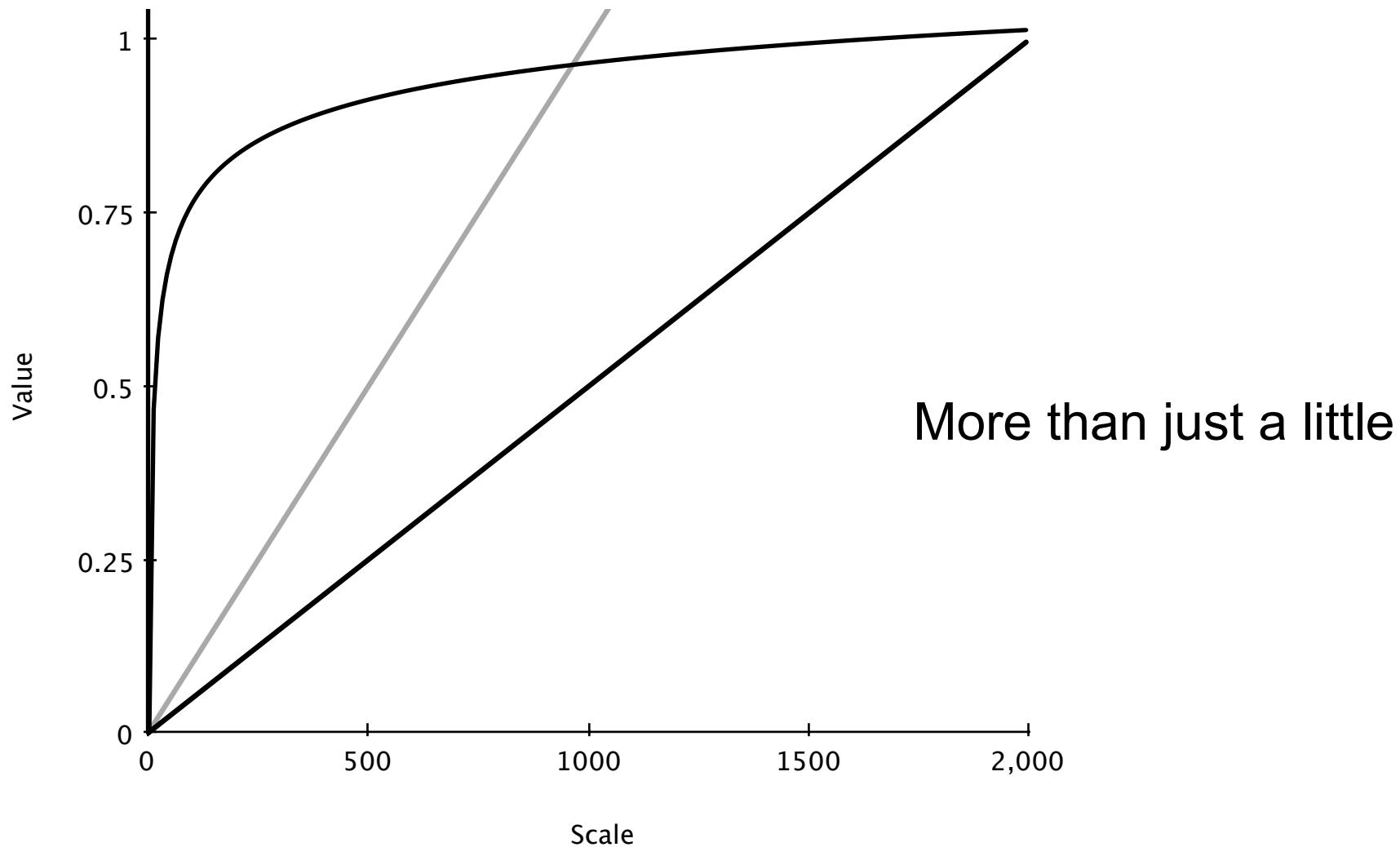


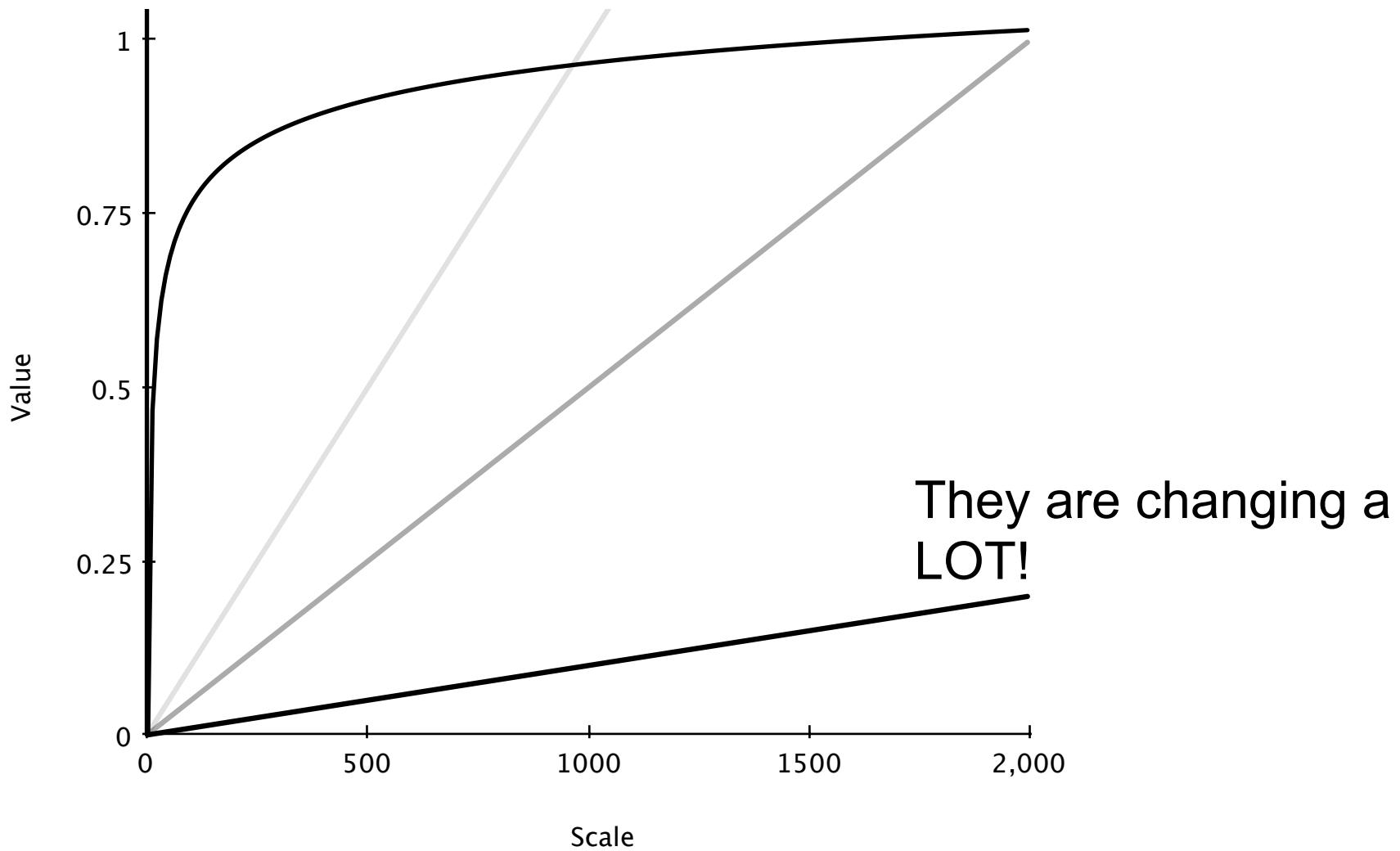


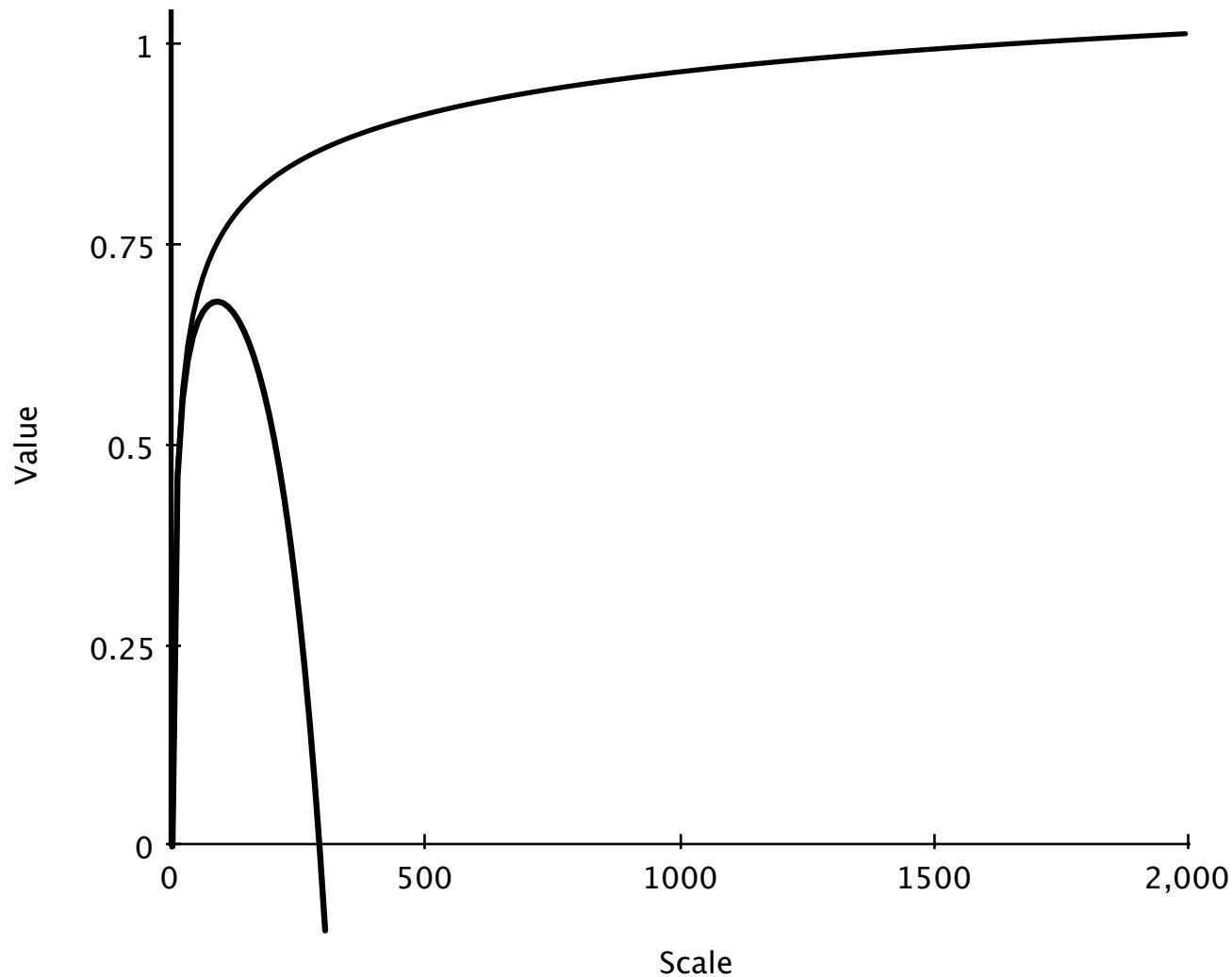


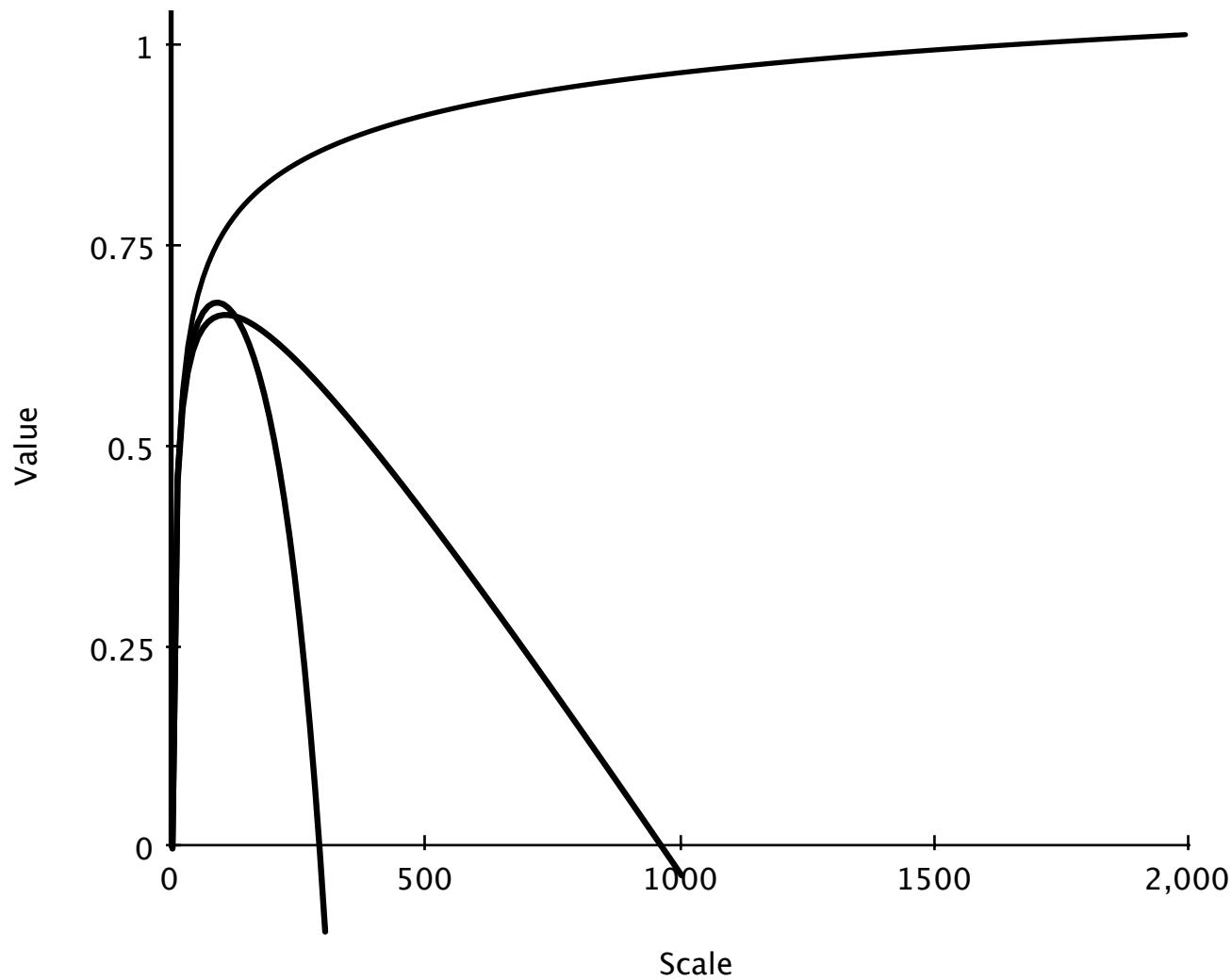


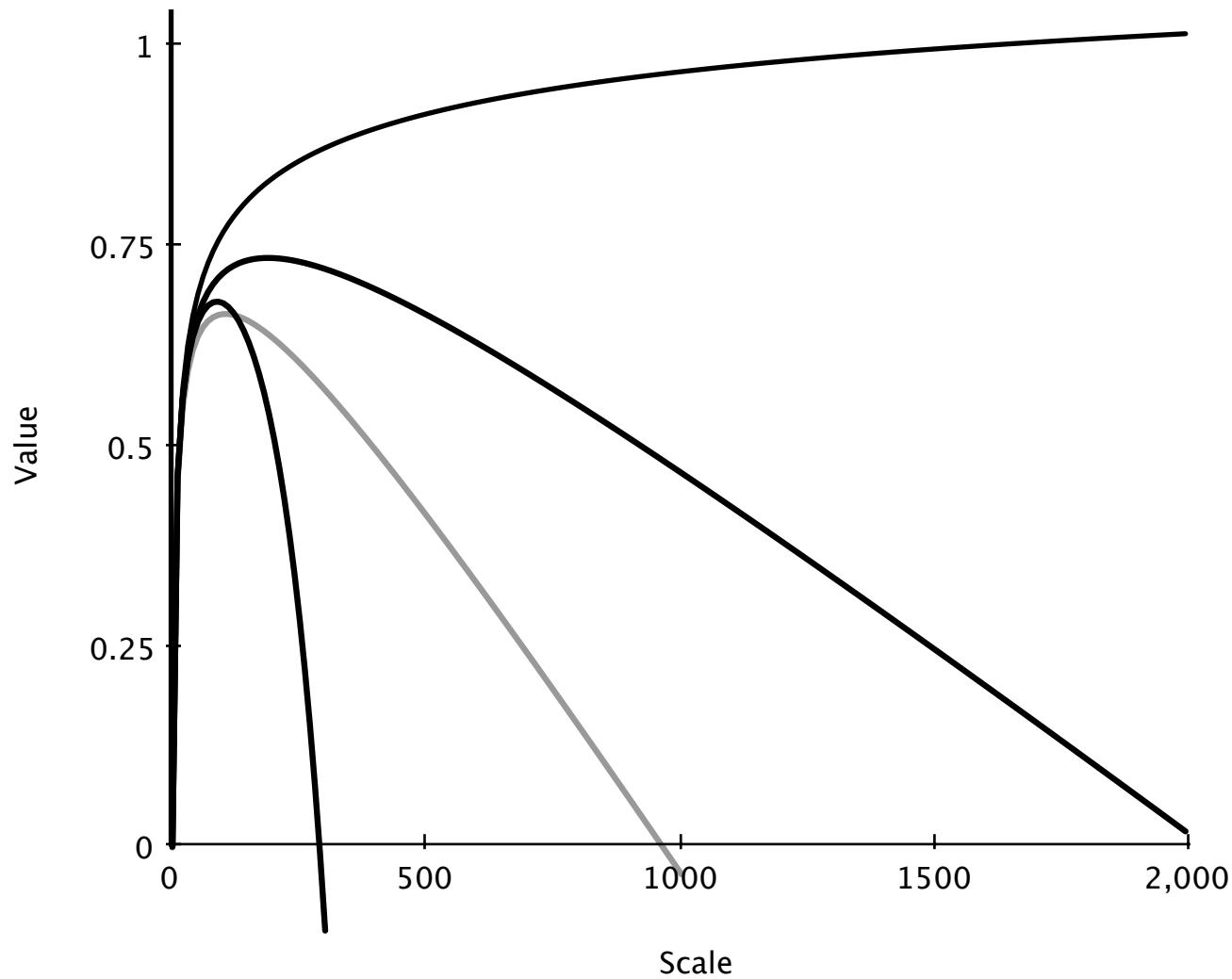
But scaling laws are  
changing both *slope* and  
*shape*

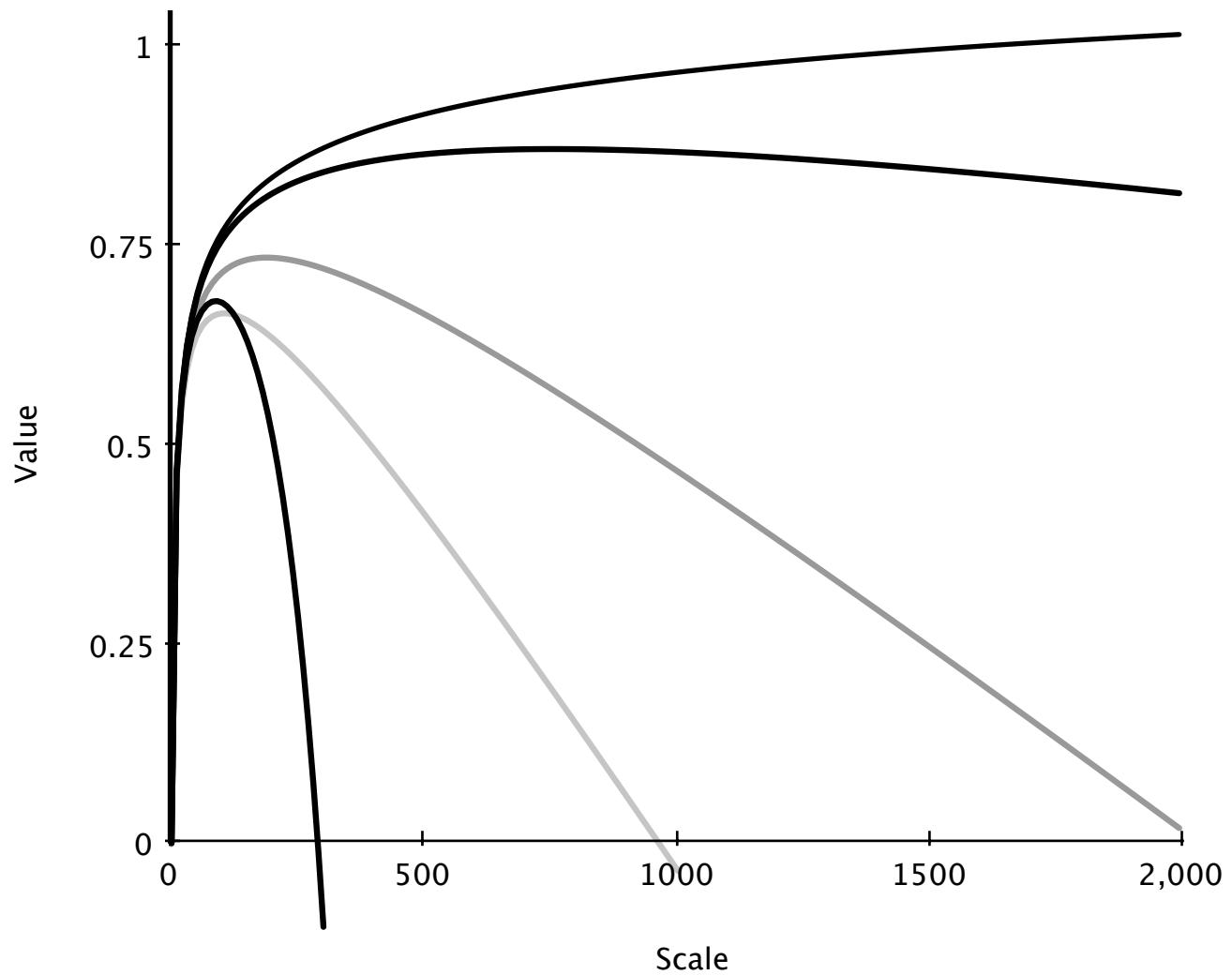


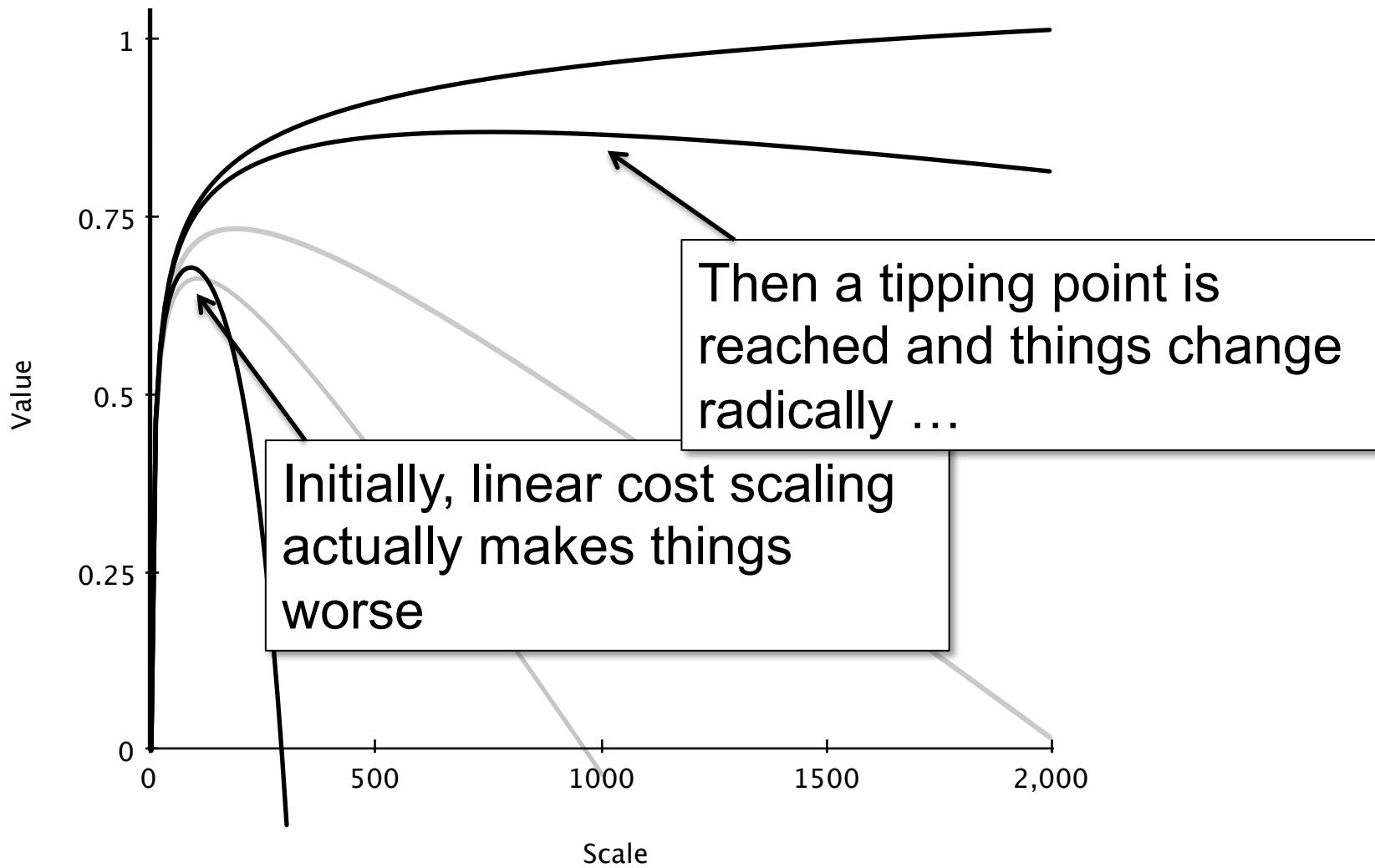












# Pre-requisites for Tipping

- To reach the tipping point,
- Algorithms must scale out horizontally
  - On commodity hardware
  - That can and will fail
- Data practice must change
  - Denormalized is the new black
  - Flexible data dictionaries are the rule
  - Structured data becomes rare



# With great scale comes great opportunity

- Increasing scale by 1000x changes the game
- We essentially have green fields opening up all around
- Most of the opportunities don't require advanced learning

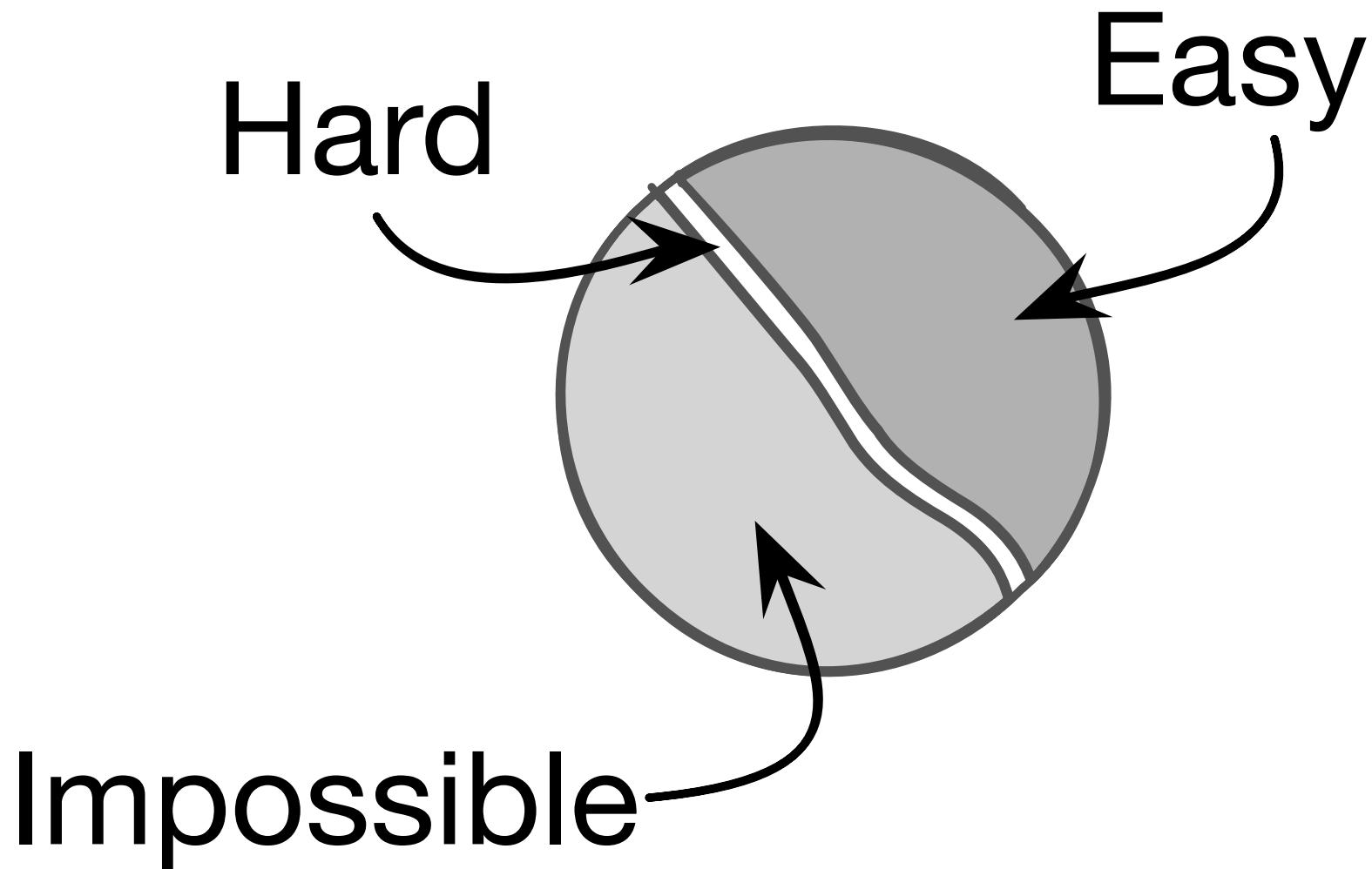


OK.

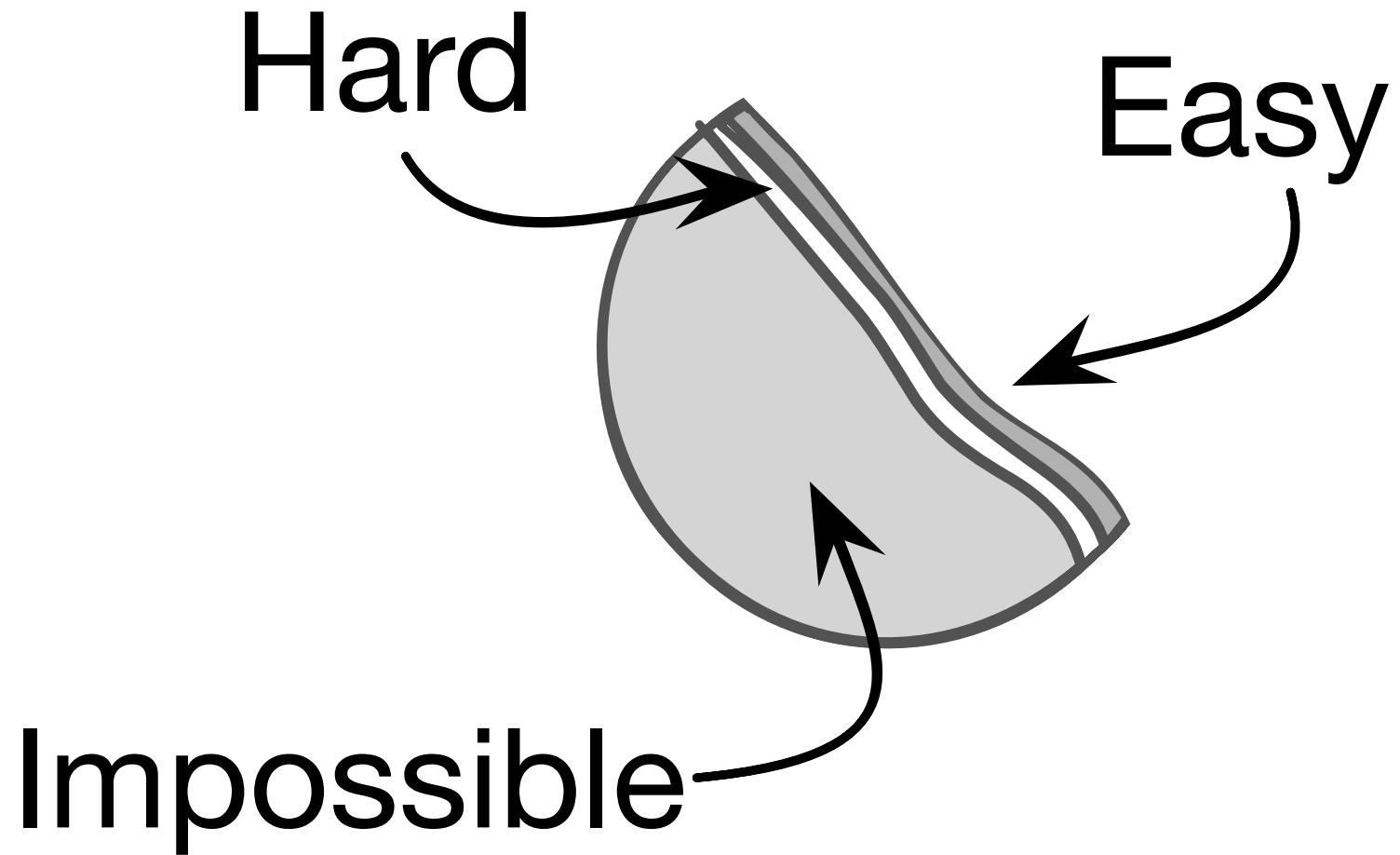
We have a  
bona fide revolution



# Greenfield Problem Landscape



# Mature Problem Landscape



# Why is cheap better than deep (sometimes)?

When we have a greenfield, problems can be

- Easy (large number of these)
- Impossible (large number of these)
- Hard but possible (right on the boundary)

In a mature field, problems can be

- Easy (these are already done)
- Impossible (still a large number of these)
- Hard but possible (now the majority of the effort)

# Some examples



# A simple example - security monitoring

- “Small” data
  - Capture IDS logs
  - Detect what you already know
- “Big” data
  - Capture switch, server, firewall logs as well
  - New patterns emerge immediately



# Another example – fraud detection

- “Small” data
  - Maintain card profiles
  - Segment models
  - Evaluate all transactions
- “Big” Data
  - Maintain card profiles, full 90 day transaction history
  - Evaluate all transactions

# Another example – indicator-based recommendation

- “Advanced” approach
  - Use matrix completion techniques (LDA, NNM, ALS)
  - Tune meta-parameters
  - Ensembles galore
- “Simple” approach
  - Count cooccurrences and cross-occurrences
  - Finding “interesting” pairs
  - Use standard search engine to recommend

# Easy != Stupid

- You still have to do things reasonably well
  - Techniques that are not well founded are still problems
- Heuristic frequency ratios still fail
  - Coincidences still dominate the data
  - Accidental 100% correlations abound
- Related techniques still broken for coincidence
  - Pearson's  $\chi^2$
  - Simple correlations

Scale does not cure wrong

It just makes easy more common



# A core technique

- Many of these easy problems reduce to finding interesting coincidences
- This can be summarized as a  $2 \times 2$  table

	A	Other
B	$k_{11}$	$k_{12}$
Other	$k_{21}$	$k_{22}$

- Actually, many of these tables

# How do you do that?

- This is well handled using G<sup>2</sup>-test
  - See wikipedia
  - See <http://bit.ly/surprise-and-coincidence>
- Original application in linguistics now cited > 2000 times
- Available in ElasticSearch, in Solr, in Mahout
- Available in R, C, Java, Python

# Which one is the anomalous co-occurrence?

	A	<i>not A</i>
B	13	1000
<i>not B</i>	1000	100,000

	A	<i>not A</i>
B	1	0
<i>not B</i>	0	2

	A	<i>not A</i>
B	1	0
<i>not B</i>	0	10,000

	A	<i>not A</i>
B	10	0
<i>not B</i>	0	100,000

# Which one is the anomalous co-occurrence?

	A	<i>not A</i>
0.90	13	1000
<i>not B</i>	1000	100,000

	A	<i>not A</i>
1.95	1	0
<i>not B</i>	0	2

	A	<i>not A</i>
4.52	1	0
<i>not B</i>	0	10,000

	A	<i>not A</i>
14.3	10	0
<i>not B</i>	0	100,000

Dunning Ted, Accurate Methods for the Statistics of Surprise and Coincidence,  
*Computational Linguistics* vol 19 no. 1 (1993)

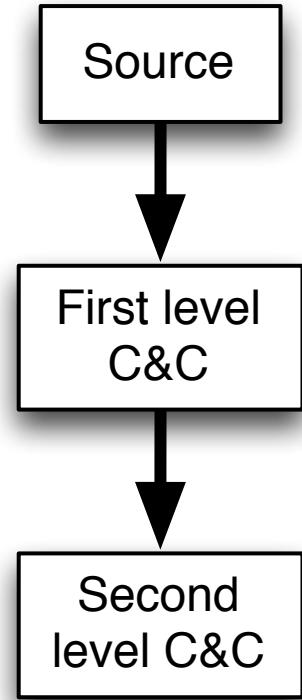
So we can find  
interesting coincidences.

That gets us exactly what?

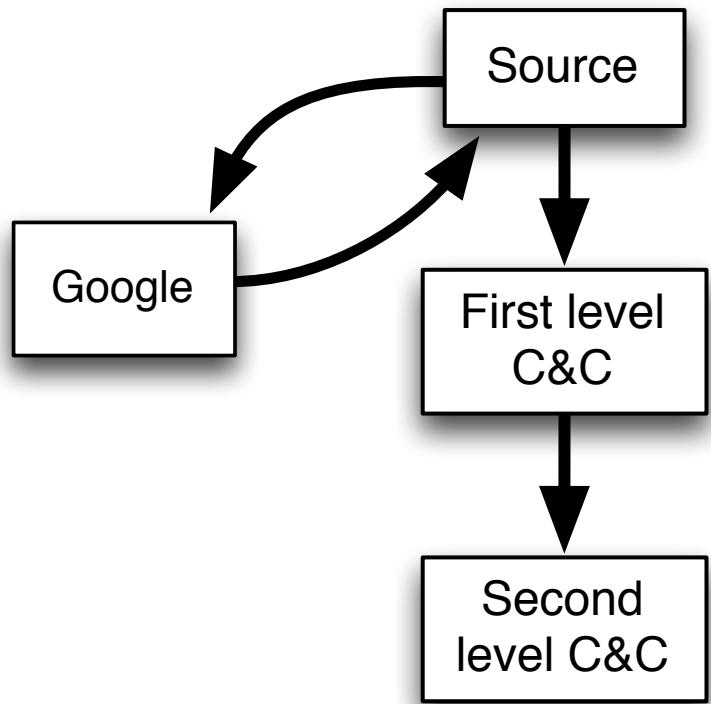
# Operation Ababil – Brobots on Parade

- Dork attack to find unpatched default Joomla sites
  - Especially web servers with high bandwidth connections
  - Basically just Google searches for default strings
  - Joomla compromised into attack Brobot
- C&C network checks in occasionally
  - Note C&C is incoming request and looks like normal web requests
- Later, on command, multiple Brobots direct 50-75 Gb/s of attack
  - Attacks come from white-listed sites

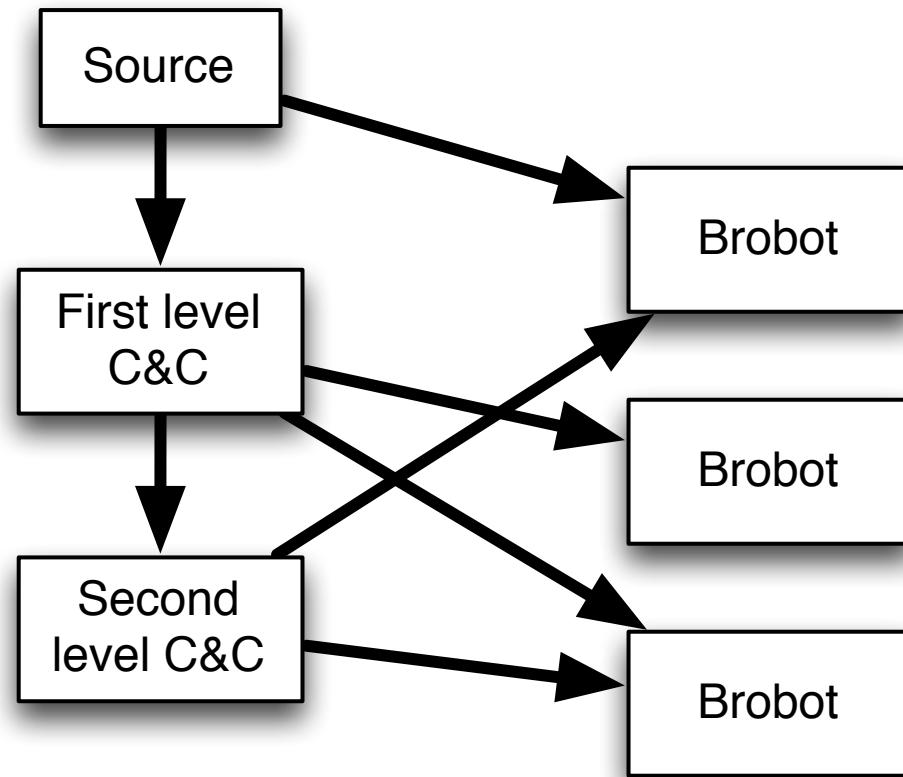
# Attack Sequence



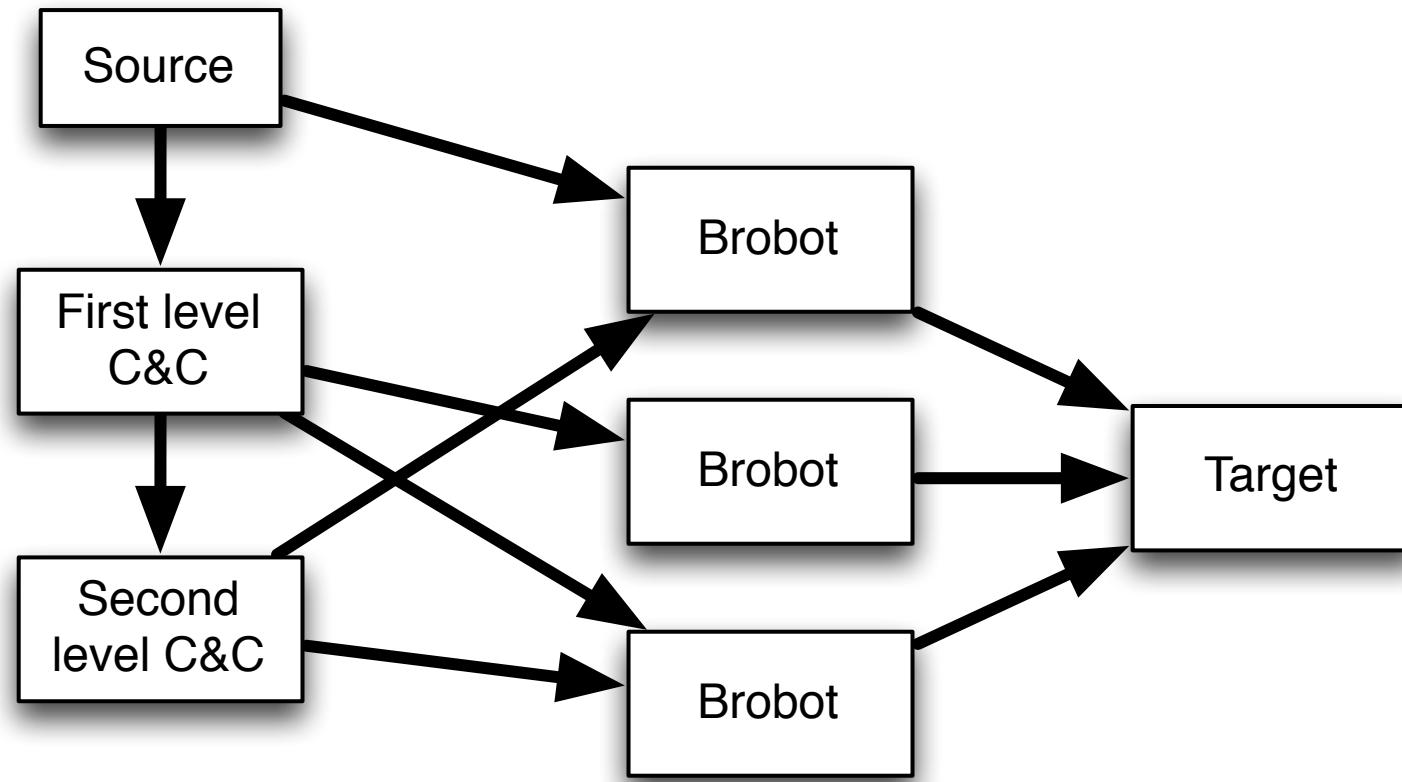
# Attack Sequence



# Attack Sequence



# Attack Sequence



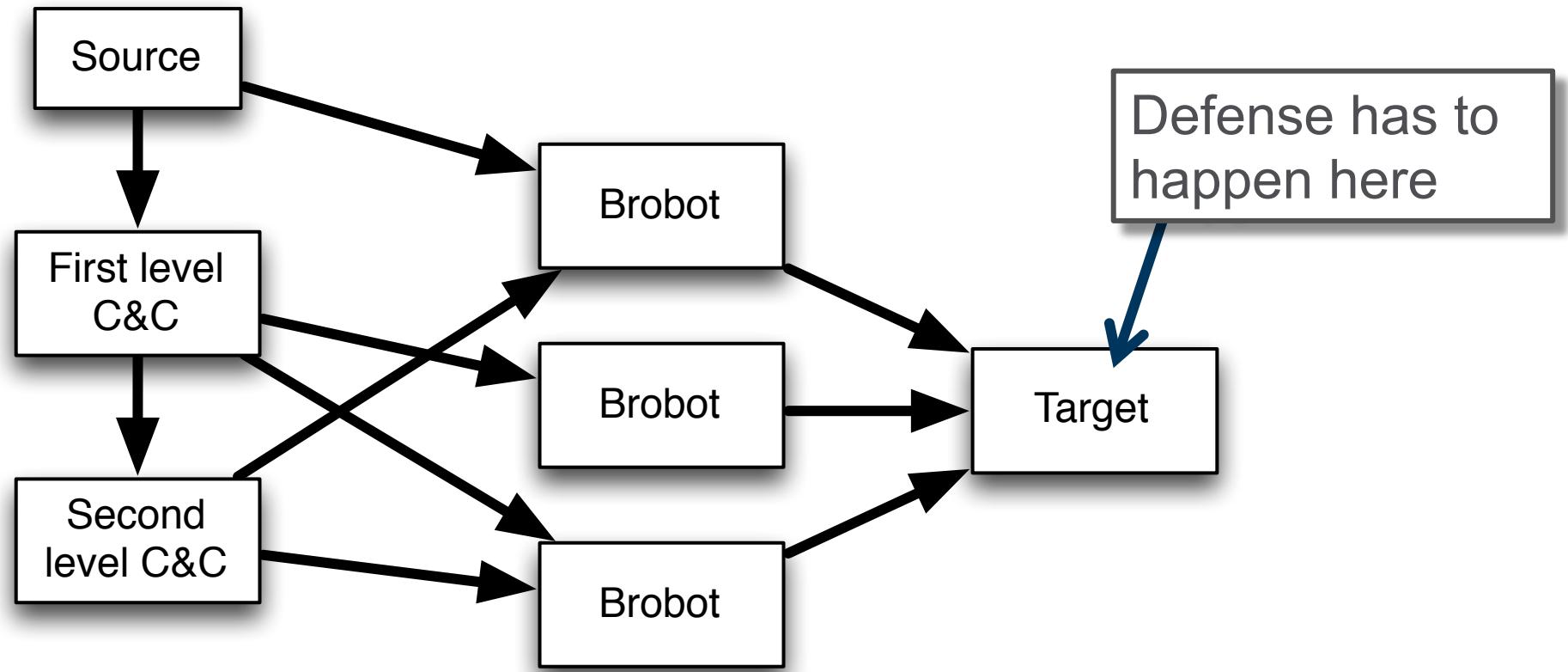
# Outline of an Advanced Persistent Threat

- Advanced
  - Common use of zero-day for preliminary attacks
  - Often attributed to state-level actors
  - Modern privateers blur the line
- Persistent
  - Result of first attack is heavily muffled, no immediate exploit
  - Remote access toolset installed (RAT)
- Threat
  - On command, data is exfiltrated covertly or *en masse*
  - Or the compromised host is used for other nefarious purpose

# APT in Summary

- *Attack, penetrate, pivot, exfiltrate or exploit*
- If you are a high-value target, attack is likely and stealthy
  - High-value = telecom, banks, utilities, retail targets, web100
  - ... and all their vendors
  - Conventional multi-factor auth is easily breached
- Penetration and pivot are critical counter-measure opportunities
  - In 2010, RAT would contact command and control (C&C)
  - In 2016, C&C looks like normal traffic
- Once exfiltration or exploit starts, you may no longer have a business

# Example 1 - Ababil



# Spot the Important Difference?

```
GET /personal/comparison-table  
Host: www.sometarget.com  
User-Agent: Mozilla/4.0 (compa  
Accept-Encoding: deflate  
Accept-Charset: UTF-8  
Accept-Language: fr  
Cache-Control: no-cache  
Pragma: no-cache  
Connection: Keep-Alive
```

Attacker request

```
GET /photo.jpg HTTP/1.1  
Host: lh4.googleusercontent.  
User-Agent: Mozilla/5.0 (Mac  
Accept: image/png,image/*;q=  
Accept-Language: en-US,en;q=  
Accept-Encoding: gzip, defla  
Referer: https://www.google.  
Connection: keep-alive  
If-None-Match: "v9"  
Cache-Control: max-age=0
```

Real request

# Spot the Important Difference?

```
GET /personal/comparison-table  
Host: www.sometarget.com  
User-Agent: Mozilla/4.0 (compa  
Accept-Encoding: deflate  
Accept-Charset: UTF-8  
Accept-Language: fr  
Cache-Control: no-cache  
Pragma: no-cache  
Connection: Keep-Alive
```

Attacker request

```
GET /photo.jpg HTTP/1.1  
Host: lh4.googleusercontent.  
User-Agent: Mozilla/5.0 (Mac  
Accept: image/png,image/*;q=  
Accept-Language: en-US,en;q=  
Accept-Encoding: gzip, deflate  
Referer: https://www.google.  
Connection: keep-alive  
If-None-Match: "v9"  
Cache-Control: max-age=0
```

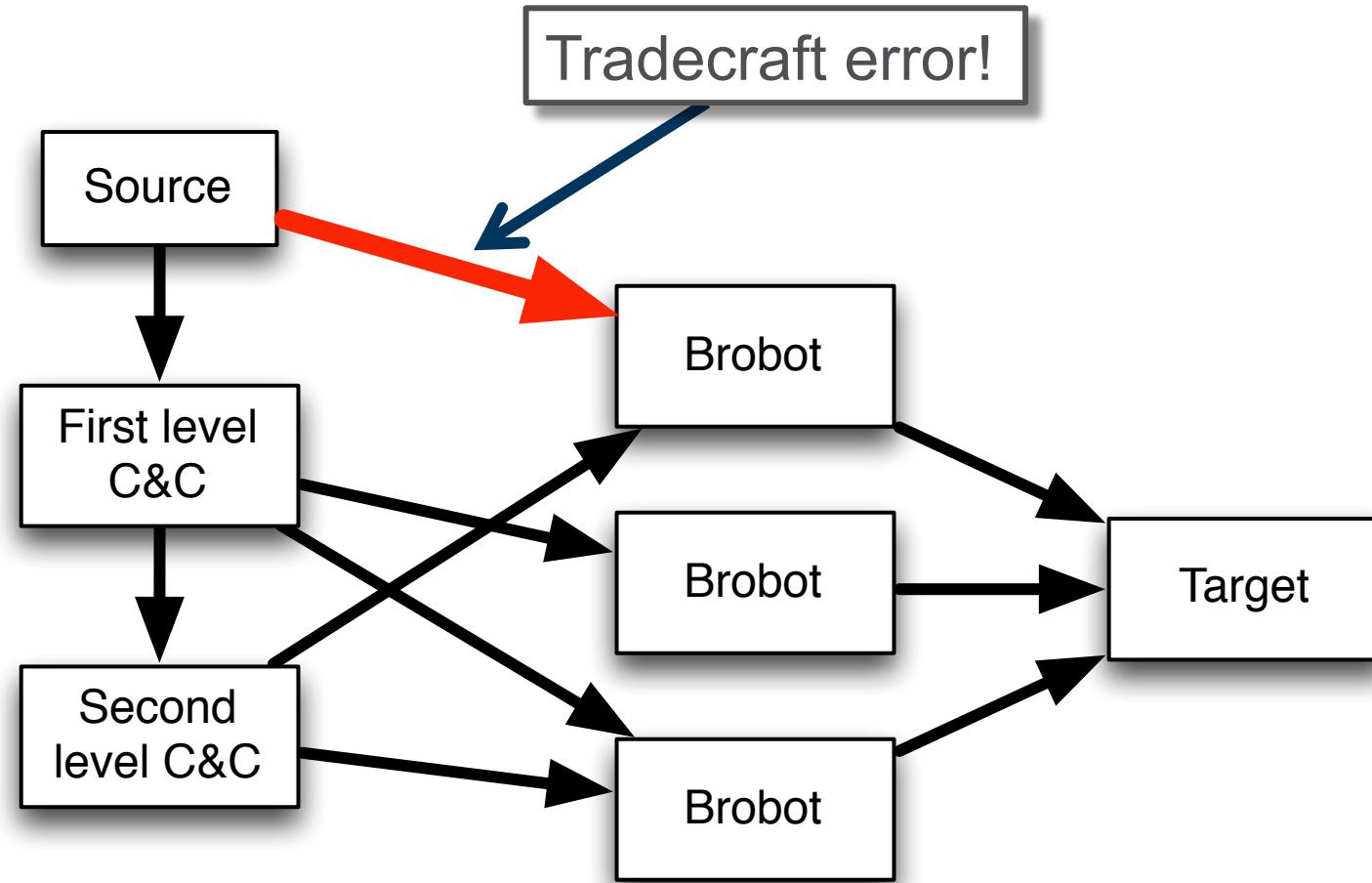
Real request

This could only be found at scale

This could only be found at scale

But at scale, it is stupidly simple  
to find

# Overall Outline Again



Large corpus analysis of source  
IP's wins big

# WANTED

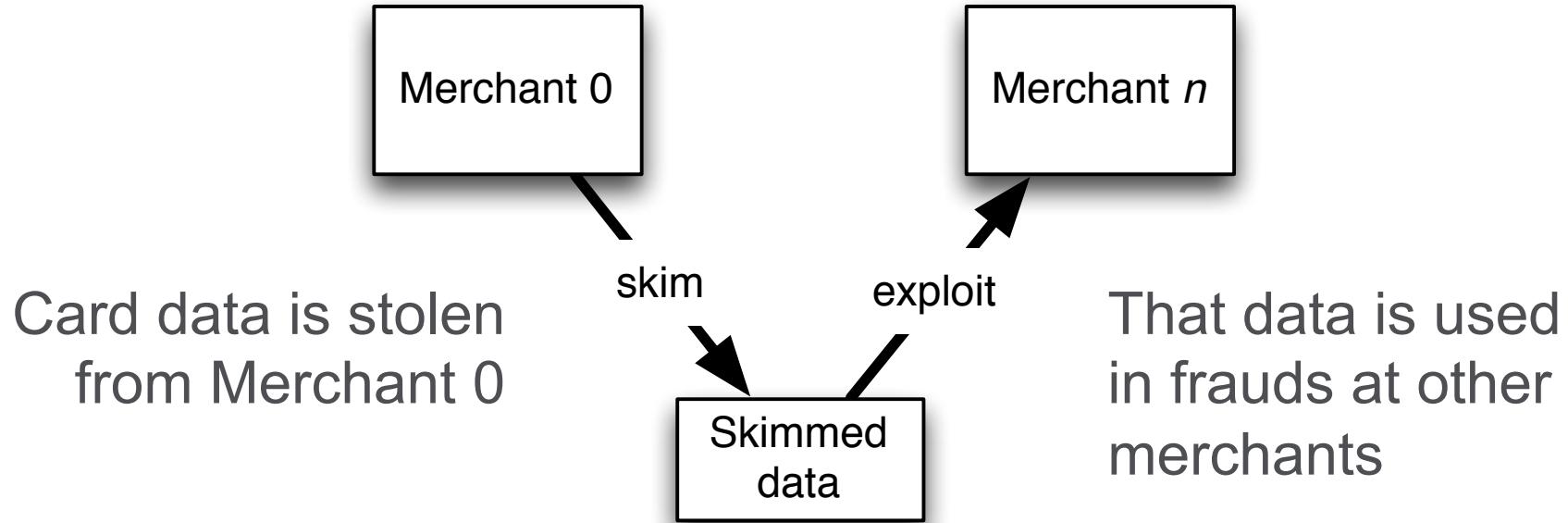
BY THE FBI



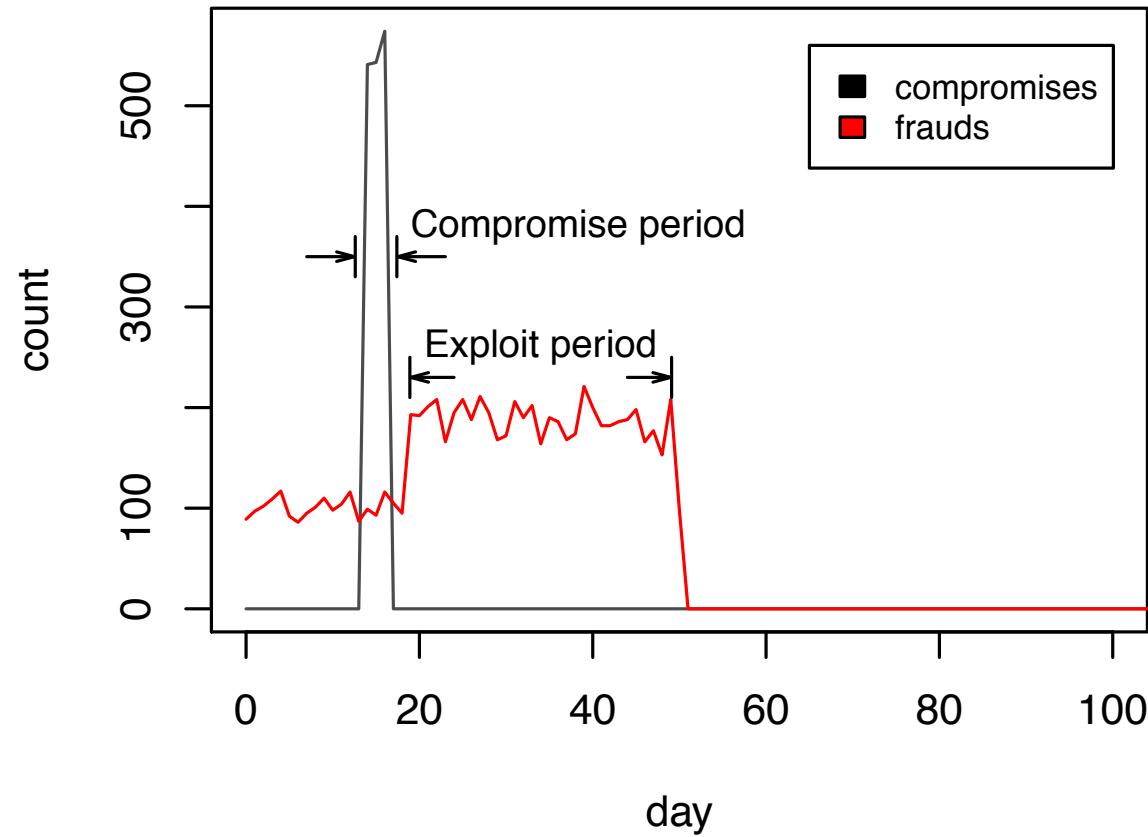
# Example 2 - Common Point of Compromise

- Scenario:
  - Merchant 0 is compromised, leaks account data during compromise
  - Fraud committed elsewhere during exploit
  - High background level of fraud
  - Limited detection rate for exploits
- Goal:
  - Find merchant 0
- Meta-goal:
  - Screen algorithms for this task without leaking sensitive data

## Example 2 - Common Point of Compromise



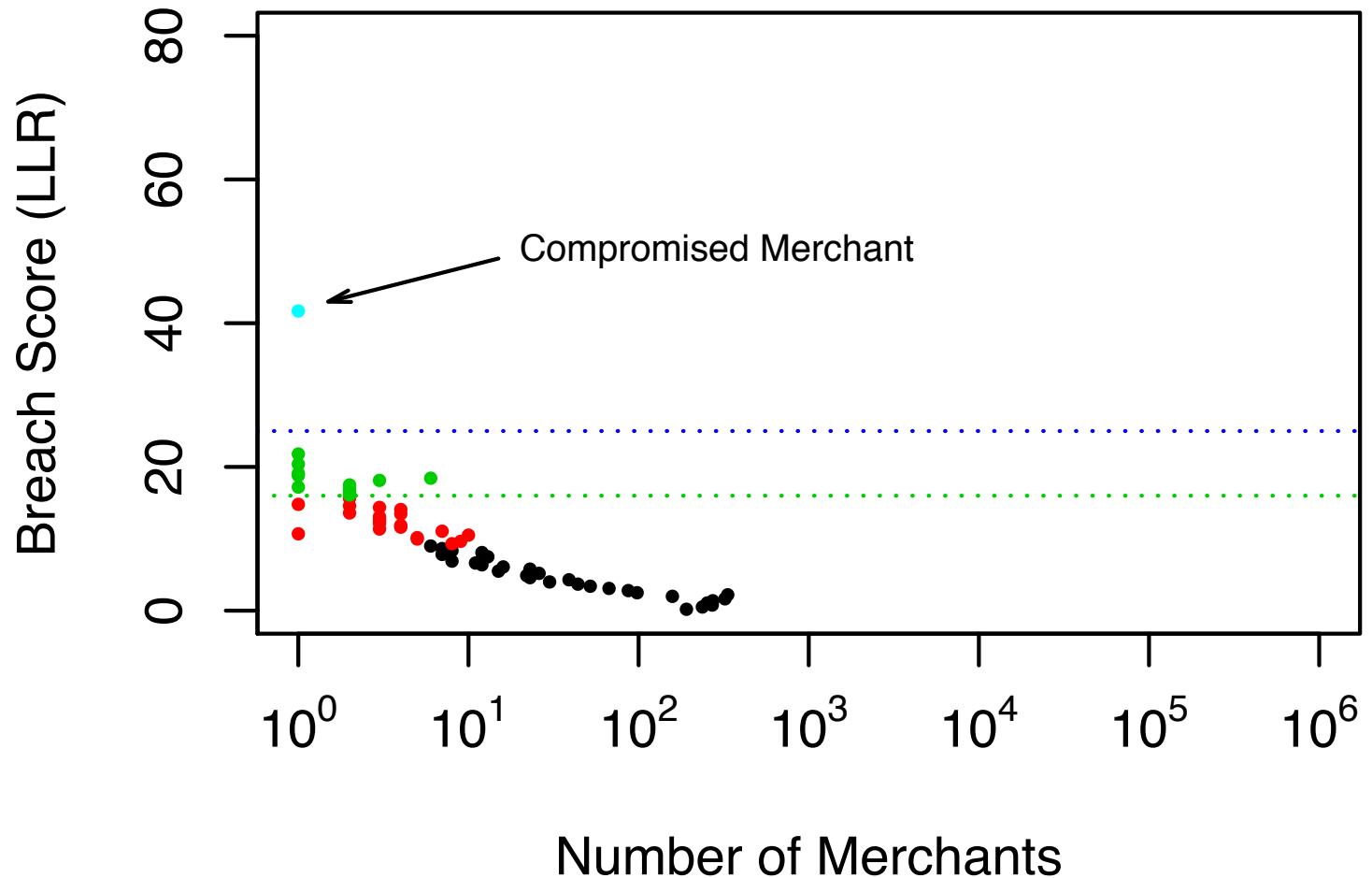
# Simulation Setup



# Detection Strategy

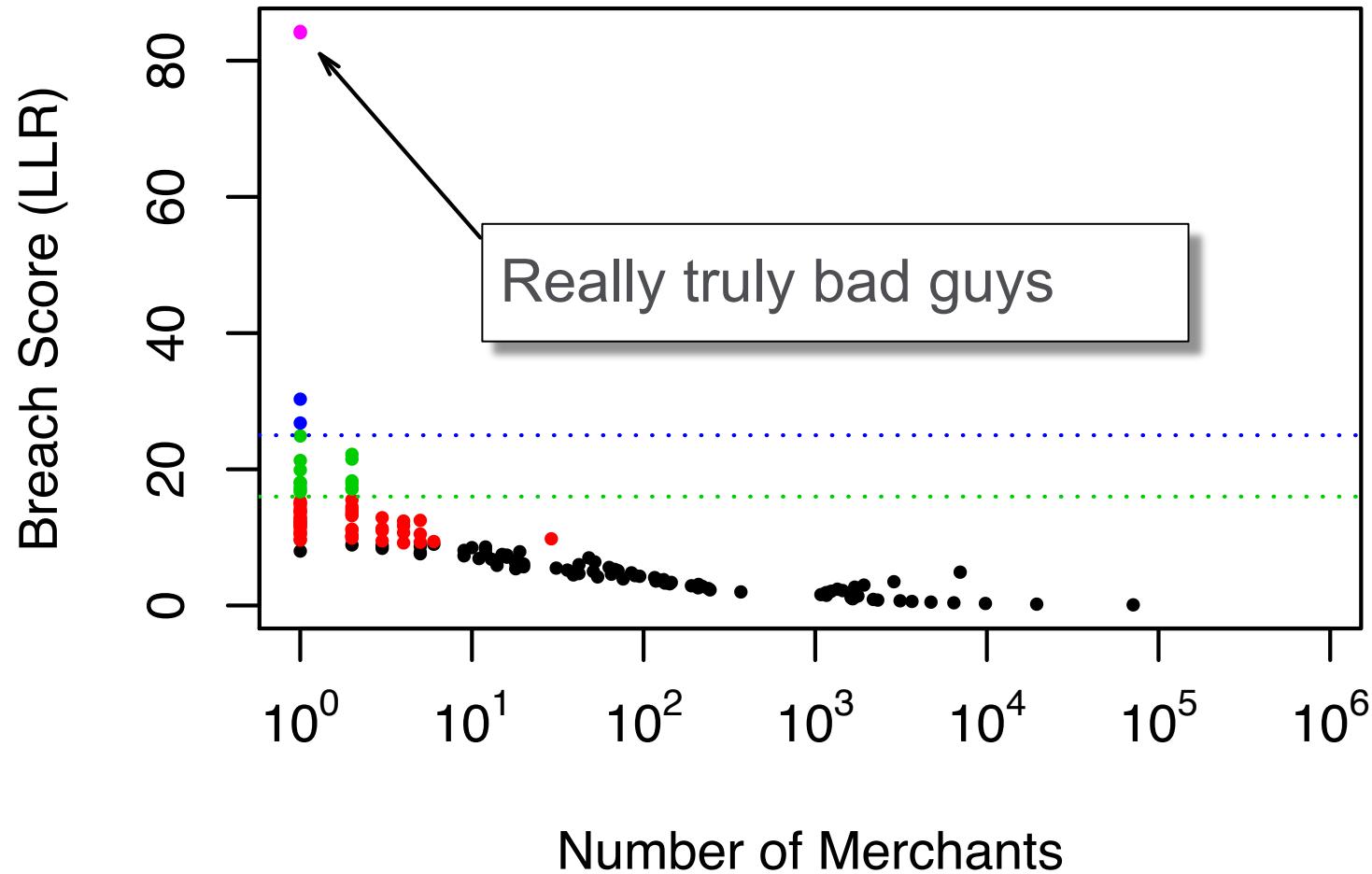
- Select histories that precede non-fraud
- And histories that precede fraud detection
- Analyze 2x2 cooccurrence of merchant  $n$  versus fraud detection

## LLR score for simulated merchants



What about the  
real world?

## LLR score for real data

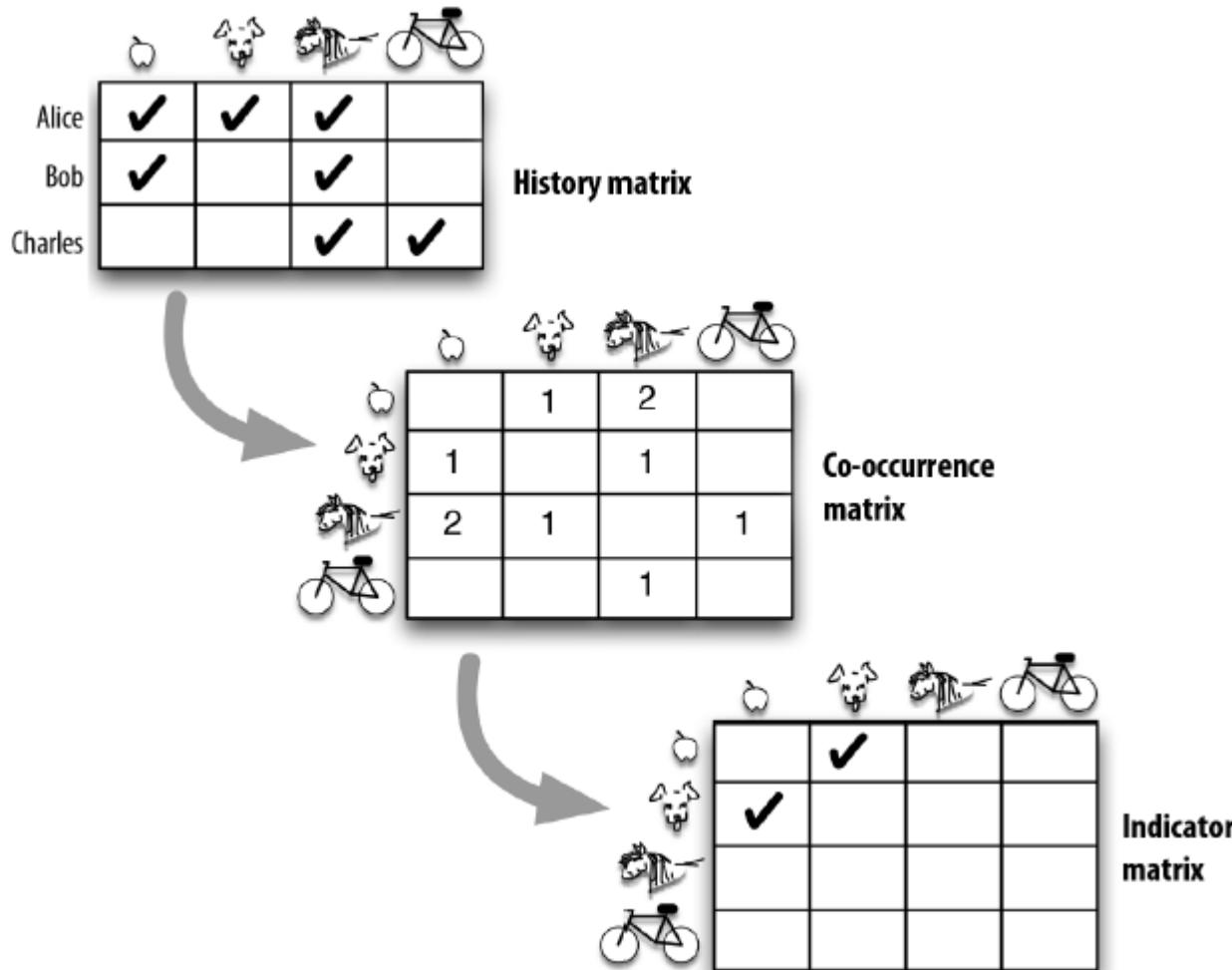


Historical cooccurrence gives high  
S/N

Historical cooccurrence gives high  
S/N

(we win)

# Cooccurrence Analysis



# Real-life example

- Query: “Paco de Lucia”
- Conventional meta-data search results:
  - “hombres de paco” times 400
  - not much else
- Recommendation based search:
  - Flamenco guitar and dancers
  - Spanish and classical guitar
  - Van Halen doing a classical/flamenco riff

# Real-life example

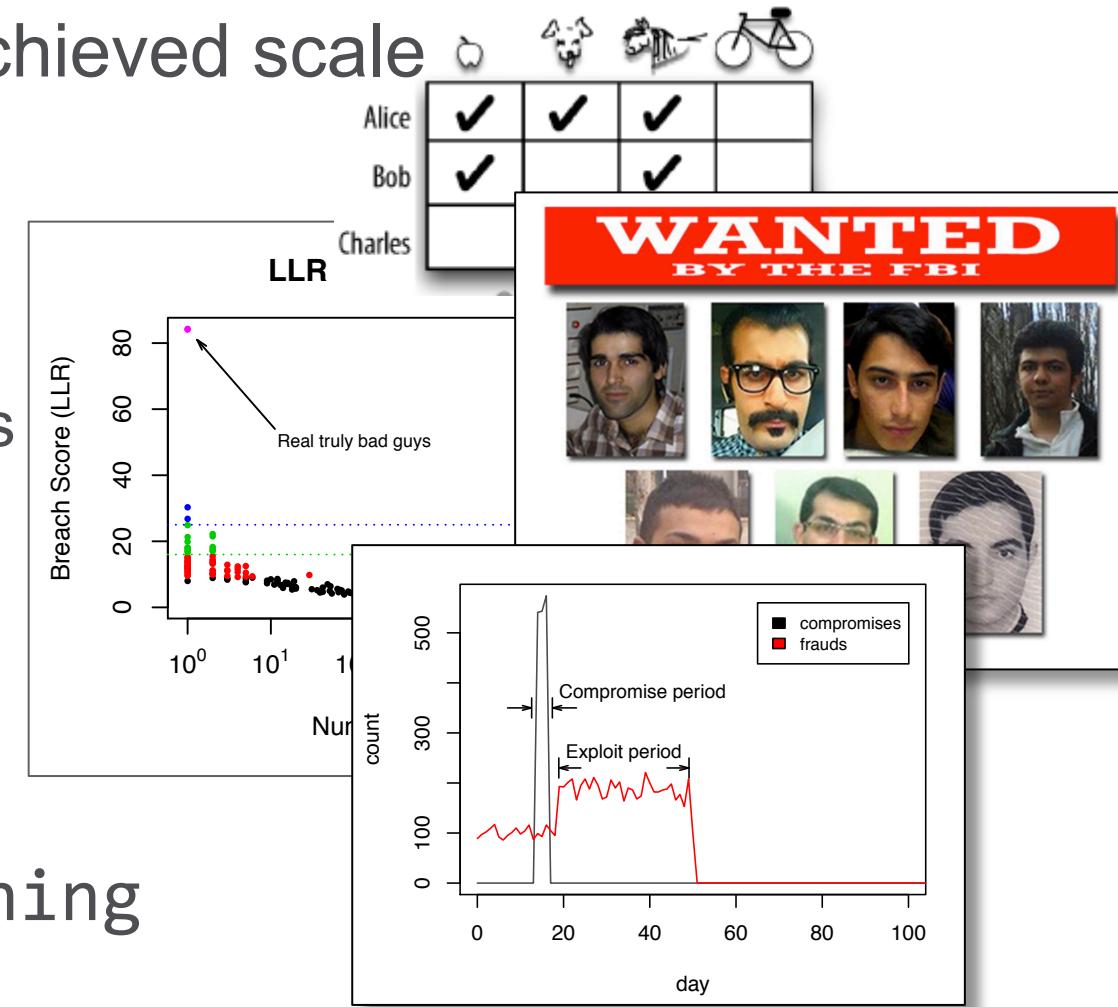
- 
- |  |   |
|--|---|
|    | <a href="#"><u>CONCIERTO CIUDAD DE LAS IDEAS PARTE FINAL</u></a><br>Music<br>58 views   |
|    | <a href="#"><u>Siudy / Buleria</u></a><br>Music<br>722 views                            |
|    | <a href="#"><u>Vicente Amigo 2ª parte Ciudad de las Ideas</u></a><br>Music<br>124 views |
|   | <a href="#"><u>Van Halen's Eruption</u></a><br>Music<br>4400 views                      |
|  | <a href="#"><u>Freestyle Flamenco</u></a><br>Music<br>653 views                         |

## So ...

- There are suddenly lots of these problems
- Simple techniques have surprising power at scale
  - Cooccurrence via G<sup>2</sup> / LLR
  - Distributional anomaly detection via *t*-digest
- These simple techniques are largely unsuitable for academic research
- But they are highly applicable in resource constrained industrial settings

# Summary

- We live in a golden age of newly achieved scale
- That scale has lowered the tree
  - Hard problems are much easier
  - Lots of low-hanging fruit all around us
- Cheap learning has huge value
- Code available at  
<http://github.com/tdunning>



# Me, Us

- Ted Dunning, MapR Chief Application Architect, Apache Member
  - Committer PMC member Zookeeper, Drill, others
  - Mentor for Flink, Beam (nee Dataflow), Drill, Storm, Zeppelin
  - VP Incubator
  - Bought the beer at the first HUG
- MapR
  - Produces a converged platform for big and fast data
  - Includes data platform (files, streams, tables) + open source
  - Adds major technology for performance, HA, industry standard API's
- Contact

[@ted\\_dunning](mailto:@ted_dunning), [ted.dunning@gmail.com](mailto:ted.dunning@gmail.com), [tdunning@mapr.com](mailto:tdunning@mapr.com)



# Q & A