



Overlapping Experiment Infrastructure: More, Better, Faster

Diane Tang, Ashish Agarwal,
Mike Meyer, Deirdre O'Brien



Why run experiments?

- Experiments:
 - Live traffic = incoming search queries
 - Experiments vs. experiment groups
 - Gathers data on impact of changes
 - How do users behave differently, if at all?
- Data-driven decisions:
 - UI
 - Algorithms

Why run experiments?

- Gathers data on impact of changes
 - How do users behave differently, if at all?
- Data-driven decisions:
 - UI

[Hotels.com Official Site](http://www.hotels.com)

www.hotels.com

Hotels.com Low Rates Guaranteed! Call a **Hotel** Expert. 1-866-925-0513

[Hotels.com Official Site](http://www.hotels.com)

www.hotels.com

Hotels.com Low Rates Guaranteed! Call a **Hotel** Expert. 1-866-925-0513

[Hotels.com Official Site](http://www.hotels.com)

www.hotels.com

Hotels.com Low Rates Guaranteed! Call a **Hotel** Expert. 1-866-925-0513



Why run experiments?

- Gathers data on impact of changes
 - How do users behave differently, if at all?
 - Test everything!
- Data-driven decisions
 - UI



Google



Google

Amazon.com: Online Shopping for Electronics, Apparel ...

www.amazon.com/

amazon.com

Remove

amazon

amazon **books**

amazon **ec2**

amazon **s3**

amazon **kindle**

amazon **coupons**

amazon **prime**

amazon **web services**

amazon **mp3**

Google Search

I'm Feeling Lucky

Why run experiments?

- Gathers data on impact of changes
 - How do users behave differently, if at all?
- Data-driven decisions
 - UI
 - Algorithms, e.g. CTR prediction
 - How many passes over the data
 - Date range
 - Different machine learning algorithms

Why run so many experiments?

- Goal: maintain innovation while growing
- **More:**
 - More simultaneous experiments
 - More variety in the types of experiments supported
- **Better:**
 - Valid experiments
 - Robust experiment design
- **Faster:**
 - Easy and quick experiment set-up
 - Experimental data available quickly and automatically
 - Quick iteration



Why is running so many expts hard?

- Infinite traffic, right? Wrong!
- High variability of metrics
 - English vs. Swahili
 - "flowers" vs. "who said 'if i had the time, this letter would be shorter'"
- Low trigger rate changes
 - e.g., weather information
- Consequence: experiments need a lot of traffic to get statistically significant results in a reasonable timeframe

Basic Experiment Definitions

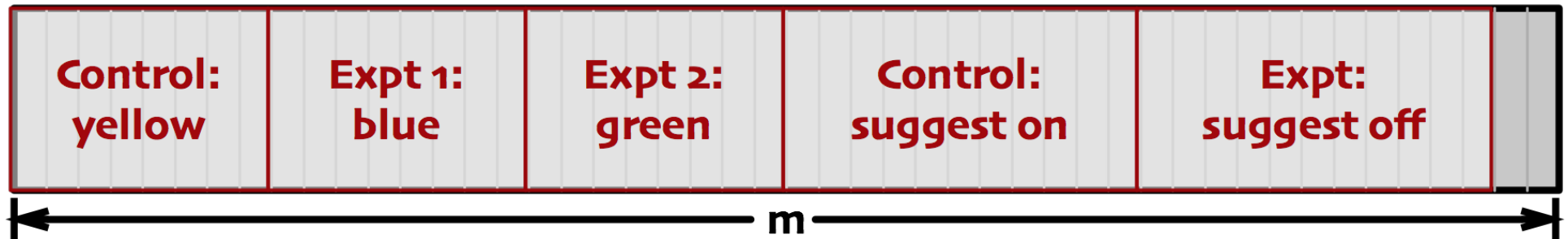
- Incoming search query request R has:
 - Cookie C
 - Conditions T
 - Query language, User country, Browser, etc.
- System has *parameters*
 - E.g., top ad background color, Google Suggest on or off
 - Default value
- Experiment:
 - Diversion: is a request in the experiment?
 - Conditions
 - Unit of diversion: cookie vs. traffic
 - Experiment parameter values



Extreme 1: Single Layer

- Our experiment infrastructure prior to 2007
- Every request in at most one experiment
- Straightforward, but insufficiently scalable
 - Variability
 - Low trigger rate

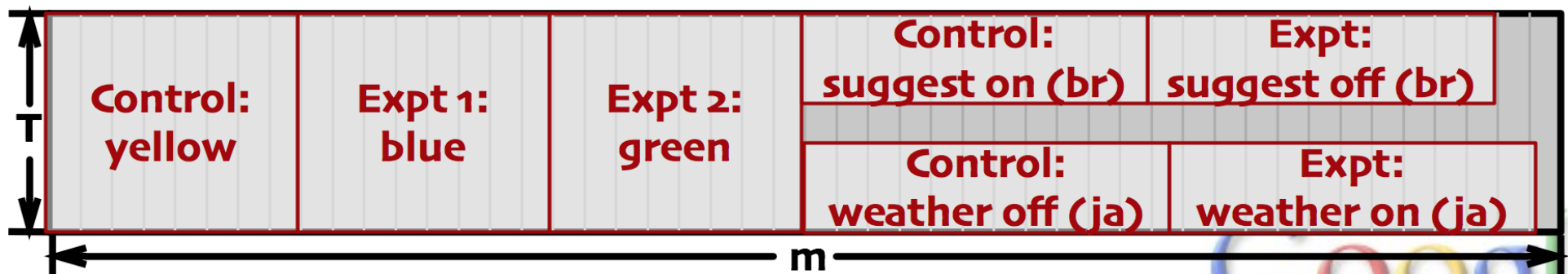
Incoming request **R**
has cookie **C**
 $f(C) \% 1000 = m$



Scaling the Single Layer

- Use incoming traffic more effectively by understanding which conditions are disjoint with other conditions
 - e.g., Brazil vs. Japan (country)
 - other examples: language, browser
- Increases scalability but more complex, more fragmentation

**Incoming request R
has cookie C
 $f(C) \% 1000 = m$
and conditions T**



Extreme 2: Multi-factorial Expt Design

- Vary each parameter independently
- Issues:
 - Must serve valid pages only
 - e.g., blue text on blue background
 - Constantly changing system
 - Adding / removing parameters
 - Different experiments use different sets of parameters
 - Can't design once and be done with it

Incoming request R

param: top ad bkgd color	Control: yellow	Expt 1: blue	Expt 2: green
param: top ad text color	Control: black	Expt 1: blue	Expt 2: green
param: suggest	Control: suggest on		Expt: suggest off

Layers: Multiplies number of expts

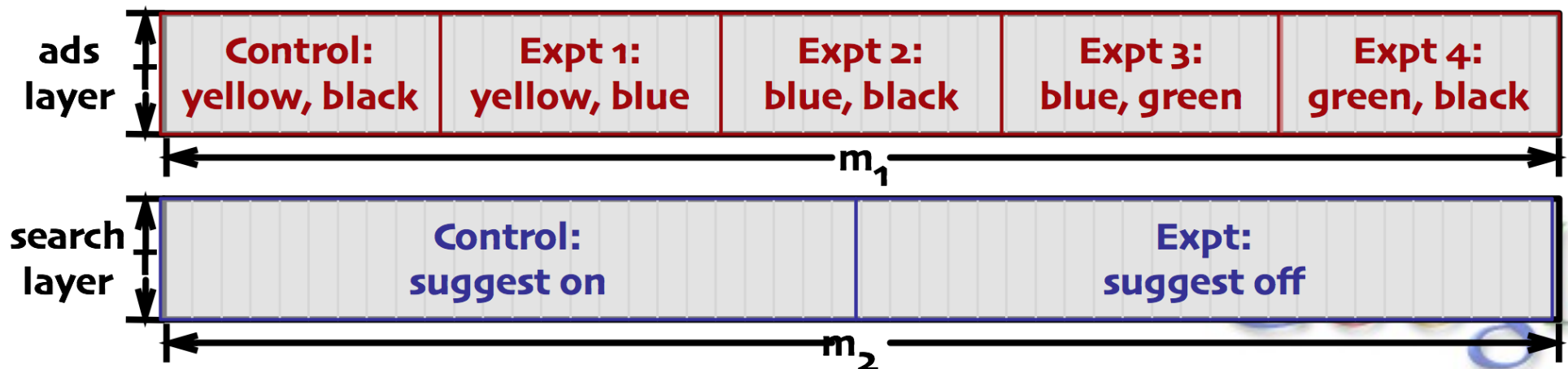
- Partition parameters into sets --> layers
- Experiments can only modify parameters associated with that layer
- Each layer independent of every other layer
- Controls and experiments must be in same layer

**Incoming request R
has cookie C**

$$f(C, \text{layer}_1) \% 1000 = m_1$$

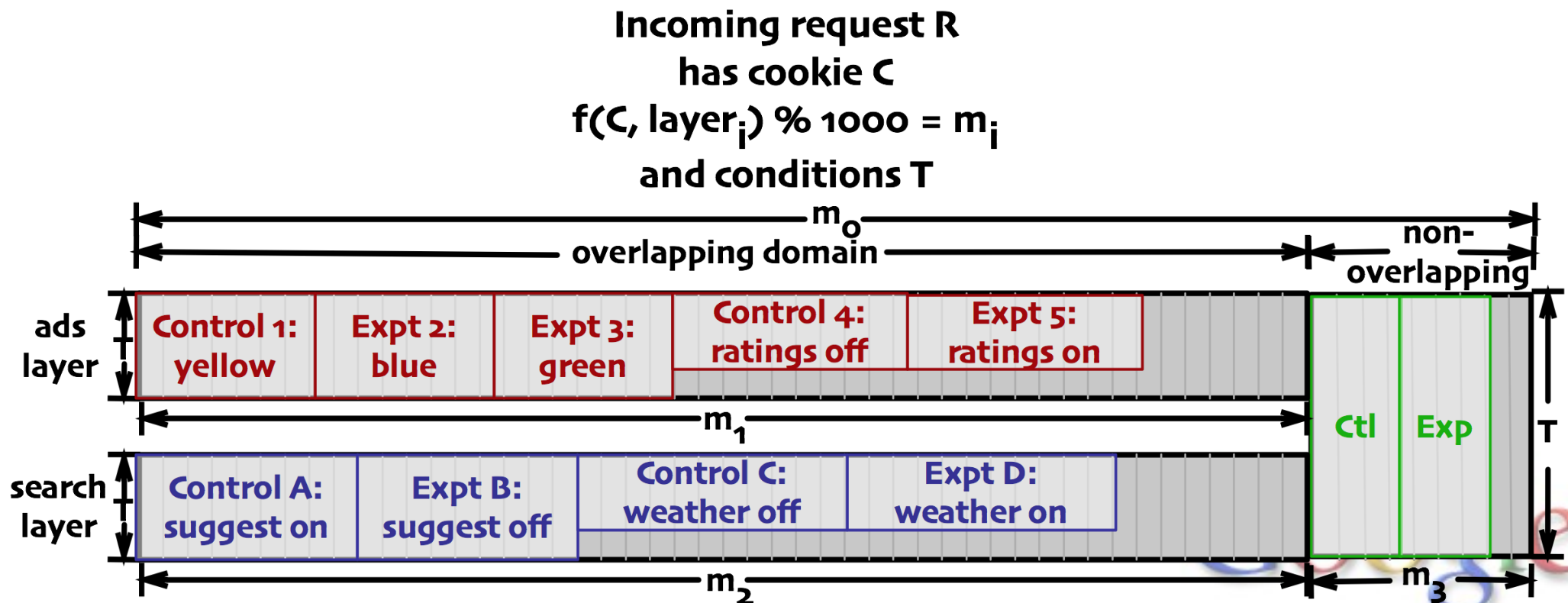
$$f(C, \text{layer}_2) \% 1000 = m_2$$

and conditions T



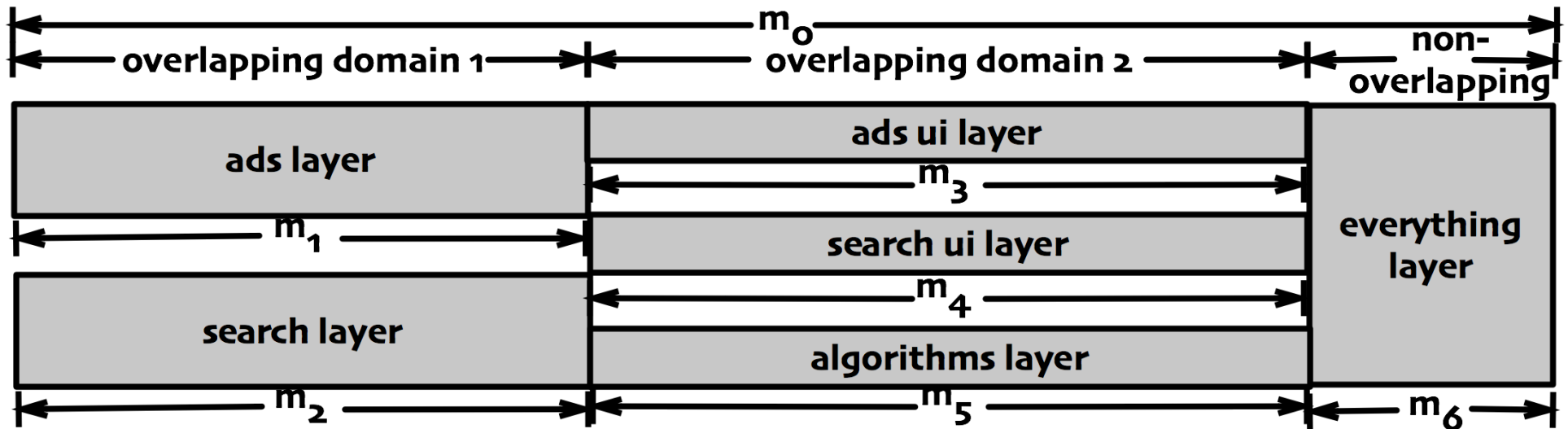
Domains: Nesting to increase flexibility

- Domains: contain layers
- Layers: contain domains and experiments
- Nesting:
 - Allows for different partitioning of parameters
 - Trade-off: less efficient use of space due to fragmentation



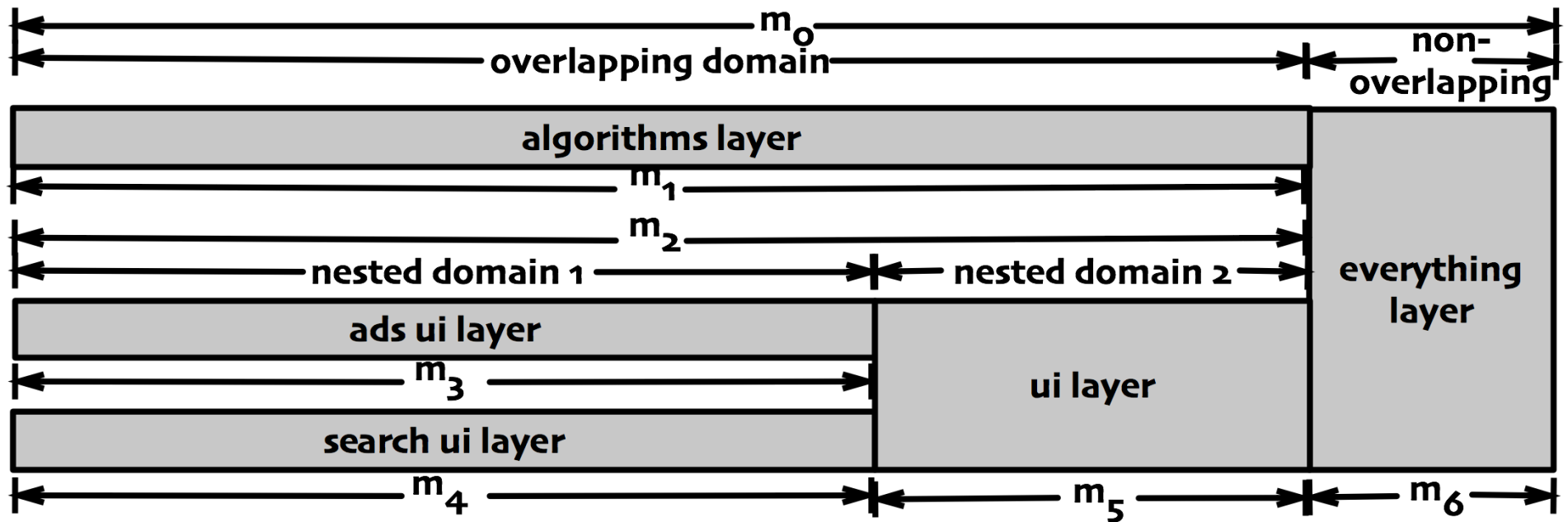
Nesting: another example

Incoming request R
has cookie C
 $f(C, \text{layer}_i) \% 1000 = m_i$
and conditions T



Nesting: one last example

Incoming request R
has cookie C
 $f(C, \text{layer}_i) \% 1000 = m_i$
and conditions T



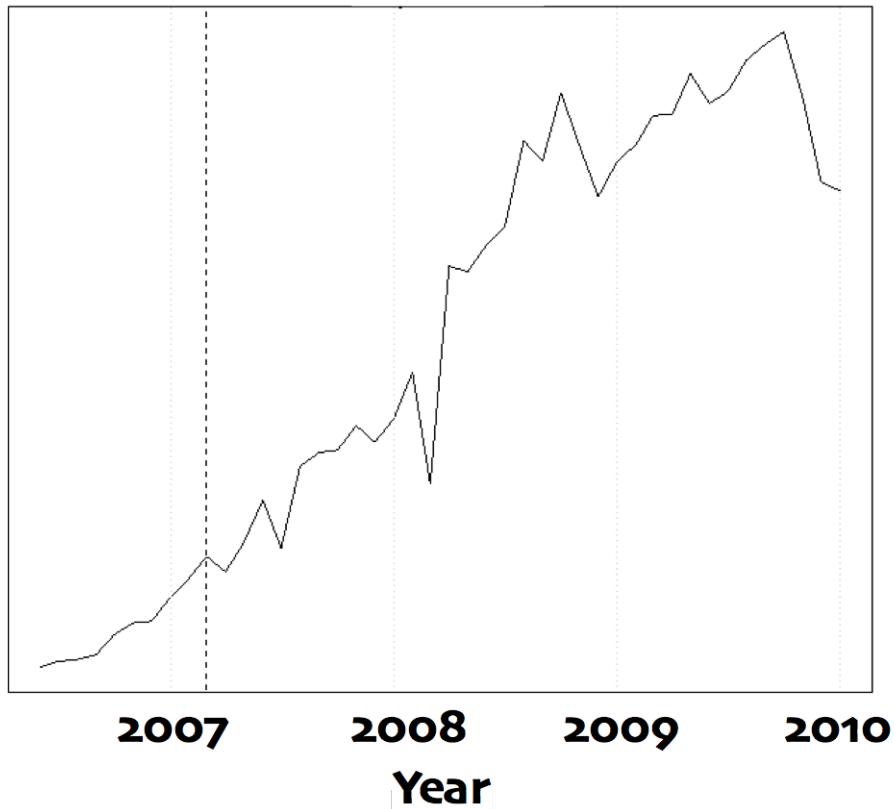
Merging Experiment Parameters

- Can we relax the constraint of associating each parameter with only one layer?
 - Consequence: request could be in two experiments, each modifying the same parameter
- How to merge parameter values?
 - Well-defined composition function, e.g., multiplication
 - Well-understood parameter
- Example:
 - Threshold t with base value V
 - Layer 1: experiment with multiplier 1.5, control: 1.0
 - Layer 2: experiment with multiplier 2.0 control: 1.0
 - 4 possibilities:
 - $t * 1.5 * 1.0$
 - $t * 1.0 * 1.0$
 - $t * 2.0 * 1.5$
 - $t * 2.0 * 1.5$

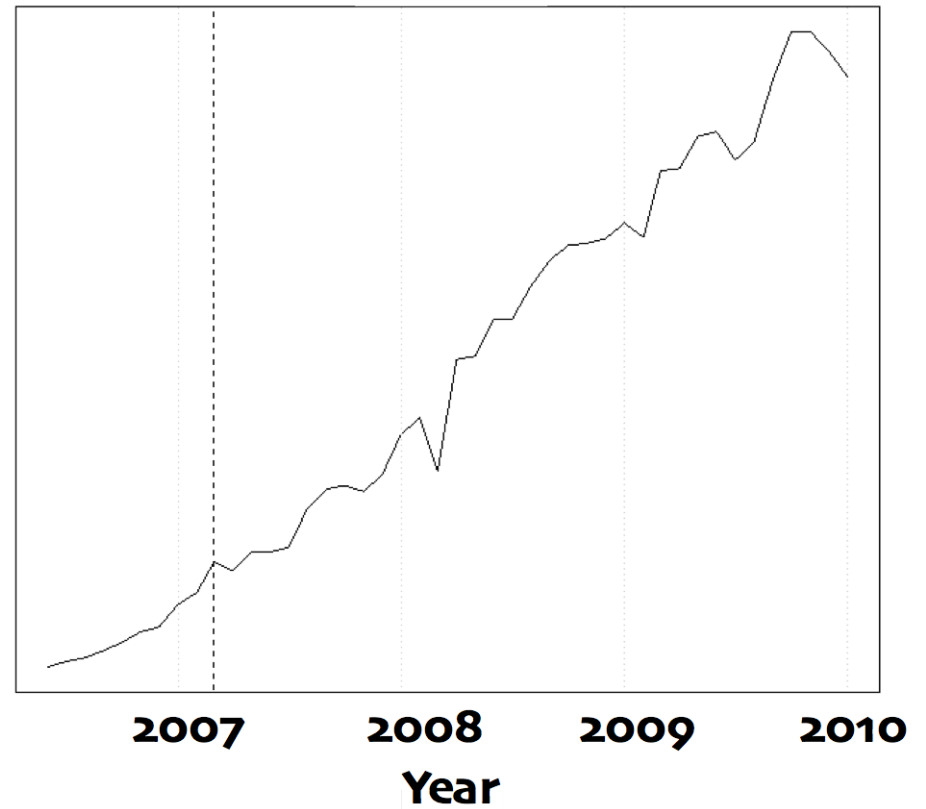


More: Results

Simultaneous Experiments



Simultaneous Expt Owners



Conclusions

- Overlapping experiment infrastructure delivers scalability & flexibility
 - Conditions
 - Layers
 - Domains
 - Mergeable parameters
- More than infrastructure needed though:
 - Tools
 - Experiment Design (sizing, finding cookies, experiment config)
 - Analysis
 - Education
 - Culture



A decorative header featuring four overlapping spheres: a green one on the far left, and a blue, red, and yellow one overlapping each other towards the right. A thin black horizontal line runs across the slide below the spheres.

Questions?

The Google logo, consisting of the word "Google" in its signature multi-colored font (blue, red, yellow, green, red) with a trademark symbol (TM) to the right.

Google™