Overlapping Experiment Infrastructure: More, Better, Faster

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

Google™

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

Google™

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

www.**hotels**.com

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

Hotels.com Official Site

www.**hotels**.com

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

Hotels.com Official Site

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

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

Google™

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

Google™

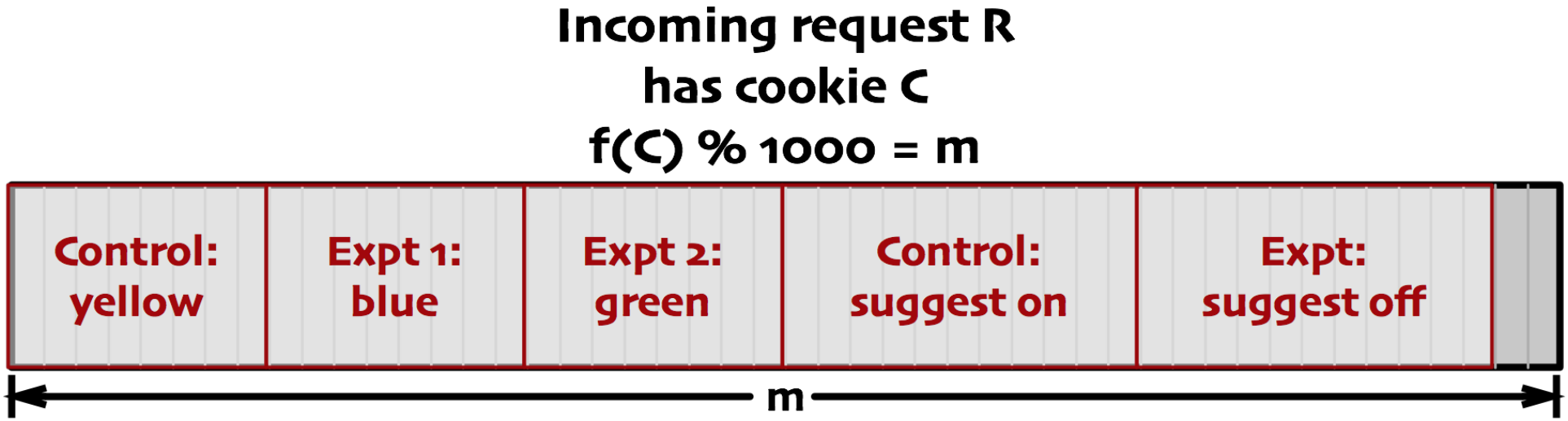
Basic Experiment Definitions

- 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

Google™

Extreme 1: Single Layer

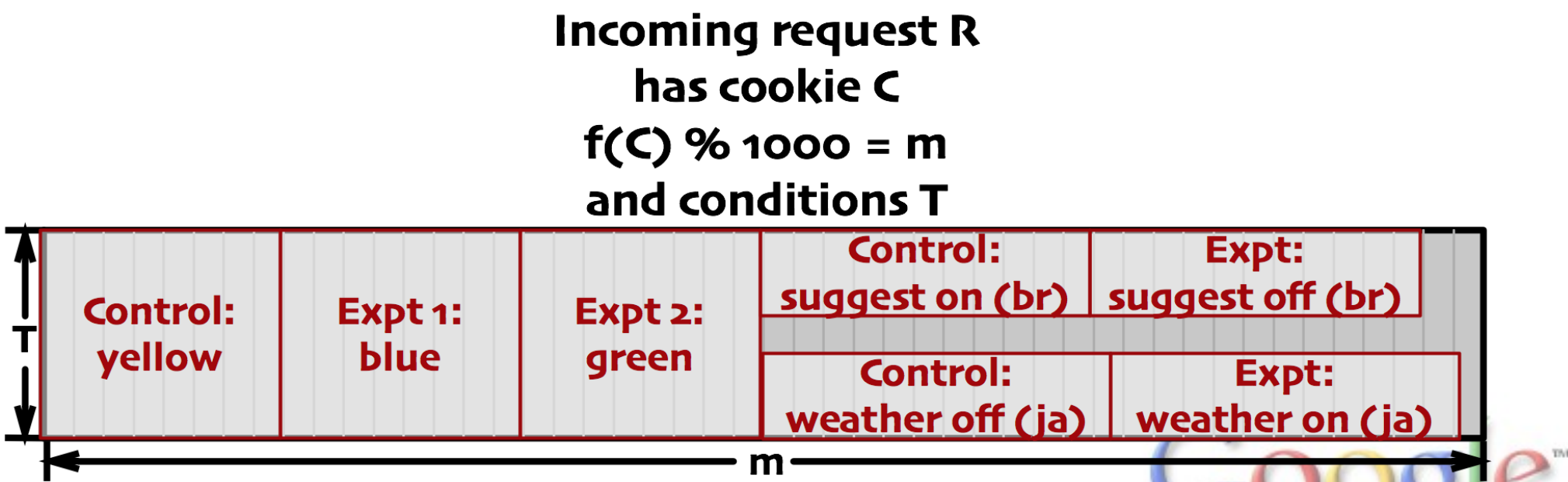
- Our experiment infrastructure prior to 2007
- Every request in at most one experiment
- Straightforward, but insufficiently scalable
  - Variability
  - Low trigger rate



Google

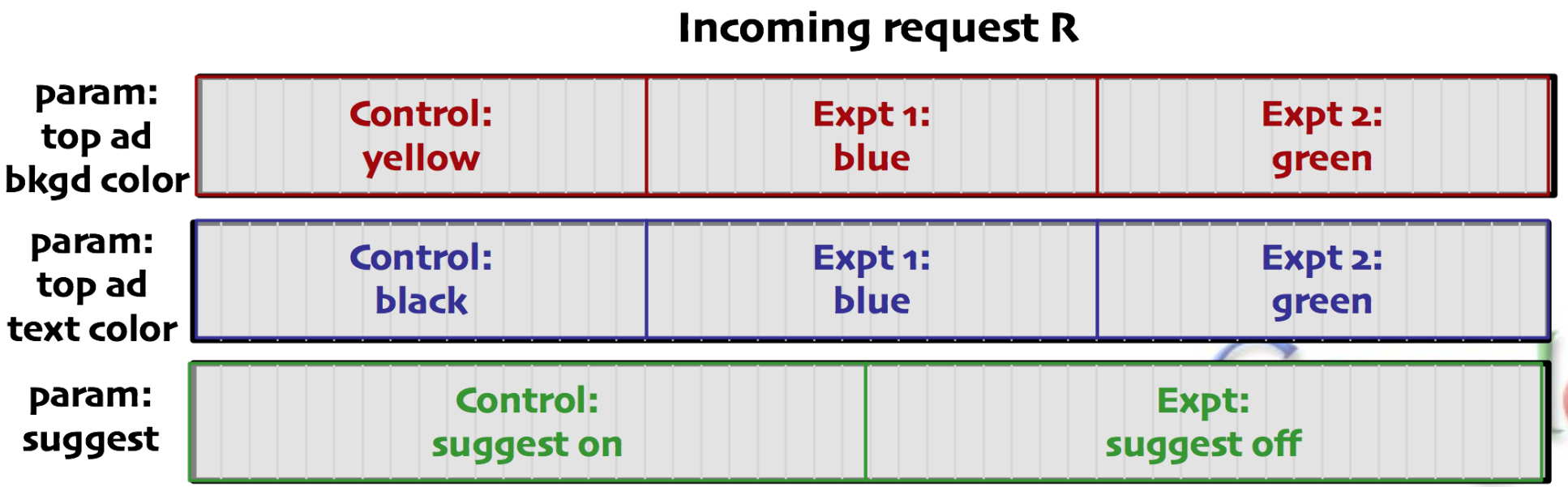
Scaling the Single Layer

- 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



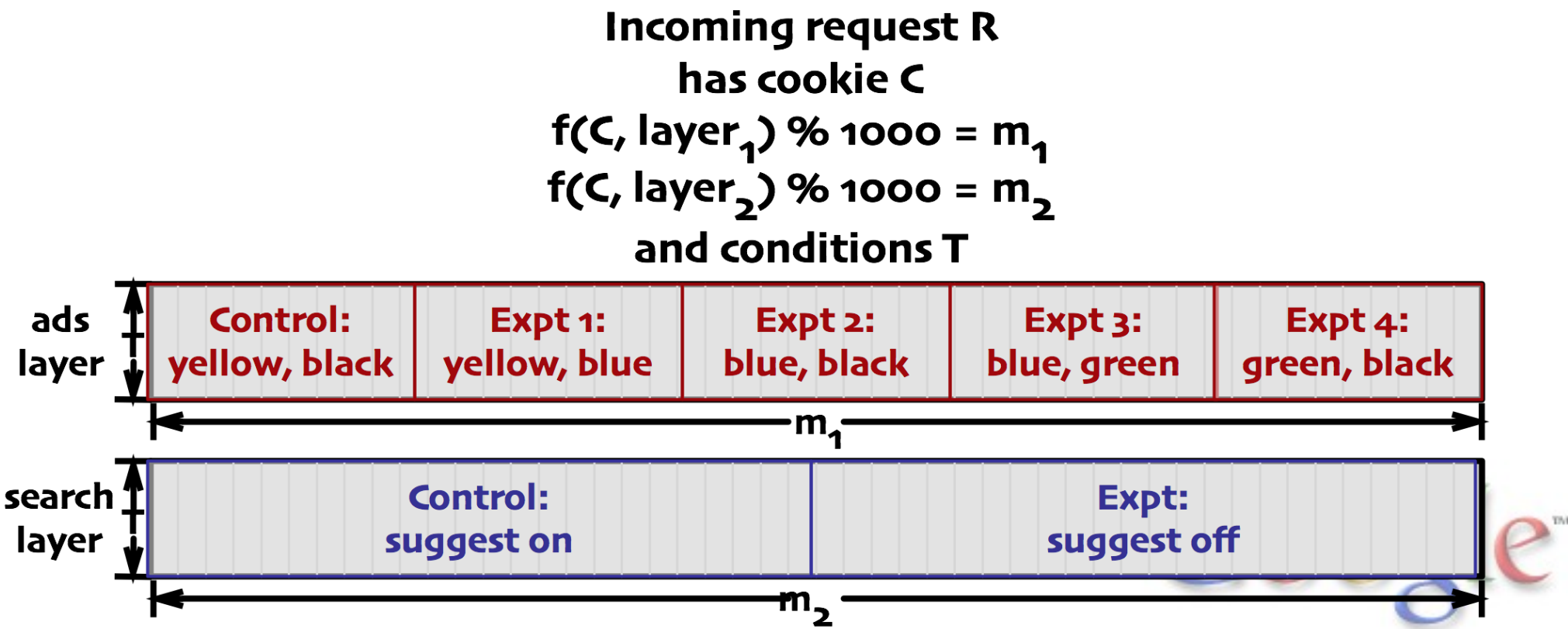
Extreme 2: Multi-factorial Expt Design

- 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



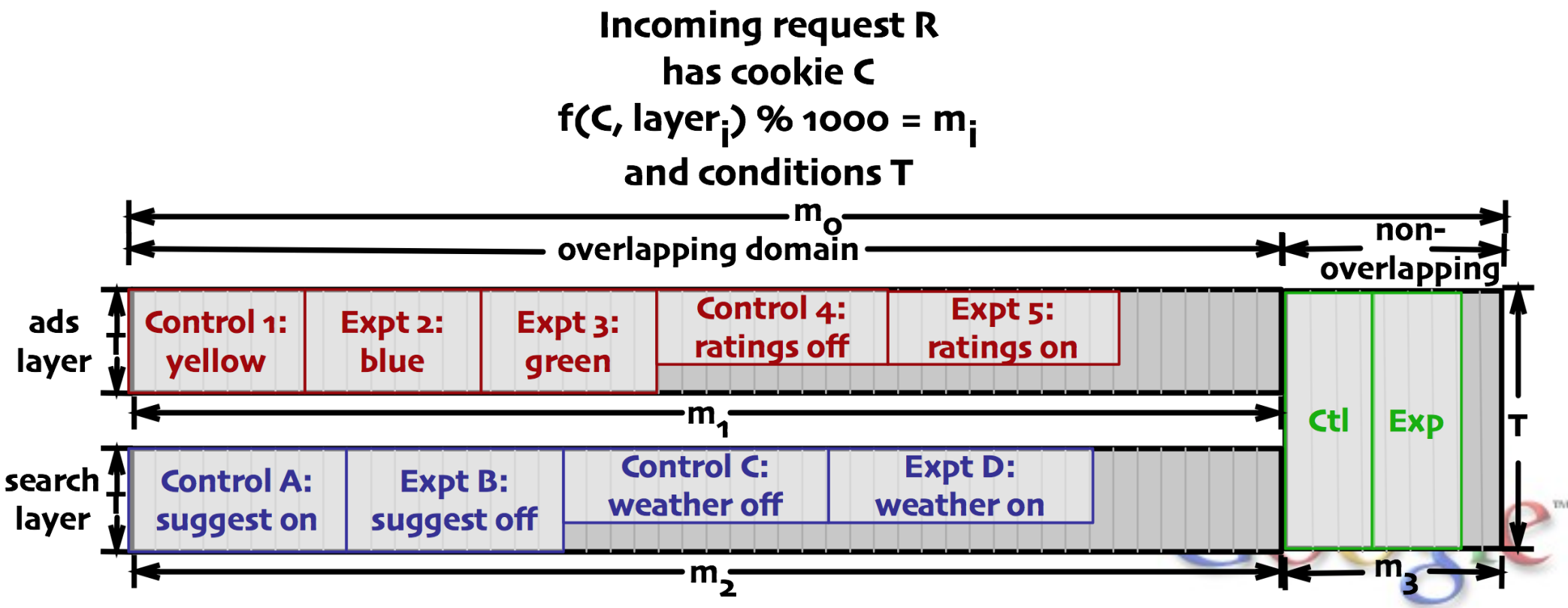
Layers: Multiplies number of expts

- 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

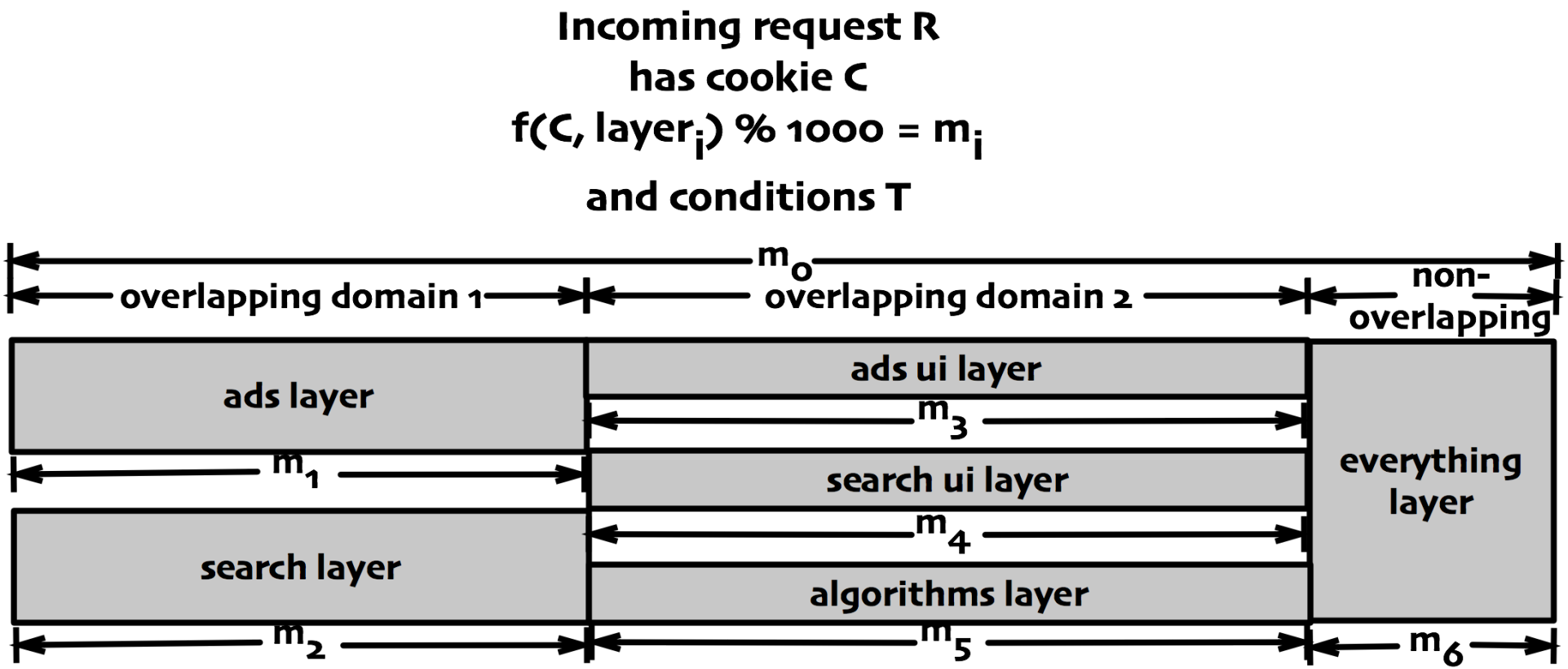


Domains: Nesting to increase flexibility

- 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

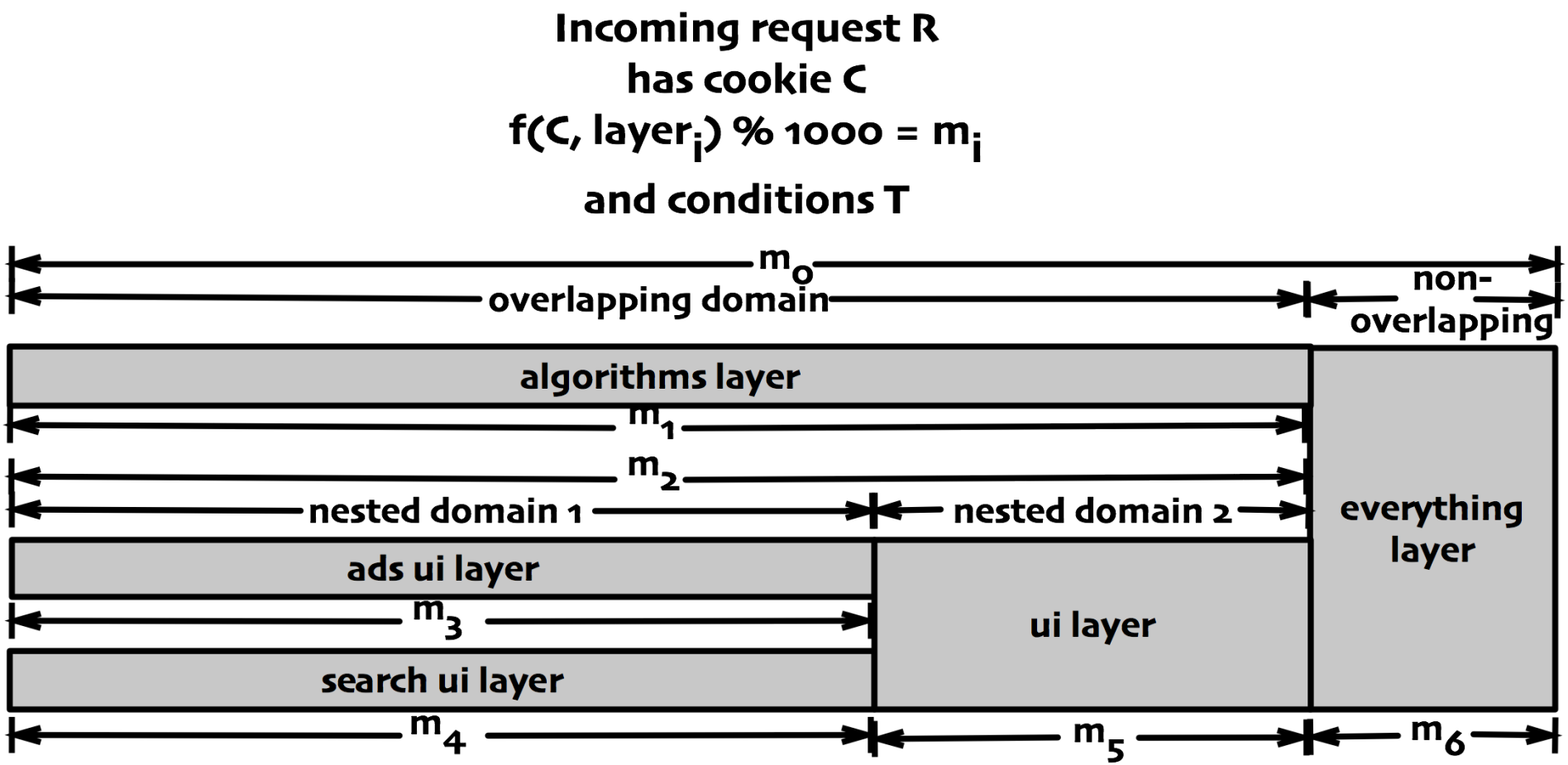


Nesting: another example



Google

Nesting: one last example



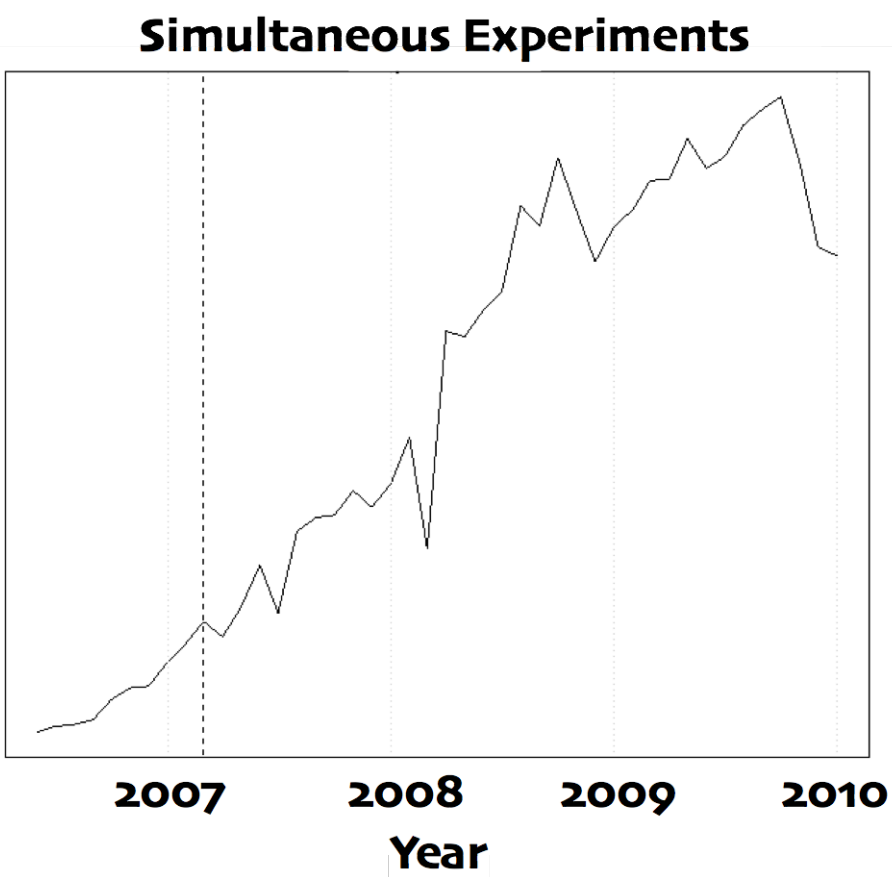
Google

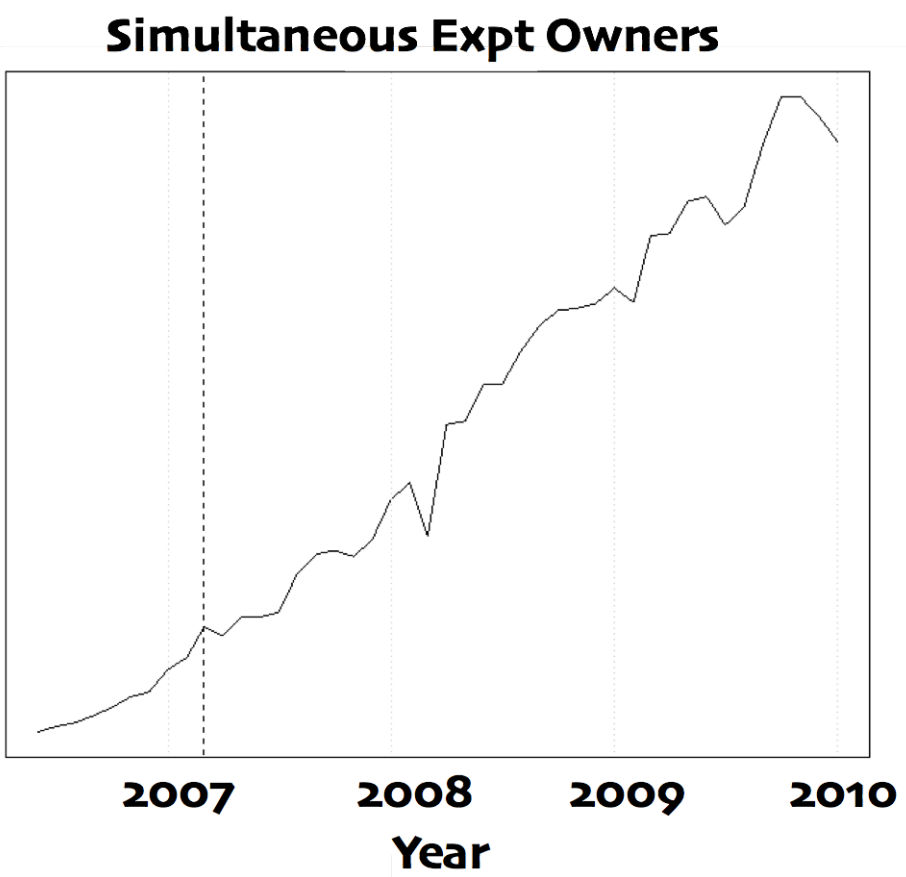
Merging Experiment Parameters

- 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

Google™

More: Results





Google™

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

Google™

Questions?