Powering Heap

Dan Robinson
Lead Engineer, Heap

whoami

- Joined as Heap's first hire in July, 2013
- Previously a backend engineer at Palantir
- Stanford '11 in Math and CS

Quick Overview

- What is Heap?
- Why is what we're building such a difficult data problem?
- How Heap is distributed.
- How we make subqueries fast.
    - Partial indexes.
    - UDFs in C.
- How distributed systems operations work.

HEAP

Spacious 3 bedroom ocean

Dan

Airbnb, Inc. [US] https://www.airbnb.com/rooms/61383?checkin=11%2F11%2F2015&checkout=11%2F17%2F2015&guests=5&s=5qOSBoYj

Photos | About this listing | Reviews | The Host | Location

$224 Per Night

Prices — Cancellation: Strict

Description

The Space

Rent 6 nights and stay for 7!

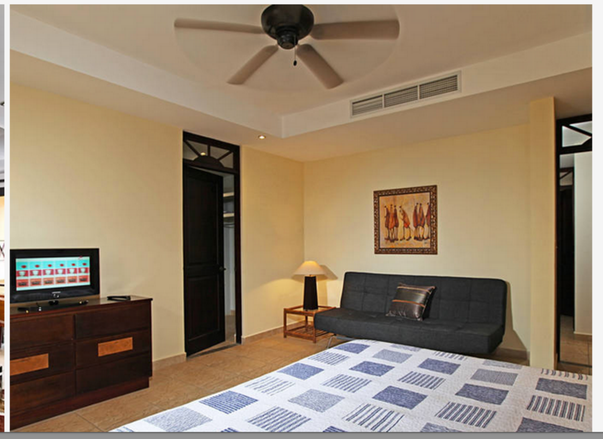
it features:

-Spacious living room with comfortable sofas and large flat screen TV.
-Terrace with an outdoor dining table.
-Master bedroom furnished with a king size bed and private bathroom.
-The second and third bedrooms have a full sized bed and they share a full

+ More

House Rules — Contact us prior booking!

Availability — **1 night** minimum stay — View Calendar



Check In: 11/11/2015 | Check Out: 11/17/2015 | Guests: 5

$224 x 6 nights	$1344
Service fee	$113
Total	$1457

Request to Book

Save to Wish List

Email | Messenger | ··· More

Report this listing

```
bookHotelButton.addEventListener("click", function() {
    Analytics.track('Booked Hotel');
});
```

Costa Rica - Airbnb

Dan

Airbnb, Inc. [US] https://www.airbnb.com/s/Costa-Rica?checkin=11%2F11%2F2015&checkout=11%2F17%2F2015&guests=5&zoom=7&sear...

Costa Rica

Become a Host

Messages

Help

Dan

Dates 11/11/2015 11/17/2015 5 Guests

Room Type Entire Home Private Room Shared Room

Price Range

$10 $228 Average $1000+

More Filters

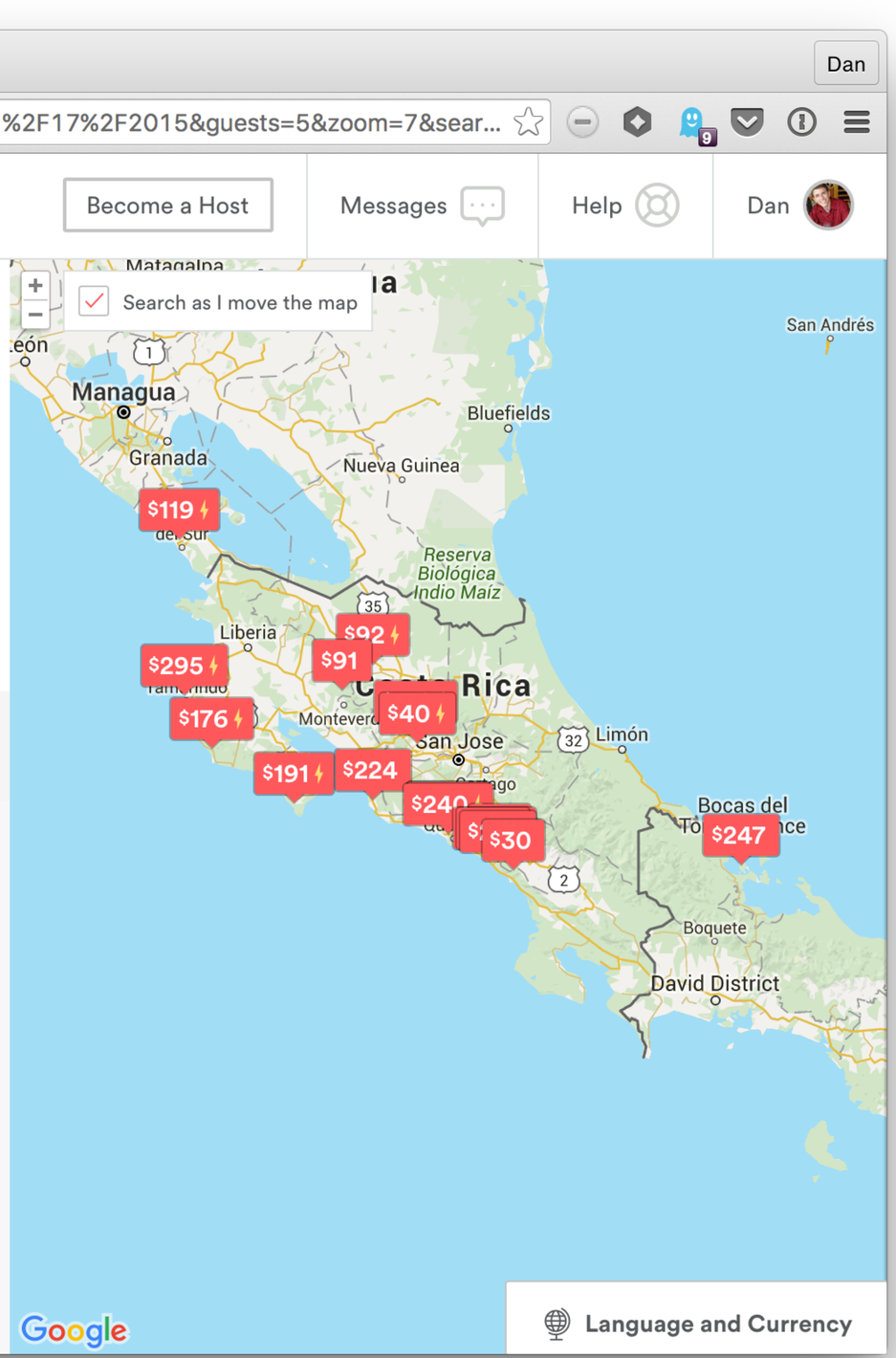
300+ Vacation Rentals · Costa Rica

#3 Fully Furnished Apartment & A...

Entire home/apt · 4.7 ★ · 78 reviews

Tropical Island Reef Cabin

Private room · 4.8 ★ · 59 reviews



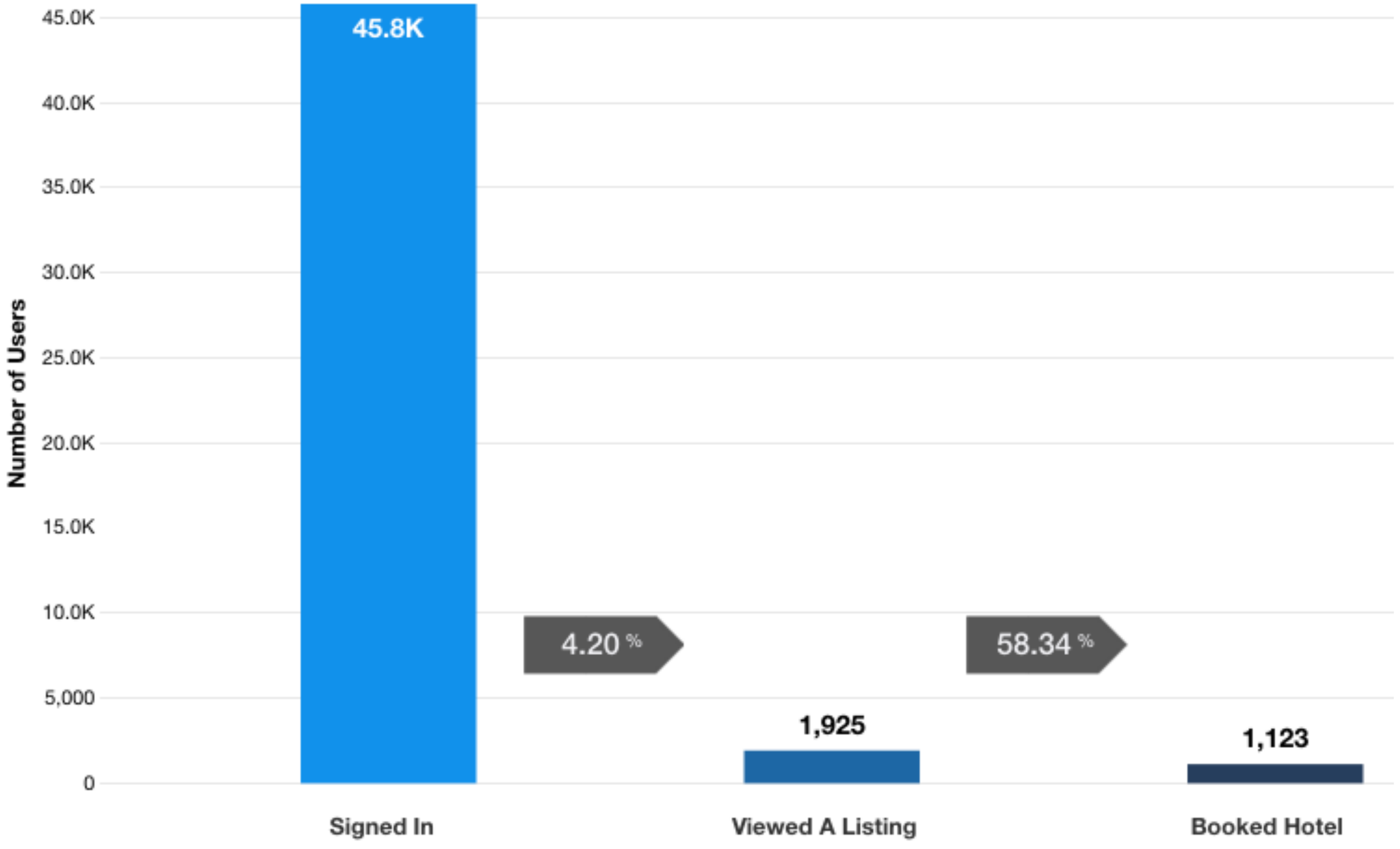
```
listingDetailPage.addEventListener("load", function() {
    Analytics.track('Viewed A Listing');
});

...

if (signInAttempt.isSuccessful) {
    Analytics.track('Signed In');
}

...

submitCreditCardButton.addEventListener("click", function() {
    Analytics.track('Entered Credit Card');
}
```


45.0K
40.0K
35.0K
30.0K
25.0K
20.0K
15.0K
10.0K
5,000
0
Number of Users
45.8K
4.20 %
1,925
58.34 %
1,123
Signed In
Viewed A Listing
Booked Hotel

Analytics is fundamentally iterative.

Capture everything that happens.

Analyze the data retroactively.

Web

To get started with Heap, paste the following code snippet before your website's closing `</head>` tag:

```
<script type="text/javascript">
  window.heap=window.heap||[],heap.load=function(e,t){window.heap.appid=e,window.heap.config=t=
  heap.load("236035469");
</script>
```

```
    user_id     |     time      |                                                          event
----------------+---------------+----------------------------------------------------------------------------------------------------------------------------------------------------
 56626881176383 | 1403873730675 | "path"=>"/", "time"=>"1403873730675", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 56626881176383 | 1403873730676 | "time"=>"1403873730676", "object"=>"session", "library"=>"web", "event_id"=>"10000736993236", "session_id"=>"2871184296", "session_time"=>"14038
 56626881176383 | 1403873733517 | "href"=>"/login", "path"=>"/", "time"=>"1403873733517", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.co
 56626881176383 | 1403873733993 | "path"=>"/login", "time"=>"1403873733993", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pagevie
 56626881176383 | 1403873738031 | "path"=>"/login", "time"=>"1403873738031", "type"=>"click", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "
 56626881176383 | 1403873738031 | "path"=>"/login", "time"=>"1403873738031", "type"=>"click", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "
 71344982188381 | 1443737793660 | "ip"=>"50.204.254.254", "path"=>"/Users/flannimal/Desktop/heap/index.html", "time"=>"1443737793660", "title"=>"Diesel Shark | App for Truck Driv
 71344982188381 | 1443737793660 | "ip"=>"50.204.254.254", "time"=>"1443737793660", "object"=>"session", "browser"=>"Chrome 45.0.2454", "country"=>"United States", "library"=>"web
 43343298401944 | 1411925936155 | "path"=>"/pricing", "time"=>"1411925936155", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pag
 78902486505078 | 1414497243775 | "path"=>"/", "time"=>"1414497243775", "title"=>"Heap Blog | Official blog for Heap", "domain"=>"blog.heapanalytics.com", "object"=>"pageview", "
 78902486505078 | 1414497243780 | "ip"=>"80.150.229.30", "time"=>"1414497243780", "object"=>"session", "browser"=>"Chrome 38.0.2125", "country"=>"Germany", "library"=>"web", "eve
 54528649436646 | 1430914612402 | "ip"=>"62.159.27.1", "time"=>"1430914612402", "object"=>"session", "browser"=>"Safari 8.0.5", "country"=>"Germany", "library"=>"web", "event_id"
 54528649436646 | 1430914612402 | "ip"=>"62.159.27.1", "path"=>"/", "time"=>"1430914612402", "query"=>"?utm_source=badge", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"
 60605760782453 | 1444015898050 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "time"=>"1444015898050", "object"=>"session", "region"=>"Virginia", "browser"=>"Chrome 45.0.2454", "cou
 60605760782453 | 1444015898050 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "path"=>"/", "time"=>"1444015898050", "query"=>"?utm_source=badge", "title"=>"Heap | Mobile and Web Ana
 60605760782453 | 1444015928598 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "href"=>"/features", "path"=>"/", "time"=>"1444015928598", "type"=>"click", "query"=>"?utm_source=badge
 60605760782453 | 1444015928990 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "path"=>"/features/data-capture", "time"=>"1444015928990", "title"=>"Data Capture - Heap | Mobile and W
 60605760782453 | 1444015941595 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "href"=>"/features/users", "path"=>"/features/data-capture", "time"=>"1444015941595", "type"=>"click",
 60605760782453 | 1444015941828 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "path"=>"/features/users", "time"=>"1444015941828", "title"=>"Users - Heap | Mobile and Web Analytics",
 47796230488402 | 1410696615406 | "path"=>"/", "time"=>"1410696615406", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 47796230488402 | 1410696615407 | "time"=>"1410696615407", "object"=>"session", "library"=>"web", "event_id"=>"8257717896", "referrer"=>"https://www.google.com/", "session_id"=>"
 47796230488402 | 1410696625798 | "href"=>"/login", "path"=>"/", "time"=>"1410696625798", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.co
 47796230488402 | 1410696627036 | "path"=>"/login", "time"=>"1410696627036", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pagevie
 47796230488402 | 1410696629203 | "path"=>"/", "time"=>"1410696629203", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 47796230488402 | 1410696631692 | "href"=>"/pricing", "path"=>"/", "time"=>"1410696631692", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.
 47796230488402 | 1410696633132 | "path"=>"/pricing", "time"=>"1410696633132", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pag
 47796230488402 | 1410696656800 | "href"=>"/login", "path"=>"/pricing", "time"=>"1410696656800", "type"=>"click", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>
 47796230488402 | 1410696657871 | "path"=>"/login", "time"=>"1410696657871", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pagevie
 47796230488402 | 1410696659768 | "path"=>"/pricing", "time"=>"1410696659768", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pag
 47796230488402 | 1410696664689 | "href"=>"/login", "path"=>"/pricing", "time"=>"1410696664689", "type"=>"click", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>
 47796230488402 | 1410696665586 | "path"=>"/login", "time"=>"1410696665586", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pagevie
 47796230488402 | 1410697234572 | "path"=>"/login", "time"=>"1410697234572", "type"=>"click", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "
 47796230488402 | 1410697237689 | "path"=>"/login", "time"=>"1410697237689", "type"=>"change", "title"=>"Login - Heap | Mobile and Web Analytics", "value"=>"dev@funzing.com", "do
 47796230488402 | 1410697240736 | "path"=>"/login", "time"=>"1410697240736", "type"=>"click", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "
 47796230488402 | 1410697240737 | "path"=>"/login", "time"=>"1410697240737", "type"=>"submit", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com",
 47796230488402 | 1410697243200 | "path"=>"/login", "time"=>"1410697243200", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "lib
 47796230488402 | 1410697253951 | "path"=>"/login", "time"=>"1410697253951", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "library"
 47796230488402 | 1410697262779 | "path"=>"/login", "time"=>"1410697262779", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "library"
 47796230488402 | 1411506838887 | "time"=>"1411506838887", "object"=>"session", "library"=>"web", "event_id"=>"8607771125", "session_id"=>"2208562923", "session_time"=>"141150683
 47796230488402 | 1411506838887 | "path"=>"/", "time"=>"1411506838887", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 47796230488402 | 1411976074500 | "path"=>"/", "time"=>"1411976074500", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 47796230488402 | 1411976074501 | "time"=>"1411976074501", "object"=>"session", "library"=>"web", "event_id"=>"8803770993", "session_id"=>"538285301", "session_time"=>"1411976074
```

Challenges

1. Capturing 10x to 100x as much data.
   Will never want 95% of it.

```
    user_id     |     time      |                                                          event
----------------+---------------+---------------------------------------------------------------------------------------------------------------------------------------------------------
 56626881176383 | 1403873730675 | "path"=>"/", "time"=>"1403873730675", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 56626881176383 | 1403873730676 | "time"=>"1403873730676", "object"=>"session", "library"=>"web", "event_id"=>"10000736993236", "session_id"=>"2871184296", "session_time"=>"14038
 56626881176383 | 1403873733517 | "href"=>"/login", "path"=>"/", "time"=>"1403873733517", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.co
 56626881176383 | 1403873733993 | "path"=>"/login", "time"=>"1403873733993", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pagevie
 56626881176383 | 1403873738031 | "path"=>"/login", "time"=>"1403873738031", "type"=>"click", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "
 56626881176383 | 1403873738031 | "path"=>"/login", "time"=>"1403873738031", "type"=>"click", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "
 71344982188381 | 1443737793660 | "ip"=>"50.204.254.254", "path"=>"/Users/flannimal/Desktop/heap/index.html", "time"=>"1443737793660", "title"=>"Diesel Shark | App for Truck Driv
 71344982188381 | 1443737793660 | "ip"=>"50.204.254.254", "time"=>"1443737793660", "object"=>"session", "browser"=>"Chrome 45.0.2454", "country"=>"United States", "library"=>"web
 43343298401944 | 1411925936155 | "path"=>"/pricing", "time"=>"1411925936155", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pag
 78902486505078 | 1414497243775 | "path"=>"/", "time"=>"1414497243775", "title"=>"Heap Blog | Official blog for Heap", "domain"=>"blog.heapanalytics.com", "object"=>"pageview", "
 78902486505078 | 1414497243780 | "ip"=>"80.150.229.30", "time"=>"1414497243780", "object"=>"session", "browser"=>"Chrome 38.0.2125", "country"=>"Germany", "library"=>"web", "eve
 54528649436646 | 1430914612402 | "ip"=>"62.159.27.1", "time"=>"1430914612402", "object"=>"session", "browser"=>"Safari 8.0.5", "country"=>"Germany", "library"=>"web", "event_id"
 54528649436646 | 1430914612402 | "ip"=>"62.159.27.1", "path"=>"/", "time"=>"1430914612402", "query"=>"?utm_source=badge", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"
 60605760782453 | 1444015898050 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "time"=>"1444015898050", "object"=>"session", "region"=>"Virginia", "browser"=>"Chrome 45.0.2454", "cou
 60605760782453 | 1444015898050 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "path"=>"/", "time"=>"1444015898050", "query"=>"?utm_source=badge", "title"=>"Heap | Mobile and Web Ana
 60605760782453 | 1444015928598 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "href"=>"/features", "path"=>"/", "time"=>"1444015928598", "type"=>"click", "query"=>"?utm_source=badge
 60605760782453 | 1444015928990 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "path"=>"/features/data-capture", "time"=>"1444015928990", "title"=>"Data Capture - Heap | Mobile and W
 60605760782453 | 1444015941595 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "href"=>"/features/users", "path"=>"/features/data-capture", "time"=>"1444015941595", "type"=>"click",
 60605760782453 | 1444015941828 | "ip"=>"72.83.222.84", "city"=>"Fairfax", "path"=>"/features/users", "time"=>"1444015941828", "title"=>"Users - Heap | Mobile and Web Analytics",
 47796230488402 | 1410696615406 | "path"=>"/", "time"=>"1410696615406", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 47796230488402 | 1410696615407 | "time"=>"1410696615407", "object"=>"session", "library"=>"web", "event_id"=>"8257717896", "referrer"=>"https://www.google.com/", "session_id"=>"
 47796230488402 | 1410696625798 | "href"=>"/login", "path"=>"/", "time"=>"1410696625798", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.co
 47796230488402 | 1410696627036 | "path"=>"/login", "time"=>"1410696627036", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pagevie
 47796230488402 | 1410696629203 | "path"=>"/", "time"=>"1410696629203", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 47796230488402 | 1410696631692 | "href"=>"/pricing", "path"=>"/", "time"=>"1410696631692", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.
 47796230488402 | 1410696633132 | "path"=>"/pricing", "time"=>"1410696633132", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pag
 47796230488402 | 1410696656800 | "href"=>"/login", "path"=>"/pricing", "time"=>"1410696656800", "type"=>"click", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>
 47796230488402 | 1410696657871 | "path"=>"/login", "time"=>"1410696657871", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pagevie
 47796230488402 | 1410696659768 | "path"=>"/pricing", "time"=>"1410696659768", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pag
 47796230488402 | 1410696664689 | "href"=>"/login", "path"=>"/pricing", "time"=>"1410696664689", "type"=>"click", "title"=>"Pricing - Heap | Mobile and Web Analytics", "domain"=>
 47796230488402 | 1410696665586 | "path"=>"/login", "time"=>"1410696665586", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pagevie
 47796230488402 | 1410697234572 | "path"=>"/login", "time"=>"1410697234572", "type"=>"click", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "
 47796230488402 | 1410697237689 | "path"=>"/login", "time"=>"1410697237689", "type"=>"change", "title"=>"Login - Heap | Mobile and Web Analytics", "value"=>"dev@funzing.com", "do
 47796230488402 | 1410697240736 | "path"=>"/login", "time"=>"1410697240736", "type"=>"click", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "
 47796230488402 | 1410697240737 | "path"=>"/login", "time"=>"1410697240737", "type"=>"submit", "title"=>"Login - Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com",
 47796230488402 | 1410697243200 | "path"=>"/login", "time"=>"1410697243200", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "lib
 47796230488402 | 1410697253951 | "path"=>"/login", "time"=>"1410697253951", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "library"
 47796230488402 | 1410697262779 | "path"=>"/login", "time"=>"1410697262779", "type"=>"click", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "library"
 47796230488402 | 1411506838887 | "time"=>"1411506838887", "object"=>"session", "library"=>"web", "event_id"=>"8607771125", "session_id"=>"2208562923", "session_time"=>"141150683
 47796230488402 | 1411506838887 | "path"=>"/", "time"=>"1411506838887", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 47796230488402 | 1411976074500 | "path"=>"/", "time"=>"1411976074500", "title"=>"Heap | Mobile and Web Analytics", "domain"=>"heapanalytics.com", "object"=>"pageview", "library"
 47796230488402 | 1411976074501 | "time"=>"1411976074501", "object"=>"session", "library"=>"web", "event_id"=>"8803770993", "session_id"=>"538285301", "session_time"=>"1411976074
```

Dropbox

https://www.dropbox.com/?heap-event-visualizer=on

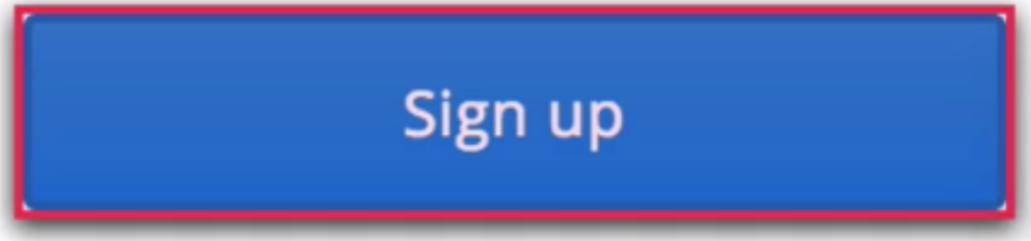
HEAP DEFINITION MODE NORMAL MODE EVENTS EXIT

Your stuff, anywhere

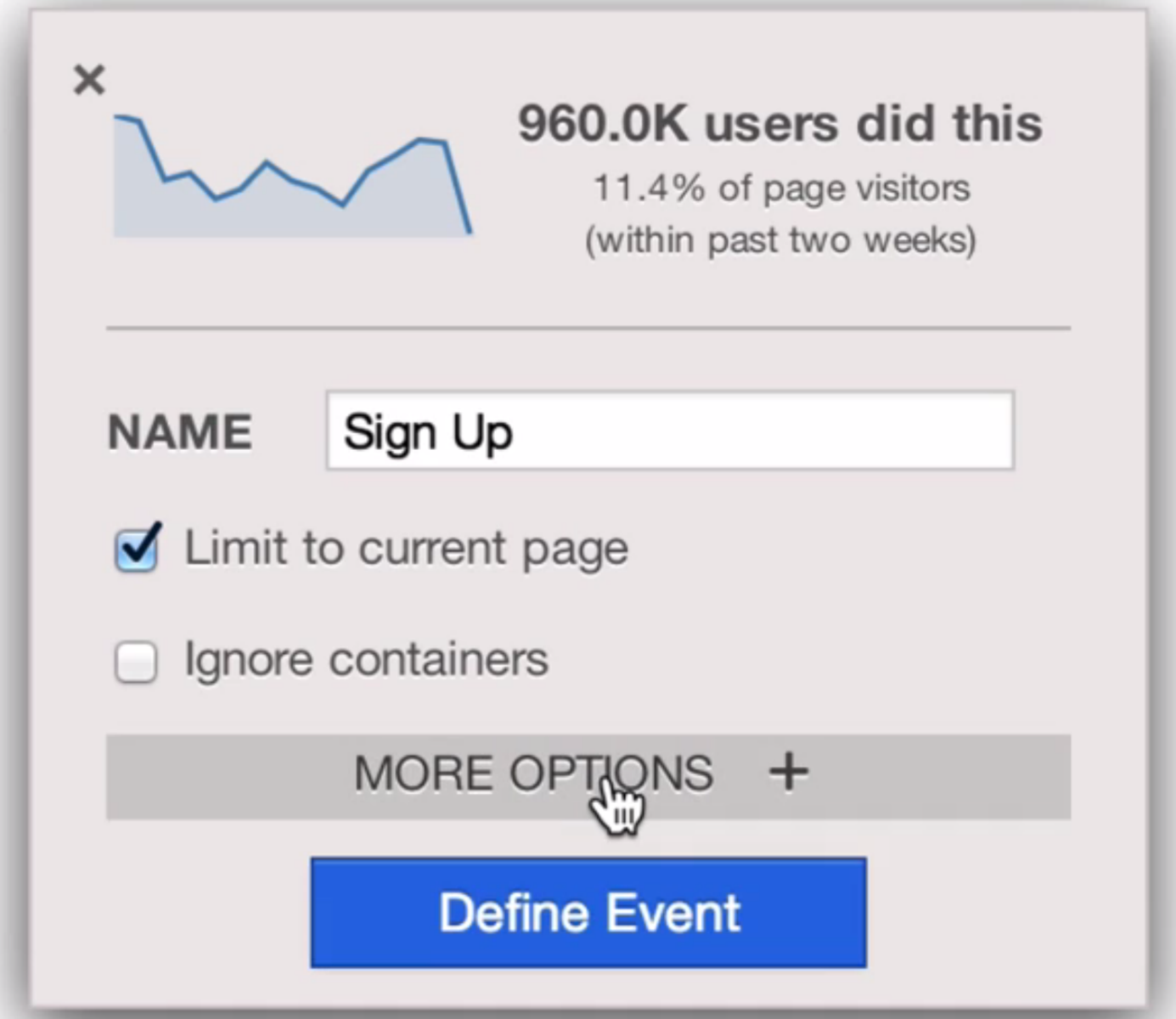
Sign up

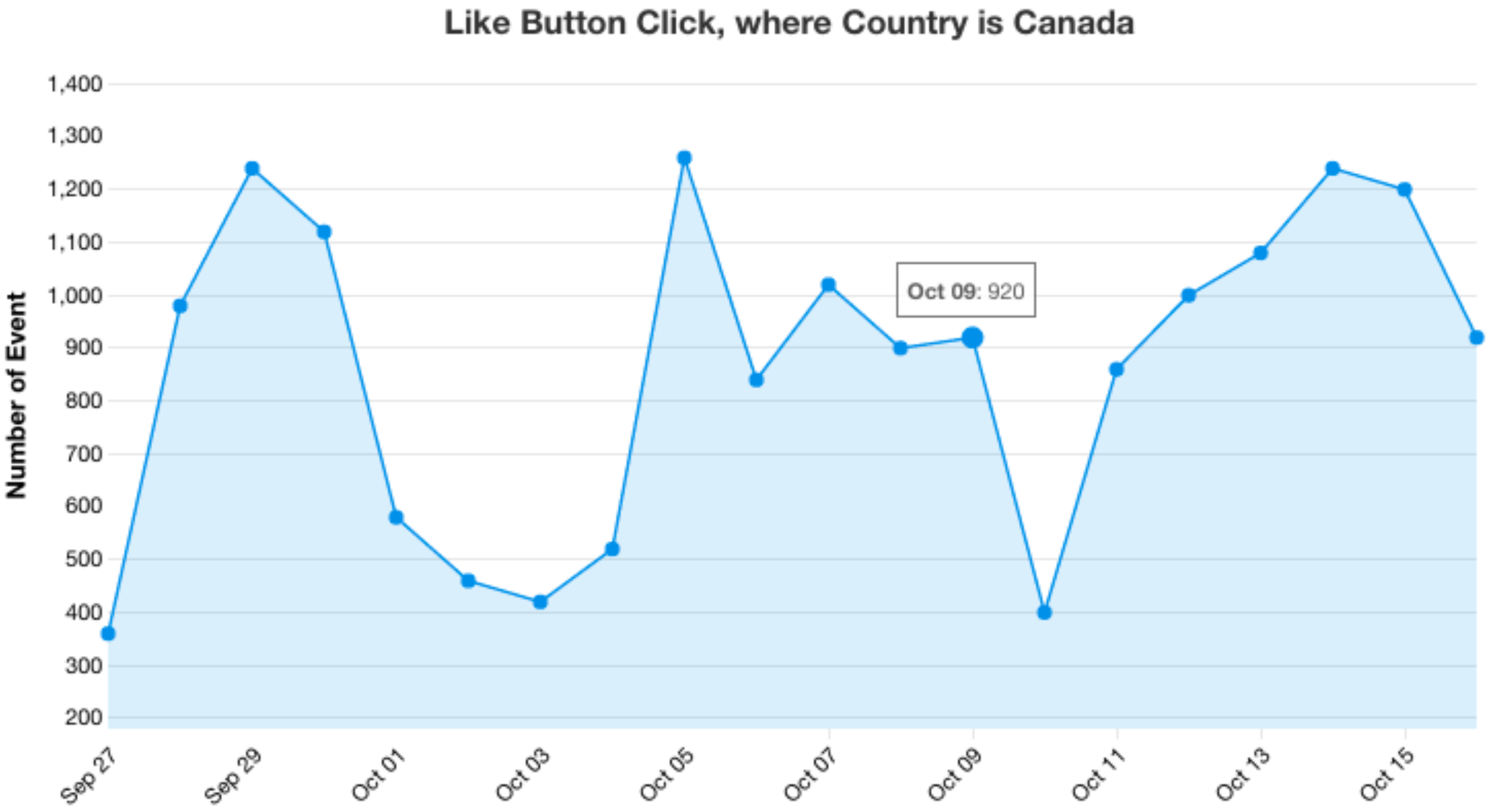
or Sign in

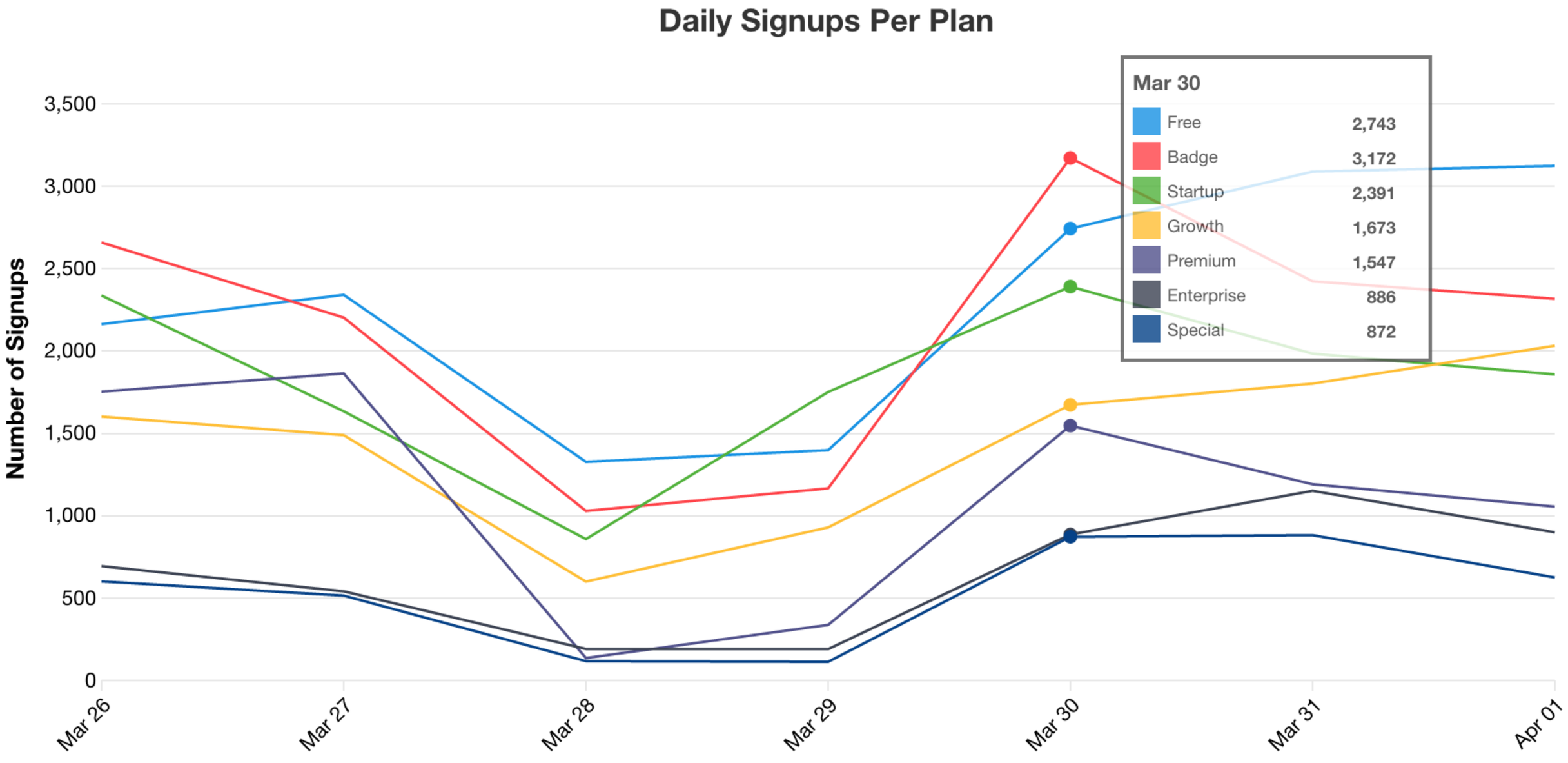
Learn more

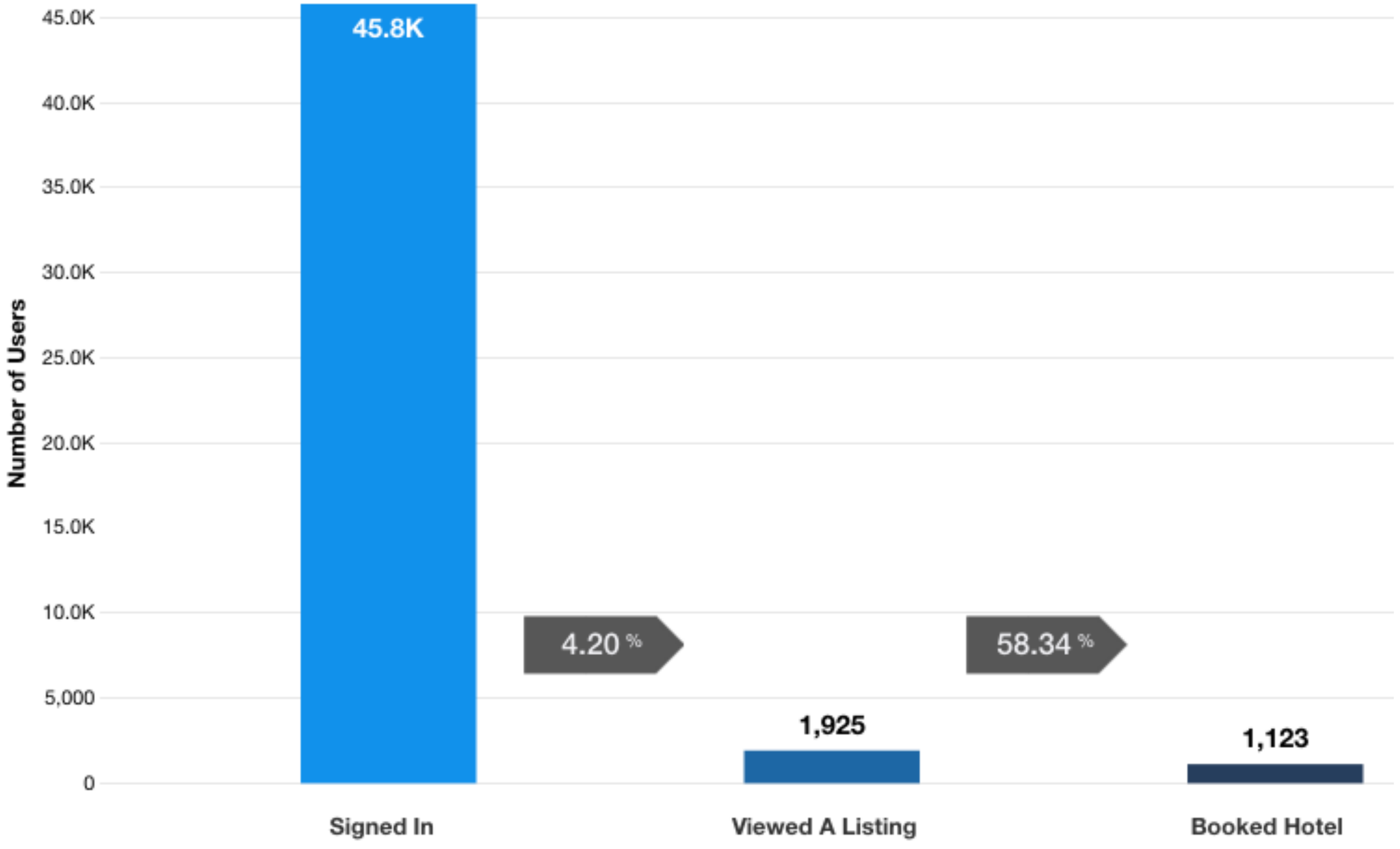

Sign up

or Sign in


960.0K users did this
11.4% of page visitors
(within past two weeks)
NAME
Sign Up
Limit to current page
Ignore containers
MORE OPTIONS +
Define Event


Like Button Click, where Country is Canada
Number of Event
1,400
1,300
1,200
1,100
1,000
900
800
700
600
500
400
300
200
Oct 09: 920
Sep 27
Sep 29
Oct 01
Oct 03
Oct 05
Oct 07
Oct 09
Oct 11
Oct 13
Oct 15


Daily Signups Per Plan
Number of Signups
3,500
3,000
2,500
2,000
1,500
1,000
500
0
Mar 26
Mar 27
Mar 28
Mar 29
Mar 30
Mar 31
Apr 01
Mar 30
Free 2,743
Badge 3,172
Startup 2,391
Growth 1,673
Premium 1,547
Enterprise 886
Special 872


45.0K
40.0K
35.0K
30.0K
25.0K
20.0K
15.0K
10.0K
5,000
0
Number of Users
45.8K
4.20 %
1,925
58.34 %
1,123
Signed In
Viewed A Listing
Booked Hotel

Date of Start Event	Users	1	2	3	4	
Aug 13 - Aug 17, 2014	60	23.33%	23.33%	6.67%	5.00%	13.
Aug 18 - Aug 24, 2014	102	23.53%	7.84%	9.80%	4.90%	4.9
Aug 25 - Aug 31, 2014	114	10.53%	7.02%	11.40%	9.65%	10.
Sep 1 - Sep 7, 2014	92	8.70%	6.52%	10.87%	8.70%	8.7
Sep 8 - Sep 14, 2014	82	7.32%	2.44%	10.98%	8.54%	8.5
Sep 15 - Sep 21, 2014	86	2.33%	2.33%	15.12%	12.79%	5.8
Sep 22 - Sep 28, 2014	103	6.80%	16.50%	9.71%	15.53%	12.
Sep 29 - Oct 5, 2014	92	18.48%	16.30%	7.61%	5.43%	6.5

Out of 114 users whose first Sign Up in the time range was between Aug 25 - Aug 31, 12 users did Login 1 week later.

SEGMENT NAME: Active Users

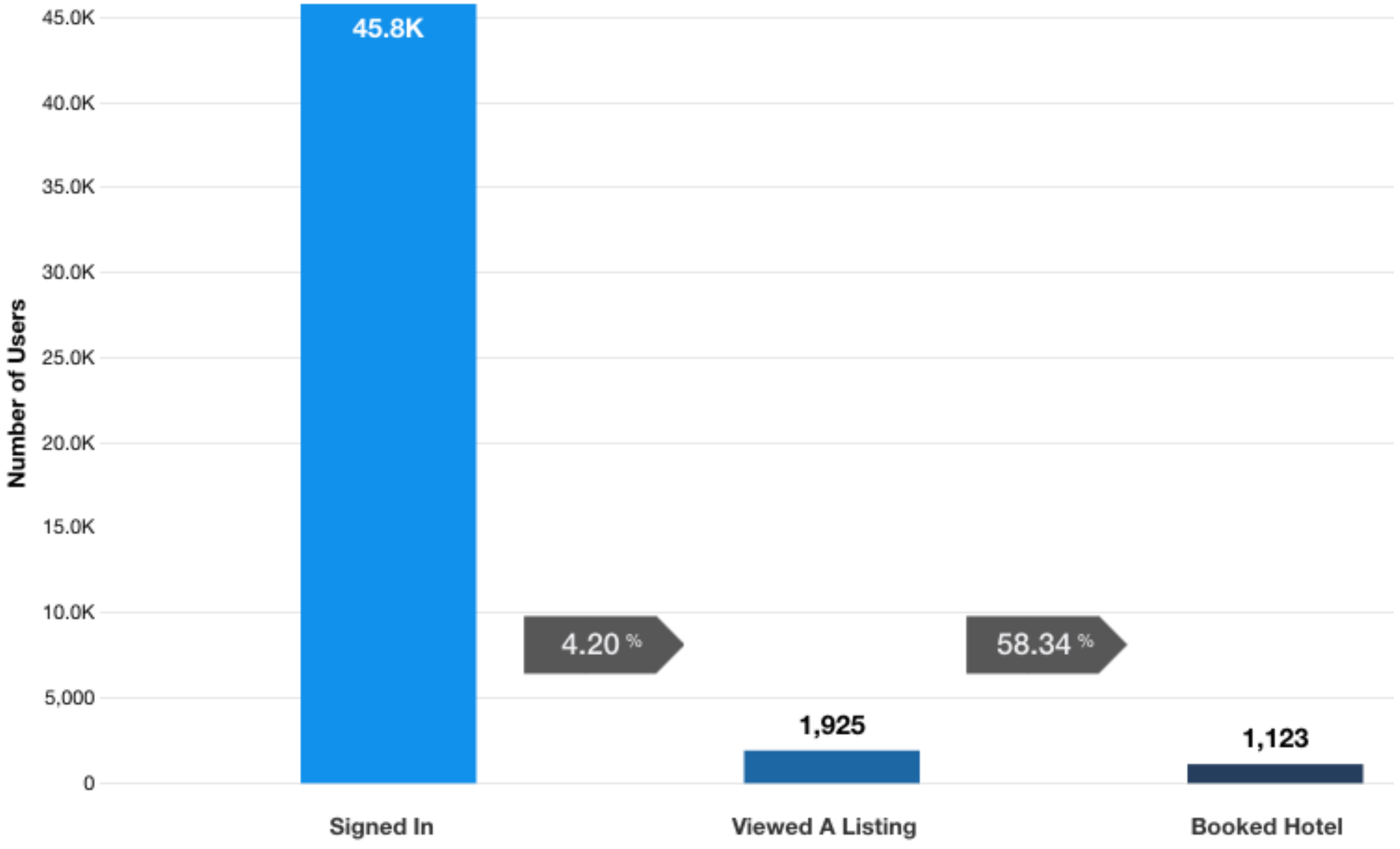
FILTER: Count of Sign In | greater than | 4 | in prior week

AND: Has done Friend Request | in prior week

or: Has done Accept Friend Request | in prior week

+ Filter

Define Segment


45.0K
40.0K
35.0K
30.0K
25.0K
20.0K
15.0K
10.0K
5,000
0
Number of Users
45.8K
4.20 %
1,925
58.34 %
1,123
Signed In
Viewed A Listing
Booked Hotel

Challenges

1. Capturing 10x to 100x as much data.
   Will never want 95% of it.

2. Funnels, retention, behavioral cohorts,
   grouping, filtering... can't pre-aggregate.

Challenges

1. Capturing 10x to 100x as much data.
   Will never want 95% of it.

2. Funnels, retention, behavioral cohorts,
   grouping, filtering... can't pre-aggregate.

3. Within a few minutes of real-time.

Data Scale

- Total dataset is ~60TB on disk and growing fast.
- Includes 80 billion events across 2 billion users.
- Of those events, 2.4 billion in the last week.
- **Too big to scale up, need to scale out.**

```sql
-- Distributed by (customer_id, user_id)
-- with the same shard boundaries.
CREATE TABLE users (
  customer_id BIGINT,
  user_id BIGINT,
  handle TEXT,
  properties JSONB NOT NULL DEFAULT '{}',
  PRIMARY KEY (customer_id, user_id)
);
```

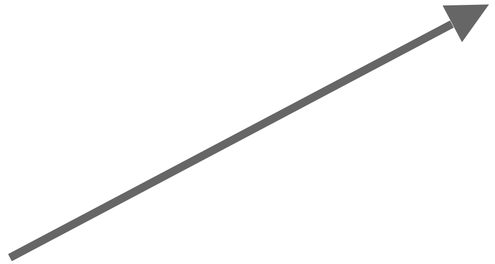
```sql
-- Distributed by (customer_id, user_id)
-- with the same shard boundaries.
CREATE TABLE users (
  customer_id BIGINT,
  user_id BIGINT,
  handle TEXT,
  properties JSONB NOT NULL DEFAULT '{}',
  PRIMARY KEY (customer_id, user_id)
);
```

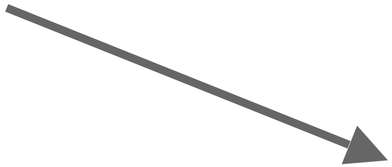
```sql
-- Distributed by (customer_id, user_id)
-- with the same shard boundaries.
CREATE TABLE events (
  customer_id BIGINT,
  user_id BIGINT,
  event_id BIGINT,
  time BIGINT NOT NULL,
  data JSONB NOT NULL,
  PRIMARY KEY (customer_id, user_id, event_id),
  FOREIGN KEY (customer_id, user_id)
    REFERENCES users (customer_id, user_id)
);
```

```
-- Distributed by (customer_id, user_id)
-- with the same shard boundaries.
CREATE TABLE users (
  customer_id BIGINT,  <----------------------
  user_id BIGINT,  <-------------------------
  handle TEXT,
  properties JSONB NOT NULL DEFAULT '{}',
  PRIMARY KEY (customer_id, user_id)
);
```

```
-- Distributed by (customer_id, user_id)
-- with the same shard boundaries.
CREATE TABLE events (
  customer_id BIGINT,
  user_id BIGINT,
  event_id BIGINT,
  time BIGINT NOT NULL,
  data JSONB NOT NULL,
  PRIMARY KEY (customer_id, user_id, event_id),
  FOREIGN KEY (customer_id, user_id)
    REFERENCES users (customer_id, user_id)
);
```

customer_id	user_id	handle TEXT	properties JSONB
123	102756	jane_123	{email: 'jane_123@mail.com', 'ab_test_grp':'A'}
123	300732		{ab_test_grp: 'B'}
678	368868		
499	628537	steve_is_cool	{utm_campaign: 'twitter'}





users

customer_id	user_id	handle TEXT	properties JSONB

users_001

customer_id	user_id	handle TEXT	properties JSONB
123	102756	jane_123	{email: 'jane_123@mail.com', 'ab_test_grp':'A'}
123	300732		{ab_test_grp: 'B'}
678	628537		
499	368868	steve_is_cool	{utm_campaign: 'twitter'}

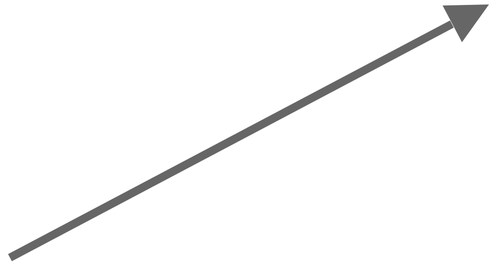
...

...

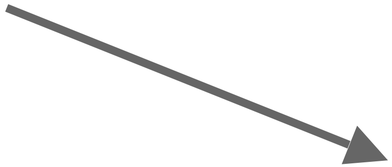
users_002

customer_id	user_id	handle TEXT	properties JSONB
756	257186	...	...
756	120554	...	...

users



```
SELECT COUNT(*)
FROM users
WHERE customer_id = 123
GROUP BY properties ->> 'ab_test_grp'
```



users_001

customer_id	user_id	handle TEXT	properties JSONB
123	102756	jane_123	{email: 'jane_123@mail.com', 'ab_test_grp':'A'}
123	300732		{ab_test_grp: 'B'}
678	628537		
499	368868	steve_is_cool	{utm_campaign: 'twitter'}

...

...

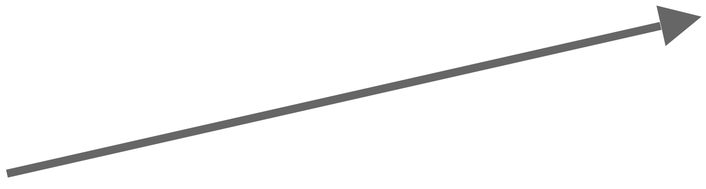
users_002

customer_id	user_id	handle TEXT	properties JSONB
756	257186	...	...
756	120554	...	...

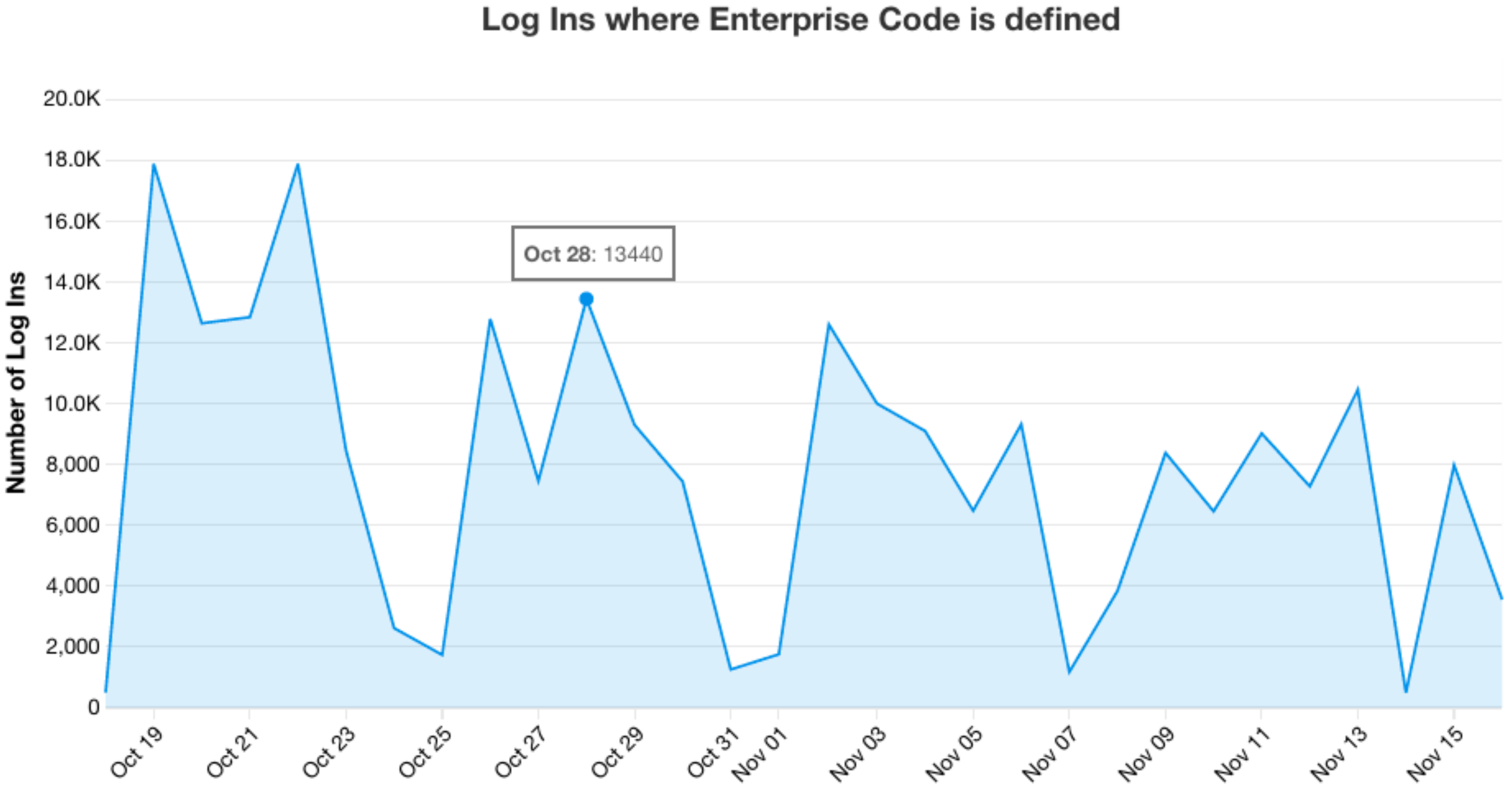
users_001

```
SELECT COUNT(*)
FROM users_001
WHERE customer_id = 123
GROUP BY properties ->> 'ab_test_grp'
```

users



```
SELECT COUNT(*)
FROM users
WHERE customer_id = 123
GROUP BY properties ->> 'ab_test_grp'
```


Log Ins where Enterprise Code is defined
Number of Log Ins
20.0K
18.0K
16.0K
14.0K
12.0K
10.0K
8,000
6,000
4,000
2,000
0
Oct 28: 13440
Oct 19
Oct 21
Oct 23
Oct 25
Oct 27
Oct 29
Oct 31
Nov 01
Nov 03
Nov 05
Nov 07
Nov 09
Nov 11
Nov 13
Nov 15

```
SELECT user_id, time, data
FROM events
WHERE
  customer_id = 135 AND
  time BETWEEN 1424437200000 AND 1429531200000 AND
  (data ->> 'type') = 'Log In'
```

```
SELECT id
FROM users
WHERE
  customer_id = 135 AND
  (properties ->> 'Enterprise Code') IS NOT NULL
```

```
SELECT
  COUNT(*) AS value


FROM (
  SELECT user_id, time, data
  FROM events
  WHERE
    customer_id = 135 AND
    time BETWEEN 1424437200000 AND 1429531200000 AND
    (data ->> 'type') = 'Log In'
) event_query

INNER JOIN (
  SELECT id
  FROM users
  WHERE
    customer_id = 135 AND
    (properties ->> 'Enterprise Code') IS NOT NULL
) user_filter_query ON (event_query.user_id = user_filter_query.user_id)
```

```
SELECT
  COUNT(*) AS value,
  date_trunc('day', to_timestamp(time / 1000) AT TIME ZONE 'UTC') AS time_bucket

FROM (
  SELECT user_id, time, data
  FROM events
  WHERE
    customer_id = 135 AND
    time BETWEEN 1424437200000 AND 1429531200000 AND
    (data ->> 'type') = 'Log In'
) event_query

INNER JOIN (
  SELECT id
  FROM users
  WHERE
    customer_id = 135 AND
    (properties ->> 'Enterprise Code') IS NOT NULL
) user_filter_query ON (event_query.user_id = user_filter_query.user_id)

GROUP BY time_bucket
```

Distributing With CitusDB

- Every customer gets a `customer_id`, every end user gets a `user_id`.
- One big table of `users`, one big table of `events`.
- Shard both tables by `(customer_id, user_id)`.
- CitusDB turns a vanilla PostgreSQL query, potentially with joins, into many local joins.

```
SELECT
  COUNT(*) AS value,
  date_trunc('month', to_timestamp(time / 1000) AT TIME ZONE 'UTC') AS time_bucket

FROM (
  SELECT user_id, time, data
  FROM events
  WHERE
    customer_id = 135 AND
    time BETWEEN 1424437200000 AND 1429531200000 AND
    (data ->> 'type') = 'Log In'
) event_query

INNER JOIN (
  SELECT id
  FROM users
  WHERE
    customer_id = 135 AND
    (properties ->> 'Enterprise Code') IS NOT NULL
) user_filter_query ON (event_query.user_id = user_filter_query.user_id)

GROUP BY time_bucket
```

```
SELECT
  COUNT(*) AS value,
  date_trunc('month', to_timestamp(time / 1000) AT TIME ZONE 'UTC') AS time_bucket

FROM (
  SELECT user_id, time, data
  FROM events
  WHERE
    customer_id = 135 AND
    time BETWEEN 1424437200000 AND 1429531200000 AND
    (data ->> 'path') = '/checkout' AND
    (data ->> 'action') = 'click' AND
    (data ->> 'css_hierarchy') LIKE '%div.checkout_modal%a.btn' AND
    (data ->> 'target_text') = 'Confirm Order'
) event_query

GROUP BY time_bucket
```

```
CREATE INDEX confirmed_checkout_idx ON events (time)
  WHERE
    (data ->> 'path') = '/checkout' AND
    (data ->> 'action') = 'click' AND
    (data ->> 'css_hierarchy') LIKE '%div.checkout_modal%a.btn' AND
    (data ->> 'target_text') = 'Confirm Order'
```

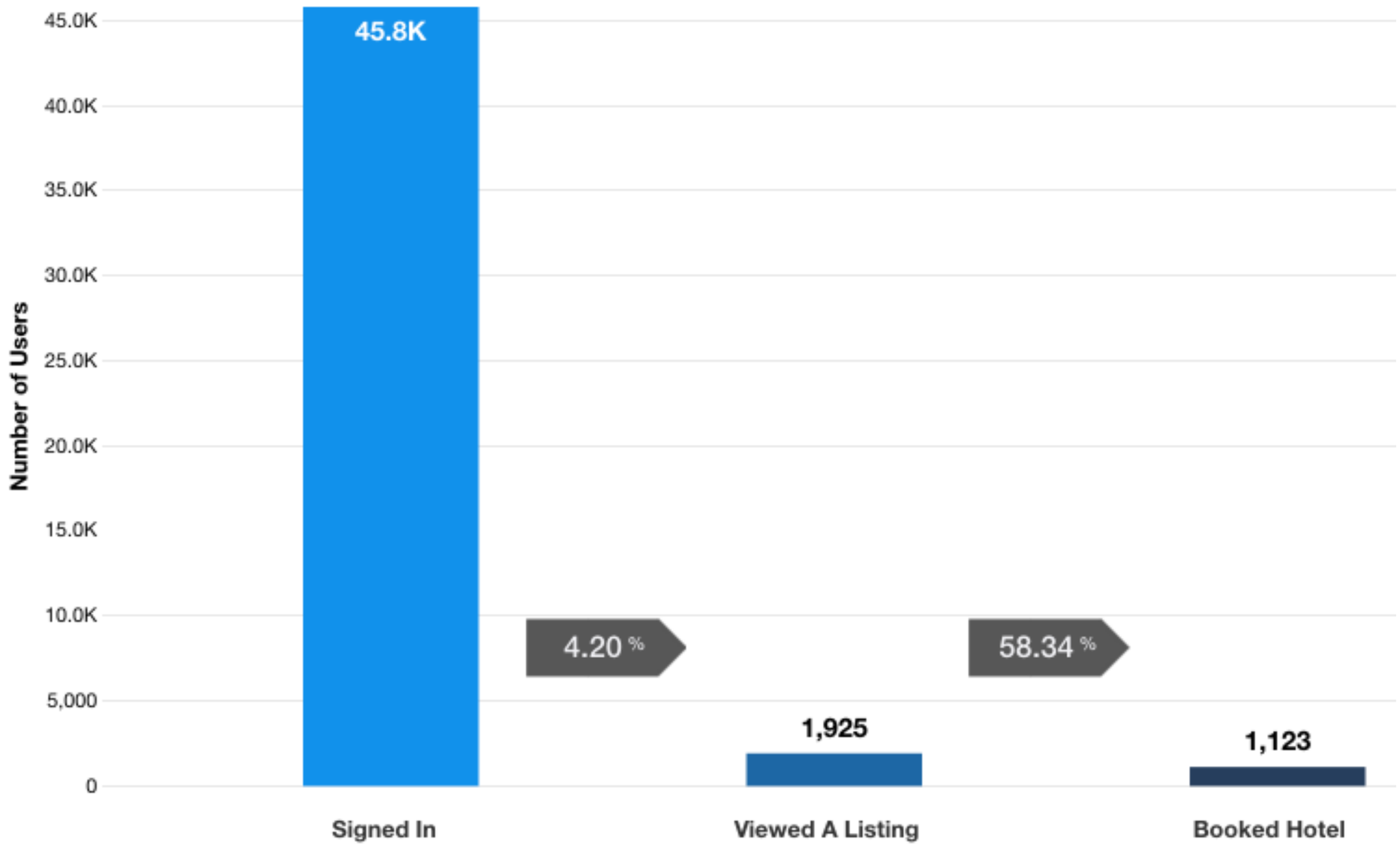
```
CREATE INDEX confirmed_checkout_idx ON events (time)
  WHERE
    (data ->> 'path') = '/checkout' AND
    (data ->> 'action') = 'click' AND
    (data ->> 'css_hierarchy') LIKE '%div.checkout_modal%a.btn' AND
    (data ->> 'target_text') = 'Confirm Order'

...

SELECT
  COUNT(*) AS value,
  date_trunc('month', to_timestamp(time / 1000) AT TIME ZONE 'UTC') AS time_bucket
FROM events
WHERE
  customer_id = 135 AND
  time BETWEEN 1424437200000 AND 1429531200000 AND
  (data ->> 'path') = '/checkout' AND
  (data ->> 'action') = 'click' AND
  (data ->> 'css_hierarchy') LIKE '%div.checkout_modal%a.btn' AND
  (data ->> 'target_text') = 'Confirm Order'
GROUP BY time_bucket
```

Partial Index Strategy

- Every event definition is a filter on the `events` table.
- Under the hood, Heap maintains one partial index for each of those filters.
- The variety of events that Heap captures is massive, so any individual event definition is very selective.
- Fits perfectly into our "retroactive" analytics framework.


45.0K
40.0K
35.0K
30.0K
25.0K
20.0K
15.0K
10.0K
5,000
0
Number of Users
45.8K
4.20 %
1,925
58.34 %
1,123
Signed In
Viewed A Listing
Booked Hotel
'1'
'2'
'3'

```
funnel_events(events INT[], num_steps INT) RETURNS INT[]
-- Returns an array of size num_steps, with 1s corresponding to
-- steps completed in the funnel, 0s in the other positions.

> SELECT funnel_events('{1, 2, 3}', 3);
{1, 1, 1}

> SELECT funnel_events('{1, 3, 2, 2, 2}', 3);
{1, 1, 0}

> SELECT funnel_events('{1}', 3);
{1, 0, 0}
```

```
SELECT array_agg(event ORDER BY time) AS events  ----------------> {1, 1, 2}
FROM (                                                             {1, 1, 1, 1}
  (                                                                {1}
    SELECT customer_id, user_id, 1 AS event, time
    FROM events
    WHERE
     customer_id = 135 AND
     time BETWEEN 1412319600000 AND 1412924400000 AND
     (data ->> 'action') = 'view_page' AND
     (data ->> 'path') = '/item_detail'
  )
  UNION
  (
    SELECT customer_id, user_id, 2 AS event, time
    FROM events
    WHERE
     customer_id = 135 AND
     time BETWEEN 1412319600000 AND 1412924400000 AND
     (data ->> 'action') = 'click' AND
     (data ->> 'css_hierarchy') LIKE '%div.checkout_modal%a.btn' AND
     (data ->> 'target_text') = 'Confirm Order'
  )
) t
GROUP BY customer_id, user_id
```

```
SELECT sum(funnel_events(events, 2)) AS funnel_results  ----------> {3, 1}
FROM (
  SELECT array_agg(event ORDER BY time) AS events  ----------------> {1, 1, 2}
  FROM (                                                            {1, 1, 1, 1}
    (                                                               {1}
      SELECT customer_id, user_id, 1 AS event, time
      FROM events
      WHERE
       customer_id = 135 AND
       time BETWEEN 1412319600000 AND 1412924400000 AND
       (data ->> 'action') = 'view_page' AND
       (data ->> 'path') = '/item_detail'
    )
    UNION
    (
      SELECT customer_id, user_id, 2 AS event, time
      FROM events
      WHERE
       customer_id = 135 AND
       time BETWEEN 1412319600000 AND 1412924400000 AND
       (data ->> 'action') = 'click' AND
       (data ->> 'css_hierarchy') LIKE '%div.checkout_modal%a.btn' AND
       (data ->> 'target_text') = 'Confirm Order'
    )
  ) t
  GROUP BY customer_id, user_id
) t
```

UDFs For Advanced Analysis

- All analyses shard cleanly by `(customer_id, user_id)`, and every query is built from a sparse set of events.
- Simple meta-formula for a family of analysis queries:
  1. Build up an array of relevant events for each user
  2. Pass the array to a custom UDF
  3. Join arbitrarily for more filtering, grouping, etc

Our PostgreSQL Wishlist

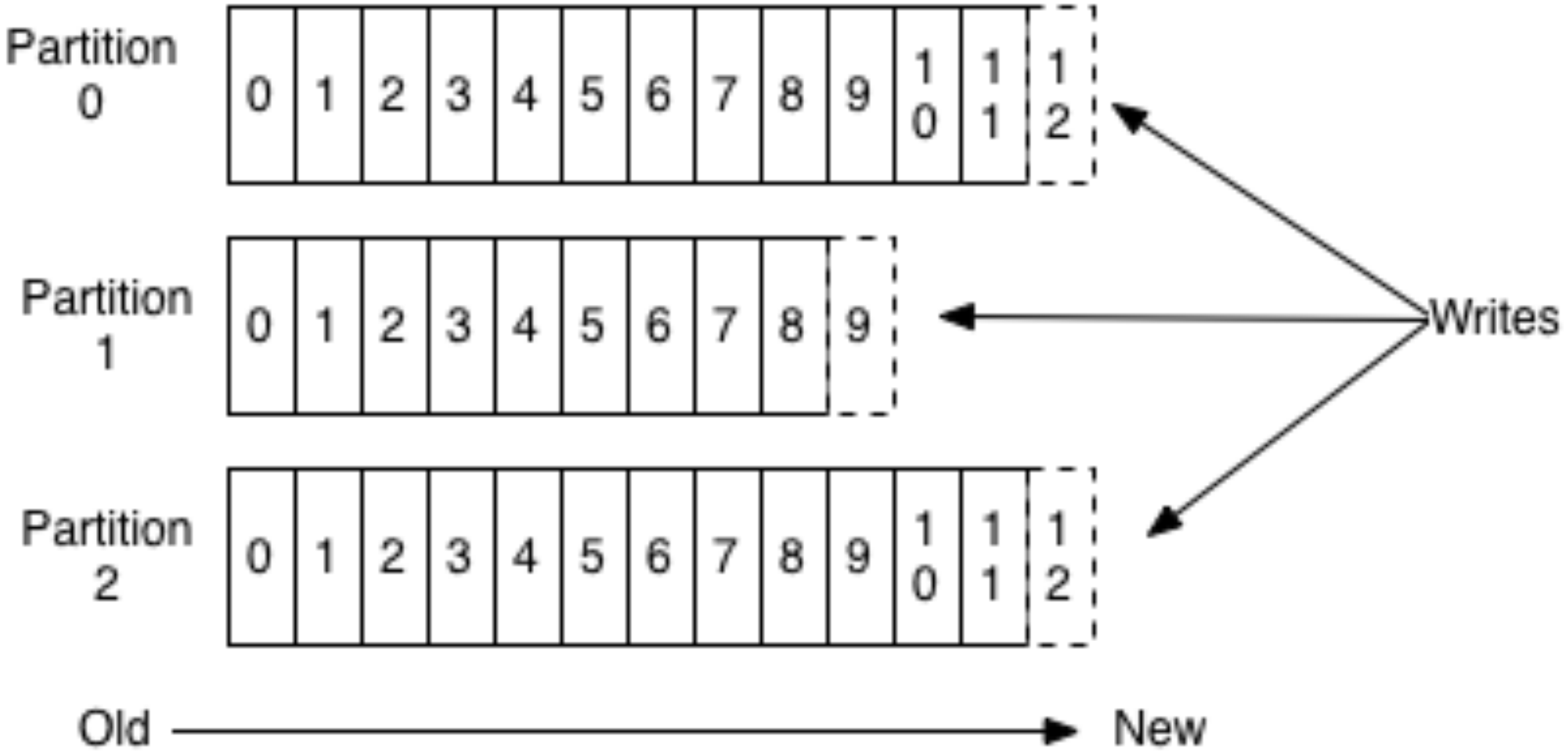
- Partial index creations using base indexes.
- Concurrent `CREATE INDEX CONCURRENTLY` calls.
- Ability to tell the query planner what we want.

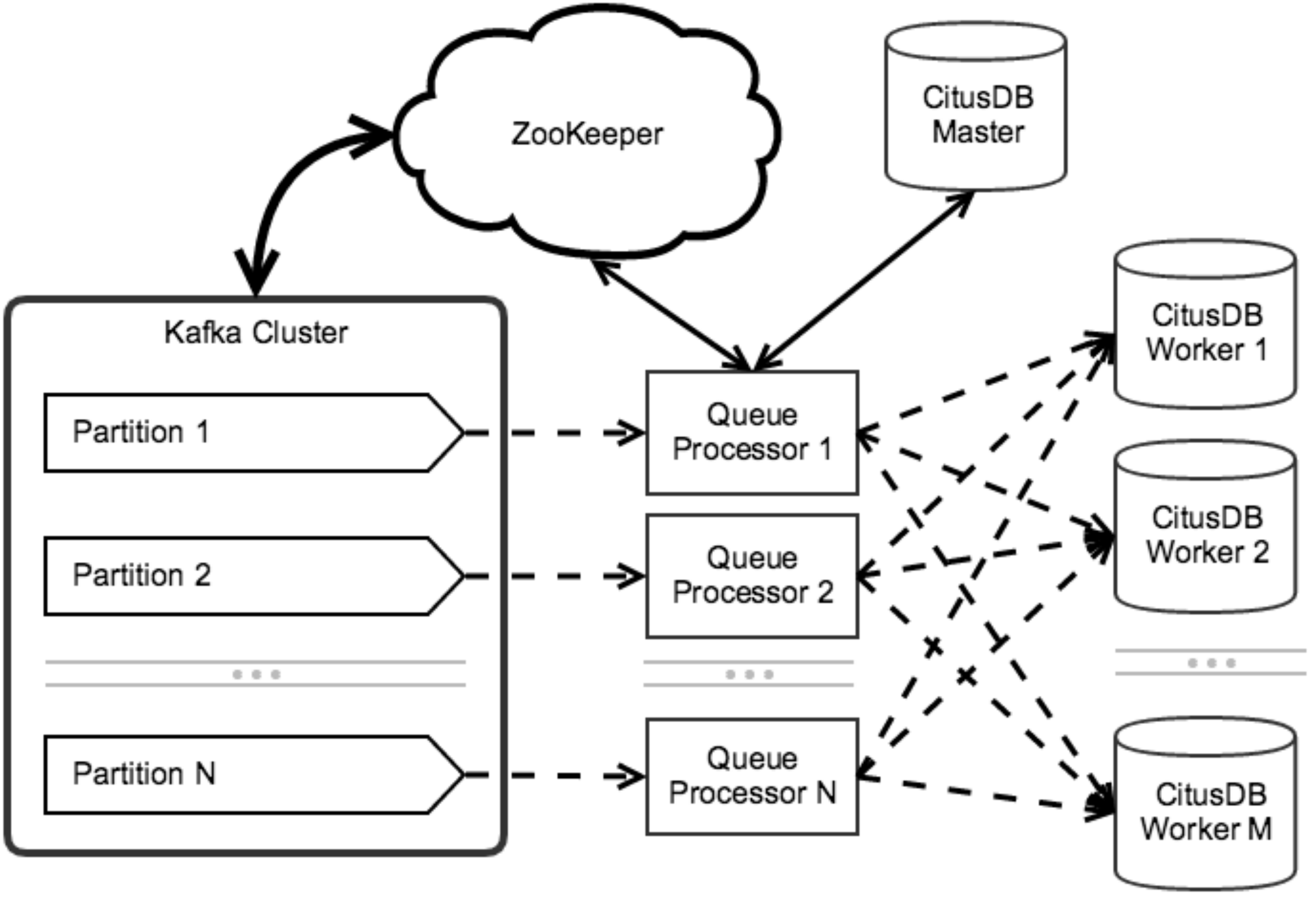
Building A Real System

- We're sharding by user, not by time range. How do we move shards, split shards, rehydrate new replicas, etc?
- Where does data live before it gets into the CitusDB cluster?
- How do we handle ingestion spikes?

Building A Real System

- Use Kafka as a short-term commit log.
- Use PL/pgSQL to turn writes into idempotent, commutative messages.
- Keep track of Kafka positions and replay data for cluster operations.


Partition 0
0 1 2 3 4 5 6 7 8 9 10 11 12
Partition 1
0 1 2 3 4 5 6 7 8 9
Partition 2
0 1 2 3 4 5 6 7 8 9 10 11 12
Writes
Old
New


ZooKeeper
CitusDB
Master
Kafka Cluster
Partition 1
Partition 2
Partition N
Queue
Processor 1
Queue
Processor 2
Queue
Processor N
CitusDB
Worker 1
CitusDB
Worker 2
CitusDB
Worker M

Future Work

- Strong majority of our queries touch only the last 2 weeks of data – can we split out recent data onto nicer hardware?
- Numerical analysis beyonds counts -- min, max, averages, histograms.
- Richer analysis, path analysis, more behavioral cohorting, data pivoting...

Questions?

Or, ask me on twitter: @danlovesproofs