



🍷 PITCH

Groups, clarified

 Ryan Singer · Feb 18, 2019


Last November I wrote a pitch for adding Groups ala BCX to BC3: [💡 Groups! - Product Strategy](#).

After looking at the pitch,  Jason pointed out it wasn't entirely clear where to expect Groups to appear. Example: Suppose you could autocomplete a Group when you @mention somebody. Does that set an expectation you should see Groups in other places where you autocomplete people, like when assigning To-Dos?

We could answer "yes" and expose Groups everywhere we can think of. But that increases the scope. And it's a bad product design smell to add functionality for "completeness" sake. It's better to understand use cases and target them.

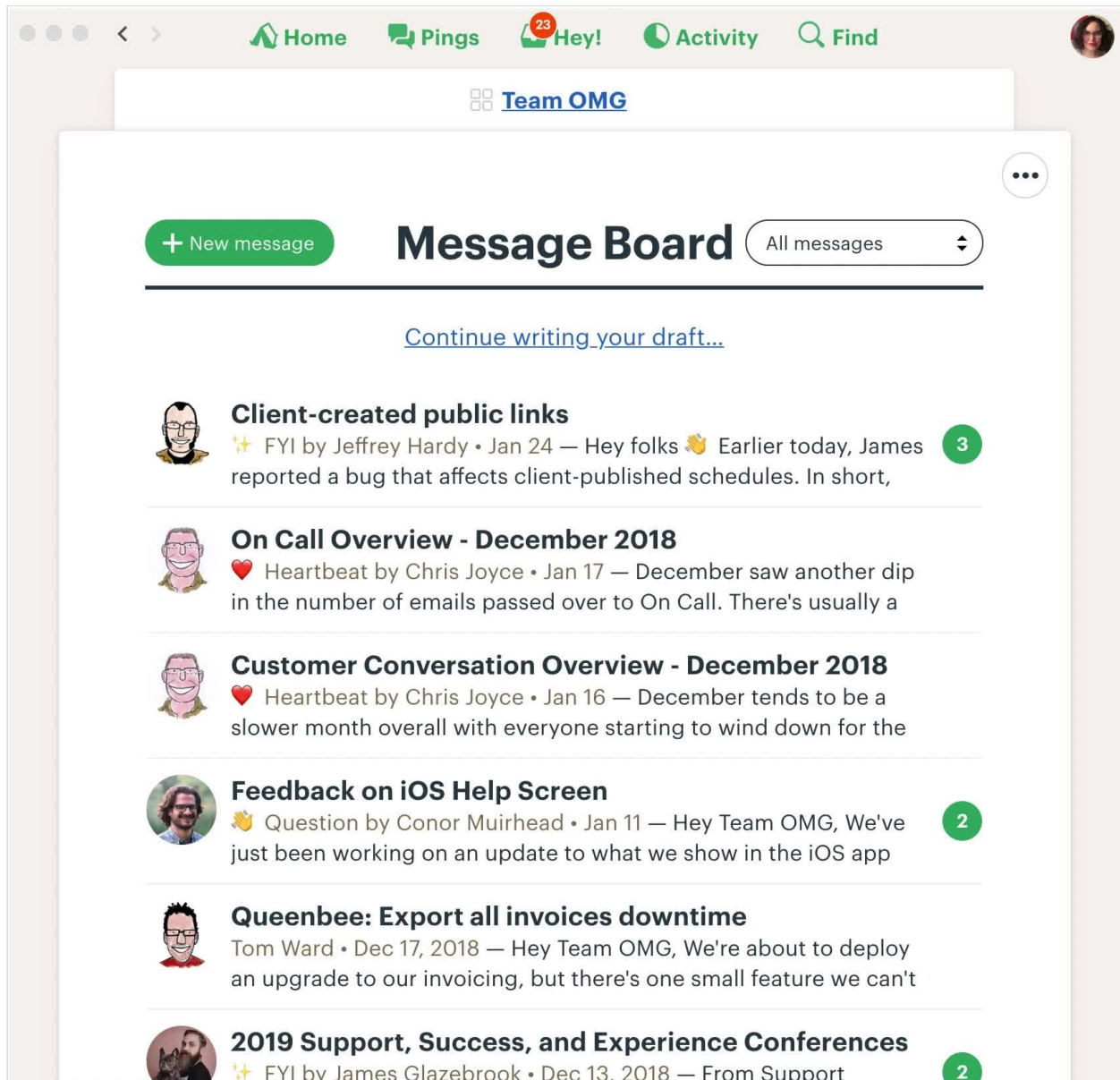
With our last BC3 cycle approaching, I wanted to take one last swing at this. Here's a narrower use case and a narrower solution that I think sets clear expectations about where and how Groups should appear in the app.

Narrowing down a problem case

To find a more specific use case to target, I turned to  Kristin for perspective. She and others in support have wanted Groups for a long time.

I asked Kristin to demonstrate workflows she's already doing today that are painful without Groups. The aim was to see some concrete "before" cases to design against.

She had two really good ones. Both of them involve **narrowing down** ~50 people to 15. Here's a video of Kristin hunting around to check off support people when posting a Message:



Narrowing recipients down when posting a message

The same problem comes up when choosing who an event is "with." Here she has to pick 15 people one by one using the autocomplete UI:

The screenshot shows the Hey! app interface. At the top, there's a navigation bar with icons for Home, Pings, Hey! (with a red notification badge), Activity, and Find. Below this is a header for 'Team OMG'. The main content area is titled 'Schedule' and features a '+ New event' button. A link to add the schedule to Google Calendar, Outlook, or iCal is provided. The calendar itself shows January and February. January 25th is highlighted in green. Below the calendar, there are event details for January 25th, including 'Crisis Drills: 2019: Winter', 'CC Out - Paternity Leave', 'SD Out', and 'TG Holiday'.

January							February						
SUN	MON	TUE	WED	THU	FRI	SAT	SUN	MON	TUE	WED	THU	FRI	SAT
		1	2	3	4	5						1	2
6	7	8	9	10	11	12	3	4	5	6	7	8	9
13	14	15	16	17	18	19	10	11	12	13	14	15	16
20	21	22	23	24	25	26	17	18	19	20	21	22	23
27	28	29	30	31			24	25	26	27	28		

Fri, Jan 25

January 1

Jan 1 - 31
Crisis Drills: 2019: Winter

CC Out - Paternity Leave

Dec 24 - Feb 1

SD Out

Jan 23 - 25

TG Holiday 3

Kristin selects people one by one to participate in an event

Watching these videos, you see a palpable pain. With this as the "before", it's clear how Groups could offer a better "after" in this use case.

@Mentions: Cool, but maybe not calm enough

After this, I asked: can we find a similar pain point involving @mentions? We couldn't come up with one as compelling.

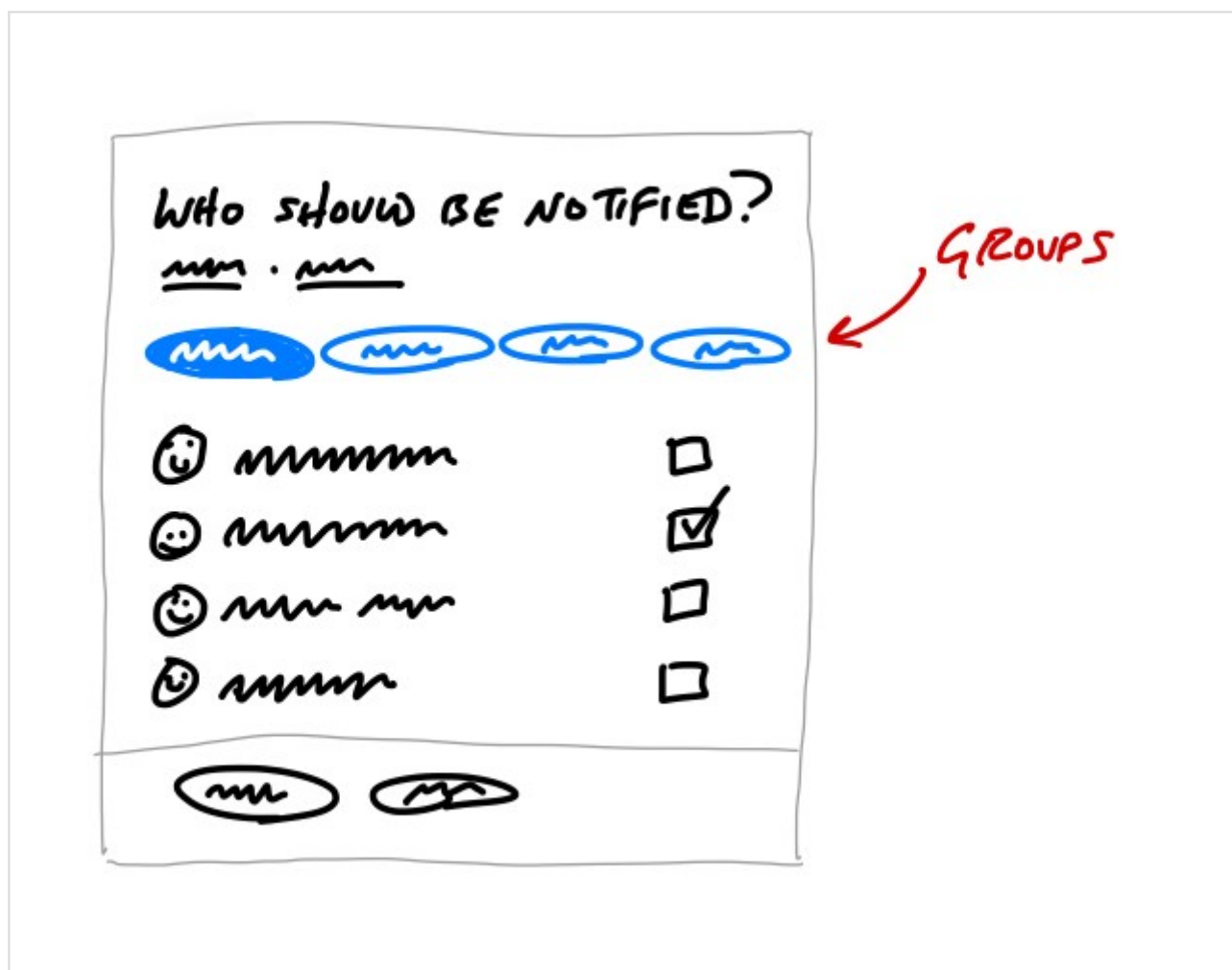
It's easy to think of cases where mentioning everybody via a Group could be handy. But is that good for our Hey menus or how we communicate with each other? With 15 people behind a

single @mention, it would be easy to spam each other without thinking too hard about it.

With that in mind, I looked for broad design solutions for those two use cases only.

Solution for choosing Message subscribers

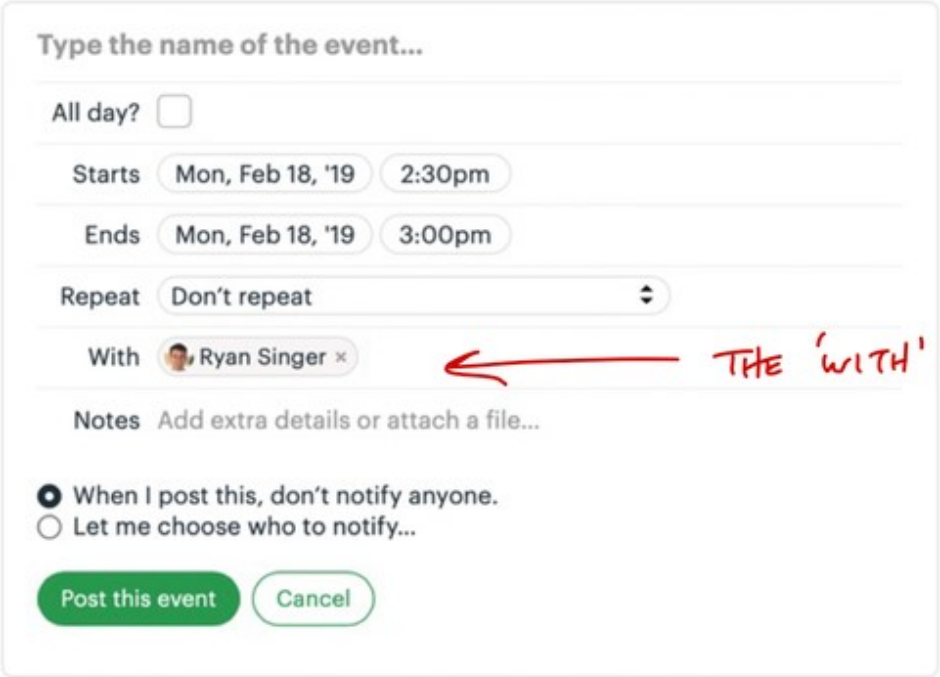
We could use a BCX-style approach to solve the first case. We have a single modal today for setting Subscribers app-wide, whether you are posting a new Message or changing who gets notified about some commentable. That "Who should be notified?" modal could offer Groups at the top. Clicking them checks members of the Groups below.



Groups on the Who Should Be Notified? modal

Do Events get a different solution?

Solving the Events "with" field case is not as straightforward. The current UI is a single autocomplete field.



The screenshot shows the Basecamp event creation interface. At the top is a text input field labeled "Type the name of the event...". Below it is an "All day?" checkbox. The "Starts" section shows "Mon, Feb 18, '19" and "2:30pm". The "Ends" section shows "Mon, Feb 18, '19" and "3:00pm". The "Repeat" dropdown is set to "Don't repeat". The "With" field contains a tag for "Ryan Singer" with a close button. A red arrow points from the handwritten text "THE 'WITH' FIELD" to this field. Below the "With" field is a "Notes" section with the placeholder "Add extra details or attach a file...". At the bottom are two radio buttons: "When I post this, don't notify anyone." (selected) and "Let me choose who to notify...". At the very bottom are "Post this event" and "Cancel" buttons.

Today's "With" field

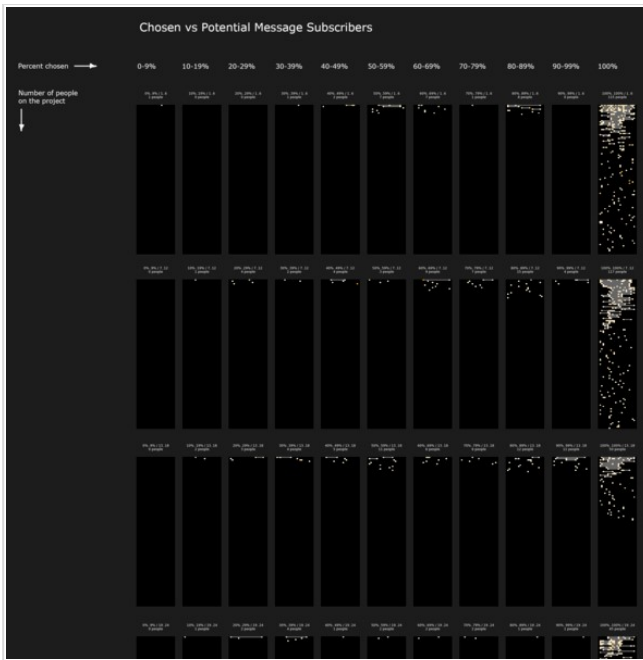
We could offer Groups inside that autocomplete, but a whole bunch of problems follow. We don't want to set the precedent that a Group might appear inside any autocomplete, per JF's concerns in the intro above. And if we did ... there are still UX problems. Suppose you chose a Group in the autocomplete. Would the Group name appear then in the With field? That'd be misleading because maybe only 2 people from the Group are on the project. We could insert the *people* from the Group into the autocomplete field instead of the Group... but then it's hard to see which Groups you already added. So autocompleteable Groups are out.

Next idea: What if we replaced the autocomplete with a modal picker ala "Who should be notified"? Then we could re-use the same solution as messages.

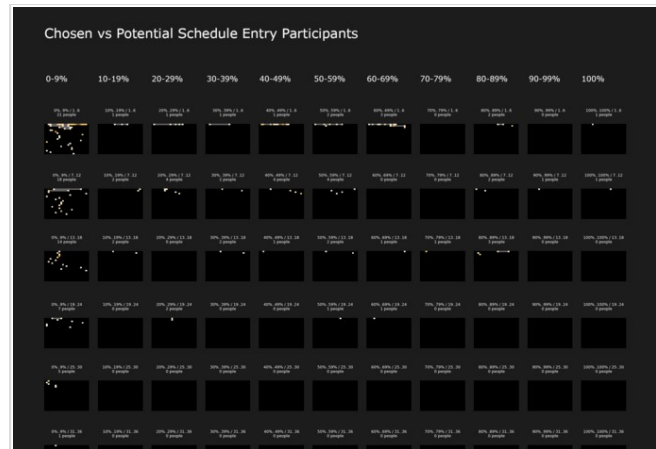
Technically that could work. Some "Choose participants..." button would launch the modal and then populate the field.

However, this could be a UX step backwards from today. Suppose the vast majority of people are autocompleting one or two names. That's a very different behavior from choosing Message subscribers, with everyone checked by default. Depending on the number of people on the project, you might have to scroll through a few screens in the modal to check the one other name you want. It's easier to autocomplete.

I took a look at the data for perspective. Indeed, the majority of Events have one or two participants. And the pattern is the opposite for Message subscribers. Here's a peek (explaining the analysis would take another write-up).



Message subscribers skew toward notifying everyone



Event participants skew the other way

The data also showed that our use case — choosing 14 people in the With field — is extremely rare. So we don't want to rethink the UI just for us on this one.

Workaround: This is only about Subscribers

I came back to Kristin with a proposal: if she had to stop using the "With" field and instead relied on the "who should be notified" featured, would that still work for her when she posts events?

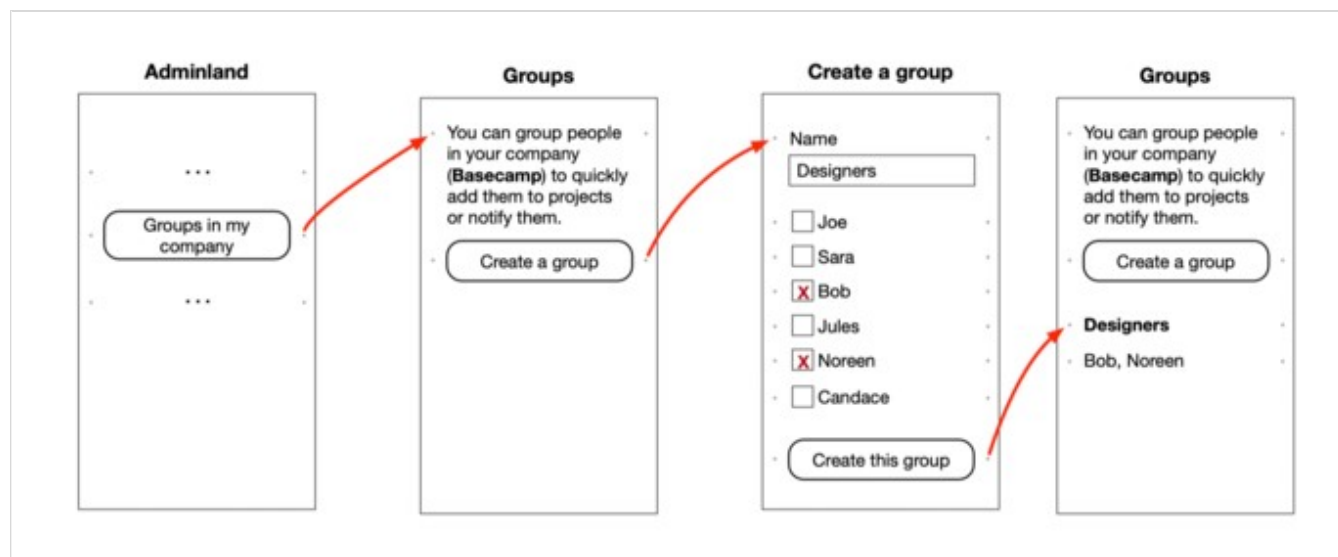
She said it would be fine for us. For others, it could conceivably be a problem because setting Subscribers vs. Participants isn't the same — Participants get a calendar integration that Subscriber don't. But in our case — and we're the outliers here — it's not a problem.

This suggests we can confine the Groups feature entirely to "Who should be notified?" — the Subscribers modal.

Proposed solution

I think we should do this for the final BC3 cycle.

We could define these "Subscriber groups" in Adminland, as pitched previously:



Adding Groups on Adminland

At a minimum, we could confine the new functionality to that Subscribers modal.

If we want to, we could expand to also invite people to projects with the same approach. The modals are nearly identical. JZ's sketch for that is on the [prior pitch](#).

What we're not doing

This goes a step further narrow than the last pitch. No autocompleting Groups anywhere. Still no connection with Teams. Just a better way to narrow down Subscribers any place in the app that we let you choose Subscribers via the "Who should be notified?" modal.

Last chance for this one in the next cycle! Hope this sharpened it down enough.