14

Choosing a Randomization Unit

> [To generate random digits] a random frequency pulse source, providing on the average about 100,000 pulses per second, was gated about once per second by a constant frequency pulse . . . Production from the original machine showed statistically significant biases, and the engineers had to make several modifications and refinements of the circuits before production of apparently satisfactory numbers was achieved. The basic table of a million digits was then produced during May and June of 1947. This table was subjected to fairly exhaustive tests and it was found that it still contained small but statistically significant biases
> *A Million Random Digits with 100,000 Normal Deviates (RAND 1955)*

Why you care: *The choice of randomization unit is critical in experiment design, as it affects both the user experience as well as what metrics can be used in measuring the impact of an experiment. When building an experimentation system, you need to think through what options you want to make available. Understanding the options and the considerations to use when choosing amongst them will lead to improved experiment design and analysis.*

Identifiers are critical as the base randomization unit for experiments. The same identifier can also be used as a join key for the downstream processing of log files (see Chapter 13 and Chapter 16). Note that in this section we are focusing on how to choose which identifier to use, rather than on base criteria for randomization itself, such as ensuring the independence of assignment (i.e., the variant assignment of one identifier should not tell us anything about the variant assignment of another identifier) as well as ensuring the independence of assignment across experiments if an identifier can be assigned to multiple experiments simultaneously (see Chapter 4).

One axis to consider in choosing a randomization unit is granularity. For example, websites have the following natural granularities:

- **Page-level:** Each new web page viewed on a site is considered a unit.
- **Session-level:** This unit is the group of webpages viewed on a single visit. A session, or visit, is typically defined to end after 30 minutes of inactivity.
- **User-level:** All events from a single user is the unit. Note that a user is typically an approximation of a real user, with web cookies or login IDs typically used. Cookies can be erased, or in-private/incognito browser sessions used, leading to overcounting of users. For login IDs, shared accounts can lead to undercounting, whereas multiple accounts (e.g., users may have multiple e-mail accounts) can lead to overcounting.

We'll focus in on the examples on this axis for websites to discuss the main considerations.

For search engines, where there can be multiple pageviews for a single query, a query can be a level of granularity between page and session. We can also consider a combination of user and day to be a unit, where events from the same user on different days are in different units (Hohnhold, O'Brien and Tang 2015).

When trying to decide on the granularity, there are two main questions to consider:

1. How important is the consistency of the user experience?
2. Which metrics matter?

For consistency, the main question is whether the user will notice the changes. As an extreme example, imagine that the experiment is on font color. If we use a fine granularity, such as page-level, then the font color could change with every page. Another example is an experiment that introduces a new feature; the feature may appear and disappear if the randomization is at the page-level or the session-level. These are potentially bad and inconsistent user experiences that can impact key metrics. The more the user will notice the Treatment, the more important it is to use a coarser granularity in randomization to ensure the consistency of the user experience.

Your choice of metrics and your choice of randomization unit also interact. Finer levels of granularity for randomization creates more units, so the variance of the mean of a metric is smaller and the experiment will have more statistical power to detect smaller changes. It is worth noting that randomizing (and analyzing) by pageviews will lead to a tiny underestimation of the variance of the Treatment effect (Deng, Lu and Litz 2017), but that underestimation is very small in practice and is commonly ignored.

While a lower variance in metrics may seem like an advantage for choosing a finer granularity for randomization, there are several considerations to keep in mind:

1. If features act across that level of granularity, you cannot use that level of granularity for randomization. For example, if you have personalization or other inter-page dependences, then randomizing by pageview is no longer valid as what happens on one page affects what a user sees on the subsequent page and the pages are no longer independent. As another specific example, if an experiment is using page-level randomization, and a user's first query is in the Treatment and the feature leads to poor search results, the user may issue a reformulated second query that ends up in the Control.
2. Similarly, if metrics are computed across that level of granularity, then they cannot be used to measure the results. For example, an experiment that uses page-level randomization cannot measure whether the Treatment impacts the total number of user sessions.
3. Exposing users to different variants may violate the stable unit treatment value assumption (SUTVA, see Chapter 3) (Imbens and Rubin 2015), which states that experiment units do not interfere with one another. If users notice the different variants, that knowledge may impact their behavior and interfere (see Chapter 22).

In some enterprise scenarios, such as Office, tenants would like consistent experiences for the enterprise, limiting the ability to randomize by user. In advertising businesses that have auctions where advertisers compete, you could randomize by advertiser or by clusters of advertisers who are often competing in the same auctions. In social networks, you can randomize by clusters of friends to minimize interference (Xu et al. 2015, Ugander et al. 2013, Katzir, Liberty and Somekh 2012, Eckles, Karrer and Ugander 2017), and this generalizes to networks generally if you consider components (Yoon 2018)

Randomization Unit and Analysis Unit

Generally, we recommend that the randomization unit be the same as (or coarser than) the analysis unit in the metrics you care about.

It is easier to correctly compute the variance of the metrics when the analysis unit is the same as the randomization unit, because the independence assumption between units is reasonable in practice, and Deng et al. (2017) discuss the independent and identical distribution (i.i.d.) assumption with regards to

choice of randomization unit in detail. For example, randomizing by page means that clicks on each pageview are independent, so computation for the variance of the mean, click-through rate (clicks/pageviews), is standard. Similarly, if the randomization unit is *user* and the metrics analysis unit is also *user*, such as sessions-per-user, clicks-per-user, and pageviews-per-user, then the analysis is relatively straightforward.

Having the randomization unit be coarser than the analysis unit, such as randomizing by *user* and analyzing the click-through rate (by page), will work, but requires more nuanced analyses methods such as bootstrap or the delta method (Deng et al. 2017, Deng, Knoblich and Lu 2018, Tang et al. 2010, Deng et al. 2011). See Chapter 18 and Chapter 19 for more discussion. In this situation, the experiment results can be skewed by bots that use a single user ID, e.g., a bot that has 10,000 pageviews all done using the same user ID. If this type of scenario is a concern, consider bounding what any individual user can contribute to the finer-grained metric or switching to a user-based metric such as the average click-through rate-per-user, both of which bound the contribution any single user can have on the result.

Conversely, when the metrics are computed at the user-level (e.g., sessions-per-user or revenue-per-user) and the randomization is at a finer granularity (i.e., page-level), the user's experience likely contains a mix of variants. As a result, computing metrics at the user-level is not meaningful; you cannot use user-level metrics to evaluate an experiment when the randomization is by page. If these metrics are part of your OEC, then you cannot use the finer levels of granularity for randomization.

User-level Randomization

User-level randomization is the most common as it avoids inconsistent experience for the user and allows for long-term measurement such as user retention (Deng et al. 2017). If you are using user-level randomization, you still have several choices to consider:

- A signed-in user ID or login that users can use across devices and platforms. Signed-in IDs are typically stable not just across platforms, but also longitudinally across time.
- A pseudonymous user ID, such as a cookie. On most websites when a user visits, the website writes a cookie containing an identifier (usually mostly random). On mobile devices for native apps, the OS often provides a cookie, such as Apple's idFA or idFV or Android's Advertising ID. These IDs are not persistent across platforms, so the same user visiting through

desktop browser and mobile web would be considered two different IDs. These cookies are controllable by the user through either browser-level controls or device OS-level controls, which means that cookies are typically less persistent longitudinally than a signed-in user ID.

- A device ID is an immutable ID tied to a specific device. Because it is immutable, these IDs are considered identifiable. Device IDs do not have the cross-device or cross-platform consistency that a signed-in identifier has but are typically stable longitudinally.

When debating between these choices, the key aspects to consider are functional and ethical (see Chapter 9).

From a functional perspective, the main difference between these different IDs is their scope. Signed-in user IDs cut across different devices and platforms, so if you need that level of consistency and it is available, a signed-in user ID is really your best choice. If you are testing a process that cuts across the boundary of a user signing in, such as a new user on-boarding process that includes a user signing in for the first time, then using a cookie or device ID is more effective.

The other question about scope is the longitudinal stability of the ID. In some experiments, the goal may be to measure whether there is a long-term effect. Examples may include latency or speed changes (see Chapter 5) or or the users' learned response to ads (Hohnhold et al. 2015). For these cases use a randomization unit with longevity, such as a signed-in user ID, long-lived cookie, or device ID.

One final option that we do not recommend unless it is the only option is IP address. IP-based variant assignment may be the only option for infrastructure changes, such as for comparing latency using one hosting service (or one hosting location) versus another, as this can often only be controlled at the IP level. We do not recommend using IP addresses more generally, however, because they vary in granularity. At one extreme, a user's device IP address may change when a user moves (e.g., a different IP address at home than work), creating inconsistent experiences. At the other extreme, large companies or ISPs have many users sharing a small set of IP addresses representing the firewall. This can lead to low statistical power (i.e., do you have enough IP addresses, especially to handle the wide variance), as well as potential skew and outlier issues from aggregating large numbers of users into a single unit.

Randomization at a sub-user level is useful only if we are not concerned about carryover or leakage from the same user (see Chapter 22), and the success metrics are also at the sub-user level (e.g. clicks-per-page, not clicks-per-user). It is often chosen in favor of increased power that comes from the increase in sample size.