Attribution Reporting API Origin Trial

Updates & Lessons

Updates

.4%

Of all page views in Chrome invoke ARA

8+

Active testers

New & upcoming features

New features!

- Debug reports for noiseless data, and various error conditions
- Multiple destination support
- More flexible attribution windows

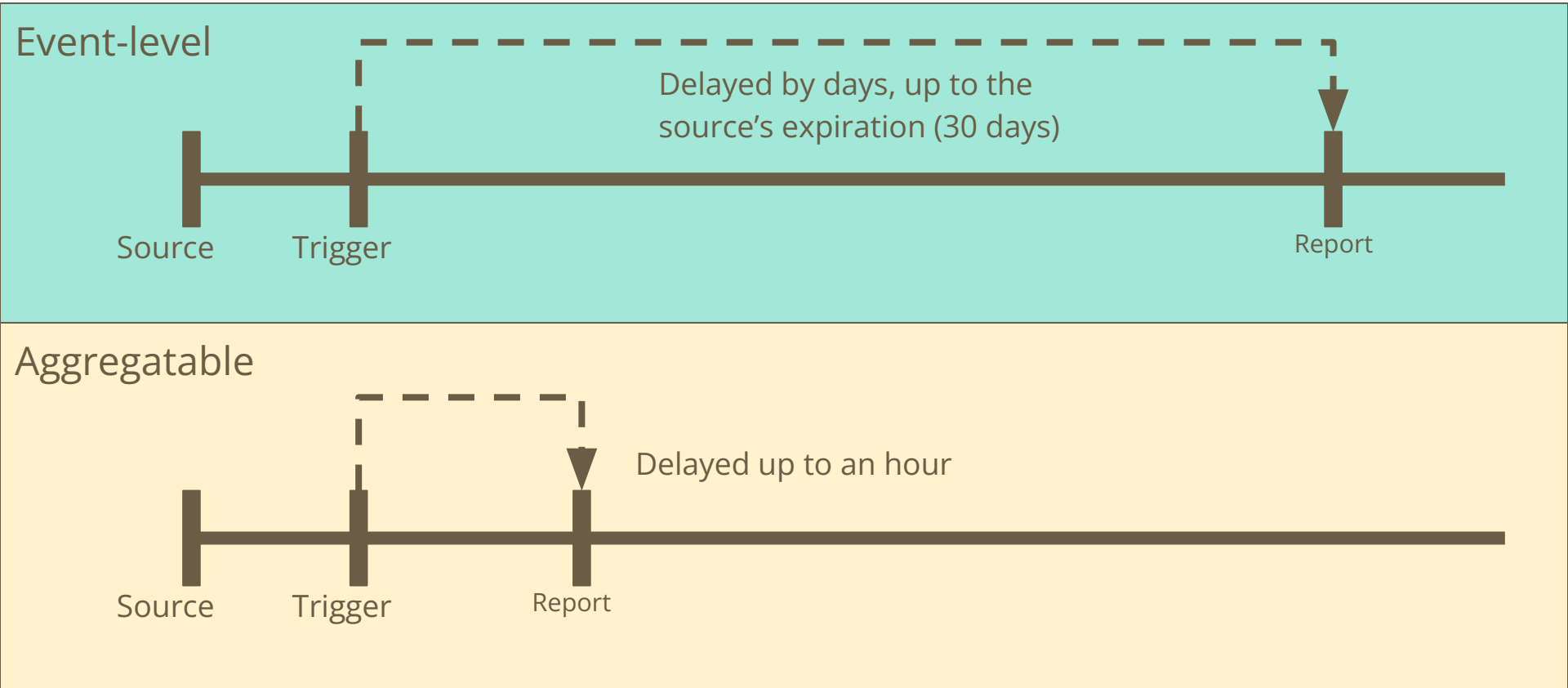
Coming soon!

- App <-> Web with Android's Privacy Sandbox
- Trigger attestation using Private State Token infrastructure
- Better FLEDGE integration
- Multi-cloud support

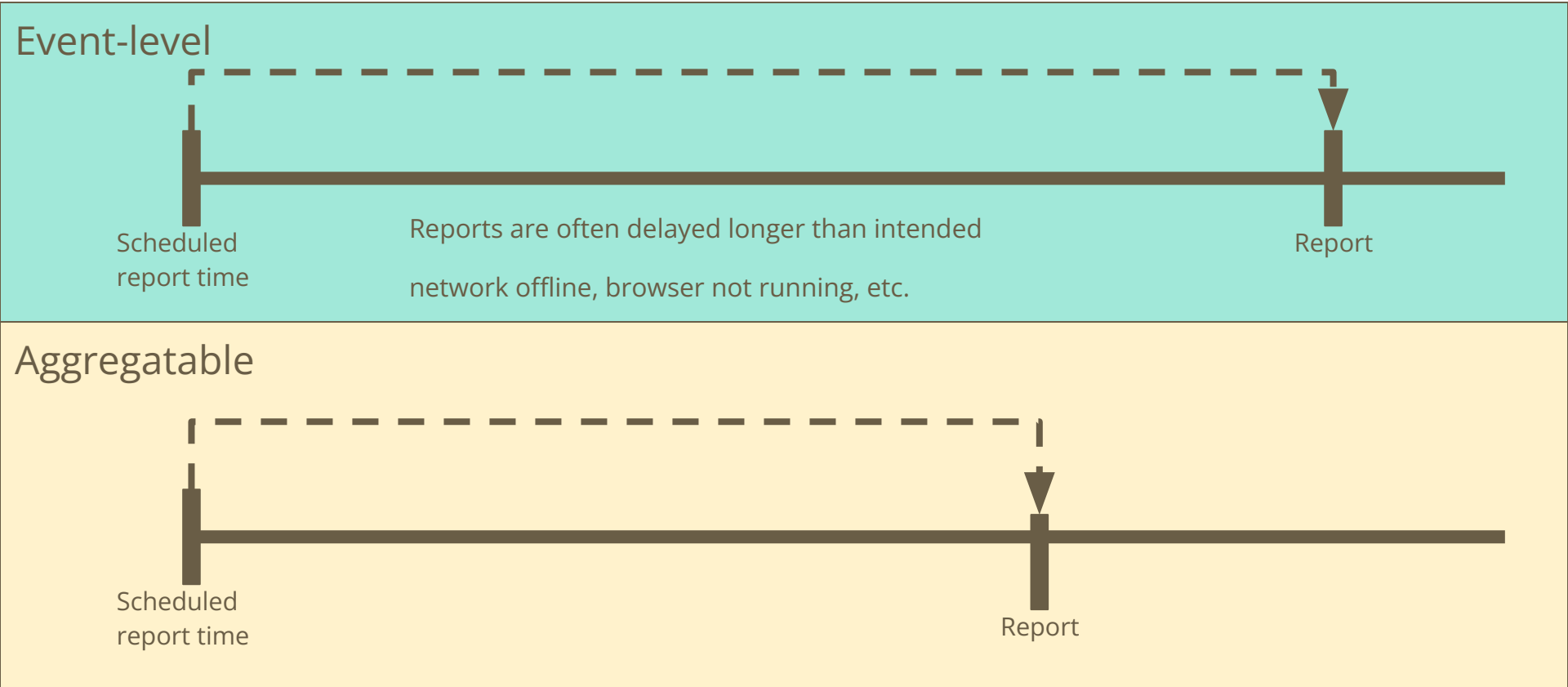
Lessons

Hopefully relevant to any private measurement solution

Report delays



Report delays: ideal vs. actual



Report delays: takeaways

- Report delays hurt utility in multiple ways
- Expect longer effective delays due to offline clients
- Delays have privacy risks as well, e.g. presence / IP tracking
- Delays are almost impossible to eliminate with on-device attribution
  - Aggregatable reports: need to be sent ~unconditionally at trigger time
  - Event-level reports: Nearly impossible without server-side infra

Debugging

Problem: the attribution algorithm and its associated failures is opaque and hard to debug

Solutions:

- Introduce temporary debugging reports to root out bugs in the implementation and deployments. Requires third party cookies to use!
  - 4 separate errors cases for source registration
  - 17 error cases for attribution triggering
  - Immediate reports for when triggering attribution succeeds
- Build and launch simulators that can be used with test data
  - Noise Lab
  - Simulation libraries

Debugging: takeaways

Callers will need insight into how complex machinery like attribution / budgeting works, but this insight might expose cross-site information!

Platforms should invest in tooling to aid debugging, and think carefully about sustainable, long-term debugging strategies.

Aggregation function

Current design exposes **fixed, noisy histograms**

But we're hearing the need for **other functionality**

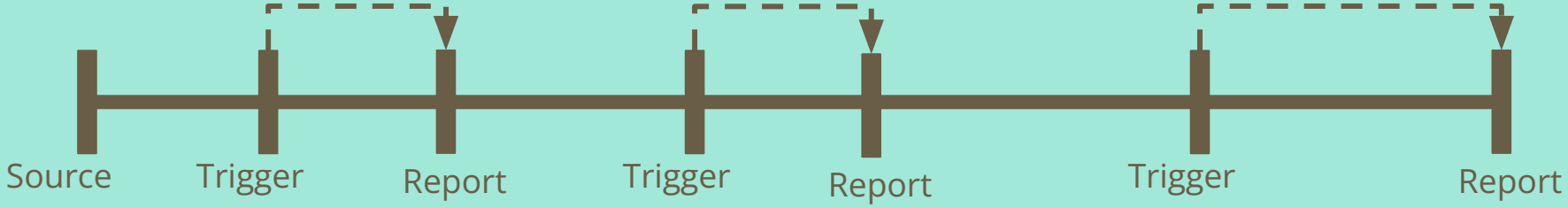
- Key discovery (e.g. exponential / non-enumerable key space)
- Distinct counting
- ML model training
- Flexible privacy budgeting

Aggregation function: takeaways

Any system should be built to **accommodate a diversity of algorithms**, to support the **diversity of private measurement use-cases**

Event-level parameters

Clicks: 3 bit of metadata, 3 reporting windows, 3 attribution: **2925 output states**



Views: 1 bit of metadata, 1 reporting window, 1 attribution: **3 output states**

Event-level parameters: takeaways

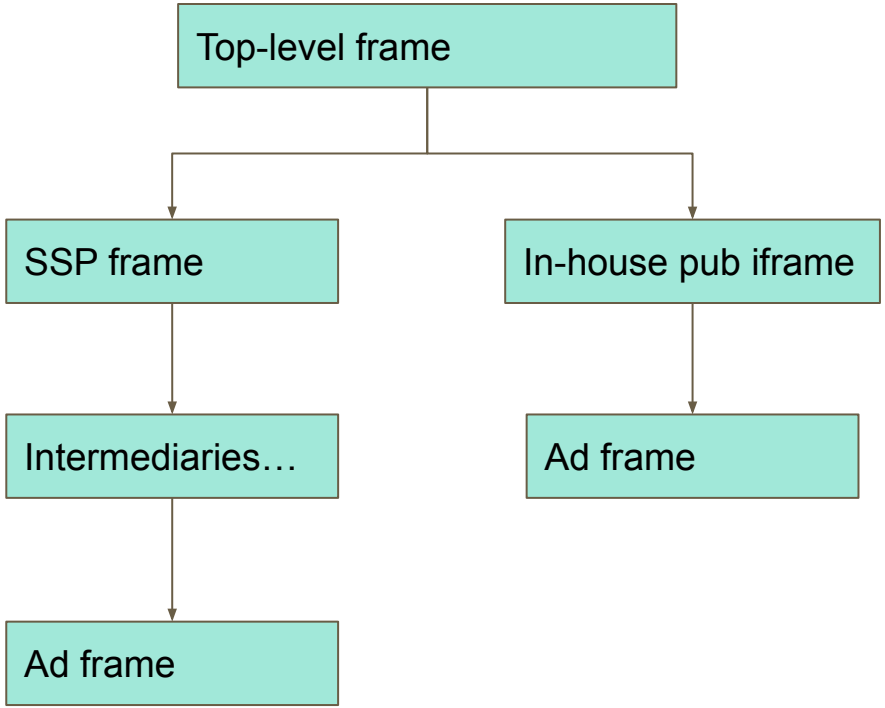
- One size does not fit all: different use-cases may require different parameterizations
  - Though we are looking for feedback on fixed parameter sets
- We shouldn't penalize use-cases that require fewer output states
  - "Paying for" utility that isn't needed in terms of noise
- State-of-the-art ML training techniques can take advantage of flexibility if callers have a prior on the features
  - Ghazi et. al. 2021 / 2022 in the "label DP" regime
  - Though mixing differential privacy and information gain privacy definitions may complicate the mechanism

Permissions

Permission Policy decides which context is allowed to register sources and triggers.

Original goal: enforce some level of opt-in for the top-level, while keeping the API adoptable.

Challenge: complex existing deployments have intermediary frames that can accidentally (or maliciously) break all measurement for downstream frames.



Arrow: must specify allow=”attribution-reporting” on iframe

Permissions: takeaways

- Initial testing with strict Permissions is difficult, “boiling the ocean”
- Need to balance privacy, security, and adoption risk
- Current thinking is to move to an opt-out model (default `*` permission)
    - Investigate changes to Permissions Policy to match our ideal

User data deletion

- With attribution on-device, users can explicitly delete pending sources
  - Leads to future attribution triggers matching nothing
- With delayed reporting, users can explicitly delete pending reports
- Double-edged sword
  - Arguably more in line with what users want user-data deletion to do if deletion is a proxy for avoiding tracking
  - Leads to unpredictable utility loss, especially since it’s difficult for the platform to surface this loss to callers

Summary

- Delays: problematic but hard to avoid for local privacy mechs
- Debugging: Helpful when the platform obscures pieces of the ads engine
- Agg function: One size does not fit all
- Event-level parameters: One size does not fit all!
- Permissions: Pervasive opt-in is problematic
- User data deletion: Trades utility for privacy, and hard to avoid for local privacy mechs

More Feedback Needed

- Tolerable epsilons (especially for aggregate summary reports)
- Specific parameterization of event-level API
- Trigger attestation design

Conclusion

- Deploying ARA has given us useful real-world experience
- Stay tuned in the future for more lessons we hope to share with PATCG to improve our private measurement work together
- We’re committed to future improvements to the API, both for privacy and utility