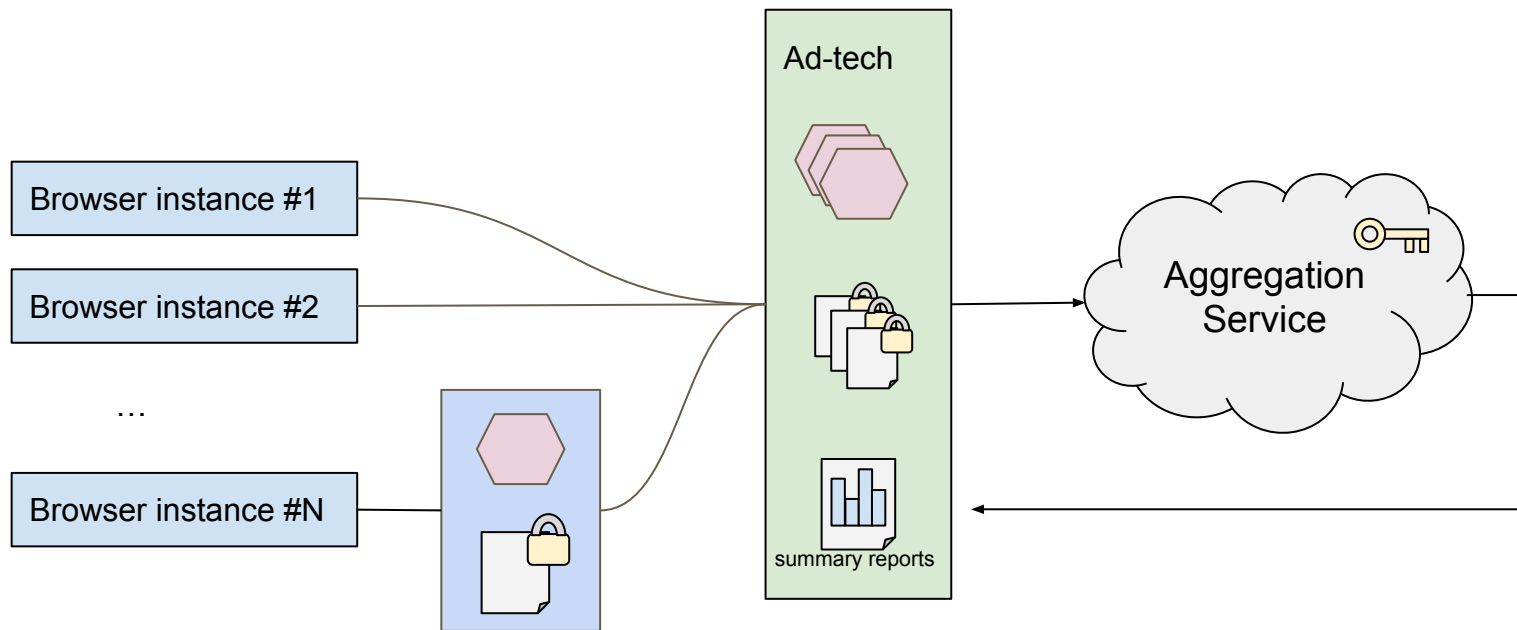# Attribution Reporting

Design space trade-offs and considerations
csharrison@google.com
johnidel@google.com

Overview: Attribution Reporting API
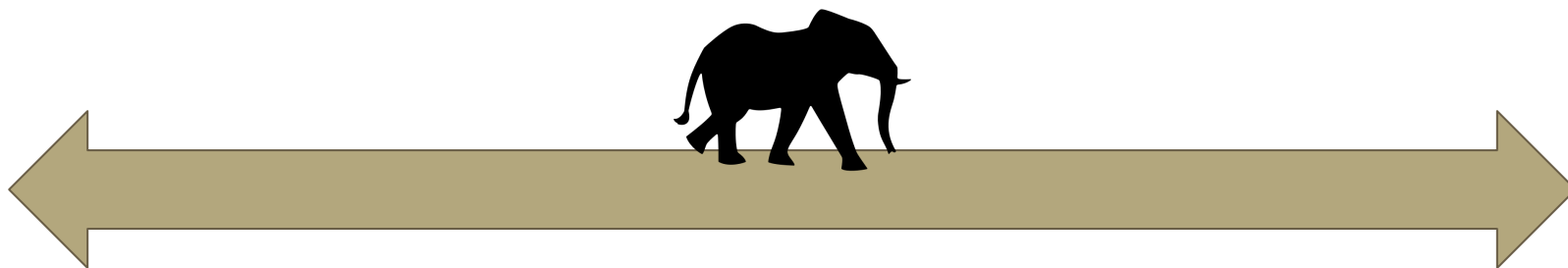

= encrypted, aggregatable reports = "histogram contribution"

= event-level report = {unique ad ID, coarse conversion ID}

Privacy vs. Utility

More private, less useful
- Delayed
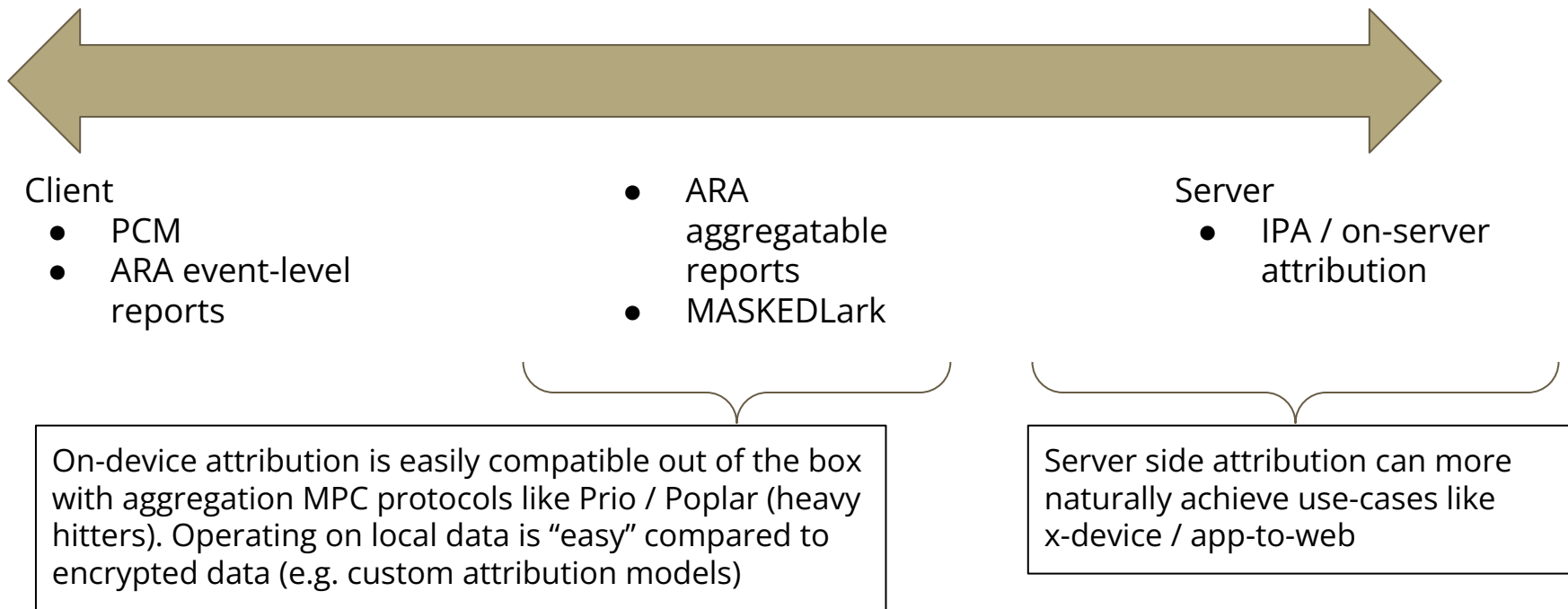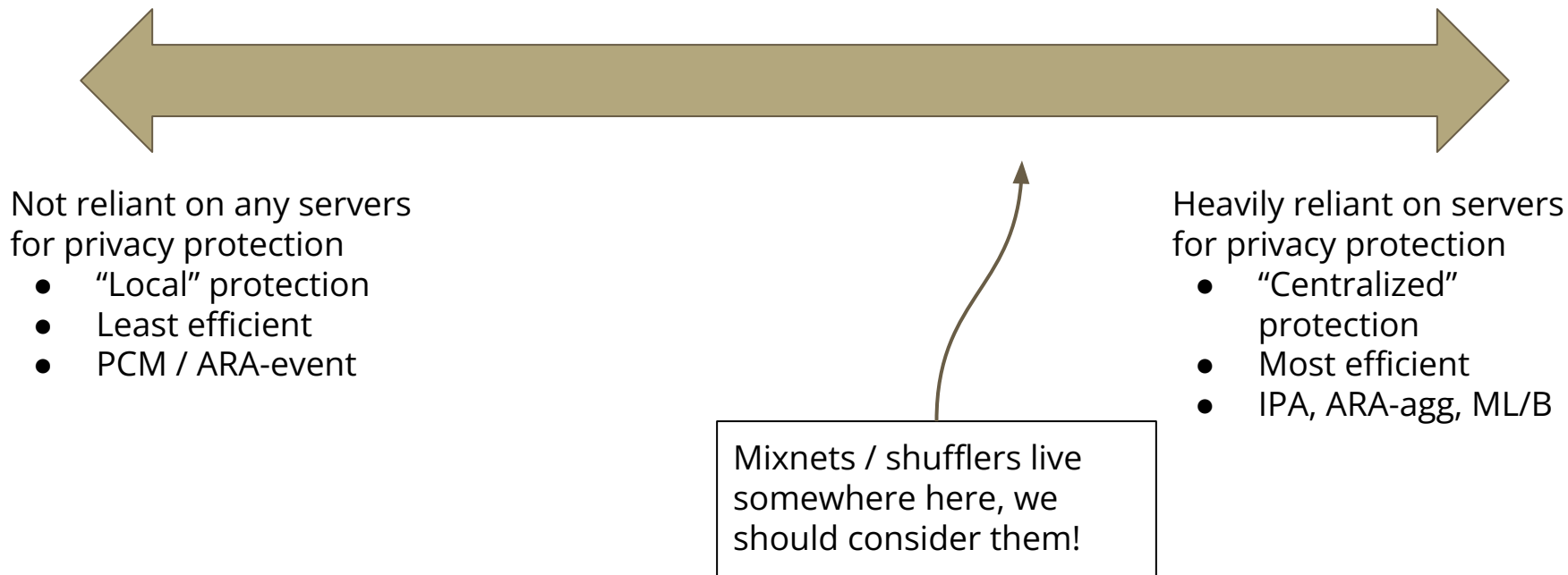- Coarse
- Noisy

May be difficult to achieve group alignment out of the gate. Solutions should be flexible and provide a framework for experimenting within the privacy / utility frontier.

More useful, less private
- Timely
- Fine-grained
- Accurate

Where does complexity live? on-device vs. in servers

Client
- PCM
- ARA event-level reports

- ARA aggregatable reports
- MASKEDLark

Server
- IPA / on-server attribution

On-device attribution is easily compatible out of the box with aggregation MPC protocols like Prio / Poplar (heavy hitters). Operating on local data is "easy" compared to encrypted data (e.g. custom attribution models)

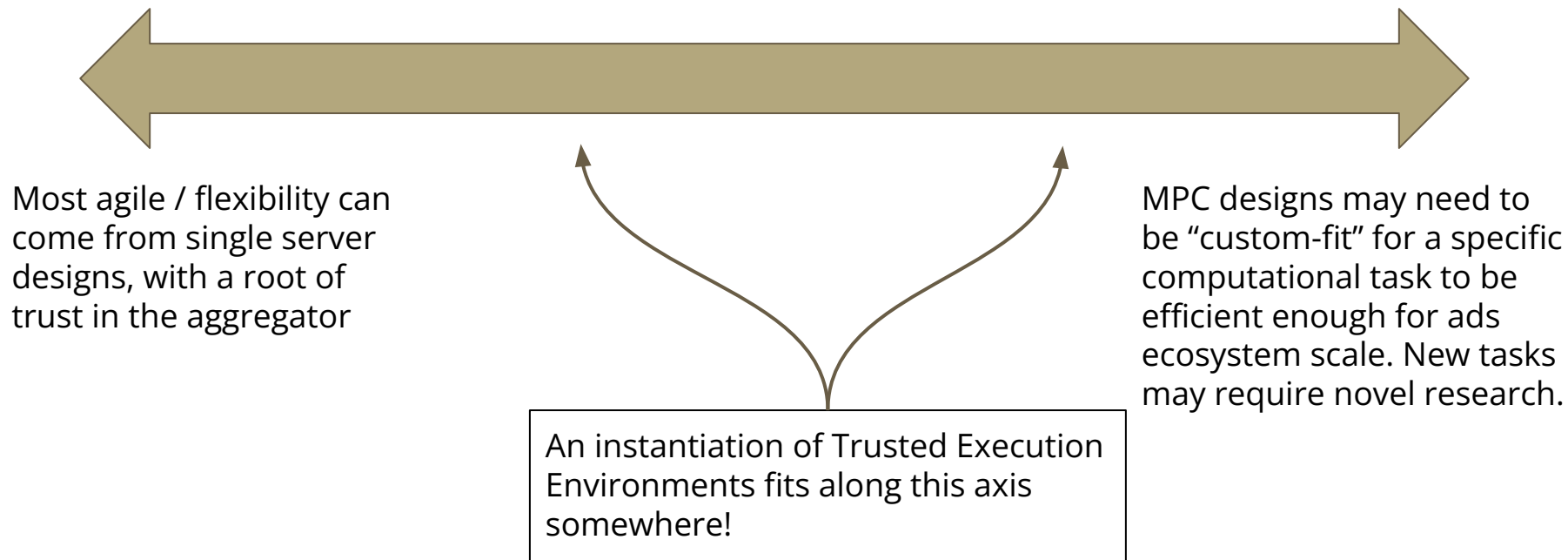Server side attribution can more naturally achieve use-cases like x-device / app-to-web

Server-reliance and Privacy Efficiency

aka "Aggregation works". Privacy efficiency = "most utility bang for privacy buck"

Not reliant on any servers
for privacy protection
- "Local" protection
- Least efficient
- PCM / ARA-event

Heavily reliant on servers
for privacy protection
- "Centralized"
  protection
- Most efficient
- IPA, ARA-agg, ML/B

Mixnets / shufflers live
somewhere here, we
should consider them!

API agility and trusted server architecture

Most agile / flexibility can come from single server designs, with a root of trust in the aggregator

An instantiation of Trusted Execution Environments fits along this axis somewhere!

MPC designs may need to be "custom-fit" for a specific computational task to be efficient enough for ads ecosystem scale. New tasks may require novel research.

Delegation & Privacy budgeting

That's all for now