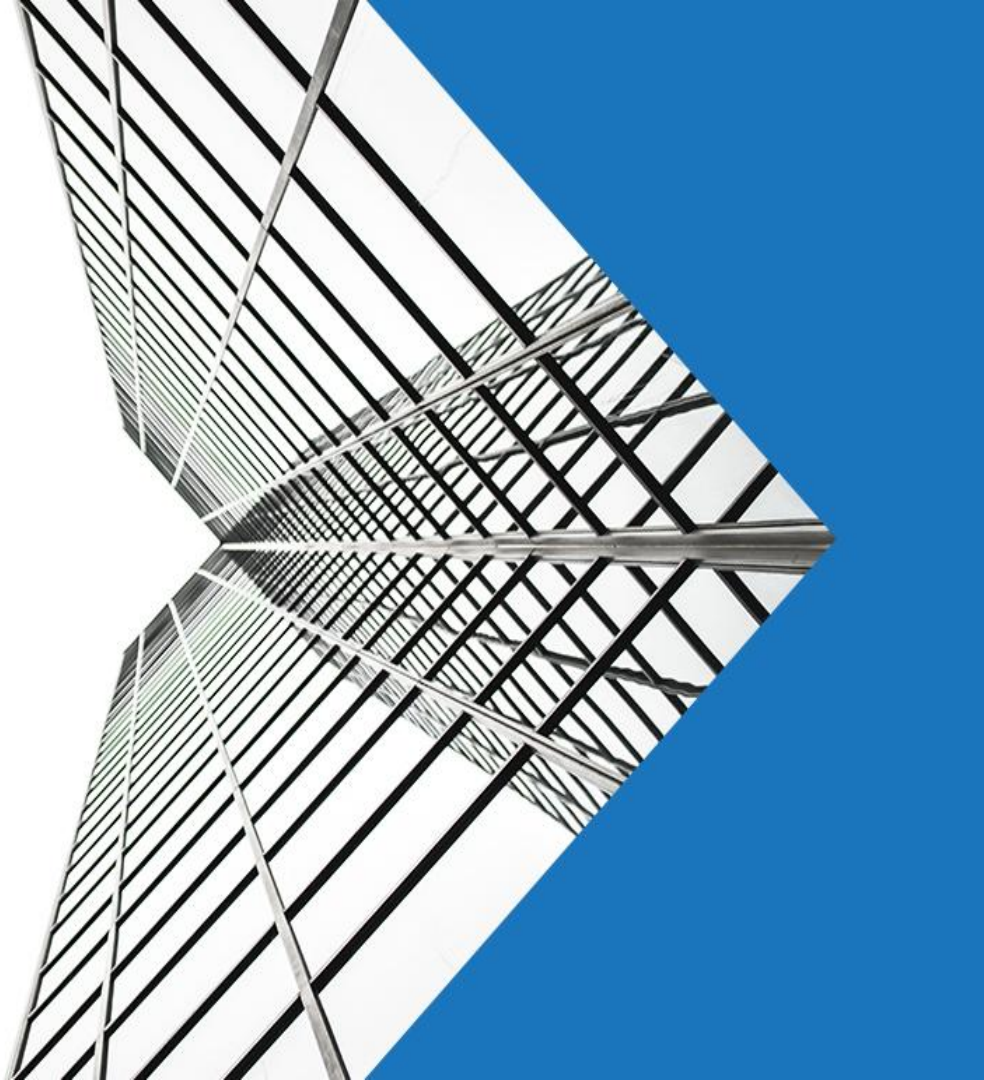
AdEx Architecture walk-through

Ivo Georgiev
CEO, AdEx Network



Benefits

- Transparent reporting for all sides
- Minimized trust required
- Minimized fees
- End users in control of their data
- Blockchain-agnostic
- Token/currency-agnostic
- Browser/runtime-agnostic
- **Wide variety of use cases, including but not limited to: display advertising, affiliate networks, and even content micropayments**

Solving ad fraud

- Streamlining relations/interactions between participants: publishers, advertisers, target audience
- Fake clicks cannot be fundamentally solved, but we can remove the “black boxes” and mitigate fraud
- We allow a wider range of methods for fake click detection
- Possibly using identity in the future

Components

- Core (the on-chain part of OUTPACE, Identity); running on Ethereum
- Validator stack
- Market
- SDK/AdView
- Registry
- Platform (formerly known as dapp)
- Satellite components: Explorer, Lounge

Terminology

- Publisher: supplier - supplies ad space/time
- Advertiser: buys the supply
- OUTPACE: a type of payment channel(s), one-to-many, unidirectional
- Campaign: “coordinated series of linked advertisements with a single idea or theme” implemented as a superset of “payment channel”
- Campaign health
- Dai stablecoin ◈

Terminology 2

- Publisher validator: a piece of infrastructure that tracks impressions/campaigns on behalf of a publisher and updates the OUTPACE channel of the campaign; somewhat equivalent of SSP
- Advertiser validator: same, but for advertisers; somewhat equivalent of DSP
- Validator = publisher validator OR advertiser validator
- Observer: like a validator; but with no active role of updating OUTPACE channels
- Leader = refers to the role of the advertiser validator in OUTPACE
- Follower = refers to the role of the publisher validator in OUTPACE

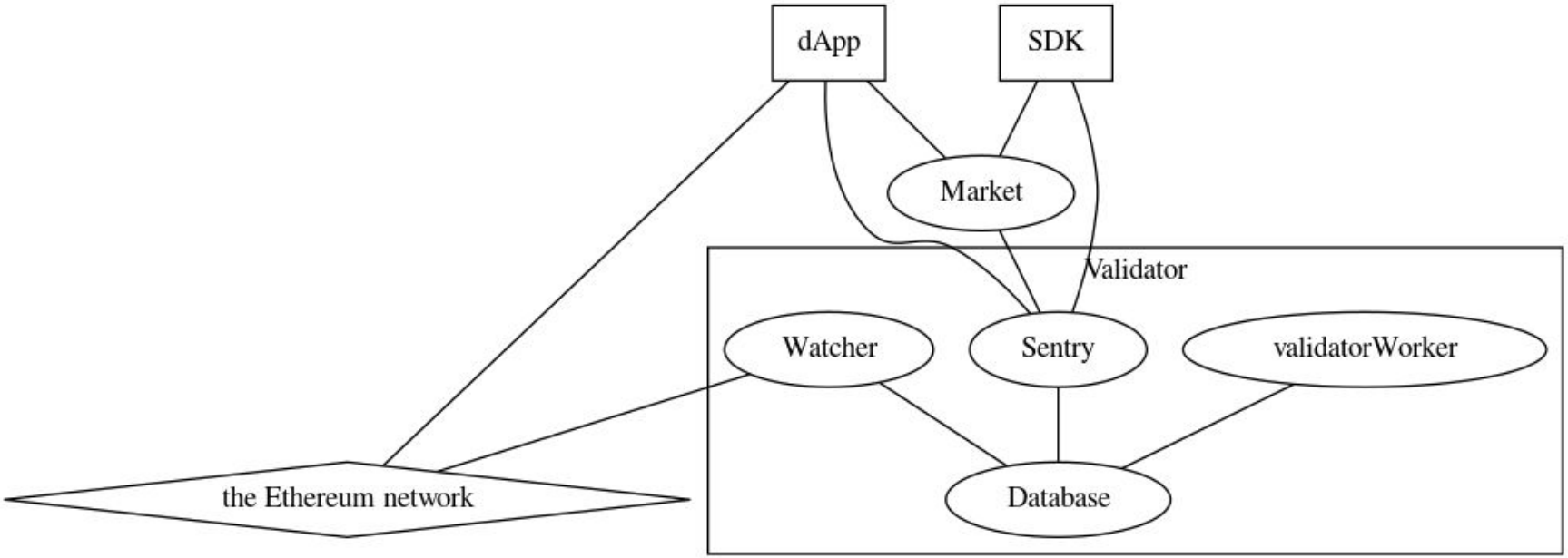
Terminology: recaps

- Advertiser validator *is* leader *is* validator
- Publisher validator *is* follower *is* validator
- SDK = Adview
- Target audience = end user

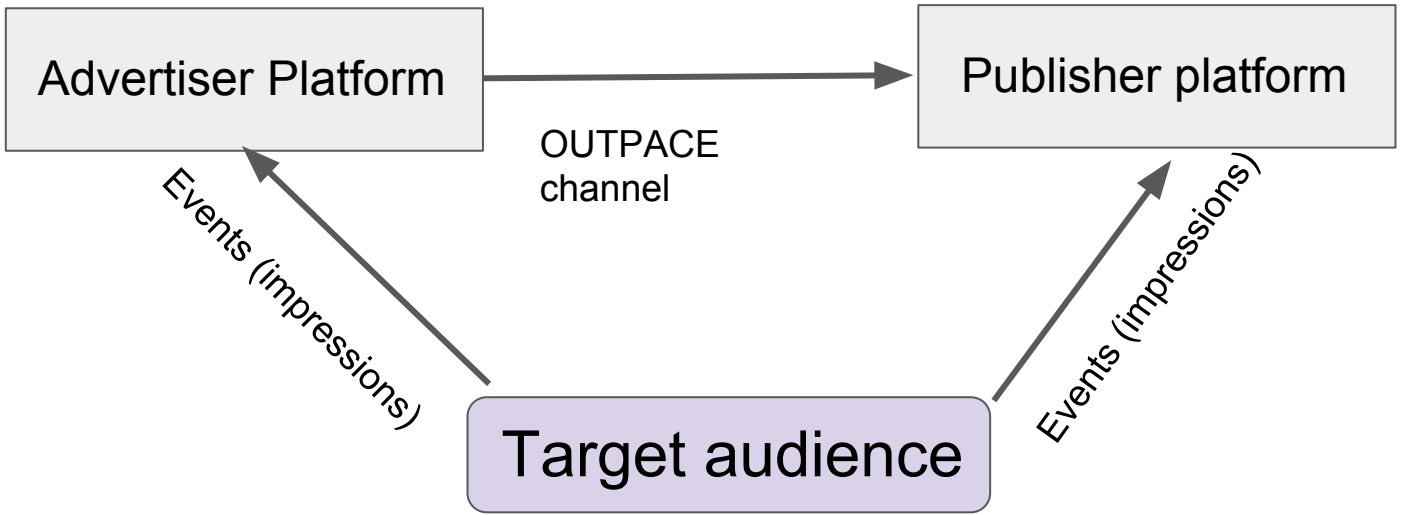
Platform

- Formerly known as dApp
- Designed for publishers/advertisers
- Transacts only in Dai ◈
- Currently quite hard to use; after the next update, we expect sign up to become much easier

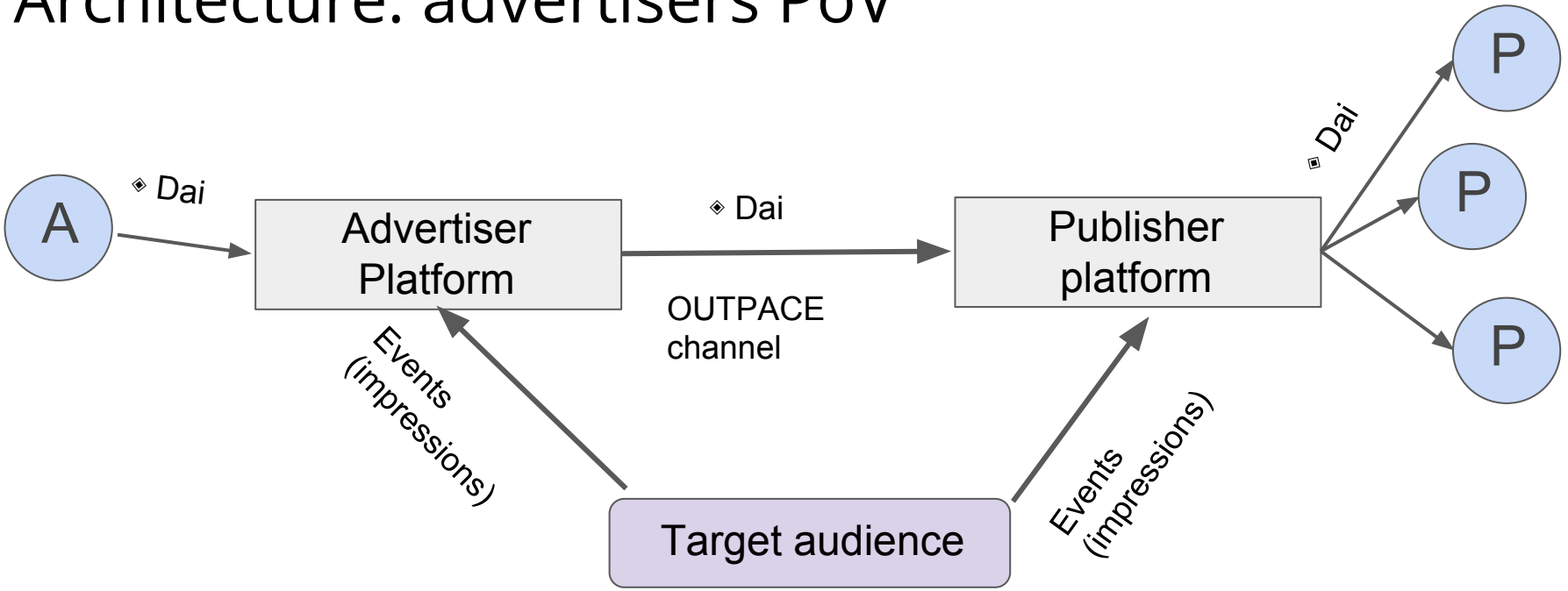
Architecture



Architecture: user PoV



Architecture: advertisers PoV



Ensuring availability: Registry

- The primary job of the Registry is to ensure validators (Publisher-side/Advertiser-side) stay online
- And don't cheat (although OUTPACE does a good job on this already)
- It guarantees that by allowing validators to stake ADX; if they go offline, portions of the deposit get burned (**slashing**)
- The higher the stake, the more you can assume they'll *stay alive*
- The Registry will be a Polkadot "parachain" due to high throughput requirements

Protecting user data: SDK, Lounge

- User data is only kept in their own browser
- Instead of cookies, we have local storage (not exempt from regulations!)
- When you delete the browser storage/cookies, or uninstall the browser, all data is lost. **Same as in any other system!**
- Publishers “report” what they know about the user to the SDK
- The SDK uses all of this information to select ads

Bidding

- In real time, header bidding
- The SDK pulls all active campaigns, and each campaign can “bid” for this particular space by changing how much they are willing to pay in real time
- The SDK sorts the bids and takes top N
- From these top N, the SDK applies targeting and picks one for the user
- If a publisher wants to increase their revenue at the expense of UX, they decrease N

Campaign (channel) health

A healthy campaign is one that:

- All validators (leader, follower) are online and are producing new states
- The leader and follower are in consensus
  - OUTPACE rules are maintained
  - Event ledgers for both validators are similar (within the allowed threshold)
- A campaign can recover smoothly from being unhealthy
  - An offline validator comes back
  - A crashed validator recovers
- Unhealthy means ads won’t be served from this campaign

Redundancy

- Problem: network/ISP failures, the SDK can’t send events to one of the validators
  - Solution: each validator may run multiple Sentries, in different regions (data centers)
- Problem: a validator may go offline (server failure)
  - Solution: the validator architecture allows redundancy, by running multiple Sentries and a DB replica set
- Problem: a validator may be DDoS-ed
  - The same issue exists in traditional adtech with ad servers, and is mitigated via DDoS protection services like Cloudflare
  - Solution: Sentries are reached through HTTPS, so Cloudflare helps; furthermore, if a validator is DDoS-ed, it will recover once the attack stops

Redundancy 2

- Problem: bugs, crashes
  - Mitigation: audits, fuzzing, tests, good engineering practices
  - Mitigation: robust architecture: can recover from most crashes
  - Solution: even in the case of a crash, the campaign will be marked as unhealthy, and once the issue is fixed, it will become healthy again

Status (as of mid March 2019)

- Core: 100% done; next step: deploying to mainnet
- Validator stack: 100% done; next step: a bunch of deployed validators
- Market: mostly done, still has to be audited/reviewed
- Platform, SDK: needs to be updated for the new protocol
- Registry, Lounge, Explorer: TBD

Links

https://github.com/AdexNetwork/adex-protocol

https://github.com/adexnetwork/adex-protocol-eth

https://github.com/adexnetwork/adex-validator-stack-js

Going further

- End users paying for content
- End users receiving rewards for being interrupted by ads
- Affiliate networks
- Physical advertising (oracles)
- Harberger tax

Thank you!

@AdExNetwork
@lvshti

www.adex.network
github.com/AdExNetwork