

# Role of Discovery & Centralization

Jari Arkko

IAB Retreat, May 2022

## Additional considerations beyond draft-nottingham, from IETF 113

Issue	What	Effect on centralization	Potential technical mitigations	Examples
Data	Data accumulates in small number of hands	Data holder becomes target of surveillance activities and commercial pressures. Competing services have difficulty to get started, if they need, e.g., AI training.	Oblivious techniques, privacy preserving measurements, data minimization, confidential computing	Oblivious DNS, PRIV/PPM
Component linkages	Systems get built with default linkages to limited set of services	OS, device, browser, and application systems get built to use very specific services. Difficult for other or local services to be used.	Discovery, modularity	DDR

# Tightly bound component linkages

What	Description
Examples	App stores, DNS resolvers for a device, network, browser, or app; Browser and its default search engine; Oblivious relay/service combination, The network of components providing ppm-like services, ...
Rationale	Need this service for our users. We are the gatekeepers to keep bad things out. System builders understands exactly what components they need and associated APIs. There may also be contracts.
Concerns	It becomes difficult for local services or non-major services to be used. Gatekeepers often extract a price for their service. Defaults almost never changed by majority of users, and in many cases you cannot opt out from the ecosystem. Whoever built the system gets to decide a lot.
Additional issues	<ol style="list-style-type: none"><li>1. Changing settings is not a practical alternative for vast majority of users</li><li>2. Only the services that can have a default bundling e.g. with major browser or OS vendor will be used</li><li>3. Internet and applications become more and more 'packaged', designed &amp; curated black box services</li></ol>
Potential solutions	Modularity. Indirection and discovery mechanisms. Regulation.
Difficulties	For many functions, it is not ok to use a random thing that someone nearby offers, for obvious security reasons. Indirection and user/os-specified trusted roots for specific functions may be helpful though.

# What to do?

- Ensure issue(s) get documented in existing documents (e.g., draft-nottingham)
- Write more specific, elaborated documents on this issue