REPUTATION AND TWO-SIDED NETWORKS

Samuel M. Smith Ph.D.
2018.04.05
sam.smith@consensys.net

Inversion

The action of inverting something or the state of being inverted:

a reversal of order

An inversion of economic order provides an opportunity to release latent value

Disintermediation Theory

Selectorate: Those in power to make decisions for others as intermediaries, e.g. their agents.

Selectorate may be merely nominal, or influential, or essential.

How distributed/centralized is the agency of the selectorate? (democratic——despotic)

Distribution of agency increases the ratio of selectorate to total population

Typically better outcomes arise overall when agency is more distributed

Increased agency results in increased autonomy (happiness) and meritocracy (performance)

The mechanism for distributing agency (increasing the selectorate) is disintermediation.

To change outcomes, don't ask those in power to be nicer, fairer, or more responsive, just disintermediate them.

Seek disintermediation, not justice.

Disintermediation Opportunities

Whenever producers/consumers of value interact through intermediaries (gatekeepers).

Example:

Social media on Facebook is intermediated by Facebook.

Facebook is the sole intermediary.

Facebook owns all content and controls all interactions in order to extract value.

What if users could interact without the value tax and interaction control of Facebook?

PLATFORM BUSINESS MODELS

A platform is a business based on enabling value-creating interactions between external producers and consumers.

A platform provides an open, participative infrastructure and sets governance conditions for these interactions.

A platform consummates matches among users that facilitate the exchange of goods, services, or social currency, thereby enabling value creation or co-creation for all participants.

A platform is the antithesis of a pipeline.

(See Platform Revolution 2016, Platform Scale 2015)

PLATFORMS VS PIPELINES

A pipeline is a business that directly creates and moves value from …

producers at one end to consumers at the other end.

Platforms eat pipelines because platforms unlock new sources of value creation and supply

not-even-mine replaces just-in-time

Primary activity shifts from …

internal mechanisms of control of the value chain

to …

external orchestration/coordination of interactions between third parties

Platform Manifesto

External eco-system is the new warehouse and supply chain
Network effect is the new driver for scale
Data is the new dollar
Community management is the new human resource management
Liquidity management is the new inventory control
Curation and reputation are the new quality control
User journeys are the new sales funnels
Distribution is the new destination
Behavior design is the new loyalty program
Data science is the new business process optimization
Social feedback is the new sales commission
Algorithms are the new decision-makers
Real-time customization is the new market research
Plug-n-play (APIs) is the new business development
Invisible hand is the new iron fist

Radical Disruptive Disintermediation

Make the governance of the interaction more distributed.

Allow users (producers and consumers) to interact without despotic control.

Allow user ownership of their content and control of their interactions.

Enable users to extract more value from their content and interactions.

Closed pipelines/platforms are opportune for disruptive disintermediation via blockchain technology.

ENABLEMENT

Platforms disintermediate pipelines

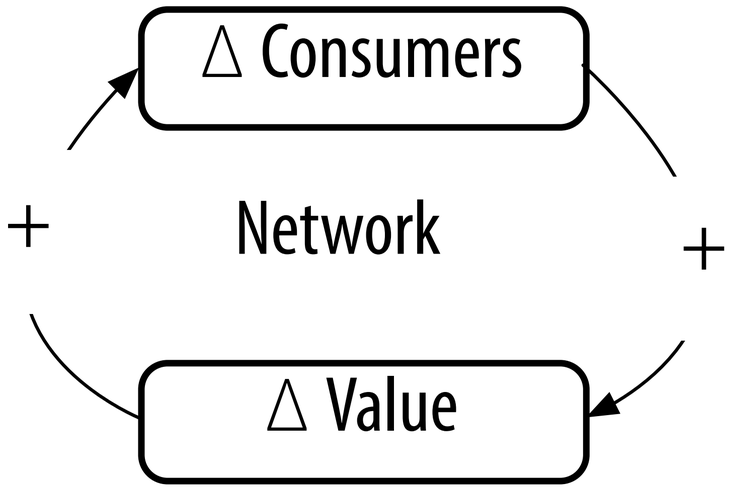
Distributed network computation enables platforms

Distributed consensus enables trustworthy platforms

Governance matters

Distributed AI provides scalable super-efficient user controlled re-intermediation

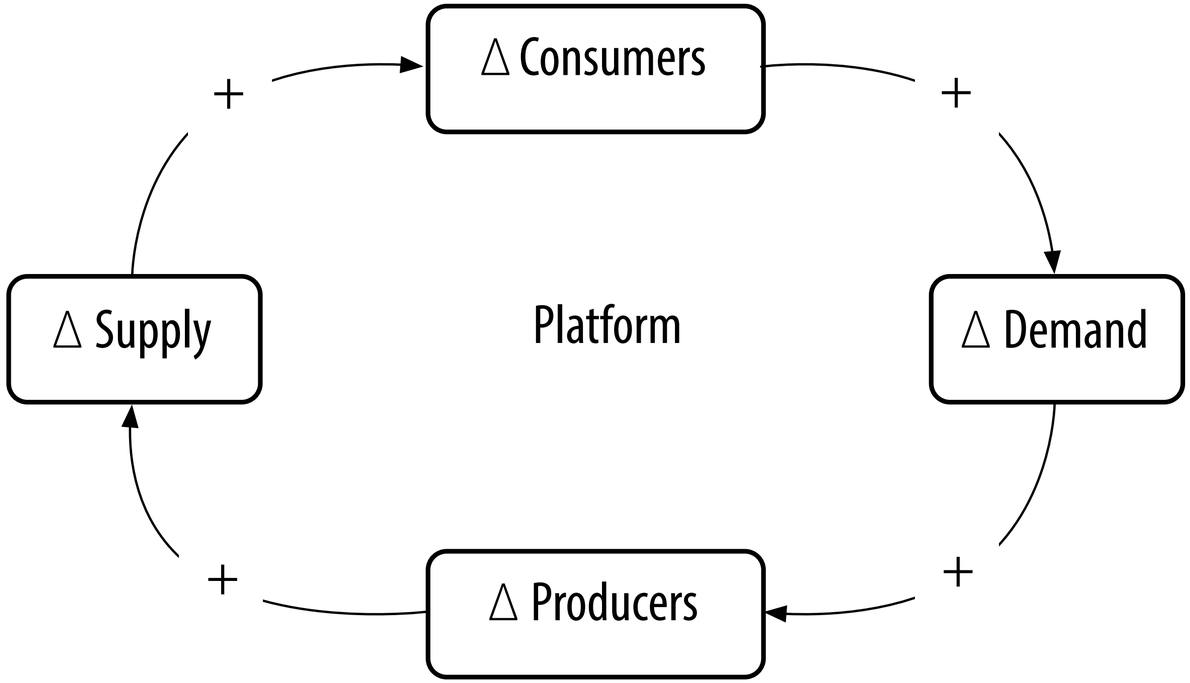
SINGLE-SIDED NETWORK EFFECT



More consumers increases value which attracts more consumers

Demand side driven

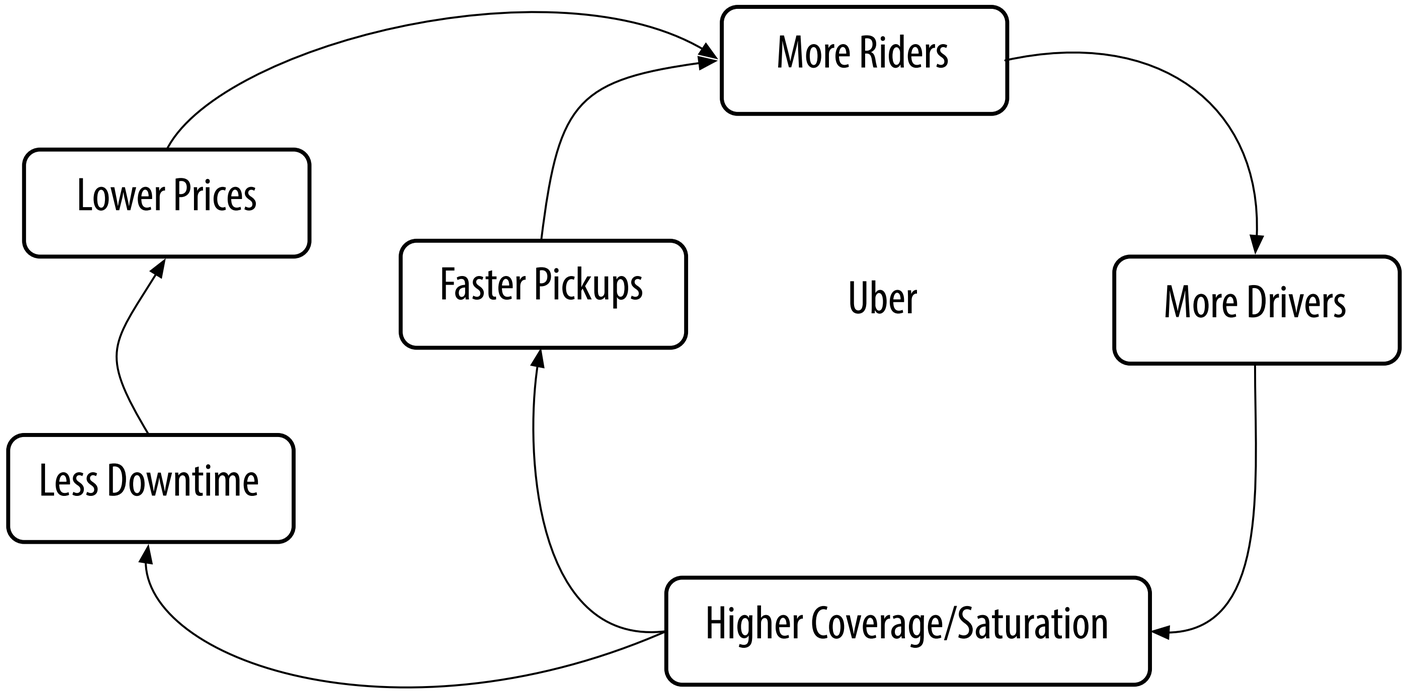
TWO-SIDED NETWORK



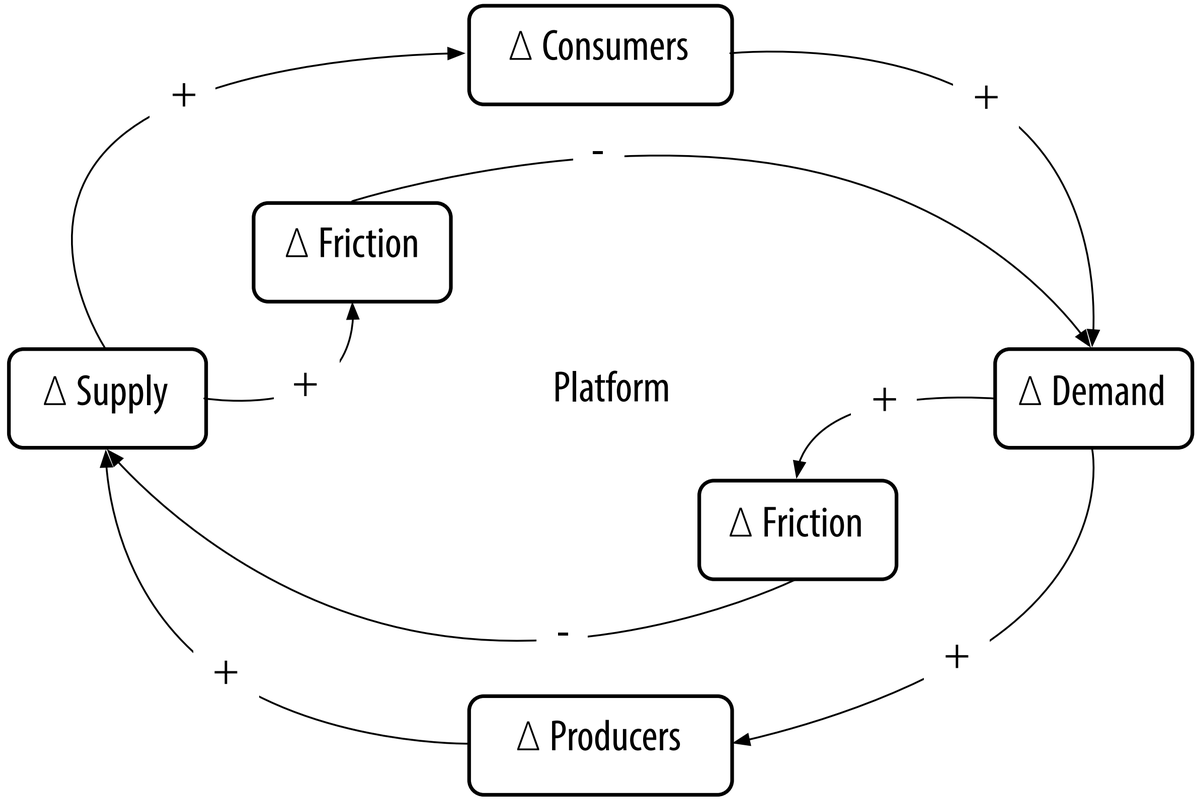
more consumers drive demand which attracts more producers

more producers drive supply which attracts more consumers

EXAMPLE


More Riders
Lower Prices
Faster Pickups
Uber
More Drivers
Less Downtime
Higher Coverage/Saturation

NEGATIVE CROSS-SIDE NETWORK EFFECTS



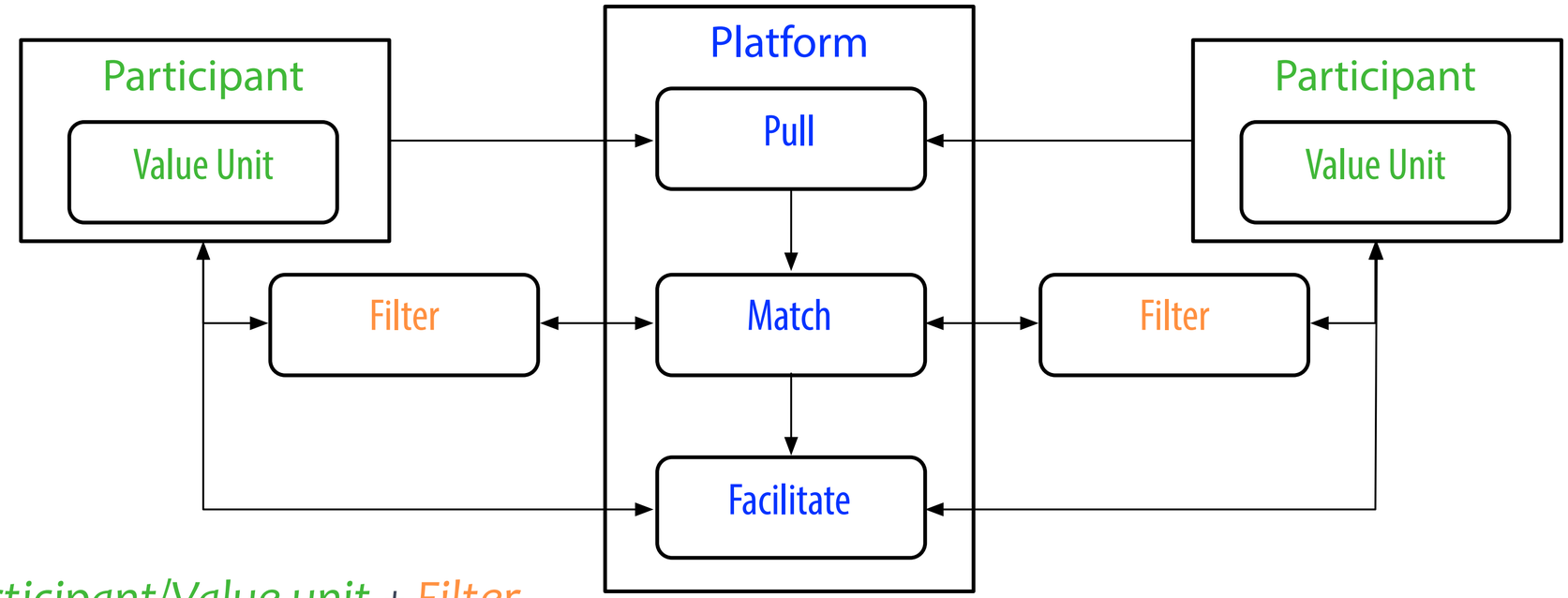
More supply choice increases friction e.g. customer confusion in producer selection thereby decreasing demand

More demand choice increases friction e.g. producer failure in customer satisfaction thereby decreasing supply

PLATFORM BUSINESS MODEL

Supply economies of scale (production efficiency)
replaced with
Demand economies of scale (network effect multipliers of value)

Two-sided network effects



Core Interaction = Participant/Value unit + Filter

Platform = Pull + Match + Facilitate

CURATION

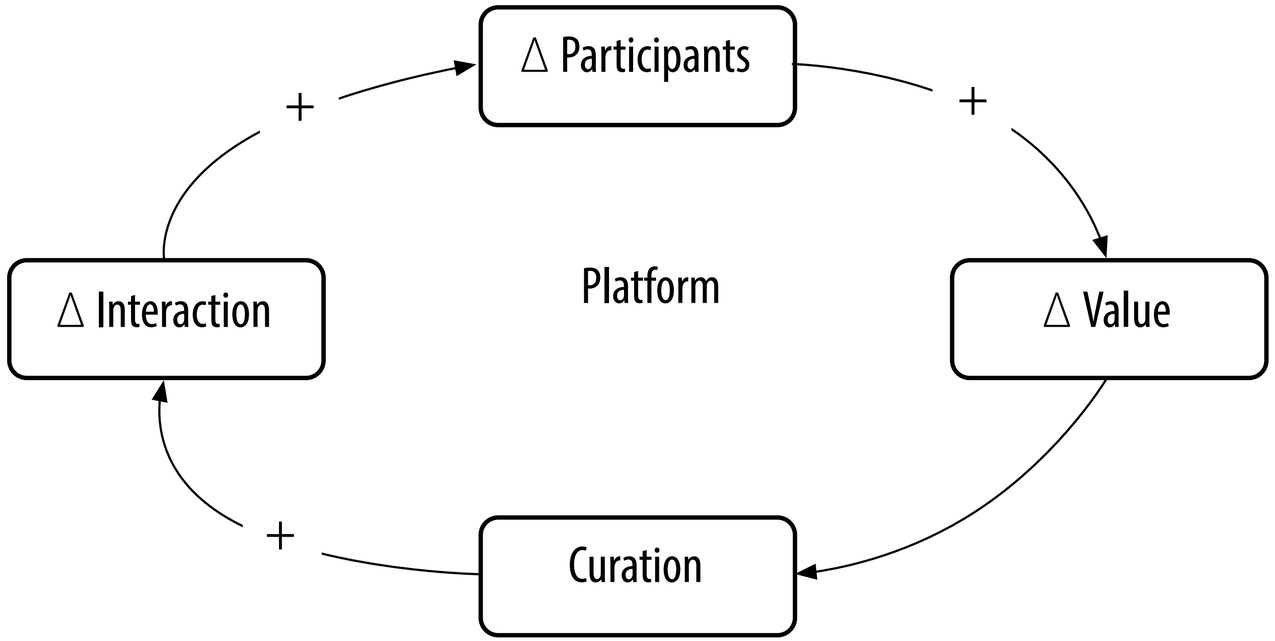
Match + Filter = Curation

Reduces negative cross-side network effects

Enhances positive cross-side network effects

Essential enabling capability for any platform

Curation is applied reputation



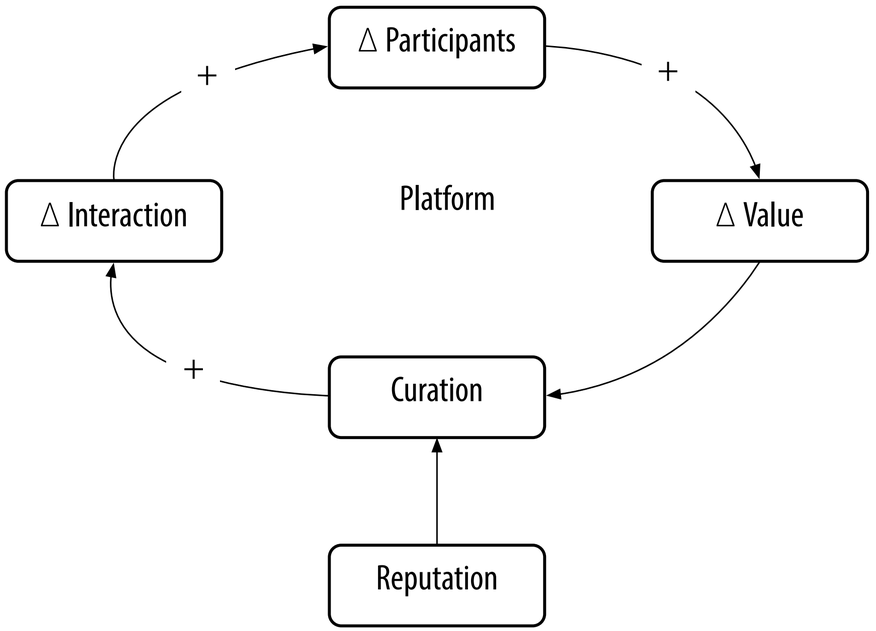
META-PLATFORM

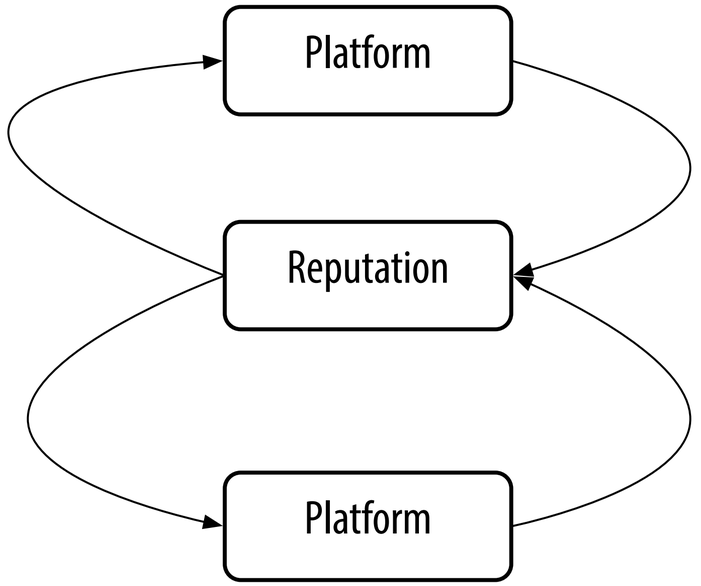
Platform that facilitates two-sided network effects across and amongst other platforms

Portable Reputation is a potential meta-platform

Contextual

Transitive





Meta-platform may be a system of network intelligence

Network effects on network effects

NETWORK EFFECTS

1-Sided *Direct*:
- Physical: Phone Company
- Protocol: Ethernet, HTTP
- Personal: Influencers
- Market: (N-Sided)

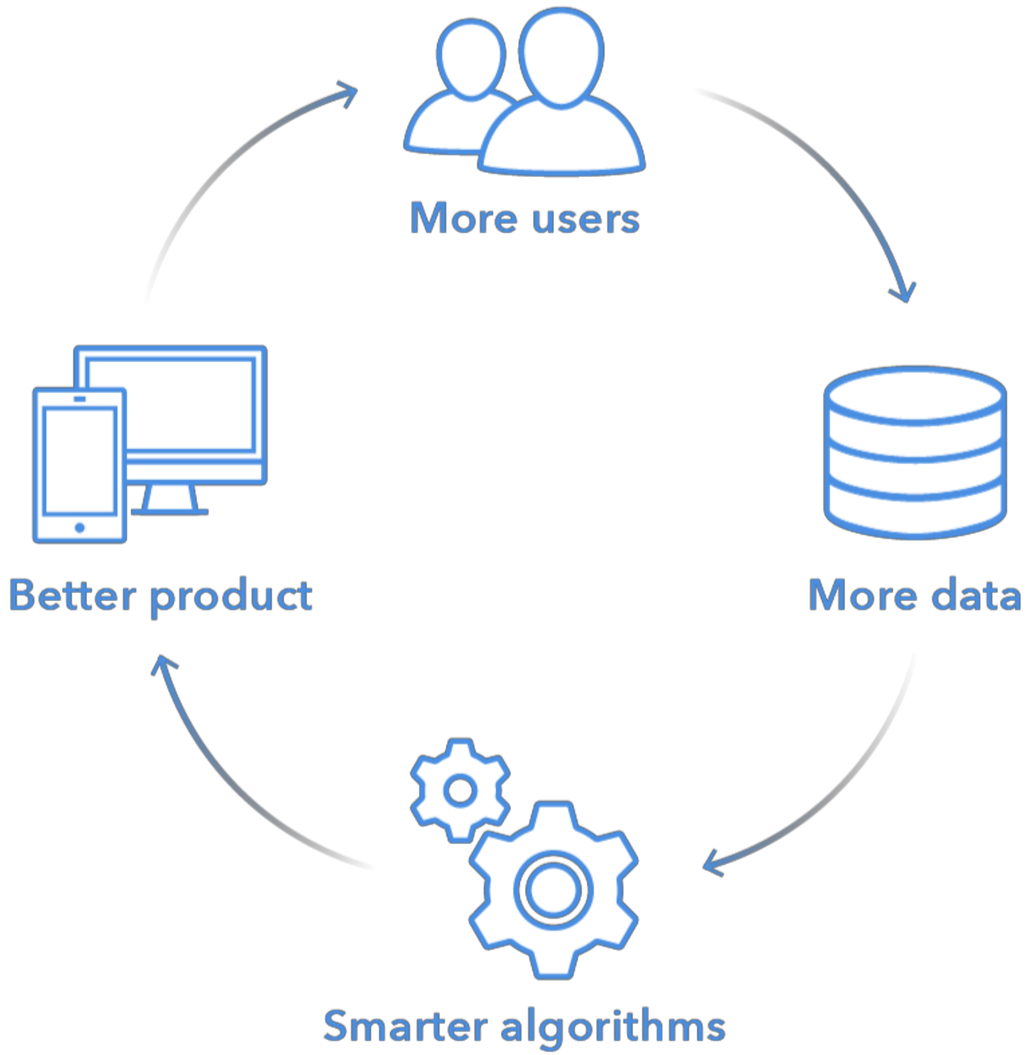
2-Sided *Indirect*: (Demand and Supply Sides)
- Marketplace: Craigslist, AirBnB
- Platform: (unique supply) iOS App Store
- Asymptotic Marketplace: 2-Sided: Uber, Lyft

Data: (Product value increases with more data driving more users)

Technology Performance: (More users increases product value)

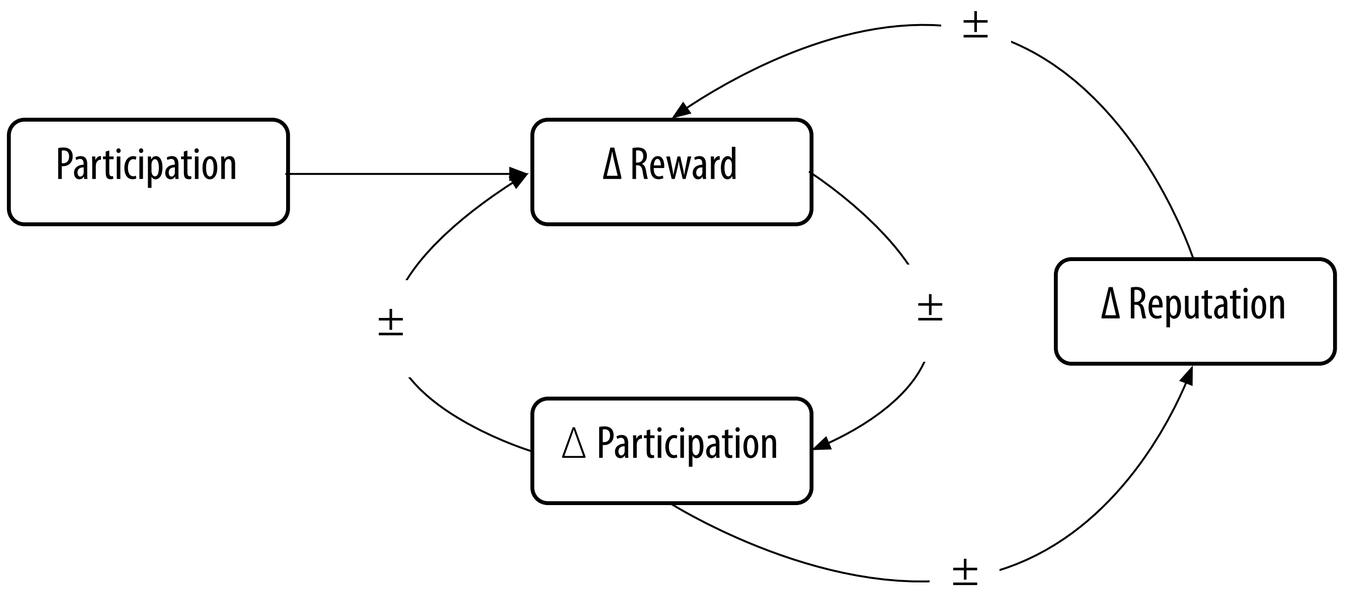
Social: (Interaction between users produces value)
- Belief
- Language
- Bandwagon

DATA NETWORK EFFECTS



REPUTATION DRIVEN INTERACTION

Graduated Participation Interfaces



REPUTATION

noun:

The estimation in which a person or thing is held, especially by the community or the public generally.

root:

Latin word reputāre, which is equivalent to re + putāre, that is, to re-think or re-consider.

usage:

A considered evaluation (measure) of past behavior used to predict future behavior.

qualification:

Confidence improves with contextual similarity.

https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/open-reputation-low-level-whitepaper.pdf

WHAT IS REPUTATION?
WHAT IS REPUTATION AI?

Contextual predictor of future behavior to enable a transaction

Closed-loop automated reasoning, not just open-loop pattern recognition

Means to filter and modulate transactions

Curator, recommender, decision aid, IA

Contextual predictors are more powerful

Behavior based predictors are more credible

Transitive predictors are more portable

COMPUTATIONAL REPUTATION

Computational generation of a reputation is to aggregate relevant instances of behavior.

Instances of behavior = reputational events or reputes for short

Reputation measures are inferred indirectly from reputes associated with an entity

Contrasting Example:

* Entities provide direct ratings of promptness of another entity

* Collect instances of behavior of entity in context from which promptness can be inferred.

Reputation from reputed behavior allows for re-scoping, re-weighting, re-combination, and re-evaluation of collected reputes

Enables arbitrary levels of nesting, precision, and granularity in the data aggregation process

https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/open-reputation-low-level-whitepaper.pdf

PORTABLE IDENTITY

Security, Privacy, Agency

(Trustworthy, Private Preserving, Self-Sovereign)

Portable Identifiers & Attributes

Decentralized (not in a silo)

PORTABLE REPUTATION

Security, Privacy, Agency

Portable Data & Algorithms

Reputation system that spans verticals and applications

Benefits from data network effects

OPEN & PORTABLE & PROPRIETARY

Open Data Formats

Open Frameworks

Open Algorithms

Open Governance

Proprietary Contextualization, Parametrization, Tuning & Data

META-PLATFORM

Two Sided Network Effects

Curation, Filtering, Modulation

Optimal Control

Graph Based Self Identity/Reputation

Identity = *Identifiers* + *Attributes*

Identifiers = *globally unique decentralized cryptonyms* + *aliases*

Attributes = *user data, proofs*

Facilitate attribute exchange between entities sufficient to enable transaction to proceed

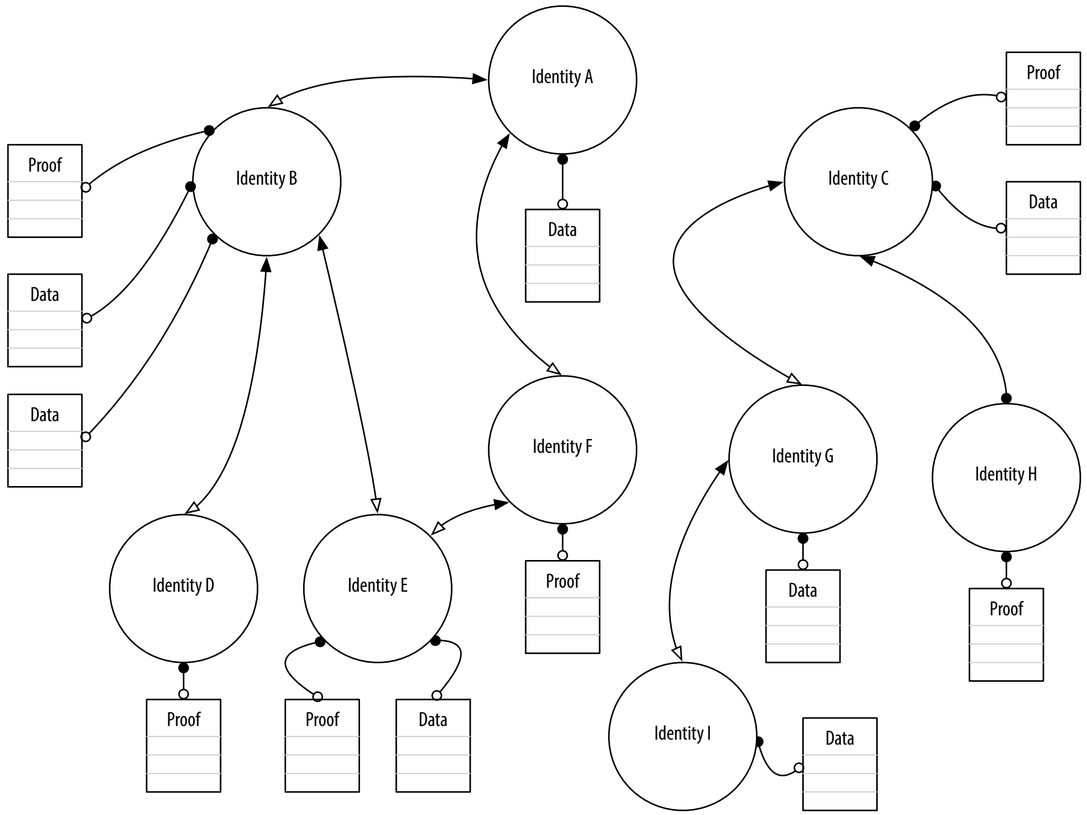
Identity System Features:

- Agency (own your own identity)
- Security (impervious to fraud)
- Privacy (least disclosure)

Agency = portable identifiers + user controlled

Security = distributed consensus + modern crypto

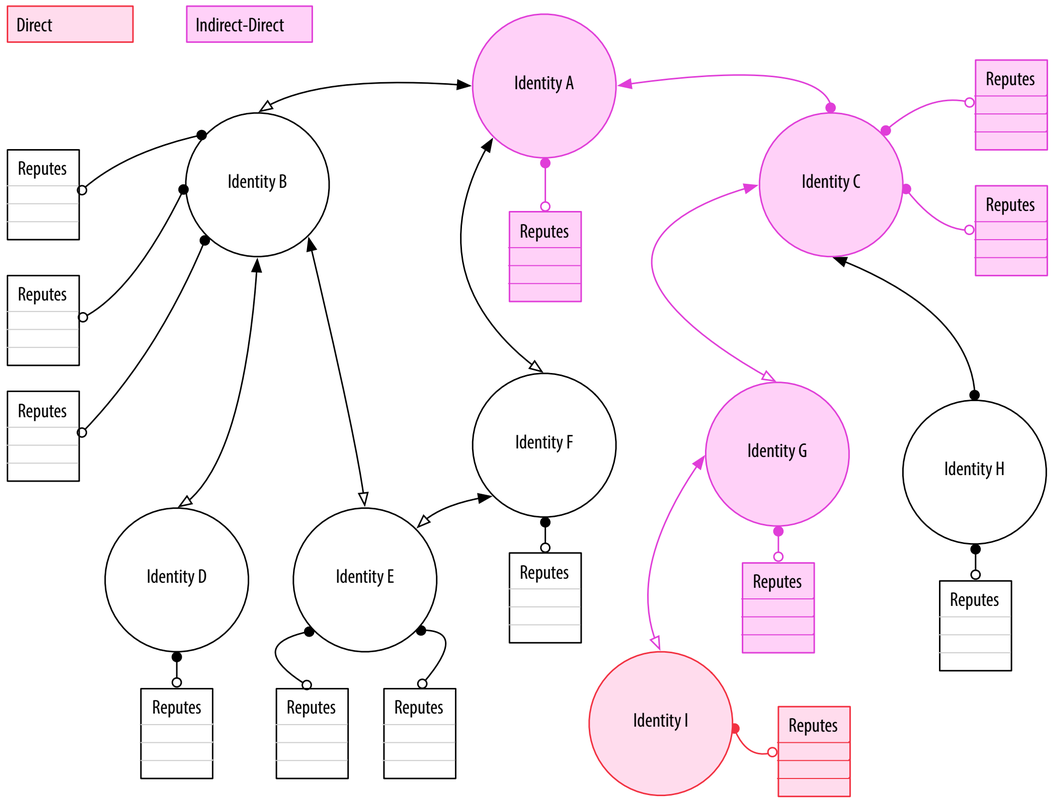
Privacy = granular graph based identities + layered disclosures + zero knowledge disclosures + group identities



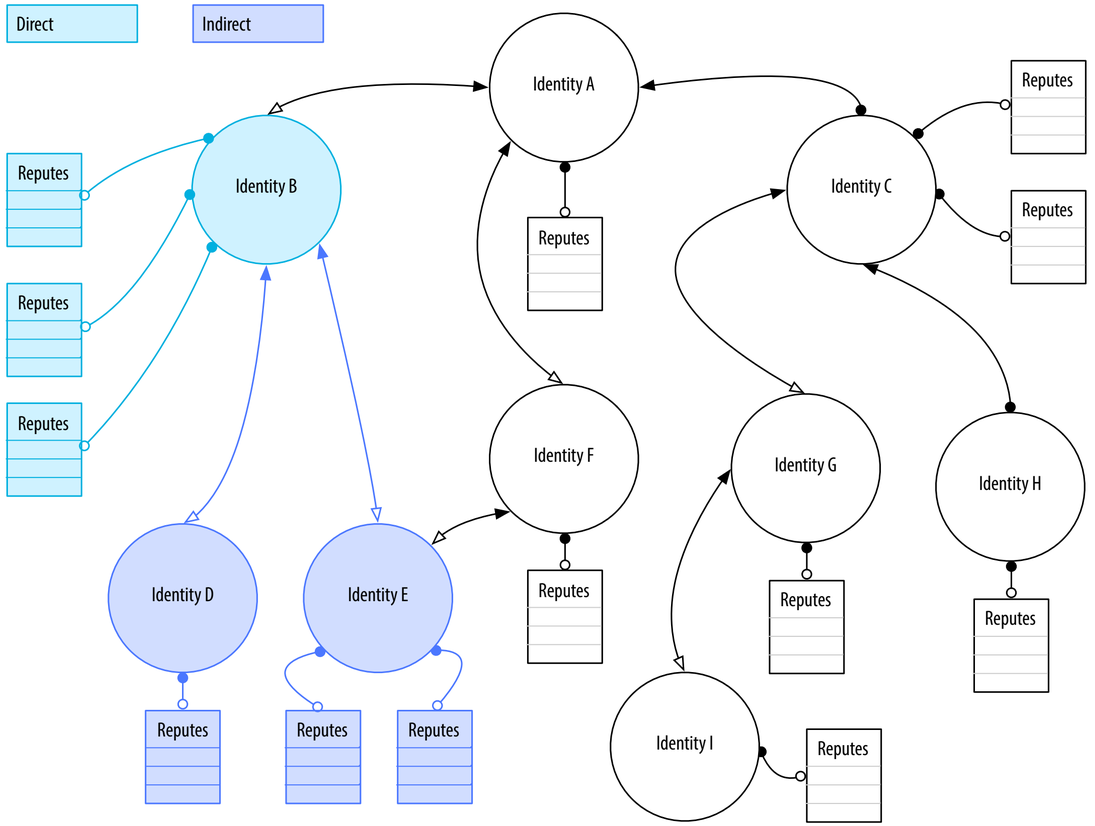
https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/Identity-System-Essentials.pdf

Graph Based Other Identity/Reputation

Repute Down



Repute Up



https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/Identity-System-Essentials.pdf

IDENTIFIERS

UUID: Universally Unique Identifier RFC 4122: UUID type 1 -5

16 byte collision resistant decentralized identifier generated with random number generator and optional name spacing data

Enables distributed applications to create unique identifiers without central authority

Prefixed namespacing allows for sorting and searching properties such as time order, lexical order, nesting etc,

URI: Uniform Resource Identifier, URI: Uniform Resource Locator, URN: Uniform Resource Name RFC 3986

```
scheme:[//[user[:password]@]host[:port]][/path][?query][#fragment]
```

Self-Certifying Identifier: Contains fingerprint of public member of cryptographic public/private key pair

Decentralized Self-Certifying Identifier: public/private key pair is generated by user not central registry

Hierarchical Self-Certifiing Identifier: selfcertroot:/path/to/related/data

Tree Hierarchical Deterministic Self-Certifying Identifier: parent/child/child/child

Tupleizable (routable) Identifiers: /channel/host/process/data = (channel, host, process, data)

DID: DECENTRALIZED IDENTIFIER

https://github.com/w3c-ccg/did-spec

https://w3c-ccg.github.io/did-spec/

```
did:method:idstring
did:rep:Qt27fThWoNZsa88VrTkep6H-4HA8tr54sHON1vWl6FE=
```

```
{
  "id": "did:rep:Qt27fThWoNZsa88VrTkep6H-4HA8tr54sHON1vWl6FE=",
  "friend": “did:igo:Xq5YqaL6L48pf0fu7IUhL0JRaU2_RxFP0AL43wYn148=/next/door”,
  "name": "John Doe",
  "zip": "94088"
}
```

```
did:method:idstring/path/to/associated/resource
```

```
did:method:idstring/path/to/associated/resource#05
```

```
{
  "id": "did:rep:Qt27fThWoNZsa88VrTkep6H-4HA8tr54sHON1vWl6FE=",
  "friend": “did:igo:Xq5YqaL6L48pf0fu7IUhL0JRaU2_RxFP0AL43wYn148=/next/door”,
  "name": "John Doe",
  "zip": "94088"
}
```

Zero Trust Computing

Diffuse trust perimeter-less security model

Security, Privacy, Agency

Diffuse trust perimeter-less security principles

The network is always hostile, internally & externally; Locality is not trustworthy.

Every network interaction or data flow must be authenticated and authorized using best practice cryptography.

Inter-host communication must be end-to-end signed/encrypted and data must be stored signed/encrypted; Data is signed/encrypted in motion and at rest.

Policies for authentication and authorization must be dynamically modified based on behavior.

Policies must be governed by distributed consensus.

Decentralized Identity Inverts Service Architectures

Conventional (centralized):

- Server creates identifiers (GUID, Database primary keys)
- Server timestamps
  - event ordering relative to server
- Server manages keys,
  - AuthN/AuthZ is indirect via client to server proxy
- Perimeter Security
- Server is source of truth
  - Server controls changes/updates to resources
  - Signed at rest problematic
  - Encrypted at rest problematic
- Server's role is 2nd party in two party transactions between client to server to client.

Unconventional (decentralized):

- Client creates identifiers (DIDs)
- Client timestamps
  - event ordering relative to client
- Client manages keys
  - AuthN/AuthZ is direct peer-to-peer
- Perimeterless Security
  - Client is source of truth
  - Client controls changes/updates to resources
  - Server cannot make changes
  - Client signs at rest
  - Client encrypts at rest
- Server's role is either:
  - Trusted 3rd party in 3 (multi) party transactions between 2 (or more) clients and server
  - Agent or proxy for a client in two party transaction with another client.

Heavyweight — Decentralized Identity — Lightweight

Decentralized Identifiers
Portability
Verifiable Claims
- Certification
- Consent Decrees

Ledger
- Revocation
- Arbitration
- Provenance

Key Management
- Hierarchical Keys
- Key Recovery
- Key Hiding
- Multisignature
- Group Keys

Identity Graph
- Attribute Based Identity
- Contextual Identity
- Transitivity
- Group Identity
- Least Disclosure

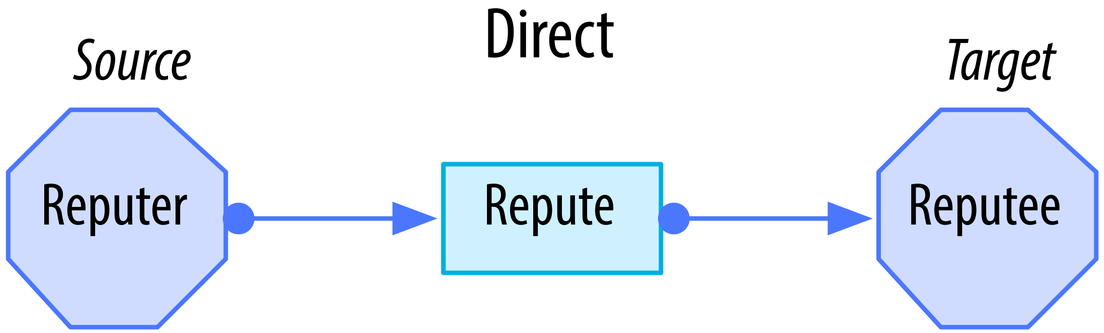
Decentralized Identifiers

Heavyweight — Reputation — Lightweight

Portable
Transitive
Cross-Domain

Eventually Portable
Domain Specific

REPUTING LEXICON



Repute = *Reputational Event (Data)*

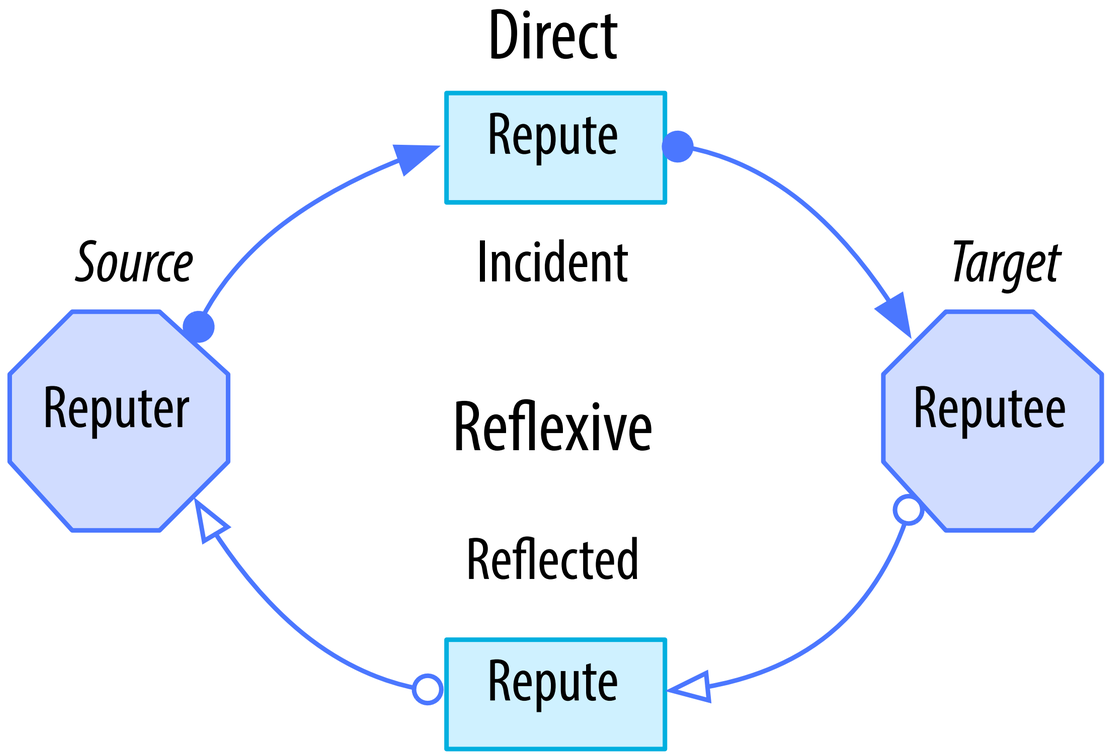
Reputee = *Reputational Entity, (Identity)*

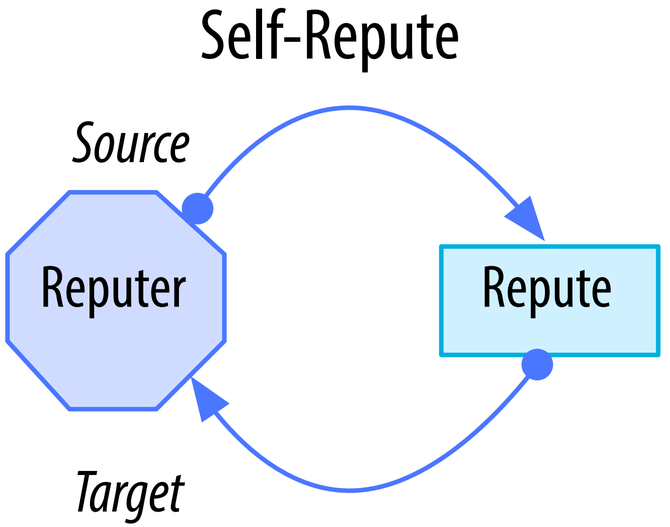
Target of Repute

Reputer = *Source of Repute*

https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/open-reputation-low-level-whitepaper.pdf

REFLEXIVITY





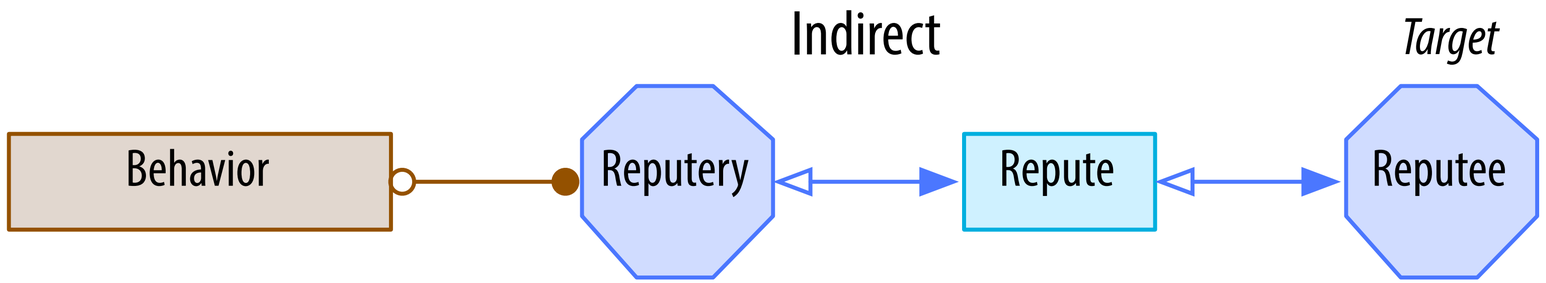
Reputation is Reflexive:

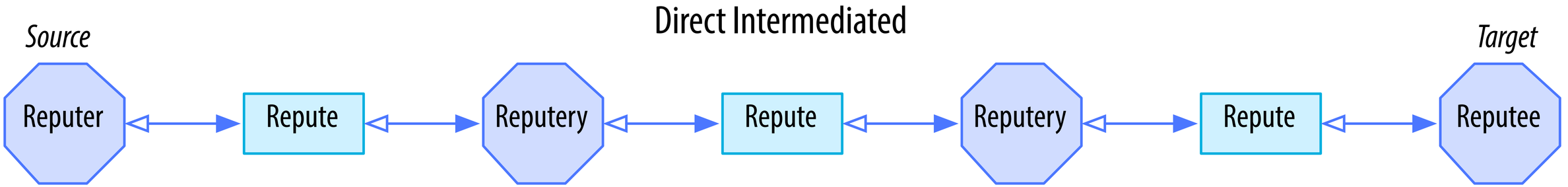
Reputer is simultaneously both a Source and a Target = Reputee

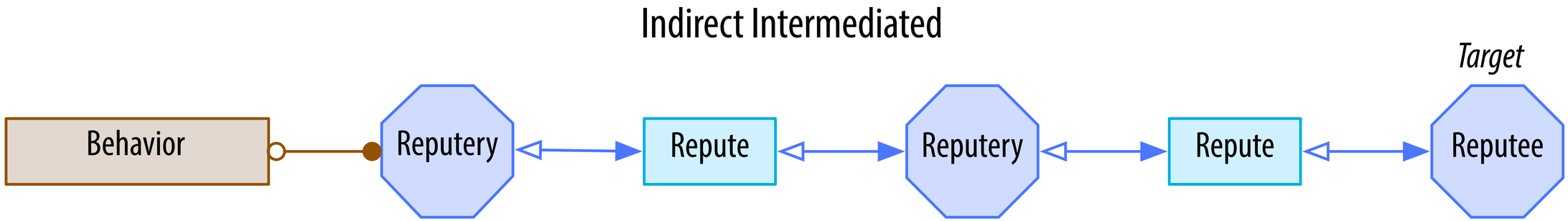
https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/open-reputation-low-level-whitepaper.pdf

INDIRECT AND INTERMEDIATED REPUTING

Reputery = *Indirect Source (Implied Reputer) or Intermediate Source*







https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/open-reputation-low-level-whitepaper.pdf

REPUTAGE

Reputage: *reputational event ancilliary data. Optional or infrequently used information associated with a repute separated from core repute for performance reasons.*

https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/open-reputation-low-level-whitepaper.pdf

REPUTET

Reputet: *reputational event transaction. A cryptographically signed and validated transaction of reputes between reputees or reputeries.*

Initiator = *first party.*

Copartent = *counter party.*

Arbiter = *trusted third party (notary).*

https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/open-reputation-low-level-whitepaper.pdf

REPUTE

```
{
  "ruid": "bcd456",
  "stamp": "2015-03-19T10:30:45Z",
  "reputee": "z4def6",
  "reputer": "5efa75",
  "curator": "2bcd4",
  "signer": "2bcd4#0",
  "detail":
  {
    "rating":
    {
      "useful": 90,
      "fair": 80,
    },
    "url": "http://myblog.com/article19/"
  }
  "tags": {},
  "reputage": null,
}
/r/n/r/n
"abcdef987654321"
```

```
{
"puid": "abcefg",
"ruid": "bcd456",
"stamp": "2015-03-19T10:30:45Z",
"comment":
{
  "cuid": "1234abc",
  "url": "http://myblog.com/article19/comment19",
  "contents": "You are so awesome."
}

}
/r/n/r/n
"abcdef9871234567"
}
```

Reputation Business Models

Reputation as a Service (RAAS)

Distributed Autonomic Service (DAS)

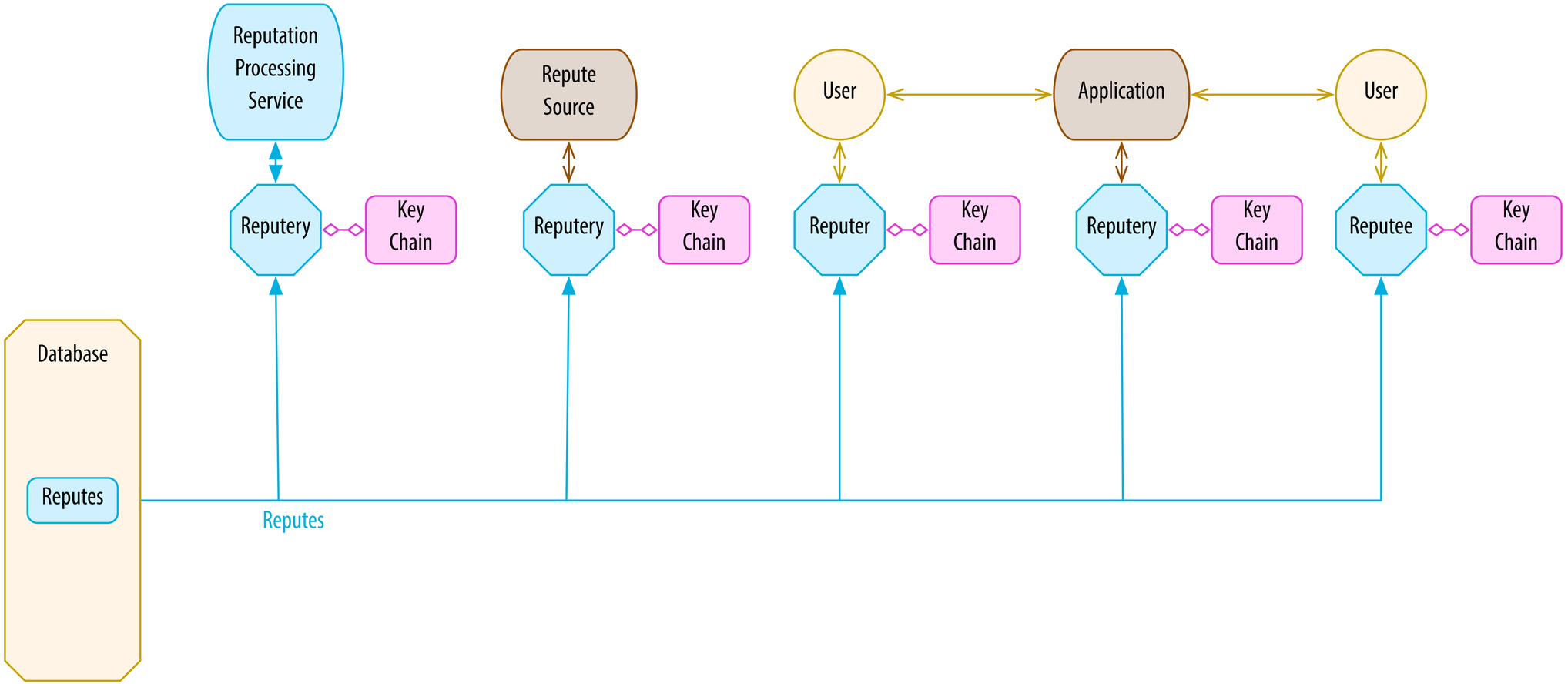
Service using autonomic computing algorithms on scalable decentralized computing infrastructure managed by distributed consensus.

distributed AI

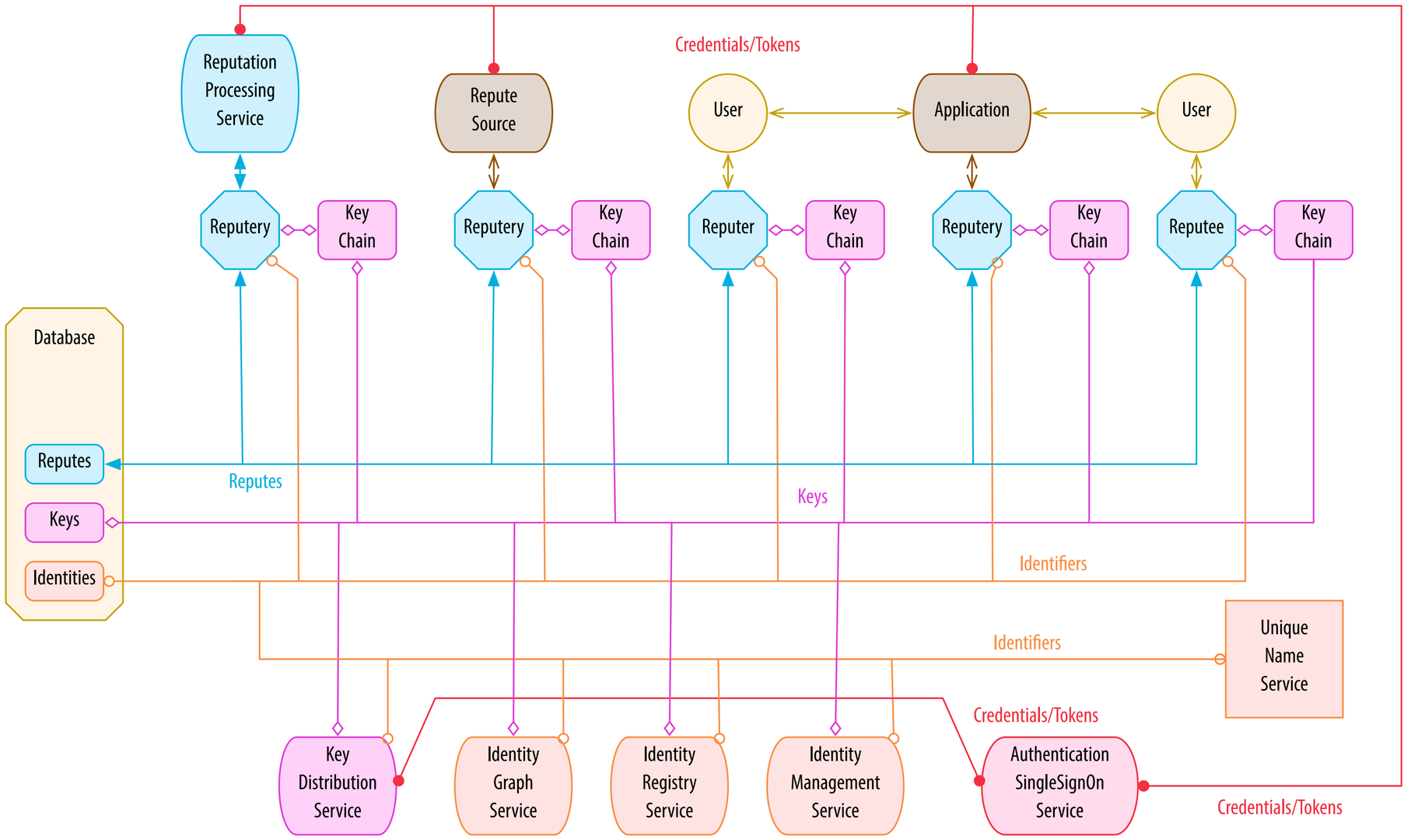
RAAS on a DAS

autonomic = self-managing, self-configuring, self-healing, self-optimizing, self-securing

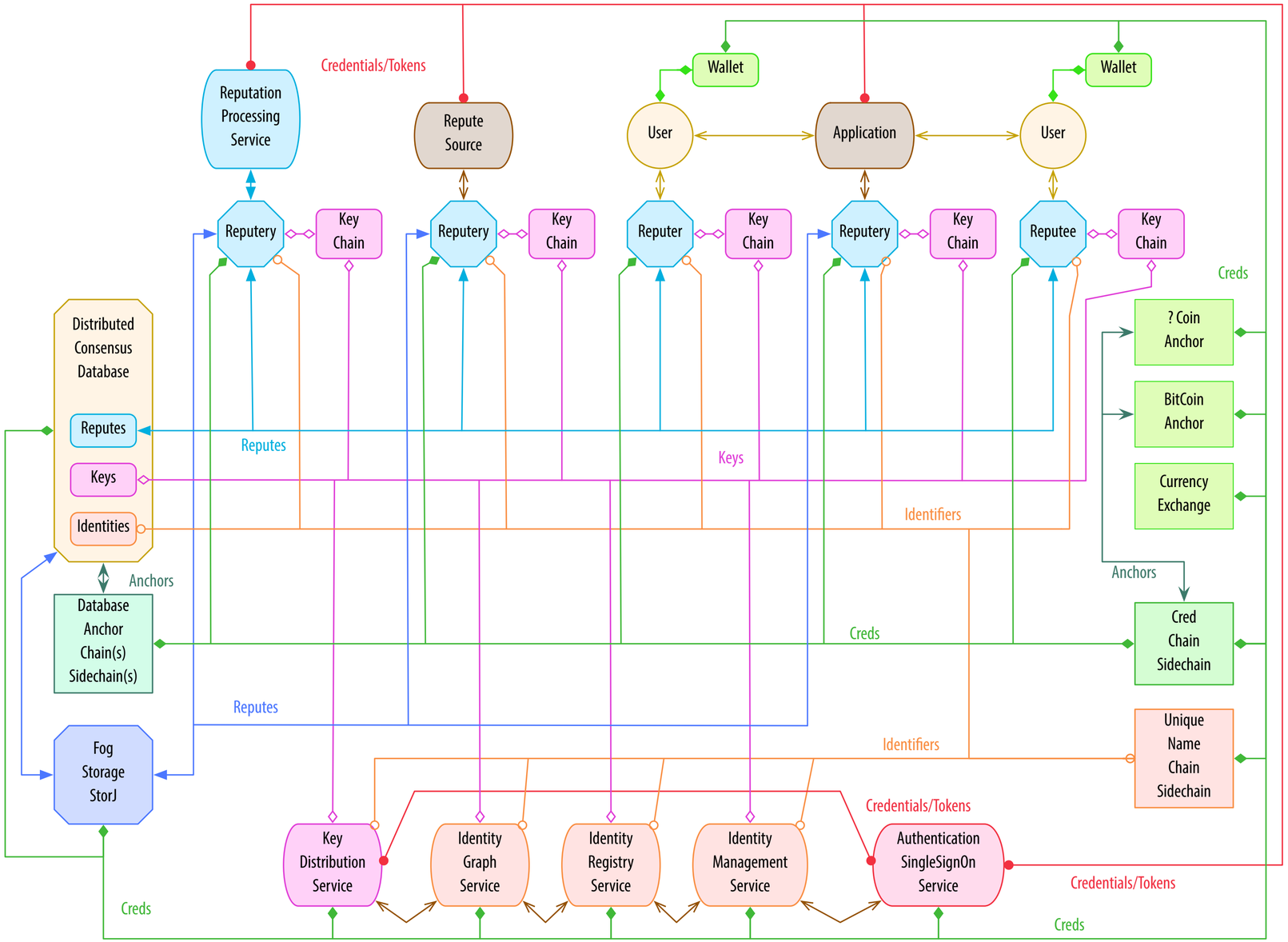
BASIC REPUTATION



Reputation & Identity

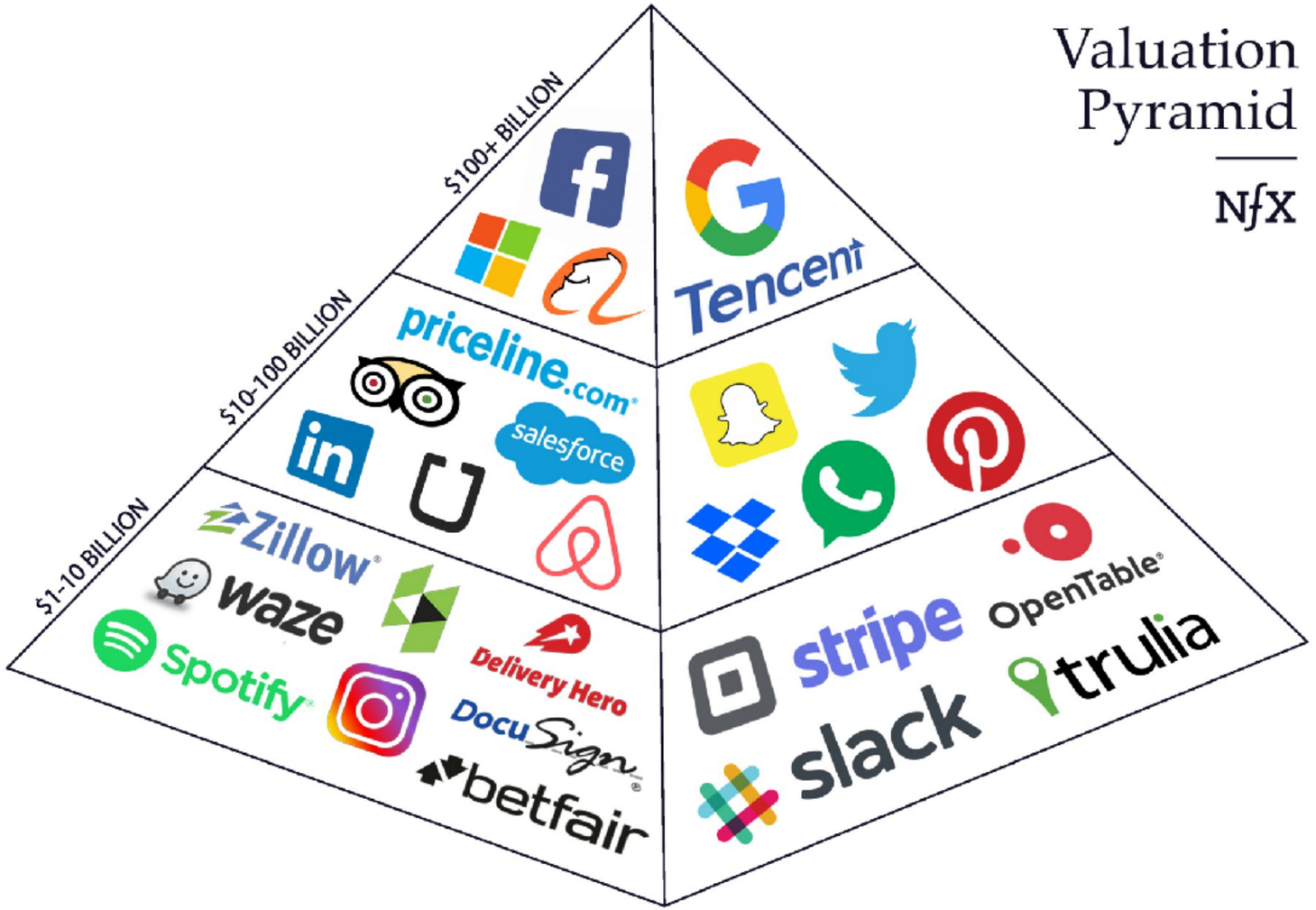


Decentralized Reputation System



https://github.com/SmithSamuelM/Papers/blob/master/whitepapers/open-reputation-low-level-whitepaper.pdf

NETWORK EFFECTS



NETWORK EFFECTS MAP