Data Portability in Practice

Babak Jahromi, Microsoft

Jan 2019

CPDP

Data Portability is non-Trivial!

Data syntactic portability

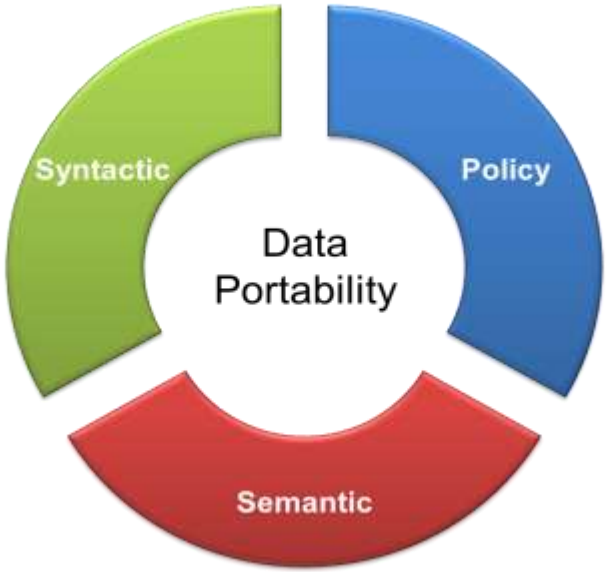
Transferring data from a source system to a target system using data formats that can be decoded on the target system. Data is structured according to the data model defined by the semantic facet, and is encoded using a particular syntax, such as XML.

Data semantic portability

Data semantic portability is defined as transferring data to a target such that the meaning of the data model is understood within the context of a subject area by the target.

Data policy portability

Data policy portability is defined as the ability to transfer data between a source and a target while complying within the legal, organizational, and policy frameworks applicable to the source and target. This includes regulations on data locality, rights to access, use and share data, and mutual responsibilities with respect to security and privacy between a CSP and a CSC.



From ISO/IEC 19941 feely available[1]

Even harder!

AUTOMOATED TRANSFER of USER DATA

A test of utilizing power of
Open Source community

Examples of **technically feasible** direct transfer of personal data

Photos/Files

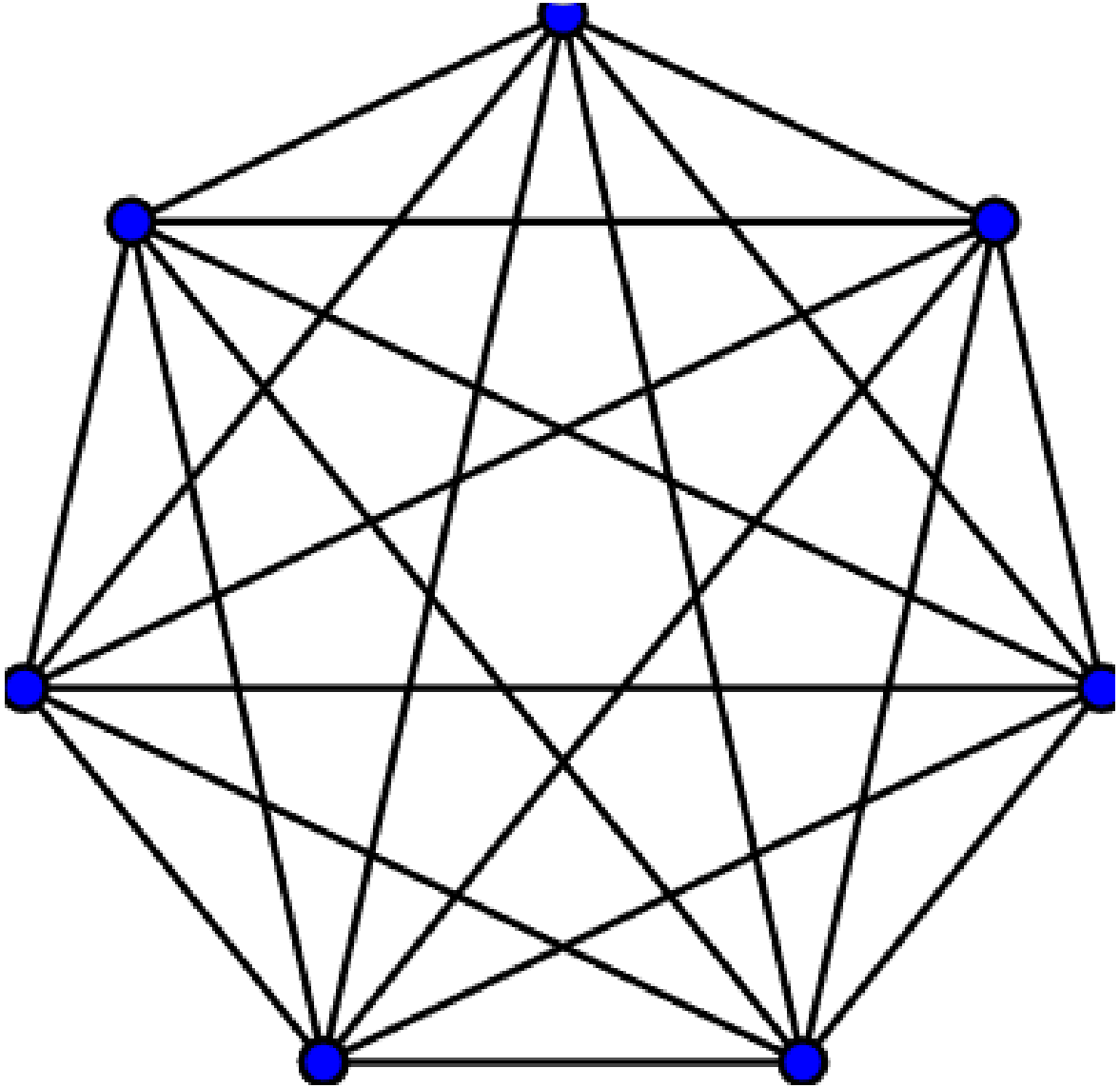
Contacts list

Calendar

Emails

Scale problem with **Direct Transfer of Data**

- Direct transfer of data requires any two providers to design, build and test together
- For N providers, there are N*(N-1)/2 such projects (O(n2))
- There is obviously a scale problem



There are [N*(N-1)/2] direct data transfers;
A problem of O(n2)

How to Determine Technical Feasibility

01

Open Process

02

Open Technologies

03

Open Source Code

04

Robust participation by all stakeholders

1

Commitment to developing techniques to address direct data transfer when technically feasible

2

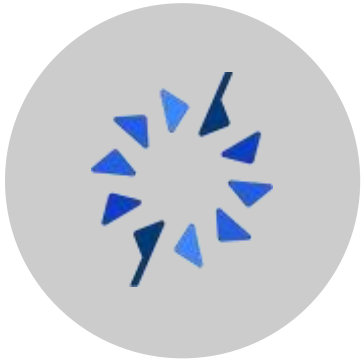
Implementing direct data transfer across a virtually unlimited number of possible pairs is complex!

3

Open source is the best way to go.

Underlying principles for the project

Data Transfer Project



- Project Page
  - https://datatransferproject.dev/
- Developer-friendly, open source
  - Hosted at GitHub (https://github.com/google/data-transfer-project)
  - Apache 2.0 License
  - Java-based
- Backed by key industry players

Use Case Categories

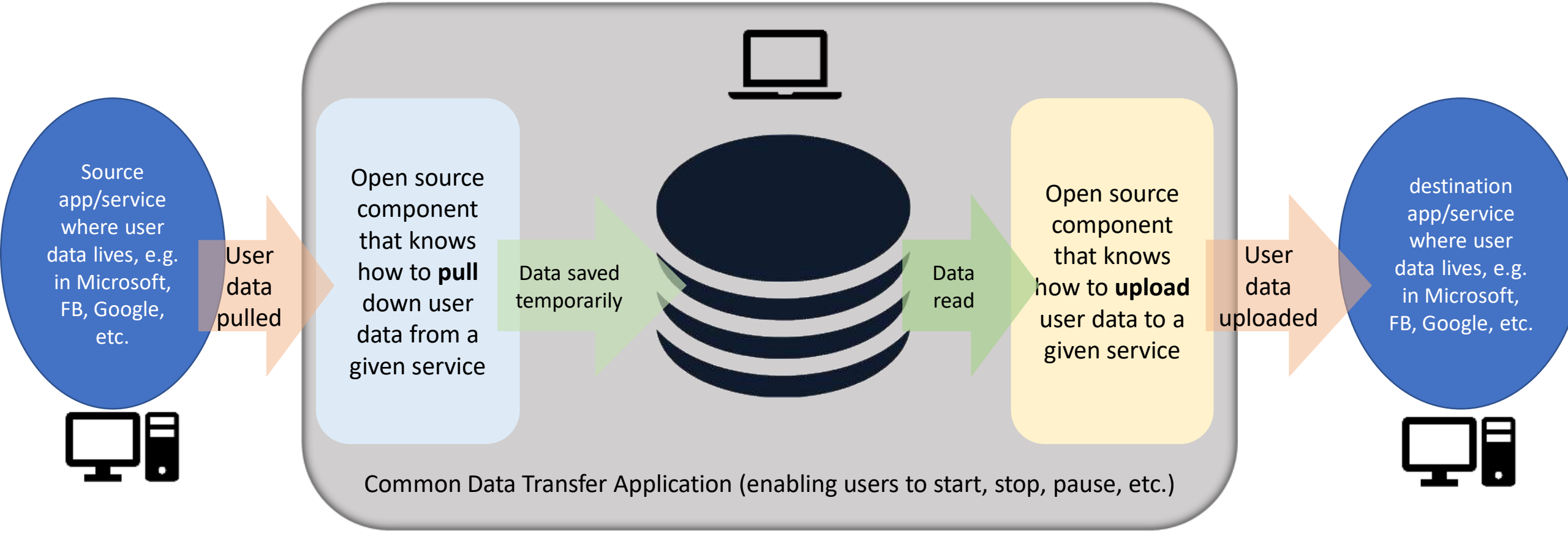
User-initiated service transfer

- A user decides to switch from one service provider to another and wants to import their data to the new service
- Including (but not limited to) contacts, photos, tasks, email, and derived data

Partner service enablement

- Based on user-consent, a service provider shares specific data on a one-time basis with another provider to complete a task
- For example, location and driving distance data is shared from a mapping service to an insurance company for the purposes of providing an accurate quote

An Open Source approach to realize the right to have personal data transmitted directly when technically feasible



"Pull" code is generic for each source service
"Upload" code is unique to source and receiving service

01

For each pair of source/destination transfer, there is a pair of open source components

02

The “source” vendors are encouraged to “seed” the open source repository by contributing reference components showing how data can be pulled or published from or to their services

03

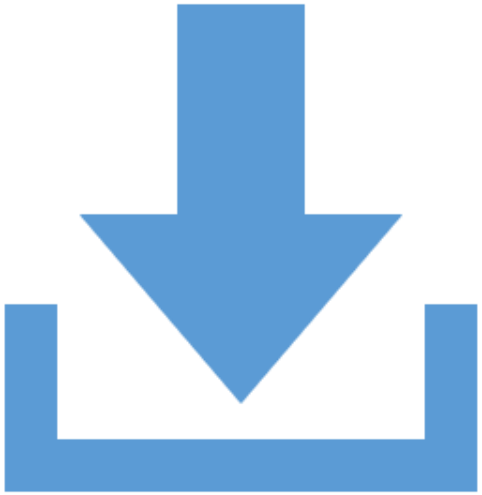
Other stakeholders are encouraged to use the “seed” contributions to enrich the functionality

04

Common application(s) hosting the components provide runtime and user interface for users to initiate data transfer requests

So what is unique about this idea?

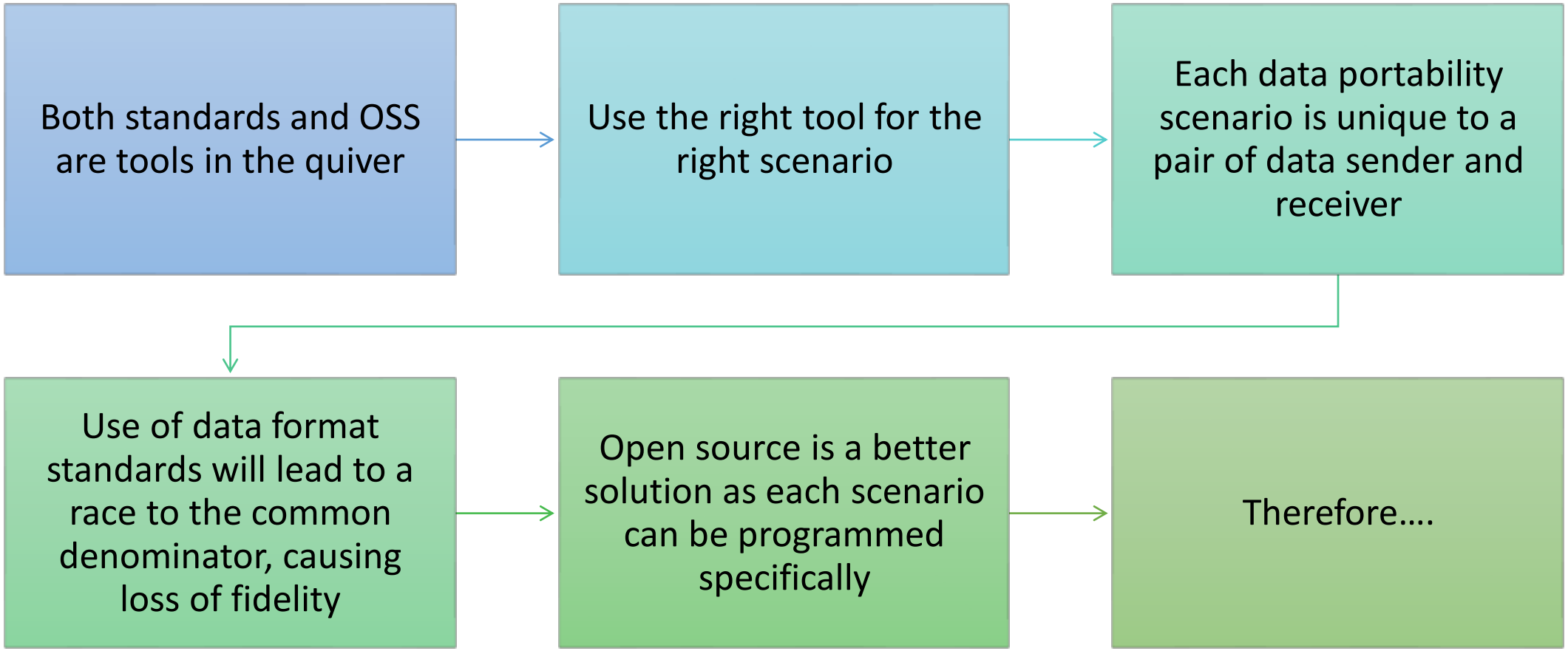
Data Transfer Program Details



1. Proof of concept for direct controller to controller transfer using the pull and upload model between two participating partners. For example, transfer of calendar and/or contacts data from Microsoft to/from another email/calendar service vendor
2. Participating partners provide and document the export (pull) capabilities
3. Participating partners provide a demonstration pull component for their own service that can be forked and used by other services to create pull capabilities. This pull application is based on the technical specification for data download and persistence.
4. Support advocacy for other stake holders to build pull/upload services for each pair of participants of the program

Appendix

Open Source vs. Open Standards: Data Portability Scenarios



Summary of facets of cloud data portability

Facets	Aim	Objects	Requirements	Examples
Data Syntactic	Receiving data in a machine readable, structured and commonly used format	Data	Common machine readable data format	XML, CSV, JSON
Data Semantic	Assured meaning of data	Data	Mutually understood ontologies and metadata	OWL, Dublin Core Schema
Data Policy	Adhering to all applicable regulations and organizational policies	Regulatory and organizational policy	Agreed set of applicable regulations and organizational policies	Confidentiality levels, privacy rights, cross border transfer

Considerations for portability of “Derived” data

Topics	Considerations
Extract and erase	Unlike cloud service customer data, which is assumed to be portable and erasable under the CSC’s control, the ability to extract cloud service derived data from the system for use by the CSC, or to enable erasure of some cloud service derived data by the CSC, is likely to need careful control and is subject to agreement between the CSC and the CSP.
Regulations	Regulations law can apply to some types of cloud service derived data. For example, some types of log information might have to be retained for a specified period and cannot be deleted.
Categorization	The detailed categorization of cloud service derived data in the taxonomy defined in ISO/IEC 19944 is intended to support the definition of the cloud service agreement between the CSC and CSP. This data categorization can be utilized when CSPs and CSCs engage in defining the portability requirements of cloud service derived data. In such cases the agreement may reference specific sub-types of derived data in the portability discussions and agreements.
Scope	Cloud service derived data has potential meaning outside the cloud service (otherwise it would fall under the categorization of cloud service provider data) and the CSC may wish to access some data categories of derived data, or to request erasure of some categories of derived data.
Aggregation	Cloud service derived data collected by a CSP is sometimes aggregated with that of other CSCs, and in many cases is de-identified to remove PII. In such circumstances, providing the data records specific to a CSC and its users is technically challenging and adds risk to the confidentiality of other tenants.
Data minimization	Making some types of cloud service derived data available to the CSC could interfere with data minimization policies designed to protect privacy and confidentiality. These policies dictate shortened data retention periods, de-identification of data and erasure or masking of records not needed to provide the cloud service. Removing these policies across all types of cloud service derived data to permit future access or erasure by CSCs is often unacceptable.
Challenges	There are circumstances, such as CSC access to log files, where the provision of certain categories of cloud service derived data specific to the CSC is an important requirement. However, the technical challenges and risk of confidentiality and privacy to other tenants means provision or erasure of these types of cloud service derived data needs to be explicitly defined and carefully controlled.
Analytics	CSPs could run data analytics algorithms on cloud service customer data. The results could also be combined with cloud service derived data collected as the user interacts with the capabilities of the cloud service(s). Such a combination should still be treated as cloud service derived data but it might have lost relevancy to a given CSC. Such a combination could generate cloud service derived data that can be the basis for offering additional new insights to the CSCs and their users about their data via new features or improved capabilities of the cloud service(s). In many such cases, the cloud service derived data is used to create the new and improved capabilities and feature set of the cloud service, but by itself it is unlikely be useful to the CSC. Therefore, these categories of cloud service derived data might not be portable.
Graph data	Some applications develop social graph data that relate to cloud service users and other artefacts that are stored in the corresponding cloud service. Such data are unlikely to be portable as they are highly cloud service implementation specific and combine cloud service customer data and cloud service derived data from multiple users and other sources. The portions of the data that are meaningful outside the social graph and are part of cloud service customer data are normally available to the CSC.
PII	Care needs to be taken not to compromise PII of a natural subject as well as that of other associated natural subjects when porting cloud service derived data.

Types of data portability in the cloud

CLOUD DATA PORTABILITY FACETS	CLOUD CAPABILITIES TYPES		
	Infrastructure	Platform	Application
Data syntactic (5.2.2.2)	8.2.2[(1)]	8.2.3	8.2.4
Data semantic (5.2.2.3)	8.3.2	8.3.3	8.3.4
Data policy (5.2.2.4)	8.4		

(1): Clause numbers in ISO/IEC 19941

The Data Transfer Project (DTP)

An open source platform that enables service providers to offer ***consent-based*** data transfer in a ***secure, reliable,*** and ***scalable*** way

Consent-Based

- Users must explicitly consent with both the source and destination services
- Both providers must authorize transfer
- Automated
- Fully auditable

Secure

- All data flow is encrypted

Reliable

- Failure recovery
- Run across multiple datacenters

Scalable

- Architected by service providers with long histories deploying large-scale web services

DTP Platform

Telco Potential Use Cases

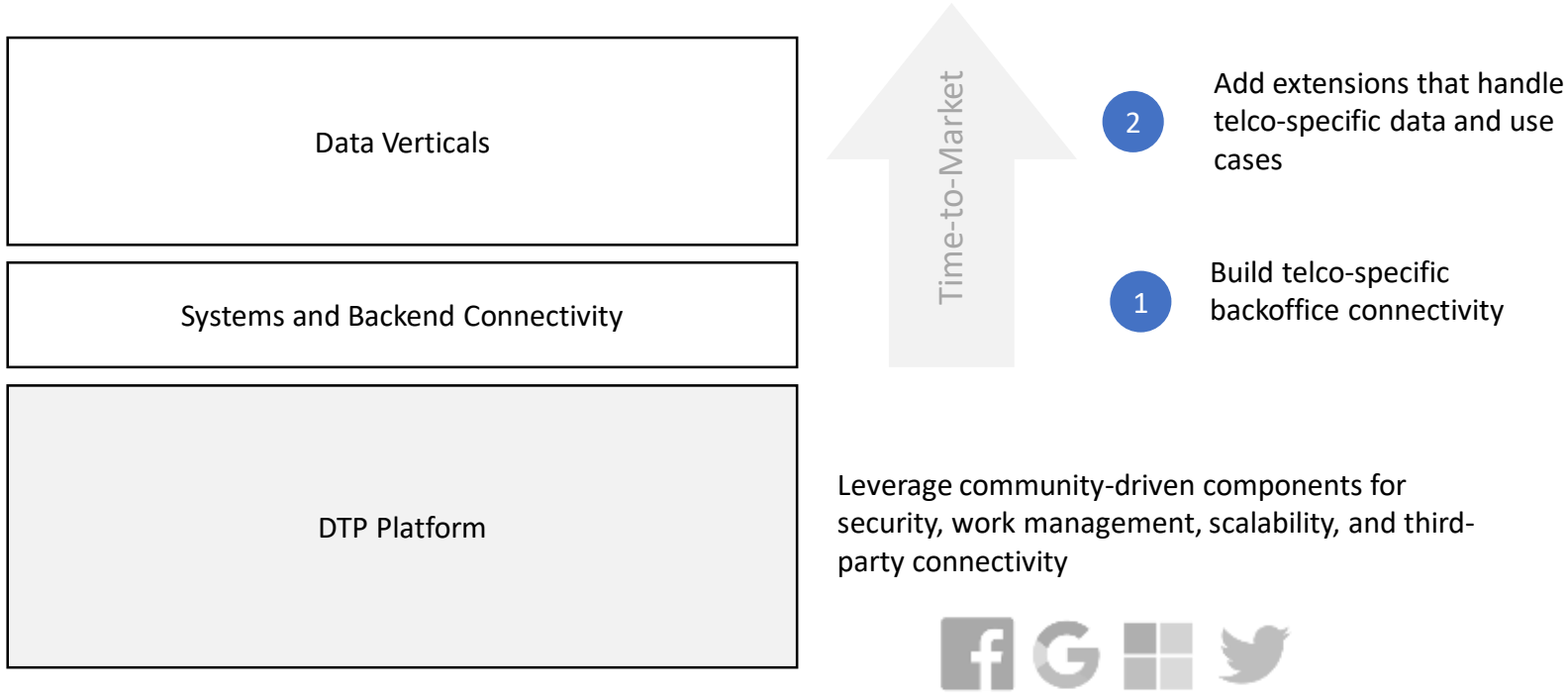
User Initiated

- Transfer contact lists from cloud storage to new providers
- Move map/geolocation history to new providers
- Move social media data to telco services
- Promote compliance with government regulations to enable data portability

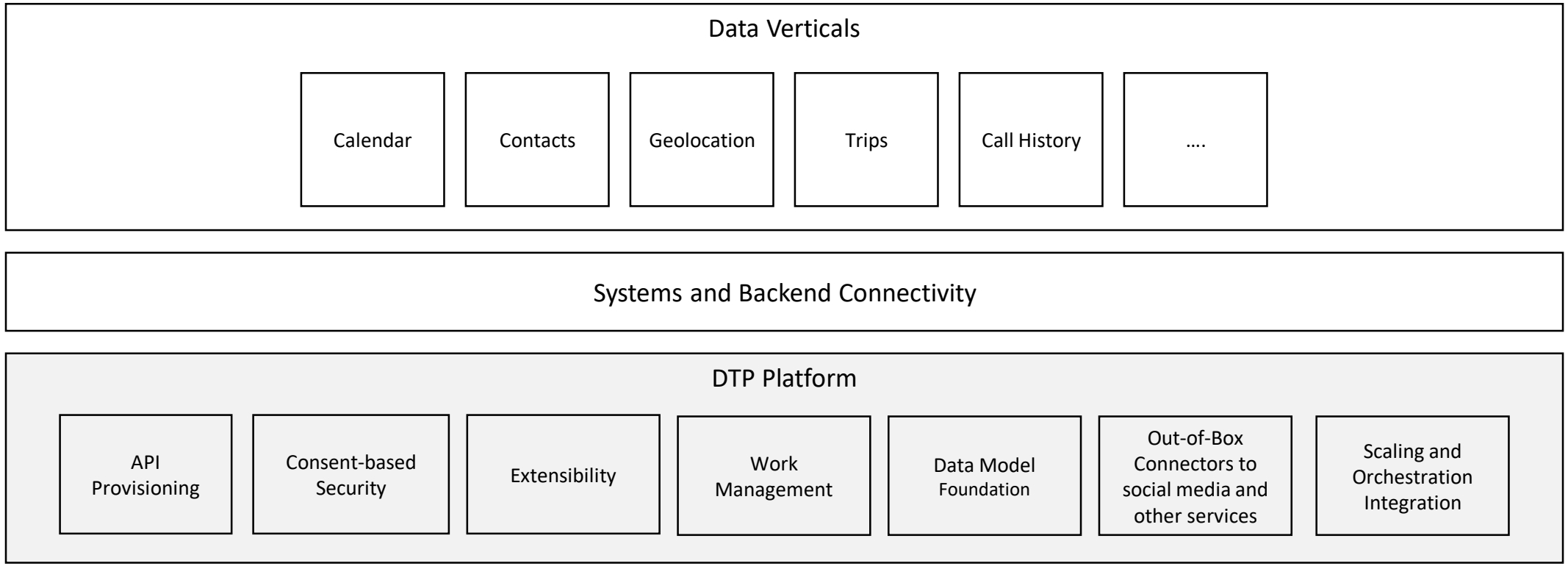
Partner Service Enable

- Automated sharing of presence and geolocation data for insurance verification
- Data transfer to fulfill key functions of a Service Delivery Platform
- Sharing of presence for service review verification
- Provide an auditable and traceable mechanism for tracking user consent when sharing data with third-parties

Reduce Costs and Shorten Time to Market



Building Blocks for Secure Data Transfer



White boxes indicate custom platform extensions telco deployers will likely develop

Architecture Highlights

Generic data transfer

- Support virtually any type of data, from small text to large binary
- Does not have to be “standard” data
  - Derived data

Secure

- End-to-end encryption
- Data is never stored or transferred unencrypted

Extensible

- Providers can write their own adapters

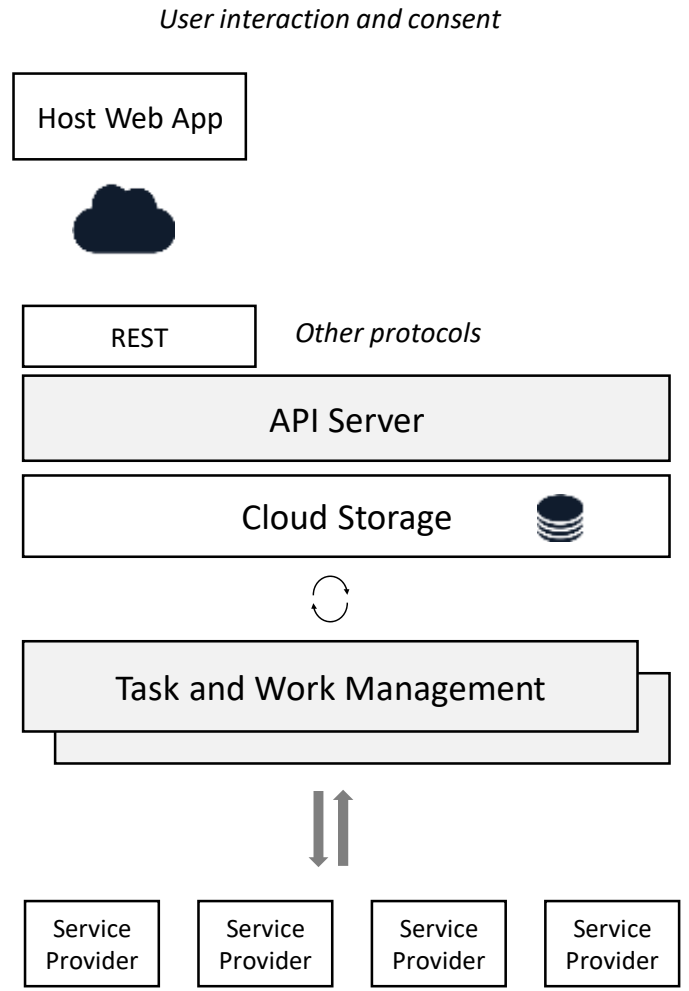
Multi-Cloud

- Adaptors to plug into multiple cloud backends
  - Google Cloud Platform, Azure

Scales Up and Down

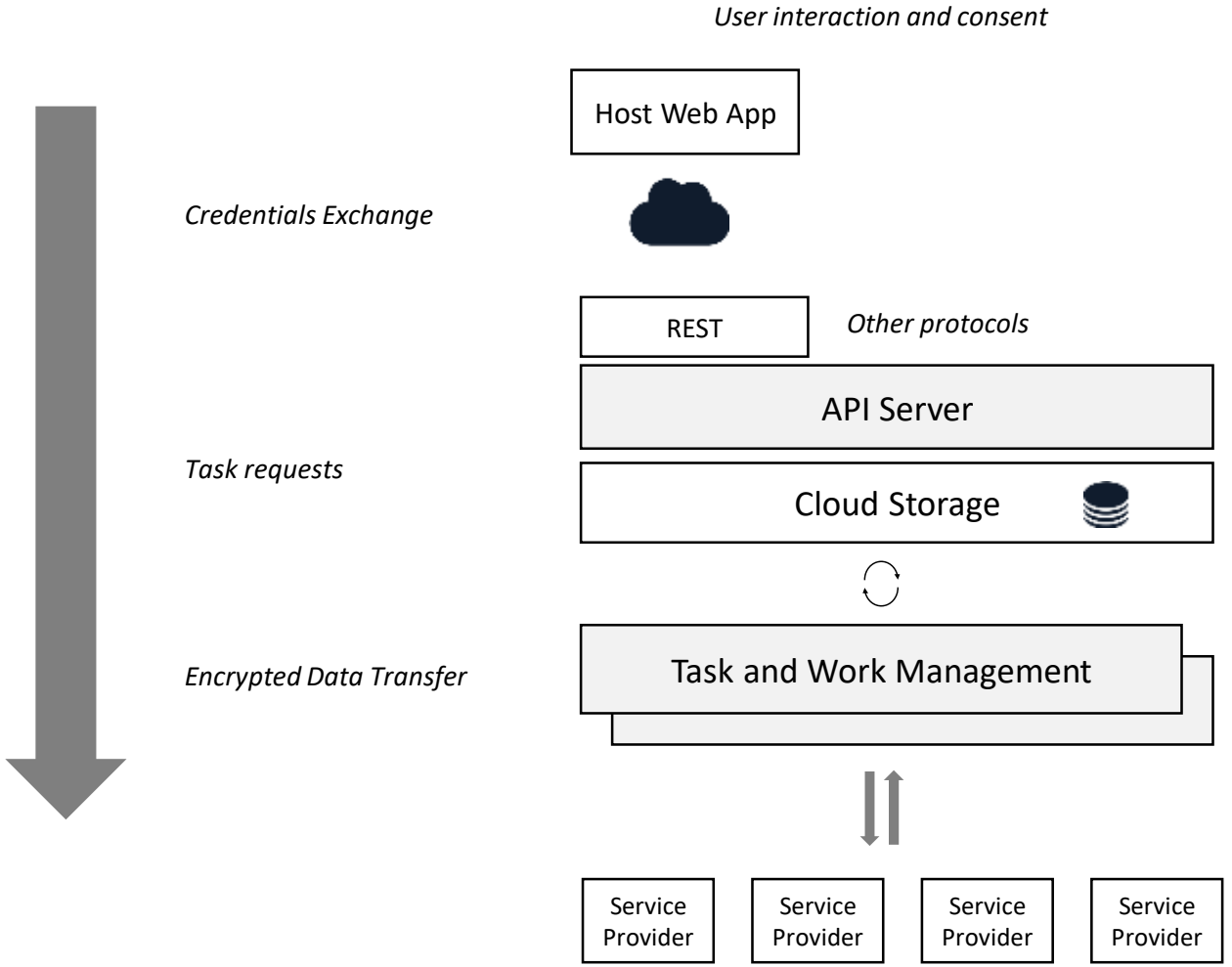
- Run on a laptop, private data center, multi-datacenter (geographically distributed)

Systems Architecture

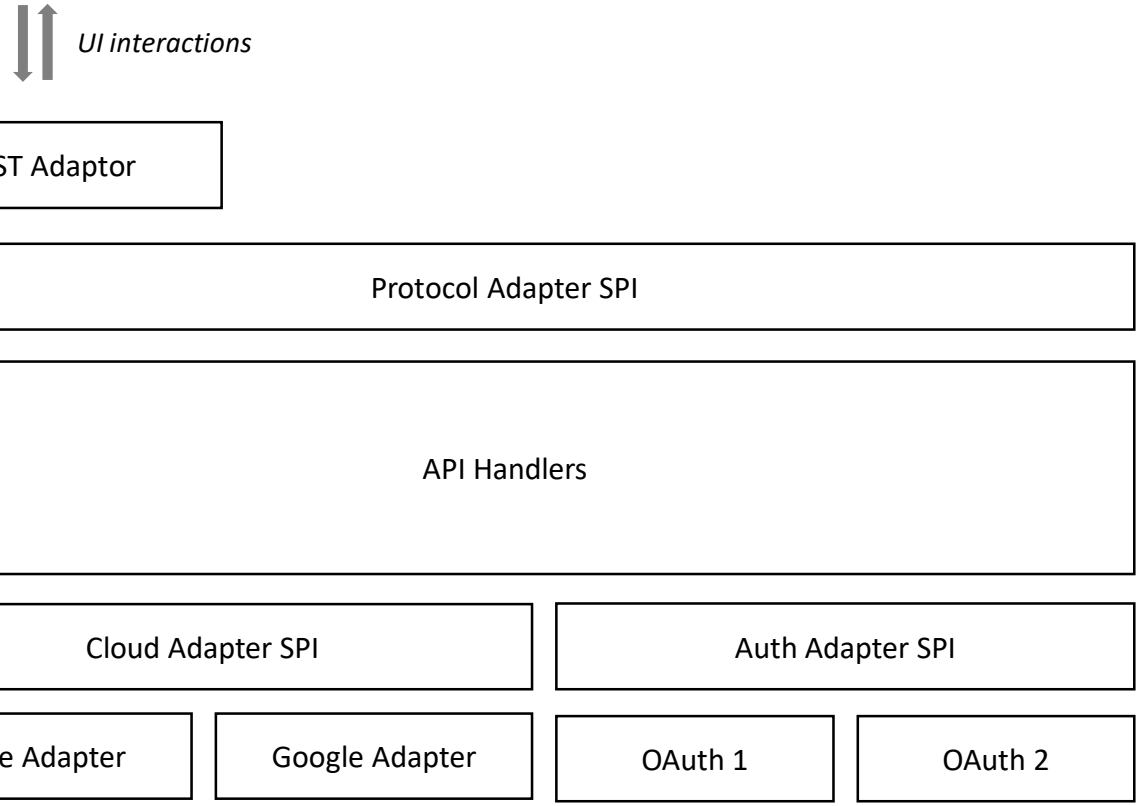


Systems Architecture

1. User selects data to transfer and authenticates with each provider using OAuth in the host web app.
2. Auth tokens are encrypted and sent as a job request to the API Server
3. Encrypted task data stored
4. Scheduled worker picks up the task, decrypts the tokens with its private key, and performs export/import



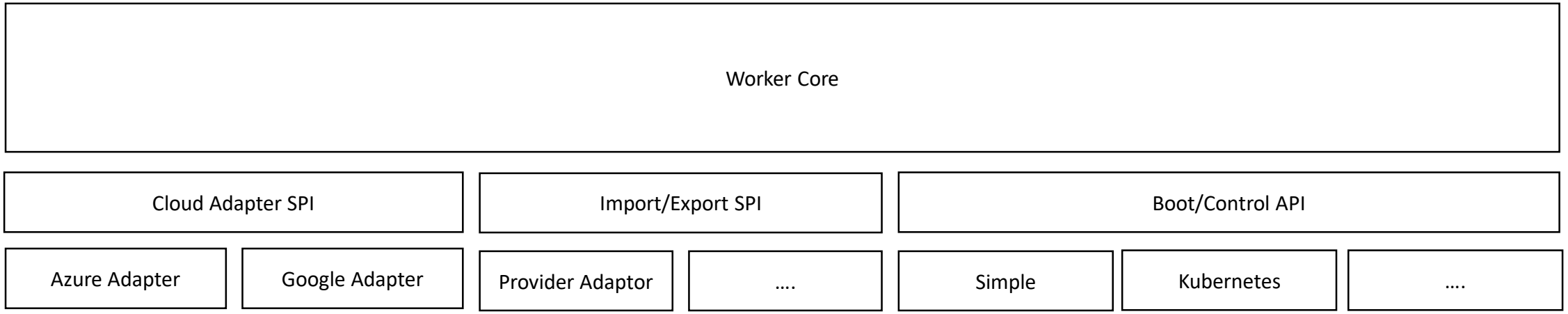
API Server Architecture



Adapters provide custom functionality to the platform

SPIs define the contracts (interfaces) adapters must implement

Task Management Architecture



Adapters provide custom functionality to the platform

SPIs define the contracts (interfaces) adapters must implement

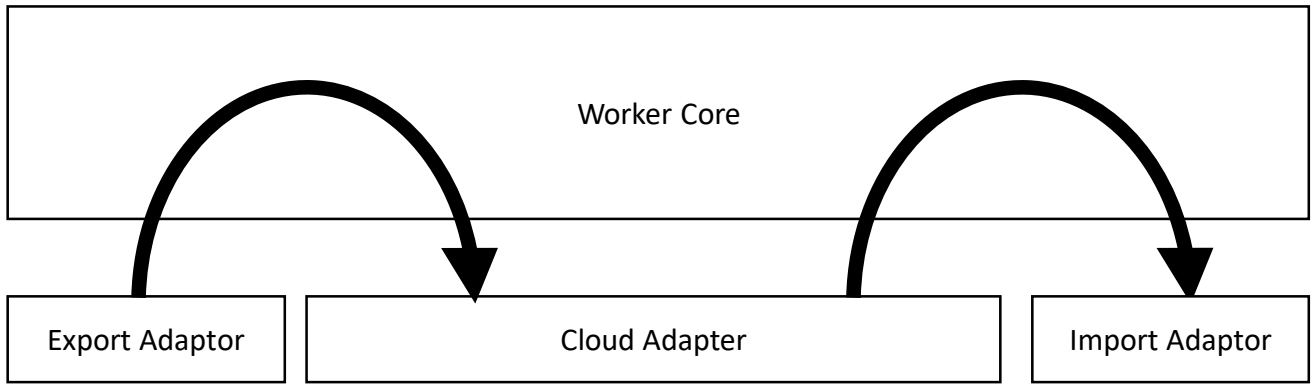
Standard Data Models

There are so many, they mean different things to different people, and they change!

- Embrace this and don't attempt to impose one canonical standard
- *The platform flows* ***extensible data types***
  - Multiple formats
  - Small text-based data to large binaries (streaming)
- Allows supported types to emerge from consensus and practical experience
- Can evolve over time

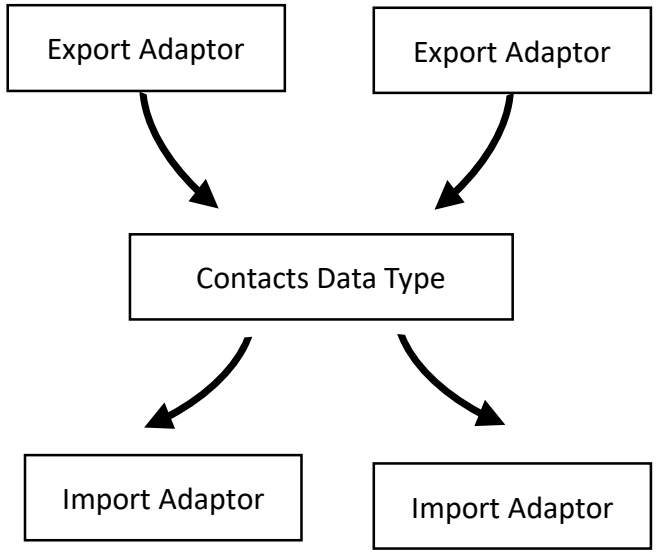
Data Architecture

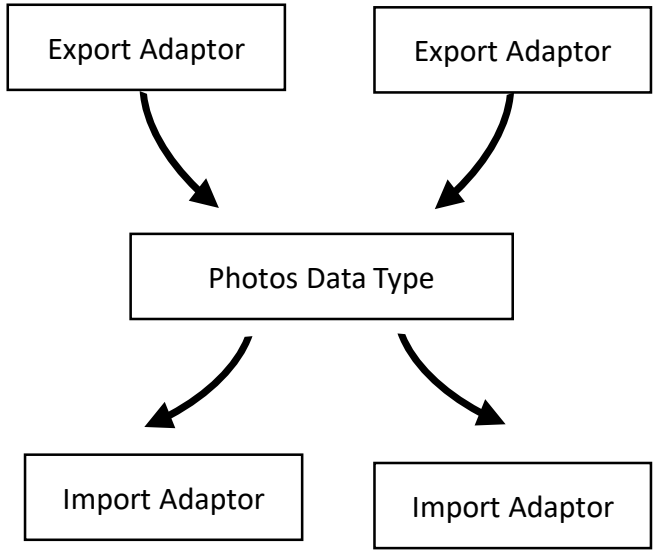
- Extenders define a JSON-based data type that is used by adaptors
  - The platform will flow and persist types



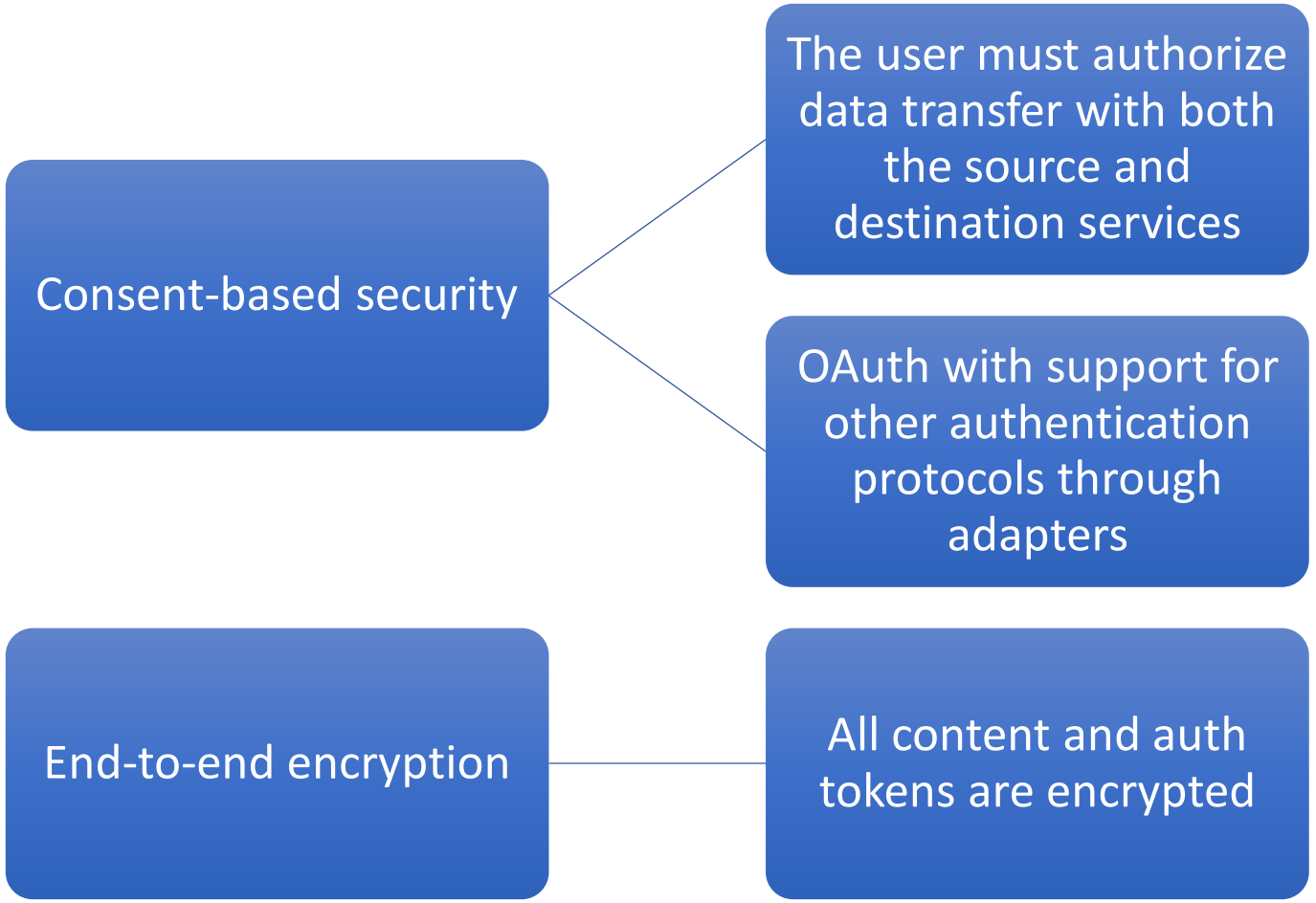
Data Type Verticals

- The platform can support multiple types for different verticals

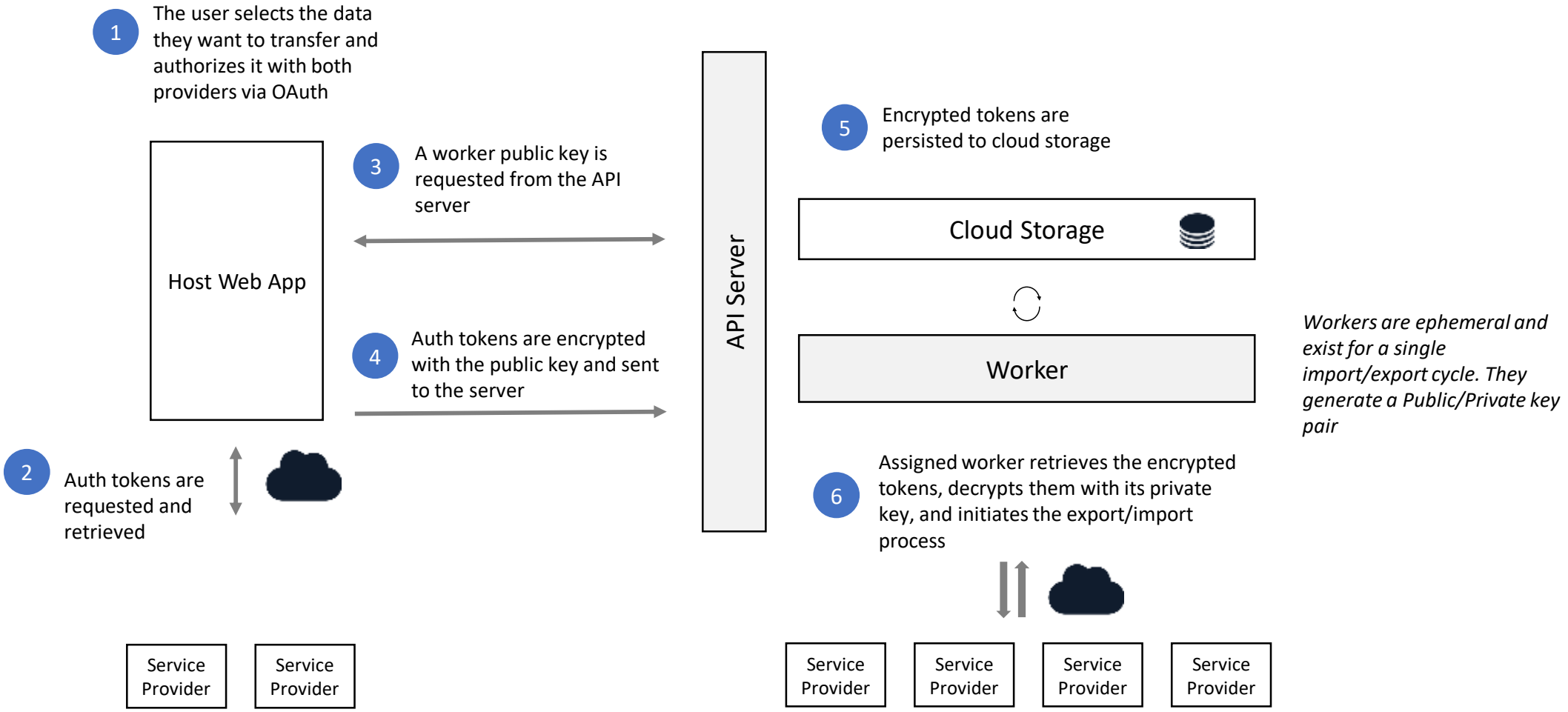




Security Architecture



Security Flow



Workers are ephemeral and exist for a single import/export cycle. They generate a Public/Private key pair

Deployment

- Simplicity is built-in
- Scales up and scales down