Improving Trustworthy and Sustainability Digital Infrastructures with

FACT

The Fair API Commitment to Trust Model

How to make better decisions for digital infrastructure by using API Terms of Service that are easy to read, understand and evaluate

Consent to the 2 standard conditions	Select different options	Receive a License and Customizable Appendix
A fair use policy A fair loyal change policy		PDF

FACT WIZARD

Get your API license now

Report Summary

“I have read the API Terms of Service”

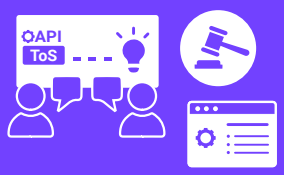
is the biggest lie of the programmable web. To increase trust, openness and safety in digital Infrastructure, we evaluated the opportunity to adopt a new framework to make API Terms of Service (ToS) easily understandable and shareable with users.

“Can we use this API safely in the long term?”

is a difficult question to answer. Most of the time, companies discover that their API access has been revoked when users complain that their app is not working as they expected, due to breaking changes to the API. Sometimes, the pricing options or other clauses trigger the API to be turned off or slowed down, but you did not read in depth the API Terms of Service.

We intend to build a framework making API Terms of Service easily understandable and shareable, for a saner and safer digital infrastructure.

By conducting interviews, surveys and focus group with API users, providers and other stakeholders, by using a legal design approach and prototype testing, we have developed a model collaboratively to define a way forward for more standardized, accessible approaches to documenting and enforcing API Terms of Service.



Our research has clarified the frictions in the user experience of API, the power imbalance and inequalities between API producers and consumers explored the diversity of situations and people involved in the API ecosystem.

Moreover, this research designed a framework for healthier API ecosystems called FACT (FAIR API Commitment to Trust) related to a FACT license describing the mutual commitments between API Providers and Users.

Particular attention has been dedicated to the creation of resources that are easily readable and understandable for a wide audience, thus contributing to the establishment of a trustworthy and sustainable framework that can be easily implemented for the essential digital infrastructures that APIs represent.

Authors

Mehdi Medjaoui

is the founder of the worldwide apidays conferences series, which started in 2012 in Paris. Mehdi is highly involved in the API community and API Industry, as an author, lecturer, consultant and investor in the API tooling space. His industry research includes publishing and maintaining the API Industry Landscape and the yearly State of Banking APIs. In 2018, he published as co-author "Continuous API management" (O'Reilly) and begasn as lecturer and invited professor at HEC MBA and EMLyon Executive MBA. In 2019, Mehdi became a H2020 European Commission expert to lead the APIS4Dgov study on public sector and government APIs. As an entrepreneur, in 2014, Mehdi co-founded OAuth.io, an API middleware for OAuth intégration used by 40,000+ developers that was acquired in 2017. Mehdi's new venture GDPR.dev develops a personal data API framework and protocol to democratize data regulations usage for mass users and compliance for applications developers, making GDPR programmable.

Célya Gruson-Daniel

is a research-practitioner and consultant at inno3, a consulting firm specializing in open models (open data, open source, commons, open science). Her research projects concern the modes of appropriation of sustainable open models within institutes and organizations through the study and implementation of different possible levers (organizational, legal, economic, technical). As a doctor of social sciences, she frequently intervenes as a teacher in various research and higher education institutes to train in digital methodologies (data collection, analysis, management) through a pragmatic and participative approach. She co-founded the HackYourResearch collective in order to create a space for sharing and exchanging on the evolution of research and open science practices.

Benjamin Jean

is currently the inno³ company's CEO, an open innovation consulting firm focused on digital transformation through IP, management and ecosystem strategies. He is also the cofounder and former president of Open Law* Le droit ouvert, a french non-profit organization promoting activities to enhance the transformation of the legal sector through digital co-creation programs based on Open data, Open Source and Open Innovation principles. He also works as lecturer at Sciences Po Paris, teaches IP in several Masters courses and is a consultant within the Gilles Vercken Law firm. He co-found and co-organizes the annual European Opensource and free software Law Event.

Mark Boyd

is an API industry analyst and writer. Mark is the founder of Platformable, which specializes in using data and building tools to support the development of open ecosystems. He is the lead author of the European Commission's API Framework for Digital Government, and has worked globally with World Bank, World Health Organization, UN agencies, private companies and non-profits on API and data governance, platform thinking, and open ecosystem strategy and best practices.

Acknowledgments

This report was made possible by the Critical Digital Infrastructure Grant. This Research For Project is being funded by Ford Foundation and Sloan Foundation as well as Mozilla, Omidyar and Open Society Foundations.

Many thanks to the funders and the digital infrastructure grantees for their support and the digital exchange all along the different phases of the project.

We would also like to thank all those who participated in the study in the form of an interview, a response to the questionnaire or participation in the workshop “New models for APITerms of Service”.

A special thanks to the enlightening exchanges on legal design with Rossana Ducato and with Rebeca Vittorazo from Platformable.

This report would not have been possible without the help of Francois Xavier CAO, and Fanny Kwedi and Romain Rouyer (Inno3).

For more information on the Critical Digital Infrastructure program, see:

https://www.fordfoundation.org/campaigns/critical-digital-infrastructure-research/ and https://opencollective.com/di-grants

Contents

Project Methodology: summary

The APIToS project is part of the digital infrastructure research grant. It has been conducted since May 2021 and aims to better understand how API Terms of Service can be leveraged for the building and maintenance of more sustainable and open infrastructures.

Methodology and frames used are in annexes.

Phase 1: Research design

The first phase of the project was an exploratory qualitative phase based on:

- **9 interviews (av. 1 h) with different profiles (API specialists, API providers, API users in different sectors: research, transport, finance, etc.) mainly from France and USA**
- **Review of initiatives concerning ToS or API or legal design in general**

Our aim was first to exchange with different people using, building or regulating API to apprehend better their understanding and interaction with API Terms of service and highlight the main issues they are facing and their opinion on a creative commons model. We also reviewed initiatives (sometimes suggested during the interview) at the crossroad with legal design and API/ToS. Our goal was to get an overview of some inspiring models and the way we should interact with them (integrate our prototype to other projects).

A cross-analysis from the different members of the research team with different backgrounds (law, STS, API specialist) was also conducted (based on the transcript of the interview, and legal analysis of ToS).

Phase 2: Survey

Phase 2 of the study consisted of running a quantitative survey. Analysis of the survey responses, alongside findings from Phase 1 interviews, helped guide the development of a prototype qualitative framework for describing API Terms of Service in a way that is understandable and readable by users, easy to produce by providers, and favors open ecosystems for policymakers.

Phase 3: Workshop

To anchor our preliminary results and test our hypotheses with the API practitioner community, we organized the “New models for APITerms of Service” online workshop/focus group of 20+ people working with or studying APIs. Attendees had different backgrounds (from research, industry, legal design, legal studies, and policy development) and explored, assessed, discussed, and validated the prototypes and resources developed by the research team.

Our online workshop design was based on the qualitative and quantitative results of Phase 1 (interviews) and Phase 2 (survey results) of the project. Based on these results, we proposed to the participants:

The main question to answer collectively:

How to build trust in the use of APIs by clarifying the commitments of API providers and empowering users?

PART 01

SOLVING THE BIGGEST LIE OF THE PROGRAMMABLE WEB

Overview

Our project commenced with the provocative and significant summary of **“API Terms of Service; Didn’t Read,”** Initiative (ToS;DR for short).

“I have read and agreed to the Terms” is the biggest lie on the web. In general, most people don’t read Terms of Service for the online and mobile services they use, and we saw this repeated with API ToS not necessarily because of a lack of interest, but perhaps because of a lack of support to deep dive into these complex, and non-standardized legal documents.
API Terms of Service are a key element of today’s digital infrastructure. Every time you use a Software-as-a-Service via its API, you are integrating the technical contract represented by the interface itself. But behind it, you are also accepting the business and legal contract represented by the rights allowed in copying or consuming that interface.

Known cases of API Terms of Service change and enforcement bring Twitter to mind. Many times in the past, Twitter almost killed its entire ecosystem of applications by abruptly changing API Terms of Service, for example in 2013 and 2018, even if then-CEO Jack Dorsey apologized and declared wanting a better relationship with developers in 2015. In 2023, incoming CEO Elon Musk has since decimated the API developer program once again: destroying businesses, demolishing countless research studies, and once again killing Twitter’s ecosystem potential with one sudden decision. Every day, many successful companies update their API Terms of Service, causing trouble in digital ecosystems, and killing the trust needed to build a sane, resilient and future-proof digital infrastructure.

Twitter is not alone, Google is also known for having killed API access to companies, for instance when they acquired QPX, an API for airfare data, or for suddenly changing API Terms of Service, like when they abruptly modified the free tier plan on Google Maps API and, according to developers, “insanely increased pricing”. Hundreds of websites broke because of this API ToS Change. Google is known to have killed so many services that there is even a community-driven Google Graveyard. Recently, Google itself announced an “Enterprise API Label”, meaning a label that guarantees these APIs, showing the importance for APIs owners to build a trustworthy relationship with their users for the success of their business.

In other fields, including academic research, scientific work is dependent on the use of APIs to have access to data or to publications. Although there was a huge movement towards open access to publications and data in research, APIs of publishers might become a new closed or controlled door slowing down the open science movement and free circulation of knowledge. Again, using 2023’s Twitter API decision as an example, the end of free access to the Twitter API has had a strong impact on research communities that cannot afford to pay $100/month or moreover, resulting in the destruction of numerous research studies globally being conducted by researchers, and PhD students based on Twitter’s data.

APIs are also a major topic for regulators to support a fair and balanced ecosystem for various stakeholders and empower those with more vulnerability.

How did such a situation develop? The main reason is that, as API users, we often don't read API Terms of service. Even API builders have difficulties writing such contracts! These are legal documents, made by lawyers for lawyers and bearing in mind developers sometimes don't even like to read the technical docs (Read The Fine Manual for its Bowdlerized version), they for sure won't read the legal docs!)
If we had read API Terms of service before ticking that box, we would have noticed that almost all of them declare that **"They can revoke your access any time for any reason"**.
Impromptu revocation is not the only problem. Other **API Terms of Service risks** include:

Breaking changes policy	**Re-use of the data**	**Ability to cache the data**
Ability to store the data	**Deprecation period**	**Support**

Of course, some API providers can guarantee contractually that you will have API access. Moreover, some current initiatives are trying to solve that lack of understanding, like the API Rating Agency measuring performance of different APIs and their underlying services or the API Terms of Service Generator (works only for Swedish law), as well as a governmental initiative called Open Terms Archive to track the evolution of Terms of Service.

Today, the research project APIToS is also working on making it easier for the whole ecosystem (developers, researchers, API producers, regulators, etc.). Funded by the Ford Foundation in the context of the Digital Infrastructure Grants, we are exploring ways to make API Terms of Service more easily understandable by humans and by machines.

The as-a-service digital infrastructure ecosystem

Digital infrastructures and digital ecosystems: some definitions and key concepts

Infrastructure-as-a-Service

A cloud-based service that provides physical, virtual, and additional storage networking products (Mohammed & Zeebaree,2021)

Platform-as-a-Service

Cloud-based infrastructure for the development of applications and technologies that are distributed over the Internet, without the need for the end user to download or handle the user interface Mohammed, Zeebaree 2021).

Software-as-a-Service

A cloud-based software delivery model in which an application is hosted by a distributor or provider and made available to customers over the Internet. Mohammed, Zeebaree 2021).

Digital infrastructures

Socio-technical systems in a digital environment that share some key features with physical infrastructures such as ubiquity, reliability, invisibility, and gateways (Plantin 2018). They play a critical role today in enabling digital economies, and are considered faster and cheaper to build compared to physical infrastructures. Digital infrastructures change often (requiring ongoing maintenance) and are characterized by frequent change of usage and adoption. They are not associated with a central organization (decentralized nature) and tend to create new challenges for their management (Eghbal 2016).

Digital ecosystems

A digital ecosystem is a network of participation opportunities that allow stakeholders (including governments and regulators, associations, industry enterprises, small and medium enterprises, researchers, community groups and individuals) to co-create, collaborate, complement, coordinate, and/or compete with each other interdependently by using common infrastructures and tools (such as open standards and data models, APIs, and open source technologies) (European Commission 2007; Boyd 2021). A digital ecosystem has the possibility to solve complex and dynamic problems automatically, while drawing on properties similar to the characteristics of biological ecosystems such as robustness, self-organization and scalability (Briscoe, Dewilde 2009). 'Open ecosystems' are those that allow all stakeholders to participate, creating more equitable opportunities for participation (Boyd 2021).

Commons

Shared resources (cultural, natural, digital) held in common by a group of people/ stakeholders (users group, communities) with a set of governance principles (norms and values) to ensure the individual and collective benefits of the use of this resources. Commons are defined as a third path between state and market (Ostrom 2015).

Key points in the functioning of a digital economy

As digital infrastructures and platforms grow, more and more companies see the opportunity to specialize into one piece of software that others will use. **Today, our digital economy is based on a cloud computing approach with three main services related to Infrastructure (IaaS) Platform (PaaS) and Software (SaaS)** (Kushida, Murray, Zysman 2011).

This "as-a-service economy" compared to former digital business models is based on the deployment of an Internet-based environment that enables the use of software, services, and infrastructure from anywhere (that is, "could computing"). This move towards "cloud computing" has several impacts and changes of usage that **give access to several facilities, easily without maintenance, or the need to invest in building external infrastructures (servers, data centers, etc.)** (Mohammed, Zeebaree 2021). In turn, cloud computing technical decisions have had impacts on business models. For example, with software-as-a-service, the protection given by the majority of open source licenses for the distribution of software is not guaranteed anymore.[1]

In this context, APIs are considered as the glue "under the hood" (Vaccari et al 2020) enabling the development of digital ecosystems with multiple stakeholders able to exchange data and resources in an automatic and standardized way. APIs are the main interface to deal and negotiate between different stakeholders, which may take on different roles in their relationships. Sometimes, stakeholders are connected as they co-create new value by sharing data and services to create new products for consumers, other times, stakeholders may work together to coordinate and agree on the use of interoperability standards or common data models, other times, stakeholders are in direct competition, and so on. This gives rise to a digital ecosystem in which every stakeholder will consume the core competencies of others through APIs, and each will focus on their own core competencies and deliver value to others via APIs. This creates interdependency and reliance on everyone within the ecosystem when generating and distributing value.

[1]The Affero GPL license has been developed to adapt to the SaaS model, as it is no longer so much access to the source code that gives the end-user their freedom, but their ability to dispose of their data and to be able to change service providers.

Key points in the functioning of a digital economy

Digital innovation has shifted organizational relationships into a “dynamic, complex (socio-technical) system” (Nambisan, Lyytinen, Yoo 2010). As part of this digital shift, digital ecosystems have emerged.

Digital ecosystems can be described as: “a number of firms - competitors and complementors - that work together to create a new market and produce goods and services of value to customers” (Hazlett 2011).

In turn, this has ‘inverted’ the role of any one firm “where production moves from the inside to outside” (Benzell et al 2022). In this new paradigm, “employing APIs helps firms grow .. by inverting the firm, enabling third party complementors, rather than improving the firm’s open value creation (Benzell et al 2022).”

This shift to ecosystems is currently occurring across all industries, most notably in areas such as banking and finance where governments have either introduced regulations in many jurisdictions to require banks and healthcare providers to make APIs available (Babina et al 2022, Vaccari et al 2020), or where governments play a facilitative role in encouraging platform and ecosystem networks in areas such as transport and logistics, weather and agriculture, and many other industries (Vaccari et al 2020).

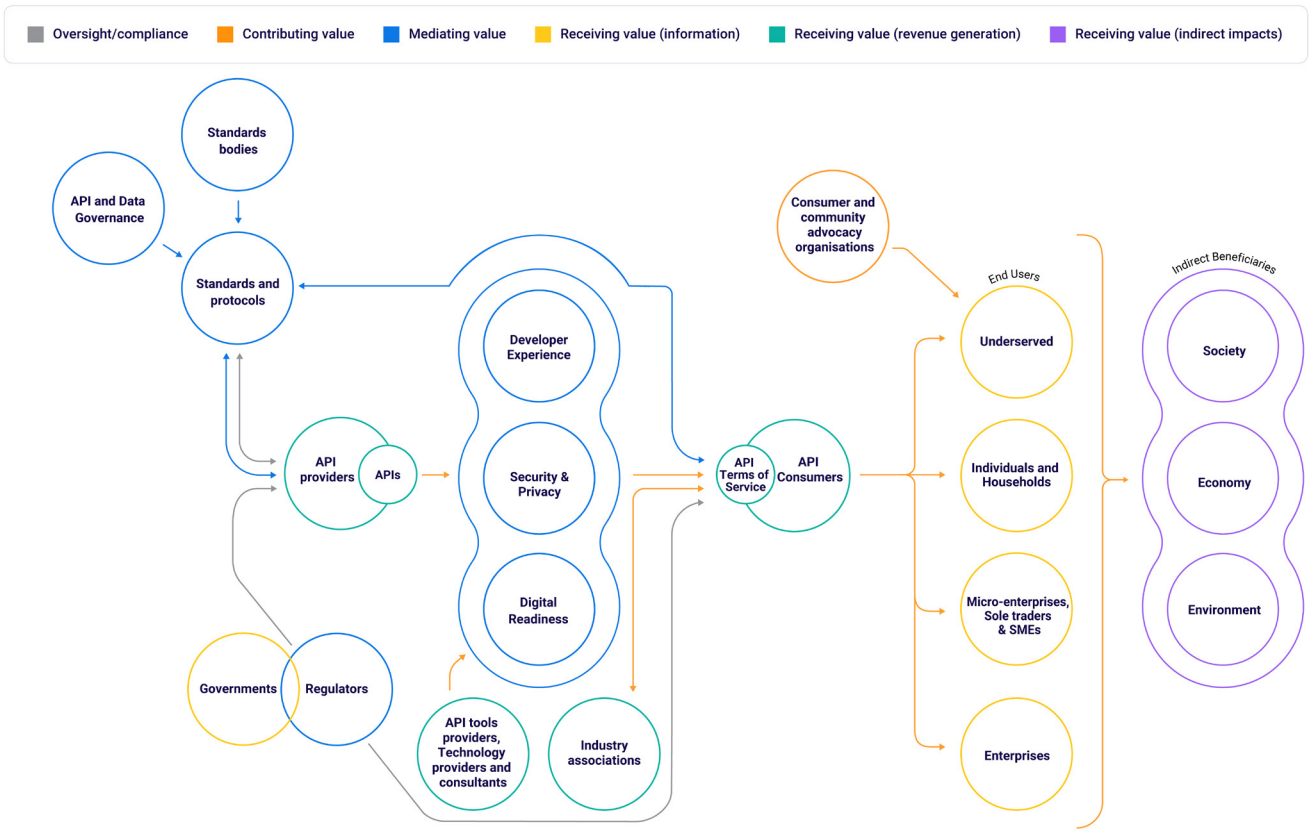
In Europe, for example, initial regulations from the European Commission aimed at enabling a data economy identified priorities for European member state governments to make data more readily available as an asset that can be used in creation of products and services. The Open Data Directive specifically notes that ‘high value’ and dynamic datasets should be identified and that APIs should be the mechanism for which these datasets are then made available for reuse (European Commission 2019).

An open ecosystem is one in which APIs enable third party co-creation and complementarity amongst various stakeholders. Whole industry sectors are encouraged (or regulated) to expose data and services as APIs. For example, in the US, healthcare software providers must use API technologies to integrate electronic healthcare records of patients as a minimum requirement to receive funding for Medicare and Medicaid patient service delivery, creating an open health ecosystem. In Brazil, banks must make datasets such as account information, and services such as payments and credit scoring, available via API so that third party providers can build additional services on top of banking infrastructure (creating an open banking/open finance ecosystem).

In such digital ecosystems, all stakeholders become interdependent. In an open ecosystem, APIs are modularized components that create the “ability to mix remix data, software and services in surprising and value creating ways” (Benzell et al 2022). API providers draw on standards and governance and API tools and practices to create robust and stable APIs. Depending on the level of developer experience, level of security and digital readiness of ecosystem stakeholders, these APIs can accelerate the creation of new products and services built by API consumers for various target segments. In turn, wider society, environmental and economic benefits can also be generated.

How value is generated and distributed amongst stakeholders in an open ecosystem

Platformable



The role and importance of web APIs

What is an API?

Application programming interfaces (APIs) are machine-to-machine digital interfaces that facilitate the exchange of data and services (functionalities).

Vaccari, et al 2020

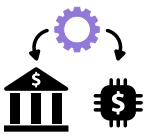
> *An API is not only a building brick, it is also a projection of a product vision, based on internal assets you can open to the world.*

Mehdi Medjoaui, 5 Ways an API is More Than An "API"

Typically, there are four types of APIs:

Public/open

APIs that are exposed to external audiences, often referred to as third-parties, who may make use of the APIs freely or via a paid model.



Partner

APIs that are made available to close industry partners and customers for specific use cases and to enable direct integration with an API provider's data and business capabilities.

Private/internal

APIs that are used internally by engineering teams to connect microservices, serverless, distributed and other modern IT architecture designs in which internal business capabilities and datasets are created as composable building blocks that can be reused for internal application design, automations and process workflows.

Shadow

APIs that may have been created by internal teams to complete a specific use case or task. Shadow APIs are not formally documented or acknowledged as APIs, and are often initially created to enable a shortcut in order for an engineer or engineering team to complete a task under a time or resources pressure to ensure a specific use case was fulfilled.

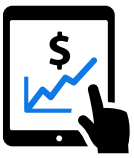
How APIs empower everyday applications

Two examples of how APIs provide the functionality in apps and services you use.

Transport

Public transport route planners draw on:

- Timetable data via API to help calculate travel journey time
- Real-time data feeds via API of where transport is currently to provide updates on how much longer to wait
- Parking space APIs to let you know where space is available to park and ride
- Mapping APIs to plot all of this information on a map application
- Places of interest APIs to indicate other landmarks and destinations on the map



Finance

Financial management apps draw on:

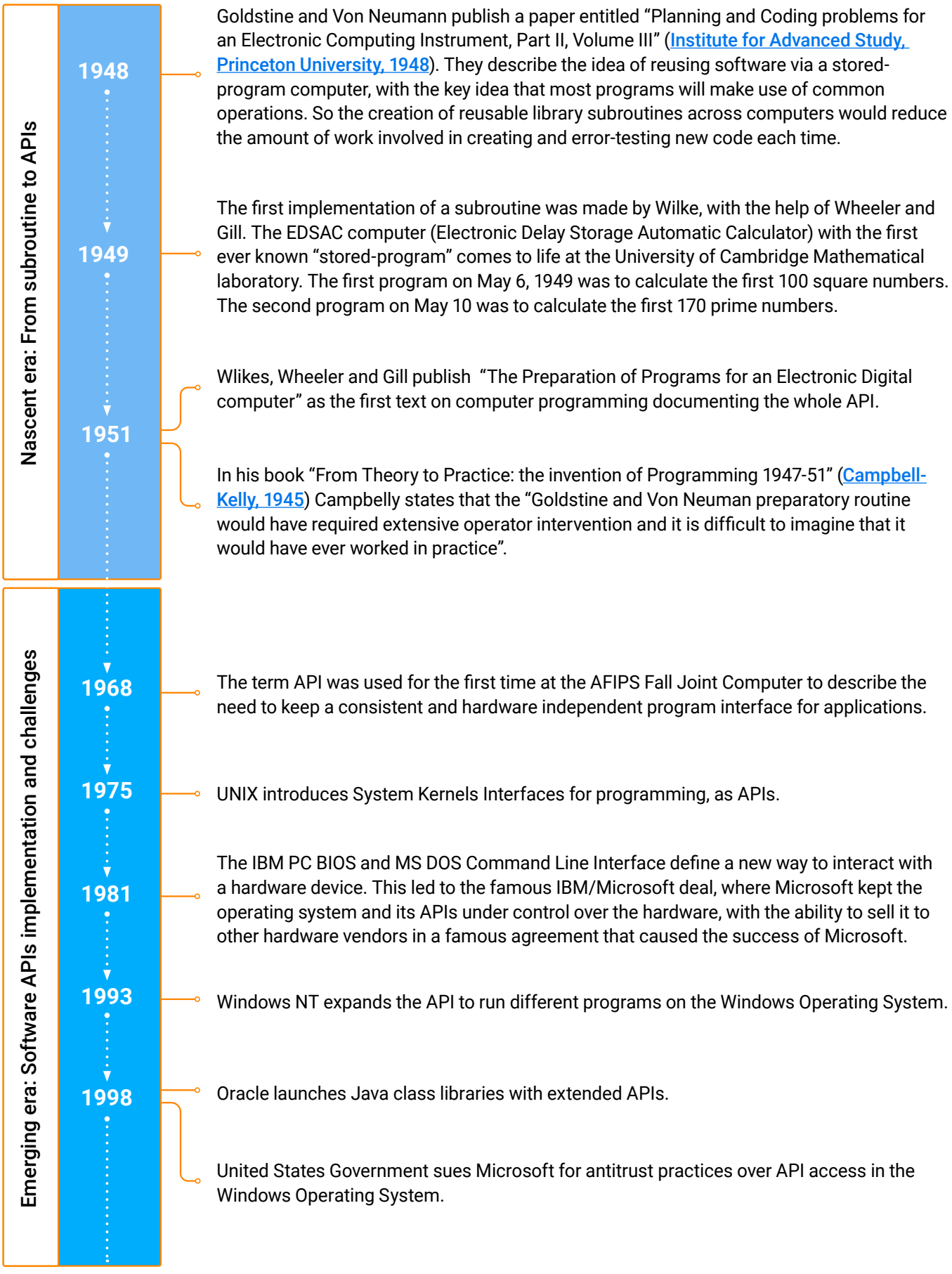
- Bank account information data via API to calculate spending patterns
- Cashflow optimisation algorithms via API to estimate how long until your money runs out
- Product information APIs to offer alternative lending and savings products
- Subscription reminder APIs to let you know if you are paying fort services you don't use
- Climate calculator APIs to assess the carbon footprint of your spending
- Payments and transfer APIs to help you move money from your bank account to your savings account

The importance of APIs

APIs expose data and service capabilities to help build new applications, products and services. They assist in interoperability by providing a consistent, standardized means of connecting various systems together. They enable automation and can help speed up product development and help scale solutions. APIs assist security and privacy because they can define the permissions of who should be able to access the data and services being connected.

Evolution of API and its crucial role in digital economy today

APIs have evolved over the last 60 years to become more standardized, scalable and secure interfaces that enable integration of services from one application or system to another. In 2022, the API mindset has reached maturity.



Goldstine and Von Neumann publish a paper entitled “Planning and Coding problems for an Electronic Computing Instrument, Part II, Volume III” (Institute for Advanced Study, Princeton University, 1948). They describe the idea of reusing software via a stored-program computer, with the key idea that most programs will make use of common operations. So the creation of reusable library subroutines across computers would reduce the amount of work involved in creating and error-testing new code each time.

The first implementation of a subroutine was made by Wilke, with the help of Wheeler and Gill. The EDSAC computer (Electronic Delay Storage Automatic Calculator) with the first ever known “stored-program” comes to life at the University of Cambridge Mathematical laboratory. The first program on May 6, 1949 was to calculate the first 100 square numbers. The second program on May 10 was to calculate the first 170 prime numbers.

Wlikes, Wheeler and Gill publish “The Preparation of Programs for an Electronic Digital computer” as the first text on computer programming documenting the whole API.

In his book “From Theory to Practice: the invention of Programming 1947-51” (Campbell-Kelly, 1945) Campbelly states that the “Goldstine and Von Neuman preparatory routine would have required extensive operator intervention and it is difficult to imagine that it would have ever worked in practice”.

The term API was used for the first time at the AFIPS Fall Joint Computer to describe the need to keep a consistent and hardware independent program interface for applications.

UNIX introduces System Kernels Interfaces for programming, as APIs.

The IBM PC BIOS and MS DOS Command Line Interface define a new way to interact with a hardware device. This led to the famous IBM/Microsoft deal, where Microsoft kept the operating system and its APIs under control over the hardware, with the ability to sell it to other hardware vendors in a famous agreement that caused the success of Microsoft.

Windows NT expands the API to run different programs on the Windows Operating System.

Oracle launches Java class libraries with extended APIs.

United States Government sues Microsoft for antitrust practices over API access in the Windows Operating System.


Maturing era: Web APIs and API-enabled digital ecosystems
1999
2000
2002
2005
2006
2010
2012
2013
2014
2015
2016
2018
2019
2021
Salesforce, the first CRM to be accessible via a web API on a web driven UI, launches with the slogan "No Software", that is "no software installed on your computer" (Software-as-a-service).
Ebay launches its first public web APIs for developers.
Roy Fielding publishes his PhD on REST APIs as the architectural style that web APIs should respect.
Jeff Bezos emails instructions to staff, known as the "Jeff Bezos Mandate on APIs" to shift the company towards service interfaces to solve software delivery issues and re-usability across Amazon, and to design APIs as if they were going to be exposed as public web services to external developers, even if they are intended for internal services.
Caterina Fake, co-founder of Flickr, declares that "APIs are Bizdev 2.0"
Jonathan Zittrain in his book "The future of the internet and how to stop it" coins the term "API neutrality" (Zittrain, 2008).
Oracle sues Google for copyright infringement on Java APIs on the Android Operating system.
A US jury rules that APIs are not copyrightable.
Twitter changes its API Terms of Service, breaking hundreds of thousands of third party applications.
President Barack Obama requires that all federal agencies have public APIs
US Court of Appeals reverses Judge Alslup ruling on copyright infringement of APIs and Google petitions the US Supreme Court.
Facebook abruptly changes its Graph API Terms of Service, breaking lots of applications.
Kin Lane and Steve Willmott launch API Commons, a Framework to share API designs, data models, and specifications.
Federal Court jury rules that Google's reimplementation of Java APIs was a fair use.
Google Maps changes its API Terms of Service pricing from freemium to pay-as you go", breaking hundreds of thousand applications and websites.
US Court of Appeals for the Federal District reverses the Federal Court jury verdict of 2016.
US policy comes into force that makes APIs a mandatory requirement for all health software used in US Government-funded healthcare.
European Commission Directive on open data and the reuse of public sector information legislates that all high value datasets across 6 key thematic categories must be made available via APIs.
US Supreme Court concludes that Google's use of Java APIs is a "fair-use".
European Commission launches API Framework for Digital Governments to guide the creation of APIs for government services.
Google establishes a new labeling of Enterprise APIs to enforce greater stability and give developers more confidence that key production APIs will not create breaking changes "for as long as customers are actively using it".

The role of API Terms of Service in addressing inequalities and asymmetrical relationships in digital ecosystems

Overcoming the complexity and chaos of emerging digital ecosystems

Today, the importance of digital infrastructures are beginning to be taken into consideration and to emerge from their invisibility (Eghbal 2016).

As digital infrastructures are seen as faster and cheaper to built compared to physical infrastructures, they also embrace the characteristics of our VUCA world (Volatility, Uncertainty, Complexity and Ambiguity). They are associated with:

Frequent change of usage and adoption

New challenges for management by their decentralized nature (Eghbal 2016).

Power relations between different stakeholders encapsulated in the design of digital infrastructures, platforms and ecosystems are inscribed in the code layer (Lessig 2015; Van Dijck 2020; Gruson-Daniel 2018).

The construction of norms and standards of data flows are the key opportunities to influence the current mechanisms of platformization and concentration of power (Plantin, Punathambekar 2018; Helmond, Nieborg, Van der Vlist 2017; Janneke Adema, Graham Stone 2017).

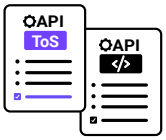
In such emerging digital ecosystems:

- **APIs** play an interface role between different stakeholders and functions (public/private users/ producers data/applications, legal/economic world).
- They can therefore be a vector for the creation of a more open, diverse and sustainable digital ecosystem that guarantees trust in relationships, or, on the contrary, they can reinforce the centralization of data flows to a few major players at the heart of platform dynamics.
- **APIToS** could play a crucial role in the orientation of these ecosystems and their dynamics (centralization of power, inequalities, free ridings, etc).
- By performing the technical, legal, and organizational norms (such as standards, protocols, and IP), APIToS allow us to better understand the underlying economic models but also the rules of data governance between ecosystem stakeholders.
- While open data and open access movements encourage greater access to data and research findings, there is also the need to ensure that the APIs that can expose data and make it available for integration into systems are also regulated to guarantee “openness” and to enable balanced governance between different actors.

API Terms of Service play a small but potentially key role in clarifying how API consumers can make use of available APIs to build products and services, access data and systems, connect with Internet Of Things infrastructure, make use of large datasets and machine learning algorithms, automate workflows and processes, and so on.

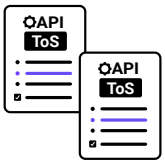
What is APIToS?

API Terms of Service (APIToS) or Terms of Use describe the legal, technical and business contract of the use of APIs and the organization/ individuals behind them.



APIToS can be part of several other legal and policies agreements related to the use of a service of the application or in the specific API documentation dedicated to the developers.

APIToS are often available on the web page of the API or the application in a footer tab with all the policies and legal information.



There are no standard models for **API ToS** but there are some common clauses related to technical and business considerations.[2]

Technical clauses

- **API Revocation policy:** Revocation is an annulment or cancellation of a statement or agreement.The revocation policy is the description of the conditions an API provider can decide to revoke access to an API and its runtime.

- **SLA:** A service-level agreement (SLA) defines the level of service expected from a vendor, laying out the metrics by which the service is measured, as well as remedies or penalties should agreed-on service levels not be achieved. It is a critical component of any technology vendor contract.

- **Breaking changes:** A breaking change is a change that requires API consumers to make changes to their applications in order to avoid disruption to their integration. The following are a few examples of changes: Changes to existing permission definitions; Removal of an allowed parameter, request field or response field; Addition of a required parameter or request field without default values; Changes to the intended functionality of an endpoint.

- **Deprecation policy and versioning:** Versioning policy is the explanation of how an API is updated (versioned), and how these versions are maintained or deprecated. A deprecated API is one that is no longer recommended for use, due to changes to how the API is being provided. The deprecation policy is the determination of deprecation context, duration, support and maintenance.

- **Rate limit and Throttling:** Rate limiting and throttling are the conditions under which an API runtime can be called, in terms of load. It often depends on the license or paid plans.

[2] In our study, we distinguish different clauses and evaluate by a survey their priority for different users.

Business clauses

- **Business model and pricing:** The business model clauses may stipulate when the API is not free for use and the payment conditions and pricing options that will guarantee continued access and consumption.

- **Reuse policy:** The reuse policy is the authorization and boundaries that define which use cases are permitted with the use of the API.

- **License:** The license of the data that is accessible via the API may be described.

- **Copyright:** Copyright rules associated with the API, implementation, and use or re-use may be described.

- **Branding:** Policies about whether API consumers are obliged to mention the source of an API with or without a link, or with or without branding material including appropriate logo or trademark use may be defined.

How do APIs fail? Lessons learnt from the past

Short term risks associated with using an API

A short term risk for an API is one that may interrupt the runtime of an API and cause trouble for third party applications, but does not threaten the main purpose of the application's overall API consumption. These risks often involve technical issues that can be solved in the short term, and often compensation mechanisms or notice periods are described in the API Terms of Service.

Example short term risk: Performance issues, SLA and outage

In the as-a-service economy, digital infrastructures rely on each other to provide capabilities. But what happens when an API fails to deliver a capability to others because of being "down"? This is the example of known outages from Saas, PaaS or IaaS applications. For example, Amazon Web Services outages quickly get easily noticed because a large part of the web ends up being down at the same time. This is why to keep trust at maximum, API providers are transparent about their uptime history, using a status dashboard or API healthcheck. Amazon Web Services share their status page, where every service status and uptime is published with colored logos, often green, orange, red, and black depending on the level of uptime availability.

Example short term risk: Breaking changes

A breaking change is a change that may require recording an application or other API integration use case in order to avoid disruption. If applications consuming an API after a breaking change aren't updated,the application will not work.

Usually, breaking changes are published and communicated in deprecation policies to allow time for developers to adapt their applications in advance of changes. Transparent communication and sufficient notice help to foster trust between API providers and API consumers.

In the past, some companies have not notified developers in advance, and some have not even told developers at all. This is the example of Facebook in their platform scaling period between 2010 and 2014. During this time third party app developers first discovered their app was broken via their own users' and customers' complaints.
A breaking change is an example of short term risk to an application that can be solved but they can occur at any time and are dependent on the company providing the API.

Long term risk issues using an API

Long term risk issues using APIs are risks that can perturb, impeach or break an application consuming an API in the long term, or that completely make an application unusable. They are often policy or business related decisions, as compared to technical terms.

Example long term risk: API Pricing

Google Maps offers mapping, route planning and location services, and APIs are available that allow these functionalities to be consumed as services in third party applications. Initial pricing plans offered a free tier for users, allowing up to 10,000 requests per day. For years, Google pushed this generous “freemium” model as an entry point that attracted new API users, and as a result built a large ecosystem of applications consuming the Google Maps APIs. Their commercial approach was even considered so aggressive that Google was even sued for antitrust practices in France, with France citing the high level of advertising of the free tier model as evidence.

But in 2018, Google Maps abruptly switched off its API free tier and introduced a “pay as you go” pricing model where users would be charged from their first API request onwards. These changes significantly impacted the relationship between API consumers and Google Maps as an API provider. Many users had not registered a credit card for any payments, and often did not have the business infrastructure for owning a new budget line for their use, which up to that date had remained below the freemium threshold of use. As a result, the mapping functionality in many applications broke, especially for startup, public sector and non-profit use cases.

This abrupt API pricing change destroyed much of the trust between API consumers and Google, as the existing API Terms of Service was expected to offer the freemium model , and the API free tiers are trustworthy.

Example long term risk: API reuse policy and business model

Linkedin is a widely used social network for professionals, and offers a set of APIs to build applications with its data. But Linkedin is also known to have used API Terms of Service to keep competitors away by forbidding specific uses of the API. Often the reasons for denying particular issues are claimed to be in the interest of the user, but at times the denial of use of the API better reflects a defense of their interests, especially against startups wanting to use business intelligence from LinkedIn to populate their own business profile datasets.

How to destroy trust in a few short years : The Twitter API story

2002 Twitter killed my business: an inside look at the ecosystem crackdown.

One developer noted they were using Twitter and ***"I had a very successful consulting business. It wasn't very big, it was more of a lifestyle business, making about $15,000 a month. And Twitter killed it — they killed it cold."*** He went on to note ***"I love the service, but I hate the company"***

2015 Twitter CEO Dorsey Apologizes to developers, says they want to reset relations

Ingrid Lunden and Drew Olanoff from Techcrunch reported that Twitter CEO Jack Dorsey spoke at a conference they hosted for developers and said he wanted ***"to make sure we have a great relationship with our developers… that we're fulfilling and serving everyone's needs….We need to have a better conversation with our developer community, with everyone in this room… We can't stand alone. We need your help."***

2018 Twitter destroy outside apps again by killing the API most of them use

According to Silicon Angle journalist, Duncan Riley, Twitter cited technical and business constraints for again shutting down ***"an application programming interface used by most third-party Twitter apps for things such as push notifications and automatic timeline updates".*** Riley concludes: ***"Certainly going forward, the next time Jack Dorsey tries to appeal to third-party developers to support Twitter, here's hoping he will be laughed out of the building."***

2023 Twitter to end free access to its API in Elon Musk's latest monetization push

Following Elon Musk's acquisition of Twitter (October 2022) , API terms of services have abruptly changed with the announcement of the end of the free access to its API. It will have stong impact for developers that use the Twitter API daily. Twitter's change will also impact research useful for society in many fields that is: hate speech, online abuse, crisis management during natural disasters, etc.

API Copyright, API Copyleft and API Neutrality: Emerging concepts in open ecosystem thinking

The case of API Copyright and “Fair Use”

In US law, APIs can be copyrighted, although there is a fair use as long as the section of the API copied is small, and trivial for the main knowledge engineer, as stated in the Google vs Oracle API copyright case.

In European and UK laws, APIs can’t be copyrighted easily and interoperability is often put forward as a key necessary characteristics of the purpose of APIs.

The Google LLC v Oracle America, Inc case is a landmark decision which deals with the intersection between software technology and intellectual property rights. In this case, Oracle had earlier sued Google for copying some parts of the Java programming language’s application programming interfaces (APIs) and about 11,000 lines of source code, which are owned by Oracle, for its early versions of the Android operating system. After a series of court trials that started in the district court of the US and worked up through appeals all the way to the Supreme Court, it was finally decided that copying some parts of an API made by Oracle was fair use under the copyright laws. The court arrived at this decision after considering some important points:

Although APIs are capable of being protected under copyright laws, Google copied only a minuscule amount from the said code

The extent of copying by Google was restricted to a limit as required to include necessary tasks in an android device

Concerning the substantiality in the copied material, it was held that Google did not copy anything unique to the implementation of the API by Oracle. It only used parts of the code, which were now familiar to systems and engineers

The case of API Copyleft

Copyleft is the legal technique of granting certain freedoms over copies of copyrighted works with the requirement that the same rights be preserved in derivative works. In this sense, ‘freedoms’ refers to the use of the work for any purpose, and the ability to modify, copy, share, and redistribute the work, with or without a fee.

Licenses which implement copyleft can be used to maintain copyright conditions for works ranging from computer software, to documents, art, scientific discoveries and even certain patents.

Theoretically, it would be possible to apply Copyleft technique to APIs. An API provider, for example, could define license or API Terms of Service clauses that state any software consuming their API could enforce a specific open ecosystem design, forcing users (through the APIToS) to contribute back value through their end use back to the network of ecosystem stakeholders.

In a 2013 report ordered by the French government about “Tax strategies for the digital economy”, Colin and Colin proposed the application of Value-Added Tax (VAT) for companies that don’t open the data they

collect from others via APIs. Value-added taxes are only paid by end users: other actors along the value chain don't pay the VAT as long as they add value to the product. Only the end user pays the tax on the final value of the product.
In a digital infrastructure value chain, or in the "as-a-service economy", that means that every company that doesn't provide APIs under the same level of openness as they benefited from themselves from using APIs must pay a tax on the value captured in the end product.

While the report did not transform into law, it did help broaden the understanding that API copyleft can have significant impacts, especially for sharing value more fairly in an ecosystem or from personal data stores under data protection regulations like the General Data Protection Regulation (GDPR) in Europe, the CCPA in California, PIPL in China or one of the 60+ other regulations on personal data currently passed or under draft around the globe.

The case for API neutrality

'Net Neutrality' means that Internet service providers or governments may not discriminate between different kinds of online content and apps and users on the Internet. It guarantees a level playing field for all websites and Internet technologies without any intervention, following two key principles:

Absolute non-discrimination: all content, sites, and platforms should be equally distributed on the network

First-come first-served: There should be no queues for sending and receiving data packets based on fees

Net Neutrality is a key reason why the Internet has been able to drive online economic innovation, democratic participation and free speech. It protects an individual's right to use any equipment, content, application or service without interference from the network provider. With Net Neutrality, the network's only job is to move data — not choose which data to privilege with higher-quality service and which to demote to a slower lane.

> *"The principles and factual assumptions that animate network neutrality— that the network has been operated in a particular socially beneficial way and that, especially in the absence of effective competition, it should stay that way—can also apply to Internet services that solicit mash-ups from third-party programmers like Google Maps or Facebook (..)*
>
> *Those who offer open APIs on the Net in an attempt to harness the generative cycle ought to remain application-neutral after their efforts have succeeded, so all those who have built on top of their interfaces can continue to do so on equal terms. If Microsoft retroactively changed Windows to prevent WordPerfect or Firefox from running, it would answer under the antitrust laws and perhaps also in tort for intentional interference with the relationship between the independent software makers and their consumers. Similarly, providers of open APIs to their services can be required to commit to neutral offerings of them, at least when they have reached a position of market dominance for that particular service"*
>
> **Professor of Law Jonathan Zittrain, In the "Future of Internet and How to Stop it"**

This mirrors the case of USA against Microsoft Windows and internet explorer for antitrust issues in 1998, where the US asserted that Microsoft was enforcing its dominant position in the market with Windows by providing a web browser within the Windows platform without giving enough access to other companies to offer their browser solution, potentially reducing the likelihood that end users would install an alternative.

The idea of API neutrality, inspired by Jonathan Zittrain, claims that platforms could declare API neutrality, as being non-discriminant to their users. A widely used set of APIs, Amazon Web Services APIs follow that principle, as Werner Vogels CTO of AWS used to call it in his "10 lessons from 10 years of AWS" as the "no gatekeeper policy, meaning that even a competitor of Amazon could build and host their application on top of AWS APIs Infrastructure as a service.

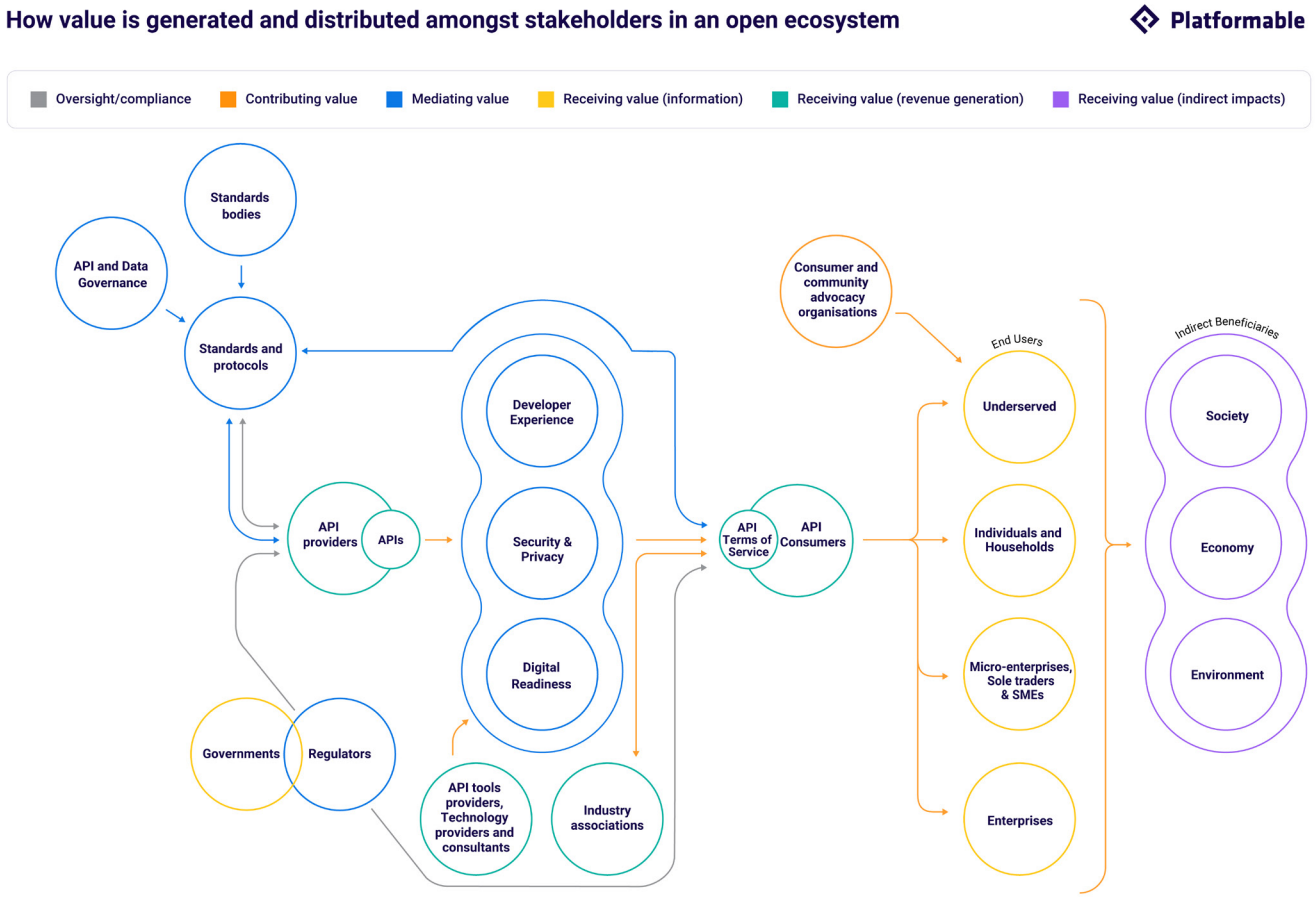
> *Skeptics may object that these relations can be governed by market forces, and if an open API is advertised as contingent, then those who build on it are on notice and can choose to ignore the invitation if they do not like the prospect that it can be withdrawn at any moment. The claim and counterclaim follow the essential pattern of the network neutrality debate.*
>
> **Jonathan Zittrain**

How APIToS could create a new trustworthy and sustainable API Ecosystem

Trust amongst stakeholders in the “as-a-service” digital infrastructure (Eghbal, 2016) has been broken many times because of opaque API Terms of Service, or because of unexpected changes and disruption. That has been the case with large tech companies that often position themselves as essential and foundational digital infrastructures: Twitter, Netflix, Facebook, Linkedin, Google and others.

More open, transparent API Terms of Service contracts could build an open and safe digital infrastructure where the application can automatically know the usage rights of the API, and the data and services being exposed (Vukovic, Laredo, Rajapogal 2014). This could help to solve many technical, compliance and/or business relationships, and could apply to government, non-profit, science, education, and business use cases alike.

Revisiting the model of how value is generated in open ecosystems, we have identified the role of API Terms of Service:



But today, API ToS are often documented by legal departments, business clauses are outlined often by marketing leads, and the APIs themselves are designed by technical teams. This brings confusion for API consumers when seeking to answer the question: Can we use this API? Can data collected through the API be reused? Do we use the API “under the hood” or do we need to declare that it is part of our processes and use the API provider’s logos and branding in our usage? What rights and limitations do we have? What are the short and long term risks of using this API? At present, all of these questions can be challenging for API consumers to answer.

PART 02

APIToS: THE PATH TO DEVELOP A SUSTAINABLE AND TRUSTWORTHY API ECOSYSTEM

About the APIToS project

Our project APIToS, part of the digital infrastructure research grant, aims to understand:

How can API Terms of Service be leveraged for the building and maintenance of a sustainable and trustworthy digital ecosystem?

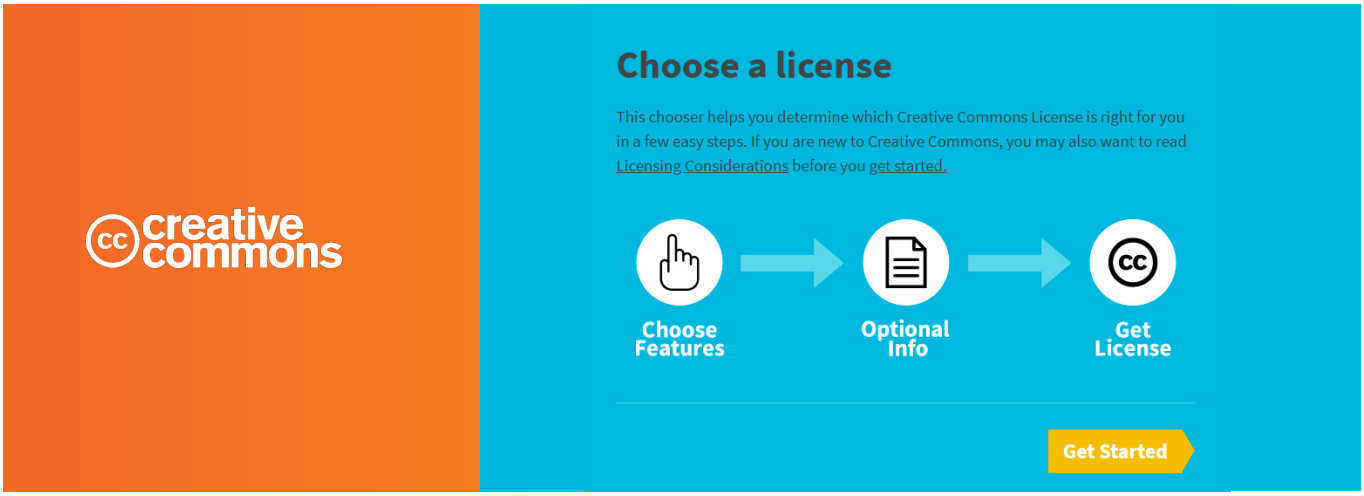
Our initial research question was inspired by the Creative Commons model and open movement. Since the beginning of the web, inspired by cybernetics and the dream of a global decentralized network, the open movement as driven open source technologies, open data, open science, and open culture. Collectively, this work is focused on making a “global social space” real.

One of the major drivers of this ideal of a connected, global, digital network has been catalyzed in particular with the development of the copyleft principle, founded by Richard Stallman. Instead of using intellectual property rights to prohibit, the core principles of content ownership under copyleft are reversed to give the right of use to everyone. On this basis, a group of academics, artists, activists, and lawyers then adapted and extended this principle by defining a series of levels of usage rights for any digital work (images, texts, etc.). This series of levels was documented through the birth of the Creative Commons licenses. The most recent, version 4.0, is today stabilized around different clauses BY, NC, ND and SA and their combination, all describing the rights to use digital content along a spectrum.

creative commons creativecommons.org

Creative Commons is a nonprofit organization that helps overcome legal obstacles to the sharing of knowledge and creativity to address the world’s pressing challenges.

In order to achieve their mission they provide Creative Commons licenses and public domain tools that give every person and organization in the world a free, simple, and standardized way to grant copyright permissions for creative and academic works; ensure proper attribution; and allow others to copy, distribute, and make use of those works.



Creative Commons licenses, along with other open norms and standards designed to be understandable by humans but also readable and interpretable by machines (internet protocol, open formats, etc.), have made it possible to implement a digital ecosystem that facilitates exchange and sharing between a diversity of public and private actors in accordance with principles of openness and transparency.

Today, open source and open data are the basis of economic models that are not founded on the retention of information and property but on the circulation of information, flow and influence. The sharing of a resource, and its circulation in this digital ecosystem generates a greater value through distribution between the different ecosystem actors, with the aim of avoiding the capture of value by any one participant as is often the case with commercial platforms today.

The creation of an open and sustainable digital ecosystem is based on relationships of trust that can be enforced through a legal basis, and implemented through rules of governance. This system regulation ensures participatory opportunities for all and avoids risks of abuse and power imbalance.

The current progress of the open data, open science, and open source technology movements inspired this project to try to create a “Creative Commons” model for API Terms of Service, as a contract that could be automatically read, controlled, and enforced to enable clear and trustworthy interactions and data and services exchanges between infrastructure and applications.

The goal of APITos project is to encourage open and fair API-enabled digital ecosystems with:

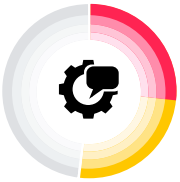
Simple rules of the game properly stated and understandable	**An integrated and coherent (but not rigid) legal framework**	**Mechanisms to foster communication and trust between all ecosystem stakeholders involved**

Main inequalities and frictions encountered in the API ecosystem today

APIToS Project survey findings

42%

of people never check on API changes or do so at the beginning of the project or when there are some visible consequences



25% **23%**

The majority of people have moderate to low reading with APIs (52%). Over 25% never read API change notifications and only 23% read them moderately

30% **30%** **8%**

Interactions with API producers are mostly moderate 30% or very weak 30% or non-existent 8%

70%

34%

30%

More than 70% of people are notified by email. The other channels are mainly social media 34% and or colleagues 30%

APIs and APIToS are a reflection of the critical asymmetrical powers interactions led by big platform API providers

Survey respondents felt that copyleft principles in APIs would have a positive impact

Thought that copyleft was appropriate for specific cases

Thought that copyleft could have a big impact on the use of software via APIs

Via interviews and surveys, we found that API and APIToS were a mirror of the major asymmetrical interactions that exist in digital infrastructures between platform providers and users:

There is pressure for API consumers due to the business and strategic decisions of big platform API providers

There are constraints on small stakeholders ability to participate in digital ecosystems at the level they would like

There is often a lack of communication between API providers and API consumers

There is pressure for API consumers due to the business and strategic decisions of big platform API providers

Frequent changes to APITos leave API consumers without the ability to interact or negotiate on these changes. As a participant of our study explained:

> *The notifications are always along the lines of 'By the way, we're going to change everything in two months. If you're not happy, stop using it.' You accept the terms or you do not use the application.*
>
> Interviewee

API modifications often reflect the strategic business directions of the API providers with direct repercussions on the functionalities that the API previously provided. This also occurs as a consequence of an acquisition, where an API provider makes decisions based on their regulatory context or preferred business model. For example, in the case of data privacy and security, acquisitions can significantly alter the level of risk for the API consumer, especially when company purchases impact ownership across data privacy jurisdictions, as was the case when Mailjet was bought by a US company, and therefore became subject to the free flow of data restrictions between US and European:

> ***Following the acquisition of Mailjet, it seemed that it has been taken over by the Americans, I'm not going to manage customer databases and contacts with a company that is now under American governance, and therefore subject to the USA [data privacy laws and not European GDPR requirements].***
>
> Interviewee

Asymmetries are also reflected between different API suppliers in a system with stakeholders having different impacts/powers. An API ecosystem is by nature interdependent: one API service might draw on data or functionalities being provided by multiple other APIs. For API consumers, each individual API has some value, but it is their work in integrating multiple APIs into a workflow or production system that generates the real value for them, but the API provider has greater power in being able to influence the sustainability od this end system, because they can change their API which in turn impacts on other APIs and the overall flow of data and service creation:

> ***The API itself, in the end, it's [value is]not that great, [it's] the effort you put into creating it and providing the service with the added value that's there, so the implementation***
>
> Interviewee

There are constraints on small stakeholders ability to participate in digital ecosystems at the level they would like

Frequent changes from smaller API providers that are dependent on bigger players can be harmful to the survival of these companies. A small change from a larger platform provider can have a critical impact and put a lot of burden on small/medium stakeholders in an ecosystem. For example, accessing an API that has introduced an expensive entrance cost:

> ***For actors like us, because in fact, we're not in an economy of very rich people, unfortunately, we have to do things as well as possible, but without it costing too much, we can't do everything by ourselves. So we are always a little bit dependent on others, but we always have to find the right balance in this world on the reuse of other APIs.***
>
> Interviewee

The centralization of power amongst a few API providers with a critical mass and strong internal legal resources can impact on the diversity and health of an ecosystem.

Small and medium stakeholders in a digital ecosystem are the first to have to deal with the impacts of digital platform regulations. This puts a lot of constraints on those who don't have allocated resources to deal with legal issues. In academia, for example, it is difficult to find resources that can help to settle the legal and ethical needs related to API and the use of data.

> *Just to underline that, especially in the last few years, it was really a big battle to be able to continue to provide the tools that we can provide, that's why, on the one hand, the lack of resources, but also the growing difficulty to manage quite a lot of different subjects, including ethical considerations [...], legal issues, it existed before, but I think that today, all that is much more pointed.*
>
> Interviewee

There is often a lack of communication between API providers and API consumers

Asymmetrical interactions are also expressed through how communication occurs between API providers and consumers. For example, potential API consumers may need to request access to platform provider APIs using a standardized form without any individual contact details being provided. Decisions on use of the APIs may be made without any direct explanations if the request is rejected, leaving API consumers unclear as to why they have been locked out of participation in digital ecosystems.

Case study: how does conducting social research become big platform decision-dependent ?

In the digital age, using social media platforms to conduct some academic social sciences research is essential. Interactions with these platforms to gain access to data can be difficult:

> *It depends a lot on the platforms, in our experience, we almost never have direct contact with the platforms, so there is no channel, there is no phone number, there is no e-mail where we can hope to get answers, and then, there are platforms that have put in place certain processes, for example, to audition, so, [...] well, you have created a tool, you have to submit it and [they will] decide] if it can continue or not, and in that context [...] it's always very few words, it's always very little interaction, it's a very asymmetrical relationship, and we, we provide documents and then we get one sentence answer.*
>
> Interviewee

API use cases and user journeys

The API ecosystem is made up of a large number of different actors involved, which makes it rich but also complex. To better understand API ecosystems and the potential role APIToS can play in the creation of more trustworthy and sustainable ecosystems, it is necessary to understand the different profiles involved, their roles, their interactions with APIs and APIToS, their practices and their demands.

Based on the project interviews and survey responses, this study designed :

User personae to help understand the amalgamation of points of view around typical scenarios where API consumers need to access an API and review API Terms of Service

User journey maps that identify the points of agreement and the challenges encountered today when reviewing APIToS ahead of incorporating APIs into production use cases

API user personae

Based on research interviews and prior industry expertise of research team members, we designed several user personae to give an overview of the different contexts where APIs and APIToS are used, the people engaged in API use, production, and regulation. Seven profiles were proposed, as shown in Table X.

API Provider Personae	API Consumer Personae	API Ecosystem Personae
API product manager for business	Developer consuming APIs	Policymaker
Platform business owner	Product manager consuming APIs	Open source/Copyleft advocate
API Provider for public sector/research		

The following section describes two of the personae, with all 7 profiles documented in the Appendices.

API use cases and user journeys

API Product Manager: the human interface between internal company teams and external API consumers

Grace is an API Product manager in a large company. Her daily role involves supporting various teams as they manage the technical aspects of APIs (advancing their design and development) and to be the interface with the community of external developers using the APIs: responding to their feedback and requests, building a community and resource library so consumer needs are met by the APIs, and making sure they receive value from the use of APIs. She discusses APIToS with different teams within the company (including platform business and legal teams as well as with the technical teams responsible for the APIs performance). One of her key challenges is to ensure that proposed API features requested by users are always aligned with the company business model.

API Provider Persona Overview

User	Age	Country	Profession	API knowledge background
Grace Hopper	35	 USA	**API Product Manager**	**High**

Story with API

Grace is API PM at a large corporation. She knopws well hot to run a large corp projects work and had a technical background. She believes in open models and platform models and had decided to become an API PM to apply it at scale.

API uses (what kind of API, frequence, name, etc.)

She is responsible of different APIs on the company portal, as much about the design, development, security. She talks to business analysts about pricing plands and customer acquisition to develop the API Business model.

APIToS uses (reading, writing, etc.)

She is involved in the APIToS discussion on the Developer onboarding process and gives back feedbacks to the platform team and legal team about users.

Interactions with API providers, legal team, etc.

She consults what other API providers are doing and talk with internal stakeholders to align the API with company business model.

Motivations/issues with API

The goal with API is to maximise the developer activation rate and the lifetime value of API consumers.

API use cases and user journeys

Developer Consuming APIs: integrating external software components to work faster

Ada is a developer who regularly makes use of open source technologies and third party external APIs. She sees APIs as external software that she can integrate to make her projects go more smoothly, and to build her solutions (whether that be an app or a an automated business workflow) faster. She reports to an architect lead, so when she is first building solutions, she jumps over the APIToS and integrates the API to test whether it will work in her proposed solution. If all goes well with her proof of concept, she lets her Architect Lead know what APIs have been used in her project and she assumes there is some review around the external API usage at that 'higher up' level.

API Consumer Persona Overview

User	Age	Country	Profession	API knowledge background
Ada Lovelace	33	Scotland	**Developer**	**High**

Story with API

Ada works with software open source software and 3nd party APIs daily. She also develops internal APIs. For her, an API is just an external piece of software that can help her achieve her goals.

API uses (what kind of API, frequence, name, etc.)

Her company project builds and consumes internal APIs for their own application. They also consume 3nd party APIs for specialised capabilities like AI, SMS, Payments, and Speech to text.

APIToS uses (reading, writing, etc.)

Ada never reads API Terms of Service she just ticks the box "I have read and accept API ToS" to get her API credentials as fast as possible to be able to try the API. She reads the docs, the pricing and the licence of Open source SDK's and libraries.

Interactions with API providers, legal team, etc.

Ada works with her Product Manager who reports directly to the Chief Product Officer. She knows there is someone in charge of every contract but she never has real interactions with them. Sometimes she goes via a 3nd party lawyer to establish all the SaaS contracts and Terms of Service.

Motivations/issues with API

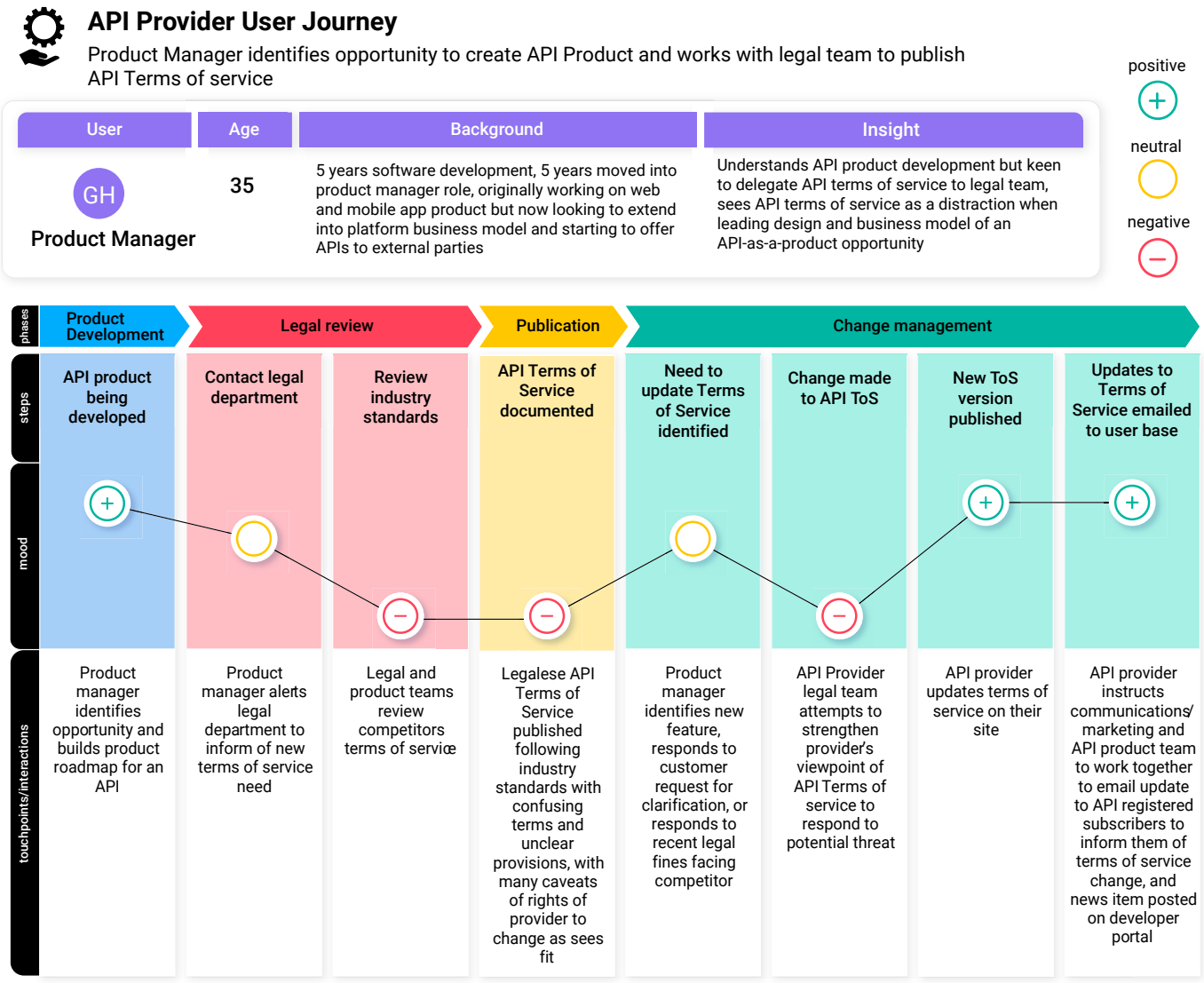
For Ada, APIs are just access to software that can be useful - it just happens to be software made by someone else and accessible via HTTP. She uses lots of open source packages, and APIs are just another tool to solve a problem.

User journey maps

As a complementary resource to the personae, we outlined two typical user journeys in order to understand the different phases of interaction with API Terms of Service depending on the user role. For this exercise, we looked at an API Product Manager like Grace from the API provider lens, and an API Developer like Ada from the API consumer lens.

API Product Manager: Identifying opportunities to create API products and working with legal teams to publish API Terms of Service

A product manager like Grace wants to create an API product. Four phases of this journey were identified: product development, legal review, publication, and change management. The most challenging stages for Grace relate to the engagement with legal teams. For example, she might start by reviewing existing APIToS of competitors. This takes time and is complicated by the lack of standards in the industry outlining what should be included in an APIToS so each one she is reviewing is different making comparison confusing. Then, when she is drafting the APIToS, she works with the legal team that is used to putting clauses in legalese text that Grace herself finds difficult to understand. Grace will also be responsible after the APITos is published to keep API consumers when there are updates or changes to the APIToS and she must create processes to communicate to users, even though she does not have contact information for all API consumers in her community.



API developers: Finding, assessing and integrating an open/public API

For API developers like Ada, five different phases were identified (need discovery, test and review, decision and deployment, and commercial production) in the user journey to find, assess and integrate an open/public API. For an API consumer, the first negative experience occurs when signing up to the developer portal. At this stage, the developer often accepts the API Terms of Service without reviewing them in depth. While this helps them work quickly towards testing the value of the API against their use case, passing over this step creates some challenges further down the track when they are giving a demonstration of their proof of concept. At this stage, their Architect Lead or another member of their leadership team may ask about any identified risks with using this external API. Security, legal or risk management teams may then get involved and slow down any further work towards completing the proof of concept project until APIToS clauses are understood and agreed. This may affect Ada's key performance indicators or work performance as her project is put on hold and begins to fall outside of deadlines as this work is conducted. Finally, when the project is moved to production use case, Ada must ensure that she has set up alerts to monitor if any APIToS changes are made so that she can alert internal stakeholders that must again review the decision to keep the project in production with any new legal clauses being introduced.

API Consumer User Journey

How an API developer finds, assesses and integrates an open/public API

positive
neutral
negative

User	Age	Background	Insight
AL Developer	33	3 years software development, including consumption of multiple APIs	Can use whatever APIs they like although their manager needs to sign off before they go into production. Often uses APIs in proof-of-concept projects or to integrate functionalities like payments (Stripe/Adyen), in-app messaging (Twilio), or mapping (OpenStreetMap)

phases	Need	Discovery		Test and Review		Decision & Deployment		Commercial production
steps	Identifies need	Seeks solution	Assesses options	Signs up to developer portal	Reviews documentation in greater depth	Implement API in proof of concept	Proof of concept assessed for possible use of API in production	Where accepted, API used in production build	API consumed in production use case
mood	neutral	positive	neutral	negative	positive	positive	negative	negative	negative
touchpoints/interactions	Developer identifies need through business requirements or product roadmap planning	Developer identifies options through Google search, API tracker, RapidAPI, Postman, ProgrammableWeb	Developer signs up for APIs or uses interactive documentation in online docs to test APIs available and reviews use cases to assess functionality and fitness-for-purpose	Developer signs up online and accesses free plan, and accepts Terms of Services with minimal review in order to test APIs in proof of concept approach. Confirms account to get API credentials	Developer follows getting started guides to begin implementations, downloads helper libraries, code samples, SDKs	Developer implements the API, debugs where necessary	Developer prepares proof of concept for internal demo and presentation to product manager, Developer unable to respond to questions about risks, impact of using external API as part of commercial production because Terms of Service are confusing or opaque on use case need and change implementation policies	Reluctantly, organisation instructs developer to build API into use case. Some APIs require approval for use in commercial application, so developer needs to prepare documentation to demonstrate how the API is being used, in other cases developer receives authorisation to sign up for paid API subscription and tries to check API Terms of Services more carefully and identify who to inform internally about key provisions	Developer establishes alerts for changelog and status page updates so that they are informed. Unsure of how to update CI/CD processes to ensure they are in compliance with Terms of Service, or whether they need to

Users and use case journeys in practice: findings from research

Interviews and community responses validated some of the concepts we outline in the user personae and user journeys. We found that they range true for both community and business-based approaches.

Ecosystem/community-driven approach

In the context of open science practices, open access to data and publications are anchored in some legal rules (such as licenses). Research respondents noted they have found it easier to create value from data based on the availability of APIs, as this researcher mentioned:

> ***"The large digital archives are now APIed, it is very easy to interface and it works very well in an open science framework precisely because, as long as we are open data, we know that we can connect via the APIs afterward to do the processing we want, it really simplifies the use of these data."***
>
> Interviewee

When the API gives access to resources that are already described under free or open licenses (source code, data, content), most of the interviewees transfer the intrinsic moral and community-driven value of the resource to the interface.

Business approach

Where the API is provided by a for-profit company as-a-service, or where costs of providing the API are passed on the API consumers, businesses protect their right to oversee how reuse of the data that is provided, and define their rights to negotiate on service level expectations, price and any potential future changes to their APIs. In the case of API-as a service, the importance is on the possible use of the API and the trustworthiness of services related.

> ***"I systematically look at the services to know what the service level of the API is, how it will work in terms of availability rate and life cycle management? How long are the old versions maintained? How are changes managed? That sort of thing. Are there any particular issues? So, it's not necessarily linked to the APIs, but to the reuse of the service as such for the package in an offer, so, it's a little bit the product that we buy with an API, it's its use as well as technical, legal?"***
>
> Interviewee

PART 03

BUILDING THE FAIR API FRAMEWORK: TOWARD A NEW MODEL FOR API TERMS OF SERVICE

What improvements can be made to APIToS?

Our research study first needed to define what elements are needed to create a trustworthy and sustainable ecosystem. Through research, interviews, focus groups and our survey, we found that there are three main points at which action can be focused:

Enhancing trust between API providers and API consumers

Simplifying the contracting process of implementing an API

Enabling the creation and sustainability of larger API ecosystems

Amongst survey respondents, facilitating trust was seen as the most important element (by 69% of respondents) to focus on:

According to you, what would be the most important criteria to improve API ToS?

Enhance trust between consumers and suppliers

Simplify the contextualization process of API

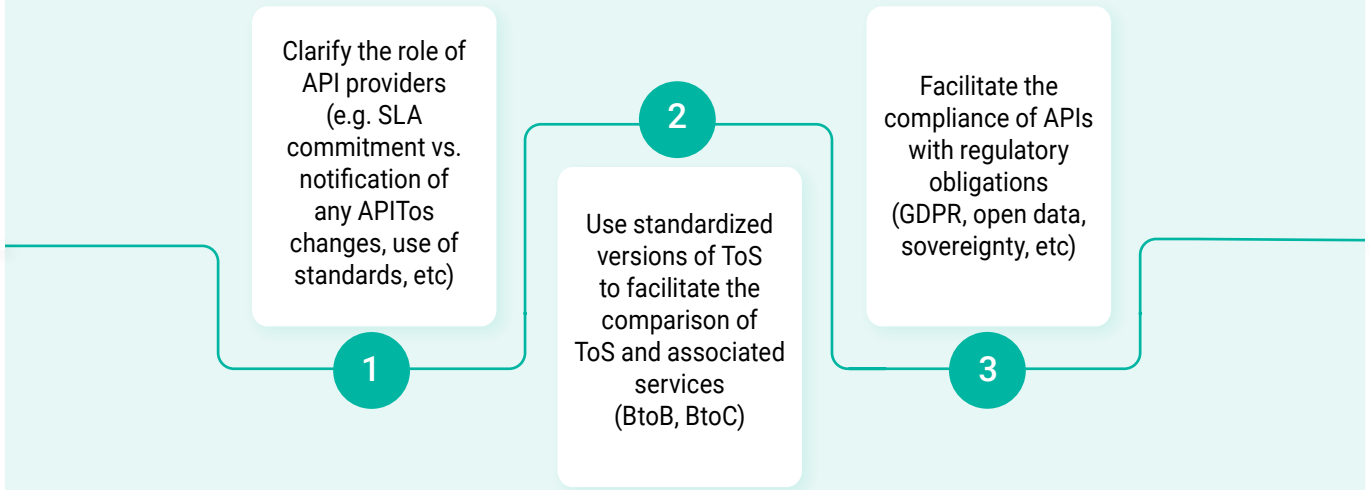
Enable the creation and sustainability of larger API ecosystems

Other

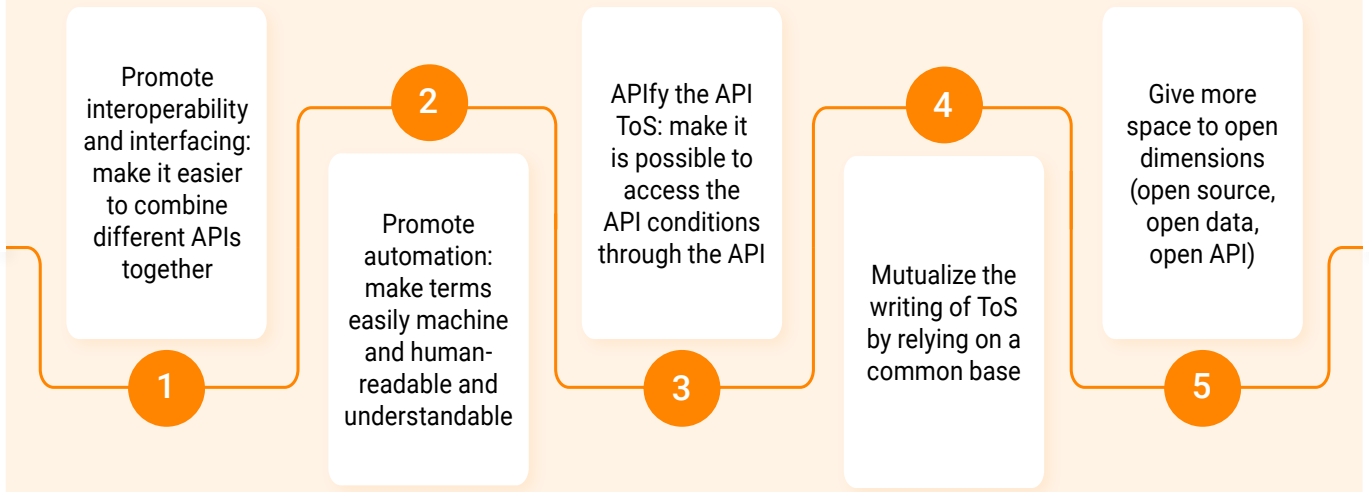
Concerning the **trust**, four actions were proposed:

1. Clarify the exchange framework between providers and consumers: clarify the rights and duties of consumers, the communication framework, and the answers provided by suppliers
2. Facilitate the tracking of ToS: track and anticipate API or API ToS changes
3. Propose a version of the ToS that is easy to understand: use natural language, apply the principles of parsimony
4. Have an open API: make an API accessible to everyone without registration/ authentication required.

Concerning the **simplification of the contracting process of an API,** we noted three main options:



Concerning the **creation and sustainability of larger API ecosystems,** five suggestions were raised:



> ***Let's say, I don't know, a company that sells services to its customers uses my Open API to give it to its customers, wouldn't that mean that all of that company's service becomes free, and that the product it would create with my API, which is free and I want everything done with it to be free, should be free for its customers? Then, is that something that's realistic? I don't know, but you can't, and then it's the producer's choice.***
>
> Survey respondent

Survey respondent comments:

> *Trust is all about consistency. If applications break when something changes, this breaks that level of trust even if it takes a few hours to fix.*
>
> Survey respondent

> *APIToS are often complex to read and so we can be unsure of what they actually mean, so have less trust.*
>
> Survey respondent

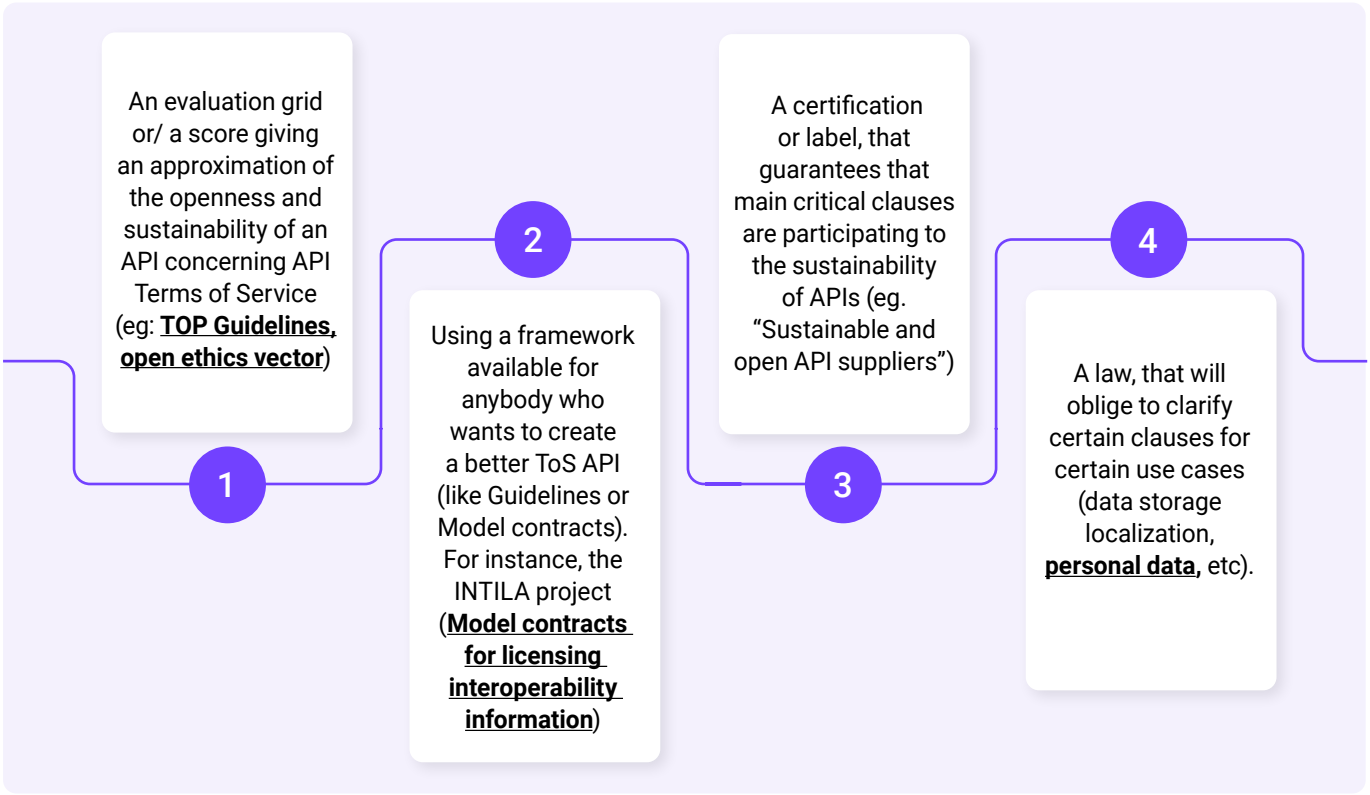
> *Consistent templates would help a lot, so that everyone can easily understand what to expect.*
>
> Survey respondent

> *Having standardized APIToS would be of great help to compare service providers and also to create your own APIToS.*
>
> Survey respondent

> *Ease of onboarding partnerships with automated processes and simple API ToS help grow the API ecosystem.*
>
> Survey respondent

APIToS standardization: different options proposed

Based on research interviews and the exploration of other initiatives at the crossroads between API and legal design, we propose other **potential impactful activities** for more sustainable digital infrastructure using and providing APIs with their API ToS:

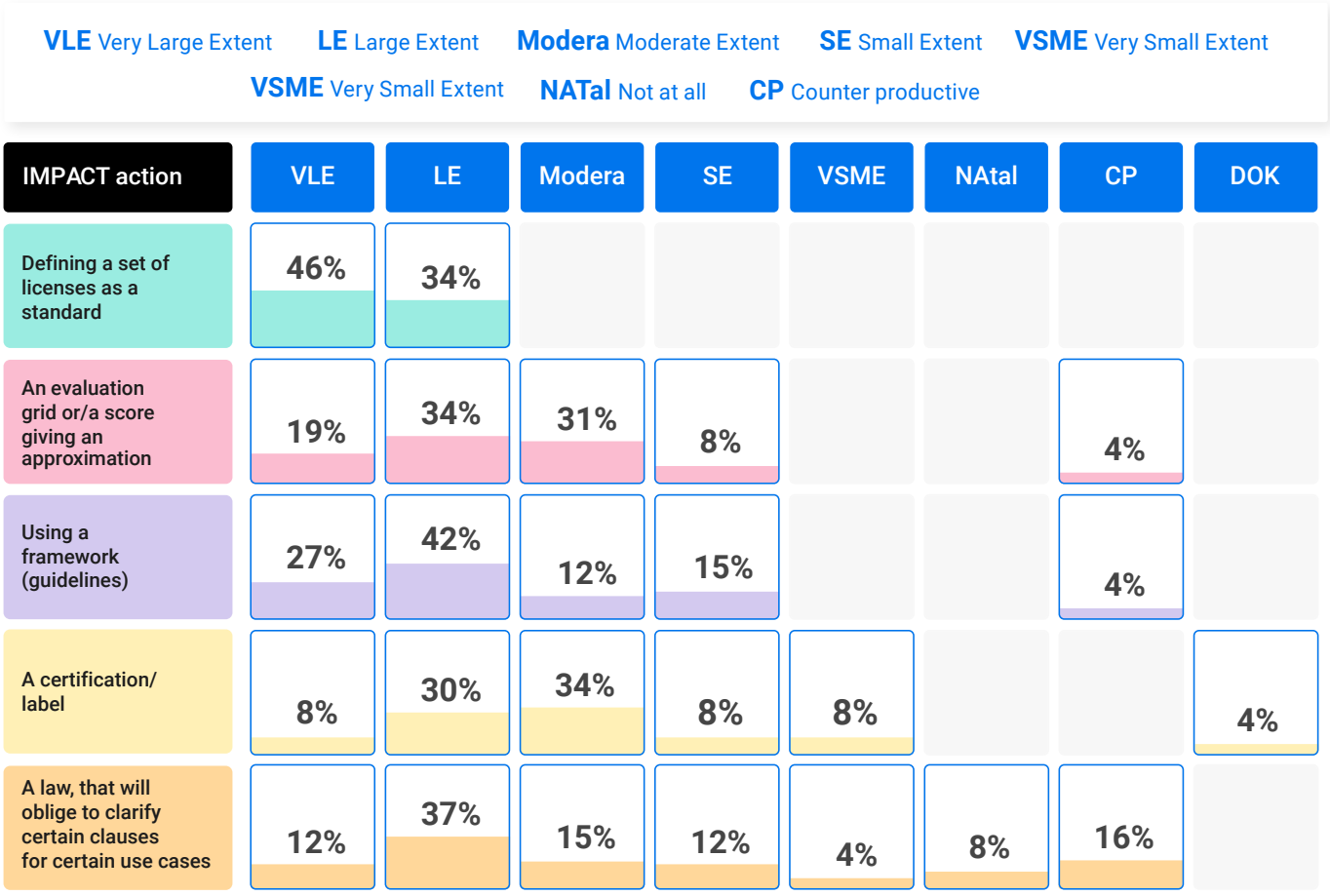


These actions have different levels of application and implications (including benefits) for ecosystem stakeholders.

Benefits for	API Provider	Facilitate interoperability and interfacing between APIs and ToS (time-saving and efficiency)	Give confidence to users	Rely on a common base, so providers do not use resources reinventing the wheel
Benefits for	API Consumer	Greater confidence in the data providers	Reduced power imbalances and asymmetries (communication, usage constraints, adaptation to changes)	Eliminating the economic stakes from the ToS to make them more sustainable
Benefits for	API Ecosystem	Rebalancing power relationships in open ecosystems	Integrating ToS into current dynamics of digital economic regulation (GDPR, data sovereignty)	

Throughout the project's engagement with industry and stakeholders, these options were discussed to identify the most appropriate and pragmatic solutions.

The most favored options were a set of licenses as a standard and the use of guidelines. Laws and a label are less approved or not encouraged by respondents.



Making API ToS readable and understandable: drawing on next generation legal design initiatives

Drawing on legal design examples: making legal agreements more user-friendly and understandable for API ecosystems

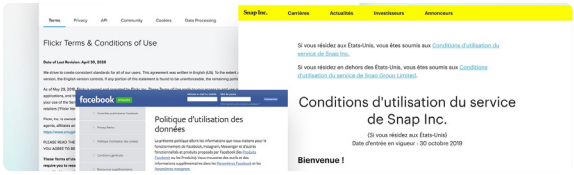
Before proposing solutions to demonstrate the idea of improving the exchange framework between API providers and API consumers, our research reviewed inspiring initiatives at the crossroad between legal design and standardization efforts.

In addition to the Creative Commons model, we studied five other initiatives that demonstrated a legal design approach.

ToS;DR

This project aims to collaboratively gather all Terms of Service with a manual evaluation of how respectful or harmful they are on specific clauses.

https://tosdr.org/



Open Terms Archives

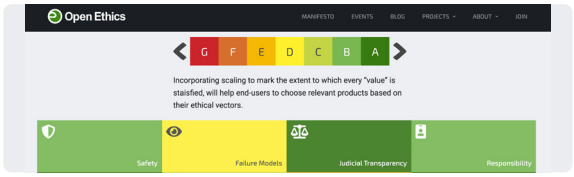
Open Terms Archive is a free and open tool for monitoring and archiving changes to the general terms and conditions of use of the main online service providers. It allows defenders of user rights, regulators, and any interested person to follow the evolution of these conditions.

https://disinfo.quaidorsay.fr/fr/open-terms-archive

Scripta Manent (part of Open Terms Archive)

Scripta Manent is an online tool that allows you to measure the changes between two dates of a contractual document of your choice. Precious for consumers, lawyers, the regulator (who can measure the platforms' commitments), the legislator (who can analyze the logic by which companies try to circumvent their texts), journalists, researchers in law, and legal rech companies. Open Terms Archives is a project aiming to collect, store and compare all Platform Terms of Service and API Terms of Service included for evaluation, notice or forensics.

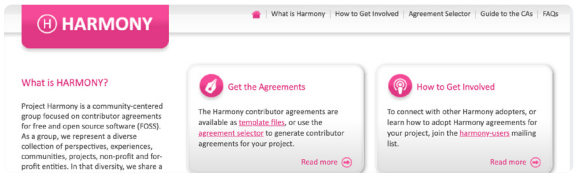
https://disinfo.quaidorsay.fr/fr/open-terms-archive/



Open Ethics

Open Ethics for AI is like Creative Commons for the content. They aim to build trust between machines and humans by helping machines to explain themselves. They're developing an open transparency protocol to help product-owners describe their AI-powered solutions in a standardized, user-friendly, and explicit way. Open Ethics is a global inclusive initiative with the mission to engage citizens, legislators, engineers, and subject-matter experts into a transparent design and deployment of solutions backed by artificial intelligence to make a positive societal impact.

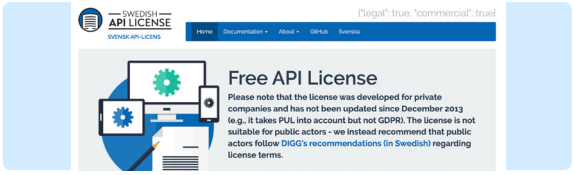
https://openethics.ai/



Harmony Agreements

Project Harmony is a community-centered group focused on contributor agreements for free and open-source software (FOSS), launched in May 2010. Contribution agreements enable more people to contribute code, by reducing the cognitive cost and legal time of reviewing contribution agreements

http://www.harmonyagreements.org/



Swedish API License

The Swedish API License is a project aiming to help companies generate standardized API Terms of Service clauses based on a standardized framework and with an automated wizard.

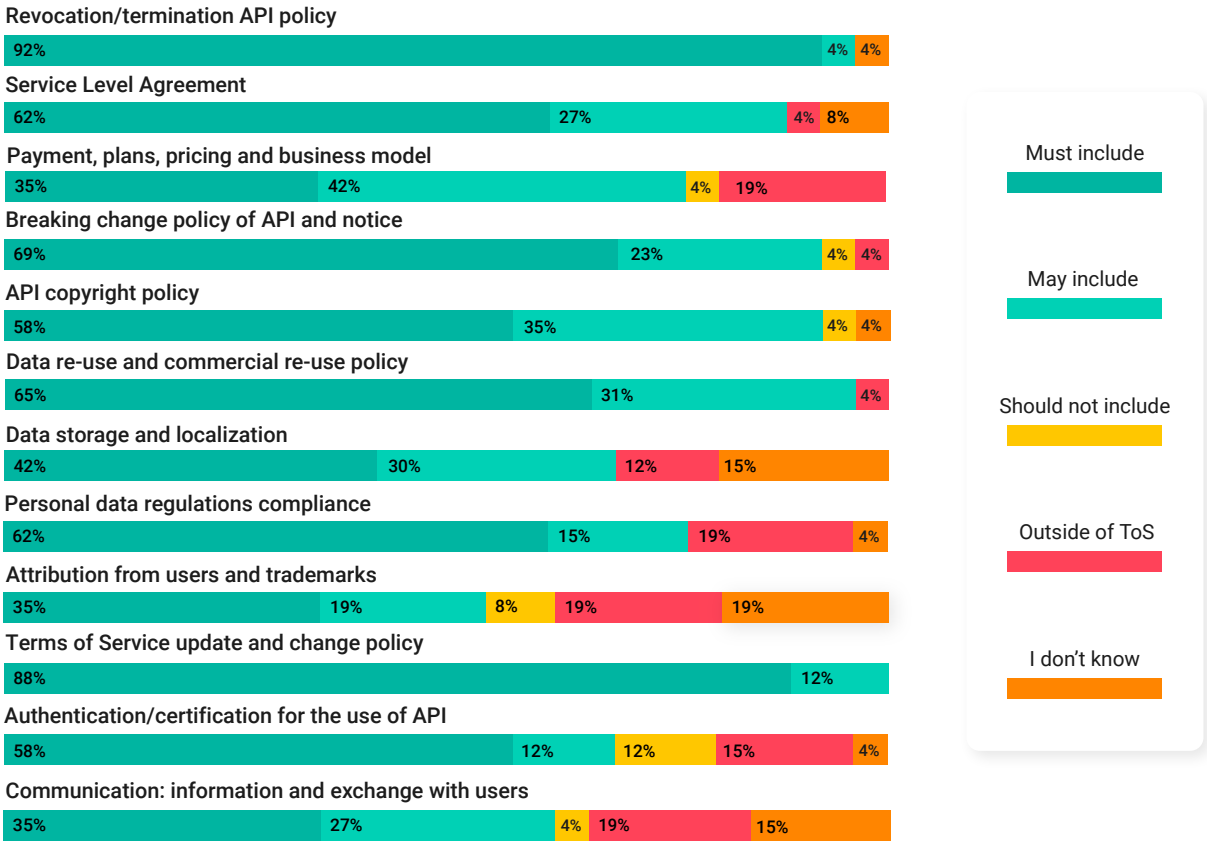
https://disinfo.quaidorsay.fr/fr/open-terms-archive/scripta-manent

Applying legal design lessons to APIToS

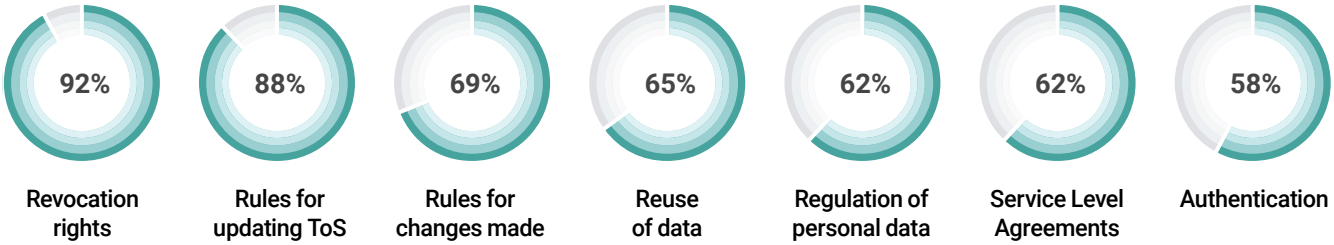
Today, APIToS are far from being standardized. Many API providers draw on existing Terms of Service and adapt them, sometimes just copy/paste-ing some other APIToS without knowing what it means for their business.

> ***Moreover, most ToS of APIs are written by people that have no idea about legal terms so they are useless.***
>
> Survey respondent

To improve APIToS readability, one solution is to standardized the different clauses used in APIToS. Our survey asked respondents which clauses should be prioritized in a standardization approach of APIToS:



The most popular clauses were:



Suggestion from a survey respondent:

> *Start small. Generation zero may be to just identify a common framework to then hang optional clauses on. From there, establishing common working for optional clauses, then for those that make the clauses mandatory - some kind of recognition, branding, or certification. Let the market then decide the utility and path forward.*
>
> Survey respondent

Machine readability of APIToS

One important legal innovation of Creative Commons licenses was the possibility to be usable and readable for humans and machines. Different solutions are available as a link with the API Definition document already existing, like OpenAPI documents. During the study, machine readability has been acknowledged as an important element for more interoperability and regulation between different APIs as this comment highlighted:

> *Their terms of service would all be the same, and there would be a machine-readable discovery to certify and say, yes, actually they all work the same.", which can be the first step for a potential federation of API.*
>
> Survey respondent

Several elements have been attached to a possible network of standardized API and the need of governance of API to regulate the changes:

> *In fact, maybe we need to think about API governance as well, why not in the clauses that describe the API and specify who has the right to change it on their own or what change process they are committing to.*
>
> Survey respondent

This federation could be based on a logic of contribution, a principle of equity, a notion of copyleft but also an ecosystem of open tools such as Open API. For example, some API providers are interested in asking some counterparts or retribution in order to use the API. For instance, some may want to still access the data provided, track queries data and analytics.

According to project workshop participants, a machine-readable APIToS could help in several ways:

Discovery

a developer may want to use APIs only with a certain type of clause, related to a company's SLA. This can be integrated into API marketplaces or search engines.

Facilitate API mashups

the ability to build and integrate APIs in integration platforms could be better managed to address rate limits of various APIs. NoCode/LowCode ecosystems could check API rules automatically, especially regulations between different countries at the API contract level.

According to workshop participants, implementing machine readability could occur at several points:

CRM/Developer relationship management

in API management solutions could be used for messaging communities in an automated way.

OpenAPI specification and Schema

could create a schema object defining consumption, production, and publishing. As one workshop participant noted:

> *In fact, maybe we need to think about API governance as well, why not in the clauses that describe the API and specify who has the right to change it on their own or what change process they are committing to.*
>
> Survey respondent

In HTTP headers

As some headers already show rate limits, links in the header could also point to more clauses of the APIToS.

There are direct use cases for machine-readable API Terms of Services, while we need more understanding on the human level to be able to automate things and enforce rules automatically. It may be also complex to find the limit between what can be automated and what part of the law will stay out of machine readability. It may be easier for technical clauses where we can put a number on it (rate limits, SLA, quota, cache time, etc).

Proposing the Fair API Commitment to Trust (FACT) Framework

Dedicated to the fundamental commitments for the establishment of a trust framework, the FACT “FAIR API Commitment to Trust” framework proposed a set of compatible clauses allowing the use of such a tool in different situations (business, research, open, etc.) and to make more visible some additional commitments such as openness (open source licenses, open data, copyleft, etc.).

FACT - FAIR API Commitment to Trust					
Category	Fair use policy			Loyal change policy	
Details	Rules on API users' rights to catching, storage and maintenance and services via the API			Fairness and Transparency of change/ Guarantees on breaking changes	
Pictogram	This API has a fair use policy 			This API has a fair loyal change policy 	
Category	API access	API specification	Ethical data policy	Loyal output policy	Reference and attribution
Details	Rules on reuse and access to API	Rules on ability to reuse with or without modifications	Rules on reuse data exposed via the API	Rules on reuse of outputs from API usage	Rules on how data and services enabled via API should be explained to external parties
Default	API Neutrality	CC Zero Licence on specification	Large data reuse	All commercial reuse allowed	Attribution requirement
Pictogram					
Options to customize					
Options to customize 1	Restrictive access rights	Share Alike licence on specification	Open Data contract	Non direct competition	No attribution
Pictogram					
Options to customize 2	Share Alike		Commercial Data contract	Non commercial reuse	Trademark enforcement
Pictogram					

This prototype without legal design layers was one of the main resources explored during project workshops. The first draft can be used as an explanation page that can be added to any website using it.

How to implement the FACT framework: feedback from a co-design workshop

Through a “New Models for API Terms of Services” focus-group/co-design workshop, several suggested improvements were proposed:

Framework appropriation by potential users

Framework integration into daily tasks of users

Possible use of the Framework in an automatic manner by machines

In general, the idea of a FACT Framework has been acknowledged as a useful tool to support API providers to ensure they have clarified key provisions when making an API available to their identified audience, and to help API consumers navigate to key areas of a Terms of Service.

While API terms of Service structures can be improved, one main point made during co-design workshops is that API Terms of Service (API ToS) themselves are only one mechanism that can be used to build trustworthy relationships and to foster trust between API providers and consumers.

Several other mechanisms within and beyond the APIToS were suggested.

Within the Terms of Service

- Be sure that the framework is working for all kinds of API
- Should be written in plain language with some short sentences to easily describe each clause
- A Terms of Service could start with a list of DO’s and DON’Ts (or MUST/ SHOULD/etc) to more clearly explain key provisions. For consumers, this would include understanding commercial use, appropriate usage, and how changes will be communicated/addressed
- Have some pictograms (not more than 6 to avoid complexity) related to this list of do’s or don’ts at the beginning or some warnings
- Links within the other contractual document to help navigate where possible

Beyond Terms of Service

- Key constraints and clauses of the terms of service should also be described in other developer resources including getting started guides, onboarding emails, resources to support developers when moving from sandbox to production APIs, etc
- In a FACT framework webpage, be sure that the link with other agreements is updated or add a sentence to indicate the commitment of the providers to give access to the other agreements (APIToS are not the only document to build trust with users)
- Indicate clearly the link with other initiatives (open terms archive, etc.)
- Trust labels from an independent authority that could be used as a trust signal on the developer portal homepage, reviewing a developer score and reviews by developers on the value of the API

PART 04

THE FUTURE OF APITOS FOR ENFORCEABLE, OPEN AND SUSTAINABLE DIGITAL

Exploration of FACT new business models for trust

In a direct transposition of the trade credit insurance model that guarantees the financial and technical stability of companies trading with each other, better standardized FACT can lead to a digital insurance model that guarantees the financial and technical stability of an API connection between two companies.

Trade credit insurance is used when a company delivers products to another company, and accepts to be paid later. The “gap” between the delivery and the payment is a financial risk, because if the company goes bankrupt during that period, the shipping company will never get its money back.

To address this risk, independent third party companies rate the financial stability of each company and depending on the rating, they guarantee (or not) the payment of invoice to the supplier. By knowing that an insurer guarantees the invoice to be paid, the supplier can make better payment conditions to the customer. It augments trust because of an existing contract that can be verified by a third party.

Standardized FACT can play a similar role to the third party insurance contract that companies engage. If companies fail to respect their engagements, a liability and compensation will be given to the other party.

APIToS & FACT: a pillar for the future of open science

Since the beginning of the internet, scientific and academic communities have been one among the first to support the principles of free circulation of scientific information and the freedom and equality of users to access information.

Open access to scientific publications was advocated by researchers, librarians, and free software activists to fight against a retention of scientific articles behind paywalls. This practice is considered unfair because researchers write articles with public funds and without payment. Having to pay subscriptions (through university libraries) moreover for digital versions (which cost less to produce) appeared as an aberration.

Today, open access has become a prerogative of the funding institutes for any publication of articles. This has required the scientific publishing world to adapt and propose new models (for example, subscription-enhanced access under ‘diamond’ or ‘gold’ levels of additional services). More than scientific articles, another question quickly arose in the scientific world: how to open access to the data and source codes of research? What is the point of publishing an article if the computational method used and the associated data cannot be verified but also re-used by other researchers? Under the terms of open science, these principles of openness and sharing are now at the heart of public research policies that try to balance the relationship between publishers and data providers and to reduce the possible abuses of some monopolistic actors.

If open access seems to be part of the new habits of these players (notably pressed by measures such as the PlanS), the devil is in the details. If we look closely, especially at the core of publishing and/or platform terms of use contracts, we quickly notice that closing mechanisms are at work. These are often more subtly hidden in the terms of services of digital infrastructures and APIs.

To illustrate this point, one of the most striking examples is the contract signed between the Swiss academic libraries consortium and Elsevier, an international scientific publisher often boycotted for its closed publishing practices (see for example the campaign The Cost of Knowledge). In several countries, academic libraries are grouping together in consortia to negotiate subscription contracts to their advantage with major players in scientific publishing. The Swiss academic consortium in a press release of 2019 highlighted the negotiations done with three major publishers (Elsevier, Springer Nature and Wiley) and the progress made in terms of open access (a pilot case of gold open access). In the face of this supposed embellishment, a more critical reading of the contract with Elsevier allows to bring some nuances. New and much more subtle mechanisms of control and closure of information were found in their APIs.

By reading the contract, we discovered that the Elsevier API is made available to access this publisher's gigantic database and thus allow for Text and Data Mining (TDM). TDM gathers a set of methods to extract information from such databases and to process a large mass of data in an automatic way. These practices are essential today to better understand scientific trends and tendencies. For institutes, it is also a way to be able to evaluate the research conducted within their community of researchers. However, in this contract, it is clearly noted that each product resulting from the API remains the property of Elsevier and is subject to a copyright and a set of reserved rights (far from the principles of licenses favored by open access and open source).

Thus, while there are efforts made to open up publications, other research objects (for example databases and data produced via APIs) are still subject to controls and restrictions on use that are contrary to the principles of open science and open access. And this in a situation where the economic models are moving from the rent from resources (payment of articles) to the possible valuation of a large amount of data (creation of recommendations, services, etc.) This example taken from the world of research shows how the dynamics of openness, whether in terms of open science, open data, or open access, must also be reflected in digital infrastructures to ensure that they are fair for all actors in a digital ecosystem.

APIs are on the one hand an excellent example of an opportunity for structured and efficient information exchange in scientific circles to facilitate information sharing and mutual benefits between several actors. For instance, the use of APIs is a must for sharing high value-added data (meteorological, statistical) between private and public actors. On the other hand, APIs can become an element of control and inequality between different actors if the rules they enforce favor withholding information from the owner or if they contain abusive clauses (frequent changes, etc.). Today, several research projects, particularly in the humanities and social sciences, are dependent on the APIs of large platforms (Twitter, Facebook, YouTube, etc.) for their analyses and it remains unclear as to the terms and conditions of acceptance of their project requiring access to these APIs.

For the opening of data, source codes, articles, and images in a common and general interest requires to understand the different layers that are intertwined in digital projects and their necessary openness but also interoperability. What is the point of accessing an article if its use in a database is limited by the conditions of use of an API?

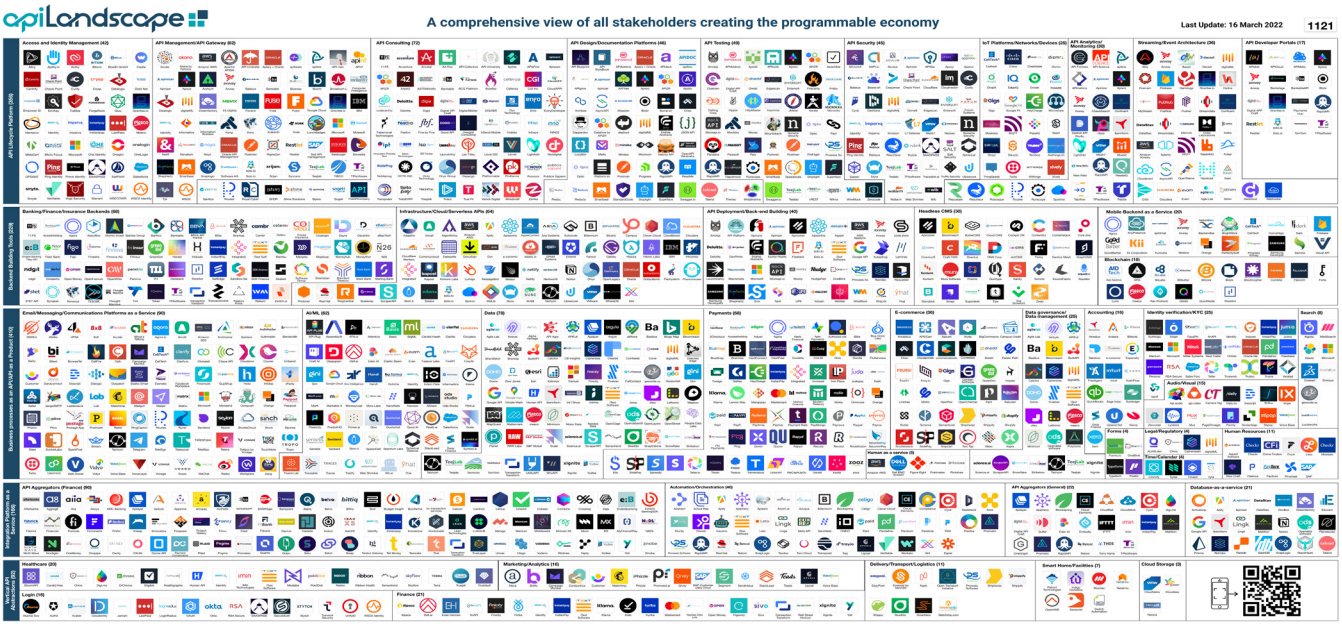
The first intuitions of the APIToS-Creative Commons project were born from this still not very visible observation of possible threatens for open science through the APIToS and the rules they establish.
With more than a year of work between complementary profiles, entrepreneur and API experts, lawyers, researchers and open activists, the APIToS project and its materialization by the FACT framework offers possibilities to act within different environments.

For research, FACT can be both an awareness-raising framework for developing vigilance in reading APIToS platforms used for research. It is also a guide to facilitate the drafting of APIs in a context where legal insights may be lacking. And finally, FACT plays the possible role of a guide to take into account the important clauses to be part of a process of openness and the creation of a fair and trustworthy ecosystem. This project is a first step that in the spirit of being free and open, has the opportunity to be shared, tested and remixed in different sectors.

Recommendations

Integrations with current API tooling ecosystem

Over the last 15 years, the digital infrastructure industry has seen a large increase in API lifecycle tooling and API management. The current API tooling landscape ecosystem includes more than 1100+ tools strong and still growing, with more and more open source tools to design, build, develop, test, deploy, manage, secure, publish and version APIs, In this ecosystem, there are many tools or initiatives that



could directly benefit from FACT documents. This ecosystem has been thriving also because of more standardization and conventions enabling more interoperability. ***We recommend that FACT be integrated into the ecosystem to help improve trust for digital infrastructures.***

FACT as a reference in APIs.Json

APIs.json is a machine readable specification that API providers can use to describe their API operations, similar to how web sites are described using sitemap.xml. Providing an index of internal, partner, and public APIs, which includes not just the the OpenAPI document, JSON Schema, and other machine readable artifacts, but also the currently only human readable elements like documentation, pricing, and terms of service. ***We recommend that a FACT document be added in the apis.json structure as a reference.***

Integration of FACT definition into API descriptions, especially OpenAPI

What is an API Description?

- **Description Validation and Linting:** Check that your description file is syntactically correct and adheres to a specific version of the Specification and the rest of your team's formatting guidelines.
- **Data Validation:** Check that the data flowing through your API (in both directions) is correct, during development and once deployed.
- **Documentation Generation:** Create traditional human-readable documentation based on the machine-readable description, which always stays up-to-date.
- **Code Generation:** Create both server and client code in any programming language, freeing developers from having to perform data validation or write SDK glue code, for example.
- **Graphical Editors:** Allow easy creation of description files using a GUI instead of typing them by hand.
- **Mock Servers:** Create fake servers providing example responses which you and your customers can start testing with before you write a single line of code.
- **Security Analysis:** Discover possible vulnerabilities at the API design stage instead of much, much later.

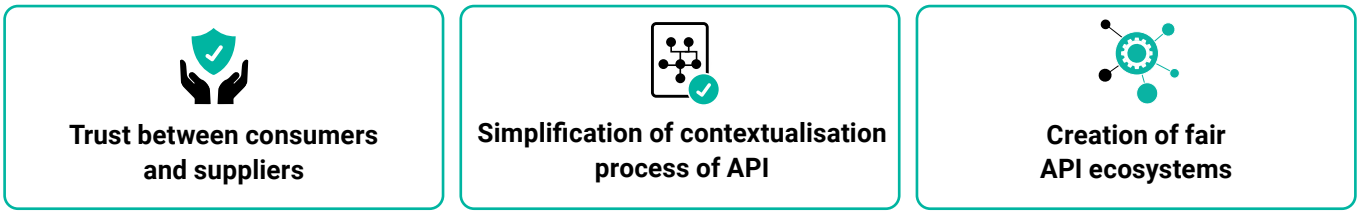
We recommend that FACT \be included directly as part of the API description. Then, the generation of clients from the API Description could include directly the API terms if service directly into API gateways and API management solutions, enforcing them at runtime and automatically, instead of staying read-only hand written legal documents.

CONCLUSION

APIs involve multiple stakeholders (API providers, API consumers, regulators, end users, activists) as we described with the persona and experience map. Everyone has their own knowledge of APIs and APIToS and various goals and aims influenced by their business/openness culture and the status of resources they are using (proprietary data rather than open data, open source software, etc.).

Our study highlighted the numerous inequalities that exist between API providers and API consumers, which are embedded in the APIToS contract. Frequent changes, lack of consistency, and opacity of legal documents are the main concerns raised by this study. This faces API consumers, developers and also API providers who may be dependent on some of the bigger platforms like Facebook, LinkedIn, and Twitter. These inequalities can impact the business model of companies but also impact public interest, as they affect research studies using digital methods and could reduce the transparency of information available related to public transport, government, etc. API consumers also face a lack of understanding of APIToS documents. Few people read them because of the complexity, lack of standardization and lack of time. The first priority for many developers is to first test an API, and later they think they will revisit and consider the Terms of Service…Long term problem become more difficult to anticipate.

With this overview in mind, three main leverages have been proposed to build a **standardized APIToS framework**:



Building trust with the clarification of an exchange framework has been acknowledged as well as the development of standardized version in order to facilitate compliance. From a technical standing point, automation by using APIs has been also raised and shows the importance of machine readability as one part of the answer. Some clauses from the various APIToS have also been prioritized by the participants of the study as main elements: revocation rights, rules for updating ToS and changes made, reuse of data, regulation of personal data and SLAs.

All of these research results were tested during interviews and a survey to help us to design a prototype of APIToS framework that could be easily used by all stakeholders. One of the challenges was to build an attractive format in the context of main users that are not used to reading the APIToS. More than a perfect theoretical model, the model must be adapted to the current ability of the API practitioners to understand and assimilate easily.

For this purpose, we called upon inspiring sources of legal design and the use of co-design workshop to propose the FACT (FAIR API Commitments for Trust framework).

The model proposed in this document and the recommendations generated during focus groups about the appropriation of the model by the different potential FACT framework users as well as through machine-readable mechanism is now available for further debate and consideration. We hope that this first model can be the object of constructive feedback and proposition of implementation in various projects to participate pragmatically in the creation of a fairer API ecosystem and more globally to more sustainable digital infrastructures.

Glossary

As-a-service (aaS)

aaS is an acronym for as a service (e.g., X as a service), and refers to something being presented to a customer, either internal or external, as a service, always in the context of cloud computing. As a Service or XaaS (Anything as a Service) offerings provide endpoints for customers/consumers to interface with which are usually API driven but can commonly be controlled via a web console in a user's web browser. We often talk in Digital Infrastructure about SaaS (software as a service), PaaS (Platform As a Service), IaaS, Infrastructure as a service) but many others are possible

APIs

Application programming interfaces (APIs) are machine-to-machine digital interfaces that facilitate the exchange of data and services (functionalities). - Vaccari, et al 2020

"An API is not only a building brick, it is also a projection of a product vision, based on internal assets you can open to the world." - Mehdi Medjoaui, 5 Ways an API is More Than An "API".

API stands for Application Programming Interface, as the interface exposing, representing, and powering the access and the transfer of resources between applications, locally or over the network. APIs enable software programs to interact with other software programs in the consumption, the adding, the processing, the modification or the deletion of data.

Commons

Shared resources (cultural, natural, digital) held in common by a group of people/stakeholders (users group, communities) with a set of governance principles (norms and values) to ensure the individual and collective benefits of the use of this resources. Commons are defined as a third path between state and market (Ostrom 2015).

Copyleft

In open source, copyleft is the arrangement whereby software or artistic work may be used, modified, and distributed freely on the condition that anything derived from it is bound by the same conditions.

Digital ecosystems

Digital ecosystems can be described as: “a number of firms - competitors and complementors - that work together to create a new market and produce goods and services of value to customers” (Hazlett 2011). A digital ecosystem is a network of participation opportunities that allow stakeholders (including governments and regulators, associations, industry enterprises, small and medium enterprises, researchers, community groups and individuals) to co-create, collaborate, complement, coordinate, and/or compete with each other interdependently by using common infrastructures and tools (such as open standards and data models, APIs, and open source technologies) (European Commission 2007; Boyd 2021). A digital ecosystem has the possibility to solve complex and dynamic problems automatically, while drawing on properties similar to the characteristics of biological ecosystems such as robustness, self-organization and scalability (Briscoe, Dewilde 2009). ‘Open ecosystems’ are those that allow all stakeholders to participate, creating more equitable opportunities for participation (Boyd 2021).

Digital infrastructures

Socio-technical systems in a digital environment that share some key features with physical infrastructures such as ubiquity, reliability, invisibility, and gateways (Plantin 2018). They play a critical role today in enabling digital economies, and are considered faster and cheaper to build compared to physical infrastructures. Digital infrastructures change often (requiring ongoing maintenance) and are characterized by frequent change of usage and adoption. They are not associated with a central organization (decentralized nature) and tend to create new challenges for their management (Eghbal 2016).

In this research, we make the distinction between Digital Infrastructure as the legal framework for digital assets (as software, standards, and protocols) and Digital Infrastructure as the global ecosystem of digital assets powering digital experiences including servers, operating systems, frameworks, and applications, representing a digital software supply chain where software consumes other software on the same server or via another server via the network. We often refer to this as Digital Infrastructure as as-a-service, as it is consumed over the networks via APIs. Although we include the first in our research, we concentrate efforts on the second as it seems to be the one under the pressure of API Terms of service issues.

Infrastructure-as-a-Service:

A cloud-based service that provides physical, virtual, and additional storage networking products (Mohammed, Zeebaree 2021).

Platform-as-a-Service

Cloud-based infrastructure for the development of
applications and technologies that are distributed over the Internet, without the need for the end user to download or handle the user interface (Mohammed, Zeebaree 2021).

Resource

A resource is anything digital that can be accessed over the network, especially across the web. On the web, a resource can be data, an audio file, a video file, a text document, a set of software functions or operations accessible behind an endpoint at a specific location URI or URL) Resources are accessible via API endpoints by other software programs.

Service

In the contexts of digital software architecture, and service-oriented architecture, the term service refers to a software functionality or a set of software functionalities (such as the retrieval of specified information or the execution of a set of operations) with a purpose that different clients can reuse for different purposes, together with the policies that should control its usage (based on the identity of the client requesting the service, for example). In the context of Digital infrastructure, a service can be executed locally or from a 3rd party digital infrastructure over the network, as-a-service.

Software-as-a-Service

A cloud-based software delivery model in which an application is hosted by a distributor or provider and made available to customers over the Internet (Mohammed, Zeebaree 2021).

Bibliography

Janneke Adema, Graham Stone. 2017. “Changing Publishing Ecologies. A Landscape Study of New University Presses and Academic-led Publishing”. Joint Information Systems Committee. https://pureportal.coventry.ac.uk/en/publications/changing-publishing-ecologies-a-landscape-study-of-new-university

Anon. 2017. “Google is shutting down the QPX Express API for airfare data”., Hacker News. https://news.ycombinator.com/item?id=15594975

Anon. 2012. “‘Twitter killed my business.’ An inside look at the ecosystem crackdown”. GigaOm. https://old.gigaom.com/2012/09/07/twitter-killed-my-business-an-inside-look-at-the-ecosystem-crackdown/

Seth Benzell, Jonathan Samuel Hersh, Marshall W Van Alstyne, Guillermo Lagarda. 2019 (Revised 2022). “How APIs Create Growth by Inverting the Firm” SSRN. http://dx.doi.org/10.2139/ssrn.3432591

Mark Boyd. 2021. “Measuring the value of open ecosystems: 1. Our model”. Platformable. https://platformable.com/measuring-the-value-of-open-ecosystems-1-our-model

G. Briscoe, P. De Wilde. 2009. “Digital Ecosystems: Evolving Service-Oriented Architectures”. Cornell University. https://arxiv.org/abs/0712.4102

Paul N. Edwards, Steven J. Jackson, Geoffrey C. Bowker, et Cory Philip Knobel. 2007. “Understanding Infrastructure: Dynamics, Tensions, and Design”. Working Paper. http://deepblue.lib.umich.edu/handle/2027.42/49353.

Paul Edwards, Geoffrey Bowker, Steven Jackson, Robin Williams, et University of Edinburgh. 2009. “Introduction: An Agenda for Infrastructure Studies”. Journal of the Association for Information Systems 10 (5): 364-74. https://doi.org/10.17705/1jais.00200.

Nadia Eghbal. 2016. “Roads and Bridges: The Unseen Labor Behind Our Digital Infrastructure”. Ford Foundation. https://www.fordfoundation.org/media/2976/roads-and-bridges-the-unseen-labor-behind-our-digital-infrastructure.pdf

European Commission, Directorate-General for the Information Society and Media. 2007. “Digital Business Ecosystems”. Publication Office of the European Union. https://op.europa.eu/en/publication-detail/-/publication/53e45e55-4bd2-42a4-ad25-27b339b051e0/language-en

Thomas W. Hazlett, David Teece, Leonard Waverman. 2011. “Walled Garden Rivalry: The Creation of MobileNetwork Ecosystems.” George Mason Law & Economics Research Paper No. 11-50. https://www.researchgate.net/publication/228254636_Walled_Garden_Rivalry_The_Creation_of_Mobile_Network_Ecosystems

Anne Helmond, David B. Nieborg, Fernando N. van der Vlist. 2017. “The Political Economy of Social Data: A Historical Analysis of Platform-Industry Partnerships”. #SMSociety17: Proceedings of the 8th International Conference on Social Media & Society p 1–5. https://doi.org/10.1145/3097286.3097324

Chris Kemp. 2021. “APIs & Software Copyright in 2021 – A View from each Side of the Pond”. Scottish Society for Computers and Law. https://www.scl.org/articles/12332-apis-software-copyright-in-2021-a-view-from-each-side-of-the-pond

Kenji E. Kushida, Jonathan Murray, John Zysman. 2011. “Diffusing the Cloud: Cloud Computing and Implications for Public Policy”. Journal of Industry Competition and Trade 11(3). https://link.springer.com/article/10.1007/s10842-011-0106-5

L. Lessig. 2008. “Remix: Making Art and Commerce Thrive in the Hybrid Economy”. Bloomsbury Academic. UK.

Ingrid Lunden, Drew Olanoff. 2015. “Twitter CEO Dorsey Apologizes To Developers, Says He Wants To “Reset” Relations”, TechCrunch. https://techcrunch.com/2015/10/21/twitter-ceo-dorsey-apologizes-to-developers-says-he-wants-to-reset-relations/

Mehdi Medjaoui. 2014. “5 Ways an API is More Than An ‘API’”. Medium. https://medium.com/@medjawii/5-ways-an-api-is-more-than-an-api-bddcdb0517ca

Peter S. Menell. 2018. “Rise of the API Copyright Dead?: An updated epitaph for copyright protection of network and functional features of computer software”. Harvard Journal of Law & Technology Vol 31, 305-490. https://jolt.law.harvard.edu/assets/articlePDFs/v31/31HarvJLTech305.pdf

Thomas J. Misa, Philip Brey, et Andrew Feenberg. 2003. “Modernity and Technology”. MIT Press.

Chnar Mustafa Mohammed, Subhi R. M. Zeebaree. 2021. “Sufficient Comparison Among Cloud Computing Services : IaaS, PaaS, and SaaS: A Review.” International Journal of Science and Business 5 (2): 17-30. https://doi.org/10.5281/ZENODO.4450129

Casey Newton, 2018. “Twitter officially kills off key features in third-party apps / A sad day for Tweetbot, Twitteriffic, and others”. The Verge. https://www.theverge.com/2018/8/16/17699626/twitter-third-party-apps-streaming-api-deprecation

Elinor Ostrom. 2015. “Governing the Commons: The Evolution of Institutions for Collective Action”. Cambridge University Press. https://www.cambridge.org/core/books/governing-the-commons/A8BB63BC4A1433A50A3FB92EDBBB97D5

Matthew Panzarino, 2012. “Developers, bracing themselves for Twitter API restrictions, call today’s post ‘ominous’”. TNW. https://thenextweb.com/news/developers-bracing-themselves-for-twitter-api-retrictions-call-todays-post-ominous

Jean-Christophe Plantin, Carl Lagoze, Paul N Edwards, et Christian Sandvig. 2018. “Infrastructure Studies Meet Platform Studies in the Age of Google and Facebook”. New Media & Society 20 (1): 293-310. https://doi.org/10.1177/1461444816661553

Jean-Christophe Plantin, Aswin Punathambekar. 2018. “Digital Media Infrastructures: Pipes, platforms, and politics”. Media Culture & Society 41(3). https://www.researchgate.net/publication/329262815_Digital_media_infrastructures_pipes_platforms_and_politics

Duncan Riley. 2018. “Twitter destroys outside apps again by killing the API most of them use”. Silicon ANGLE. https://siliconangle.com/2018/08/16/twitter-destroys-third-party-apps-killing-api-use/

J.P. Simon. 2021. “APIs, the glue under the hood. Looking for the ‘API economy’”. Digital Policy, Regulation and Governance, 23(5), 489-508. https://doi.org/10.1108/DPRG-10-2020-0147

Ishveena Singh. 2018. “Insane, shocking, outrageous: Developers react to changes in Google Maps API”, Geoawesomeness. https://geoawesomeness.com/developers-up-in-arms-over-google-maps-api-insane-price-hike/

Lorenzino Vaccari, Monica Posada, Mark Boyd, Mattia Santoro. 2021. “APIs for EU Governments: A Landscape Analysis on Policy Instruments, Standards, Strategies and Best Practices”. Data 2021 6(6): 59. https://www.researchgate.net/publication/352213555_APIs_for_EU_Governments_A_Landscape_Analysis_on_Policy_Instruments_Standards_Strategies_and_Best_Practices

Erik Van der Vleuten. 2004. “Infrastructures and Societal Change. A View from the Large Technical Systems Field”. Technology Analysis & Strategic Management 16 (3): 395-414. https://doi.org/10.1080/0953732042000251160

Jan Van Dijk. 2020. “The Digital Divide.” Polity. UK.

Werner Vogels. 2016. “10 Lessons from 10 Years of Amazon Web Services”. All Things Distributed. https://www.allthingsdistributed.com/2016/03/10-lessons-from-10-years-of-aws.html

Maja Vukovic, Jim Laredo, Sriram Rajagopal. 2014. “API Terms and Conditions as a Service”. 2014 IEEE International Conference on Services Computing (SCC). Conference Paper. https://www.researchgate.net/publication/288485656_API_Terms_and_Conditions_as_a_Service

Youngjin Yoo, Ola Henfridsson, Kalle Lyytinen. 2010. “The New Organizing Logic of Digital Innovation: An Agenda for Information Systems Research”. Information Systems Research 21(4):724-735. https://www.researchgate.net/publication/220079642_The_New_Organizing_Logic_of_Digital_Innovation_An_Agenda_for_Information_Systems_Research

Jonatham Zittrain. 2008. “The Future of the Internet--And How to Stop It”. Yale University Press. https://dash.harvard.edu/bitstream/handle/1/4455262/Zittrain_Future%20of%20the%20Internet.pdf

Project Methodology

The APIToS project is part of the digital infrastructure research grant. It has been conducted since May 2021 and aims to better understand how API Terms of Service can be leveraged for the building and maintenance of more sustainable and open infrastructures.

Discover our website: **https://www.apitos.org/**

Phase 1: Research design

The first phase of the project was an exploratory qualitative phase based on:

- 9 interviews (av. 1 h) with different profiles (API specialists, API providers, API users in different sectors: research, transport, finance, etc.) mainly from France and USA
- Review of initiatives concerning ToS or API or legal design in general

Our aim was first to exchange with different people using, building or regulating API to apprehend better their understanding and interaction with API Terms of service and highlight the main issues they are facing and their opinion on a creative commons model. We also reviewed initiatives (sometimes suggested during the interview) at the crossroad with legal design and API/ToS. Our goal was to get an overview of some inspiring models and the way we should interact with them (integrate our prototype to other projects).

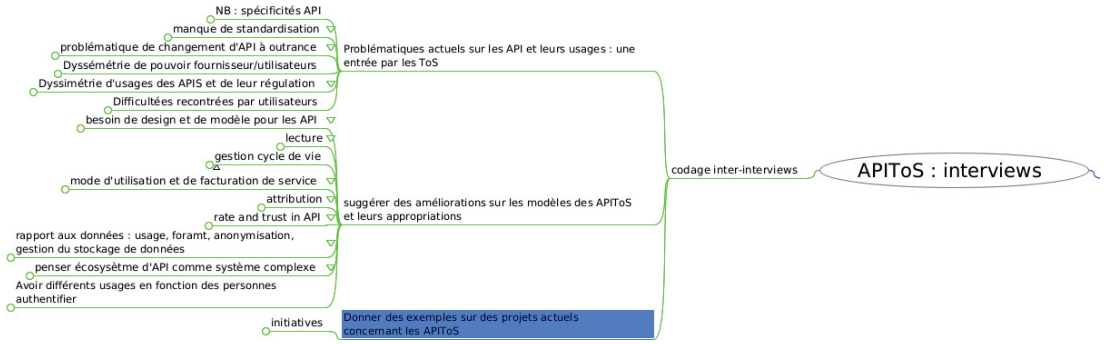
A cross-analysis from the different members of the research team with different backgrounds (law, STS, API specialist) was also conducted (based on the transcript of the interview, and legal analysis of ToS).

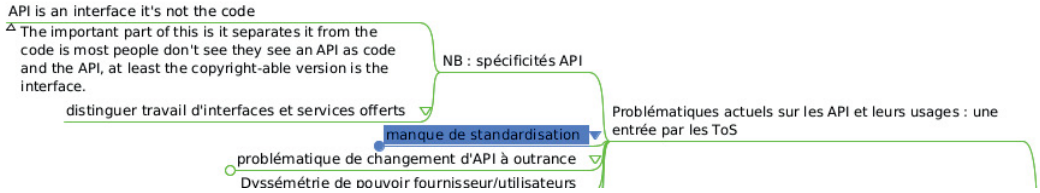
Interviews: sectors and activities represented

Interviews gave us a specific view of issues encountered by people using or developing API for their professional or militant activities. Despite the small number of interviews counterbalanced by in-depth analysis, we were able to explore various sectors and types of organizations (small or big companies, public administration, and institutions).

- Public transport: we had two interviews related to the transport sector, one with a company developing API for mobility services in France and one with an API expert who was working as a consultant for public transport in the USA.
- Bank activities were also explored with one interview.
- Academia and public research have also been one of our topics of interest. API is crucial nowadays for research to explore different databases with text and data mining methods and create a new one. Fair use of API is related to the open access/open science revendications. We had some highlights from researchers developing tools and platforms for academics using publishers API or social media API.
- Our exchange was also with regulators or API experts, working on the field of standardization of API or Terms of Services in Europe and North America.

Details about qualitative analysis of the interviews





Phase 2: Survey

Phase 2 of the study consisted of running a quantitative survey. Analysis of the survey responses, alongside findings from Phase 1 interviews, helped guide the development of a prototype qualitative framework for describing API Terms of Service in a way that is understandable and readable by users, easy to produce by providers, and favors open ecosystems for policymakers.

Phase 3: Workshop

To anchor our preliminary results and test our hypotheses with the API practitioner community, we organized an online workshop/focus group of 20+ people working with or studying APIs. Attendees had different backgrounds (from research, industry, legal design, legal studies, and policy development) and explored, assessed, discussed, and validated the prototypes and resources developed by the research team.

Our online workshop design was based on the qualitative and quantitative results of Phase 1 (interviews) and Phase 2 (survey results) of the project. Based on these results, we proposed to the participant:

The main question to answer collectively:

How to build trust in the use of APIs by clarifying the commitments of API providers and empowering users?

- The first prototype of an APIToS framework to foster trust between users and providers and API (Fair API Commitments for Trust (FAIR) framework).

- A set of resources to explore the use of API in diverse contexts and for different profiles.
- Some examples of legal design related to digital content and APIs.

Objective

- To have an idea of the solutions to implement with two phases during the workshop
  - **Phase 1:** Selection of the most relevant ideas based on the answers to the questionnaire and interview and reopening by the participants (divergence). Converge on what seems most relevant and prioritize 3 key ideas
  - **Phase 2:** See how to design these key ideas with at the end a proposal for a first prototype (Brainstorming of the possible design).

NB: we are using the double diamond model of design methods (divergence convergence X2)

Workshop Design

Introduction

- Round table discussion (15')

Phase 1 (1h)

- Introduction: presentation of the project and quality results + questionnaire (20')
- Miro table with key ideas (by theme): addition by the participants of the ideas which could be missing (20')
- We prioritize the ideas with a how/now/wow matrix (20')

Phase 2 (40')

- We take back the 3 main ideas and by group we work on the possible design ideas. (20')
- We prepare a synthesis for a restitution of each group. (20')

Conclusion (5')

Workshop organization, feedback and comments

Workshop organization

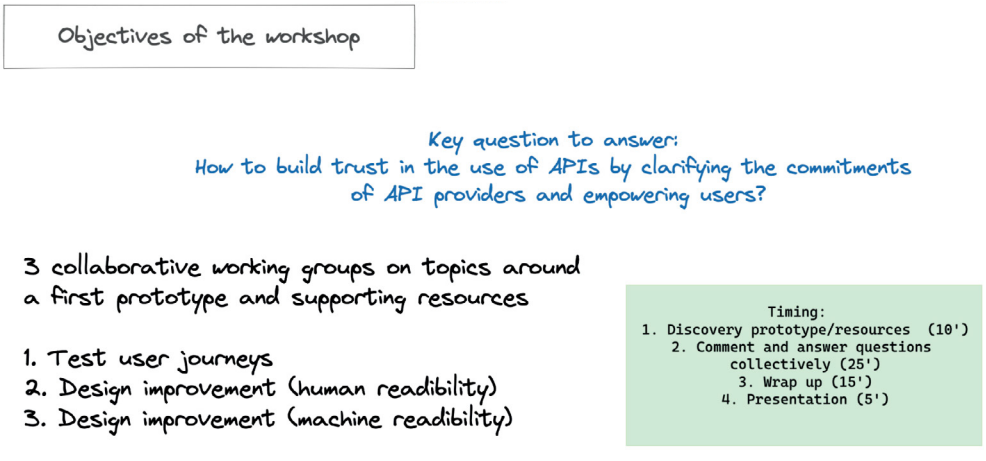
The workshop started with a presentation of the objectives built around one key question:

“How to build trust in the use of APIs by clarifying the commitments of API providers and empowering users?”

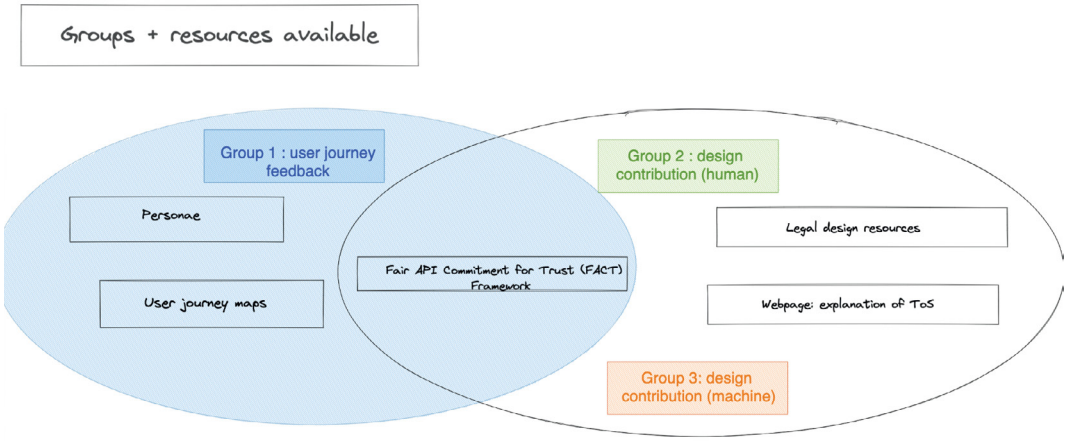
This question is based on the result of three main questions asked in the survey (Phase 1):

- **According to you, what would be the most important criteria to improve APIToS ?**
- **How would it be possible to enhance trust between consumers and suppliers?**
- **Among the following commitments, what is the level of importance of each clause to be included in the standardization approach of APIToS?**

In the workshop, to answer this question, we split the participants into three different groups, after exposing them to the pre-workshop material and phase 1 results. The goal was to analyze different user journeys (group 1), design improvements for human readability (group 2), and design improvements for machine readability (group 3).



Each group was directed to specific resources presented above (persona, user journeys,. legal design examples) and with a specific timing to work together.



Timing:

1. Discovery prototype/resources (10')
2. Comment and answer questions collectively (25')
3. Wrap up (15')
4. Presentation (5')

Examples of the slides presented to the workshop participants.

Annexes

Interviews frame

Hello, we are conducting a study about the impact and challenges of API “human and machine readable” Terms of Service (ToS) for a more open and transparent digital infrastructure. In that context, we are looking for practitioners who have been facing API consumption or API providing challenges regarding APIToS.

Part 1: APIToS Overview

General questions (owners, users, regulators)

What has been your relationship to API and API Terms of Service so far?

ToS Challenges

How would you define the challenge of API terms of service for you?

In your opinion, what is the Top10 APIToS ? Or what are the worst APIToS ?

What are the problems you encountered involving API Terms of Service?

How did you overcome these challenges?

ToS Criteria

What are the main criteria that you look in the API terms of service?

Have you ever changed your decision about using an API because of API Terms of Service?

According to you, what are the ones that are the most critical? Why?

Can you list them by order of importance?

In your opinion, how could these problems be resolved? Who must be involved to do what?

Specific questions for:

Regulators

Are you in contact with the main API owners? Have you already discussed about APIToS? What are the main trends concerning APIToS?

Do you think there are specific API ToS challenges that might require a regulation (ie regarding competition law). Do you know some interesting cases involving regulators & API ToS providers?

Alternatively, do you think that stakeholders can successfully address such a need (if yes, do you have any example or idea to share?)

Users

From your user's perspective, what APIToS challenges are you facing when using a different API? From both a legal and a technical point of view (including when you are exposing the results of such API to your own customers)?

Owners

ToS writing

Who is responsible for drafting the TOS in the company? How were they drafted? What has been the evolution in recent years? Are these elements known by the whole company? What repercussions in the different departments of your company and outside?

Are you updating them regularly, continually, rarely? Do you have specific processes for it?

Do you have a technical (full or semi) implementation of those ToS and, if so, who is in charge of them?

ToS Challenges

Can you describe the TOS, the different parts presented? The major elements? In your opinion, what are the main issues in the implementation of these TOS? How do they impact/follow the company's economic developments?

Interaction with users and other Stakeholders

How is the contact with users? A dedicated service? Is there a feedback of usage? Any critics? How do you update your TOS? Do you manage it with users?

How are your TOS inspired by other TOS of stakeholders? Do you have exchanges with them? Do you need to (co) adapt/adjust?

Do you have restrictions related to the international, European legal framework? What are the main concerns about legislative changes?

Part 2: APIToS-CC Model (future)

We are building a system where stakeholders can easily know the important clauses of an API Terms of the service contract, in a user-friendly way, a little bit like a creative commons model for APIToS.

Creative Commons models and other inspiring projects for APIToS

What do you know about Creative Commons?

What do you like the most about the initiative?

According to what we said, how would you see it applied to API ToS?

Do you know of any inspiring projects on this topic?

Implementation

How to implement this kind of model? With you? Which features will be useful (Machine-readable, automatization)?

Do you have specific needs to structure/organize APIToS CC model? What will be important criteria for a model? What are the challenges, main impact, or opportunities?

Relation with data

How this model could be related to open data, linked open data? How it could enhance these complementary movements?

Appropriation and use

What will it change with other stakeholders?

What are the limits? How to be sure it will be used?

Specific questions for:

Regulators

In your opinion, what is important to include in the project in order to help stakeholders to regulate themselves? What might be your place as a regulator? Do you think it might be interesting to articulate such a project with other regulatory tools you're working on?

Users

In your opinion, how can the project be successfully adopted by users, do you see any "nice to have" features in order to facilitate and help its adoption by users (legal design)?

Owners

What are your expectations regarding such standardized APIToS, what might be the "need to be '' part of the project in your opinion in order to foster its adoption (brand communication, technical specification corresponding to standardized APIToS, etc.)?

Part 3: Exploration of ideas and Proposal feedback

Presentation of a draft framework and asking from. To 10 how they are important Presentation of a draft framework and asking from. To 10 how they are important

- **SLA performance**
- **Reusability/License of the data**
- **Catchability**
- **API Copyright**
- **Attribution of source/Branding**
- **Revocation policy**
- **Breaking changes policy**
- **Payment terms**

Survey results

As of May 3rd, 2022

- 45 incomplete responses
- 26 complete responses

NB. Concerning the incomplete answers, the people who started answering the questionnaires had very weak interactions with API or not at all. (response by curiosity of the questionnaire?).

Profile and interaction with API

Profile

- The majority of respondents develop or use APIs. Several people (42%) are also involved in API management (legal, performance, business).
  - Comments on others profiles : API research or building API Sandboxes for API performance
  - 50% of the profiles have been working with APIs for 5 to 10 years, 30% for less than 5 years.

Interaction with APIs

- The majority of people have moderate to low reading with APIs (52%). Over 25% never read API change notifications and only 23% read them moderately.
- More than 60% rely only slightly to not at all on APIs to make decisions.
- A few respondents (30%) seek advice (largely or moderately) from others for APIs, while 39% do not seek any advice.
- 10% take part in writing APIs (actively), but most do not write APIs and do not rely on others to do so at all (53%).
- Interactions with API producers are mostly moderate (30%) or very weak (30%) or non-existent (8%)

NB : A few profiles 1 or 2 people stand out as subject specialists with an awareness of the subject, a regular look at the APIS and their modifications.

Organizations jobs

- The people who responded
  - are mainly from private sector or freelance work in the main IT sectors 58% or from B2C or B2B, regulated sectors,
  - 18% come from public administration and few from the academic public.
- The majority of the respondents work for companies with 1 to 49 employees or in very large companies (more than 50% of the cases) with 500 to 1000+ employees.
- The respondents come from France (26%). The other nationalities represented are South Africa, Poland, UK, USA, Luxembourg, Belgium, Brazil, Germany

Personae

API Provider Personae	API Consumer Personae	API Ecosystem Personae
API product manager for business	Developer consuming APIs	Policymaker
Platform business owner	Product manager consuming APIs	Open source/Copyleft advocate
API Provider for public sector/research		

API Consumer Personae

API Provider Personae

GRACE HOPPER
API Product Manager

Profession API Product manager Age 35
Country USA
Knowledge background with API

Story with API (FEEL/DO)

Grace is API PM at a large corporation. She knows well how to run large corp projects work and had a technical background. She believes in open models and platform models and had decided to become an API PM to apply it at scale

Interactions with API providers, legal team, etc.

She consults what other API providers are doing and talk with internal stakeholders to align the API with company business model

API uses (what kind of API, frequence, name, etc.) (DO)

She is responsible of different APIs on the company portal, as much about the design , development, security. She talks to business analysts about pricing plans and customeracquisition to develop the API Business model.

Motivations/issues with API (THINK)

The goal with API is to maximise the developer activation rate and the l lifetime value of API consumers

APITOS uses (reading, writing, etc.) (DO)

She is involved in the APITOS discussion on the Developer onboarding process and gives back feedbacks to the platformteam and legal team about users

Travis Kalanick
Platform Business Owner

Profession Age 41
Country : Brazil
Knowledge background with API : mild

Story with API (FEEL/DO)

Travis has built a successful business with a famous ride sharing application. During all the development process, he always asked himself the "Build versus Buy" decision about using a 3rd party service for a feature, or building its own. For travis, speed and agility for execution is the more important but he wants to take every decision for the longer term. Business continuity is key to sustain hyper growth

Interactions with API providers, legal team, etc.

When developers have chosen to work with an 3rd party service delivered via an API, Travis always reach customer support and sales team to negotiate better Terms of service, or enterprise ones

API uses (what kind of API, frequence, name, etc.) (DO)

Travis is not a developer but knows what is an API and understand that the interface is as much important as the organization behind it. Travis has been involved in every critical 3rd party service and API adoption for the company.

Motivations/issues with API (THINK)

For Travis, APIs are a great way to build on top of others have done and scale, and he wants to use the same strategy for its platform opening an API too.

APITOS uses (reading, writing, etc.) (DO)

Travis is the only one reading carefully API ToS of service when going in full production to secure the reliability of the business for his customers. He makes always a final decision bases on contract policy,

Sat Nakamoto
API Provider for Research

Profession : Researcher Age 35
Country Japan
Knowledge background with API : low

Story with API (FEEL/DO)

Sat is a researcher working in the public sector with data sets from open data sources and from his research. He knows how to program and how to put a server online using most common tools and platforms and infrastructure.
Sat wants to put his research online and get contributions from others

Interactions with API providers, legal team, etc.

Sat is mostly working with other researchers of his lab and has few interactions with other departments.

API uses (what kind of API, frequence, name, etc.) (DO)

He is familiar with he concept of API as the technical interface to access ressources. He consumes API from others

Motivations/issues with API (THINK)

Sat wants his research and data to be useful to the world and that the supply chain of knowledge attribution is respected

APITOS uses (reading, writing, etc.) (DO)

Sat is not interested a lot by legal stuff, and is not capable of writing contract. For him, a small text disclaimer in a README repository is already good enough

API Ecosystem Personae

URSULA VAN DER LEYEN
Policy Maker

Profession : Policy maker Age 55
Country : Sweded
Knowledge background with API : low

Story with API (FEEL/DO)

She believes in a market economy as long as there is fair competition and economic progress for citizens and consumers. She sees the impact of technology and the digital economy in people lives and considers that platform monopolies are bad for the economy and for citizen rights

Interactions with API providers, legal team, etc.

She interviews experts
on platform business models,
expert lawyers
and policy makers

API uses (what kind of API, frequence, name, etc.) (DO)

She has no technical background, she sees APIs as a way to govern platform policies

Motivations/issues with API (THINK)

She understands that open API can open market and kill rent to the benefit of users,
to the condition
there is less value capture

APITOS uses (reading, writing, etc.) (DO)

She understand API are part of platform policies.

RICHARD STALLMAN
Open source/
Open models advocate

Profession : Open advocate Age : 65
Country : USA
Knowledge background with API : high

Story with API (FEEL/DO)

Believes APIs copyright issues and API SaaS models are a concern for governing open ecosystems. Consume many OSS projects and believe in respecting interface contract to keep interoperability between software systems.
Is able to decide to use a software component or not depending of its licence or governance model. Has worked in the Open data movement and has been an advocate in developer communities about open models

Interactions with API providers, legal team, etc.

He will contact directly API owners to give feedbacks or ask for a different license, non profit, academic purpose, for good etc

API uses (what kind of API, frequence, name, etc.) (DO)

Consume mostly open source software self hosted or hosted by 3rd parties.
The interaction with APIs is mostly technical to make software work together, except in some cases where he uses Hosted services and their API.
He checks at least if there is a free tier or enough alternatives in case of issue

Motivations/issues with API (THINK)

For him APIs are a interface to increase interoperability between open systems.
They are the best security against vendor locking and open ecosystems of Open applications

APITOS uses (reading, writing, etc.) (DO)

He will read terms of service, especially about the reusability rights and the licence of the data.

FACT License

Your API Terms are Fair, Transparent and Trustworthy. It's a FACT.

0 Intro

0.1 Abstract

The FACT (for FAIR API Commitment to Trust) license is a customizable Contract that expands upon traditional API Terms of Service (ToS). Its purpose is to promote API usage and allow API Users to develop interoperable products, whether they be new APIs or other types of products.

Using a standardized set of contracts that are widely accepted as safe by the community, the **Core API Provider** and **API Users** can reduce transaction costs. The license also aims to raise **the users' awareness of key licensing issues** that the parties need to address. Furthermore, it **helps companies to define a clear, unambiguous and legally-binding framework** that encourages the development of products interoperable with their technologies. This can lead to the creation of high-value products and services in a healthy, trustworthy and safe digital infrastructure ecosystem.

The FACT Contract is drafted neutrally to cover all rights related to API usage including Intellectual Property rights such as patents, copyrights and trade secrets, regardless of their validity. The standard terms of the FACT license, along with all other specific binding documents, listed in the "API contractual commitments summary", make up the entire contract.

0.2 Legal notice & Disclaimer

As a result of our research, the FACT license is distributed under a Creative Commons Attribution 4.0 International Public Contract (CC BY 4.0) [1]. This original Contract is a pilot based on the INTILA (Interoperability Information License Agreement), and will evolve in response to feedback and usage.

We are not a law firm and do not offer legal services or advice. The Distribution of this Contract does not create a lawyer-client or any other kind of relationship. We provide this Contract and related information on an "as-is" basis, without warranties regarding the Contract, any material licensed under its terms and conditions, or any related information. We disclaim all liability for damages resulting from their use to the fullest extent possible.

[1] See : https://op.europa.eu/en/publication-detail/-/publication/5adbafd8-2498-4e57-9f11-f7d8026aa88b/language-en.

1 Preamble

1.1 Origin of the project

FACT is a result of APIToS-CC research project funded by the Critical Digital Infrastructure Grant from the Ford Foundation, Sloan Foundation, Mozilla, Omidyar, and Open Society Foundations[2]. The project aimed **to explore the best framework for API terms of Service. As a contract, that can be automatically read, controlled and enforced** between digital infrastructure and applications to improve the trustworthiness and sustainability of digital infrastructures. This Contract is the original version that is both human-readable and easy to automate. The goal is to allow each API provider to auto-generate a specific version of the Contract complemented with its additional terms, and to make the entire set machine-readable.

1.2 Our goals

Web APIs play a vital role in the development of our digital society, placing increasing responsibilities on API providers. They have the power and responsibility to ensure fair, valuable, and socially ethical use of their APIs, which is essential for the sustainable and ethical growth of their ecosystem.

We aim to promote a fair offer and use of API ToS by developing and making available to all users a common agreement that ensures sustainable trust with providers.

This Contract which we suggest calling FACT (acronym for “FAIR API Commitment to Trust”) should allow Core API Providers, in both the private and public sectors, as well as API Users (primarily businesses although end users may directly benefit) to reduce mistrust and ambiguity by relying on a set of commonly accepted contract terms which are considered safe by a large community.
It should also help companies define a clear, unambiguous and legally consolidated framework that encourages the use of APIs and the development of products or services interoperable with such APIs.

1.3 Mutual commitment and full agreement

This Contract outlines and legally enforces the main commitments voluntarily agreed to by the Core API Providers or by the API Users.

The contractual relationship between an API provider and an API user can be based on a multitude of heterogeneous contractual documents, which undermines secure collaboration. Although this Contract does not provide one single solution, but **a set of good practices (standard terms with different options) and greater transparency by requiring the Core API Provider** to bring together all other enforceable documents (such as private contracts, trademark over the name, or privacy rights / data protection rights over information in the contents) into a single “API contractual commitments summary” (see Annex A).

The API Core Provider should use the FACT License with other agreements (such as the Open Data License or any other license). In this case, the Core API Provider should describe in the Appendix (Annex A) the rights govern what contents together in the individual record or in some other way that clarifies what rights apply. This Contract, i.e. the FACT and the Annexes, constitutes the entire agreement on all matters concerning the API User’s right to use the API and to exploit **API Users’ Product** or **API Users’ Services**. In case of contractual

[2] APIToS-CC project: https://opencollective.com/di-grants/projects/apitos-cc

conflicts, the FACT core commitments of the Contract should prevail on any other of these other contractual documents listed in the “API contractual commitment summary” (see Annex A).

This Contract may be sufficient on its own as a minimum set of requirements in a situation where access is open to all, without financial compensation. Used independent permits as described below: access to the API by anyone, grants a free license for the API specification, allows for free reuse of the data only within the context of using the API (by either the API Users or their end-users), does not impose limitation on outputs, and requires minimum attributions.

2 Definition

Particular attention has been paid to using a wording that remains accessible and easy to understand, notably for non-lawyers. Nonetheless, the document uses a specific legal vocabulary that must be referred to in this type of document.

- **“Core API Provider”** means the Provider of the API licensed under this Contract.

- **“API User”** means any individual or legal entity exercising rights under this Contract, API Users irrevocably accept this Contract and all of its terms and conditions by exercising any rights granted by the Contract such as the use of the API, the Development or Exploitation of API Users’ Products or Services.

- **“API Specification”** means the documentation containing the technical description of the API provided by the Core API Provider under this Contract.

- **“API Users’ Product”** means any data set, computer file, computer program, or any electronic hardware device, or any combination of these, which implements the API Specification to communicate with the API, or which creates, writes, reads or displays files in accordance with the API Specification and in a way that would infringe on the licensed Intellectual Property if the API Users did not benefit from the license grant provided in this Contract under Article 3.2.

- **“API Users’ Services”** means services offered by the API Users to third parties, including End Users, using the API Users’ Products, but without Distributing the API Users’ Products. API Users’ Services include providing remote access to third parties to the functionalities of the API Users’ Product over a network without enabling such third parties to make or receive copies or units of the API Users’ Product).

- **“End User”** means a third-party customer (individual or entity) to which a copy of, or access to, an API Users’ Product or a service based thereon is Distributed or otherwise provided for this third-party customer’s own purposes and not for sublicense or further distribution to others unless expressly authorised by the Contract.

- **“Intellectual Property”** means any and all intellectual property rights related to the use of the API, including all the following and all rights in, arising out of, or associated therewith: (i) procedures, designs, inventions, and discoveries; (ii) works of authorship, copyrights, and other rights in works of authorship; and (iii) know-how, show-how and trade secrets on a worldwide basis, including all Patents issued or issuable thereon, but excluding all trademarks, trade names, or other forms of corporate or product identification. “Patents” means all classes or types of patents (including, without limitation, originals, divisions, continuations, continuations-in-part, extensions, re-examinations and reissues, as well as any patent-like intellectual

property rights protecting inventions such as Utility Patents), and applications for these classes or types of patent rights in all countries of the world that are owned or controlled by the Core API Provider during the term of this Contract.

- **"Distribute or Distribution"** means licensing, distributing, offering for sale, selling, leasing, providing online access to, communicating, exporting, importing or otherwise making available the API Users' Products in any manner to a third party – referred to as the End Users – so that such third party is provided with the API Users' Product.

- **"Database"** means a collection of content (information, images, audiovisual material, and sounds all in the same database, for example) arranged in a systematic or methodical way and individually accessible through the API thanks to the Contract. The use of the Database and the data itself might be governed by specific licenses.

- **"Develop"** means to design, develop, make or have made, and/or produce or have produced API Users' Products in accordance with the API Specification.

3 Core API Provider's Commitments

3.1 Neutrality

The Core API Provider understands the importance of providing a neutral infrastructure and commits to being entirely neutral in delivering its services. The Core API Provider will avoid any behaviour that could favour or disadvantage some users (whether organisations or individuals) or fields based on non-objective criteria.

Although FACT encourages an Open API approach, there may be some objective restrictions that the Core API Provider needs to impose:

- **Developer access rights.** Technical access conditions for developers (identification keys in particular), when present, are detailed in Annexe A (link to user or developer ToS and dedicated documentation).

3.1.1 Restrictive rights (option)

- **Actors and fields access rights:** If there are objective criteria that discriminate against certain users or fields, such as research actors, the precise conditions are listed in Annexe A (actors and fields restriction).

3.1.2 ShareAlike license on API Access (option)

- **API Access - Share Alike:** The API User is required to maintain the FACT Contract for any API Users' Products or API Users' Services provided to their End Users, as described in Article 4.4.

The Core API Provider agrees not to impose, any, other restrictions on the access and use of its API.

3.2 API Access and grant of rights

The API is available for free and accessible as long as the conditions mentioned in article 4 (API Users' Commitments) are respected. These conditions include the development of third party internal or external applications, the development, and publication of production products that consume the API internally or externally, and the distribution and provision of access to the API to End Users. The following terms describe under what conditions the API is provided, how the API Users are expected to use the API as well as the API Users' rights and obligations when using the API-content in their own products (API Users' Product) or services (API Users' Services). The goal of this Contract is to grant the API User rights of use, that would otherwise infringe upon the exclusive rights of the Core API Providers. It is not, and should not serve as a commercial service agreement (such Agreement, is covered in Annexe A if applicable).

Subject to all this provisions and in terms of API Users' compliance, the Core API Provider grants to the API Users an access and a license on the Core API Provider's Intellectual Property (see duration in Article 10, worldwide, non-exclusive, non-transferable and personal license) only for the following purposes:

- (i) **to Develop API Users' Products** (to "Develop" meaning – in this Article and elsewhere in this Contract – to design, develop, make or have made, and/or produce or have produced API Users' Products in accordance with the API Specification);

- (ii) and **to Exploit API Users' Products** in one or more of the way(s) defined as follows (hereunder collectively referred to as the "Exploitation" of the API Users' Products or to "Exploit" API Users' Products):

    - a) to Internally Use API Users' Products; (to "Internally Use" meaning – in this Article and elsewhere in this Contract – to install, use and deploy the API Users' Products and to make them available to the API Users' personnel, for use internally by the API Users for the purpose of general business practices but not to offer API Users' Services or for Distribution purposes;

    - b) and to Offer API Users' Services to End Users;

    - c) and to Distribute API Users' Products to End Users;

3.3 API Technical Specification Access

3.3.1 Continuous access (Source of truth for Interoperability)

If the Core API Provider uses an official API Specification document, the Core API providers commits to using the API Specification as the sole source of truth for tracking interoperability changes on the API.

If not, the Core API Provider will **provide the API Users with access to the complete Technical Specification** via its online site or other reasonable method determined by the Core API Provider from time to time as described in Annex A. The Core API provider to using this source as the source of truth for development for the tracking interoperability changes on the API.

3.3.2 Modifications

If and when the Core API Provider makes corrections and modifications to the API, the Core API Provider will provide an updated Technical Specification (new version of the Technical Specification), by posting it in the Core API Provider online site or communicating through other means to notify API Users of the adoption of the updated version.

The Core API Provider **will notify API Users of any significant change to its API at least three months before implementing them, allowing** API Users to update their implementation gradually. **Past versions of the API will remain accessible for at least one year** from the date of publication of the API Specification change.

3.3.3 Specification Contract

The Core API Provider waives and/or agrees not to prevent or assert any right or authority that would prohibit from using the Specification to exercise the licensed Rights. Simply making and using Products or Services authorized by this Section 3.3 will be permitted under this Agreement.

3.3.4 CC Zero License on specification (by default)

The Core API Provider applies a **CC Zero License to the Specification** (see **https://creativecommons.org/publicdomain/zero/1.0/legalcode** for the full text of the license).

3.3.5 Specification Contract

The Core API Provider could apply a Creative Commons Attribution-ShareAlike 4.0 International License to the API Specification (see **https://creativecommons.org/licenses/by-sa/4.0/legalcode for the full text of the License**).

3.4 Loyal change policy

3.4.2. Purpose

The Core API Provider retains the right to update and modify the API, API-content and this Contract in accordance with its Change Policy. If such a policy exists, **it outlines the types of changes that may occur, how they are communicated to Users and the notice period given.** This policy is included in the API contractual commitment summary, as described in Annex A).

However, the Core API Provider is dedicated to ensuring that its Change Policy is fair, that any changes are transparent and protect the interests of API Users. Therefore:

3.4.2. Technical modifications of the API

The Core API Provider commits to:

- **Notify the API Users at least 3 months** in advance (or more) of any change to the API or major version updates to the technical interface;

- **Publish all minor versions** (adding a number) for all non-breaking changes to the API interface;
- **Announce the API Deprecation period at least 3 months** in advance (or more) and maintain it for at least 12 months before sunsetting.

In exceptional cases, the Core API Provider may modify the API to solve security or continuous access issues. In these situations, it will make sure to maintain optimal API compatibility and inform API Users as soon as possible.

3.4.2. Contractual modifications

The Core API Provider commits to documenting all updates to the **API ToS** and providing direct communication to API Users for any major change at least 3 months in advance.

The Core API provider also commits to justifying and providing notice period of at least 3 months before **revoking of API access, except in cases where** the current use is deemed unlawful where an exceptional and justified explanation will be given, and evocation may be immediate. The Core API Provider will provide a FAIR period of 3 months to the API consumer to prove that its use was not against API ToS, End User License Agreement, or platform policy and regain API access.

The Core API Provider will also respect a minimum 3 months notice period before making any change to the **documents listed on the "API contractual commitments summary" (Annex A)** (this includes Service Level Agreement, pricing, Restriction on Data Reuse applications, etc.).

All modifications to this Contract must be properly documented by releasing a new version of these terms. The Core API Provider will keep a record of all API changes in an API Changelog, which will be published, versioned and maintain a history. In case it is not possible to maintain an API Changelog, the Core API Provider will assign a version number to each new version, set out at the end of the document, and ensure that previous versions remain publicly available.

By continuing to use the API after a change has been executed, API Users confirm their acceptance of the updated terms. However, if API Users do not accept a change, they must discontinue their connection to the API in accordance with the section "Term and Termination" section below, and will no longer be entitled to use the API.

3.5 Ethical Data policy

3.5.1 Data Policy

3.5.1.1 GPDR Policy

The General Data Protection Regulation (GDPR) sets out detailed requirements for companies and organisations on collecting, storing and managing personal data. It applies to both European organisations that process personal data of individuals in the EU, and organisations outside the EU that target people living in the EU. The Core API Provider ensures it is compliant with the GDPR regulation.

Unless specifically stated, **the Core API Provider does not collect or manage any personal data in the sense of the GDPR.** However, i any personal data is accessible via its API, the Core API provider guarantees that the data is compliant with Data protection requirements such as Purpose, Legal basis, Data Category, Data retention,

recipients, and data Localization. In such case, the Core API Provider will add a link to its GDPR policy on the API contractual commitment summary (Annex A).

3.5.1.2 Data license

To the fullest extent permitted by applicable law, but not in violation of it the Core API Provider hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of its Copyright and Related Rights, as well as associated claims and causes of action, whether presently known or unknown (including current and future claims and causes of action), in the data provided through its API in all territories worldwide for the maximum duration allowed by applicable law or treaty (including future time extensions), in any current or future medium and for any number of copies, and for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes. The Core API Provider makes this waiver for the benefit of each User API and their Final Users, with the full intention that such waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the peaceful enjoyment of the data.

3.5.1.2.1 Open Data license (option)

The Core API Provider should combine the Contract together with another license for the Databases and their contents. In case of multiple sets of different rights, the Core API Provider must specify which rights apply to which contents on the API contractual commitment summary (Annex A).

3.5.1.2.1 Commercial data (option)

The reuse of the Database and its contents is subject to compliance with a dedicated commercial license, which is reproduced in Annex A. Therefore, API Users must comply with this commercial license before any Distribution of the database or its contents, whether directly or through the Exploitation of API Users' Products.

3.6 Loyal Output Policy

The Core API Provider does not make any claims regarding API Users' Products (subject to the conditions of use of the database as specified above) as long as the API User complies with the commitments set out in Article 4 of the contract.

The User API must refrain from engaging in any behaviour that is directly competitive with the Core API Provider, as mentioned in Article 4.4.

4 API Users' Commitments

4.1 Strict implementation & implementation of the last Technical Specification

The API user agrees to always use the latest version of the API whenever possible, and to update their API implementation as much as possible to align with the latest version, as long as it is adding capabilities and not breaking the API, and the Core API provider has given sufficient notice. The API User also agrees to stay informed and attentive to communication from the Core API Provider, and acknowledges an interface contract

may changeover time. If the API Specifications are updated in accordance with Article 3, the API Users will make a reasonable effort to modify their API Users' Products to comply with the updated Specifications within six (6) months of the update. If a specification is open and public, such as in an Open API document for instance, the API User agrees to use that Specification as the primary source of truth for their development.

4.2 FAIR use

The API User agrees to take responsibility for ensuring that their API Users' Products:

- Comply with the API Specification and only connect to the API as instructed by Core API Provide;
- Meet the highest standards of quality and integrity for similar products;
- Do not violate any laws;
- Are developed and exploited in compliance with safety laws, regulations, and applicable agency approvals or prior authorization.

API Users agrees to not use any technical means to gain unauthorized access to, disturb or deactivate the API provided by the Core API Provider. This includes, but is not limited to, refraining from viruses, worms, Trojan horses or other forms of malware in the API or on the website where the API is provided.

Furthermore, the API User agrees to indemnify, defend and hold Core API Provider harmless from any and all claims, damages, losses, liabilities, costs, and expenses (including reasonable attorneys and counsellors' fees) arising from the development and exploitation of API Users' Products. This indemnification clause maybe restricted by adding that it is directly linked to a faulty implementation of the Interoperability Information stated in the technical Specification.

4.3 API Attribution – References to the Core API Provider

4.3.1 Attribution requirements

When using the API content in products or services developed by the API User is obligated to provide:

- **Appropriate Attribution** to the Core API Provider by prominently and conveniently displaying a notice that clearly indicates that the API User's product or service is based or contains data from the API. The API User's right to use the Core API Provider's trade name and Trademark is limited to source references in accordance with this clause, unless otherwise specified in a specific Trademark Policy provided by the Core API Provider (see Annex A: API contractual commitments summary). The API User is not entitled to use the Core API Provider's trade name or trademarks in any other way.
- **A clear statement that the work does not come with any warranties,** except to the extent that warranties are specifically provided by the Core API Provider as mentioned in Annex A, and provide instructions on how to view a copy of this Contract. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list shall meet this criterion.

4.3.2 No attribution (option)

The Contract does not grant permission to use the trade names, trademarks, service marks, or names of the Core API Provider, **except as required for reasonable and customary use in describing the origin of the API.** The API User can't use them to endorse or promote User Products or User Services based on the API without express prior permission of the Core API Provider.

4.3.3 Trademark enforcement (option)

The API User must comply with any additional extra attribution requirements beyond the standard attribution described above, such requirements shall be specified in Annex A. Any reference to the Core API Provider shall comply with the Reference and Trademark Use Policy as outlined in Annex A. Any reference to the use of the API shall also comply with the Reference and Trademark Use Policy as set forth in Annex A.

4.4 Non-direct competition [Loyal Output Policy +]

The API User may use the data obtained from the API only in combination with their own products or services. The presentation of the data obtained from the API must depend wholly on the API User's offer to one or multiple third parties.The API User may not use the data obtained from the API to create a product or service that directly competes with the Core API Provider's products or services. The API User may not sell the services provided to them by the Core API Provider, neither as such nor through redistribution or reselling.

4.5 Share alike [API Access +]

To ensure compliance with this Contract, the API User must apply the FACT license (this version or later) to their own API product or API service. The annexes must be tailored to their own contractual environment.
In this case, the API User must include the text of, or the URI or hyperlink to, the FACT license on their website and all API User's Products made. The API User may satisfy this condition in any reasonable way based on the medium, means, and context of his own Exploitation. The User API cannot offer or impose any additional conditions or different terms on their API User's Products or API User's Services that restrict exercise of the rights granted under the terms of this license.

5 Liability and Warranties

5.1 Limitation on Liability

Except in cases of wilful misconduct, gross negligence or damages directly caused to natural persons, neither party shall be liable for any direct or indirect, special, incidental or consequential, material or moral, damages of any kind, arising out of this Contract, or the use of the Interoperability Information, Technical Specification, Development, and Exploitation of API Users' Product by the API Users or any third party, either under a theory of contract, tort (including negligence), product liability or otherwise. This includes, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, loss of data or any commercial damage, even if the Core API Provider has been advised of the possibility of such damage.

The API Use acknowledges that any Services provided to End Users using the API are solely the responsibility of the API User The Core API Provider shall be held accountable, directly or indirectly, for any User Services based on the API. The API User agrees to assume all responsibilities in relation to End Users, and shall not hold the Core API Provider liable for any potential liabilities arising out of such services.

If the API User Exploits the API, the API User recognizes and agrees to indemnify the Core API Provider against any losses, damages, or costs arising from claims, lawsuits, and other legal actions brought by a third party.

5.2 Jurisdiction

In the event of any dispute arising out of or related to this User Agreement that cannot be resolved through mutual agreement, the parties agree to submit to the exclusive jurisdiction of the courts located in [insert jurisdiction].

5.3 Applicable Law

This Contract and any dispute or claim arising out of or in connection with (including non-contractual disputes or claims) shall be governed by and construed in accordance with the laws of **[complete with jurisdiction]**, except for conflict of law provisions.

5.4 Disclaimer of warranty

The Core API Provider provides the Interoperability Information on an “as-is” basis without any warranties (express or implied), including but not limited to the implied warranties of merchantability, fitness for a particular purpose, absence of defects or errors, accuracy.

However, the Core API Provider does warrant that the Intellectual Property is owned by or licensed to the Core API Provider and that the Core API Provider has the power and authority to grant the Contract.

The API User, its employees, agents, or distributors do not have the right to make any representation, warranty, or promise with respect to the Interoperability Information. **This disclaimer of warranty is an essential part of the Contract and a condition of the license grant.**

This Contract does not entitle the API Users to any support for the Interoperability Information, unless the API Users makes separate arrangements with the Core API Provider and pays all fees associated with such support.

5.5 Term & Termination of the Contract

The license grant term is a period of one year, starting as of the Effective Date. At the end of the term, the Contract shall be tacitly renewed for a period of the same length, unless notice of the termination is given by one of the parties at least 6 months before the date of such termination. In case of acquisition, the core API providers engage to guarantee continuity of the FACT to the acquiring company, by contractual engagement.

The license grant will terminate automatically upon any API User’s breach of this Contract or, in accordance with our Change Policy as mentioned in Annex A, permanently or temporarily discontinue the API or the API Users’ Products in part or in its entirety.

5.6 Validity

All rights and restrictions may be exercised and be applicable only to the extent that they not violate the applicable law or limited to the extent necessary to render this Contract legal, valid and enforceable.

If and to the extent that any obligations, or limitation of the rights in accordance with this Contract, are not valid because of User API rights as a consumer, they shall not be applicable. Accordingly, nothing in this Contract is intended to exclude or limit the User API rights as a consumer.

If any provision of the Contract is invalid or unenforceable under the applicable law, this will not affect the validity or enforceability of the Contract as a whole. Such provision will be construed and/or reformed so as necessary to make it valid and enforceable.

This Contract is personal to the API User and is not assignable in whole or in part by the API User without the prior written consent of the Core API Provider.

6 Annex

6.1 Annexe A: API contractual commitments summary

The Core API Provider list is required to provide a comprehensive list of all the contractual commitments that may be imposed on an API User in this specific context, as described within the Contract. In addition, other links may be included to ensure the API User have access to complete information (such as scripta manent, TOSDR, etc.). The API User s obligated to review and comply with all of these documents.

Title	URL	Short description

6.2 Annexe B: How to Apply These Terms to Your product?

If you are using FACT for your API product, it is mandatory to include a link to this version of FACT on your website, typically as a link called yourapidomain.com/legal/fact.

For any other use, you should include a FACT-License.md file that references the entire Contract along with your product.

Additionally, you should provide information on how to contact you via email or any other means.