



Introduction

In 2024, Google will be the final browser to completely eliminate cookies from the internet. This will fundamentally change how business(s) identify their customers online and personalize their experiences.

Having specialized in top-of-funnel technology for over 5+ years, Whym proposes a solution to aid brands, advertisers & data companies in a compliant, transparent way through the use of a new API we call 'Doppel'.

RYAN HORNBERGER
RHENEE BARTLETT
KELLY NYLAND

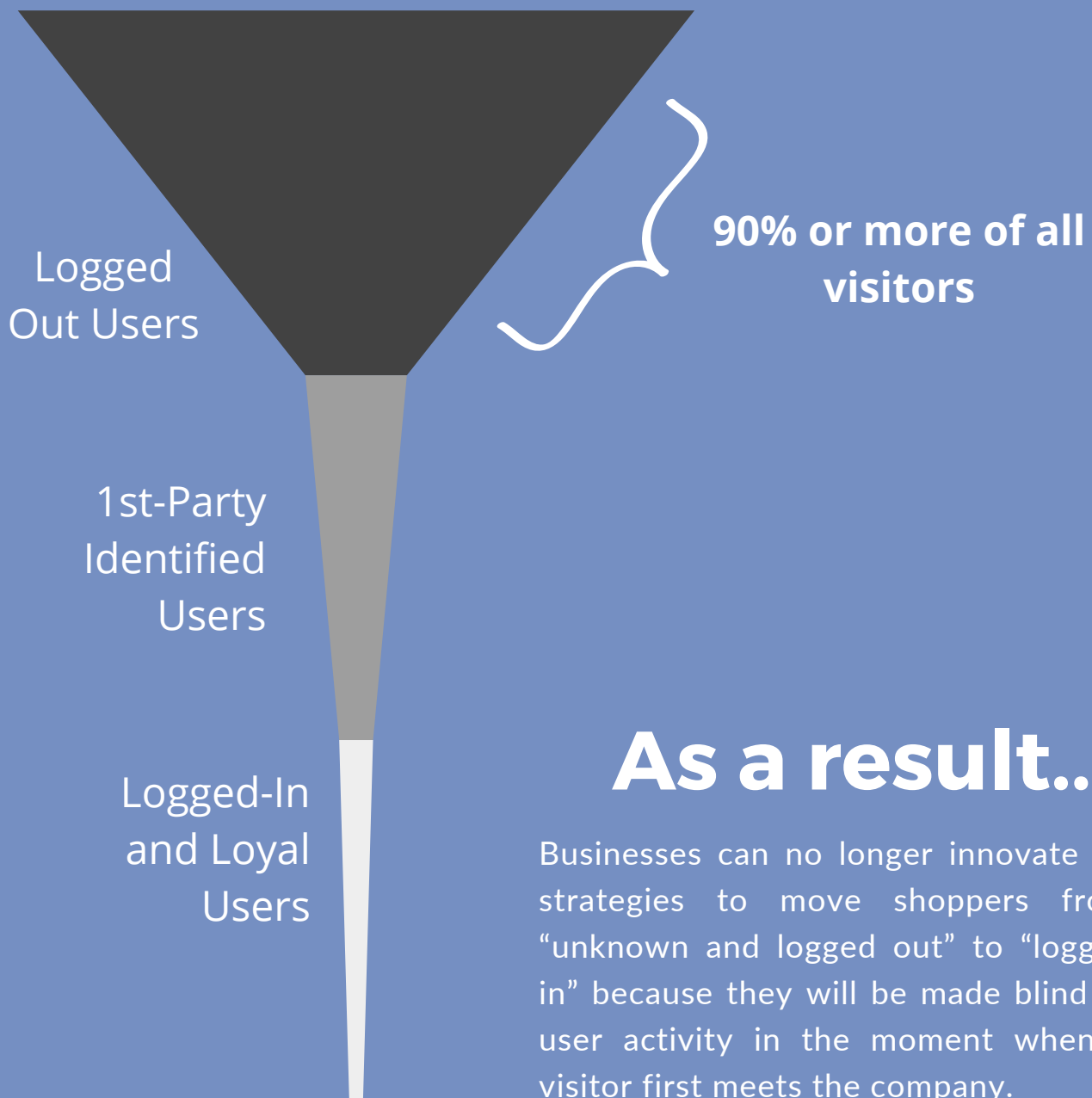
In 2024, every browser will be a 'privacy browser', and that's not good.

In 2024 Google Chrome will become the last major web browser to rebrand itself as a "privacy browser", and in this moment the internet will become, ironically, less private and less innovative.



Companies will lose visibility to top-of-funnel visitors.

Due to 2024 browser changes, the primary business model for ad-funded content destinations, cross-domain shopper retargeting solutions, and “guest” checkout opportunities will die.



We propose a solution to replace internet cookies for the foreseeable future.

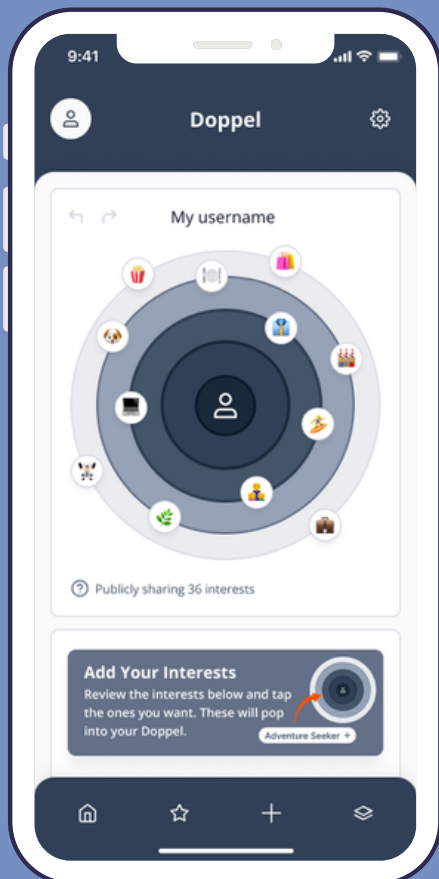


Regulators and Browsers now block a critical web technology (cookies), making online marketing far less effective and logged-out user experiences cumbersome.

http://



Meanwhile, users do not want a world where they need to log in to every site the visit just to browse content online.



Our technology allows companies to get the data they need in a regulation-compliant way through a new intentional and 'always-on' online identity protocol called 'Doppel'.

To prove how easy and effortless the application of this technology can be, we demonstrate it's use through our user-friendly shopping app and browser extensions.

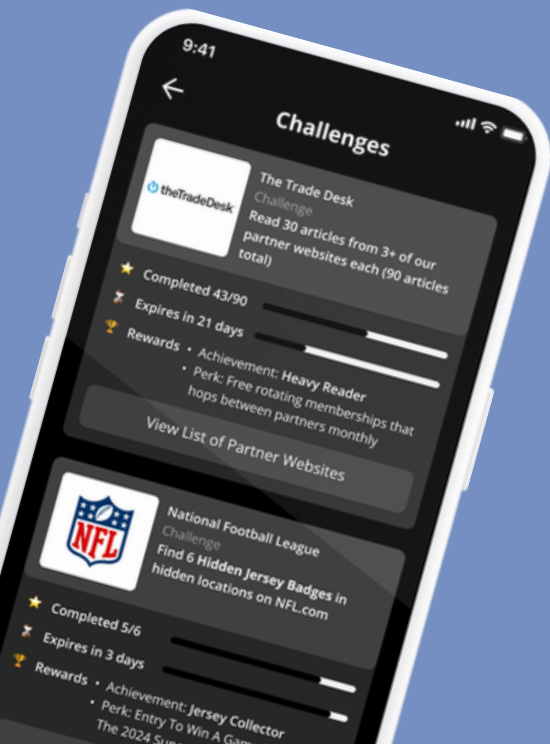
whym.

Share your email less often with Doppel

Read for free with Doppel

**Exclusive deals when you
use Doppel**

*No more Cookie Popups
with Doppel*



**Soon with
Doppel Challenges
& Achievements**

whym.

Doppel

Digital identity, after cookies.

PATENT PENDING

CONFIDENTIAL DRAFT
PROPRIETARY SPECIFICATIONS
COPYRIGHT 2023

RYAN HORNBERGER
RHENEE BARTLETT
KELLY NYLAND

The free, logged-out internet is dying.

Over the past decade visiting an online publication, news article, blog, or similar destination would have immediately presented the visitor with the article they are looking for without limitation. Their price of admission was an advertisement somewhere on the page.

Today many major publications (and even small ones) demand that users log in before they can even read the first article on their site. The reason?

In the name of privacy, pushed by regulators, browsers killed the free internet.

In an effort to make the internet more private, browser companies have made it impossible for advertising companies to decide which ad to show visitors because they cannot identify the visitors on a site (or even how many visitors are on the site at a given time).

Google has offered a solution, but it makes Google the sole owner of your online identity; and advertising competitors to Google can't work with that.

In response, companies funded by advertising have decided they will force users to log-in so that they may know the user's identity. This done that they can advertise to the user. Some publications are moving to a subscription-first model to stay alive, but all destinations now need some way to identify their visitors.

The result:

All previously-free (ad funded) content providers will require users to give up their email or log in before they can access any content.

Advertising and content creators see login as the necessary, although undesirable, solution to an existential crisis. The big problem with this solution is that a significant percentage of visitors will not log in and conversion of these visitors is extremely difficult.

The outcome: the free internet, the logged-out internet, has no viable financial model.

That leaves the world with a login-required internet.

We think the internet can, and should do better.

A 'Doppel' is an online, logged-out, identity that users share - on purpose.

As a typical consumer:

Offline (IRL) when individuals go out into the world they do not wear a driver's license, passport, or other identity on their vest. Neither do they expect that every place they enter will require visitors to show ID. Yet every store they enter and every sales associate will make basic assumptions about them without asking a single question to service their needs.

The user achieves this magical outcome through simple, often subtle, but always intentional decisions in the way they dress themselves, groom themselves, and present themselves to others in public.

Conversely, **the internet of "privacy browsers" forces users to appear nameless, faceless and indistinguishable from every other guest; destroying any opportunity for personalization or an above average user experience.** The only option users have, is to log-in and reveal their complete identity on every site they visit. This is the equivalent of showing ID at every store they enter.

Here at Whym, we believe that the next 20 years of online experiences should not force users to log in or manage cookies with every visit. This is where 'Doppels' come in.

A 'Doppel' is like an Avatar, Profile or Character that visitors use online

No email needed. This is **not the user's true identity**. This is their 'Doppel'. Doppels can carry any appearance, any set of public attributes, and any profile data the user wants to present about themselves.



Don't worry, when the need arises, **Doppels don't prevent the user from revealing their true identity and logging in where they wish;** they simply allow the user to have a personalized experience online and **reserve their true identity until they have a good reason to share it.**

‘Doppels’ are transferable, shareable, and copyable.

Doppels can take users beyond simply presenting a public face in their online behavior, Doppels can give users digital identity super-powers:

- Users can build a Doppel, share it with a significant other, and allow their partner to see the internet through their eyes and with their interests in mind.
- Users can find a celebrity with excellent taste who is sharing their Doppel and explore the internet with that celebrity’s specialized perspective and treatment.
- Users can copy a Doppel from another user or their own library and take that Doppel on a new journey that gives it a unique flavor on the internet.

‘Doppels’ are an intentional exchange of value between users and advertisers, merchants, and content creators.

Users “project” a ‘Doppel’ online because they want to be treated differently than other people. Users don’t want to be faceless, nameless, and unidentified. In the same breath, users don’t want to show ID at every door they enter.

Doppels and Advertising. In exchange for sharing a Doppel intentionally, advertisers can give users personalized ads, and this can pay for the content users read online without needing users to log-in everywhere they go. Users get to read articles without logging in, advertisers get to give users ads they actually care about, and content creators get paid to keep making good content. Meanwhile, users remain only identified as their Doppel without giving up critical personal info.

Doppels and Merchants. With ‘Doppels’ Merchants can show users products relevant to their interests, mediated by the ‘Doppel’ they are currently using. Shopping for loved-ones can take on an entirely different meaning when users can borrow a partner’s Doppel and shop through their eyes.

Doppels and Content Creators. Sites powered by free content, or even subscription-based services can serve users personalized content and experiences based upon the Doppel the user is currently projecting. In one moment a user may want to see information with a sports emphasis, and next that user may want to see information with an emphasis on world events.

Under all conditions ‘Doppels’ are intentional. Users actively engage by projecting a specific set of interests & traits that communicate how they want to be treated in that moment.

‘Internet users understand that advertising pays for the internet. They just don’t like how creepy this tracking can be. Let’s fix that.

Users know Advertisers want to track them to show them more effective advertising. Users also know that merchants want to identify that they are a returning users to give the user relevant product recommendations. Users know content creators need to be paid if to keep making content, and users know advertising is the primary vehicle by which they are paid.

Advertisers have simply done a poor job at packaging advertising services in a way that makes the user supportive of their mission and business model.

Doppels make tracking feel less creepy.

Rebranding 3rd-party tracking through better storytelling

The advertising business model attempts to learn about user interests to match potential brands and products with the customers most likely to want them. At the heart of the model is a reasonably understandable and noble purpose.

As advertisers worked to make advertising more effective a “creepy effect” surfaced and ultimately triggered social responses, legislative responses, and technological responses.

Advertisers still have time to rebrand their services to maintain a valuable business model into the future. That rebranding must make major strides to show users noble intention.

Doppels rebrand tracking by projecting to users noble intentions and efforts for privacy.

Here is the new deal:

Users of the internet will intentionally present a ‘Doppel’ for trackers to track.

In exchange Whym aims to make users more comfortable with how that tracking works. **A new set of rules will be enforced through technological design rather than the honor system:**

- When logged out - trackers, advertisers, content creators, and merchants can only see the user’s ‘Doppel’ identity and can only make decisions on how to personalize their experience based on the ‘Doppel’ the user presents.
- If users choose not to present a Doppel, trackers are blind.
- If users choose to log-in or self-identify, advertisers may use that information as before.

‘Doppels’ track the trackers, making tracking visible and controllable for the first time

The Doppels platform adds new technological foundations to create transparency and improve understanding between the end user and the companies that would track the user. Here are some of the benefits:

- Every organization that consumes Doppel data is recorded and tracked so users can inspect who has been using their Doppel and where.
- Individual users can shut off any tracking entity they believe is misusing the Doppel while allowing other trackers to remain engaged.
- Tracking Companies, Advertisers, Content Creators, And Merchants will have the ability to explain their usage of Doppel data so users can decide if they are comfortable with each provider.

Users will, on demand, be able to identify which companies are using their data and inquire after it’s use.

Increased transparency paired with relatable user experiences around Doppels will dispel misconceptions about tracking and make tracking solutions feel less creepy. Even more, Doppels and the experiences around them create an environment where users want to be tracked, encourage it, and play an active role in developing the service.

Creating a Doppel & one-time opt-in

The Doppel API and backing user experiences are intended to create a new identity layer for the internet. This new identity layer is simultaneously “private by default”, but also “opt-in pseudo-public” by default.

How do you have both privacy and a public face at the same time? Doppels.

Remember, Doppels:

- have no private information embedded in them,
- are created, modified, and deleted hand-managed by the user themselves,
- and, can be abandoned at any time with no significant user consequences.

This ‘pseudo-identity’ empowers the user to be opt-in by default.

Users can make as many Doppels as they need and switch between them at-will

This empowers users to present the face they want to present, in the moment they want to present it.

A typical user can have a Doppel for work, a Doppel for fun, a Doppel for their partner, and a Doppel representing any other aspect of their life.

Doppels are sharable, transferrable, and copyable. Doppels can be created and abandoned with ease.

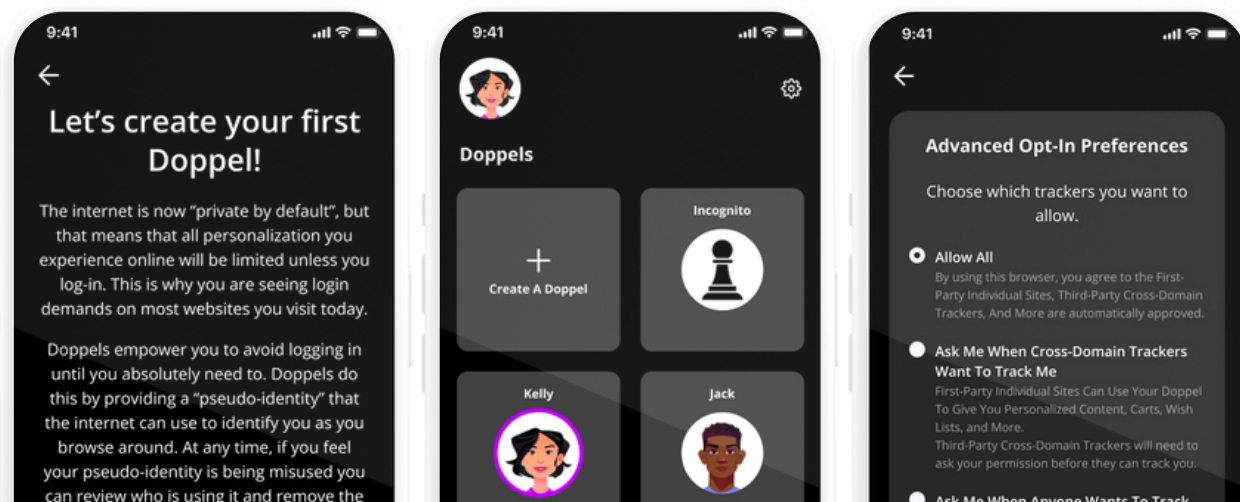
Users only present one Doppel at a time, but they can switch between Doppels at-will.

Users have a one-time opt-in at the very beginning of their journey.

This Opt-In is then transferred intentionally to each new tracking party the user encounters. The user wants this for convenience and agrees to the process because there is much more transparency and visibility in the way the Doppel system tracks the trackers.

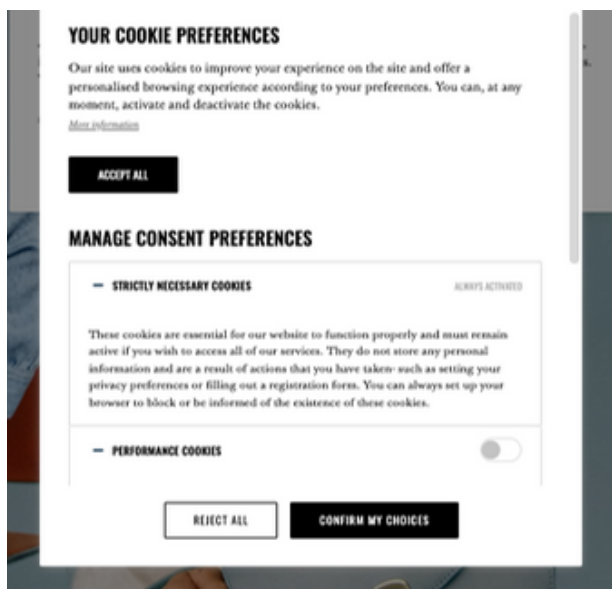
When User A opens the Doppel experience for the first time they will be asked a question about how much auto-approval they are comfortable with.

After this election is made, any category of tracking provider that requests tracking access will be checked against this automatic rule and, when possible, the user will be opted-in to the experience without seeing an opt-in dialog.



The end of the cookie opt-in banner

Online Websites and Web Applications are required by some regulators to publish a “Cookie Opt-In Banner” on their site at the beginning of every user experience. While we cannot change the regulations, we can **provide the user an option to “automatically make elections and close” the banner of every site they encounter.**



The website provider still publishes the banner for compliance but the user enables an “automatic opt-in approval” before they ever arrive.

Reviewing tracker behavior

In exchange for auto opt-in, the users will receive more transparency about how trackers are using their data, where their data is collected, and what data is being tracked.

Better transparency reduces the “creepy factor” and improves Advertiser, Tracker, and User relationships.

‘Doppels’ are opt-in, regulation-tolerant, and incentive driven.

The Doppel API and related User Experience and related Data Architecture are intended to provide similar tracking capabilities to the early-internet third-party-cookie solution, but in a model that differentiated in the following ways:

Opt-In up-front, one-time, across all destinations for 3rd-Party use

The moment a visitor begins using the Doppel Platform they are asked to opt-in for tracking of their Doppel cross-domain. Because this opt-in happens at the very top of their experience the visitor has no need to opt-in again and again.

Each time a new tracker requests access to Doppel data, that tracker is verified against a pre-approved list of trackers (each partner maintains their own list). **The default behavior of our Doppel service is to automatically opt-in to all new trackers**, but visitors may review and modify partner approvals on this list as needed. This is achieved by obtaining a one-time opt-in at the moment a visitor opens the browser for the first time. **Visitors may choose to expect an opt-in request each time a new tracker is met if they desire.**

By default, tracking partners do not have immediate access to any identifiable information about a visitor. This is enforced by all modern browsers by removing the third-party cookie and other mechanisms for tracking a visitor cross-domain. This is the same behavior as any modern browser.

The Doppel Platform creates a new higher-level identity layer and expose only that identity layer to tracking parties. This layer is much more transparent, manageable, and secure. Visitors can elect to exclude certain (or all) tracking parties at will. They can inspect tracking parties at any time.

Regulation tolerance

Doppels are a new tracking device that is:

- opt-in by design
- intentionally shared by visitors
- disassociated from a user's direct identity
- central to the browser experience

It is far less likely that regulators will see a meaningful need, or capability to regulate data that visitors choose to share with platforms intentionally. This is increasingly true if the form of identity shared contains no PII or highly regulated content.

User inspectability by design

The Doppels platform is engineered so that tracking parties cannot evade detection, but tracking parties will willingly want to share their own information with the visitors to drive consumer trust.

- Every organization that consumes Doppel data is recorded and tracked so users can inspect who has been using their Doppel and where.
- Individual visitors can shut off any tracking entity they believe is misusing the Doppel while allowing other trackers to remain engaged.
- Tracking Companies, Advertisers, Content Creators, And Merchants will have the ability to explain their usage of Doppel data so users can decide if they are comfortable with each provider.

Visitors will, on demand, be able to identify which companies are using their data and inquire after its use.

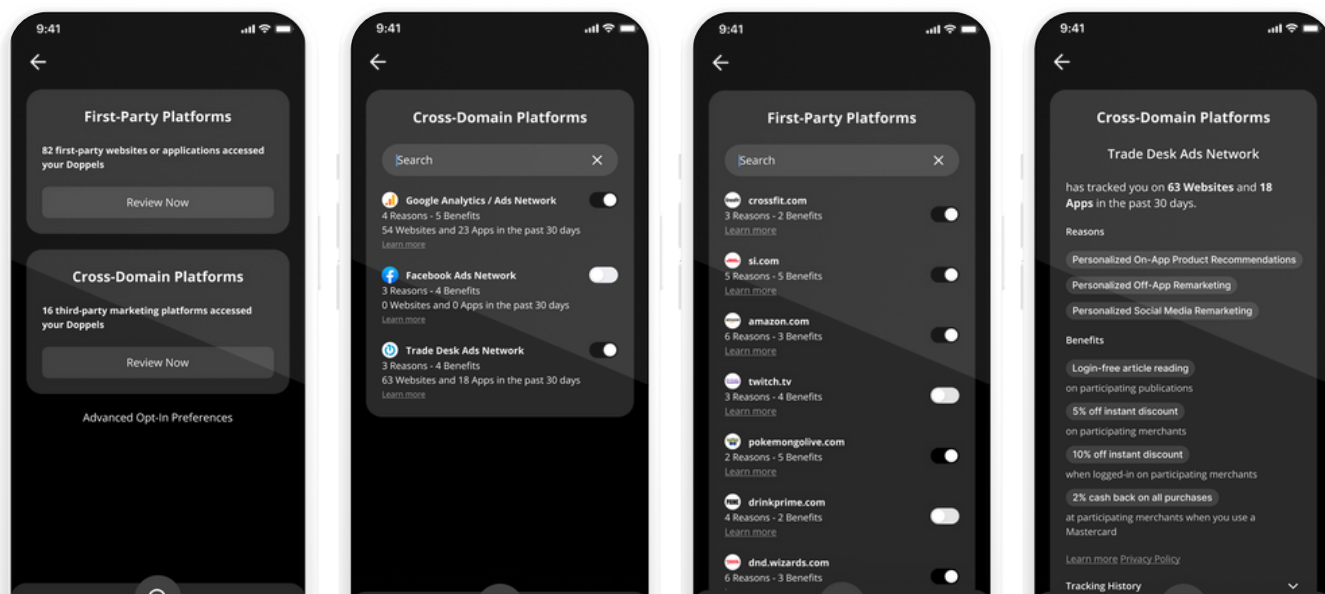
User manageable by design

Advertising and tracking partners benefit from visitor visibility and management because bad-actors can be weeded out of the system without needing to kill the entire capability for all users of the internet.

Visitors can pre-select bad actors, self regulate, and signal to the overall identity system which tracking parties are trustworthy. The more trustworthy a tracking party becomes, the more solidified they become in the overall ecosystem.

Capable of supporting interesting financial and value models previously impossible.

Ask for more details.





Sign Up

Google Form:
<https://forms.gle/SLmT3PCTcN4vHZM69>

Get On The Waitlist

Sometimes touching the product is far more useful than reading pages and pages explaining the product to you. Our Engineers are currently testing an 'Alpha' Doppels experience today that is **embedded in our Whym Shopping App & Chrome Browser Extension**.

We are nearing our Closed Beta Launch (scheduled for Q1 2024).

Get on the waitlist today and we'll get you early access to our beta implementation so you can begin testing and integrating the service early.

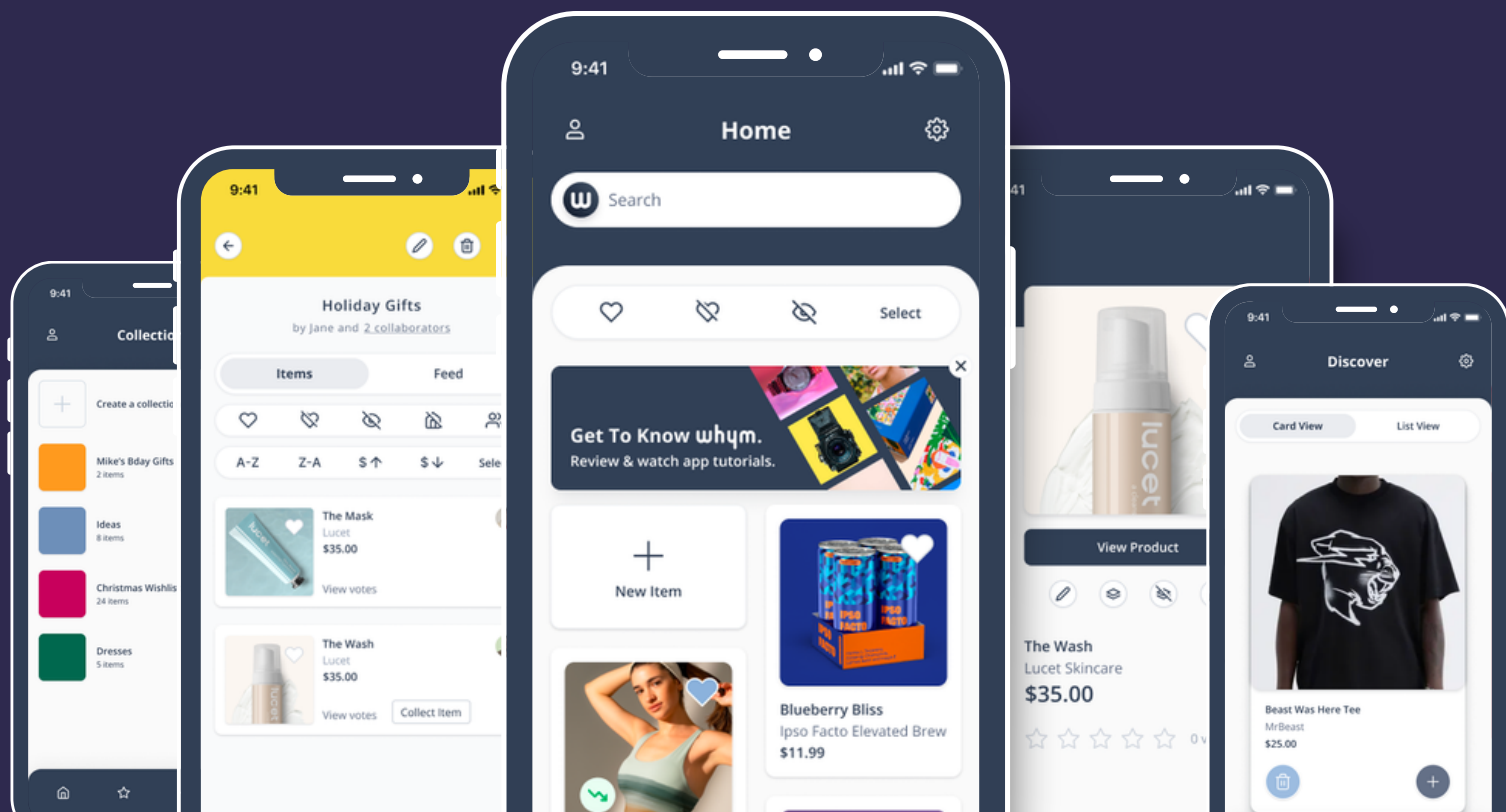
Build your Shopping Inbox



Get it on the
App Store



Add to
Google Chrome



whym.

‘Doppel’ Identity Platform Deep Dive

PATENT PENDING

CONFIDENTIAL DRAFT
PROPRIETARY SPECIFICATIONS
COPYRIGHT 2023

RYAN HORNBERGER
RHENEE BARTLETT
KELLY NYLAND

Note: This content represents a continuous work that is evolving regularly.

We aim to keep this document updated with the latest evolution of the platform design, intended uses, and implementation.

As a result of the ongoing and iterative nature of our development process, the final implementation of this solution may differ from the contents of this document.

CONFIDENTIAL DRAFT
PROPRIETARY SPECIFICATIONS
COPYRIGHT 2023
PATENT PENDING

RYAN HORNBERGER
RHENEE BARTLETT
KELLY NYLAND

Doppel IDs and Tracking IDs

Doppels have multiple IDs depending on the party requesting the information.

Internal Service ID (Private)

Internal to the Doppel Platform is a unique ID that is never published or distributed outside of the application. This ID helps the Doppel Platform maintain a single, unchanging identifier for managing the Doppel.

First-Party Tracking ID (Domain Constrained)

When any party requests a “Doppel-tracking-id” in **first-party mode**, a unique ID is generated that is keyed against the domain name of the requesting party website (or app id of the requesting application). This ID will not change from visit to visit but it is not the same ID used by third-party trackers. If this ID is shared cross-domain it has no use to external users because it will not match any other party’s identifier.

Third-party tools servicing first-party software solutions can use this same ID for all first-party related work and will be blocked from cross-domain tracking.

Third-Party Tracking ID (Cross-Domain)

When any party requests a “doppel-tracking-id” in **third-party mode**, a unique ID is provided that is the same for ALL third-party trackers.

This id will not change from visit to visit, or even domain to domain, but it is not the same ID used by first-party applications. This ID is intended to be shared on a broader uncontrolled data space.

Doppel “reset” behaviors

When a user desires they may, at their own discretion, reset the tracking id on the Doppel. This will be marketed to the user as a “reset all trackers” feature and explained as a means of clearing any tracking entity’s hold on their data. When a Doppel is reset the following behaviors are followed:

- **Internal Private ID:** Remains the same. This ID is never exposed so it is not leaked.
- **First-Party Tracking IDs:** All first-party IDs are replaced with new unique IDs
- **Third-Party Tracking ID:** Replaced with a new unique ID

Doppel explicit interest signals

In addition to tracking IDs, Doppels are intended to allow the user to curate the information they project to the internet for personalization. Users can be as thorough or silent as they want in their explicit interest signals. The key here is that the user gets to personally curate the data revealed in this database.

Further, each time a tracking party attempts to use this data, just like when they attempt to use the Doppel ID, the Doppel Platform will record this behavior and provide tracking analysis data in a report to users upon request.

Doppel Tracker Registration and how it is used

Introduction

The Doppel Platform, through the use of Doppels, creates an opt-in by default environment. However, in exchange users obtain much better visibility into the parties involved in their tracking and take control over their allowances for each party, one-by-one.

This is managed by requiring each tracking party to produce a registration document and host it on an https domain with a signed https certificate. This registration document is used to create an index of all trackers currently using the user's Doppel data, where the trackers have visibility, and what information the trackers are observing.

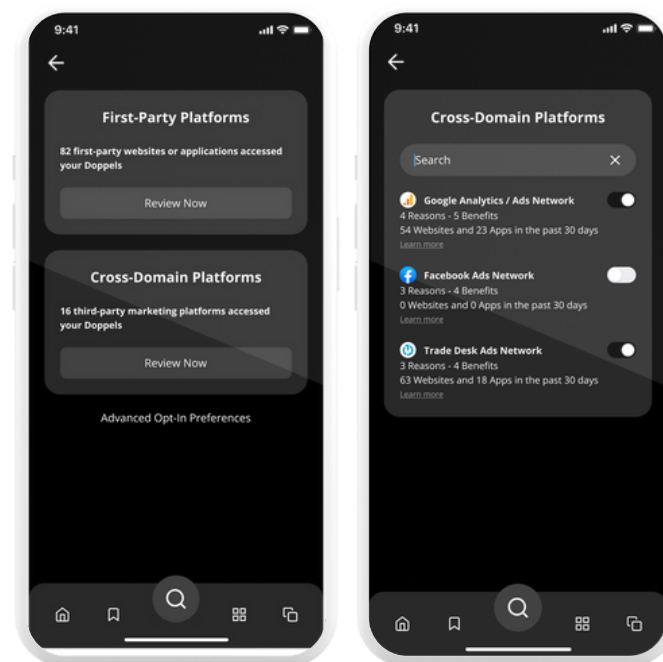
The Doppel Platform then uses this information to produce reports for the user, upon their request, of recent tracking behavior. We also give the user the ability to create additional restrictions on a per-tracker basis.

User review and changing of current opt-in tracking providers

After a user has made opt-in elections, or a system has automatically approved certain opt-ins, a user must be able to review what their opt-in choices have been and change their elections.

User Experience Story

At any time a user may view a list of current opt-in elections and modify their elections. Users may also review which providers are accessing their data, what specific data they are accessing, and where they are accessing this data from.



Enforced by the platform

The Doppel Platform maintains a list of opt-in customization choices, separated by first-party and third-party tracking. Each tracker has an on-off switch as well as a means to inspect their user of the tracking data.

Tracker-self-identification is engineered into the integration. We have built defenses for tracker impersonation, misidentification, and attempts at obfuscation. Trackers must agree to this level of observation in order to use the Doppel service.

Request a copy of our API documentation for further details on how these features are enforced and defended.

API availability through Pixels and Websites

Companies may access Doppel tracking data from their web libraries, pixels, and primary websites.

All companies, regardless of their relation to the Doppel user will have the ability to request access to Doppel data through the Doppel Browser Embedded* API.

Definitions

First-Party Data

First-party data refers to information collected directly from individuals or customers by a company or organization.

Third-Party Data

Third-party data refers to information that is collected and aggregated by external sources or data providers. This data is collected from a variety of sources, often not directly related to your organization.

First-Party powered by Third-Party

The company combines its first-party data with relevant third-party data. For example, a retail company might integrate its customer purchase history (first-party data) with third-party data on consumer preferences and shopping behaviors to create more targeted marketing campaigns.

* Doppel Browser Embedded API - The Doppel Service is currently deployed as a Browser Extension. This extension embeds a new API service that appears to all web applications as if the service is delivered by the browser itself.

Merchant and Publishers

First-Party - Any Tracker desiring to access Doppel data about a visiting user will be able to do it using the 'first-party' mode of the Doppel API. The '**first-party**' mode will only give this provider data when the user is on the first-party website or application, but this data will not match when the user visits other websites or apps.

Utilities Powering First-Party Features

First-Party powered by Third-Party - Any application supporting the core features of a first-party solution desiring to access Doppel data about a visiting user will be able to do it using the '**first-party**' mode of the Doppel API. The data received in this mode is only useful on the domain from which it was requested.

Cross-Domain Tools

Third-Party - Consider tools and platforms that support features that need to span domains (like login solutions, payment solutions, and more). These providers use the '**third-party**' mode of the Doppel API so that they can have the same ID for the same user across many domains and provide cross-domain services.

Cross-Domain Trackers

Third-Party - Tracking platforms are considered to be the backbone of online personalized experiences. We want to support any number of innovations that such platforms may invent. The only thing the users ask in return is some transparency on how the data is used, where tracking is happening, and how often that tracking is happening. Use the '**third-party**' mode of the Doppel API.

Financial partnership w/ Advertisers, Publishers, and other providers

The 'Doppel' Identity Framework (patent pending) is an "exchange of value" system. Identity information becomes a commodity supplied by a source application to websites and on-device applications.

We believe the best possible business model for the Doppel Platform would be to install the technology into a stand-alone Browser. For this current iteration of the platform we are built as a Browser Extension.

Now that all current browsers are Privacy Browsers and they are focused on making third-party platforms blind to the top of the funnel, user identity is a very important and in-demand product the Doppel API can deliver.

To power and fund the service we intend to charge for services rendered. The specific price is to be determined, but the model seems relatively simple:

- Companies desiring tracking services request and receive identity information.
- A count of service usage is maintained and the party billed.

Financing the identity service also empowers partners to influence future development.

Presently browsers are the primary and most effective gateway users have to researching, discovering, and interacting for the first time with newly discovered merchants, content providers, and more.

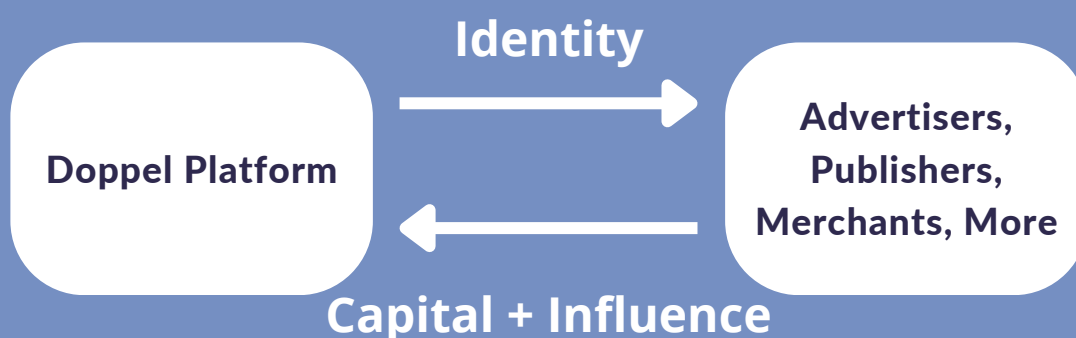
It is unfortunate then that the most important suppliers of content on the internet have no influence over how the browser is developed.

While we are not currently deploying this technology as a stand-alone browser, we believe the future of this service will be bundled as a browser.

We have engineered the Doppel API to have a strong constraint on the service: **the tracker must identify themselves if they want access to the Doppel data.**

While this serves the user well by giving them a way to review who is tracking them, it is also a key tool in helping us build a financial engine that can be used to fund a browser (or in the short term the browser extension).

This funding mechanism is a virtuous relationship between the platform and the tracking parties. Identity consumers become the primary customers of the service, and this gives the tracking parties influence over future development for years to come.



Billing by partner self-identification

The financial foundations of the 'Doppel' Identity Framework are simple to describe: trackers using the Doppel API pay for their usage. The complexity of implementation of such a service is where the framework's magic exists. The first-step of this operation involves identifying the correct billable party when they access the API.

Tracking providers can call the Doppel API with or without a financial relationship with the Doppel service immediately. Of importance, each **tracker must identify** itself during tracking requests using signatures powered by https and public key infrastructure.

The Doppel database will be tracking usage rates of each tracker and as trackers climb above a certain minimum threshold the Doppel Provider sales team can call them to discuss a financial payment plan for continued service.

For potential partners who do not want to sign an agreement for payment for the service, the service can be shut off, on every instance, across the internet that offers the Doppel API.

Partners who sign the agreement see no disruption in service and continue to receive the ID information.

Partners who agree to become Strategic Investors before the service publicly launches will, with their investment, receive a unique pricing plan that complements the value of their investment and the risk they took in helping us build this platform to it's full potential.

Base data packages include

- Doppel IDs on each website visit

Upsell features may include

- Full Doppel content including explicit interest signals, ML-backed potential signals, and website specific historical signals (like recently viewed products)

Future upsell features may include

- Sales Attribution signals
- One-Click Login capabilities
- more...

Defining a Doppel API Partner

.Advertising Networks, Data Providers and Aggregators, and similar are initial-case top-shelf partners we are considering for the Doppel API.

These providers have code integrated on various websites throughout the internet including merchant sites, content sites, and more to service their need for cross-domain tracking.

These providers have an immediate and pressing need for shopper identity in a 3rd-party context. By proxy, through advertising, this model will meet the most immediate needs of many D2C merchants and content creators.

Merchants and Content Creators are also invited (and encouraged) to access the Doppel API to better provide more personalized shopping experiences on their sites.

Any number of other parties, ie. data providers / networks, retailers, credit card providers, etc., may also be interested in direct access to 'Doppel' data, and all are invited. Provided that each party accurately identifies themselves to access this data.

Self-identification is tied to files hosted on tracking party domains

In an attempt to make the service easy to approach for any company desiring to use the Doppel API service, we have engineered the service to avoid any need for prior registration with the Doppel Provider. The API standard, as described below is auto-discoverable and immediately available to anyone who configures themselves for the service.

To access the Doppel API and receive Doppel data the requesting entity simply needs to self-identify by installing a file on the tracking party's own first-party website domain. (If the tracking party is The Trade Desk, then The Trade Desk installs this file on TheTradeDesk.com, not the websites that they provide services to.)

The details of the requirements, format, and contents of this file are detailed in the integration guidelines. Contact us for a copy.

- This file must be hosted on an HTTPS certificate domain
- This file contains details about the requesting party's identity
- This file contains details about how to communicate Doppel data back to the party in a secure way

Request a copy of our Integration Guidelines and Documentation for more details.

Preventing Doppel API partner "impersonation"

The 'self identification' model we just described can invite 'impersonation' attacks from bad actors. The solution to preventing impersonation is to encode responses so that only the requesting party can read them.

The API employs a number of security protocols to ensure that only a valid requesting party can read the responses of the API.

Doppel API trackers attempting to impersonate

other providers are then unable to read the responses from the Doppel Platform because the responses are protected in a number of ways that only the true requesting party can read.

Tracking parties must avoid decoding encrypted data on the client-side and should only decode encrypted data during server-side operations.

Preventing Doppel API bill inflation attacks

We have made it a priority to discourage impersonation, but some of these defenses (not all) do not prevent attacks that would create over-billing. You could imagine that a malicious actor could make requests pretending to be another party without ever intending to decode the responses. This attack is simply aimed at driving higher bills for the attacked party.

This attack is mitigated using a two-stage validation operation

1. Doppel API usage will **only bill once for the same data during the same session on the same site**; regardless of the number of requests to the API.
2. Doppel API consumers can publish a **blacklist** file indicating to the Doppel Platform that there are a number of malicious domains that are misusing the api. This will disable the API on those domains when the requests are made on behalf of these specific API consumers. An alternative model is being considered as a whitelist file.

Preventing unauthorized sharing of limited Doppel API data

The Doppel API classifies exposed data into 2 categories: 1) shareable, and 2) not-shareable. Usage of the API requires all parties to agree to the licensing terms of the API usage. These terms define which pieces of data cannot be shared beyond the initial receiving party.

Existing financial models and 'Pixels' remain unaffected

While the Doppel Identity Framework creates a new financial model for funding browser (or browser extension) development on the internet, it **does not require every independent merchant to create a relationship directly with the Doppel API.**

Note - This relationship is possible, see below, just not required.

First, let's evaluate existing financial models and see how they would work.

Advertising Pixels

Demand-side advertising buyers (merchants, retailers, brands, small businesses, etc.) will still install pixels from their Advertising Partners (Trade Desk, Google, Facebook, Tiktok, Klaviyo, Attentive, etc.). Supply-side advertising real-estate suppliers (content creators, news outlets, media providers, etc.) will still install pixels from their Advertising Partners. Neither the Demand-Side partners nor the Supply-Side sites themselves need to personally interact with the Doppel API directly for the purpose of advertising. **Advertising platforms can access the Doppel API directly from their pixel code and use the identity they receive to make decisions in ad auctions.**

The Doppel API will see that the Advertising Platform requested the 'Doppel' data and **will bill the Advertising Platform (not the merchant) for the usage of the API**, even if the request comes from the merchant domain. The Demand-Side Customers pay the Advertising Platform, the Advertising Platform pays the Supply-Side Content Providers, and the Advertising Platform now sends some funds to the Identity Provider (the Doppel Platform).

Browser Search Licensing

In the event that we do launch or partner with a major browser, existing browser contracts for licensing of search technologies remain completely detached from the Doppel Identity Framework. It is a completely separate stream of revenue for the browser.

Browser-User Enhanced Features Subscriptions

Browsers that make revenue from subscriptions for enhanced features may continue to do so. Once again, on a completely separate revenue stream.

Payment Processing Networks

The Doppel Identity Framework is not entangled with checkout or payments directly. The network focuses on an entirely distinct Identity service. Billing will be tied to identities delivered, not orders completed.

Content Creators Funded By Ads

Free "Ad-Sponsored" content online is the backbone of modern news and social conversations. While Social-Media is the key "First-Touch" social graph, all deeper reading and conversation still happens in News Networks, Blogs, Journals, Magazines, and other Long-Form Content environments. These places still depend on advertising to drive revenue for their content. This model remains unchanged. The key upgrade here is that these platforms still have a chance at achieving financial stability without users needing to sign-in.

Without Doppels this business model is lost, with Doppels it has a chance to continue.

whym.

‘Doppel’ Identity Platform Developer API

For Technical Staff To Implement And Test The Doppel API

PATENT PENDING

CONFIDENTIAL DRAFT
PROPRIETARY SPECIFICATIONS
COPYRIGHT 2023

RYAN HORNBERGER
RHENEE BARTLETT
KELLY NYLAND

Note: This whitepaper represents a continuous work that is evolving regularly.

We aim to keep this document updated with the latest evolution of the platform design, intended uses, and implementation.

As a result of the ongoing and iterative nature of our development process, the final implementation of this solution may differ from the contents of this document.

CONFIDENTIAL DRAFT
PROPRIETARY SPECIFICATIONS
COPYRIGHT 2023
PATENT PENDING

RYAN HORNBERGER
RHENEE BARTLETT
KELLY NYLAND

Doppel API Available In Browser-Hosted Environments

The Doppel API is delivered into the root javascript environment of every webpage when a website is visited by a user when the services are available by browser extension or browser. We aim to support all of the following environments in time:

Current

- **Safari iOS - By Browser Extension**
- **Chrome OSX - By Browser Extension**
- **Chrome Windows - By Browser Extension**
- **Kiwi Browser Android - By Browser Extension**

Future

- Doppel Browser iOS
- Doppel Browser Android
- Doppel Browser OSX
- Doppel Browser Windows

Consumers of the Doppel API (tracking parties) do not need to install specialized SDKs or any specialized tooling to access the Doppel API.

First-Party and Third-Party Providers Use The Same API

Setup Requirements

The Doppel API requires all tracking parties (including merchant or content websites, but also all third-party trackers) to identify themselves to the service so that users can review tracking behaviors of different parties.

While the API is the same, each party receives slightly different treatments. Read more below. In all conditions the tracking party must self-identify.

Defining A “Tracking Party”

It may seem obvious to understand that third-party tracking solutions like Google Analytics, Facebook Pixel, and related solutions are “Tracking Parties”. What is not so obvious is that the Doppel API considers ALL parties who access the API to be tracking parties. This includes:

- **Merchant Websites**
- **Content Websites**
- **Merchant-Installed Tooling** or other Service Companies (Like Wishlist Providers)
- **Cross-Domain Solutions** (Like Single-Sign-On or One-Click-Payment Providers)
- **AND all third-party trackers**

Install A Domain-Based Identification File

All tracking parties are expected to identify themselves to the Doppel API so that the API can produce reports for users about which parties are tracking them. To ensure that tracking parties are identifying themselves truthfully we need to check a tracking party against a trusted identity infrastructure. In this case we will be verifying identities based on the tracker’s domain name using https.

To accomplish this a tracking party needs to:

(1) **Decide on a domain name to be the “primary” domain of tracking.** The domain chosen will need to follow specific requirements:

(a) Any domain will work, and the domain DOES NOT need to match the domain the user is visiting. For instance:

(b) When a user is visiting **MerchantA.com**, MerchantA may use a tracking url of **p.MerchantA.com** or **MerchantAParentCompany.com**. Third-Parties may use **TrackerB.com**. The domains do NOT need to match.

(c) The tracking domain chosen must be capable of hosting a static file for download.

(d) The tracking domain chosen must have a connected and valid https certificate. The file will only be downloaded by the Doppel Platform if the https certificate is valid.

(2) **Host a marketing page describing their organization on the same domain.** Whichever domain you choose, you will be expected to give the user a url that they can click on to learn more about your company and how they are tracked.

(3) **Host a privacy policy and terms of service on the same domain.** Also on the same domain you choose you will need to provide the user multiple urls that the user can click on to read more about your privacy policies.

(4) **Create a private/public key pair for serverside decoding of Doppel data.** Doppel data is always packaged as an encrypted bundle so that unannounced third-party observers are unable to digest the data received. This data is exclusively packaged for the requesting party. Read more below.

(5) **Create and host a “Doppel Tracker Identification File”.** This is the file the Doppel API will use to identify your tracker and encrypt its communication to you. The file is formatted in the following form.

The “Doppel Tracker Identification File” can be installed anywhere on the tracker’s primary domain but it must have the following naming convention:

```
<any-name>.doppel-tracker.json
```

The suffix for the file must always be **“.doppel-tracker.json”**

These tracking files have specific configurations. A tracking party may choose to have more than one tracking configuration depending on the situation. If you intend to use more than one configuration you can host more than one tracker file. These tracker files do not need to be in the same directory. For Example:

```
/path/a/to/file/social-media-first-party.doppel-tracker.json  
/path/b/to/file/somewhere/else/social-media-third-party.doppel-tracker.json
```

The contents of the file will be a json object following a very specific format. Here is an example configuration file for a **first-party tracker**:

```
/path/to/merchant-a.doppel-tracker.json
```

```
{  
  "version": 1,  
  "tracker": {  
    "authDomain": "merchantA.com"  
    "displayName": "Merchant A"  
    "displayIconUrl": "/path/to/icon.png"  
    "learnMoreUrl": "/path/to/learn-more/page"  
    "privacyPolicyUrl": "/path/to/privacy-policy"  
    "termsOfServiceUrl": "/path/to/tos"  
  },  
  "trackingMode": "single-domain-first-party"  
  "security": {  
    "publicKey": "example-jwk-public-key"  
  }  
}
```

NOTE: Continue to the next page to learn how to create a usable public key.

NOTE: **First-Party** trackers receive a tracking ID that is not transferrable to other domains. **Third-Party** trackers receive an identifier that is sharable and usable across domains.

Another example for a **third-party tracker**:

```
/path/to/tracker-m.doppel-tracker.json
```

```
{
  "version": 1,
  "tracker": {
    "authDomain": "trackerM.com"
    "displayName": "Tracker M"
    "displayIconPath": "/path/to/icon.png"
    "learnMorePath": "/path/to/learn"
    "privacyPolicyPath": "/path/to/privacy"
    "termsOfServicePath": "/path/to/tos"
  },
  "trackingMode": "cross-domain-third-party"
  "security": {
    "publicKey": "example-jwk-public-key"
  }
}
```

Install this file anywhere on the domain. When you begin using the Doppel API you will have the opportunity to tell the API where to find the file. See below.

Specific details of the contents of each field of this file are outlined below.

Creating your Public/Private Keypair

NOTE: Alternative models for accessing the Doppel API that do not require encryption may also be available in the near future. Contact our engineers if you have an interest connecting to the Doppel API without encryption.

IMPORTANT - Private Keys Must Private.

Your private key must remain private. It should never be installed on public facing javascript libraries. **This private key is intended to be kept for exclusive use by servers that are used to return queries or record tracking data.**

IMPORTANT - Key Rotation And Key Caching.

Changing your keypair is possible, and regular

changing of your keypair is recommended. When you make a new keypair you will need to make sure that your server-side decoder will continue to be capable of READING messages signed with the prior keypair for a period of time to account for cached copies of your public key on user devices.

Generating Your Public/Private Keypair.

The Doppel API returns all answers in an encrypted. The Doppel API uses RSA-OAEP asymmetric public-key encryption to send your service encrypted Doppel data.

You can use any library or utility to generate your private and public key-pair. The below code example uses the console in your browser to generate a keypair.

```
let keyPair = await crypto.subtle.generateKey(
  {
    name: 'RSA-OAEP',
    modulusLength: 4096,
    publicExponent: new Uint8Array([1, 0, 1]),
    hash: 'SHA-256',
  },
  true,
  ['encrypt', 'decrypt'],
);
```

Your public key needs to be exported into a JSON Web Token and converted to a JSON string.

```
const publicKeyAsString = JSON.stringify(
  await crypto.subtle.exportKey(
    'jwk',
    keyPair.publicKey
  )
);

console.log(publicKeyAsString)
```

Place the string contents of 'publicKeyAsString' into your 'security.publicKey' field of your *.doppel-tracker.json file.

Don't forget to export your private key for use on your servers for decoding Doppel data later.

```
const privateKey = await crypto.subtle.exportKey(
  'jwk',
  keyPair.privateKey
);

console.log(publicKey)

// if you want to transfer your private key as a
string you may need to use 'JSON.stringify'.

let privateKeyStr = JSON.stringify(privateKey)

console.log(privateKeyStr)
```

Using your Private Key to Decrypt Doppel Data (Server-side)

IMPORTANT - There are two motivations behind why we encrypt Doppel data.

1) End-User Privacy - End users can control which parties have access to their Doppel data, disabling one party while enabling another. Without encryption this level of control would not be possible.

2) Billing Impersonation Prevention - Without encryption it would be possible for a party to impersonate another, read the Doppel data, and have the bill for use of the service sent to the impersonated party.

To prevent both of these risks, Doppel data is returned in an encrypted form that only the requester can read. It is up to you to make sure that your private keys are kept private.

Absolute security requires that you decode all Doppel data server-side.

Any RSA-OAEP capable decryption library will work but the following example uses the Node.js 'Web Crypto API' called 'subtle'.

```
const { subtle } = globalThis.crypto

const privateKeyStr = /* load your private key */
const privateKeyJWK = JSON.parse(privateKeyStr)

const privateKey = await subtle.importKey(
  'jwk',
  privateKeyJWK,
  {name: 'RSA-OAEP', hash: 'SHA-256'},
  true,
  ['decrypt']
);

const decryptDoppel = async (encDoppel) => {

  // convert the encDoppel to a byteArray
  const encDoppelBytes = Uint8Array.from(
    atob(
      base64EncryptedMessage
    ),
    c => c.charCodeAt(0)
  );

  // decode to a new byte array
  const decDoppelBytes = subtle.decrypt(
    {
      name: 'RSA-OAEP'
    },
    privateKey,
    encDoppelBytes
  );

  // convert to a string
  const decDoppelStr = String.fromCharCode(
    ...new Uint8Array(decDoppelBytes)
  );

  // convert to an object
  const decDoppel = JSON.parse(decDoppelStr)

  return decDoppel
  // returns an IDoppelData
};
```

NOTE: Alternative models for accessing the Doppel API that do not require encryption may also be available in the near future. Contact our engineers if you have an interest connecting to the Doppel API without encryption.

The Importance Of Pixels For Third-Parties

Historically “Pixels”, or “First-Party Installed, Third-Party Tracking Code” have been installed by third-party trackers onto merchant and content sites to have visibility into a user’s cross-domain behaviors.

The Doppel API does NOT blindly send all tracking data from every website on the internet to all trackers that request it.

Third-Party trackers MUST have pixels installed on the First-Party sites to have access to Doppel API Data. For example:

MerchantA is only interested in tracking user behavior on their own site.

Merchant A - Limited to 1 Website of Doppel Data

TrackerM is interested in having tracking data for every website on the internet. But they only have pixels installed on 52 websites on the internet.

Tracker M - Limited to 52 Websites of Doppel Data

TrackerZ is interested in having tracking data for every website on the internet. They have a pixel installed on over 32% of the internet’s websites.

TrackerZ - Limited to 32% of the internet of Doppel Data

All Doppel Data is accessed directly from the user’s browser through javascript installed on the first-party website. Third-Party trackers cannot obtain primary Doppel data without a pixel installed on the first-party website. First parties must write on-website javascript to access Doppel API data.

Third-Parties Access The Doppel API Through Their Pixel Code Installed On Each Destination Website

To access the Doppel API a third-party solution must add Doppel code to their pixel library installed on each merchant website.

One of the most important advantages of this method of installation is that every third-party should be able to deploy this upgrade without consultation with the first-party domain owner.

The use of the Doppel API is entirely tied to the tracking party, the first-party is not billed or involved in a third-party tracking relationship with the Doppel API.

Checking Doppel API Availability

Not all browsers support the Doppel API, so verifying a browser has the extension that enables the Doppel API is the first step to using Doppels for user identification.

The following javascript should be added to the first-party top-level javascript (if the tracking party is the first-party) or to the third-party pixel library code (if the tracking party is a third-party)

If the javascript call returns a result from “window.doppel” then the browser extension is installed and supports the API. If not, then it does not. Simple.

```
if(window.doppel) {  
  // Doppel API available  
} else {  
  // Doppel API NOT available  
}
```

Open A Secure Channel To The Doppel API

Now that you know the Doppel API is available, you must identify your tracker and request access.

Now that you know the Doppel API is available, you must identify your tracker and request access.

Unlike Cookies the Doppel API does not blindly return a response to any requester. It will first need to be informed which party is requesting this information and how they intend to use it.

The Doppel API is asynchronous and promise-based.

```
if(window.doppel) {  
  window.doppel.register({  
    authDomain: 'merchantA.com'  
    authFilePath: '/a.doppel-tracker.json'  
  }).then(  
    (secureDoppelApi) => {  
      if(secureDoppelApi) {  
        // access is granted, proceed  
      } else {  
        // access DENIED. See below.  
      }  
    });  
}
```

Async/Await Is Supported

If you are writing your code using modern Javascript or Typescript versions that support “async/await” the code is much cleaner.

Not all code installed by first-parties or third-parties support “async/await”. If you are unsure it is likely that your library does not support this feature.

```
if(window.doppel) {  
  const secureDoppelAPI = await window.doppel.register({  
    authDomain: 'merchantA.com'  
    authFilePath: '/path/to/merchant-a.doppel-tracker.json'  
  });  
  
  if(secureDoppelApi) {  
    // access is granted, proceed  
  } else {  
    // access DENIED. See reasons below.  
  }  
}
```

[Documentation Continues On The Next Page]

Request Encrypted Doppel Data

The secure Doppel Api endpoint provides a pre-loaded, authenticated, reusable endpoint to request Doppel data. This allows you to ask for smaller quantities of data early and larger quantities as needed without needing to re-

instantiate the library.

Remember, not all browsers support the Doppel API, so verifying a browser has the extension installed to support the Doppel API is the first step to using Doppels for user identification.

Assuming access has been granted you will receive an interface that is ready to take requests regarding the current user's Doppel.

Accessing the information in the API is simple:

```
let encryptedDoppelData = null

if(window.doppel) {
  const secureDoppelAPI = await window.doppel.register({
    authDomain: 'merchantA.com'
    authFilePath: '/path/to/merchant-a.doppel-tracker.json'
  });

  if(secureDoppelApi) {
    encryptedDoppelData = await secureDoppelApi.read({
      scopes: [
        'trackingId',
        'freeFormDescription',
        'preApprovedOptInChoices',
      ]
    });
  }
}

// in this moment personalization libraries can make
// server queries for personalized results
const personalizedResults = await yourApi.queryPersonalization({
  query: 'data',
  encryptedDoppelData: encryptedDoppelData
});

// or tracking parties can send tracker signals
yourTrackingApi.track({
  trackData: 'data',
  encryptedDoppelData: encryptedDoppelData
});

// NEVER TRY TO DECODE THE DOPPEL DATA IN YOUR CLIENT CODE. DECODING SHOULD
HAPPEN EXCLUSIVELY ON YOUR SERVERS
```

Decoding Encrypted Doppel Data (Server-side Only)

Once your server-side receives a request that includes Doppel data, the Doppel data can be decoded using your private key and used for personalization of the user's experience.

Decoding of Doppel data should always happen on the server-side of your systems to preserve the privacy of the Doppel data from unauthorized third-parties.

The following example uses node.js but any language and back-end library will work.

```
const { subtle } = globalThis.crypto

const privateKeyStr = /* load your private key */
const privateKeyJWK = JSON.parse(privateKeyStr)

const privateKey = await subtle.importKey(
  'jwk',
  privateKeyJWK,
  {name: 'RSA-OAEP', hash: 'SHA-256'},
  true,
  ['decrypt']
);

const decryptDoppel = async (encDoppel) => {

  // convert the encDoppel to a byteArray
  const encDoppelBytes = Uint8Array.from(atob(base64EncryptedMessage), c => c.charCodeAt(0));

  // decode to a new byte array
  const decDoppelBytes = subtle.decrypt(
    { name: 'RSA-OAEP' },
    privateKey,
    encDoppelBytes
  );

  // convert to a string
  const decDoppelStr = String.fromCharCode(
    ...new Uint8Array(decDoppelBytes)
  );

  // convert to an object
  const decDoppel = JSON.parse(decDoppelStr)

  return decDoppel
  // returns an IDoppelData
};
```

API Documentation

Registering With The Doppel API

The Registration API is exposed by the browser extension as 'window.doppel'. At this endpoint there is a registration function. Successful registration will give you a secure endpoint to communicate with the API. A failed registration means that your registration is not valid or that the user has rejected your request.

window.doppel.register(registration:IDoppelApiRegistration)

returns **Promise<null|ISecureDoppelApi>**

Request access to the Secure Doppel API.

PARAMETERS

IDoppelApiRegistration

Object with the following keys and values

authDomain	string 'tracker.example.domain'
authFilePath	string (the location of a IDoppelApiRegistrationFile) '/path/to/merchant-a.doppel-tracker.json' The contents of this file are formatted after IDoppelApiRegistrationFile

RETURNS (AFTER THE PROMISE RESOLVES)

ISecureDoppelApi

Documentation of this API endpoint will be found in the pages below.

Null

If your registration request has failed you will receive a null response back. Failure reasons include 1) your registration is invalid or registration file is unreadable; 2) the user has blocked you from using their Doppel data.

REMOTE DATA REQUIRED

IDoppelApiRegistrationFile

Documentation for the contents of this file will be found in the pages below.

Contents Of The Domain-Based Identification File

See the engineering documentation above on the placement, naming, and https requirements for this file in the section named “Install A Domain-Based Identification File”.

The Registration file must be hosted on YOUR DOMAIN. Third-Parties do not install this file on the First-Party domain, only installing it on their own domain. This file must be hosted using HTTPS.

/path/to/merchant-a.doppel-tracker.json

See the above section on “Install A Domain-Based Identification File” for an example of this file.

IDoppelApiRegistrationFile

(a file containing a json object with the following nested values)

tracker	authDomain	string 'your-domain.com'
	displayName	string 'Your Public Company Name'
	displayIconUrl	string '/path/to/png/icon.png' (the image referenced must be a PNG)
	learnMorePath	string '/path/to/page/or/document'
	privacyPolicyPath	string '/path/to/page/or/document'
	termsOfServicePath	string '/path/to/page/or/document'
trackingMode	TTrackingMode	
security	publicKey	stringcreate private/public keypair (described above) and put the public key here
	See the above section on “Install A Domain-Based Identification File” to learn about the encryption requirements of this key.	
version	TDoppelApiRegistrationFileVersion	

TDoppelApiRegistrationFileVersion

'1.0'

TTrackingMode

'single-domain-first-party' | 'cross-domain-third-party'

peek example

```
{
  "version": 1,
  "tracker": {
    "authDomain": "your-domain.com",
    "displayName": "Your Public Company Name",
    "displayIconUrl": "/path/to/png/icon.png",
    "learnMoreUrl": "/path/to/page/or/document",
    "privacyPolicyUrl": "/path/to/page/or/document",
    "termsOfServiceUrl": "/path/to/page/or/document"
  },
  "trackingMode": "single-domain-first-party",
  "security": {
    "publicKey": "e"
  }
}
```

Authenticated Secure Doppel API

After registration you receive access to a “Secure Doppel API”.

```
const secureDoppelAPI = await window.doppel.register{/*registration goes here*/});
```

If the Doppel API returned this object to you then you are authorized to use it. If you received a **null** response then you have been blocked from using the secure Doppel API (either by user choice or due to an invalid registration).

secureDoppelAPI.read(request:ISecureDoppelRequest)

returns **Promise<IEncryptedDoppelResponse>**

Request specific Doppel data, encrypted so that only the approved reader can use it.

PARAMETERS

ISecureDoppelRequest

Object with the following keys and values

scopes	TDoppelScope[]
--------	----------------

TDoppelScope

'trackingId'
| 'freeFormDescription'
| 'preApprovedOptInChoices'

RETURNS (AFTER THE PROMISE RESOLVES)

IEncryptedDoppelResponse

see the next page for details on the contents of this object.

example

```
const encryptedDoppelData = await secureDoppelApi.read({
  scopes: [
    'trackingId',
    'freeFormDescription',
    'preApprovedOptInChoices',
  ]
});
```

Encrypted Secure Doppel API Response

The secure Doppel API responds with an object that contains an encrypted Doppel record and some additional metadata.

```
const encryptedDoppelData = await secureDoppelApi.read({ /*scopes requested*/ })
```

The contents of this response are outlined here.

IEncryptedDoppelResponse

encDoppel	string (encrypted IDoppelData) an encrypted object, formatted as a utf-8 string, encrypted using the public key provided in the Doppel Tracker Registration File. Can only be decrypted using the tracking party's private key. Decrypted the contents are converted to a IDoppelData								
doppelProvider	** NOTE THIS SECTION IS INCOMPLETE. WE MAY NOT NEED THIS FEATURE. THE FINAL VERSION OF THIS DOCUMENTATION WILL OUTLINE THE DECISIONS WE MAKE ON THIS FEATURE> <table><tr><td>userAgent</td><td> string 'Doppel Browser Ext. 1.3' (or something similar). We are still working out the identifiers here. </td></tr><tr><td>signature</td><td> string (signature string) using a private signing key Doppel will take the encDoppel and generate a signature confirming the extension that created the Doppel data </td></tr><tr><td>authDomain</td><td> string (doppel-browser.com) </td></tr><tr><td>authFilePath</td><td> string (the location of a IDoppelApiProviderSignatureFile) '/path/to/merchant-a.doppel-provider.json' The contents of this file are formatted after IDoppelApiProviderSignatureFile </td></tr></table>	userAgent	string 'Doppel Browser Ext. 1.3' (or something similar). We are still working out the identifiers here.	signature	string (signature string) using a private signing key Doppel will take the encDoppel and generate a signature confirming the extension that created the Doppel data	authDomain	string (doppel-browser.com)	authFilePath	string (the location of a IDoppelApiProviderSignatureFile) '/path/to/merchant-a.doppel-provider.json' The contents of this file are formatted after IDoppelApiProviderSignatureFile
userAgent	string 'Doppel Browser Ext. 1.3' (or something similar). We are still working out the identifiers here.								
signature	string (signature string) using a private signing key Doppel will take the encDoppel and generate a signature confirming the extension that created the Doppel data								
authDomain	string (doppel-browser.com)								
authFilePath	string (the location of a IDoppelApiProviderSignatureFile) '/path/to/merchant-a.doppel-provider.json' The contents of this file are formatted after IDoppelApiProviderSignatureFile								

To use this encrypted Doppel data you will need to send the 'encDoppel' contents to your servers for private decryption. This can be done in-line with query commands for personalized data, track commands for pixel tracking, and more.

Decrypted Doppel Data

The secure Doppel API responds with an object that contains an encrypted Doppel record and some additional metadata. Transmit this data to your servers where you can decode it privately.

```
const decryptedDoppelData = await yourDecodeFunction(encDoppel)
```

The decoded data will take the form of a json object with the following content.

IDoppelData

trackingId	null string unique domain-specific ID if in first-party mode OR unique, internet-wide ID if in third-party mode. Read in the above sections about first-party vs third-party mode. null if not requested in scopes
freeFormDoppelDescription	null string a freeform text string provided by the owner of the Doppel describing the Doppel however they wish null if not requested in scopes
preApprovedOptInChoices	null TPreApprovedOptInChoice[] user choices on pre-approvals for cookie banners null if not requested in scopes
more features coming soon...	more features coming soon...

TPreApprovedOptInChoice

'all-cookies-pre-approved'
| 'no-cookies-pre-approved'

** additional opt-in choices are in development but we have not yet confirmed the final format of these options. Look for future documentation to see future choice options

NOTE - Additional features of the Doppel Data are in development now. We call these features "Explicit Signals". As these features become available we will update the documentation. to detail how to access this information.

How Doppel API Tracks Usage

The Secure API follows a very specific structure., Each piece of data is only retrieved when a function is called requesting the data. By hiding each piece of data behind a scoped function the system can measure and record which trackers are using which pieces of data and can expose that information to the user for transparency.

Engineering Example

If Tracker M only cares to know the user's tracking ID, Tracker M will only access that API:

```
encryptedDoppelData = await
secureDoppelApi.read({
  scopes: [
    'trackingId',
  ]
});
```

When the user reviews the behavior of Tracker M the user will see that they have only been using the trackingID:

```
Tracker M
(Cross-Domain)
• Tracking ID
```

Alternatively, Merchant Z may be using many of the API features:

```
encryptedDoppelData = await
secureDoppelApi.read({
  scopes: [
    'trackingId',
    'freeFormDescription',
    'preApprovedOptInChoices',
  ]
});
```

When the user reviews the behavior of Merchant Z the user will see the complete picture of the API's the merchant is using:

```
Merchant Z
(This Website Only)
• Tracking ID
• Opt-In Choices
• User Doppel Customizations
• Freeform Doppel Description
```

User Review And Changing Of Current Opt-In Tracking Providers

If the user questions a specific provider's usage of their Doppel data, each tracking provider can provide the user with a link to "Learn More". This link can take the user to any destination online where the tracking provider can author a persuasive article encouraging the user to keep their access enabled.

```
Merchant Z
(This Website Only)
• Tracking ID
• Opt-In Choices
• User Doppel Customizations
• Freeform Doppel Description
```

[LEARN MORE](#)

SSL Certificate Identity Review

The known information about the tracking party must be hosted by the tracking party's own first-party domain which must be hosted by with SSL certificates. The End-user (and our quality team) will be given a UI by which to inspect the SSL ownership of the tracking party,

If additional requirements are needed to validate the identity of tracking parties (due to bad actors) we will add additional validations as needed.

whym.

**Get in touch with our team
to learn more about how to use the
Doppel API.**

hello@whym.tech