Leveling Your Business Up The Privacy Maturity Pyramid

Build a foundation for delivering successful privacy outcomes at scale

ethyca

Contents

Introducing the Privacy Maturity Pyramid

In thousands of conversations with privacy stakeholders from privacy program managers to general counsels, from heads of privacy engineering to the single scrappy DevOps engineer who's drawn the assignment to "solve for CCPA," a pattern emerges regarding a business's journey to privacy maturity.

It takes the form of a hierarchy of needs, in that a business must solve lower-order problems to progress to the next level of the hierarchy. Since "Privacy Hierarchy of Needs" sounds a little clumsy, internally we've taken to calling this journey the ***Privacy Maturity Pyramid.*** We can use this framework to evaluate where any organization is on its privacy journey and make assessments about where it should focus efforts to level up privacy outcomes.

We use the Privacy Maturity Pyramid to better understand our users, and we're sharing it with you because we believe it's a useful tool for any business to build its privacy roadmap, inform resourcing and procurement decisions, or simply orient itself in the rapidly changing world of data privacy regulation.

Who can use this framework, and how?

You will find this document useful if:

- You're a **legal**, **compliance**, or **engineering** professional seeking to establish or level up a privacy program in your business.
- You may be starting from scratch on privacy due to a new regulation impacting your business.
- Or you may be part of a privacy team that has built processes to accommodate for laws like GDPR or CCPA, but is feeling the strain of inefficient or insufficient privacy processes.
- You may even have purchased a privacy technology solution for some part of your privacy operations, but are concerned that it is not satisfying the expanding range of privacy use cases for your business.

Here are the questions that this document will help you answer:

- Am I prioritizing the right challenges in order to level up my team's privacy operations?
- What lower-level gaps exist in our privacy ops that are hindering our ability to solve the challenge in front of us?
- What built or bought solutions are the best match for the privacy use cases of our business?
- Ultimately, how do I get my team to deliver better privacy outcomes faster?

So, let's take a first look at the Privacy Maturity Pyramid together…

Be Trusted
Privacy by Design

Keep Promises
DPIAs, Privacy Checks in CI

Make Promises
Data Subject Requests, Consent Management

Know My Environment
Data Classification, Data Mapping

Solving the Privacy Pyramid

For the remainder of this document, we're going to walk through solving for each level of this Pyramid, and the different approaches that organizations can take to tackling them. These choices are often a function of two constraints: **the complexity of their data stack** and **the privacy resources in their business.**

We'll also introduce one additional lens that will be important for how we evaluate approaches: **ongoing maintenance.** As will be seen, there are often trade-offs to consider between the upfront level of effort and ongoing maintenance work for different privacy solutions; when making decisions that will have ongoing business impact, it's important to do this analysis upfront.

In most cases, we find that a Privacy-as-Code approach offers considerable long-term benefits in the form of time/cost savings, more reliable outcomes, and end-to-end measurability of privacy performance. We'll explore why that may be, as well as the pros and cons of additional approaches, for each step of the pyramid.

Privacy-as-Code:

An approach that treats personal data in such a way that its privacy attributes are explicit and governable at a code level

Approaches To Scaling The Privacy Maturity Pyramid

1	2	3	4
Know My Environment Data Classification & Mapping	**Make Promises** Data Subject Requests & Consent	**Keep Promises** DPIAs & Privacy Checks	**Be Trusted** Privacy by Design

Level 1: Know My Environment
Data Classification & Mapping

An organization may begin their privacy journey with a pain point that lies anywhere on the Privacy Maturity Pyramid - for example, they may say “We need to be able to honor CCPA ‘Do Not Sell Requests”. But they will ultimately realize that **to solve for any privacy pain, an organization first needs to understand its data environment.**

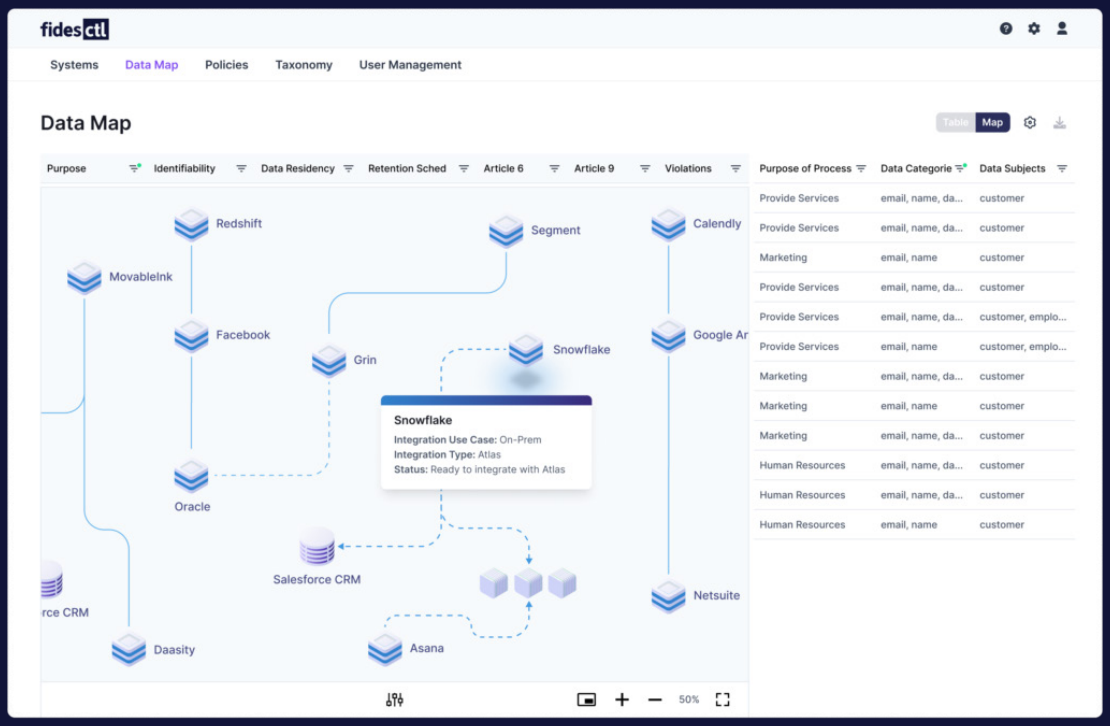
The first level of the Privacy Maturity Pyramid means being able to document - and ***maintain*** - accurate records of what’s going on in data infrastructure. It’s evidenced by answering the following questions

- What types of personal data does the company collect?
- How long does the company retain this personal data?
- Why does the company collect and process this personal data?
- What methods does the company use to process this personal data?
- What other companies receive this personal data?

There may be an explicit legal requirement to document these for regulators, depending on where a business operates. Best known is the European Union’s GDPR Article 30 requirement for a RoPA - a Record of Processing Activities. A RoPA document can look as simple as the CSV file below:

Controller					
Organization Name and contact details		**Data Protection Officer (if applicable)**		**Representative (if applicable)**	
name	Con Troller	**name**	DataPro Tection	**name**	Rep Resentative
address	123 demo street, New York, NY, USA	**address**	123 demo street, New York, NY, USA	**address**	123 demo street, New York, NY, USA
email	controller@demo_company.com	**email**	dpo@demo_company.com	**email**	representative@demo_company.com
phone	+1 555 555 5555	**phone**	+1 555 555 5555	**phone**	+1 555 555 5555

Or it can be turned into a more elaborate data visualization to be sliced, diced, and filtered by different business users like this:



No matter what the visualized deliverable looks like, the process for building it will be the same: a full inventory of the company's data infrastructure to discover and document the PII contained therein.

It sounds like a lot of work, and it can be, but it's important to note that there are tools that can fully automate the process of generating and maintaining a business data map, from identifying systems with PII, to labeling the PII in those systems, to visualizing those systems for business users and legal compliance requirements.

Let's see how companies tackle Level 1 of the Privacy Maturity Pyramid, bearing in mind that their approach will be informed by **the complexity of their data stack** and the **privacy resources in their business.**

Step 1: Identify the systems you own that contain PII

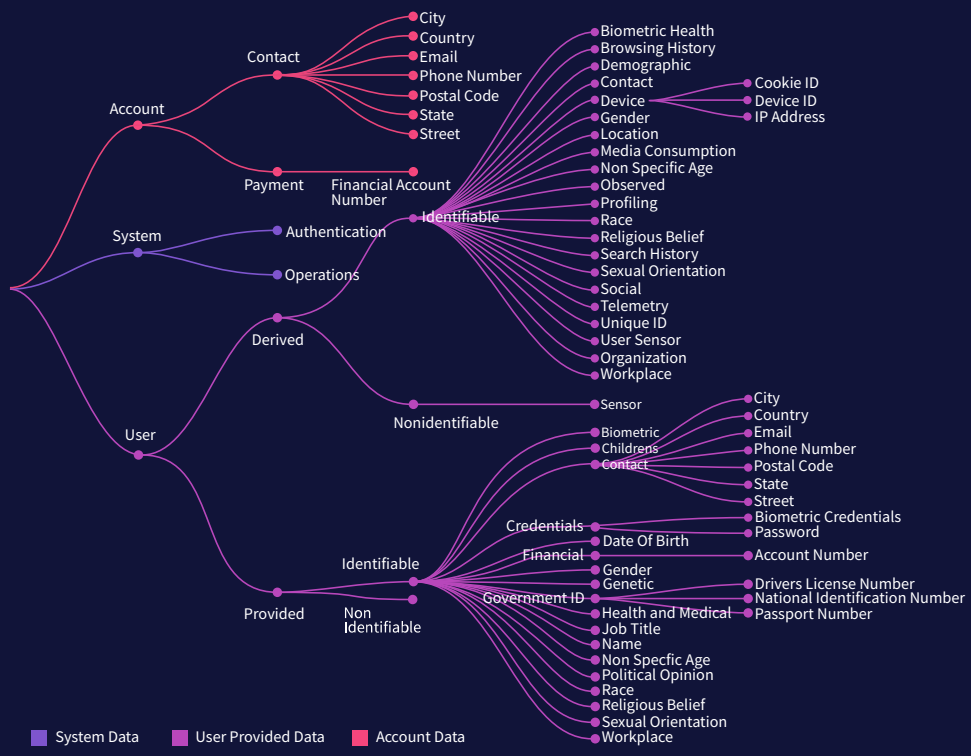
For a small eCommerce company built on a simple SaaS stack, this step won't consume many cycles. But as data infrastructural complexity increases, simply identifying all the systems that contain PII can be very time-consuming. Large enterprises will often have hundreds of legacy databases with under-documented contents. In these cases, automated scanning tools that can surface lists of PII-containing platforms hosted on, say AWS, can save significant time and resources for a business. Scanning tools can also provide assurance that the system cataloging exercise has been exhaustive.

Step 2: Label the types of PII contained in the systems

Labeling PII data types is the next step in building a data map. In plain terms, this means going through all of the data systems identified in Step 1 and assigning labels, or tags, to the different types of PII residing in each one. Think of it as creating a metadata layer that describes the characteristics of data for privacy governance purposes.

In order to complete this exercise, a business will need to agree upfront on a classification system, or taxonomy, for its personal data inventory. Taxonomy work has been an important part of privacy and privacy engineering studies for decades, and established taxonomies can provide a great starting point for this exercise.

Nevertheless, most businesses will need to come up with a bespoke taxonomy for PII labeling that accommodates their unique requirements, and it's important their taxonomy approach is extensible and adaptable to evolving needs. As an example, Ethyca's privacy engineering platform uses an extensible open-source taxonomy to underpin its privacy governance tools..



Now: the labeling process. This entails some version of using metadata labels to tag PII and databases according to their privacy characteristics.

Actually tagging data at scale can be incredibly time-consuming without tools to facilitate the process. The gold standard for this step of data mapping is using Machine Learning classification tools to automate the tagging of data according to an agreed-upon taxonomy. This can eliminate thousands of hours of collaborative work between lawyers and engineers (note that we at Ethyca offer a classifier that integrates with the Fides privacy engineering platform and tags data according to the Fides taxonomy).

Step 3: Collate and display the results for the business user

The last step of building the data map is getting it to a usable form for the stakeholders that need it. At the top of the list are the legal and compliance professionals that may need to produce a RoPA for compliance with laws like GDPR. But the data map is far more than an exercise in legal box-ticking. **An accurate, up-to-date inventory of business data is the foundational step for literally all of the privacy deliverables a business needs to produce.**

Assessing the value of different data mapping approaches

Sharp-eyed readers may have already spotted one of the core challenges inherent in data mapping: business data infrastructure is not static. Quite the opposite. In fact, product teams are shipping updates everyday that add new fields of data collection. Engineers are importing tables into a new database and marketing teams are purchasing a new SaaS tool for email prospecting.

Even after completing steps 1-3 above, if data mapping is approached as a "point-in-time" exercise, there is no guarantee that the data map fully accounts for all personal data processing in the business ***today***. It is a common refrain among privacy professionals that a data map is often out-of-date within days of its completion. With the tech stack constantly evolving, new third-party services or in-house processing activities bring new complexities to data mapping.

Point-in-time data maps can seem sufficient, but they leave legal and engineering silos intact. With this approach, subsequent maintenance difficulties are all but assured.

Manual or automated point-in-time data mapping opens up serious inefficiencies and inconsistencies that could pose privacy risks. Inevitable changes to data infrastructure will render a data map obsolete. The only way to solve these inefficiencies is for the system to actively declare its privacy characteristics synchronously alongside the inventoried personal data. A proactive approach in declaring such privacy characteristics produces a dynamic, "always-on" data map as a ***byproduct*** of an overarching privacy program.

A reactive approach to data mapping cannot account for the next set of privacy needs a business will face, which is more sophisticated requirements around risk in specific data processing. For instance, a company may need to evaluate whether adding a new third-party app will keep the company in compliance with GDPR. Beyond anticipating internal compliance needs, this kind of risk evaluation might be a legal obligation, since laws like GDPR require companies to assess the impact of certain processing activities.

A Privacy-as-Code approach, on the other hand, produces a data map practically as a byproduct of a more sophisticated overarching privacy program. In actively addressing questions of risk that reactive approaches struggle to capture, proactive Privacy-as-Code tools can preemptively build a comprehensive inventory of the needed information for a data map.

Level 2: Make Promises
Data Subject Requests & Consent

A solution for honoring privacy rights — data access requests, data erasure requests, "Do Not Sell" requests, and enforcing consent — will often be a business's introduction to the technical challenges posed by privacy compliance. This challenge is an entryway into the Privacy Maturity Pyramid, but as previously discussed, it's not the foundational aspect of building a privacy program - that's the previous ***Level 1: Know My Environment*** step we discussed.

Nevertheless, organizations first index on being able to Make Promises to users, and the reason is simple: across diverse regulations like the European Union's GDPR, China's PIPL, and more, one of the common threads is a suite of rights granted to individuals. In other words, businesses need to be able to make promises to users about their data. For citizens in jurisdictions with comprehensive privacy regulations, privacy rights will typically include terms like:

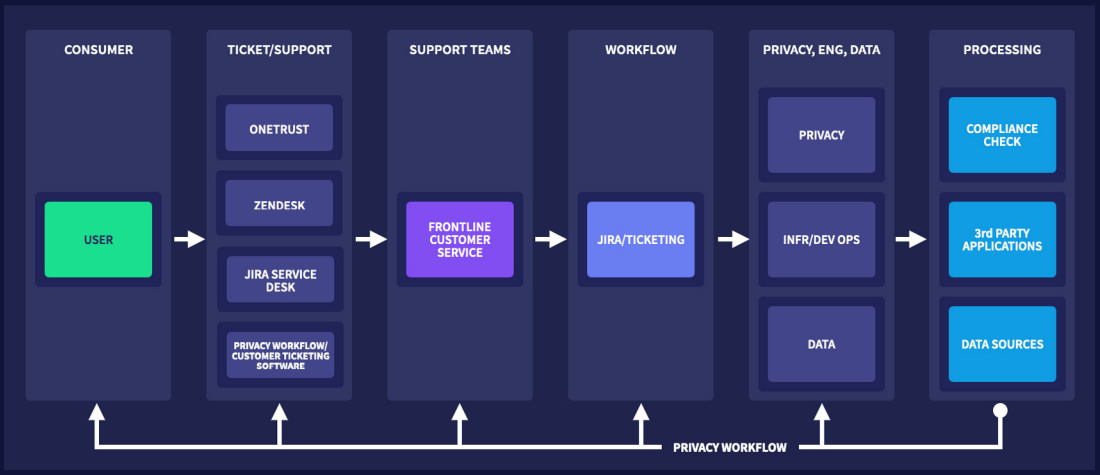
- An individual's right to request a copy of the personal data that a company holds on them, often referred to as an access request.
- An individual's right to request that a company erases the personal data that the company holds on them, often referred to as an erasure request or Right To Be Forgotten (RTF).
- An individual's right to withhold consent from particular uses of their personal data such as data sales, often referred to as "Do Not Sell My Personal Information" in the context of California privacy law.

These data subject requests, or DSRs, are among the most visible aspects of a modern privacy operation, and understandably so. They are inherently user-facing responsibilities, and the proper fulfillment of those users' privacy requests can represent a significant demand on business resources.

Fulfilling a DSR involves finding the relevant data and applying the relevant operation to the requesting individual's data: returning a copy of it, erasing it, or withholding it from downstream data operations per a consent request. Let's explore some approaches to solving Level 2 privacy challenges and evaluate their relative merits.

Approach 1: Ticketing workflows

In the crudest version of privacy rights fulfillment, a request for access or erasure is sent to an intake email address. Upon receiving the request, a privacy program manager or the customer success team files tickets with various system owners to retrieve PII related to the requestor and compile or erase it from the system. Scaling this setup in any meaningful way is a huge challenge (but there are privacy vendors that offer this approach as a core part of their products). That workflow looks something like the this:



Approach 2: API-level Integration

This approach is increasingly popular and viable for businesses using a wealth of third-party SaaS tools in the tech stack. Privacy tools can automate the retrieval and/or erasure of data from those SaaS tools using API calls and knowledge of the underlying data structures of the third-party application. While this offers huge time savings over a ticketing workflow, there are common challenges to scaling this approach across a business of even modest complexity. We'll explore them below.

Approach 3: Privacy-as-Code

A Privacy-as-Code approach yields a data infrastructure in which privacy attributes of personal data are explicit and governable in the code environment. A business employing Privacy-as-Code for data subject rights can build fine-grained, complex tools for fulfilling requests according to custom business needs, or use off-the-shelf DSR tooling that automates requests via the privacy metadata layer. Privacy-as-Code also provides fail-safe mechanisms for propagating user consent preferences throughout data platforms that would otherwise need to be manually updated by respective system owners.

Assessing the value of different privacy rights fulfillment approaches

Ticketing workflows can certainly be used to manage users' data rights, but they present a set of challenges that are not easily solved without embedding privacy systems deeper into technical infrastructure.

For instance, ticketing workflows that assign fulfillment of these requests to individual system owners performing bespoke queries are common. They also consume a lot of time. This implementation might involve hours of an engineer backtracking to find a single user's records for an erasure request, and then more time spent meticulously erasing this user's data to preserve referential integrity between databases. Finally, a privacy manager in the GRC team might need to document and deliver proof of the fulfilled erasure request to the requester. A UK survey found that fulfilling a single DSR can take an average of 66 hours. In this process, the company can only assume that the manual process covered all of the needed requirements for an erasure request.

A DFIN survey found that fulfilling a single DSR can take

on average.

API-integrated automated solutions to DSRs have taken hold among privacy tech vendors and businesses, streamlining DSR fulfillment while still treating privacy as something to be addressed 'outside' of the tech stack.

Most commonly, these solutions are cloud SaaS applications that rely on an understanding of the data architecture of a business's third-party software, and the ability to orchestrate data operations in that software through API calls. There are undoubtedly significant efficiency benefits offered by this approach versus the purely manual approach previously described. But businesses quickly run up against design limitations for this solution.

Most importantly, businesses of any significant size do not rely solely on third-party platforms with APIs in their data infrastructure.

For business storing PII in proprietary databases, i.e., data living in MongoDB, MySQL, Redshift, etc, there is no pre-set schema or API to describe data structures in a way an off-the-shelf privacy tool can read. Therefore, there's no easy way to automate rules for how governance can be enforced on the data.

In the case of cloud SaaS privacy vendors, third-party integrations, legal treatments of PII types, and erasure processes delivered by these solutions are all black boxes — privacy program managers and their teams have no control over these important levers.

This makes privacy change management — in the form of new platforms, new privacy regulations, or new business needs around how data rights should be enforced — challenging and costly on an ongoing basis.

A Privacy-as-Code approach to fulfilling DSRs showcases the following advantages that are consistent with the principles of Privacy by Design.

Proactive not Reactive; Preventative not Remedial

Privacy as the Default Setting

Privacy Embedded into Design

In other words, whenever any sort of data platform—owned or third-party—is added to business infrastructure, it proactively describes the privacy characteristics of its data in a way that businesses can easily standardize and automate fulfillment of privacy rights in that system. This 'bottom-up' approach will lead to significantly better, more durable outcomes when compared to 'top-down' approaches like ticketing workflows or API calls.

Levels 1 & 2 of the Privacy Maturity Pyramid are tightly connected

Let's zoom out for a wider look at the Privacy Maturity Pyramid and explore the tight relationship between Levels 1 and 2. Companies will inevitably realize that buying an automated solution for DSRs is functionally useless without a comprehensive understanding of all the data the business holds and where it lives. In other words, it's impossible to prove compliance with user data rights without a comprehensive, accurate record of the entire business' data inventory (as described in ***Level 1: Know Your Environment***).

A business thus typically realizes that it needs tight links between the foundational element of its privacy program —a dynamic, comprehensive data map —and the applications it uses to uphold users' privacy rights. Otherwise, privacy rights fulfillment programs are constantly playing catch up with a data stack that is evolving underneath their feet. Privacy program managers and their teams will feel the ongoing and significant pain of this breakdown.

Level 3: Keep Promises
DPIAs & Privacy Checks

If a business has satisfactorily tackled levels 1 and 2 of the Privacy Maturity Pyramid, they deserve real credit. Solid processes for knowing the data environment and making promises to users in the form of a dynamic, comprehensive data map and reliable automated processes for privacy rights fulfillment, demonstrate real privacy maturity—particularly for large, complex tech stacks.

In other words, if you're early on in the journey of building a privacy program and want to understand how to tackle the must-haves, we recommend you stop here. We'll be waiting for you when you're ready for Level 3!

Assuming the first two levels are in place, a business will next encounter level 3 of the Privacy Maturity Pyramid. That is, they will seek the ability to Keep Promises to users on an ongoing basis. What does this mean practically? It means that a business needs to be able to honor commitments to user privacy rights over time as tech stacks evolve, new products are shipped, and new privacy regulations emerge.

The privacy deliverables that demonstrate the ability to Keep Promises are most often Data Protection Impact Assessments, Privacy Code Reviews, and Auditable Reporting of privacy activities. These are complex deliverables to produce efficiently at scale, and in 2022, we can confidently say that few businesses have attained this level of privacy sophistication. But the capability should be on the radar of anyone seeking to build a roadmap towards better privacy outcomes; as it's fast becoming table stakes for global businesses. Let's look at what goes into Keeping Promises to users.

Data Protection Impact Assessments

Numerous privacy regulations worldwide require companies to conduct data protection impact assessments, or DPIAs, prior to processing personal data under certain circumstances. For example, if a company wishes to process sensitive personal information, the company might be required by law to first evaluate and document foreseeable privacy risks to the individuals whose data is being processed. The process for conducting a DPIA might involve questions like:

- What personal data will this technology process?
- What technical measures are in place to uphold privacy and security?
- What privacy responsibilities do we have to our existing customers if we implement this new technology?

Inadequate risk evaluation has led to GDPR fines of more than **$3M**

Risk evaluation also factors into in-house processes. Prior to adopting a new technology, such as a third-party chatbot for customer service or an in-house analytics process, a privacy-minded company reviews the product for privacy concerns. In doing so, the company can avoid costly legal fines and reputational damage that would come with implementing new technology that violates privacy requirements.

Practical risk evaluation with Privacy-as-Code

Let's imagine a company that only considers privacy after its product is launched. This company reactively implements a third-party service to automate DSR fulfillment and data mapping, completing the legal box-checking requirements without embedding privacy into product development. Suppose this company is considering a new tool for email marketing. The company's product counsel advises that the company conduct a risk evaluation. In order to evaluate risk prior to adopting the tool, the company needs to know its existing regulatory responsibilities and the technical privacy needs. The company assigns a team of engineers to fill out a detailed legal report with regulatory compliance on the line and pressing deadlines around a product release. Under this pressure, the engineers manage a successful and compliant risk evaluation. But this demanding process recurs, and the backtracking is not sustainable.

Diligently assessing privacy risk at scale on an ongoing basis results from building risk assessment as a feature of the technical infrastructure. This might look like engineers adding a descriptive layer to each resource in the data infrastructure, with details such as whether the resource processes data classified as sensitive, contains children's data, or runs automated decision-making. This low-friction addition to existing infrastructure makes risk evaluation iterable, scalable, and sustainable.

Privacy code reviews

GRC and engineering teams often speak different languages when it comes to the shared responsibility of privacy. One group is well-versed in new regulatory requirements, the other in technicalities of data flows. The overlap is limited, and this spells serious inefficiencies, particularly when it comes to privacy code reviews. This is where the rubber meets the road, where code must be shown to actually align with stated business policies on data use.

We commonly hear the practice of legal privacy code review described as a 'nightmare' by both compliance and engineering stakeholders. As an example, it may involve a quarterly or bi-annual gathering of these two siloes to review product updates in granular detail and check them against the business' stated data policies. We often hear how these meetings result in the always-jarring realization that the code in production is processing data in a way that contradicts the

company's stated policies, and has been doing so for some time. The lack of synchronous data policy enforcement capabilities in technical infrastructure is a persistent pain point, even for businesses that have mastered all the lower tiers of the Pyramid Maturity Pyramid.

To stay on top of the constantly evolving regulatory landscape, a team of GRC specialists and engineers might ask: ***How can we translate legal requirements directly into technical guardrails in the codebase?*** For instance, the legal team might need to modify the privacy policy, requiring an update to what engineering divisions can access a certain subset of personal information. To effectively translate a policy into codebase guardrails, the company needs a standard language for describing privacy and data processing activities. With this language, a policy update can be codified into a series of technical requirements enforceable in code reviews.

In addition to describing policies in the codebase, granular privacy policy enforcement calls for a layer of description in data processing activities. This description might look like a summary of the personal data categories, data subjects, purposes for processing, and degree of identifiability associated with data in a given database. Because this description uses the same language as the codified policy, code can then be checked against policy using clear standards. When the policy evolves, the updated check will identify any instances of data processing that must be realigned.

This fine-grained privacy policy enforcement in the codebase is only possible with a true Privacy-as-Code approach. If a team can translate a policy into a series of access control requirements in the codebase and describe the privacy behaviors of existing resources, then the team must already be implementing a suite of code-level privacy considerations—which is the essence of Privacy-as-Code.

Auditing and reporting

Throughout each of these problem spaces—user data rights, data mapping, risk evaluation, and semantic policy enforcement—a company must maintain a comprehensive log of activity. In addition to legal reporting requirements, such as under Articles 30 and 35 of GDPR, thorough documentation keeps a company informed of internal privacy needs and opportunities for improvement. An accurate reporting mechanism is an important component of showing compliance rather than simply telling it.

We identified how certain approaches could satisfy the challenges of fulfilling user data rights and data mapping with important limitations. The auditing/reporting component of each of these problem spaces can also be satisfied with those manual or semi-automated approaches. But again, it leaves the important limitations unaddressed. Most significantly, without a true Privacy-as-Code approach, efforts to report on privacy will inherently be reactive and out-of-date from the moment they are generated.

The current process for DPIA creation in large organizations speaks to this: yes, GDPR compliant businesses are performing manual impact assessments that log reviews of major data processing updates, and it's hugely time-consuming and inefficient. To review every data processing activity update that takes place in a large business through manual, reactive means is simply untenable. Implementing Privacy-as-Code is the only method for operating an efficient, watertight risk evaluation reporting system at scale.

Level 4: Be Trusted
Privacy by Design

As a business grows, it will almost certainly encounter each of the four privacy problem spaces described in this section. Certain approaches like ticketing workflows or API-based automation offer a bandage that might seem sufficient in the early stages of a privacy program. But reactive privacy—even with automated tools—cannot deliver on the higher-order privacy needs like risk/privacy evaluation and semantic policy enforcement. At best, it serves as a point-in-time solution for mapping data and honoring subject rights. Ultimately, the end-state goal of a business tech stack is true Privacy by Design.

Privacy by Design has been viewed as the privacy-optimizing approach to systems design since it was introduced by Information and Privacy Commissioner of Ontario, Dr. Ann Cavoukian, in 1995. Years later, when Europe drafted the GDPR in the 2010s, Privacy by Design was enshrined as a core principle. The use cases for Privacy by Design are thus tied closely to a business's need to build a privacy operation that ensures ongoing regulatory compliance. Per Cavoukian's framework, a system that features Privacy by Design adheres to the following principles :

- Proactive not reactive
- Privacy as the default setting
- Privacy embedded into design
- Privacy as a positive-sum opportunity rather than zero-sum tradeoff
- End-to-end security
- Visibility and transparency
- Respect for user privacy

Q1: What value is yielded by true Privacy by Design?

There are no silver bullets to privacy, but practicing Privacy by Design is one of the wisest moves a team can make in their efforts to achieve compliance, avoiding fines, and earning customers' trust. With Privacy by Design implemented, teams across GRC and engineering can enjoy immense time and labor savings by not having to backtrack and patch up a product that's already been deployed and processing real-time PII.

Privacy by Design brings significant benefits to a company's bottom line. When data systems are designed to robustly account for the PII they store, fulfilling an access request goes from being a 66-hour task to a 17-second one. Proactive privacy unlocks time for all privacy stakeholders to focus on compliance around their highest risk activities, and engineers can look forward to new innovations instead of backtracking.

Q2: Why is Privacy by Design so hard to achieve?

If you've made it to this section of the document, you'll understand that the vast majority of today's data infrastructure was not designed with privacy in mind. As a result, the status quo is retroactive, box-checking privacy. It can seem like an uphill fight to implement Privacy by Design, especially for companies that have already gone to market. But Privacy by Design is possible—teams just need technical tools fit for the task. That's where the Privacy-as-Code approach we have used comes into focus.

Q3: How can Privacy-as-Code enable Privacy by Design?

Let's return to our working definition of Privacy-as-Code:

An approach that treats personal data in such a way that its privacy attributes are explicit and governable within the code environment

The link between Privacy-as-Code approaches and the principles of Privacy by Design is clear. Code is the fundamental building block of the modern business technology stack, so the ability to impose privacy governance at the code level is the only way to deliver on many, if not all, of Privacy by Design principles.

At Ethyca, we build and maintain Fides, a privacy engineering platform that saves teams thousands of hours on privacy tasks each year while delivering superior outcomes thanks to its Privacy-as-Code approach.

Teams using Fides get better privacy outcomes faster thanks to:

- First-Class Experience: powerful monitoring, administration, and reporting tools for business users.
- Privacy Intelligence: premium tools to automate complex engineering tasks such as data mapping, classification, and de-identification.
- Reliable Connection: a deep catalog of integrations with third-party providers that just work, guaranteed.

If you're interested in using the power of Privacy-as-Code to speedrun your way up the Privacy Maturity Period, go check out our open-source Fides repos on Github, our get in touch with our solutions team to talk about how we can help with your particular privacy use case.

More Resources For Leveling Up The Privacy Maturity Pyramid

Privacy is inherently cross-functional, which means a wide array of stakeholders need to speak the same language.

Moving any organization up the Privacy Maturity Pyramid requires finding alignment and plenty of cross-functional education.

After reading the first sections of this document, you might be finding yourself with more questions than you started with—questions like "How do I start a conversation with my engineering leaders about privacy?" or "Where can I get more information about Privacy-as-Code that's accessible to non-technical colleagues?" We have curated this list of resources with those questions in mind.

You know the privacy needs and challenges of your business better than most anybody, so you'll know which resources would resonate with which internal stakeholders. All of this content is free and ungated. Furthermore if your team wants to reach out to leaders working at the forefront of privacy technology, you're welcome to join the Slack workspace for the Fides privacy engineering community.

For you

Having gotten familiar with the concepts in this document, you're ready to take the next steps.

- Take a high-level look at the structural problems of privacy, as our CEO Cillian makes the case that privacy belongs within the Software Development Life Cycle.
- Learn the ropes of privacy engineering, with this general introduction to the field.
- Understand what is involved in translating a privacy policy into a codebase restriction, using this step-by-step guide for non-developers.

Share these resources for your team

As a privacy champion, you can share these resources with colleagues who are not yet familiar with a developer-centric approach to privacy.

- Get to know Privacy-as-Code with the International Association of Privacy Professionals, in this profile on Fides.
- Explore this visualization of the Fides taxonomy of personal data, which codifies privacy behaviors into specific categories conducive to technical implementation.
- Watch our team walk through important business use cases for the Fides devtools, achieving compliance while remaining innovative and adaptable.

Share these resources for your engineers

For the individuals who build and maintain the processes that handle personal data in your organization. The resources below will introduce them to devtools that shift privacy upstream and reduce friction.

- See our Engineering Manager Thomas La Piana give a demonstration of Fides, automatically conducting privacy checks on code.
- Check out the open source code for the Fides devtools, and clone the repository.
- Try out the Fides devtools using this hands-on tutorial.

We wish you the best of luck as you level your business up the Privacy Maturity Pyramid!

License Apache 2 License CC BY 4.0

The Privacy Engineering Platform

Join the Fides community.
Let's fix trust in tech together.

ethyca.com/fides