

## Playbook: Government as a Platform

Richard Pope  
*Senior Fellow, Government Platform*  
*Ash Center for Democratic Governance and Innovation*

---

November 2019



## **Playbook: Government as a Platform**

---

Richard Pope  
*Senior Fellow, Government Platform*  
*Ash Center for Democratic Governance and Innovation*

---

November 2019



## **ABOUT THE AUTHOR**

Richard Pope (@richardjpoppe) was part of the founding team at the UK Government Digital Service, working as a product manager for the first versions of GOV.UK, which went on to win the Design of the Year award in 2013, and co-authoring the Digital by Default Service Standard, which all UK government digital services must meet. He worked with policy and delivery teams across the UK government redesigning government services in policy areas as diverse as welfare, land registration, and employment.

He was involved in the early days of the UK civic tech scene and worked at various start-ups as a technologist and designer.

## **ABOUT THE ASH CENTER**

The Roy and Lila Ash Center for Democratic Governance and Innovation advances excellence and innovation in governance and public policy through research, education, and public discussion. The Ford Foundation is a founding donor of the Center. Three major programs support the Center’s mission: The Program on Democratic Governance, the Innovations in Government Program, and the Rajawali Foundation Institute for Asia. This research paper is one in a series published by the Ash Center for Democratic Governance and Innovation at Harvard University’s John F. Kennedy School of Government. The views expressed in the Ash Center Policy Briefs Series are those of the author(s) and do not necessarily reflect those of the John F. Kennedy School of Government or of Harvard University. The papers in this series are intended to elicit feedback and to encourage debate on important public policy challenges.

This paper is copyrighted by the author(s). It cannot be reproduced without permission. Per the Ash Center’s Open Access Policy, this paper is available for download to the public free of charge at [ash.harvard.edu](http://ash.harvard.edu).

## **ACKNOWLEDGEMENTS**

This playbook was written as part of a fellowship at the Harvard Kennedy School digitalHKS initiative.

Thanks to the following people for their feedback, contributions and help:

Adam Conners, Amy Hupe, Andrew Greenway, Ben Mcguire, Ben Welby, Daniel Abadie, David Eaves, James Darling, Mairi Robertson, Matt Sheret, Nicola Barker, Pete Herlihy, Pia Andrews, Sam Smith, Sam Smith, Sarah John, Steve Marshall, Trilly Chatterjee, Vanessa Rhinesmith, Will Myddleton, Zara Rahman and many others who took the time for a quick chat or suggested links to include.

This paper is copyrighted by the author(s). It cannot be reproduced or reused without permission.

## CONTENTS

<b>Introduction</b>	<b>1</b>
<b>Definitions and Terminology</b>	<b>5</b>
<b>Users</b>	<b>7</b>
<b>Identifying Platforms</b>	<b>11</b>
<b>Designing for Self-Service</b>	<b>16</b>
<b>Data Infrastructure, APIs and Open Standards</b>	<b>19</b>
<b>Working across Silos</b>	<b>22</b>
<b>Reuse</b>	<b>25</b>
<b>Funding and Operations</b>	<b>27</b>
<b>Adoption</b>	<b>30</b>
<b>Designing Services on Top of Platforms</b>	<b>32</b>
<b>Identity, Trust and Consent</b>	<b>40</b>
<b>Explaining Government as a Platform</b>	<b>50</b>

## **INTRODUCTION**

If you walked into a government office, anywhere on the planet in the 20th Century, you would probably have found three things: a records department managing paper files, index cards or microfiche; administrative staff processing and validating information, either in person over-the-counter or in back-offices, executing the rules set out in different government policies; finally, you'd find forms. Lots and lots of forms.

On the outside of the building, you'd find a sign that contains a word like "driving," "passport" or "tax." That's because most government departments were, and still are, thematically organized, with responsibility for everything from the top to the bottom of an area of government.

As governments began to put more of their services online, this was the world they digitized. So, today, if you were to look at the technology stack of a typical government agency—let's say a motoring agency with responsibility for issuing driving and vehicle licences—you'd see something like this:

- A series of online transactions—apply for a driving licence, renew a licence, report a change of address—that would more or less map directly to the paper forms they replaced.
- There would be admin systems for things like managing payments or validating addresses—things that humans used to do manually, but these days are semi-automated. The business rules once written out in a policy document have been turned into code.
- Finally, there would be a series of databases—for our hypothetical driving agency, that might be a database of drivers and their cars, and some sort of case-management system for keeping track of applications. These databases have (mostly) replaced the records departments and paper files that once took over whole floors of government buildings.

This is a pattern that is replicated across government—the pattern repeated in different agencies and at different layers of government. So, if you were to look at land registration, social security, or tax departments, in central government or local government, you'd probably see a similar pattern.

### **Digitizing a World of Paper**

You might be asking, why is this a problem? Surely making things work more efficiently is a good thing? There are several reasons why not.

When it comes to using data from one part of government in another part of government—maybe someone wants to be able to prove they own a car to the tax office—the prevailing practice is what is referred to euphemistically as “data sharing.” It’s referred to as “sharing,” but it’s actually closer to copying-and-pasting. Records in databases are not like pieces of paper in a file, so “data sharing” often means multiple copies of the same data end up scattered across government, which then has implications for accuracy, security and privacy.

As well as duplication of *data* there is duplication of *effort* too. Remember, our hypothetical driving agency? It probably has a payment processing system so people could pay for their driving licences. But other agencies also need to process payments, so public money ends up being spent on the multiple systems, all doing the same thing. The same is true for other commons tasks too, things like messaging or printing.

Thirdly (and most important), is the impact on the design of services that the public have to use. In computer science, there’s an adage called “Conway’s law.”<sup>1</sup> It is named after computer programmer Melvin Conway, and it says that:

organizations which design systems . . . are constrained to produce designs which are copies of the communication structures of these organizations.<sup>2</sup>

Government is this idea writ large—with services designed around the organizational structure of government rather than the needs of people. This means a family looking to move house, someone recovering from an illness or someone looking to start a company, will have to deal separately with many government agencies. Some people fall through the gaps, and you have to ask: what that does to those people’s trust in the state?

It also leads to a monopoly of provision—a one-to-one mapping between department and services, which often means it’s impossible for other parts of government,

.....  
1 Melvin E. Conway, “How do Committees Invent?,” *Datamation*, 14, no. 5 (1968):28–31.

2 Conway, “How do Committees Invent?,” 28–311.

or companies and charities, to build complementary services. That's because the data and the business logic are locked away inside these organizations.

### **Platforms Are Changing Governments**

Looking around the world, we can see a different approach emerging. One of cross-government platforms that are beginning to break down organizational silos, save money and change the types of services that can be delivered to the public.

For example, in the United States, the US Veterans Administration has begun to build its services around APIs. These allow veterans charities and other organizations (not just government) to build public-facing services.<sup>3</sup> This work is part of a federated data effort across central and state governments to build APIs and exchange data to agreed formats.<sup>4</sup>

In India, the Aadhaar identity platform has started to link together services from across different parts of government for a billion people. In the process, it has created a national debate about privacy and digital rights.<sup>5</sup> Aadhaar is part of IndiaStack, which the government refers to as “a set of APIs that allows governments, businesses, startups and developers to utilize an unique digital infrastructure to solve India's hard problems toward presence-less, paperless, and cashless service delivery.”<sup>6</sup>

Estonia is starting to design new services around “life events,” like having a child, that transcend the boundaries of government agencies. They can do this because of common cross-government data infrastructure they have developed over the past decade.<sup>7</sup> And because neighbouring countries have adopted that infrastructure, it holds out the promise of simpler trade, and citizens accessing services across borders.<sup>8</sup>

.....  
3 US Department of Veterans Affairs, “Put VA Data to Work,” <https://developer.va.gov>. Retrieved 5th June 2019.  
4 Julia Lindpaintner, “The US Data Federation wants to make it easier to collect, combine, and exchange data across government,” 5th March 2019, <https://18f.gsa.gov/2019/03/05/the-us-data-federation/>.  
5 UIDAI, Addhaar Dashboard, [https://uidai.gov.in/aadhaar\\_dashboard/index.php](https://uidai.gov.in/aadhaar_dashboard/index.php). Retrieved 5th June 2019.  
6 “What is India Stack?,” <https://indiastack.org/about/>. Retrieved 5th June 2019.  
7 Sam Trendall, “We have only scratched the surface’—Estonia's CIO on what's next for the world's most celebrated digital nation,” PublicTechnology.net, 18th February 2019, <https://publictechnology.net/articles/features/'we-have-only-scratched-surface'---estonia's-cio-what's-next-world's-most>.  
8 Finnish Patent “Estonian Ministry of Justice: The Ministry of Justice entered into an agreement with Finland for the exchange of business register data,” [https://www.prh.fi/en/uutislistaus/2019/P\\_17654.html](https://www.prh.fi/en/uutislistaus/2019/P_17654.html).



In the UK, Italy, and Argentina shared components solve common problems for the whole of government. For example: sending a text message, hosting a web application or taking a payment. These components are run centrally, rather than by individual agencies, helping civil servants to deliver new services faster and creating more consistent experiences for citizens.<sup>9 10 11</sup>

### **Government as a Platform**

Taken together, what we are seeing is the realization of the world sketched out by Tim O’Reilly in his 2011 article “Government as a Platform,” where he posed the question: what if government was organized more like an operating system?<sup>12</sup>

Government as a Platform holds out the promise of radically better services for the public. And to do so in a way that makes it simpler and faster for both civil servants and politicians, the private sector and non-profits, to meet people’s needs. A world of government reorganized around shared components, APIs, standards and canonical datasets.

### **Who This Playbook Is For**

Around the world, digital service units along the lines of the UK’s Government Digital Service, Italy’s Team per la Trasformazione Digitale, 18F, and the Canadian Digital Service are springing up. Teams like these have brought modern digital ways of working into the heart of government, adopting agile software development, user centered design, and the use of cloud hosting and open-source technology.

Along with new practices, these organizations have brought new professions into government:<sup>13</sup> product managers, software engineers, user researchers, service designers, data scientists and chief digital officers. These people are the intended audience of this playbook: practitioners looking for approaches to implementing platforms in government.

.....  
9 HM Government, “Design and build government services,” <https://www.gov.uk/service-toolkit#gov-uk-services>. Retrieved 30th April 2019.

10 Team Digitale, “Projects,” <https://teamdigitale.governo.it/en/projects.htm>. Retrieved 30th April 2019.

11 Mike Bracken, “Argentina just made driving licences digital,” *Public Digital Blog*, 12th February 2019, <https://public.digital/2019/02/12/argentina-just-made-driving-licences-digital/>.

12 Tim O’Reilly, “Government as a Platform,” *Innovations*, 6, no. 1 (2011):13–40.

13 Or in some instances re-introduced capabilities long-since outsourced.

As such, there is some level of minimum capability that an organization needs before it starts trying to implement many of the examples listed in this playbook. If you are working in an organization that is just starting on its digital transformation journey, then probably the best thing to do is focus on building an effective user centred design capability first.

Even for more digitally mature organizations, attempting to do everything listed here at once is probably a bad idea. Every organization will have strategic decisions to make about where to start. Please view this as a starting point rather than a detailed journey map.

Wherever possible, it includes footnotes and links to further reading and quotes from practitioners.

## **DEFINITIONS AND TERMINOLOGY**

In any multidisciplinary field, definitions and terminology have a tendency to create trip-hazards. This is definitely the case in “platform land.” A *service* may mean something different to a *service designer* than it does to a *software engineer*; for some a *platform* carries with it assumptions about multi-sided market places, for others, it’s just a component to be built upon.

In this playbook, the following definitions are used:

### **Services**

Public-facing services allow citizens or their representatives to achieve a desired outcome. With appropriate governance, services can be provided by any layer of government, and by commercial or third sector organizations. Services are the things that are built on top of platforms.

### **Digital proofs**

The term *digital proofs* is used to cover the various different ways that someone can prove a fact (for example that they are entitled to a free prescription). This includes

cryptographically signed data, and one-use codes or barcodes that can be used to verify a fact online.

### **Government as a Platform**

The term “Government as a Platform” is used to refer to the whole ecosystem of shared APIs and components, open-standards and canonical datasets, as well as the services built on top of them and governance processes that (hopefully) keep the wider system safe and accountable.

### **Shared APIs**

Shared APIs expose the business logic of government, for example, calculating a benefit payment or checking the status of an application. They are referred to as “shared” to indicate that they are available for use beyond the agency that developed them.

### **Platforms**

The term *platform* is used in the generic sense to refer to any shared API, shared component, trust and identity system, or register.

### **Registers**

Registers are canonical datasets that are used and trusted across government and beyond. For example, a single list of postal addresses or a single list of registered companies. They expose the data via APIs to agreed open-standards and have appropriate governance and ownership in place.

### **Shared components**

Shared components solve common problems for the whole of government (and potentially beyond). For example: sending a text message, hosting a web application, taking a payment or signing a document.

### **Trust and identity systems**

Trust and identity systems ensure that data is only accessed for appropriate purposes, and that use is understandable and trusted by citizens or their representatives. For example: authentication, trust verification, data exchange and key management systems.

## **USERS**

It is one of the tenets of user centred design that services should be designed around the needs of users. Platforms are building blocks, used to create many different public-facing services, so the primary users of a platform tend not to be the public themselves (the end-users of services), but rather the people and organizations who use them to *create services* for the public. A platform might have users across different tiers of government and non-government organizations.

### **1. Think of platforms as part of many services**

If successful, a platform will be part of many public facing services, for example, an API for validating and geocoding addresses might be used in services that let people tax their car, find a hospital, or apply for benefits. As such, a team developing a platform may have only a limited relationship with the end-users of those services, and the public will experience the platform as part of a wider service, not in isolation.

This “one-to-many” nature of platforms also means that any changes you make to them will affect many different types of user and many different services. This comes with a responsibility to understand the likely impacts of any changes you make and to work in an open way. The teams using your platform need to be able to anticipate changes early and ensure they work for users of their service.

Conduct joint user research with teams using your platform, so that you can understand how your platform manifests itself in different services. Work in the open and maintain good communications with your users, so you understand how they are using the platform and ensure they understand any upcoming changes.

## 2. Understand *who* your users are

Because shared components, APIs, registers and identity systems are building blocks of public-facing services, it's important to understand the users of a platform are often not the public (at least not directly). Designing a good platform will require an understanding of their needs, but first you need to identify who they might be. Some important user types to understand include:

- **Developers** wanting to integrate with your platforms quickly and simply.
- **Designers** of a public-facing service who may need to understand how your platform will fit into the broader design of a service, or include it in prototypes they are testing.
- **Procurement managers** who need to understand any contractual or financial arrangements.
- **Administrators** who need to manage the use of the platform, view usage reports or changing their settings.
- **The team developing the platform** who will need to understand how other users are using the platform.
- **The general public** (and their proxies, such as journalists, charities or elected representatives) who want to understand how you are spending their money and managing public data.
- **Politicians or senior officials** who might have statutory responsibility for your platform or the services using it.

You should identify people and organizations who might use your platform, and understand their needs (remembering that they may not work in government at all and may work in non-technical roles such as procurement).

## 3. Understand *where* your users are

Looking around the world, government platforms tend to have users in one of three groups:

1. Limited to a single tier of government—for example used by a single state in a federal system, or only by central government.

2. Government-wide—used across central, municipal and regional government.
3. Society-wide—for example, in addition to use in government, a platform is also used in banking, or by third sector organizations and commercial companies.

To take one example, we can see this pattern if we look at the identity and authentication platforms from the US, UK and India. Login.gov provides a way to add single-sign-on to digital services, but is limited to US federal agencies.<sup>14</sup> GOV.UK Verify provides a similar service (albeit with additional levels of identity verification and multiple identity providers) for central government in the UK, but there have also been trials to use it in local government too.<sup>15, 16</sup> While in India, the Aadhaar identity platform is used well beyond government services for people to open bank accounts (and not without controversy).<sup>17</sup>

It is important not to take away from this that “the broader the usage the better the platform.” There may be very good reasons for focusing on a subset of institutions early on. There may also be circumstances where a single society-wide platform has too many significant risks associated with it. (Identity is indeed one area where a single, centralized system might be undesirable!) The key thing to understand is that this is a strategic decision that can be made, and one that can change over time. Who your users are will change too.

.....  
14 login.gov describes it’s aim as to make “federal benefits, services and applications easier and more secure.” US General Services Administration, “login.gov,” <https://login.gov>. Retrieved 5th June 2019.

15 Joanne Southern, “Guest post: working with GDS on transforming local services,” *GOV.UK Verify Blog*, 22nd March 2017, <https://identityassurance.blog.gov.uk/2017/03/22/guest-post-working-with-gds-on-transforming-local-services/>.

16 Angelica Mari, “London council trials Verify-based digital ID scheme for vulnerable citizens,” *Computer Weekly*, 20th May 2019, <https://www.computerweekly.com/news/252463658/London-council-trials-Verify-based-digital-ID-scheme-for-vulnerable-citizens>.

17 “Banks can use Aadhaar for KYC with customer’s consent: RBI,” *Economic Times*, 29th may 2019, <https://economictimes.indiatimes.com/industry/banking/finance/banking/banks-can-use-aadhaar-for-kyc-with-customers-consent-rbi/articleshow/69568435.cms>.

**Example: Identifying users of Government as a Platform in the UK**

*In the UK, much of the early work of the Government Digital Service had been focused on projects with significant public-facing elements. As the organization began to focus more explicitly on platforms, it had to change how it thought about users:*

Many of us joined the Government Digital Service to work directly on things which matter to users. We realized that, organizationally, our job had shifted to meeting the needs of other civil servants, so that they can meet the needs of their users and, as a consequence of that, public services improve and the quality of people’s experience of the state is transformed.

Citizens are not the only valid users—even GOV.UK itself has a huge number of civil servant users who are using it as a publishing tool.

We needed to start thinking about our users as being service teams around government. Shifting from thinking about making services for citizens to making products for service teams was a tough shift for GDS.

To identify the needs of our users, we did 150 interviews with service teams across government. We identified those teams from service assessments, the GOV.UK performance platform and also the 4,000+ PDF forms that are hosted on GOV.UK. We asked them things like: Tell us about your users? Tell us about the needs you are meeting? Tell us about the needs you can’t meet but would love to meet?

—Ben Welby & Will Myddelton, Government as a Platform programme 2013, 2018 (UK)

## IDENTIFYING PLATFORMS

Platforms solve problems once, meeting the *common* needs of users, rather than addressing the same problem multiple times, and in slightly different ways. Those users might be elsewhere in government, in a different tier of government or outside government altogether (for example, a private company or charity).

Your organization will need to prioritize three things to help it identify where there is a genuine opportunity to do this, and where the opportunity is purely superficial:

1. The capability to identify common needs and duplication.
2. Creating clear routes for “point solutions” from specific products to develop into more general platforms.
3. An understanding of what the right level of abstraction is for a platform (good platforms do one thing well).

### **1. Identify common needs and duplication**

Chances are you are already aware of lots of duplication. It might be multiple services all using different payment systems or address lookups. It might be developers from different teams creating their own login system or hosting setup. In fact, this seems to be the normal state of most organizations developing digital services.

However, just hoping that shared APIs, components and registers will emerge organically from this is unlikely to lead to results. For example, no shared components, APIs or registers emerged from the UK Government Digital Service’s *Exemplar* programme (which saw teams setup across government to redesign digital services).



There is no single approach to doing this, but approaches you might like to consider include:

- Conducting interviews and surveys to understand where common needs might exist.<sup>18</sup>
- Using Wardley Mapping to identify duplication.<sup>19</sup>
- Use governance processes, such as service standard assessments, to collect data about common needs.<sup>20</sup>
- Review legislation and guidance to identify where one service might benefit from API access to another.
- Review code repositories to identify teams working on the same problem.
- Create a service map or user-journey map and use it to identify common needs.<sup>21</sup>
- Create a map of the data ecosystem and use it to identify common needs, and datasets that can be adapted into common registers.<sup>22</sup>

Regardless of approach, all these activities take some time and effort, and will be competing against other priorities. So, make it clear where responsibility to spot common needs and duplication across teams lies.

.....  
18 In the UK, the Government Digital Service conducted 150 interviews with teams across government with the aim of understanding their common needs. It led GDS to understand they should prioritize a form building platform over one that, for example, accepted payments from the public, because more services needed it. Richard Pope, "Interview with Will Myddelton—UK Government as a Platform programme," 29th October 2018, <https://medium.com/platform-land/interview-with-will-myddelton-government-as-a-platform-3aff4ebcb3e8>.

19 Wardley Mapping is an approach to visualising the components required to deliver a service. Simon Wardley, "An introduction to Wardley (Value Chain) Mapping," *Bits or Pieces?* (blog), 2nd February 2015, <https://blog.gardeviance.org/2015/02/an-introduction-to-wardley-value-chain.html>.

20 The Government Digital Service identified potential teams to work with to help identify platforms from "service assessments," which are a step in the governance process for all new digital services. Richard Pope, "Interview with Will Myddelton—UK Government as a Platform programme," 29th October 2018, <https://medium.com/platform-land/interview-with-will-myddelton-government-as-a-platform-3aff4ebcb3e8>.

21 The UK Ministry of Justice created a visual map of all the activities that take place in the justice system, agnostic from the organizations who run them. Mike Bracken, "Mapping new ideas for the digital justice system," *Government Digital Service Blog*, 18th August 2015, <https://gds.blog.gov.uk/2015/08/18/mapping-new-ideas-for-the-digital-justice-system-2/>.

22 The Open Data Institute have developed a process for documenting and mapping data ecosystems. Open Data Institute, "Mapping data ecosystems," 23rd March 2018, <https://theodi.org/article/mapping-data-ecosystems/>.

## 2. Provide a path from products to platforms

A less top-down approach than identifying common needs and duplication is to ensure there are clear routes for “point solutions” from individual products to develop into shared platforms. A point solution is something that has been developed as part of a single service, that might be useful as a more general platform for multiple services.

For example, the authentication platform, Login.gov, started life as a larger project called MyUSA, which aimed to provide a single account and to do list for interacting with government. The authentication part of the project was developed into login.gov by 18F and the United States Digital Service.<sup>23</sup>

Another example is the letter printing feature of the GOV.UK Notify messaging platform. It makes use of the spare printing capacity at the UK driving licence agency’s secure printing facility that was created when printed “tax discs” were abolished.<sup>24</sup>

This approach is similar to that taken by many teams at commercial platform companies, where problems are first solved within product teams, but then supported with staff and funding to turn them into more general solutions.

Within Google, setting up a new team is often off the back of what is already happening organically. Google tends to create a set of core, high-level values while allowing space to innovate. Things that are mandated are really high-level, rather than day-to-day—everything rolls-up to a higher objective.<sup>25</sup>

For commercial services, the “route to market” is potentially clearer—for example, an Amazon platform can ultimately appear in the AWS dashboard alongside the company’s other offerings where developers can use (and pay) for it. For the approach to be a success, teams working in government will need to be supported to generalize their point-solutions into platforms.

.....  
<sup>23</sup> 18F, “18F/myusa,” <https://github.com/18F/myusa>. Retrieved 26th June 2019.

<sup>24</sup> @yahoopete (Pete Herlihy), “One of the coolest things about this, is that we made a deal with the DVLA to use their secure, modern on-site print facilities for our letter fulfilment. Having stopped printing tax discs etc a few years back, they had capacity. And we could maximise the savings to taxpayers.” 16th April 2019, [https://twitter.com/yahoo\\_pete/status/1118245112482598912?s=12](https://twitter.com/yahoo_pete/status/1118245112482598912?s=12).

<sup>25</sup> Interview, Dr. Adam Connors, Senior Google Engineering Manager.

### **3. Identify real-world platforms**

Platforms normally get talked about in terms of shared digital capabilities—things like hosting, identity or payments. As the GOV.UK Notify printing example above and commercial platforms like Grab (a broker for motorbike taxis in South East Asia) and JustPark (a broker for car parking spaces in the UK) show, it is possible to put platform-like capabilities around physical goods and services.

Most governments operate some public-facing offices—places like tax offices or labour exchanges. As with many digital services, these only tend to offer the services related to the part of government that owns them: you can't get help filling out a passport application at a tax office. What if local offices were seen as a shared capability for any government service that needs to talk to people face-to-face? Or a place to take cash payments for people without bank accounts?

Understand where there might be opportunities to build a platform around fragmented, real-world capabilities.

### **4. Get the level of abstraction right**

Developing a collective understanding of what makes a good platform is important. A general rule is that if a system is trying to do multiple things, or is highly customizable, it probably isn't a good platform. Platforms generally do one thing well—for example send a message, check the status of an application, verify a fact or geocode an address—small pieces loosely joined, rather than monoliths that try to handle every edge-case or be too smart.

It's worth noting how many of the platforms on the Amazon Web Services Dashboard start with the word "simple," and generally how the services offered by AWS, Google Cloud and Twilio and other commercial platform providers are focused on a clear task.

Simple platforms are easier for developers to use because it is clear exactly what they do. Simple platforms also create less hard dependencies, making it easier for developers to replace them at a later date.

**Example: Prison visits and the potential for a booking platform, UK**

*In the UK, the Ministry of Justice developed a service called “Visit someone in Prison” which allows someone to book a slot to visit a prisoner. Initially, the team thought there might be a route to developing a more general “booking platform,” but this turned out to be the wrong level of abstraction.*

Good platforms abstract away a clearly-defined problem. We found that “Bookings” weren’t a clearly-defined thing and, typically, their definition was coupled to the nature of the wider process or organization they’re part of.

That means that any “booking platform” has to be able to be configured to handle those organizations’ processes and systems, which could be as simple as meeting room bookings (one person can book a meeting room at a time, and they specify the duration) or as complex as booking prison visits (each prison specifies its visit slots and their capacities, visitors may not be allowed to be present with prisoners who aren’t the person they’re visiting, and more).

Often, the back-ends that handle these bookings are also already tightly integrated into other parts of the organization, so they can’t simply be replaced with a new platform. Not only that, but this complexity also dramatically changes the interface that the user needs to see.

Once we came to realize all of the factors that play into bookings, we concluded a single “booking platform” wasn’t really something that could exist.

—Steve Marshall, Head of Hosting and Digital Operations,  
Ministry of Justice (UK)

## DESIGNING FOR SELF-SERVICE

For a platform to serve all its potential users it needs to be usable without direct support. This means designing a service that actively helps different types of users to “self-serve.” A platform that is designed for self-service is one that users can find, understand what it does and start using without intervention.

### 1. Create landing pages for each platform

You can’t know who all your potential users might be—they might be in another part of your organization, in another government agency or outside of government altogether (for example a startup developing a new product). So, firstly, your users need to be aware of the platform. Then, if they are aware of it, it is likely that many will ask the same questions, things like “Where are you hosted?,” “Is it secure?,” “Is there a Python library?,” “Can I download a CSV of the data?”

Many digital service units are creating public landing pages for their platforms to help users discover them and understand how they work. Australia, Canada and France have public API stores that list APIs that are available to build on.<sup>26, 27, 28</sup> While units in the UK, USA, Italy and India have all created landing pages for their shared components, and the UK has landing pages for each register.<sup>29, 30, 31, 32</sup>

Create a public landing page for each component, API and register that lists key information about the each platform. Ensure that registers are independently discoverable from shared APIs and components. For registers containing open data it should also be easy to identify how to download a bulk dataset from the landing page.

Keep a record of questions that people are regularly asking, such as average uptime, and aim to answer them on the landing page (not hidden elsewhere on an FAQ page).

.....  
26 Australian Government, “api.gov.au,” <https://api.gov.au>. Retrieved 5th June 2019.

27 “api.gouv.fr,” <https://api.gouv.fr>. Retrieved 9th march 2019.

28 Government of Canada, “API Store,” <https://api.canada.ca/en/homepage>. Retrieved 5th June 2019.

29 HM Government, “GOV.UK Service Toolkit,” <https://www.gov.uk/service-toolkit#gov-uk-services>. Retrieved 5th June 2019.

30 18F, “cloud.gov,” <https://cloud.gov>. Retrieved 5th June 2019.

31 Agenzia per ill’Italia digitale, “API—Il catalogo delle API REST italiane a disposizione degli sviluppatori,” <https://developers.italia.it/it/api>. Retrieved 23rd April 2019.

32 HM Government, “GOV.UK Registers,” <https://www.registers.service.gov.uk>. Retrieved 17th June 2019.

## **2. Design good documentation**

Well-written documentation is essential to help your users understand how to use your platform.<sup>33</sup> Github's 2017 Open Source Survey found that incomplete or confusing documentation was the biggest problem encountered by developers.<sup>34</sup>

As with any good content, developing good documentation takes time and an understanding of the needs of your users. The UK Government Digital Service maintain a template for different teams to publish technical documentation, which has users across government.<sup>35</sup> Adoption of that template increased when it was itself, well documented.<sup>36</sup>

Ensure your team has access to technical writers and tests content with users to ensure it continues to meet their needs over time. Use automated tools such as Swagger to help keep documentation up to date.<sup>37, 38</sup>

## **3. Go where your users are**

Developers and designers who are building services using your platforms will have existing tools and workflows. By understanding which package managers, languages and tools they use, you can make it quicker and simpler for them to get started.

For example, the US Web Design System and the Australian Government Design System are published as Node.js packages that can be installed via the NPM package management system.<sup>39 40</sup> This makes them simpler to use and to track updates.

Try and emulate a high-quality open-source vendor. Create and maintain packages in the languages your users use most, and publish code samples online.

.....  
33 HM Government, "How to document APIs," <https://www.gov.uk/guidance/writing-api-reference-documentation>, GOV.UK 15th February 2019.

34 GitHub, "Open Source Survey," <https://opensourcesurvey.org/2017/>. Retrieved 30th June 2019.

35 alphagov/tech-docs-template, "alphagov/tech-docs-template: A template for building technical documentation with a GOV.UK style," <https://github.com/alphagov/tech-docs-template>. Retrieved 30th June 2019.

36 Interview, Rosalie Marshall, Lead Technology Advisor & Head of Data and API Standards, Government Digital Service.

37 "Create Great API Documentation," <https://swagger.io/solutions/api-documentation/>. Retrieved 30th June 2019.

38 HM Government, "Writing API reference documentation," GOV.UK, 15th February 2019, <https://www.gov.uk/guidance/writing-api-reference-documentation#generating-api-reference-from-code>.

39 "uswds (United States Web Design System)," npm, <https://www.npmjs.com/package/uswds>. Retrieved 5th June 2019.

40 "uswds (United States Web Design System)," npm, <https://www.npmjs.com/org/gov.au>. Retrieved 8th June 2019.

#### **4. Let users try it out**

Commercial platforms like Twilio, Google’s Cloud APIs and GoCardless let users try out the capability of their platforms. They do this by either allowing free access at low usage or providing a sandbox environment.

The eSign component, which is part of IndiaStack and allows users to integrate digitally signed documents into their services, includes links to sandboxes as part of their documentation.<sup>41</sup>

Create ways for people to try your platform quickly and without exposing sensitive information (for example, by using fake data). Think carefully before requiring registration or anything else that might unnecessarily slow users down.

#### **5. Understand the needs of procurement and finance**

Designing for self-service is not just about meeting the needs of developers and designers. Working hard to make your platform easy to procure is also important.

The GOV.UK Notify team spent lots of time writing Memorandums of Understanding in plain English. They also calculated that they could offer the platform to many users without charging.<sup>42</sup>

When planning your user research, make sure to include people responsible for procurement and finance. Map out procurement “pain-points” and prioritise addressing them.

#### **Example: Designing GOV.UK Notify for self-service**

*The GOV.UK Notify team aimed to build for self-service from day one. They discovered that it was an effective way of increasing adoption for the shared component.*

From day one of GOV.UK Notify, our hypothesis was that we needed to build something that was highly scalable. That meant we needed to do what everyone else on the web was doing: design for self-service. Jeff Bezos doesn’t come around your house if you need to spin-up a new EC2 instance.

.....  
<sup>41</sup> “About eSign API,” <https://indiastack.org/esign/>. Retrieved 8th June 2019.

<sup>42</sup> HM Government, “Pricing,” <https://www.notifications.service.gov.uk/pricing>. Retrieved 26th June 2019.

If you have a government email address, you can create a trial account. We also got really rich feedback by saying to people, “rather than a meeting or a phone call, have a go first, and then come back to us if you get stuck.” This was a bold step when we only had a few users but got easier as adoption grew, and the insight gained was invaluable.

This approach also allowed us to sell what we were doing through our presentations and blog posts etc. where the call-to-action wasn’t “get in touch,” it was “go and try it.” Within the team we called this “permission to play.”

— Pete Herlihy, Lead Product Manager, Government Digital Service

## **DATA INFRASTRUCTURE, APIs AND OPEN STANDARDS**

With legacy systems, data gets shared and duplicated. In a platform ecosystem data is accessed as needed, via APIs using agreed open-standards. Rather than being held in multiple places and multiple formats, data is stored in canonical registers.

### **1. Create canonical registers of data and manage them on behalf of multiple services**

The development and maintenance of canonical data registers—agreed lists of facts like addresses, refuse collection days, or the location of health facilities is critical for enabling other services. Registers need to be managed for the broader ecosystem of public-facing services, rather than tied to a single, specific service.

For example, [adresse.data.gouv.fr](https://adresse.data.gouv.fr) is the French register of addresses and is operated by the Etalab, the government unit responsible for open data. The data is made available via an API under an open licence so that it is free and reusable by anyone.<sup>43</sup>

Identify and actively develop registers, ensuring that they have clear custodianship, appropriate licensing and are designed for self-service. Your aim should be to maximize the benefit the data can provide.

.....  
<sup>43</sup> Etalab, “Foire aux questions—adresse.data.gouv.fr,” <https://adresse.data.gouv.fr/faq>. Retrieved 8th June 2019.



## **2. Use open, emergent processes to agree open-standards**

Open standards mean that different teams can be confident that they are speaking the same language, and the designers of public facing services can build on top of reliable, predictable data.

The process of agreeing standards need not be onerous. Large-scale, standardization efforts outside of government show what is possible, from OpenStreetMap to the Fast Healthcare Interoperability Resource specification, and Internet Engineering Task Force. Each effort favours progress, working implementations, public evolution and concrete use cases over perfection and completeness.

In the UK, the Cabinet Office solicits suggestions for the adoption of existing open standards via GitHub.<sup>44</sup> Businesses, citizens or civil servants can suggest areas that may benefit from the adoption of standards. There is then an open process for the suggestion and adoption of open standards.

Create a process for agreeing and adopting standards. Aim to keep it simple and flexible.

## **3. Build on APIs and create new APIs**

Developing a platform ecosystem requires both canonical data registers, and the business processes of government (for example, the status of a visa application) to be exposed as APIs. This will not happen by itself. It requires APIs to be prioritized by teams developing services and for space to be made for the development of shared data infrastructure.

When Argentina was developing its new digital driving licence, rather than build a monolithic system, the National Road Safety Agency provided an API, this was then integrated into the miArgentina app where the digital driving licence service is housed.<sup>45</sup> And when 18F developed the cross-government FOIA.gov for managing freedom of information requests, they also provided an API for the public and agencies to use it programmatically.<sup>46</sup>

.....  
<sup>44</sup> "alphagov/open-standards," GitHub, <https://github.com/alphagov/open-standards>. Retrieved 5th June 2019.

<sup>45</sup> Mike Bracken, "Argentina just made driving licences digital," *Public Digital Blog*, 12th February 2019, <https://public.digital/2019/02/12/argentina-just-made-driving-licences-digital/>.

<sup>46</sup> US Department of Justice, "Developer resources," <https://www.foia.gov/developer/>. Retrieved 8th June 2019.

So, if you are developing a public facing service, ask yourself “what APIs could we expose? What registers should exist to run this service? What registers and shared APIs already exist that we should be using?”

#### **4. Provide bulk downloads for open data in addition to APIs**

Not everyone who wants to make use of open data from a register will wish to do so via an API. For example, a statistician or journalist may want to use it in a spreadsheet package.

Provide the ability to download open data in bulk, in addition to via APIs. Work to understand the needs of non-developers.

#### **Example: Digital driving licences in Argentina**

*As already noted, when Argentina was developing its new digital driving licence, rather than build a single system, it started with an API.*

The agency still has the data but pushes miArgentina an image with QR code and digital signature via an API. Eventually, we want every agency to be an API. The next areas we are looking at are car insurance, vehicle ownership and disability certificates.

The future of government digital services is transforming government into a public API where companies and citizens can build, connect and interact. That’s building government as a platform, or better, as a service.

— Daniel Abadie, Subsecretario de Gobierno Digital,  
Secretaría de Gobierno de Modernización (Argentina)

## WORKING ACROSS SILOS

While some platforms and tools may be centrally created and operated, others may emerge from a particular project, or be collaboratively developed between different parts of government.

Achieving this can be challenging in a world where government projects and agencies are accustomed to having control over everything within their policy domain, while often having little control over that which is adjacent. The dominance of a “not invented here” culture could render moot the work done by platform teams.

By fostering a culture where teams are encouraged to think about users beyond their own domain, and by investing in shared tools, these silos can begin to breakdown.

### 1. Work in the open and use cross-government tools

If teams can't find each other's work, the chances of them working together are slim. By understanding the needs of teams *across* government, in areas like source control, project management and deployment, you have an opportunity to reduce barriers to collaboration.

Google, for example, has a single version control system for its code, which thousands of engineers across the organization have access to. It enables them to find, reuse and improve existing components.<sup>47</sup> Estonia and the USA both emulated this, with publicly searchable repositories of source-code.<sup>48, 49</sup>

Shared project backlogs, such as those operated by the British Columbia Design System allow anyone with an interest to suggest an improvement.<sup>50</sup> The Federalist publishing platform maintained by 18F allows anyone to submit issues via github and has a public chat room for users to ask questions.<sup>51</sup> The Pipeline project in the UK, lets

.....  
47 Wired, “Google Is 2 Billion Lines of Code—And It’s All in One Place,” 16th June 2015, <https://www.wired.com/2015/09/google-2-billion-lines-codeand-one-place/>.

48 “code.gov,” <https://code.gov>. Retrieved 3rd March 2019.

49 “Estonia creates a public code repository for e-governance solutions,” e-estonia, April 2019, <https://e-estonia.com/code-repository-for-e-governance/>.

50 bcgov, “Component and Design Pattern Backlog,” GitHub, <https://github.com/bcgov/design-system/projects/1>. Retrieved 12th June 2019.

51 18F, “18F/federalist: Federalist builds the user-interface of government websites,” GitHub, <https://github.com/18F/federalist>. Retrieved 12th June, 2019.

different local governments share the projects they are working on and solicit for collaborators.<sup>52</sup> While, in Italy, the team leading the development of a national population register maintains a public forum where the 8000+ municipalities involved in the project can ask questions.<sup>53</sup>

Identify opportunities for using common, cross-government tools for things like source-control, project management and messaging. Publish code in the open, and under a suitable open license that makes reuse simple. Create a public register of projects currently in development, ideally with links to open backlogs.

## **2. Foster a culture of sharing**

Shared tools are not enough on their own, and while publishing open-source code is necessary, it is probably not sufficient when it comes to collaborative working. For this reason, a recent study of code.gov has recommended that, in addition to publishing code in the open, agencies also invest in community management and ensure that the reasons for publishing code are clear to all.<sup>54</sup>

The barriers that exist to prevent people from contributing to a project may be cultural rather than because of tooling.<sup>55</sup> People may be time poor, lacking in confidence or permission, so you will need to work to foster a culture where teams across government think beyond their immediate domain. Community events, like those organized by #OneTeamGov and #GovDesign, create spaces for people to share their work and establish shared ways of working. Both are grassroots efforts that bring people working on digital government projects from different agencies (and different countries) together.<sup>56 57</sup>

.....  
52 LocalGov Digital, “Welcome to Pipeline from LocalGov Digital,” <https://pipeline.localgov.digital>. Retrieved 12th June 2019.

53 Agenzia per l’Italia digitale, “Recenti ANPR—Anagrafe Nazionale argomenti—Forum Italia,” <https://forum.italia.it>. Retrieved 15th June 2019.

54 Jake Rashbass and Mairi Robertson, “The People’s Code,” April 2019, <https://ash.harvard.edu/files/ash/files/20190506paefinalash.pdf>.

55 CodeDotGov, “Always Improving: Making the Contribution to Repos Better,” *Medium*, 7th May 2019, <https://medium.com/codedotgov/always-improving-making-the-contribution-to-repos-better-3858db7c5511>.

56 One Team Government, <https://www.oneteamgov.uk>. Retrieved 5th June 2019.

57 #GovDesign, <http://gov-design.com>. Retrieved 5th June 2019.

Governance process and guidance also need to support (and hold to account) teams in using platforms and looking for opportunities to collaborate. The digital service standards, playbooks and assessment processes that many countries have created provide an opportunity to do this.

Celebrate and reward people for working collaboratively across silos, be that via community events, performance reviews or encouraging them to write about their work. Ensure that the processes you create for people to use and contribute to platforms are open and inclusive.

### **3. Get the leadership incentives right**

Teams need to feel they have the permission and support to work in this way, and that means the leadership incentives need to be right. Cross-silo working is hard if managers are focused on their own domain rather than the value to the wider organization. For this reason, one of Amazon's leadership principles is "ownership." Managers are encouraged to "act on behalf of the entire company, beyond just their own team" and "don't sacrifice long-term value for short-term results."<sup>58</sup>

Leaders taking a platform approach to government need to be comfortable allocating time and money to support work that could have a wider benefit, and help everyone feel it is part of their job to work together.

#### **Example: Building a community of contributors for the GOV.UK Design system**

*The GOV.UK Design is an open-source project that takes contributions from across UK central government.*

When we started working on the contribution model for the GOV.UK Design system, we optimised it for the scenario of one individual contributing a whole design pattern including all of the research, code, design and guidance.

We treated with that as our "stress case," believing if we could deal with that, other types of contribution would be easier to support in comparison. But we've since realised that if you build a model around the people who have the time, confidence, permission and knowledge to do that, it may only work for a small, highly-engaged and relatively privileged group of people.

.....  
<sup>58</sup> Amazon, "Leadership Principles," <https://www.amazon.jobs/en-gb/principles>. Retrieved 4th June 2019.

We're now considering how to make contribution possible for people who might find it harder to contribute. It's important to understand what barriers they might face, and what we could do to reduce them.

You also need to foster a spirit of collaboration. The right tooling, processes and documentation can help, but it's as much about creating the cultural infrastructure to support collaboration as anything else. That's hard to measure, but you can see it when it's happening.

—Amy Hupe, Senior Content Designer, GOV.UK Design System

## **REUSE**

Not everything needs to be built from scratch. Reusing existing platforms from other governments can reduce development time, and putting wrappers around commercial commodity services can allow for the use of commercial offerings with less risk of lock-in or loss of control.

### **1. Borrow from other governments**

While government platforms are a relatively new thing, there are opportunities to reuse the work of other governments. This includes reusing entire platforms, learnings from user research, guidance and standards.<sup>59</sup>

One example of this is the Digital Transformation Agency in Australia reusing the UK's GOV.UK Notify messaging platform.<sup>60</sup> Because the code was open-source, developers were able to quickly evaluate and set up their own implementation.<sup>61</sup> Another example is the reuse of analytics.gov by city and state governments across the US.<sup>62</sup>

During the discovery phase of a platform project, understand what other governments might have approached this problem before. Try contacting the teams and asking for advice.

.....  
<sup>59</sup> See <https://www.platformland.org/mapping/> for lists of platforms, standards and design systems.

<sup>60</sup> DTA, "govau/notify," GitHub, <https://github.com/govau/notify>. Retrieved 25th June 2019.

<sup>61</sup> Interview.

<sup>62</sup> Melody Kramer, "Tips for adapting analytics.usa.gov from Tennessee, Boulder, and Philadelphia," *18F Blog*, 6th January 2016, <https://18f.gsa.gov/2016/01/06/tips-for-adapting-analytics-usa-gov/>.

## **2. Put wrappers around commercial, commodity components**

In some instances, there may be components available on the commercial market that could be used as part of a platform ecosystem—for example, cloud hosting or digital payment providers. Creating thin wrappers around these can minimize the risk of lock-in, maintain control over the user experience and enforced standards. For example, in the US, cloud.gov is built on top of Amazon Web Services, and meets the standards set by FedRAMP (a standard approach to security governance across the US federal government).<sup>63</sup>

This approach may also open up new opportunities to manage demand and increase robustness. In the UK, GOV.UK Pay supports multiple payment providers and GOV.UK Notify awards contracts to multiple SMS gateways.<sup>64, 65</sup>

The approach of building wrappers around commercial components will work best for those which have some of the characteristics of “commodities” (i.e., they are well-defined, stable, and can be swapped in or out as required). You can use Wardley Mapping to help understand if potential components are commodities or not.<sup>66</sup>

### **Example: Reusing GOV.UK Notify in Australia**

We knew Notify had been very successful in the UK.

It was good for us as the source code was all there and available for our developers to go through, and that it was designed to run on similar cloud infrastructure. That meant that when we started the Alpha of our notifications platform, we could go in and see if GOV.UK Notify was fit for purpose.

It was a huge boost to getting started, but there was still work to do. The UK is single timezone based on UTC and Australia has three. We also had to change currency, financial year and phone number format to work for Australia. Then we

63 “What is cloud.gov?,” <https://cloud.gov/overview/overview/what-is-cloudgov/>. Retrieved 24th June 2019.

64 HM Government, “GOV.UK Pay’s Payment Service Provider,” <https://www.payments.service.gov.uk/payment-service-provider/>. Retrieved 24th June 2019.

65 Mark Say, “GDS awards SMS firms GOV.UK Notify contracts,” UKAuthority, 30th January 2019, <https://www.ukauthority.com/articles/gds-awards-sms-firms-govuk-notify-contracts/>.

66 Simon Wardley, “How commodity is something?,” *Bits or Pieces?* (blog), 15th September 2015, <https://blog.gardeviance.org/2015/09/how-commodity-is-something.html>.

had to slot in our design system (which was a bit harder because of our federal government system).

Reusing something like Notify fits with our ethos: get it out early and then improve it and make it fit for what Australia needs.

—Felicity Hitchcock, Product Manager, Tell Us Once and Platforms, DTA

## **FUNDING AND OPERATIONS**

Changing user behaviour, and evolving privacy and security risks, mean platforms need to be developed and operated by a team with an ongoing understanding of the needs of their users. Those teams should be in a position to continuously improve the platform and respond to any risks. It is important they are funded for the long-term, rather than treated as fixed-term projects, or as quasi-commercial entities and expected to pay their own way.

### **1. Create standing teams to develop and operate new platforms**

Bugs are a fact of life, as are malicious attacks. It's also a side effect of a successful platform that it's hard to know exactly how users will make use of it. This means user-behaviour and security risks will change over time. As such, a multidisciplinary team who are incentivized to improve and operate the platform is the preferred approach to the development of new platforms.

These teams do not need to be large, but they do need to be invested in the long-term health of the platform. Android, for example, has a small, very senior team. The lead developer for the Android platform, which powers 2.5 billion active devices worldwide, has a few dozen people on her team. The core framework team is an even smaller group of people, and many of them have been there since the beginning.<sup>67</sup>

### **2. Fund teams for the long-term (but start incrementally)**

Funding platforms on a short-term basis is at odds with the idea of shared infrastructure (who would build on top of an established platform if they didn't know if it was

.....  
<sup>67</sup> Interview, Dr. Adam Connors, Senior Google Engineering Manager.



going to be re-funded each year?) It also makes the retention of the team needed to operate and continuously improve a platform harder.

Cost-recovery based funding is also not easily compatible with the development of platforms for the simple fact that, because the cost-per-transaction decreases as the user base increases, early users are saddled with higher costs.

For governments used to modes of funding not suited to platforms this may represent a significant change both practically and culturally. But it's important to get it right. If the funding processes within your organization restrict you to only seeking funding on a short-term or cost-recovery basis, see if it is possible to discuss the benefits of a platform approach with commissioners, finance and approvals teams. Work with them on how to create longer-term business cases better suited to the sustainable development of platforms.

This does not mean that each platform needs multi-year funding from day one—teams should be supported with funding models that allow them to prove the value over time.

### **3. Understand any legal constraints on funding models**

Some organizations may be required by law to recover costs, and therefore find it harder to develop platforms. There may also be constraints on the procurement of services between central and local, or federal and state governments. In the long run, such issues may not make for a very good operating model. In the short-term, it is important to understand such constraints.

### **4. Fund emergent platforms**

While the need for some platforms may be more obvious, and the funding (potentially) easier to secure (for example, hosting or payments meet clear needs for multiple users), some needs will be more emergent.

As previously discussed in “Identifying platforms,” “point solutions” developed from within product teams, with the potential for broader use will need support and funding made available to support them.

Review how you would offer funding and support to a team who have identified a shared need as part of the development of a service.

### **5. Make charging a strategic choice, not a budgetary one**

Charging for a service may be too high a bar for some users, not necessarily because of the per-use cost, but due to the administrative costs associated with procurement. The team behind the GOV.UK Notify platform realized they could serve more users by offering the service free to the majority of them. They were only in a position to be able to do this because they were centrally funded.

In a limited number of cases, charging for high usage may be a way of ensuring efficient use.

Make an active decision about if and when to charge based on the needs of your users.

### **6. Understand when a new organization might be needed to operate a platform**

Platforms cut across the hierarchical structure of government: shared components mean that agencies may no longer operate their own payment or hosting systems, and registers and APIs mean that they may become more dependent on data from elsewhere to operate a service. While there may be tactical reasons for developing a platform in one part of government or another, ultimately the question must arise: “where is the best place to operate it for the wider good?”

In the UK, the success of the UK Government Digital Service was in part that it was a new organization at the center of government, with responsibilities for digital across central government.<sup>68</sup> In India, the Goods and Services Tax Network (GSTN) operates APIs and other infrastructure needed to operate the national harmonized sales tax. GSTN is a non-profit company owned partly by the national and state level governments.<sup>69</sup>

While it may be outside the remit of a product manager or a Chief Digital Officer to create new government agencies, having an opinion about the characteristics of an organization that should operate each platform is not.

.....  
68 Bryan Glick, “GDS is ‘sidelined’ and government as a platform ‘is dead,’ says Francis Maude,” *Computer Weekly*, 14th September 2017, <https://www.computerweekly.com/news/450426316/GDS-is-being-sidelined-and-government-as-a-platform-is-dead-says-Francis-Maude>.

69 Goods and Services Tax Network, “About Us,” <https://www.gstn.org/about-us/>.

### **7. Allow for multi-tenancy and self-hosting**

While there is often little reason for two different government agencies to use separate instances of a shared component, there may be legitimate reasons in some cases. For example, different tiers of government within a federal system may need to keep data separate.

Consider if having the option of both “software as a service” (multi-tenancy) and self-hosting would meet the needs of more users.

## **ADOPTION**

While designing for self service is probably the most important thing you can do to encourage people to use a platform, it is not the only factor. At different stages you may need to employ additional approaches including ensuring that early adopters become evangelists, learning to say “no,” and the smart use of mandates.

### **1. Use early adopters to demonstrate capability and build trust**

While the aim should be to design for self-service and a broad user base, there will always be a first user. You will probably find more enthusiasm for teams looking to be the second or third adopter (once they can point to the fact someone else is using it). Give the 1st user “VIP treatment,” co-locating with them if necessary. Map out how you will make your team available to support and address their concerns.

Once a platform has been adopted by one team in an agency, it is likely that others will adopt it too. This is partly because of trust, and partly because any contractual problems will likely have been solved. Blog and talk about your work so other teams know this. Be open about what you are learning through user research and observed usage. If you are able to, publish your roadmap in the open.

### **2. Do “sales”**

Explain to potential users across government and beyond how it will make their lives easier, as well as the lives of citizens. Be clear and confident about the value that you are delivering in terms of savings and usage (e.g., it’s going to “save this much time”

or “it’s going to be this much more valuable in terms of marginal costs” or “your integration costs are going to be lower”).

Be open and transparent about costs to the users of your platform. Help them understand the unit costs and how those will reduce as more services use the platform. Do this even if they are not directly paying for the service (as it will help them understand the cost if that becomes necessary in the future).

Inviting potential users to try the platform out, or showing a working prototype that uses it, is worth a million slick presentations.

### **3. Understand the risks of high-volume users early on in development**

There are risks and opportunities around trying to get a platform adopted by high profile services, especially early on in its development. While it might lend credibility, it can also risk the platform becoming tightly coupled with the demands of one use-case. They may also have demands around scale and robustness that it might be hard to satisfy early on in development.

Don’t be afraid to say no to feature requests, even from big users, especially if it is unlikely to meet a common need that other users have.

### **4. Mandate the right thing (and only once it works)**

While the aim should always be to design platforms that meet the needs of users and are so good they want to use it, there is a role for mandates. The key appears to be getting the level of the mandate or principles right.

At Google, for example, things that are mandated are high-level, rather than day-to-day—everything rolls up to a higher objective.<sup>70</sup> In Estonia, the interoperability between different government agencies using the X-Road platform is not mandated by law, it is the product of a set of guiding principles.<sup>71</sup>

Be careful about mandating the use of any particular platform and only consider it once you have proved it is useful.

.....  
<sup>70</sup> Interview, Dr. Adam Connors, Senior Google Engineering Manager.

<sup>71</sup> Rainer Kattel and Mergel Ines, “Estonia’s digital transformation: Mission mystique and the hiding hand,” UCL Institute for Innovation and Public Purpose working Paper Series, (IIPP WP 2018-09) (2018).

## DESIGNING SERVICES ON TOP OF PLATFORMS

The way services get created changes in a platform ecosystem. Rather than being built from scratch, they are assembled from common components.

This characteristic, used in combination with a service design approach, means new types of service could become possible (or at least much simpler to create). These include services that allow people to interact with multiple parts of government at the same time, real-time services, and services from the public, private and third sector that overlap and complement each other.

### 1. Design services, not policies

In a platform ecosystem, designing new services largely becomes a job of joining together platforms. With this reduced effort and reduced duplication comes the potential to save time and money. However, increased efficiency is not the most significant opportunity.

The biggest opportunity is the potential for radically better public services. That's because it does not just become simpler and quicker to design services, it also becomes simpler to design new types of service that are designed around the needs of a user rather than the organizational structure of government.<sup>72</sup> Looking around the world, we can begin to see that approach happening.

Estonia is starting to design new services around “life events,” like having a child, that transcend the boundaries of government agencies. They can do this because of the common cross-government data infrastructure they have developed.<sup>73</sup> Singapore's GovTech agency is following a similar approach and has developed a “Moments of Life” service.<sup>74</sup>

.....  
72 See this blog post by former UK Head of Design for Government for a fuller exploration of this idea. Lou Downe, “Better services with patterns and standards,” *Government Digital Service Blog*, 6th August 2015, <https://gds.blog.gov.uk/2015/08/06/better-services-with-patterns-and-standards/>.

73 Sam Trendall, “We have only scratched the surface”—Estonia's CIO on what's next for the world's most celebrated digital nation,” *PublicTechnology.net*, 18th February 2019, <https://publictechnology.net/articles/features/'we-have-only-scratched-surface'---estonia's-cio-what's-next-world's-most>.

74 GovTech Singapore, “The Tech Behind The Moments Of Life (Families) App,” <https://www.tech.gov.sg/media/technews/the-tech-behind-the-moments-of-life>. Retrieved 13th June 2019.

The UK government Service Standard (which all new government services must meet) includes the following requirement for teams designing services to work beyond the boundaries of an individual government agency:

Work toward creating a service that solves one whole problem for users, collaborating across organizational boundaries where necessary.<sup>75</sup>

Also in the UK, the different steps needed to complete a task like “employing someone” or “learning to drive” organized in one place on the GOV.UK website, regardless of which government agency a user has to interact with.<sup>76</sup>

Take advantage of the opportunity of platforms design services to solve whole problems for people. Look for opportunities to make end-to-end service design part of official guidance and governance processes.

## **2. Think of services as built from many platforms**

Just as platforms are part of many services, so services are made from many platforms. This means that a team designing the service can focus on the areas that are unique to the domain they are working in, and prototype a working service much faster.

A great example is the Christmas tree permitting service operated by the US Forest Service that allows people to legally harvest a Christmas tree for their home from a national forest. The service uses pay.gov (to take payments), login.gov (for authentication), is hosted on cloud.gov, and uses a generic permits API.<sup>77, 78</sup> The user interface is built using the US Web Design system as a base.<sup>79</sup>

.....  
75 HM Government, “Service Standard,” <https://www.gov.uk/service-manual/service-standard>. Retrieved 5th June 2019.

76 Sam Dub and Gabrielle Acosta, “Building a better GOV.UK, step by step,” *Government Digital Service Blog*, 17th October 2018, <https://gds.blog.gov.uk/2018/10/17/building-a-better-gov-uk-step-by-step/>.

77 18F, “18F/fs-open-forest Wiki,” GitHub, <https://github.com/18F/fs-open-forest/wiki>. Retrieved 5th June 2019.

78 18F, “Ongoing site architecture 18F/fs-open-forest Wiki,” GitHub, <https://github.com/18F/fs-open-forest/wiki/Ongoing-site-architecture>. Retrieved 5th June 2019.

79 18F, “18F/fs-fork-uswds: The Forest Service fork of US Web Design System is a design system for building fast, accessible, mobile-friendly Forest Service websites,” GitHub, <https://github.com/18F/fs-fork-uswds>. Retrieved 5th June 2019.

The other implication of this one-to-many principle is that a service team will not have control over the entire user journey, and this may take some adjustment for teams used to controlling the totality of the user experience.

When designing services that use platforms, understand what existing platforms you could use to develop your service. Work in the open and maintain good communications with teams operating platforms so you can ensure you understand any upcoming changes.

### **3. Allow for multiple, overlapping services, built on a common platform**

Historically, policymakers have had a couple of clear, choices to make when it comes to the delivery of a service to the public. Should it be delivered centrally or locally? Should it be done by the private sector or the public sector? A healthy platform ecosystem could start to turn those choices into false choices.

That's because it allows governments to follow both a retail approach to providing services (where government offers a public-facing service) **and** a wholesale one (where the charities and private companies also provide services).<sup>80</sup>

Because platforms do the “heavy lifting,” it becomes more feasible to build multiple services on top of them. These services could be provided by any layer of government, and by commercial or third sector organizations, in such a way that is ok if they overlap, complement or duplicate each other.

One recent example of this wholesale and retail approach is recreation.gov, which is a public-facing government service for finding and booking campsites and tours on federal land. It is built on an API that is also used by a commercial vacation website.<sup>81</sup> These services overlap and to some extent compete with each other, but serve slightly different audiences.

.....  
80 Aneesh Chopra and Nick Sinai, “Wholesale Government: Open Data and APIs,” *Medium*, 9th April 2015, <https://medium.com/@ShorensteinCtr/wholesale-government-open-data-and-apis-7d5502f9e2be>.

81 A recent example is recreation.gov, which has a public facing website for finding and booking campsites and tours on federal land. It also has an API that is used by commercial websites to do the same thing. (I think this article is a bit unfair on the government run website). Lindsey Smith, “Booking campsites on Recreation.gov is a mess. Here’s the solution,” *San Francisco Chronicle*, 4th March 2019, <https://www.sfchronicle.com/travel/article/Recreation-gov-is-a-mess-Here-s-the-solution-13655884.php>.

It could also become much simpler to interact with multiple layers of government at once in a single service. Designing services that allow people to interact with central and local government within a single service.

#### **4. Design real-time services (but be cautious of the risks)**

One of the characteristics of modern digital services is that they operate in real-time. If you want to buy something on Amazon, you see if it's out-of-stock immediately, and pricing on Uber responds to demand in real-time. The teams operating those services also benefit from real-time feedback loops so they can see where users are getting stuck and what's not working as it should.

Registers, and shared APIs should make these things simpler for government services, so that, for example, users could track the exact point in the process their welfare application is at.

Real-time services will also come with new risks. For example, the Indian hotel chain Oyo plans to share guests check-in data with state governments in real-time (replacing an existing, manual process). The company's motivation is to reduce errors and make things faster for hoteliers and guests. But it has also raised concerns from privacy campaigners about the affordances and unintended consequences of such a system.<sup>82, 83, 84</sup>

When designing real-time services, a key decision for services designers is going to be where there should be a human in the loop.

Design services that take advantage of the real-time nature of platforms, but consider negative consequences as well as opportunities.

.....  
82 "Oyo to share customers' arrival, departure details with authorities," *Business Today*, 15th January 2019, <https://www.businesstoday.in/buzztop/buzztop-corporate/oyo-to-share-customers-arrival-departure-details-with-authorities/story/310152.html>.

83 "Oyo Rooms wants to share digital record of your check-ins with government," *India Today*, 15th January 2019, <https://www.indiatoday.in/technology/news/story/oyo-rooms-wants-to-share-digital-record-of-your-check-ins-with-government-1431439-2019-01-15>.

84 @internetfreedom / Internet Freedom Foundation (IFF), "Everyday we loose more of our privacy. This also means a live update the next time you check into a hotel room. Without a warrant or a legal request, instantaneous. Scalable, deeper, richer and in real time," 15th January 2019, <https://mobile.twitter.com/internetfreedom/status/1085169410799751168>.



## 5. Design event-driven services

Lots of government services require people to report when things in their life change, and this places a lot of responsibility on the user—they need a good mental model of the service to know what to report when and how. (The UK’s Universal Credit benefits system requires people to internalize over 30 types of change-of-circumstance!)<sup>85</sup>

A different approach is to design services that both respond to events, and make use of data held elsewhere. Trust and identity systems, along with shared APIs and registers could make it simpler to design services that forgo the need for lots of secondary digital transactions (things like “update this,” “report that,” “change this,” “re-apply for that.”)

Rather than placing the burden on users, look for opportunities to use APIs and registers to design services that proactively meet the needs of users based on events.

### **Example: Building US Forest Service Christmas tree permits service on top of platforms**

The Christmas tree permitting service operated by the US Forest Service lets people to legally harvest a Christmas tree for their home from a national forest. It’s built on top off several platforms, including pay.gov and cloud.gov.

Before Open Forest, there was no online purchasing option for these permits, limiting the public to buying permits at mostly remote USFS district offices or sometimes from local vendors during weekday business hours. From launch to the the end of the holiday season, Open Forest sold nearly 5,000 permits across the four pilot forests.

Though 18F has delivered many products to production since its founding in 2014, this is our first product to include payment functionality. Purchasers can use their credit card, debit card, or bank account number to pay for a permit through the US Department of the Treasury’s Pay.gov. Leveraging this existing service helped us implement a modular approach; our product team focused on

.....  
<sup>85</sup> Citizens Advice, “Check if a change affects your Universal Credit,” <https://www.citizensadvice.org.uk/benefits/universal-credit/what-youll-need-to-do-on-universal-credit/check-if-a-change-affects-your-universal-credit/>. Retrieved 5th June 2019.

the unique Christmas tree sale process, rather than reinventing how to make a secure payment. For the Forest Service staff in the district offices, it minimized the administrative burden of processing large sums of cash.

Why did we design and build a print-at-home solution rather than a mobile one? With limited cellular service and temperatures dropping below freezing that can quickly drain a mobile phone battery, a digital solution might not be that durable.

—Laura Gerhardt, 18F (extract from a blog post)<sup>86</sup>

### **Measuring and Monitoring**

The primary users of a platform tend not to be the public themselves, but rather the people who use them to create services for the public (be it civil servants, charities or private sector companies). Ultimately, the success of a platform will be perceived by the success of those services, not the platform itself. This means platform teams need to understand what they need to measure to operate the platform, and what data their users need to provide great services to the public.

#### **1. Understand what data your team needs**

A team developing a platform will have clear a set of needs that relate to the operation and improvement of that platform. These may include some of the following:

- Are people self-serving? (For example, what’s the ratio of meetings to users)?
- Are people using the documentation?
- What code libraries are being used?
- Where are the users (central government, local government, private sector)?
- How many distinct services and organizations are using the platform?
- What is the service up-time? (Up-time targets will likely vary depending on the type of platform.)

.....  
<sup>86</sup> Extract from a blog post published by Laura Gerhardt published after the launch of the service. Laura Gerhardt, “Buying Christmas tree permits online with Open Forest,” *18F Blog*, 12th February 2019, <https://18f.gsa.gov/2019/02/12/open-forest-launch-post/>.

Conduct user research with your team and understand what data they need to do their job effectively.

**2. Understand what data your users need**

Because platforms are component parts of services, the success of a platform will ultimately be perceived by the success of those services. As such, it is critical to understand what data your users will need access to as part of the operation of their service. Google, for example, maintains specific tools to help Android developers understand app performance.<sup>87</sup>

Much of the data users need may be domain specific, so conduct user research with them to understand their needs.

**3. Use a “show back” approach to build common understanding of cost and efficiency**

The ideal is that platforms are funded centrally and operated for free to create the broadest possible public good. However, it is still important that your users understand the cost and wider value of running each platform. This approach is called “show back” (as opposed to “pay back” where users are charged for use).

If it is not possible to show back actual monetary costs, start with the percentage of use represented by different users (for example the number of API calls or transactions). The aim is to create a shared mindset of ownership and an understanding of cost.<sup>88</sup>

Use a “show back” approach to create a shared understanding of efficiency and cost among your users. Where possible, measure cost-per-use and be transparent to users of the platform, even if they are using the platform for free.

**4. Publish performance data in the open**

Publishing performance data can help users better understand what a platform does and how it is being iterated.

Italy’s Team per la Trasformazione Digitale publish data about the rollout of the National Resident Population Register, including register size and the number of

.....  
87 Google, “Google Play Console,” <https://developer.android.com/distribute/console>. Retrieved 9th June 2019.

88 This white paper from Amazon explains how to use measurement and transparency to create a shared understanding of cost and efficacy. Amazon, “Creating a Culture of Cost Transparency and Accountability,” March 2018, <https://docs.aws.amazon.com/whitepapers/latest/cost-optimization-transparency-accountability/introduction.html>.

municipalities that have migrated to it.<sup>89</sup> India’s Aadhaar and Estonia’s X-road also have public dashboards that show usage, while the UK’s GOV.UK Pay publishes data about the organizations using it and the total value of the payments that it has processed.<sup>90, 91, 92</sup>

Publish performance data about your platform in the open.

### **5. Prepare for unexpected user behaviour**

It is in the nature of a platform that you don’t know exactly how or when people will use it. If someone is going to do something new with your platform, they will likely not tell you in advance. This is especially true if you have designed for self-service.

Understand the roadmaps of any large services making use of your platform so you can better anticipate their usage. Use analytics to understand emerging use-cases of the platform.

### **6. Create a public system status page**

It is standard practice for commercial platforms to maintain a status page that details the current status of the system.<sup>93</sup> Many digital service units are following this practice with the UK, Australia and US (among others) maintaining status pages for their platforms.<sup>94, 95, 96</sup>

It is also standard practice for commercial platforms to publish public incident reports detailing any outages or security issues.<sup>97</sup>

Create a public status page for each platform and ensure users can find it. Once resolved, publish incident reports detailing outages or security issues.

.....  
89 Team Digitale, “National Resident Population Register (ANPR)—Digital Transformation Team,” <https://teamdigitale.governo.it/en/projects/anpr.htm>. Retrieved 9th June 2019.

90 UIAI, “Aadhaar Dashboard,” [https://uidai.gov.in/aadhaar\\_dashboard/](https://uidai.gov.in/aadhaar_dashboard/). Retrieved 9th June 2019.

91 “X-TEE,” <https://www.x-tee.ee/factsheets/EE/#eng>. Retrieved 9th June 2019.

92 HM Government, “Dashboard, GOV.UK Pay, GOV.UK,” <https://www.gov.uk/performance/govuk-pay>. Retrieved 9th June 2019.

93 For examples see: <https://status.twilio.com>, <https://www.gocardless-status.com>, <https://status.aws.amazon.com> and <https://status.stripe.com>.

94 “GOV.UK Pay Status,” <https://payments.statuspage.io>. Retrieved 9th June 2019.

95 “cloud.gov.au Status,” <https://status.cloud.gov.au>. Retrieved 9th June 2019.

96 “cloud.gov Status,” <https://cloudgov.statuspage.io>. Retrieved 9th June 2019.

97 See <https://aws.amazon.com/message/41926/> and <https://gocardless.com/blog/incident-review-api-and-dashboard-outage-on-10th-october/> for examples.

### **7. Use the privileged position of the platform respectfully**

Because platforms centralize, the teams operating them have the power to see how it is being used for a range of different users. This can be used to identify usage patterns that can be generalized into new features, or used to improve how others use the platform. However, with great power comes great responsibility. Understand the ethical and privacy implications of any new data analysis or collection.

#### **Example: Monitoring and tooling for Android**

Google maintains specific tools to help Android developers understand app performance and respond to issues.

The dynamic is that whatever you do with your platform, you are perceived through the lens of the applications that run on it, so you need to provide tooling as well as APIs.

We have a tonne of reporting that's built into the platform. If an application crashes, the system shows a dialogue: "Send report?"—with user permission, it'll send back anonymized (non-personally identifiable) system information to the core team (so we can debug our platform) and it'll send it to the app developers so they are aware of issues. A bunch of energy is invested in Play Developer Console where we give developers a tonne of metrics & tools to help them improve their applications.

—Dr. Adam Connors, Senior Google Engineering Manager

## **IDENTITY, TRUST AND CONSENT**

When it comes to digital identity and the exchange of data between organizations, many of the requirements for the safe operation of a platform ecosystem, such as an effective and legal framework for privacy and digital rights, are beyond the gift of digital service teams. However, there are many things that you can do to minimize the risks to privacy and trust. As such, please do not read this section as anything approaching a complete list, instead, see it as a set of practical things that should be within your control.

### **1. Design for an identity ecosystem (not an identity system)**

Digital identity is often thought of in terms of a singular system—a single digital identity system to join everything together. However, there are several risks with this approach.

It potentially creates a single point of failure and it risks joining together data sets that are best kept separate. The cost (both to privacy and in financially) of single system being compromised is higher.<sup>98</sup>

It may also not reflect the reality that many countries have multiple de facto systems of ID—a country may operate different identity systems at a municipal and national level. For example, Sweden operates multiple identity systems, including IDs issued by banks accepted for certain government services.<sup>99</sup> European countries are beginning to accept each other’s digital IDs as part of the eIDAS program.<sup>100</sup>

Finally, people may have legitimate reasons for wanting to keep different aspects of their lives separate through different identities (for example, keeping health information separate).

Understand these risks when thinking about implementing or utilizing a digital identity system.

### **2. Design for trust escalation (and failure)**

There are many services for which proving an identity will not be necessary at all. There are also services, or stages of services, that demand different levels of trust. For example, applying and paying for a licence may require a lower level of trust than someone changing the bank their benefit payments are paid to.

Services should be able to establish enough trust (and only enough trust) to allow their users to achieve their desired outcome.

.....  
98 e.g., South Korea had to reissue government IDs at huge cost after a data breach. Iain Thompson, “South Korea faces \$1bn bill after hackers raid national ID database,” *The Register*, 14th October 2014, [https://www.theregister.co.uk/2014/10/14/south\\_korea\\_national\\_identity\\_system\\_hacked/](https://www.theregister.co.uk/2014/10/14/south_korea_national_identity_system_hacked/).

99 BankID, <https://www.bankid.com/en/>. Retrieved 8th February 2019.

100 “Trust Services and Electronic identification (eID),” <https://ec.europa.eu/digital-single-market/en/trust-services-and-eid>. Retrieved 26th June 2019.

Inevitably, there will also be circumstances where people are unable to meet the level of trust online (or they may be unable to go online at all). As such, providing a real-world fall-back is important. In the UK, this idea has been explored through the use of a chain of high street stores for checking documents in person.<sup>101</sup>

Avoid putting an identity verification step in services that don't need it. Map out the levels of trust needed for each stage of a service and integrate identity verification and authentication steps accordingly. Provide real-world fall-backs for when a process cannot be completed digitally.

### **3. Understand digital identity in context**

In many contexts, digital identity systems have the opportunity to make services simpler by creating a consistent way for people to verify their identity to different institutions—be that government, or refugee agencies providing aid. They also have the potential to offer opportunities for marginalized and undocumented communities to access vital services, such as access to justice, and welfare.

However, centralized systems, and particularly systems which gather biometric data, also bring huge risks and challenges to the privacy and security of people whose data is collected. This is particularly the case when mechanisms for recourse or accountability in are absent or fall short.

The way data that is collected and stored against someone's identity also affects how those people's rights can be exercised. For example are there are options for people to represent themselves as they see themselves (for example, non-binary genders, people from countries that aren't recognized by government institutions).

These risks and benefits will affect different groups in different ways (for example, by gender).<sup>102</sup> Also, as new datasets (such as healthcare or immigration status) are joined together through a digital identity, those risks will change.

Understand the implications of how data is associated with a digital identity for different groups. Provide clear routes to address mistakes or unintended consequences.

.....  
<sup>101</sup> Open Identity Exchange, "Face-to-face identity proofing to help people obtain an assured digital identity," <https://oixuk.org/wp-content/uploads/2017/01/F2F-white-paper-final.pdf>.

<sup>102</sup> Zara Rahman, "Digital ID: why it matters, and what we're doing about it," 13th September 2018, <https://www.theengineroom.org/digital-id-why-it-matters/>.

#### **4. Ask people for less data**

Collecting and retaining data to personalize a platform or service comes at a cost to a users time and privacy. The more data collected and held, the greater the security and privacy risk.

Collect only the minimum data needed to operate a platform or service.

#### **5. Understand where data lies on the open-shared-closed spectrum**

Registers should be made as open as possible, but understanding of what “as possible” means requires a deliberate decision about where data lies on the spectrum of “open-shared-closed.”<sup>103</sup> This should be informed by data impact and privacy assessments, security and ethical considerations.

Understand where data held by a register or shared API sits on the open data spectrum. Use this knowledge to design an appropriate data access model and governance process for the data.

#### **6. Make some data harder to join together**

There can be unintended consequences when data is joined together.

A clear example this is important comes from the UK health and immigration services.

In November 2018, the UK National Health Service finally pulled out of a controversial “memorandum of understanding” with the Home Office (which is responsible for immigration enforcement). It had granted Home Office immigration officers access to data about patients to help them trace people breaking immigration rules.<sup>104</sup> The withdrawal followed objections from Public Health England and the House of Commons Health and Social Care Committee that people not seeking medical attention, or deliberately missing vaccinations for fear of data about them being shared, posed a risk to public health.<sup>105, 106</sup>

.....  
<sup>103</sup> Open Data Institute, “The Data Spectrum helps you understand the language of data,” <https://theodi.org/about-the-odi/the-data-spectrum/>. Retrieved 26th June 2019.

<sup>104</sup> Jasmin Gray, “NHS Pulls Out Of Data-Sharing Deal With Home Office Immigration Enforcers,” *Huffington Post*, 12th November 2018, <https://www.huffingtonpost.co.uk/entry/nhs-data-sharing-home-officeuk5be97198e4boe843889a1b5d>.

<sup>105</sup> Denis Campbell, “NHS will no longer have to share immigrants” data with Home Office,” *Guardian*, 9th May 2018, <https://www.theguardian.com/society/2018/may/09/government-to-stop-forcing-nhs-to-share-patients-data-with-home-office>.

<sup>106</sup> Alan Travis, “NHS chiefs urged to stop giving patient data to immigration officials,” *Guardian*, 31st January 2018, <https://www.theguardian.com/society/2018/jan/31/nhs-chiefs-stop-patient-data-immigration-officials>.



While this may be an extreme example, understanding such risks is critical. It is also important to be transparent to the public about when and how data is being joined together or exchanged. (There are many different design patterns available for informing users about data use, or asking for explicit consent.)<sup>107</sup>

Understand the risks of joining together different datasets. Be transparent to the public about data use.

### **7. Create APIs that answer questions**

When designing APIs, rather than sharing full database records, create endpoints answer specific questions. For example, “does this car have a permit to park in this area?—yes or no?,” “Is this person entitled to work in this country—yes or no?” This allows services to find the information needed to operate that service, without all the negatives that come with traditional approaches to “data-sharing” in government.

Look for opportunities to limit the amount of information that is exposed via APIs.

### **8. Use digital proofs to join services together**

Digital proofs offer the opportunity to keep some of the good things about paper certificates while removing some of the risks. They also allow people to prove things to whomever they like and however they like.

This means the designers of a service don’t need to guess upfront all the different situations that someone might experience. For example, someone can prove to one part of government that another part of government agrees a certain fact about them—for example, that they own a property—without the need for duplicating and sharing entire datasets between organizations. It’s up to the organization or individual receiving the proof to decide if they trust it or not, which allows for flexibility.

One example is Digilocker, part of the IndiaStack project. Digilocker is a personal data store for cryptographically signed government documents. Among other things, people can use it to store a digital representation of their driving licence, or a university certificate signed by the organization that issued it.<sup>108</sup>

.....  
<sup>107</sup> IF, “Data Permissions Catalogue,” <https://catalogue.projectsbyif.com>. Retrieved 26th June 2019.

<sup>108</sup> National eGovernance Division Ministry of Electronics & Information Technology (MeitY), “About DigiLocker,” <https://digilocker.gov.in/about.php>. Retrieved 5th April 2019.

Another example is the approach used by the GOV.UK “Share driving licence” service. It lets users create a one-use code that a third party (for example a car-rental company) can use to check the details of a licence.<sup>109</sup> This has the advantage that it clearly puts the user in control of how and when they share data about themselves.

Digital proofs, in combination with apps like DigiLocker for storing and managing digital proofs, are an evolving area, with the Android platform and the W3C both actively developing approaches.<sup>110, 111</sup> The important thing to consider however, is not the technology, but where the principle of putting verifiable facts in the hands of the public would be most beneficial.

### **Transparency, Accountability and Governance**

Transparency, accountability and governance processes are not an end in themselves. The aim should be to make sure things are working as they should be, and that people understand what “should be” looks like.

As with trust and consent systems, there are limits to what a digital service team can do in this area (many responsibilities will lay with elected representatives and empowered regulators). So, again, please do not read this section as a complete list, instead, see it as a set of practical things you can do to improve accountability and understanding.

#### **1. Put accountability, recourse and transparency at the point of use**

A platform ecosystem in combination with services designed to “just work” could obfuscate how government works. This is not supposed to suggest that people have a brilliant mental model of how government works today, but there is an obligation on teams designing platforms and services to ensure they make that understanding better, not worse. Taken as a whole, the system should actively educate people about how their government works, where power and accountability lie and how services are performing.

.....  
<sup>109</sup> Sally Meecham, “A new way to view your driving licence info online,” *Government Digital Service Blog*, 7th October 2014, <https://gds.blog.gov.uk/2014/10/07/a-new-way-to-view-your-driving-licence-info-online/>.

<sup>110</sup> Mishaal Rahman, “Google is working on securely storing Digital Driver’s Licenses in Android,” XDA Developers, 5th March 2019, <https://www.xda-developers.com/google-android-digital-drivers-license/>.

<sup>111</sup> W3C, “Verifiable Credentials Data Model 1.0,” 28th March 2019, <https://www.w3.org/TR/verifiable-claims-data-model/>.

For each service linked to from the GOV.UK publishing platform, there is a corresponding page that explains some elements of the current performance of the service and an explanation of the underlying user need for the service.<sup>112</sup> While the information is currently limited, it is easy to imagine it expanding to include data about both the real-world analytics, the data and the underlying legislation used to operate the service.

In New South Wales, Australia government services make use of a shared component, (in this case, a wrapper around a commercial CRM), for adding feedback to digital services.<sup>113</sup>

Put information about performance and accountability at the point of use of services. Conduct user research to understand the information citizens, their representatives, or civil society organizations might need to understand how a service works and who is accountable for it.

## **2. Ensure platforms have owners and registers have custodians**

Clear product ownership (one person who can direct the work of a team to meet the needs of users) is just as important for platforms as it is for public-facing services. Even in situations where some element of a platform is outsourced, you will still need to maintain the expertise in-house to monitor and audit.

The consequences of incorrect data in a platform ecosystem are potentially higher than in legacy systems (an incorrect data item in a register could affect many services). As such, data custodianship also needs to be clear (and this may be distinct from product ownership). A data custodian is accountable for the quality and accuracy of data.

Ensure that there is a product owner for each platform and service and that each register has a clear custodian.

## **3. Decide the “rules of the road” for non-government services**

For platforms that allow non-government services to be built on top of them, there may need to be clear rules about the type of services that can be built on top of them (for example, if they make use of sensitive data, or require users to understand a particular piece of information).

.....  
<sup>112</sup> For an example see <https://www.gov.uk/info/jobsearch>.

<sup>113</sup> NSW Government, “Feedback Assist,” <https://www.digital.nsw.gov.au/digital-design-system/components/reusable-components/feedback-assist>. Retrieved 25th June 2019.

Consider the risks of government being disintermediated and of data misuse. Understand which parts of the user experience government will always need to control.

#### **4. Put users in control of the data about them**

Put users in control of their data. Millions of engaged curators are the best protection government has against fraud, and that citizens have against misuse. Users should be able to see what services have access to their data and what access has been made.

By default, Estonian citizens have the right to see *all* data held about them, unless explicitly restricted.<sup>114</sup> In India, holders of an Aadhaar identity can view the history of how it has been used.<sup>115</sup>

Design ways for people to access and understand how data about them is being used.

#### **5. Make understanding risks part of the design process**

Governance should not be something that just happens in the realm of policy documents and committees. Design and development processes also need to make space for an understanding of risks and ethical considerations. This is true both of platforms (what might the unintended consequences be of collecting, modelling or using data in a particular way?) and the services built on top of them (who benefits and who might lose?).

There are a few emerging approaches to this, including “consequence scanning” and “data ethics canvases.”<sup>116, 117</sup> Adopting a human rights review of key digital infrastructure has also been suggested.<sup>118</sup>

Understand where approaches like these could fit into your methodology.

.....  
114 Riigikantselei, “Public Information Act,” Section 43-8, <https://www.riigiteataja.ee/en/eli/518012016001/consolide>.

115 Unique Identification Authority of India, “Aadhaar Authentication History,” <https://uidai.gov.in/305-faqs/aadhaar-online-services/aadhaar-authentication-history.html>. Retrieved 25th June 2019.

116 Open Data Institute, “What is the Data Ethics Canvas?,” 5th April 2017, <https://theodi.org/article/data-ethics-canvas/>.

117 Doteveryone, “Consequence Scanning,” <https://doteveryone.org.uk/project/consequence-scanning/>. Retrieved 25th June 2019.

118 Beatrice Martini, “What is at stake for human rights in the design of Internet protocols?,” Kennedy School Review, 13th May 2019, <http://ksr.hkspublications.org/2019/05/13/what-is-at-stake-for-human-rights-in-the-design-of-internet-protocols/>.

## **6. Expose the rules and design for monitoring**

Digital systems encode the law in software. They take the intent of politicians and parliaments and make them manifest in the form of automated decisions, online forms and content. This raises the question of how people (or more likely their representatives) understand if a system is working as it should be?

One approach is to examine the code itself. Publishing code and associated software tests in the open could potentially help third parties understand how government systems work and better hold them to account.

Another approach is to expose data that helps third parties monitor the quality of a platform or service, or the quality and integrity of a dataset. There are few examples of this happening in a systematic fashion, but in the US, the Web Integrity Project (WIP) points to what is possible.

The WIP project is independent of government. It automatically monitors for changes to federal government websites looking for changes which it uses to understand changes in government policy that, among other things, might affect people's ability to access services.<sup>119</sup> Currently, it does this by crawling government websites, but this could be made much easier by exposing content changes via APIs.

Look for opportunities to expose code and data about platforms and services that might help third parties hold government to account.

## **7. Maintain an archive of user interface changes**

Small design choices can have big impacts on how people access and understand services.<sup>120</sup> Seeking out screen-shots of government software is also increasingly becoming a human rights monitoring activity—recent studies of China's Social Credit System and Saudi Arabia's "Absher" have both made use of screen-shots to understand the impact on people's rights.

.....  
<sup>119</sup> "Web Integrity Project," <https://sunlightfoundation.com/web-integrity-project/>. Retrieved 26th June 2019.

<sup>120</sup> As immigration solicitor, Jonathan Kingham, notes in this article on the digitization of Brexit era immigration systems in the UK: "Unlike with Rules and legislation changes, there was little opportunity to scrutinise the detail of what are in fact significant changes to the immigration system prior to their coming into force (bar selective 'user testing,' which is rarely transparent to all). And, as with so many tech developments, as the process or 'app' itself increasingly takes centre stage over the content (in this case the law) that underpins it, there are risks." 8th January 2019, "'Computer says no': facing up to the full implications of a digitised immigration system," <https://www.freemovement.org.uk/computer-says-no-digitised-immigration-system/>.

Maintain a public archive of any changes to the user interfaces of services (and platforms that have public-facing elements).

### **8. Design for verifiable use of data**

Numerous examples around the world show that access rules and legal frameworks are not enough to limit the misuse of data. There are emerging approaches that allow for the verifiable use of data through recording access in immutable databases such as Trillian or Amazon QLDB.<sup>121, 122, 123, 124</sup>

Estonia's X-Road data exchange uses this approach to create a tamper-proof log of data accessed across the system. (Contrary to reports, X-Road does not use blockchain).<sup>125</sup>

Don't just rely on access controls to limit the use of data.

#### **Example: The Web Integrity Project**

The Web Integrity Project systematically monitors changes to 30,000 federal agency webpages using software that crawls URLs every few days and alerts us when a change has been made. A team of analysts review the changes, and the Web Integrity Project extensively reviews and vet and changes we decide to report on. The regular monitoring of websites forms the basis of our efforts to hold the federal government accountable by revealing shifts in information and public access to resources. We work with journalists to make our findings public, and we produce policy analyses to evaluate and recommend changes to web governance practices and help ensure access to valuable Web resources.

Many of the changes we find are minor or routine. Others reflect stated shifts in policy. Some reveal the removal of large amounts of useful information

.....  
121 Open Data Institute/Register Dynamics, "Putting the trust in data trusts," 14th April 2019, <https://www.register-dynamics.co.uk/data-trusts/index.html>.

122 Emily Mattiussi, "Monitoring cloud data with Trillian," *IF Journal*, 3rd April 2019, <https://www.projectsbyif.com/blog/monitoring-cloud-data-with-trillian/>.

123 "google/trillian," GitHub, <https://github.com/google/trillian>. Retrieved 25th June 2019.

124 "Amazon Quantum Ledger Database (QLDB)," <https://aws.amazon.com/qldb/>. Retrieved 25th June 2019.

125 Petteri Kivimäki, "There is no blockchain technology in the X-Road," Nordic Institute for Interoperability Solutions blog, 26th April 2019, <https://www.niis.org/blog/2018/4/26/there-is-no-blockchain-technology-in-the-x-road>.

or data from public view. And sometimes we see alterations to websites that are more like spoilers: they foreshadow changes in policy that will come later after a much longer, more open rulemaking process or a public announcement of the policy shift.

A government API that included data on content changes, the creation and removal of pages, and usage statistics would be of immense use in our work. We could build a more automated and efficient workflow with access to information on which pages were changed, removed, or added and exactly when.

—Dr. Sarah John, Director, Web Integrity Project

## **EXPLAINING GOVERNMENT AS A PLATFORM**

Government as a Platform represents a new paradigm, not an incremental improvement on today’s system of government. Because of this, explaining platform approaches to senior stakeholders and politicians is both critical and potentially challenging. It’s critical because political sponsorship is important for such a radical change. It’s hard because it may not map neatly onto today’s business-cases and political cycles.

### **1. Find the metaphor that works for you**

There is no single “killer metaphor” for Government as a Platform, and different approaches seem to have worked in different situations.

In his article that coined the phrase *Government as a Platform*, Tim O’Reilly cites the philosophy of the Unix operating system as one of several inspirations for how government *could* work. There have been various attempts at codifying the *Unix philosophy*,<sup>126</sup> but this one from 1994 sums it up nicely:

Write programs that do one thing and do it well. Write programs to work together.  
Write programs to handle text streams, because that is a universal interface.<sup>127</sup>

.....  
<sup>126</sup> “Unix philosophy,” Wikipedia, [https://en.m.wikipedia.org/wiki/Unix\\_philosophy](https://en.m.wikipedia.org/wiki/Unix_philosophy).

<sup>127</sup> Peter H. Salus, “A Quarter-Century of Unix,” 1994.

*Rebooting India*, the book that details the development of Aadhaar and other platforms, contains a quote from the former Indian President Pranab Mukherjee comparing technology platforms with those of a railway system:

See, it is just like a railway platform. Different trains pull up at a railway platform, each with a different destination, and people get on and off depending on where they are headed. In the same way, the technology platform is a central location where various state government, institutions and citizens can gather. All government services are offered on the same platform, and citizens can enrol for all eligible services in one place.<sup>128</sup>

The UK Government Digital Service tried various metaphors to explain a platform approach to government (many of which are listed below), before arriving at the concept of building blocks:<sup>129</sup>

We think there's a simpler, easier way. The same public services, but designed and built differently. It's an idea called "government as a platform." It breaks things down into smaller parts, like building blocks. Each block does one job. It's easy to connect blocks together, and scale them up when demand increases. If some part of the service breaks, we can fix it or upgrade it easily. Platforms can be opened up, too, allowing third-party services to use government data.<sup>130</sup>

Below are a few different approaches, metaphors and descriptions that may be useful when explaining the idea of platform government.

### ***Building blocks***

By creating reusable building blocks we can make it quicker and cheaper to deliver public-facing services. As a consequence of that, public services will improve, and the quality of people's experience of the state will be transformed.

.....  
128 Nandan Nilekani and Viral Shah, *Rebooting India: Realizing a Billion Aspirations*, Penguin Books, 2016, page xxviii.

129 Interview, Matthew Sheret, former Writer and Strategist Government Digital Service, 21st September 2018.

130 Government Digital Service, "Government as a Platform," YouTube, 9th December 2014, <https://m.youtube.com/watch?v=ZzPU6Pdw05s>.



***Duplication and cost***

Government is doing X thing multiple times. That's wasting money and is confusing for citizens.

***Digital infrastructure***

Platforms represent shared digital infrastructure for our country. A modern government needs to approach digital infrastructure in the same way it does physical infrastructure, such as roads, or legal infrastructure, like company law.

***A modern technology stack***

Modern technology companies like Amazon, Google and Facebook take a platform approach. Digital age governments are beginning to emulate them.

***The burning platform***

We already have a technology platform of sorts today. The question we need to ask is: is it up to the job? Do we have the capability in place to be able to respond to threats and to continuously improve it?

It's not about fixing technology, it's about fixing a policy area

Platforms are not about fixing our technology. They are a way of enabling us to fix a broken policy area (e.g., welfare, transportation).

***International advantage***

The first governments to adopt this approach will have an advantage over other countries.

***End-to-end service design***

Radically new types of service could become possible (or at least much simpler) under the Government as a Platform model. Services can more easily be designed around the needs of citizens rather than the organizational structure of government.

***Wholesale government***

Shared foundations mean that services can be provided by any layer of government, as well as by commercial or third sector organizations. They can overlap and complement each other.

***Institutional reform for the digital age***

If you designed government today, you wouldn't design it around siloed government departments.

***Transparency***

Platforms can be designed to be inherently more transparent than today's systems.

***Avoid lock-in***

Because platforms are modular, it will prevent us being locked in to a single set of technology.

**2. Show don't tell**

Ultimately though, there's no substitute for showing a working demo.



*A publication of the*

Ash Center for Democratic Governance and Innovation  
Harvard Kennedy School  
79 John F. Kennedy Street  
Cambridge, MA 02138

617-495-0557  
[www.ash.harvard.edu](http://www.ash.harvard.edu)