Public Health Agency of Canada | Agence de la santé publique du Canada



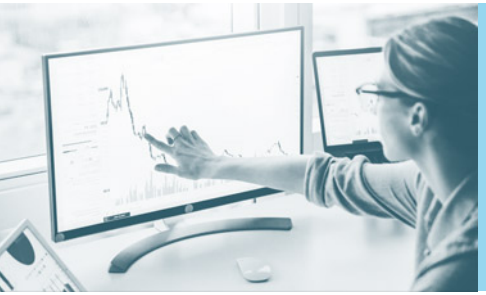
DATA SERVICE TEAMS
FIELD GUIDE

1.2.0 (E)



Canada

SECTION ONE



WHAT IS THE **DATA IMPERATIVE**?

1

The rationale for *Data Service Teams* and an overview of the risks to be managed in data-intensive, evidence-based public health

SECTION TWO

WHAT ARE **DATA SERVICE TEAMS?**

7

A basic description of *Data Service Teams* and data services, plus the logic and touchstones that guide design and development

WHAT MAKES PUBLIC HEALTH DATA DIFFERENT?

SECTION THREE

HOW ARE **DATA SERVICE TEAMS** ORGANIZED?

24

The organization of *Data Service Teams*, including roles and activities, with an emphasis on how teams manage themselves

SECTION FOUR

HOW TO **UNLOCK THE POTENTIAL** OF DATA?

35

Data pipelines, partnership networks, and technological architectures that enable new services to be built quickly and robustly

whatever the sources + need to bring in epidemiological data in, to feed more sophisticated forms of analysis can take place

maybe, what is public health data and how is it used section

SECTION FIVE

HOW DOES **THE PROCESS** UNFOLD?

57

How a hypothetical service project is likely to proceed over time based on the logic model, touchstones, and various logistics

SECTION SIX

WHAT **ADDITIONAL RESOURCES** CAN HELP?

73

A compendium of "springboards," or tools and readings for team members who want to delve more deeply into specific topics

HOW TO USE THIS GUIDE

This guide was designed with several audiences and purposes in mind. It is worth reviewing a few design features aimed at putting the guide to best use.

Who is the audience? The primary audience are those who are working in *Data Service Teams*, either as core members or collaborators. Consider the guide as an on-boarding resource to help everyone get up-to-speed. It is also an evolving compilation: a one-stop roll-up of the ideas and techniques used by the team showing what works and what does not. An extended audience within the Agency will also find the guide useful as a general learning tool to modern approaches to data management and service design. Indeed, others in the Public Service (and beyond) may draw inspiration from the methods discussed. To facilitate that broad appeal, the guide is not overly officious in style, nor does it assume familiarity with the Agency.

Thought went into the composition of the guide to make it an easy read and practical reference. Each page is devoted to a major topic. Sidebars delve into issues of extra importance. A keyword highlighted in each paragraph makes the text easier to parse. Annotated visuals appear throughout.

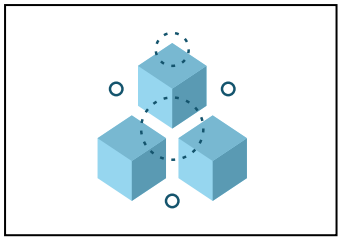
Moreover, different types of pages cover different content. These have been colour coded to make the guide easier to scroll through (see figure). That modular design makes it easier to insert new material and share individual pieces. More importantly for the busy reader, the space constraint means that topics are discussed concisely.

Live links (🔗) can be clicked to open additional material if reading the digital version on a device connected to the Internet.

If you have any inquiries related to this guide, feel free to reach out (@).

Topic Pages

Every key topic is covered in a single page, with an item worth highlighting discussed more fully in the sidebar.

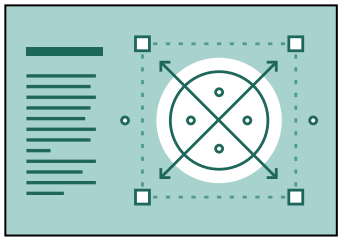


Model Pages

Major analytical models that provide an overview or detailed explanation are covered in seperate pages.

Focus on Roles

Key positions and jobs in the process are discussed in greater detail on special sections.



Focus on Process

Playbook plays and dialogue templates of particular interest are explained, with visual aids.

Springboards

Annotated recommendations for further learning are offered, with emphasis on technical materials.

1 WHAT IS THE **DATA IMPERATIVE**?

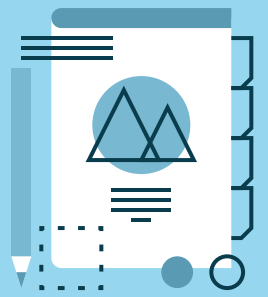
CHANGING THE WAY WE BUILD ADVANCED DATA CAPABILITIES

Welcome. If you picked up this **field guide**, the chances are you are about to collaborate on a *Data Service Team*. Why bother reading a guide about collaboration when you have been collaborating your whole career? Sadly, most technology projects fail to deliver services that impress. The services tend to arrive late and over-budget. They work awkwardly and frustrate users. Unfairness and bias get baked in. The wrong problems are addressed and the services quickly fall into disuse. A few are so clunky they become a major public embarrassment. Thus, a big dose of humility is advisable before rushing to code the next data dashboard or mobile app. This is hard work.

Why begin on such a sour note? Co-creating worthwhile services that people cannot do without is **a solved problem**. We rely on well-designed data services all the time in daily life. Perhaps it is the health app on your phone that counts your steps "auto-magically" and gently encourages you to move a bit more. It could be the weather app you glance at before stepping out the door.

These effortless experiences make awkward services all the more glaring. Mature methods of *client-centred service design* and *rapid development* guided the creation of our most-loved services. Those methods are a major departure from "business as usual" within the Public Service workplace. Indeed, the old habits and tired frameworks of public administration are a major impediment to getting things done in the semi-autonomous, multi-disciplinary, fast-moving team *you will be working in*.

This field guide is here to help. It is a **crash course** on this new way of building. Most collaborators will not have spent any time working in a design studio nor entrepreneurial start-up, the spaces where these methods are usually applied. Even experienced designers and developers will have to rethink how such methods apply to public-health data services. Indeed, there are three big areas of risk that this type of data service has to contend with. It is worth going through each one and considering their implications for your specific project.



THE FIELD GUIDE

A field guide is a compendium of practical advice for taking up a challenging pursuit. It is not some sort of cookbook crammed with step-by-step recipes. To the contrary, for any sufficiently complex challenge, there is little value in presuming too much and implementing a rigid one-size-fits-all approach. Thus, this field guide offers guidance about exploring complexity thoughtfully without feeling over-whelmed or cast adrift.

Despite the flexibility inherent to these methods, there are a number of fundamentals that every project will have keep top-of-mind throughout the process. In other words, crucial touch-stones (such as client involvement) are not luxury options to be added if convenient. They are necessary for the project's success. By the same token, there are dysfunctional patterns of work with an abysmal track-record that have to be called out. Both will be highlighted throughout this guide.

A. **DATA STEWARDSHIP**

Data is how a public-health agency observes the state of a population's health. Without accurate, timely, and reliable data, decision-making becomes guesswork. Unfortunately, a lot of health data goes **under-utilized**. How so? Perhaps the data lacks relevance to the challenges at hand. Methodological shortcomings undermine usefulness. The data may not go to those best able to take advantage of them. It may not arrive in a form that is genuinely useful. The job of the *Data Service Team* is to spearhead the more effective use of data.

A **data service** is not just managing a database and reporting out key findings. It is the end-to-end stewardship of data to ensure the right data is flowing to the right people at the right times in a form that best serves their practical needs, all in a way that furthers public-health objectives. Advanced technologies are crucial enablers but are not the whole story. Basic assumptions about the subject matter will have to be revisited. How the surveillance program works will likely be tweaked. New techniques for explaining with data will inevitably be involved. Most importantly, the range of users may expand. For example, it might make sense to feed the data directly to point-of-care practitioners, not just epidemiologists and policy-makers at headquarters. The data may even be put into the public's hands so that everyone can make intelligent choices for themselves.

The challenge is that data flows through **elaborate supply chains**. Cases are measured at one end of the chain. At the other end, findings are used to change minds and alter behavior for the sake of improving health outcomes. In between, the data is processed in a variety of ways (see page 37). Any bottleneck is a costly delay. Yet the in-between steps often happen outside of the Agency: other governments; non-governmental organizations (NGOs); laboratories; point-of-care facilities; the list goes on. Thus, stewardship involves leading complex networks of partners, many of whom will have to be involved in any change initiative. All those supply-chain logistics have to be managed too.

THE BIG DATA ERA

Advanced **data science** techniques are opening up entirely new fields of health research. Computational genomics, agent-based simulations, and machine-learning algorithms are just a few examples. They all rely on "big data," large quantities of automatically generated data. Stewardship also involves exploring the potential of these advancements for public health in Canada.

Any new technique will add risks and raise ethical issues. The hype surrounding technological breakthroughs often benefits from a "reality check" and a big-picture perspective. Thus, a critical eye has to be applied to ensure that technological advancements do not cause more trouble than they are worth.

Those working in *Data Service Teams* will have to take an active interest in these new developments and think creatively about how they can be applied most appropriately.

B. **PUBLIC TRUST & CONFIDENCE**

The second set of risks relate to **public expectations**. The public looks to government for guidance about coping with health challenges. It is expected that decisions are made quickly on the basis of high-quality, scientific evidence. That is especially true in an emergency, even though some uncertainty is to be expected. Mixed messages and awkward evasions only undercut the credibility of public officials. That opens the door for politicization of health data and faddish interventions, which only sow confusion. Building a reputation for trustworthiness and capability has a lot to do with framing of health issues adeptly with the help of data.

That legitimacy imperative puts an onus on the Agency to better anticipate challenges and make sure adequate data is being generated in preparation. Yet even with the best data at hand, the facts do not just speak for themselves. Findings have to be made **meaningful** and practical.

New techniques exist for communicating complicated ideas through **data storytelling**. Interactive data displays can give lay audiences direct access to meaningful data tailored to their circumstances. Apps on mobile gadgets allow people to use data to complete daily tasks. Rich information graphics, such as those found in science magazines, make technical subjects easier to relate to. Motion graphics and videos can step people through a complicated storyline very quickly. Tailored alerts can reframe expectations of emerging dangers. All these techniques (and others) constructively frame health issues and make them less abstract.

Such techniques can also be deployed badly. They can be used to dumb down issues and market glib messages. Some applications can be more style than substance. Thus, care needs to be taken to make sure that data stories are told with **scientific integrity**. That requires new skills in communicating science and medical knowledge. Moreover, the media chosen has to match the task at hand. Creating a new app or info-graphic should be done for the right reasons to suit the right audience, not just done because it is trendy.

STORY TURBULENCE

Public health messages circulate in a competitive arena. The rush to claim certainty from ambiguous findings can be overwhelming. Snap interpretations of data (“hot takes”) abound. Outright misinformation can capture the public imagination faster than fact-checkers can respond. Amid all that “story turbulence,” mixed messages, cryptic jargon, and awkward disclaimers from public health channels only make matters worse by encouraging unhelpful speculation. Just reporting raw data is not enough.

The solution is twofold. First, decision-makers should have the latest, high-quality data at their fingertips to quickly frame public health issues as they arise. Second, the evidence has to be meaningful, crafted into data stories that everyone can relate to. Even better, the stories can be personalized to cut through the noise and speak to individuals and their unique circumstances.

C. **ACCELERATING CAPABILITIES**

The third big risk involves internal coordination and capability development.

Data touches on virtually everything the Agency does. Good data stewardship is a shared responsibility. Unfortunately, the skill sets involved are difficult to acquire. Thus, *Data Service Teams* are not only tasked with working on projects but also building capabilities along the way. Acquiring rare skills (such as programming) can be a struggle for a surveillance program. That may mean that some members of the team stay with a data service long after it is in place. A data service is never truly "finished" insofar as there will always be further improvements to implement. Seen in this way, *Data Service Teams* are instruments for accelerating the development of **technical capabilities** across the Agency.

Moreover, building data services often involves cultivating a larger ecosystem of stakeholders with an interest in advancing the state of public-health data. Much of the work of *Data Service Teams* involves bringing others into the design circle as genuine co-creators. At minimum, users of the data are involved. A wider array of stakeholders can also be invited in, especially those involved in data pipelines and those on the cutting edge of health research. Thus, the teams also accelerate capability development across **diverse networks** involving those outside the Agency who can contribute to better health outcomes.

Part of that challenge involves identifying where data infrastructure needs to be built-out and finding opportunities to **streamline processes**. Any data service can be thought of as a value chain, with important contributions made at each stage in the relay of data (see sidebar). *Data Service Teams* help instill an ethos of service where everyone plays a role in removing undue burdens and pain-points that get in the way of the effective use of data. That is how more timely data gets delivered reliably.

All told, even though *Data Service Teams* build in quick bursts of creative energy, they are an important linchpin role in developing lasting capabilities across the Agency.

VALUE CHAINS

A lengthy process of creating and using data can also be thought of as a **value chain**. In other words, each stage in the processing of data is expected to demonstrate a valuable contribution to the whole. That includes support functions that are not directly involved in data supply chains. Improvements in timeliness, quality, and meaningfulness at each stage help the overall chain work better. Empty busywork and low-value bottlenecks are weeded out. Investments in new platforms and methods can add major efficiencies.

Moreover, each line of surveillance and data service should be able to articulate its **value proposition** to those who use the data. How is the data used to improve real-world health outcomes? What risks are amplified by not having the data? Answers to such questions should readily come to mind when experiencing a service.

STRATEGIC OBJECTIVES & RISKS

DATA STEWARDSHIP

GOAL. Using data to its full potential because it goes to the right people at the right times to accomplish tasks that further public-health objectives

RISK. Crucial sources of data go untapped and existing data is chronically under-used because it fails to meet client needs

- **Relevance.** Right data goes to right people
- **Fit.** Data comes in form that is useful
- **Meaningful.** Findings are easy to interpret
- **Timely.** Data arrives soon enough to act
- **Complete.** All the data needed is available
- **Focused.** Data drives right policy outcomes
- **Open.** Data made accessible by default
- **Portable.** Data and methods described

PUBLIC TRUST & CONFIDENCE

GOAL. Upholding high standards of ethics and scientific integrity while relating findings through data stories that audiences understand and trust

RISK. Awkward reporting of data that sends mixed messages and confusing advice which undermine the credibility of expert authorities

- **Competent.** Demonstrates expertise
- **Clear.** Messaging is accurate, accessible
- **Scientific.** Balance of evidence reflected
- **Forthright.** Humble about limitations
- **Pathbreaking.** State-of-the-art methods
- **Ethical.** Takes care to be fair and inclusive
- **Unbiased.** Findings not skewed untowardly
- **Verifiable.** Findings can be checked
- **Provinenced.** Origins and changes are clear
- **Privacy.** Personal details not divulged

ACCELERATING CAPABILITIES

GOAL. Building robust data services quickly in response to emergencies and emerging priorities by cultivating data capabilities across the Agency

RISK. Hacky services that take too long to build, lack important features, are prone to disruption, and otherwise offer an underwhelming client experience

- **Speed.** Services are built quickly
- **Responsive.** Various needs are catered to
- **Reliable.** Built on firm foundations
- **Client-centred.** Data users put first
- **Goodwill.** Process promotes good vibes
- **Careful.** Burdens not shifted onto others
- **Sustainable.** Services viable in the long term
- **Efficient.** Effort is not wasted
- **Secure.** Protections are made inherent

2 WHAT ARE **DATA SERVICE TEAMS?**

DATA SERVICE TEAMS

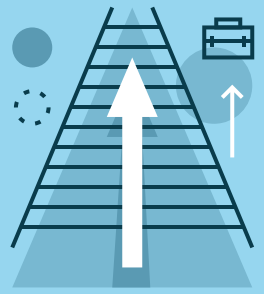
To recap, a **data service team** is a diverse group of co-creators who come together to design and develop ways of improving public health through the use of evidence. These innovations are generated quickly, even if fundamental changes are made to how the Agency stewards data.

The team is a "platoon on a mission," a diverse group brought together for a short-term project. It partially disbands as a solution is getting ready to go live. All the while, the team **builds capabilities** amongst those who normally run the program. The aim is not to improve the way data is used once and for all, but to equip a program with the ethos and skills to continuously improve long after a new solution is in place.

Team members come from all over the Agency to assemble the right mix of talents and experiences for a particular project. Others with specialized skill may be brought on-board as the need arises. All high-quality services are **client-centred** insofar as they prioritize the varied needs, perspectives, and abilities of would-be users. Therefore, data service teams invite clients (and key stakeholders) into the design circle as genuine co-creators by using dialogue-based collaboration methods.

The team explores the art-of-the-possible to create data services that clients gladly rely on. The discipline of **service design** includes a variety of methods for delving deeply into the nature of problems, reframing the issues, and coming up with imaginative options. Team members may not be used to working within a free-flowing, design-studio milieu. Thus, some reorientation is expected as everyone on the team becomes familiar with a different toolkit of approaches.

Fledgling idea are quickly turned into working services. Making is not just "implementation," but a valuable generator of ideas in its own right. Once a viable version is up and running, it can be revised through real-world testing. That **rapid development** approach was pioneered within high-tech start-ups to mitigate project-management risks and avoid lackluster services. It also increases the likelihood of a blockbuster success.



LADDER OF INNOVATION

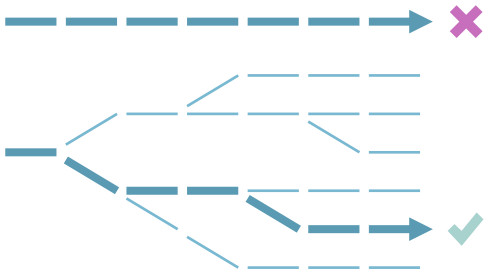
When faced with a challenge, the temptation is to do a quick "one-and-done" fix. That may be what circumstances call for. However, every challenge is an opportunity to transform the way we do things in a more fundamental way. *Data Service Teams* are organized to make sure those opportunities are not passed up. Team members ask themselves:

- How can we **do things better** to improve the way things work (incremental improvement)?
- How can we **do better things** to achieve our goals (major leaps)?
- How can we **choose better goals** to advance our mission (comprehensive rethinking and redesign)?

Asking those questions encourages everyone to explore problems deeply and strive towards ambitious innovations.

DESIGNING THE DESIGN PROCESS

Design processes are not pat formulas with everything planned out in advance. Methods are applied flexibly as the team explores the problems with humility. Instead of imagining a linear sequence, think of it more like a tree.



Thus, the first goal of the team is to **design the design process** and continually adjust it as circumstances demand.

Each line segment represents a different *play* (or activity) selected from a **playbook**. All sort of methods are available to move the project along but not all will be relevant. For instance, a team may have to generate new ideas to solve a particular problem (*ideation*). Several brainstorming exercises could be conducted. Many sources of inspiration could be consulted. The setting in which deliberations take place could be shifted to expose everyone to novel influences. Experiments could be run. Ultimately, the team selects the most promising options, leaving many others on the table. The simple act of trying new things will dislodge habitual ways of thinking about the problem. Many playbooks are available to suggest plays (page 77). They help teams get unstuck when struggling to come up with ideas. They also expose teams to considerations that may not have otherwise come to mind. As the team learns to work together, it may develop its own plays and pass them along.

Even with lots of improvisation and optionality, there is an **overall logic** to the Agency's design process. It has two parts. First, there are several essential touchstones that guide decision-making throughout the process (see sidebar). Second, there is a general logic model—called the *double diamond*—guiding the sequencing of activities. The rest of this section is devoted to explaining these organizing principles.

ESSENTIAL TOUCHSTONES

Regardless of how the process is organized, it remains oriented by nine crucial touchstones:

- Design thinking
- Human-centred Design Thinking
- Agile Development
- Participatory Design
- Open Data
- Evidence-based Policy Making
- Data Management
- Data Storytelling
- Systems Thinking

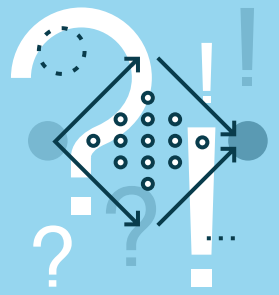
Each of these touchstones is taken to heart while making decisions at every point in the process: none are luxury options. Teams are free to organize themselves as they see fit so long as all of these touchstones are put into practice, rather than just going through the motions.

A. **DESIGN THINKING**

Design involves prefiguring how something is made and could be made better. That entails three things. Design is a *sensibility*, or a discontent with the merely adequate and an active interest in finding clever alternatives. It is also a *way of seeing*, or reflecting on the implications of small details while also being mindful of how everything works together as a whole. Design is also a *body of practical knowledge* about what works and what does not within a particular field. For example, within *user-experience (UX) design*, a community stewards best practices related to human-computer interaction, user interfaces, human variability, and so forth.

Everything made by humans is designed but not necessarily designed well. **Badly designed** products and services work awkwardly and are error prone. Worse, they cause unintended side-effects that have the potential to do harm. What causes bad designs? The wrong problems are solved because everyone was too quick to jump to solutions. Points of contention are glossed over. Sub-optimal trade-offs are made for the sake of expedience. Not enough homework was done into the substantive issues and usage scenarios. Diligent designers avoid such problems by methodically thinking through a challenge. The team will inevitably face constraints and difficult trade-offs. These are not excuses to do less. They are opportunities to rethink how things are done and trade-offs are handled with ingenuity.

What does the **process** entail? There are bouts of exploration and idea generation. There are also bouts of synthesis and refinement. Such divergent- and convergent thinking (see sidebar) are organized into group activities. Brainstorming as a group is crucial for making sure everyone is on the same page. That is not merely a routine check-in meeting, with everyone then doing their own tasks in isolation. The point is to think through issues together because each team member brings particular talents to the table. Giving everyone a voice in deliberations also generates “buy in.” Only once collective deliberations have run their course do team members work on their own.



DIAMONDS OF TEAM DELIBERATION

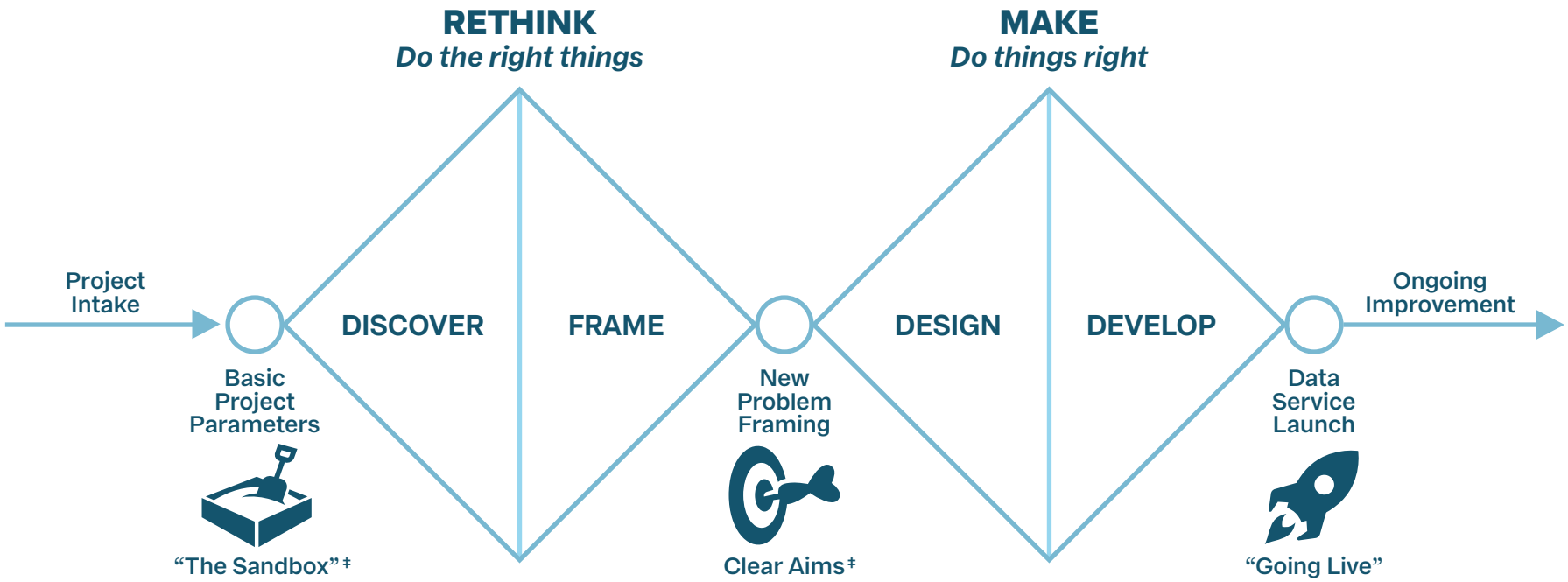
Exploring a *problem space* involves different thinking styles.

Early on, judgement is suspended to consider a wide variety of ideas with an open mind. New ideas are thought up. Hunches are sussed out. Interesting ideas are sought from unfamiliar sources. All that generation of possibilities is **divergent thinking**.

At some point, scrutiny is applied to the array of ideas on the table. Promising ones are refined and built upon. Less practical ones are reformulated or set aside. Complementary ideas are brought together and made workable. All that selection and synthesis is **convergent thinking**.

The “diamond” is a visual metaphor for successive rounds of divergent and convergent thinking. In practice, design deliberations involve several rounds, or multiple diamonds.

DOUBLE-DIAMOND MODEL†



Divergent Thinking: Exploring the problem space and revisiting basic assumptions

- **Orientation** to design thinking (How to collaborate better?)
- **Stocktaking** (What has come before?)
- **Field research** (What do we need to know?)
- **Client awareness and involvement** (Who are clients? How to involve?)
- **Context of use** (What are the clients' worlds?)
- **Surfacing issues** (What problematic assumptions?)

Convergent Thinking: Synthesizing a new understanding of the problem and clientele

- **Client Profiling** (What types of clients and orientations?)
- **Client Journey** (What steps in the service workflow?)
- **Reframing** (How to think differently about the issues?)
- **Foresight** (What emerging issues and ongoing trends?)
- **Synthesis** (What is the nature of the problem to be over-come and design priorities?)

Divergent Thinking: Exploring promising options for solving the problem

- **Ideation** (What potential solutions?)
- **Constraints** (What practical limits?)
- **Human Factors** (How is diversity accounted for?)
- **Ethics** (What moral dilemmas? Potential harms? Exclusions?)
- **Mock-ups** (How to make tangible?)
- **Data Needs** (What new data sources?)
- **Service Channels** (What media and technology?)

Convergent Thinking: Selecting a design and rapidly building a working version

- **Rapid Prototype** (How is best option put to the test?)
- **User Experience** (How does whole service work from clients' perspectives?)
- **Architecture** (What infra-structure is needed?)
- **Openness & Security** (How is data access and integrity handled?)
- **Reuse** (How to share data and technology?)
- **Field Tests** (How to test with real clients in realistic scenarios?)
- **Alpha Build** (How to get the service running?)

† The generic version of this model was popularized by: Design Council UK, Eleven Lessons: Managing Design in Eleven Global Companies (London: Design Council, 2007) p. 10.

‡ See page 59.

B. **HUMAN-CENTRED SERVICE DESIGN**

Designers have **orienting priorities**. The best services start from the vantage point of the client and work backwards from there, rather than just doing what is most convenient for administrators of the service.

Who are the clients? Even though *Data Service Teams* are an internal service, the surveillance programs requesting new projects are not the clients per se. They are suppliers of data. They become members of the team as co-creators. The actual clients are the **users of the data**. At the end of the day, they are the ones who have to be satisfied with the data service.

A distinction is made between *direct clients* and *downstream clients*. Direct **clients** are the ones who interact with the data to make decisions and complete tasks. A data service has to cater to their task flows (and the context of use) to be useful. For example, direct clients may be policy analysts or epidemiologists. They feed data into analytical models to generate findings and advice. Who are they serving? That is, who are the client's clients? A data service has to cater to them too. These are the ultimate beneficiaries of the data. If the data service does not account for their needs and circumstances, the desired health outcomes will not likely materialize. Moreover, it might make sense to turn downstream clients into direct clients, giving them direct access to the data to make decisions on their own.

Human-centred design does not presume an ideal client. Instead, designers look at real clients as they actually are, flaws and all. Plus, there is no such thing as an "average" client. A client-base will be diverse, with clients varying in their needs, wants, situations, interests, aptitudes, social supports, and personal quirks. These may have to be accounted for in the design of the service. For example, a large share of the population cannot see colours. Thus, when data are presented, charts and graphs have to use a colour palette that accommodates them. Prospective clients will have to be consulted as a reality check. Versions of the service will be tested with them. Better still, clients can be involved in design decisions.

CLIENT EMPATHY

Empathetic teams are sensitive to realities of the diverse users of data. There are three sides to empathy:

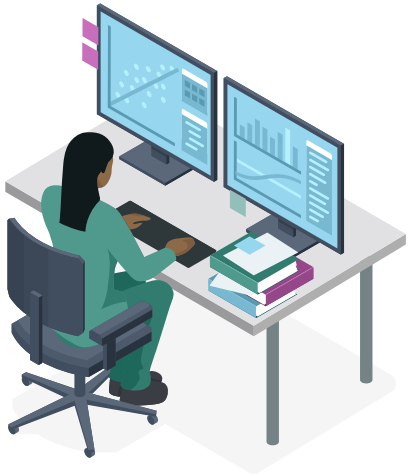
- considering matters from the perspective of others, or the proverbial walking in another's shoes by seeing things from their vantage point (**empathetic reasoning**);
- taking an active interest in the welfare of others, or caring about their situation and fortunes (**empathetic concern** or compassion); and,
- appreciating the felt experiences of others, especially as they undergo difficulties (**emotional empathy**).

Most people over-estimate their ability to be truly empathetic. Moreover, there are few occasions for designers and developers to socialize with the actual users of data. Thus, extra effort is required to learn what clients go through when using a service.

CLIENT-CENTRED FOUR-STEP

WHO ARE THE CLIENTS?

Who will use the data directly? How many different groups (or *segments*) of client are there? For example, clients can be differentiated by role (such as policy analysts, emergency managers, or epidemiologists). For each segment, what are their needs, wants, situations, interests, aptitudes, social supports? For example, how data literate will members of a segment tend to be? How do they think about the subject-matter? Or existing data services? How can we involve clients in the design process? What aspects of the service are likely to matter most to them?



WHAT IS THE USAGE SCENARIO?

What tasks is the client performing with the data? What contextual factors complicate those tasks? In what forms does the data need to arrive to be most useful? What ways can the data be presented to be most meaningful within the task flow? What sort of service experience do you want clients to come away with? What are the steps in that journey? How is the service discovered and accessed in the first place? How can the data service be made integral to how a client works?

WHO ARE THE CLIENT'S CLIENTS?

Who are clients trying to serve or influence directly? What are the data needs of these downstream clients? What are they trying to accomplish? How can they be grouped into segments for purposes of the data service? What are their needs, wants, situations, interests, aptitudes, and social supports? What can be added to the data service to help clients help clients? How can the data service be changed to cater to downstream clients directly?

WHAT IS THE END GAME?

What do you ultimately want people to do? What patterns of thinking and behaviour are you trying to change for the sake of public health? What are the ultimate public health goals this data service will be contributing to? What does success look like? How will you know that the data service is making the right contribution? And not resulting in adverse consequences? What sources of feedback are available to alter the way data is used?

C. **PARTICIPATORY DESIGN**

How are those outside the Agency involved as contributors? Genuine participation is not just an occasional consultation to acquire information and consent. Clients, data suppliers, health experts, and other **stakeholders** usually have insights about making a service better that only emerge when they are treated as collaborators. They are brought into the *design circle*; that is, bought into the deliberations where actual decisions are made. Not only does that involvement make a service better, it turns everyone into advocates for the service.

That is where participatory design techniques come into play. All the important decisions in a *Data Service Team* are made within dialogue sessions (see sidebar). Those with a biggest stake in the service's success are involved (see next page).

A few misnomers about involvement should be dispelled up front. First, clients and other stakeholders are not simply asked what they want from a service. Such questions rarely yield useful feedback because the possibilities do not readily come to mind.

Many will also have narrow interests that have to be balanced for the sake of the public good. Moreover, these are people who lead busy lives and their time has to be respected. They cannot be involved full-time. Thus, the timing of their involvement has to be carefully coordinated. Too often, clients and stakeholders are consulted on unimportant details while treating the big decisions as a *fait accompli*. Instead, they should be brought in for the discussions that matter to them the most.

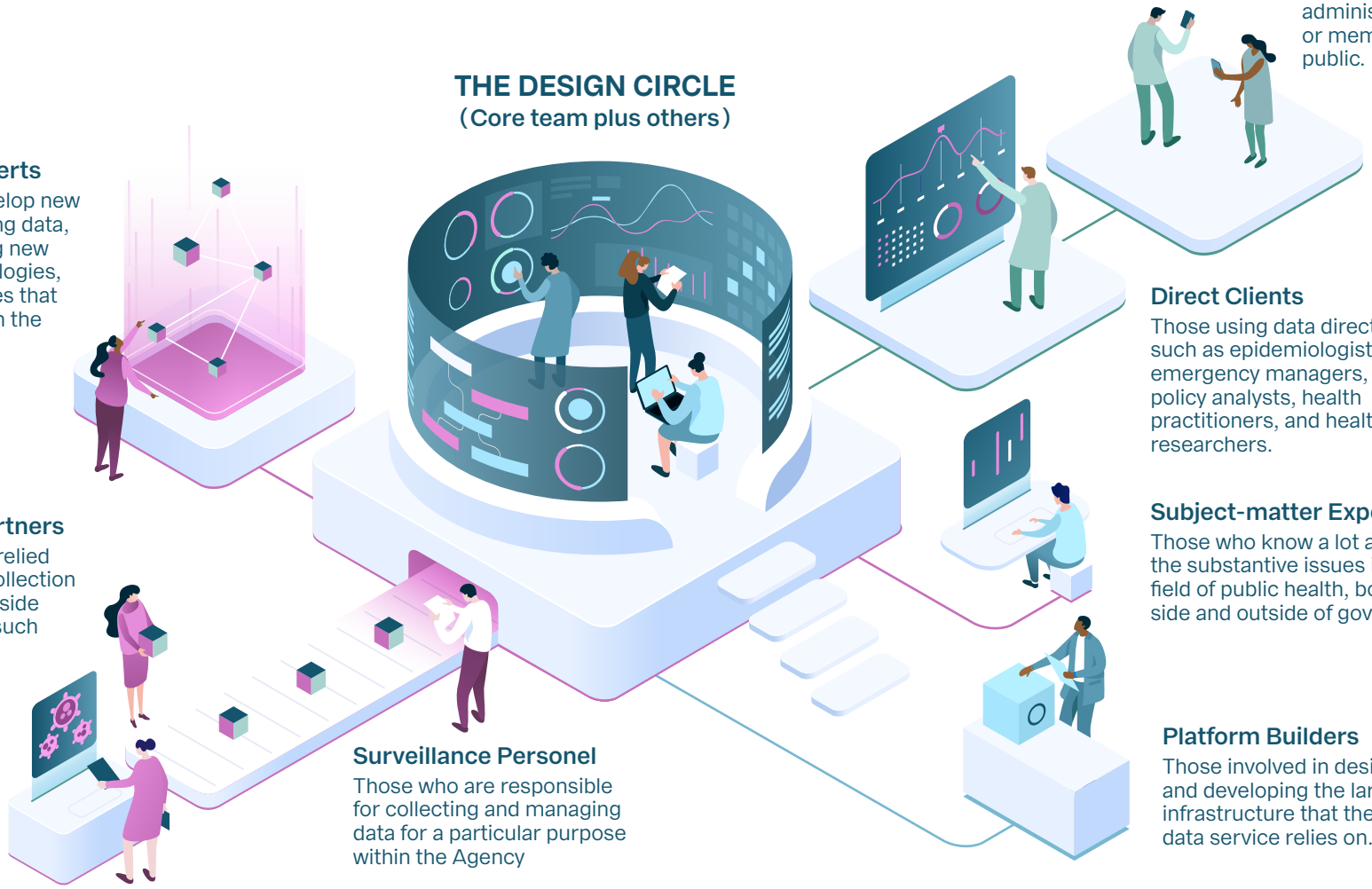
Often the *Service Design Team* will go to where the participants are, instead of *visa versa*. For example, a behavioral scientist on the team might explore the worlds and mindsets of clients. They will see first-hand how data can be best used given the various contextual constraints and demands involved. Likewise, fledgling services will be tested in the field by developers to see how things work with prospective clients under real-world conditions. That is the opposite of the "build it and they will come" mentality that causes services to misfit demand.

DIALOGUE

Dialogue is an open and earnest conversation in which everyone participates on an equal footing. Rank is set aside and the quality of everyone's contributions is judged on the merits. A diverse group can then learn from each other by pooling their personal talents and experiences. Disagreements are inevitable. Those are worked through in a spirit of generosity instead of scoring debating points. To prevent the conversation from veering off along unproductive tangents, focus is maintained by putting *making* (the creation of a service) at the centre of deliberations. Project considerations are broken down into smaller pieces and discussed separately. All the while, the goal is to avoid premature closure of discussion that creates a false sense of consensus. Matters are hashed out for as long as necessary or are parked as loose ends to be dealt with when the timing is more suitable.

INVOLVEMENT MODEL

Besides the core group, a Data Service Team brings others into the design circle. Specialists involved in the surveillance program are brought on-board a co-creators. Some clients and partners are brought in as well, with the participation made more focused on what matters to them. Other experts and builders are brought into the circle depending on the nature of the challenge at hand and the needs of the moment. Their level of involvement may vary from project to project depending on team needs.



Downstream Clients

Those who rely on advice generated with data, such as policy-makers, media influencers, health-care administrators, patients, or members of the general public.

Direct Clients

Those using data directly, such as epidemiologists, emergency managers, policy analysts, health practitioners, and health researchers.

Subject-matter Experts

Those who know a lot about the substantive issues in the field of public health, both inside and outside of government.

Platform Builders

Those involved in designing and developing the larger infrastructure that the data service relies on.

Methods Experts

Those who develop new ways of analyzing data, such as building new models, technologies, and data sources that may be of use in the data service.

Upstream Partners

Those who are relied upon for data collection but who are outside of the Agency, such as NGOs, other jurisdictions, and personnel in point-of-care facilities.

Surveillance Personel

Those who are responsible for collecting and managing data for a particular purpose within the Agency

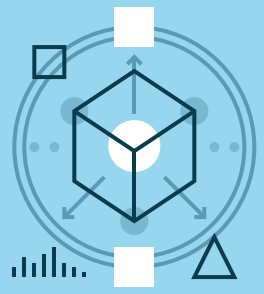
D. **RAPID DEVELOPMENT**

The abysmal track-record of technology "mega-projects" has lead to a major rethinking about how development projects are managed. Gone are the days when development would be intricately planned out in advance, with exhaustive specifications and elaborate timelines. **Rapid development** is now organized around the *Agile model*. Small teams develop services iteratively with extensive client involvement. Workable software is created quickly so that it can be scrutinized and tested in a hands-on ways with quick feedback. Received wisdom about constraints is challenged. Teams repeatedly revisit assumptions and are able to quickly change course if something does not make sense; better to fix what is not working early instead of finding show-stopping flaws after everything has been built. Above all, Agile is *adaptive*: the approach continually adjusts as understandings of the service evolve.

The way Agile works in government is not exactly the same as the approach found in lean business start-ups. Official guidelines add a few extra hurdles to the process.

Nevertheless, the **process-logic and general spirit** remains. Software architecture is broken down into modules. That allows existing code to be refactored for present purposes and reused. Newly created code can also be made available for other projects. New modules are created swiftly in quick bursts of activity. Integration of modules into the larger whole happens continuously. Team members check in with each other regularly to compare notes and coordinate amongst themselves. All the while, the team does not lose sight of the ultimate goals of the project.

For Agile to succeed, time has to be set aside for **learning** or else it will just be a case of "doing the wrong things faster." That starts by not jumping into coding but exploring the problem inquisitively. If uncertainties exist, research and experiments are conducted in the same spirit of rapidity. Lessons are drawn from prototypes and workable versions of the service. Even after the service is launched, the strains placed on the service by actual clients provides new feedback with which to make improvements.



RAPID PROTOTYPING

Most breakthrough ideas are not imagined out of the blue but emerge from the process of making. Many promising ideas prove to be unworkable when implemented. Even viable ideas are difficult to explain and have to be shown to get the message across. Thus, there is a benefit to testing out ideas as mock-ups and prototypes as early as possible.

Mock-ups are visual renditions of an idea to make it more tangible, including sketches and physical models. **Prototypes** are rudimentary versions of the service that demonstrate some functionality. For example, prototyping software can be used to simulate services, perhaps with actual data. Role-play with paper-craft models also counts as a prototype. Both mock-ups and prototypes solicit feedback that is more concrete, less abstract. If the prototype ultimately disappoints, it is done in a "fail fast" approach to risk management, then on to the next idea.

E. **OPENNESS**

Access to quality data makes any data service possible. However, data collection is highly distributed: no single organization controls all the data necessary to promote public health. **Sharing data** is in everyone's collective interest. Building a data pipeline may involve negotiating access to someone else's data source and securing their ongoing cooperation. In turn, no single institution has all the in-house expertise to make the most of available data. Indeed, it is not always obvious who is best placed to offer break-through insights. Only once data is shared broadly do some of its most valuable applications emerge.

Sharing extends to other building blocks of the service. Much of the software code required to build a service quickly and securely is repurposed from elsewhere, drawn from online repositories. New code written for one project can be reused in other projects to save effort. Such **open-source approaches** to software development also bring outside scrutiny to code, making it safer, more trustworthy. Statistical models and practices are shared in a similar way, especially since they are now embedded within software code. Methods from elsewhere are reapplied, not reinvented.

Last but not least, science itself relies on the sharing of findings, methods, and data. Without openness, the peer review necessary to advance **scientific knowledge** would be lacking. The Agency relies on that knowledge to exercise judgement.

Making all that sharing easier is a priority for *Data Service Teams*. The **spirit of openness** informs how a service is organized. For example, there are benefits to building *open APIs* (Application Programming Interfaces), the connectors to data streams that data services plug into. Sharing becomes highly efficient as the data supply chain takes advantage of automated, always-on data streams, instead of relying on periodic, manual hand-offs of data files. Moreover, the lessons drawn from data service projects (including setbacks) are also circulated so that others can learn too. There is no longer any excuse to hoard information.

OPEN BY DEFAULT

Given all the benefits of sharing, the building blocks of data services (data, code, knowledge, and methods) are made open by default. If restrictions are necessary, the **onus** is on those adding restrictions to justify their imposition. What counts as "openness" is not entirely straightforward—the meaning of the term is debated. To some, it means "anything goes." To others, openness can be preserved while adding sensible precautions to sharing. There are privacy, security, and ethical imperatives. The providence of data and findings also has to be verifiable. Nevertheless, these are design constraints to be mitigated while retaining a spirit of openness. For example, instead of open data streams, trusted researchers may be given access to data within secure enclaves. Even if such constraints are necessary, they are done in a way that minimizes the burden on others.

F. **EVIDENCE-BASED POLICY-MAKING**

New challenges, methods, and data sources, are emerging all the time. *Data Service Teams* have to stay on top of those developments or risk data services becoming prematurely obsolete.

Public health policy is based on **empirical evidence**; that is, reasoned conclusions drawn from systemic observation (data). The gold standard of evidence comes from experimentation using the scientific method. Not all subjects are amenable to that mode of understanding and alternatives are required to expand the scope of inquiry. For example, epidemiologists and the social scientists rely on the analysis of population-level data. Even as the pool of data grows, some uncertainty will remain, especially amid fast-moving situations. Computational models, simulations, and estimates then provide the basis for making decisions instead of relying on fallible intuitions. All of these empirical methods have their strengths and weaknesses. Danger comes from ignoring methodological limitations. Thus, a good data service helps decision-makers rigorously assess the balance of evidence, while signaling where not to jump to premature conclusions.

The translation of findings into policy proposals is not politically neutral. In a democracy, elected leaders will be expected to infuse policy with political values while making judgement calls. By the same token, the evidence does not always speak for itself. Data has to be analyzed to generate findings, with findings turned into arguments about efficacious policy interventions. There is always a risk that **biases, blind-spots, and mental traps** will skew thinking. The goal is evidence-based policy-making, not policy-based evidence-making.

Any data service should be designed to reinforce good scientific practices and uphold democratic values. Promoting **scientific literacy** may be part of the service. *Data Service Teams* cannot lose sight of the larger goal of a data service, which is to improve public health outcomes in a verifiable way. That too involves the scrutiny of the evidence.

THE SCIENCE IN DATA SCIENCE

The era of big data has inspired new data collection and analytical tools. These methods show great promise for public health applications. Even so, there is a risk that analysts are mislead by an overabundance of data given that the tools make it easier to mistake “noise” for “signal.” Exploratory data-mining techniques infer patterns from the data, often with the help of automation, rather than apply the hypothesis-driven methods of **scientific inquiry**. Due caution has to be exercised when integrating these methods into workflows.

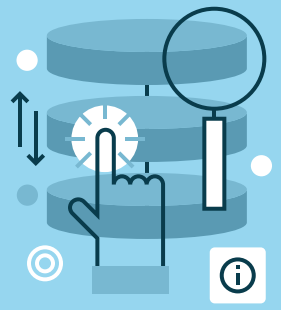
Moreover, data analytics for public health is heavily reliant on scientific knowledge for interpretation. The explosion of scientific publications makes it hard to keep apprised of the state of the knowledge in any field. Thus, **“knowledge mobilization”** is integral to a data service (page 54).

G. **DATA MANAGEMENT**

Services of all kinds are becoming more and more dependent on data. Data is treated as an asset, enabling the service to do new things to add value. However, data becomes a challenge to manage as it grows in volume and variety. For example, the logistics of data pipelines can get overly complex. Data can become systemically biased and used to make ethically dubious decisions behind the scenes. The threat of a data breach is ever present. Indeed, the inappropriate handling of data can compromise people's privacy, often inadvertently through routine divulgence of seemingly trivial facts. These challenges are amplified for *Data Service Teams* because they build services for which data is the core "product." That product has to be trusted. Therefore, good **data management practices** are a big part of what the team does.

Thankfully, the era of "big data" has finally given rise to **mature architectures and management methods**. Data pipelines are becoming more modular, transparent, and flexible, as well as less brittle and error-prone. Data usage can be inherently secure. Processes are becoming automated. Sadly, the public sector can be slow to adopt the latest generation of technological methods. It can take a while to upgrade skills and adopt the latest standards; it is too easy to just build systems using whatever methods have already been learned. For *Data Service Teams*, however, lagging behind the state of the field is not an option. Being stuck with adolescent systems while mature methods abound is an awkwardness the Agency cannot afford.

At minimum, data has to be made useful, timely, equitable, transparent, and evolving. Data has to comply with the FAIR principles of good data management (see sidebar). That happens with everyone on the team being mindful of the challenges associated with data throughout the process: data management is a **team sport**. It also involves building out a mature data infrastructure as part of how individual projects are managed. Those methods and architecture are given their own section in this guide (page 35).



FAIR PRINCIPLES

Data are managed according to four key principles represented by the acronym FAIR:

- **Findability** is the ability to easily discover, locate, and determine the relevance of data, with meta data, cataloging, and the like;
- **Accessibility** is the availability of the data for use without undue obstacles, with the onus placed on openness;
- **Interoperability** is the use of standard file formats that allow data to be used by a variety of applications (with no dependence on particular vendors); and,
- **Reusability** is the use data for multiple purposes, with a methodology and provenance that have been well documented.

These are foundational to good data stewardship and are mandated by the government-wide data strategy.

H. **DATA STORYTELLING**

Data is meaningless without **context**. When reporting findings, it is necessary to situate particular data points to guide interpretation. Why is this finding relevant? Is the magnitude large or small? Compared to what? Should the finding be taken at face value, or are there important qualifications that need to be considered? What are the real-world implications? Such questions should be answered as part of the act of communication. Too often, data displays are random jumbles of indicators; easily dismissable as "factoids" because they lack context and cohesion. For example, historical trends, comparisons, or scientific benchmarks are missing, with the reporting of findings lacking any sort of interpretive anchor.

Data storytelling involves adding narrative- or interpretive aids to the presentation of data in order to highlight findings and make them more meaningful. Sometimes the presentation of data is actually *story-like*: stepping the audience through a sequence of findings that additively convey a larger message, with illustrative examples making the data more relatable. Sometimes the data presentation is *map-like*: a display presents an arrangement of data points that do not have to be interpreted in a preset sequence; instead, the audience is free to peruse the data in a way that suits their own interests and circumstances. Sometimes, the data is *dashboard-like*: an instrument panel that helps guide the completion of tasks, with parts of the display drawing attention to noteworthy findings as they emerge. Which approach is best? That depends on the usage scenario, the audience, and the communications medium (next page).

All the chart- and graph formats used in office workplaces had to be learned. Thus, when presenting data visually to a general audience, it may be necessary to add suggestive cues and annotations (that is, **affordances**) to signal what a visual element is supposed to do or mean. Advanced data displays make use of novel visualization techniques which take extra guidance to figure out but, once learned, can be powerful tools for seeing what is going on.

VISUAL & MULTI-MEDIA ENGAGEMENT

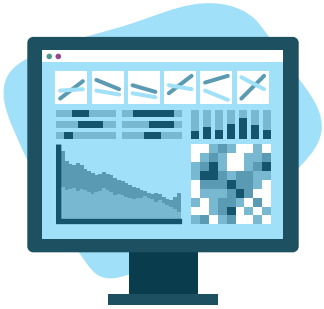
Humans are visual animals. In terms of both cognition and persuasion, it is more efficient to "show and tell" rather than just tell. When it comes to data, a table full of numbers can be very taxing to parse and interpret. Yet a well-chosen chart, graph, or illustration can make data meaningful at a glance. Making findings **"glance-able"** requires thoughtful design to guide the eye and add helpful cues. There is nothing intuitive about a rote "data dump."

Not everyone can take advantage of visuals. For example, eyesight degrades with age and a portion of the population is blind. Thankfully, our **multi-media** age offers a big toolkit of options. Visual displays have to work with assistive technologies. Moreover, accommodating special needs (by adding sound, for example) may result in a better experience for everyone.

TYPES OF DATA SERVICE

It helps to think imaginatively about what a data service can do for clients given particular usage scenarios. The following lists some of the most common functions and media.

FUNCTION



ANALYSIS

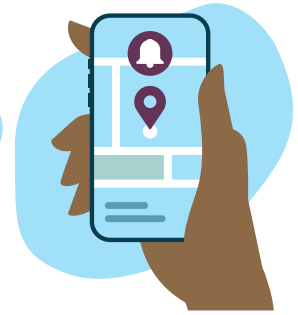
Visual displays of data that help make sense of complex patterns of data, such as data dashboards, interactive infographics, geo-spatial maps, and automated reports

REFERENTIAL

The storage and retrieval of data to complete a task, such as a search or directory services used to look up facts or point to better sources of information

MONITORING

Tracking activities within a domain and drawing attention to points (or patterns) of interest, such as services that alert people to note-worthy events and conditions, or emergent threats

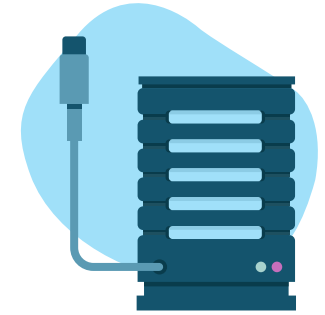


CONTEXTUAL

Services that sense (or are otherwise aware) of the context and are able to provide situation-relevant data, or even execute automated tasks depending on circumstances

VERIFICATORY

Confirming that state or provenance of an object of interest, such as a health-pass service that confirms a person's vaccination status



DATA ACCESS

Allowing people to tap into data for their own purposes, including experts gaining access or third parties building their own services, all while maintaining privacy provenance, and security

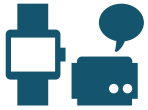
MEDIUM

Open Web

Mobile Apps

Kiosk or On-site Display

Messaging Service



New Consumer Gadgets

Physical Media

Secure Enclave

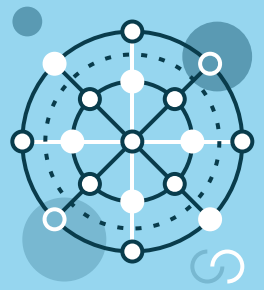
API (Self-serve Data Stream)

I. SYSTEMS THINKING

Public health policy-making operates in an inherently complex world. The human body is a complex self-regulating system. The subsystems that ward off threats to health (such as the immune system) are complex systems in their own right. Threats to human health, such as viruses and other microbes, evolve within complex ecosystems. Indeed, human society is an amalgam of complex systems that make patterns of behavior hard to predict. All that complexity presents a challenge to policy interventions. A problem cannot simply be fixed in a linear way, with clear-cut results expected from straight-forward interventions. Dynamic reactions create all sorts of unanticipated consequences, including harmful side-effects. Data helps policy-makers cope. Yet if data only captures a thin slice of reality, it can cause myopic thinking. Thus, data services use **complexity science and systems thinking** to approach policy interventions holistically.

Creating a data service is also an act of **system building**. Data services are full of interconnected parts and partnerships that are expected to work together seamlessly as a mesh. Overly complicated or finicky arrangements make data services brittle—vulnerable to breakdown and security threats. Goal conflicts and other chronic dysfunctions undermine a service's coherence and viability. Sources of ongoing feedback are necessary for the service to improve and adapt to the times (see sidebar). Any data supply chain has to have an ongoing source of perpetuation (such as funding and expertise) to sustain it. Motivations that rely on the anxieties and infatuations of the moment eventually peter out. Thus, the way a data service is architected requires system thinking too.

Data Service Teams are mindful of **problematic system dynamics** throughout the design process. For example, tools and analytics can be susceptible to systemic bias, such as when machine learning applications are trained using skewed data sets. Injustice then becomes baked into the data service. Teams are attuned to such problems.



CORRECTIVE FEEDBACK

Any adaptive system learns from **feedback**, which allows for adjustment or course correction. A data pipeline is no different. Information about how data is used is fed back to the pipeline to make improvements. The impact of policy choices made using data is also necessary, especially in the area of public health. Taking a "set it and forget it" attitude can end up costing lives if the data is faulty or repeatedly misused.

Moreover, fixating on a narrow sliver of public health outcomes to the neglect of others can be dangerous. Every policy intervention takes place amid complex societal systems. There may be unanticipated knock-on effects that affect public health elsewhere. Corrective feedback needs to detect these **downstream- and side effects**, not just focus on what is front and centre. Otherwise, today's health solution risks becoming tomorrow's health problem.

LEAVING COMFORT ZONES

Many of these touchstones may seem unfamiliar. Moreover, there are times when team members will go out of their way to try unfamiliar approaches to see what works. Struggling lower down on the **learning curve** can be an uncomfortable place to be. We all take pride in demonstrating competence and worry about being perceived as an impostor. By the same token, trying something new can be thrilling and spur professional growth. Know that the Agency's culture rewards experimentation and continuous learning. Not everything tried will work out as expected and that is okay. Any set-back is the source of valuable lessons and renewed efforts proceed from a position of new-found wisdom.

The danger is reverting back to "business as usual" while paying lip-service to the new methods. One Public Service designer has dubbed that **design "theatre."*** Teams go through the motions with sticky notes and dialogue activities but do not really engage with actual clients. Worthwhile ideas are generated but ignored. Worse, dysfunctional planning processes are reframed with the language of design but not the substance. In our fast-moving world, every moment we are not leaping forward is a loss of ground. Skill-sets grow stale. Stay-the-course policies stop working. By taking these new methods to heart, we can stay a step or two ahead.

To avoid backsliding, the team keeps track of the **bad practices**—*anti-patterns* and *workplace myths*—that threaten to get in the way of success. By raising awareness of outdated ways of doing things and being clear about their shortcomings, there is less opportunity for them to creep back into the process. That involves much scrutiny of traditional ways of doing things; probing questions are asked as a matter of course. Why have we always done things this way? How will that get us closer to our ultimate objective of improving health outcomes? How does that help the clients of data? Does that habitual way of doing things actually work? How do we know?

* Tanya Snook, "UX Design has a Dirty Secret," *Fast Company*, October 18, 2021.

ANTI-PATTERNS

An **anti-pattern** is any set of practices that have a track-record of not working but practitioners continue to implement them anyway. Why do these practices persist? They may have once served a purpose but have become outdated. Some become entrenched habits before their efficacy was fully assessed. Some never really worked but seem like the "professional" thing to do. In any case, adopting anti-patterns undermines the likelihood of project success.

A similar problem emerges with **workplace myths**. A team can be too quick to impose limitations on itself, believing that a non-existent rule or policy prohibits certain actions. Worthwhile ideas may not be given due consideration because someone invoked an imagined rule early on. Worse, relying on myths can become an avoidance behavior that prevents a team from investigating the actual rules and their underlying rationale.

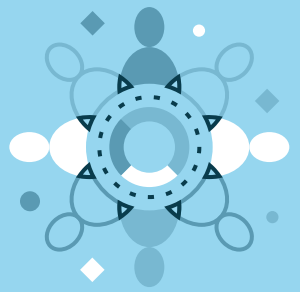
3 HOW ARE **DATA SERVICE TEAMS** ORGANIZED?

CORE TEAM MEMBERS

In its **core configuration**, members of the *Data Service Team* are brought together from across the Agency. This small group works full-time on the project and recruits others into the design circle. Who are the members of that nucleus? At minimum, there will be a Design Lead who is the main locus of accountability for the project (see page 27). There is a coordinator who handles the administrative tasks. At least one developer will handle the technical aspects of making the data service. A design specialist will do the technical design work. There is often a need for a subject-matter expert or a researcher of some kind, such as a social- or behavioral scientist. Not all relevant roles can be determined at the beginning. Additional full-time roles are brought in as the need arises.

The team then grows to include those who are involved on a part-time basis (see next page). As mentioned, invited co-creators lead busy lives and their involvement should touch on decisions that matter most to them. Usually any major deliberation will involve this **extended team**. Who are these members? At minimum, that larger group involves those running the surveillance program. Since the data supply chain includes those working outside of the Agency, an outside partner is often involved. In a client-centred team, direct clients are also recruited, with downstream clients brought in as it makes sense. Everyone in the extended team are genuine co-creators with full weight given to their contributions, even if they are not involved at every step of the process.

There is a larger retinue of partners that take part as well. For example, external researchers and model builders may be asked to present to the team. Stakeholders within the broader public health community can also offer insights. Others within the Ministry may be given a voice in deliberations. Building an **extended network** broadens expertise and enlists cooperation. Members of that network are not full co-creators unless it makes sense to give them a larger role. Nevertheless, their input is listened to intently.



REQUISITE VARIETY

Not everyone can be brought into the design circle. The team would start to look like a conference if every possible stakeholder was brought in—some selectivity is necessary. On what basis should selection take place?

The principle of **requisite variety** stipulates that any system should be as diverse as the environment in which it operates to be adaptive. Likewise, a team should have a sufficient range of knowledge, skills, and experiences to match the challenge being tackled. Teams also benefit by taking into account other dimensions of human diversity, especially those which reflect the clientele of the service.

Diversity is crucial for exploring a problem holistically and uprooting taken-for-granted assumptions. It also exposes team-members to a larger variety of perspectives necessary for creativity and equity. Efforts should be made to include edge-cases and those at the margins of the service.

TEAM ROLES

INTAKE

Intake Coordinator

Responsible for managing proposals for new projects

CORE TEAM

Design Lead

Maintains overall project view, facilitates dialogue, and promotes holistic approach

Design Specialist

User-experience, service, data visualization or related field

Admin. Coordinator

Tasks related to the management of the project

Tech Lead

Software or web developer, or relevant technical expertise

Lead Analyst

Public health researcher, design researcher, social scientist, or behavioral scientist

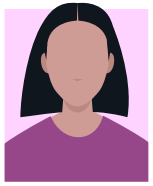
EXTENDED TEAM

Data Manager

Data collection and database management duties

Direct Client

Internal user of data, such as policy analyst, emergency manager, or epidemiologist

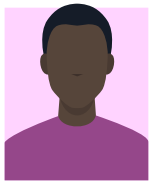


Surveillance Practitioner

Subject-specific knowledge and partnership relations

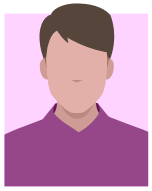
Downstream Client

User of materials from direct clients or ultimate beneficiary of data



Upstream Partner

External partner involved in data collection



Platform Developers

Managing software and tools that all projects can benefit from

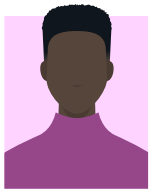
EXTENDED NETWORK

Subject Expert

Knowledge about subject-matter, policy, or data analysis within the field

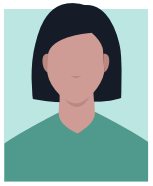
Government Stakeholders

Those with interest in the data, either within the Ministry or the Public Service more generally.



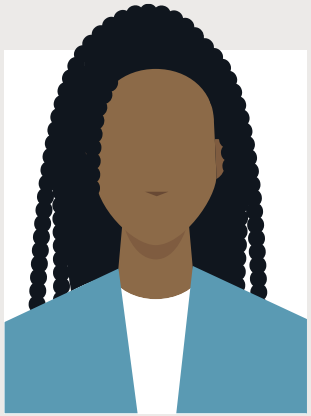
Methods Experts

Developer of methods, models, and technologies of interest to project



Public Health Stakeholders

Members of the broader community with a professional interest in health outcomes



THE DESIGN LEAD

Within any Data Service Team, someone has to keep an eye on the overall process to ensure that everything is proceeding apace according to expectations. That is not simply an administrative coordinator, which is a separate role. Instead, the **design lead** is a practitioner of design methods who takes an active interest in the substantive subject-matter under discussion. They play a lead role in maintaining the ongoing dialogue and making sure design methods are applied with integrity, instead of just going through the motions.

At various points in a project, someone will inevitably "drop the ball" by failing to complete a task or deliver on expectations. The design lead has to pick up the slack so that the project does not stall. Thus, the team's **locus of accountability** resides with the design lead who, at the end of the day, is empowered to make judgement calls on behalf of the team to resolve log-jams and remove obstacles.

At the same time, the design lead is not "the boss" of the process. Participatory design relies on care-taking leadership and good-faith negotiations so as to not sap the motivations of everyone involved. Thus, a certain amount of **inter-personal ability** is expected of person taking up the role.

What else is expected? It is worth reviewing the core responsibilities of the design lead. The design lead:

- is the **custodian of the process** who helps the team "design the design process" while making sure important activities are not neglected;
- does whatever **stage management** is necessary to gently nudge the process along and keep it from veering off-course;
- **facilitates dialogue** when it is helpful for someone with a project-wide view to guide the conversation, rather than a specialist;
- takes an **active interest** in the substantive subject-matter (the details of the public-health challenge) and encourages others to do likewise;
- **brings in** new skills, talents, stakeholders into the design circle as the need arises;
- acts as an **enabler** by making sure everyone has what they need and does not have to resort to short-cuts and sub-optimal trade-offs;
- **builds bridges** with related efforts to maximize the changes of success for everyone;
- makes sure that government rules and policies are **complied with** and understands their appropriate application; and,
- **helps resolve** any dysfunctional conflict within the team.

PROJECT INTAKE

The idea for a data service has to start somewhere. Some projects will be initiated by senior leaders in the Agency who identify an important priority. That may include an urgency that requires a data service set up quickly. Most will be initiated at the behest of surveillance programs who are struggling with data application issues. In any case, an Intake Coordinator (see next page) solicits **proposals** and helps make them project-ready. Every attempt is made to minimize the burden on those making proposals and keeping them "in the loop" about the prospects for a fully fledged project. A review takes place to judge suitable for a *Data Service Team* and if capacity exists to bring together another team. Even if a proposal is unsuccessful, it is not neglected. There may be other ways to satisfy demand and the Intake Coordinator helps explore the options.

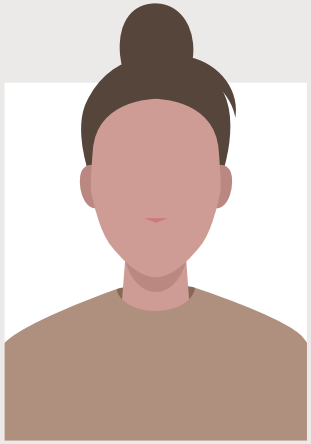
Once a project is accepted, the **basic parameters** for the project are laid down and a Design Lead is recruited. The parameters represent the "sandbox" a team can play in, so to speak. Indeed, the idea of constructing "the sandbox" is a helpful way of thinking about these initial parameters. A full framing of the problem would be premature because, within the Double Diamond process, there has to be a team exploration of the issues. Nevertheless, the team needs some initial guidance to go by. Thus, "the sandbox" includes basic goal setting, resourcing, and scoping of the project. (Details of what is included can be found on page 59). The Design Lead will be involved in those negotiations to offer a basic viability check.

The Design Lead will then assemble the core members of the Data Service Team based on that initial guidance. Soon after, everyone involved meets to initiate the project formally. That is a "warm hand-off" (see sidebar) where everyone is involved to make sure that nothing important goes unattended. The intake process is now over. The team then begins the project with a general orientation to design thinking and client-centred service design, which is especially helpful for the uninitiated. The core team also begins work "designing the design process."

THE WARM HAND-OFF

In projects run according to a strict division of labour, work is passed along from one team-member to the next in a sequential chain. For example, research is sent to designers who, in turn, send designs to developers. A lot of miscommunication happens during those **"hand-offs."** Some good work also falls by the wayside as fumbles inevitably happen. For that reason, everyone is involved in research, design, and development activities in a *Data Service Team*. Grand hand-offs are avoided.

The one exception happens during project in-take as the team is brought together. Doctors call that bridging function a **"warm hand-off."** When a generalist physician refers a patient to a specialist, they all meet as a group to discuss matters. That reduces the likelihood of misunderstandings and awkward transitions in care. The same sort of group referral happens at the start of a data-service project.



INTAKE COORDINATOR

The **intake coordinator** acts as a liaison between members of a would-be *Data Service Team* and the rest of the Agency for purposes of generating new projects. Interest is generated. Expectations are managed. To fully understand the core duties of this role, it is worth distinguishing between **three sets of activities**: (1.) outreach activities; (2.) the intake process; and (3.) the project launch.

With **outreach**, everyone is made aware of *Data Service Teams* as an internal service and their potential benefits. Questions about service offerings are answered. Past successes are highlighted. Goodwill is cultivated. The Intake Coordinator is always on the lookout for promising opportunities that could be put forward as potential projects. In other words, they act as a scout who offers encouragement to reluctant prospects. Senior decision-makers are also consulted to figure out their bigger picture priorities and areas of concern.

The Intake Coordinator is the custodian of the **intake process**. Promising opportunities and strategic priorities have to be reconciled in the selection of viable projects. Several important tasks along those lines are worth listing:

- Members of surveillance teams are encouraged to do their **homework** by documenting their data and forwarding relevant information. Efforts are made to keep such reporting burdens to a minimum.
- The Intake Coordinator conducts a **basic "sniff test"** of proposals to determine which ones are a good fit for *Data Service Teams*, as opposed to other development channels. Viable candidates are then submitted to the intake process for evaluation.
- The Intake Coordinator **manages traffic** with a human voice. Those who submitted proposals are kept in the loop. No one is left hanging in a way that sacrifices goodwill. If a proposal is not selected, alternative avenues for dealing with challenge are recommended.
- Administrative intake tasks are managed.

Finally, the Intake Coordinator brings members of the surveillance program and *Data Service Team* together to start the project. Importantly, that does not mean "throwing projects over the wall." Concierge-style care is involved. That is where the **warm-handoff** comes in. Those involved in a surveillance program are invited into the design circle as co-creators. Members of the Data Service Team are apprised of all the necessary details. The tentative project parameters are reviewed ("the sandbox") so that the team has a basic orientation from which to explore the problem space and develop a fuller understanding of the nature of the problem.

WORKING OUT IN THE OPEN

What does it mean to operate a **semi-autonomous** team in practice? In a basic sense, the *Data Service Team* is given the time and resources to develop a service without meddlesome interference. That partly involves executives freeing up staff to become full-time members of the team. It also involves providing "air cover" to ward off distractions (see sidebar). The team is not burdened with unnecessary reporting requirements. The aim is to provide sufficient autonomy to allow everyone to focus on the project, especially during phases of the project in which everyone needs to come together for group deliberations.

That does not mean that teams are completely free to act on their own accord. As with any technology project, there will be hurdles imposed by oversight bodies. For example, centrally imposed project management constraints have to be abided by. Privacy, equity, and security requirements will also have to be implemented. These are all **design constraints** that the team will have to work with. Autonomy is not freedom from constraints. To the contrary, it is freedom from arbitrary meddling so that the team can concentrate on the job to be done, including keep track of official requirements.

As the project proceeds, there will be setbacks, iterations, and scrambles. All that dynamism means there is no progress meter showing time to completion in any simple sense. How is progress shown? By **working out in the open**. There will be design concepts, mock-ups, and "clickable" prototypes to show others. Completed software modules and partial builds may highlight interesting features. Resources created for the project are made available to others as their built. In sum, there is a lot to show for all the effort. Moreover, whatever task board the team uses to coordinate will indicate progress in a loose sense. Anyone interested in the project should be able to look at the task boards to get a sense of how things are going, rather than teams fill out time-consuming progress reports. Reporting and oversight burdens are kept to an absolute minimum.

PROVIDING COVER

In any participatory design process, many of those involved are volunteering their time and effort for the good of the team. Their deliberations are made in good faith with an expectation that their contributions will be actively listened to. Nothing saps the goodwill of co-creators faster than having group decisions vetoed arbitrarily by someone higher up the chain of command, someone not privy to the group's deliberations. Moreover, all sorts of extraneous requests made on team members risks distracting the group. **Demotivating interference** is a major cause of project failure.

Executives within the Agency play an important role in providing **"air cover"** for a *Data Service Team* and its members. They look out for external demands that may scuttle a team's progress and do everything in their power to shield the team. Without that protection, the likelihood of major delays grows significantly.

GROUP DIALOGUE SESSIONS

Sufficient time is set aside for **group dialogue sessions**, the spaces in which collective brainstorming and lesson drawing happen. These are not periodic meetings to touch base, which occur frequently throughout the project. These are dedicated *design charrettes* (see sidebar) to talk through issues at length without interruption. Each session is broken down into topics which are explored through playbook plays or open dialogue. Questions are raised and tentative answers are found. Each session can be held in physical space or an online co-working set-up. Some may even take place in the field within "pop-up studios" so that conversations can be had with clients close to where the action is. In any case, the idea is to get away from the usual distractions.

The idea of holding some sort of retreat may seem like a luxury afforded to senior decision-makers. It is actually integral to **rapid development**. A large portion of the work contained within double-diamond logic model can be accomplished within a couple of weeks. That involves a few rounds of deep conversation, each taking place over a couple of days or so, with research and experimentation happening in-between. What should be avoided is a series of brief meetings in which nothing gets fully resolved because not everyone is in attendance, conversations repeat, and there is only enough time to touch on subjects briefly. The early stages of the project then drag on for months. The idea is to hold quick bursts of concentrated collective work so that team members can then focus on separate tasks related to making the service.

One benefit of group dialogue sessions is to remind everyone of the dangers of **overly segmented roles**. Treating each team-member as a specialists with their own delimited set of duties is overly constricting. Insights come from getting a range of perspectives on a particular piece of work instead of treating it as a fiefdom. Moreover, there is an upskilling that comes from team members teaching each other the technicalities of their work so that they become well-rounded team-players.

DESIGN CHARRETTES

Design dialogues are often associated with whiteboards and sticky notes. Indeed, group deliberations often happen within **design charrettes**, or studio workshops in which co-creators explore ideas jointly. It is within these extended sessions that everyone talks through issues in a way that gets ideas out into the open … literally. Ideas are jotted down. Concepts are sketched. Paper-craft models are mocked-up. By the end of the session, the assembled materials are a visual record of deliberations.

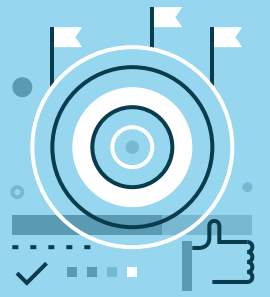
Orchestrating charrettes can be difficult in the modern office work-place. An extended retreat in which everyone can meet can be difficult to schedule. People working from home in different cities rules out fully in-person meetings. There are distractions to manage. Nevertheless, there are virtual whiteboard tools and online meeting spaces that can be used **virtual charrettes**.

LEAN PRODUCTION

A service design project will often meander as the Data Service Team explores issues and weighs options through improvisational dialogue. Yet it would be a mistake to assume the process does not involve disciplined methods for **managing activities** between group deliberations. Within the Agile approach to rapid development, there are a variety of frameworks that can be used. For example, Scrum, Kanban, and Extreme Programming (XP) are mature methodologies for organizing development activities at the task level. These methods prioritize speed and learning, while preserving quality control.

The basic logic of lean production is to strive to get the a basic version of the service up and running as quickly as possible. Contrary to myth, this is not a "minimum viable product," which implies a slapdash version of the service that barely works. A great deal of forethought and care goes into the development of early builds. The project is broken down into more manageable pieces. The research, design, and development tasks associated with each are kept track of and periodically reviewed. The team then proceeds to work over **brief spans** (several weeks in duration) to complete these pieces. These are often called "sprints." That term is associated with the long hours and burnout experienced within technology start-ups. In *Data Service Teams*, a more sustainable pace is chosen so that everyone's personal well-being can be respected. Moreover, an overly frantic pace encourages sub-optimal trade-offs ("technical debt") that usually end up causing problems later on. Indeed, lean production is more about keeping extraneous busywork out of task flows to prevent bugs, distractions, and burnout.

Regardless of the framework chosen, the usual routine is to hold daily check-in meetings. These are used to keep track of tasks and discuss coordination issues. As a piece of work reaches completion, the results are scrutinized and lessons are drawn. All of this is kept track of on some sort of display (see sidebar). That is how the service is built as a series of building blocks.



TASK MANAGEMENT

Within a rapid-development workflow, **frameworks** are used to organize activities at the task level. There are several to choose from. For example, *scrum boards* and *kanban boards* are two different ways to keeps track of which tasks are to be done, in progress, and completed. Each has its own rules about how time and tasks are managed. Within the in-person workplace, these often take the form of actual whiteboards with tasks listed on sticky notes. Within a distributed workplace, specialized software serves the same function of managing group workflows.

This guide is agnostic about which method to use. What is important is that the team pick the **task management tool** members are familiar with and which works for them. Those with oversight responsibilities can often dip into whatever board the team is using to get a sense of the progress without the need for distracting reports.

BRINGING CLIENTS INTO THE CIRCLE

Clients involvement happens in a number of ways. Many clients will be fellow public servants within the Ministry, making it easier for them to justify spending large chunks of time in the project as co-creators. Those who are not can be brought in for more focused discussions. Much client involvement involves research and testing. Consideration is given to selection of these subjects so as to not unduly bias the service. Those at the margins of the service are given special attention, both for equity reasons and because looking at "edge cases" is highly instructive. All of that requires *client intelligence gathering* and a mapping of the clientele (both current and potential). There is a method for that (see next page). That big picture allows the *Data Service Team* to be more strategic in its client engagements.

A variety of **design methods** exist to better understand clients who are not sitting around the table. The clientele is often divided into *segments*, each potentially exerting their own demands on the service (see sidebar). Some clients will be made more vivid using detailed *personas*, or profiles that spell out all the details about the client relevant to the service, including the context of service use and relevant lifestyles. The *service journey* for particular clients can be drawn out, with large "jobs to be done" and small micro-tasks identified. Most importantly, the *mindsets* (goals, assumptions, mental models, and so forth) of those using the service are explored.

At every stage in the project, it is important to scrutinize notional understandings of clients. The **implicit proxies** (notional stand-ins) that the service is designed around have an enormous influence on how the service works. Too often, services are designed around imagined clients that bear a suspicious similarity to the developers themselves. They can also be superficial stereotypes, something to be vigilant about when creating highly abbreviated personas. The most important take-away point is that, when in doubt, look to actual clients. Thus, much of the organization of the team involves maximizing client exposure.

CLIENT INTELLIGENCE

A variety of analytical tools exist for keeping track of real-world clients. Who are the different types of client? The clients of the same kind are put into groups (or *segments*) based on shared attributes that are relevant to the data service. These should not be confused with design *personas*, or detailed profiles of actual clients that serve as an aid to decision-making. As every data scientist knows, inferring too much about individuals based on characteristics of groups they belong to is called the *ecological fallacy*. To confuse matters, in the public sector, persona development tends to revolve around composite personas which combine the profiles of several real clients into one. That is done to preserve privacy. Yet, at the same time, it muddies the waters with respect to what the unit of analysis is. Thus, much care is taken when generalizing about who clients are and what level of description is being used.

TEAM CULTURE

In the spirit of "designing the design process," this overview of organization does not overly specify every aspect of how *Data Service Teams* organize themselves. The point is not to micro-manage how work gets done but outline the basic contours of this new mode of working. Each team member will have their own **preferred styles of working**. How the team works together as a unit will have to negotiated to make the most of everyone's talents. Indeed, it is that mutual accommodation that gives a team a productive working relationship and *esprit de corps*.

In a similar vein, not every role can be specified with precision nor filled with members who have the ideal skill-sets. Most teams will launch with members who may not be a perfect fit for the mission. Some just-in-time learning will be involved to pick up new skills on the fly. Much learning will be done on the job. Indeed, that is what it means to develop capabilities, as opposed to merely acquire them from elsewhere. It is more important that the team is forthright about where it lacks capabilities and finds ways of overcoming deficiencies quickly. The culture of the team rewards humility and the striving for **professional self-improvement**.

NETWORK OF DATA SERVICE TEAMS

Each data service project is a learning process. Data Service Teams as a model works best when lessons and materials are shared across projects. Collectively, the teams act as a **community of practice** that stewards a body of practical knowledge. Insights into what worked and what did not can then be used to evolve the way teams organize. Some of that cross-fertilization happens automatically as particular members work in successive teams. There is also benefit from periodically meeting to pool insights and experiences.

Ideally, lessons learned are documented as projects complete. These are not merely sanitized case studies that portray everyone as heroes. These are warts-and-all accounts that are honest about difficulties and shortcomings experienced so that enduring lessons can be drawn.

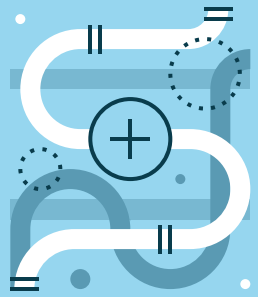
4 HOW TO **UNLOCK THE POTENTIAL** OF DATA?

DATA PIPELINES

Think of the flow of surveillance data as a supply chain or **data pipeline**. Case detection and measurement happen at one end. The influence of the data on public health outcomes happens at the other end. In between, various operations are performed on the data to add value. Some operations relate to quality control, privacy, and security. Others involve analysis and reporting of findings. The major operational segments are listed on the next page. As with any supply chain, not everything happens in one organization. External partners are relied upon because they are best equipped and situated to complete certain operations. Technological infrastructure enables that chain of relations and allows data to flow.

Most requests for *Data Service Teams* are about improving one segment of the pipeline. For example, a data dashboard is requested to better report data. A segment may be missing, such as visual storytelling. These may be worthwhile improvements and obvious places to start. However, fixating on single operations without thinking of the **whole chain** risks passing up important opportunities. How so? Building a data service that clients genuinely value usually involves rethinking how the whole surveillance program works. For example, if clients think the current data is irrelevant to what they do, what good is it to simply improve the way it is presented by adding a fancy dashboard? Moreover, the existing pipeline may not supply data in a timely, reliable, and secure fashion. Thus, building a data service is an opportunity to modernize the chain's underlying infrastructure. Architectural changes can open up all sorts of new service possibilities. Thus, teams map out the whole supply chain early in the project when taking stock of the surveillance program more generally. As they work, they can update and fill-in the missing infrastructural pieces.

Not everyone on the *Data Service Team* will relate easily to the underlying technological infrastructure. Yet unlocking the full potential of data involves knowing a few things about **data-pipeline architectures**. This section explains those essentials.



SUPPLY-CHAIN LOGIC

Thinking about surveillance programs as an integrated whole is often necessary for delivering the intended client experience. Certain operations may be unnecessarily laborious. Bottlenecks and pain-points in the process may slow things down. Awkward workarounds might create security problems. Many of these flaws prevent users from getting the data they need early enough to do something with the findings. A public health agency's *raison d'être* is promoting better health in the population in the here and now, not documenting history. With a **supply-chain logic**, data is treated like a valued product that appeals to various clients and uses.

Data Service Teams may have to think creatively about where **automation and streamlining** can make everyone's job easier. Reducing the amount of monotonous busy-work frees up people to apply their talents on making the data more useful.

SURVEILLANCE-SYSTEM OPERATIONS

A comprehensive data pipeline contains many segments spanning several organizations and organizational units. Each segment is major operation performed on the data which involve a bundle of related tasks. The entire pipeline can be divided into three main sections:

- COLLECTION
- ANALYSIS
- APPLICATION

ESTABLISHING CASES

An instance is detected by the surveillance system and a ***case record*** is produced. Detection often takes place at ***point-of-care sites*** (hospitals, clinics, mobile testing stations, or practitioner offices) or inspections at ***sentinel sites*** (places of interest within a designated region). ***Sentinel practitioners*** are those with special case-identification responsibilities.

MEASURING

On-site measurement differs depending on the nature of the surveillance. Examples include:

- case report or flagged record;
- survey (paper and online);
- sensor data from automated systems; and,
- the collection of physical samples.

SCREENING

There is often a local assessment of whether a case fits the official definition of the condition of interest. The ongoing review of assessment criteria and coding rules takes place based on the evaluation of evidence and practical feasibility. Sensitive data may be scrubbed from case reports early in some surveillance processes. Protocols improve inter-rater reliability for the sake of data consistency.

SPECIALIST PROCESSING

Samples may be sent for processing at a special facility (usually a laboratory) with the right technical equipment and expertise. Additional data is generated. Storage of samples is also a data management issue, as further analysis may be required in the future, although some types of samples cannot be preserved in long-term storage.

CULLING OTHER RECORDS

Meta data about cases may be stored in records found within administrative databases. Some data is drawn solely from these databases, not first-hand or on-site sources. Official records may be combined to construct a separate database that sheds light on a public health condition of interest. Privacy rules govern the cross-referencing of records.

COMPILING & ENCODING

Reports and surveys are encoded into a database. Databases may be sub-setted to create a special-purpose database. Or databases from various jurisdictions may be compiled to create a comprehensive database. With technology-mediated measurement, encoding happens automatically. Meta data for the full database is added. One database is considered the “source of truth” or core version.

PERMISSION SETTING

The sensitivity of data is clarified and access permissions are set. Access is usually based on job role, with trusted outsiders (such as academic researchers) sometimes granted access under strictly controlled conditions.

CODE CLEANING

The code is cleaned, with errors detected and fixed. Anomalies in the data (such as suspicious outliers and correlations) may be scrutinized for quality control purposes.

ADJUSTING

The data may have to be manipulated to account for biases or errors in the collection process. For example, sampled data may have to be weighted to account for biases in the sampling process. Or data may have to be adjusted to account for cyclical patterns, such as weekly or seasonal variation in detection.

SCRUBBING

For purposes of enabling broader access, a version of the database may be created with sensitive information removed or obscured to prevent de-anonymization.

SYNTHESIZING

Variables may be constructed based on data and model projections, such as with life-expectancy (synthetic variables). Machine learning technologies may be used on very large databases (such as patient records) to generate synthetic databases that anonymize data while retaining major patterns.

DOCUMENTING

A data manual is produced in some form. All the variables in the database are described. Methodological issues with data collection and manipulation are outlined. Weaknesses in the data are noted so that analysts can compensate. Ethical issues associated with usage are mentioned.

ARCHIVING

The database is stored in a *data management system*. At minimum, a data catalogue points to the location of the database and includes basic descriptive information (such as keywords and other forms of meta data). Ideally a central repository offers access to data and documentation based on FAIR principles: Findability, Accessibility, Interoperability, and Reusability.

ACQUIRING DATA

Data of interest is identified through a process of discovery. Arrangements are made to gain access to databases or obtain copies of the data.

CONTINUED

POLICY ANALYSIS

The implications for government policy are elaborated. That often involves bringing disparate bodies of knowledge and expertise together to reflect on the practical implications of particular findings. Recommendations and further research questions are generated.

ANALYTICAL VISUALIZATION

Data visualization is an important tool for spotting patterns, establishing relations within the data, and drawing conclusions. These visualizations help analysts think about the data. Note that the visuals created for analysts are often highly technical and are usually unsuitable for communicating findings to down-stream audiences.

MODELING

Analytical models are applied to the data to draw out the implications. These models may involve estimating to a larger population or projecting trends into the future. An ongoing review and testing of models takes place to ensure predictive accuracy based on established standards and corroborating evidence.

DATA MINING

Exploratory Data Analysis (EDA) is performed to figure out what possibilities the data offers. Interesting patterns may be revealed ("knowledge discovery") that suggest directions for further inquiry. Relevance of the data to downstream analysis is explored in this initial pass.

PREPARING DATA

Data files are translated into particular formats to work with analytical software. Data structures are reorganized to conform with analytical models. Databases may have to be merged and reconciled.

DOCUMENTING FINDINGS

Findings are compiled in a technical report that explains data patterns of interest and related methodological issues. Policy implications may be mentioned. This documentation is a record of all potentially relevant findings, not necessarily findings in their most usable form. The format of documentation is often dictated by conventions established by expert communities.

RESEARCHING CLIENTS

Efforts are made to figure out what are the most appropriate audiences for the data (or particular findings) and what form is most useful to them. That includes understanding the context of use, particularly the tasks that the data will inform. It is not enough to know the immediate clients of the data and their worlds, but also clients who are further downstream.

REACHING OUT & EDUCATING

The various audiences for the data cannot be expected to be highly versed in the subject-matter. That may involve teaching potential clients of the data about medical or health issues involved. It also may involve cultivating data awareness and literacy to encourage evidence-based decision-making.

VISUAL STORYTELLING

Data stories can be tailored to various forms of media: interactive web sites, mobile device apps, downloadable documents, and so forth. Each of these information products has its own constraints, features, and usage scenarios. Moreover, data streams may be made available for third-party providers to develop their own information products.

INFO PRODUCTS & PLATFORMS

Information is displayed in ways that are most meaningful for the intended audience. Some data is turned into a compelling narrative that explains findings and adds context necessary for interpreting findings. Storytelling can also include dashboards and non-sequential forms of data display that allows clients to peruse the data and explore scenarios.

Ultimately, the end result of the pipeline should be a major contribution towards creating a healthy population that can be appreciated by the public at large. Any evaluation of a data service's effectiveness should be judged on that basis. So long as intended public health outcomes are found lacking, there will always be room for improving the what data is collected, how it is collected, and how it is put to good use.

IMPROVING CONTINUALLY

Feedback will shed light on opportunities for improvement, not just with implications for information products, but for any aspect of the data pipeline. Ongoing improvements are made.

TRACKING & FEEDING BACK

Patterns of data usage are tracked. Client perceptions of the findings are assessed. Unanticipated uses cases may emerge by asking clients how they use data. Areas of under-use and usage difficulties are discovered, along with any other aspect of the client experience. Ongoing usage is tracked, which may become a new data stream with relevance for health policy.

APPLYING FINDINGS

For data to be worthwhile, it has to be applicable within real world usage scenarios. Thus, data are applied within a context of use to inform actions and change minds in ways that improve public health.

DISCOVERING SERVICES

The clientele for data stories and information products has to be made aware of their existence. They also need to know how to find and access them. That often involves forms of outreach, such as marketing. As importantly, it involves making it easy for clients to discover stories and products on their own as the need arises.

MANAGING PARTNERSHIPS

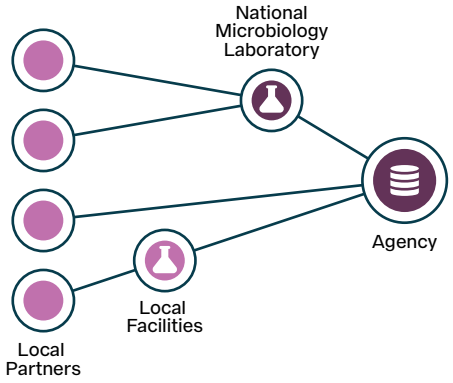
As mentioned, there are dependencies on **outside partners** because that is where scientific expertise, jurisdictional authority, and logistical capability reside. For example, many non-governmental organizations are devoted to rare diseases. It makes sense to engage their unique knowledge and skill-sets for the collection and analysis of data, especially in fledgling fields of health that are evolving rapidly. Provinces and territories may also control much of the data collection.

There is a lot of **variation** in partnership arrangements cross the Agency's many surveillance programs. For the collection process (the first third of the pipeline) these arrangements can be generalized into three types (see below).

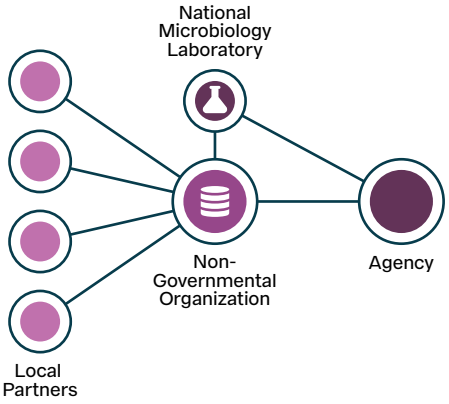
Modern supply chains are becoming ever more **integrated and efficient** despite reliance on diverse external suppliers. Investments in advanced infrastructure let everyone monitor the state of the pipeline and perform tasks quickly. That works because suppliers are also encouraged to adopt up-to-date methods, often with the help of additional resources and know-how.

At the very least, partners should be invited into the design circle as **co-creators** whenever a major change is considered. Partners usually have the expertise and situational awareness to offer crucial insights about where data services can be improved. Moreover, excluding partners from deliberations is unlikely to generate the enthusiastic buy-in necessary to make change work. The blind-spots inherent to imposing solutions from afar create all sorts of unanticipated problems. It is better to involve partners as early as possible.

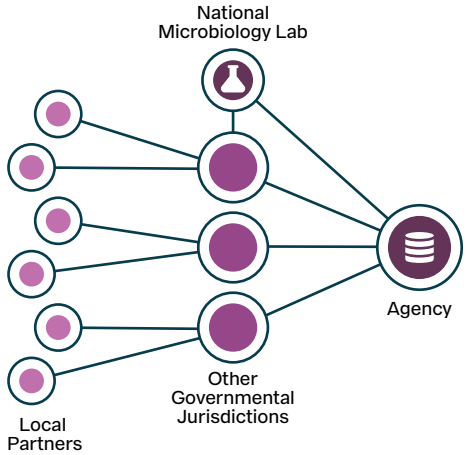
DIRECT PARTNERSHIPS



PARTNER-MEDIATED NETWORKS



INTER-JURISDICTIONAL AGGREGATION



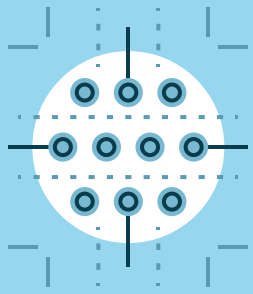
Specialized Facility Main Database

TIDY, FLEXIBLE COUPLINGS

The danger with any supply chain is that it becomes too **rigid, brittle, and costly** to run. If changes to one part of the chain causes lots of other things to break, adapting to changing client needs and policy priorities becomes a major ordeal. If the chain relies on lots of laborious coordination tasks, data flows become slow and financially unsustainable. Many surveillance pipelines are built amid a flurry of interest (such as a health emergency), only to get frozen in the moment without any means for ongoing improvement. Such pipelines drift towards irrelevance and under-use. Moreover, new services cannot be built quickly in supply-chain systems full of brittle entanglements.

If the pipeline is architected properly, on-going changes within each segment can happen with little-to-no tinkering elsewhere. The pipeline is made **highly modular**, with dependencies between segments pared down to the essentials and standardized. There is no need for impromptu coordination nor makeshift work-arounds. The transfer of data from one segment to another is made inherently secure. All the information needed to work with the data is coupled with the data itself; that is, the flows of data are *self-describing*. Many tasks are automated, free-ing people from menial chores and enabling the more timely delivery of data. Data service projects will help build out that infrastructure.

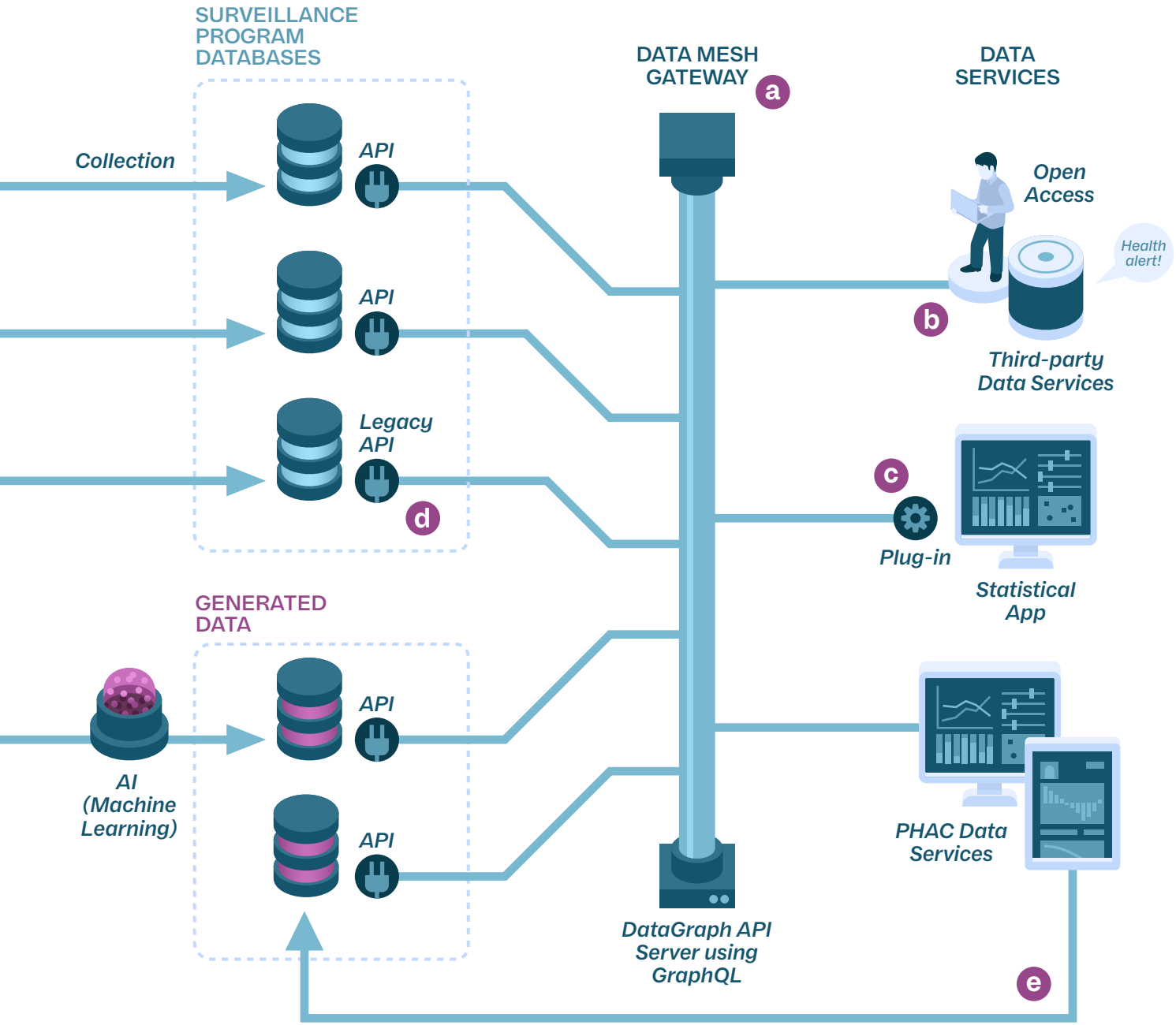
In the past, the tendency was to centralize all the data into a monolithic repository (a *data warehouse* or *data lake*). These days, databases are left with the teams best able to manage them and are linked together into a decentralized network (a **data mesh**, see sidebar). How can *Data Service Teams* promote good architectural practices throughout the network? How can teams then build new services within existing data pipelines? To answer those questions, it helps to separate out how the technological infrastructure is supposed to work upstream and downstream of the main database.



DATA MESH

The sources of data are multiplying and spread all around. They are also growing in scale. How do services tap into these sources? Centralization is no longer practical. Yet working with distributed data sources is messy without a technological architecture called a **data mesh**. What does that entail? Databases are managed in a decentralized way by those with a strong understanding of the subject-matter and context of collection. In the Agency's case, that is the surveillance programs and external partners. Data does not languish in silos, however, but is treated *as a product* others are expected to use. All the information needed to use the data is bundled with the data itself (it becomes self-describing) and made accessible within a self-serve infrastructure. Data is then routed through a common "gateway" that allows multiple data streams to be accessed at once (see page 42).

DOWN-STREAM ARCHITECTURE



NOTES

a

The Data Mesh Gateway automatically reconciles database variables (according to schema) and reroutes the data streams. Maintaining the gateway becomes unwieldy if additional tasks are assigned to it. The general idea is to keep endpoints “smart” and the pipes “dumb.”

b

Database owners maintain control of access based on their own policies. Tapping into the gateway is “self-service” insofar as anyone can easily see what data is available to them and get needed information from the data stream itself (data as self-describing).

c

Researchers and analysts can use the software they prefer. An app-specific GraphQL plug-in allows statistical apps to connect to the gateway.

d

Legacy APIs can be upgraded or a wrapper can be placed around the API to ensure compatability.

e

Data services are expected to generate data in order to function. Some of that data may be of use to other services and access can be afforded through the gateway too.

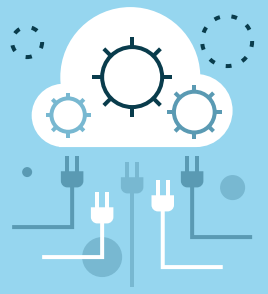
A. **FEDERATED DATA**

Data is archived in a secure, networked, and backed-up **database**, which stands as the one official version (the "single source of truth," in data-management lingo). A surveillance program may not have one if, for example, it works off of spreadsheets. Perhaps database is also be stuck on an obsolete system, such as obsolete database format or server set-up. Upgrading to a modern database will be an early priority.

The database communicates with other systems through an *API* (see sidebar). That too may be missing and will have to be added. If APIs already exist, a data service could use them to access data directly. However, legacy APIs are usually rigidly over-specified for particular uses. Any change causes the pipeline to break. It is better to upgrade APIs to route data through a mesh "gateway" (API server) called a **datagraph**. If a service wants to access data from multiple databases, separate connections do not have to be built to each one. That is a coordination nightmare. The datagraph becomes the single point of access that any service or application can plug into (as shown on the previous page).

Data has a **federated structure** within the datagraph. Database variables are described and organized into *subgraph schemas* (taxonomies) that *Data Service Teams* help specify. Multiple schemas are automatically combined into a *unified schema* that handles all the rerouting mechanics. Users tapping into the data do not need to know anything about that complexity. They are presented with a simplified *API schema* telling them all they need to know about data they can access. Thus, teams building data services can easily discover and tap into multiple data sources by using one public-facing schema.

If minor changes have to be made to either the database or the service, that can be done without having to coordinate. Everything just continues to work. If major changes risk causing some sort of breakage, the **open-source software managing the datagraph** alerts everyone in advance. The subgraph schema details can then be reworked without any sort of interruption to the data service.



API DESIGN & GRAPHQL

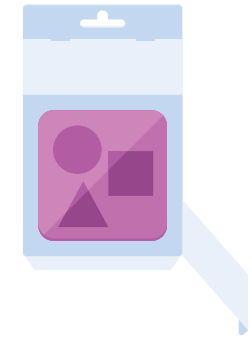
An **Application Programming Interface** (API) is the connection through which applications acquire data. A *data-stream* is an on-demand transfer of data through an API. It is no longer tenable to pass along data by mailing data files around or the like. All data should flow through APIs these days.

How do *Data Service Teams* learn about what APIs exist? And what data they offer? Too often, APIs are archived in an "API store," a dumping ground of miscellaneous loose ends. It is better to plug them into the *datagraph*, an automated clearing-house where all the data becomes discoverable, self-describing, and easily routed to a service. The open-source query language that makes the datagraph work is called **GraphQL**. Popular programming languages used by developers and statistical applications used by public health professionals (see next page) all have their own GraphQL extensions.

DATA MESH CORE PRINCIPLES

DATA TREATED AS A PRODUCT

Data is prepared for immediate use elsewhere, rather than languishing in local silos. Data becomes self-describing, discoverable, address-able, and trustworthy.



DATA
Organized observations

META DATA
Data about the data

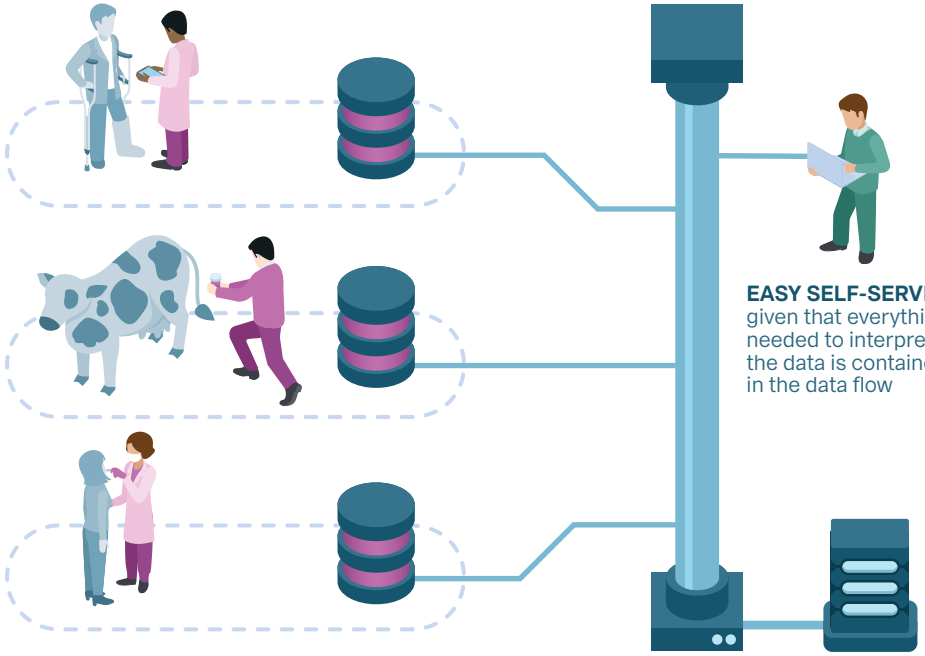
IN-FLOW CODE
Code for bringing in and consuming upstream data

OUT-FLOW CODE
Code for APIs that enable down-stream access

FLOW-CONTROL CODE
Code for policy enforcement, such as access restrictions and provenance assurance

DECENTRALIZED DATA OWNERSHIP

Functional operations that are closest to the action (domain) are best situated to collect, manage, and control the data (ownership). As suppliers of a data product, the owners talk to users to understand their needs and strive to continually improve quality and relevance.



DECOMPOSITION
of data collection makes sense given the logical breakdown of activities and jurisdictional authority

DECENTRALIZED
management of the data, with any shared infrastructure not infringing on data owners' autonomy

SHARED SELF-SERVE PLATFORM

A common, scalable data-pipeline infra- structure prevents duplication of effort while ensuring data owners' autonomy. Everything needed to use the data is handy so that users can serve themselves without having to interact directly with data owners.

EASY SELF-SERVE
given that everything needed to interpret the data is contained in the data flow

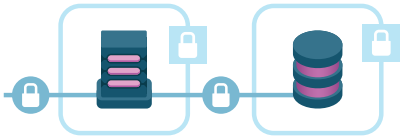
DATA GRAPH
aggregates various data flows so that users can access multiple databases without establishing seperate pipelines

DATA GRAPH MANAGER
dynamically generates schema for reconciling variables across data-bases, alerting every-one to major upstream database changes, and preserving access restrictions. Examples include Apollo's GraphOS and The Guild's GraphQL Hive.

FEDERATED GOVERNANCE

Global standards preserve inter-operability, security, and automated operations of the platform while preserving data owners' ability to apply to context as they see fit. Any reconciliation of data across domains is done dynamically according to an evolving schema.

BASIC TECHNOLOGY STANDARDS
to connect to the shared platform



HIGH SECURITY STANDARDS
across the entire data supply chain

GLOBAL STANDARDS
for classifying conditions and cases based on up-to-date scientific evidence

B. **PLUGGING INTO THE DATAGRAPH**

Diverse specialists (both inside and outside the Agency) want to access data using their own software applications. These are the direct clients of data. Moreover, if data services are to be designed for other clients, the infrastructure should enable those services to be built quickly and without hacky technical workarounds. The current **patchwork** of systems and kludgey interconnections prevents that. A *data mesh* run with a *datagraph* gateway reconciles all that makeshift complexity. At the same time, the mesh accommodates all sorts of diverse wants and requirements.

Currently, the process by which **specialists access** data is arduous and often not up to high standards of security. With a simple software extension, specialists can draw out up-to-date data from the datagraph using their preferred software applications (see sidebar). Even better, those outside the Agency can do likewise in a way that is both secure and open.

The implications of the datagraph are more profound for **designed services**. For example, a data dashboard or mobile application can plug into multiple data sources. That opens up new possibilities to seamlessly bring together data on the same topic (or, indeed, multiple topics). It also enables new forms of data storytelling that are better tailored to the clients' context.

On a technical level, the benefits are even more profound given the **directional flows** of data. Surveillance data is a *one-way flow*, with the provenance of data assured by highly secure pipelines. Yet many services will generate new data that is also managed by a database. For example, a client may have settings for a service that have to be stored somewhere. Statistical models may generate new data of value to others. In either case, other services may benefit from accessing that data. That involves a *two-way flow* of data in and out of databases. Thus, separate databases store different types of data with different security needs and access controls. The complexity of those connections would get out of hand without coordination by the *datagraph*.

SPECIALIST SOFTWARE

Public-health experts rely on a range of software applications to analyze data and construct statistical models. Many are unwilling to give up application they have grown dependent on to do their jobs; our skills are closely tied to the tools we use. Thankfully, no one is being asked to give up desktop software within a data-mesh system. **Off-the-shelf extensions** can bridge the gap between the datagraph and specialist applications. For example, GraphQL extensions exist for Jupyter Notebooks, Tableau, and SAS Viya, the most common applications in the Agency and broader research community. Specialists will find it easier to tap into data streams. Specialist software may or may not be directly relevant to the data service being built. Regardless, *Data Service Teams* should make it easy for specialists to add extensions and configure their workstations as a way of making the data more openly accessible.

C. **SECURITY AS ENABLER**

If designed recklessly, a data pipeline and data service become a large attack surfaces, vulnerable to malicious actors who can do irreversible damage. Building **inherent-security methods** into every component (see sidebar) allows teams to move faster. There are fewer security testing hurdles to jump through if security overseers are satisfied that a data service teams has gone beyond normal precautions.

The move to **zero-trust networks** will be a major advance. Traditional networks rely on *perimeter security* around an outer boundary, with everything inside treated as a trusted private network (see next page). Unfortunately, once a hostile actor gains a foothold in the network, it is easy to gain unauthorized access to other parts of the system. Zero-trust networks solve that problem with end-to-end encryption of data flows, with access to any software applications (such as a database) requiring authorization. Everything is protected because it the network is treated as if it is already compromised. Best of all, services can be made more convenient for users with less annoying authorization routines.

Within the larger web of data partnerships, security becomes an enabler. Other organizations are leery about the security implications of outside help. If *Data Service Teams* maintain security standards that are much higher than what is already in place, that care goes a long way towards reassuring partners that any cooperation will not cause security headaches later on. Better security should be an sold as a **partnership benefit**.

The *Data Service Team* is given a lot of flexibility about what tools it uses to build services. For example, if the team insists on writing parts of a service in a particular programming language, it needs to know the security weaknesses of that language and remove the vulnerability. An advantage of using libraries of ready-made components (page 47) is that **security good practices** can be applied across the library. In any case, it is worth inviting a security expert into the process early to scrutinize the tools and components the team intends to use.

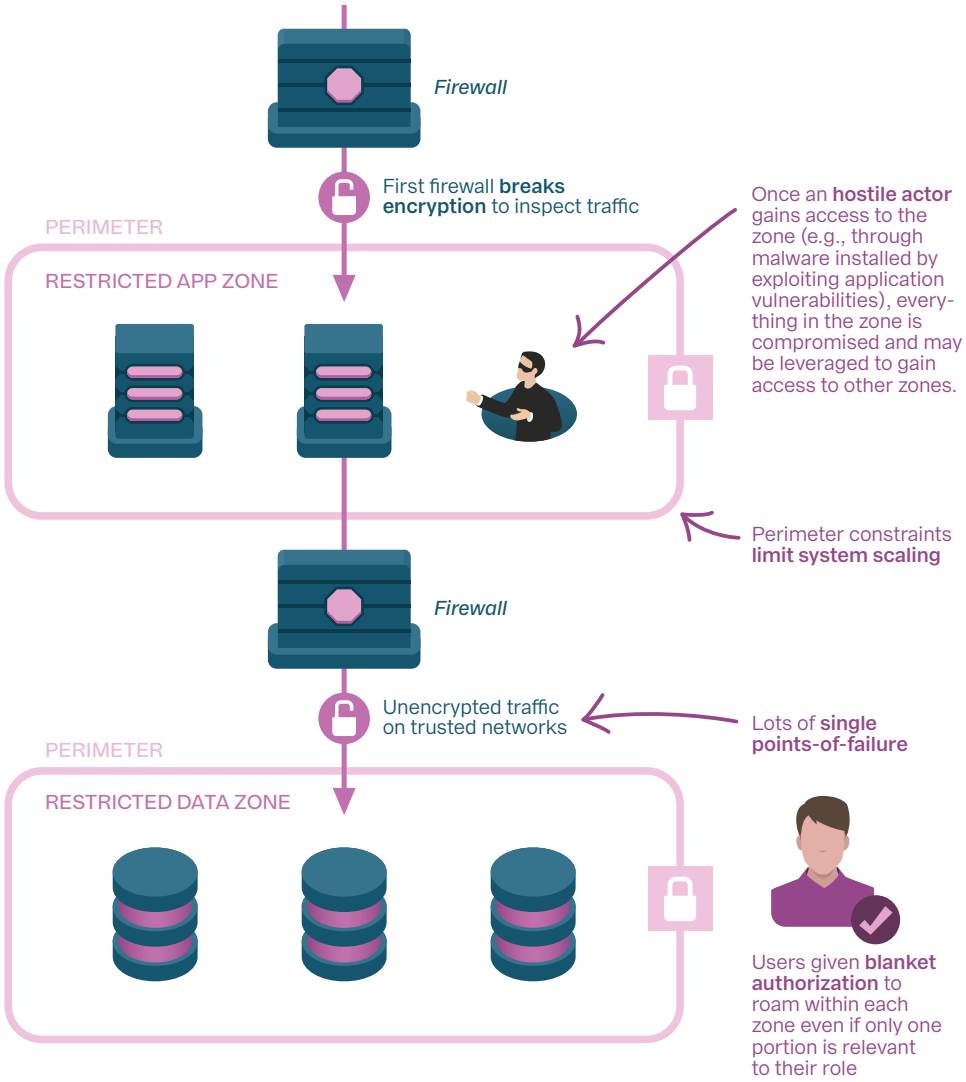
INHERENT SECURITY

Making services **inherently secure** starts by streamlining the code by removing frivolous features, unnecessary dependencies, and out-dated legacy code (“cruft”). A software module should only be as complicated as it needs to be to perform its core functions. Attack surfaces are reduced in size by compartmentalizing systems so that if one part becomes compromised, others remain secure. Thus, the main task is to reduce complexity of software that can add hidden vulnerabilities. Just as importantly, the complexity should not just be foist onto users. Reducing security risks for the system overall should not add new risks for clients to worry about. Similarly, reducing complexity should not make the job of partners harder, thus creating vulnerabilities elsewhere that are harder to control. Inherent security also involves using software that does away with entire categories of threat.

ZERO-TRUST NETWORKS

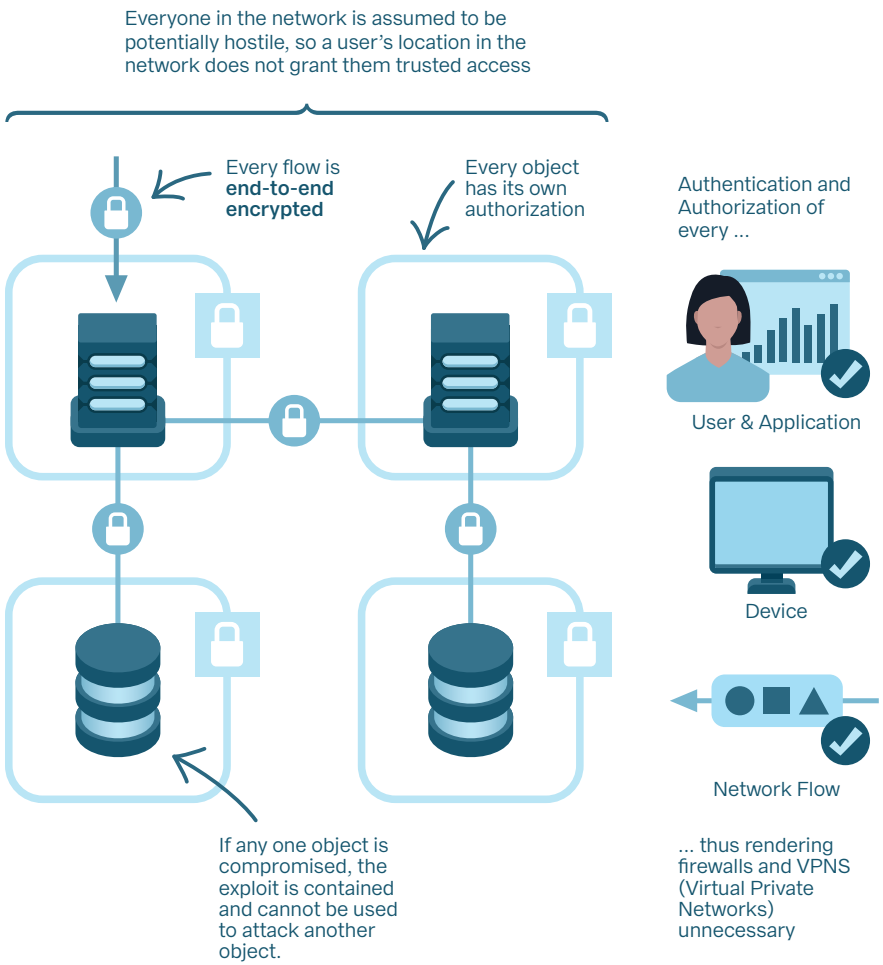
TRADITIONAL PERIMETER DEFENSE NETWORK

High levels of inconvenience caused by onerous safe-guards create a false sense of security as vulnerabilities persist



ZERO-TRUST NETWORK

Everything is protected with encryption and authorizations are more sensitive to who is accessing data, with what devices, and for what purposes

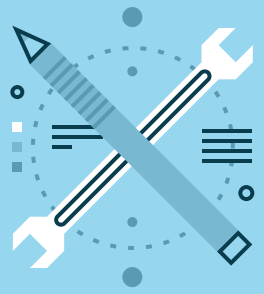


D. **DEVELOPMENT AS ASSEMBLAGE**

Software reuse allows services developers to build services faster, more securely, and with consistent quality. Why build from scratch when there is a wealth of existing software libraries available in open repositories? These can be reworked and added to so that the Agency has libraries better suited to its particular needs. The ideal workflow is a **"software factory."** Most of the pieces for building a service are available as pre-fabricated modules and service development is more like factory assembly. As both the datagraph and software libraries expand, development times plummet. What might have taken years to build takes only a few weeks. As staff turnover happens, as it inevitably does, newcomers with basic programming skills can easily figure out the libraries and start building new data displays right away. Given all those benefits, some government organizations are mandating the software-factory approach.

The principles of open reuse can happen across the data pipeline. However, front-end libraries are a priority (see sidebar). These are the most visible part of the service where data gets applied. In keeping with the factory analogy, there may be **multiple assembly lines** representing different types of library. A popular general-purpose library can build any service if augmented by other libraries. Compatibility layers smooth out the wrinkles. For rich data displays, however, that is a second-best approach. Advanced graphics and visual cues are difficult to build. Code becomes overly complicated and the user experience tends to be clunky.

The New York Times and World Health Organisation (WHO) provide the proof-of-concept for a **two-pronged approach**. They use general purpose libraries for much of their web and app services. Yet when it comes to rich data visuals, they have a separate assembly line using libraries that are specifically designed for visuals (see next page). Small teams can then assemble extremely advanced data stories, creating light-weight software that is a joy to interact with. Instead of plain charts, graphs, and maps, more insightful visuals "wow" clients.



FRONT-END LIBRARIES

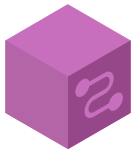
Front-end libraries are repositories of software for interfacing with the user and applying data, such as with data visualization. In the example shown on the next page, there are two groups of libraries, both using the JavaScript programming language. One is a general purpose interface library for a variety of web-based services (React), with a version for mobile apps (React Native). Another was developed by data visualization specialists to tell data stories (Svelte). A larger library (SvelteKit) provides much of the functionality found in general-purpose libraries. Other graphic libraries are easier to plug-in (notably D3). Even better, graphics built by non-coders in data-graphics applications (such as Datawrapper or Mapbox) are highly compatible. Advanced graphics are then easy to build with larger array of charts, graphs, and maps available. User interaction with the data graphics is also much more fluid and snappy.

THE “SOFTWARE FACTORY”

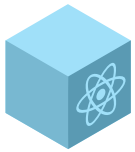
General Web Services

Mobile Devices (iOS & Android)

GENERAL SERVICE LIBRARIES

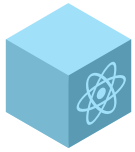


Relay
Connects to Datagraph



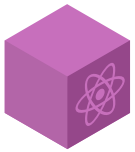
React
General Interface

JSX
Webpage Layout



React Native
Cross-platform Mobile Apps

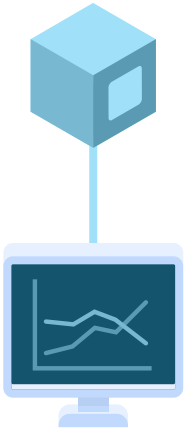
ADVANCED DATA VISUALS



React D3
Compatibility Layer for D3

D3
Advanced Data Visualization

Other
Specialized visuals (such as 3D maps)



JavaScript Visuals Generated by Software Apps

COMMON LOOK & FEEL

Design Tokens
Graphic and Interface Details

Consistent look and feel across services while tailoring to specific media

E. **VISUAL DESIGN LANGUAGE**

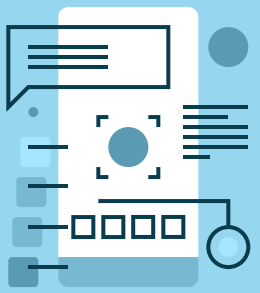
A unified **visual style** gives the Agency's services an identifiable brand identity. A polished "look and feel" also bolsters credibility and otherwise enhances the viewer's service experience. Style includes colour palettes, text formatting, and the design of interface and chart elements. For data visualization, the clean and consistent styling of graphs, diagrams, charts, and maps is crucial for making findings easy to interpret. Moreover, the way style is handled partially dictates how accessible a service is. There is nothing frivolous about service-interface styling—it is not just "eye candy."

Government agencies have difficulties sticking to a common style. Communications gatekeepers often put up **procedural roadblocks** to check for compliance. The track-record of such arrangements is spotty and major project delays are the result.

A better way is to put visual elements into a common library called a **visual design language** (also called "UX design system"). How interface elements are expected to look and work under particular circumstances is all specified in advance. Elements are kept at the fingertips of designers and developers by encoding them into *design tokens* (see sidebar), which can be invoked with a click of a button while using whatever software is most appropriate for the task. Tokens can be specified for various charts, graphs, and other forms of data visualization. A *library of charts* shows reference implementations of the visual design language. Those charts can be reused or serve as guides for the development of more complicated chart types. All told, the visual design language cuts down on development times while effortlessly increasing consistency.

The WHO has provided a **proof-of-concept** for this approach to data visualization workflows.* The *World Health Data Hub* (🔗) is a repository of interactive data visualizations, all developed using an in-house, visual-design language.

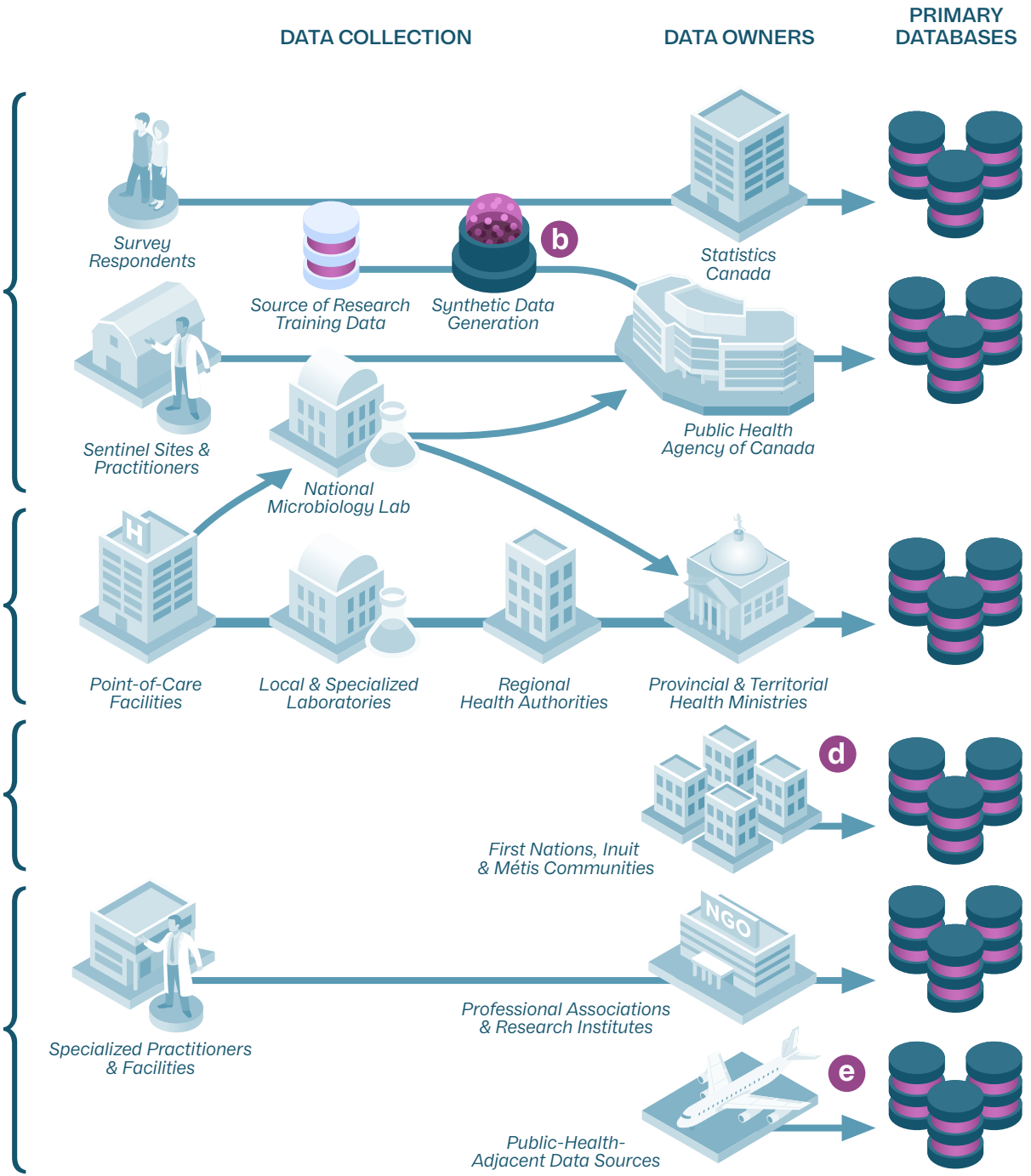
* Moritz Stefaner, "WHO Data Design Language," *Truth & Beauty*, September 8, 2022. 🔗



DESIGN TOKENS

A **design tokens** is a handy way to specify and store interface design features. This method is on the road to becoming a standard of the W3C, the governance body responsible for the World Wide Web. Although originally designed to streamline web development, design tokens are more broadly applicable. They are *technology-agnostic* insofar as they can be applied using whatever software a team-member prefers, including rapid-prototyping software (such as Figma) and task-managers (such as Trello). At the same time, design tokens are *application-sensitive*; that is, versions can be tailored to particular media and client need. For example, desktop web, mobile web, mobile app, and print products can all have their own variations of a style that better suits the medium. Thus, a diverse range of service scenarios are experienced by clients as a unified brand.

UP-STREAM DATA SOURCES



a

Under the federated data-mesh model, those closest to the domain retain control over the data and the policies that control access.

b

A priority for expanding the pool of data available is to incorporate new methods, such as the use of machine learning to generate synthetic data.

c

The four types of supply chain are loosely represented here. Teams will have to map out the particular chain early in each project. New pipelines need not be limited to these forms so long as they respect jurisdictional authorities.

d

Data is collected by communities with the help of organizations such as the *First Nations Information Governance Centre* (🔗) based on their OCAP (Ownership, Control, Access, and Possession) principles.

e

A neglected opportunity is to tap into data sources adjacent to public health, such as water works. Many third-parties will offer their own open data streams.

A. **HELPING PARTNERS MODERNIZE**

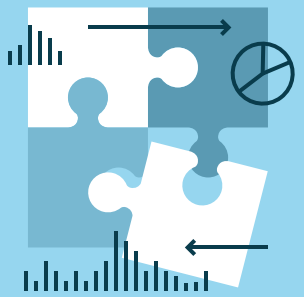
The discussion up to this point assumes that the data needed to run the service is already available. That is not so in many cases. Consider a few reasons.

First, some data collection operations may be behind the times technologically. Upstream data sources may not have APIs and, instead, are passed around as file transfers. Even though everything works with APIs these days, it may take some negotiation and donations of technical expertise to make that happen. **Upgrades** are areas where *Data Service Teams* can offer a helping hand. None of that assistance impinges on partner independence but merely provides an efficient mechanisms for performing existing functions.

Second, there may be missing data. Data collection efforts may have to be expanded. Entirely **new surveillance activities** may emerge out of a data service project. That may not be happening because client signals about what data they would like are not reaching upstream partners. *Data Service Teams* play a big role in making upstream pipelines more sensitive to client needs. Assistance may also be provided to build new and sustainable data sources.

Third, upstream partners may lack the **infrastructure** to make streamlined data-collection operations viable. Helping to build out (and sometimes host) that infrastructure may be a worthwhile investment of a team's resources.

There is another layer of infrastructure that team should concern itself. Modern supply chains operate with a real-time, full-scale view of how the whole system is working. Products moving through physical supply chains have a "digital twin" that is tracked in the cloud. That **monitoring function** allows everyone in the chain to see where bottlenecks and other unanticipated blockages are occurring. A similar sort of system can exist with data pipelines, allowing for automatic tracking of how data is flowing through the system. Problems can then be flagged earlier and interventions launched faster. That sort of monitoring capacity can only be built atop modernized pipelines.



PAN-CANADIAN STRATEGY

The federal government has championed the idea of a Pan-Canadian Health Data Strategy. Policy-makers stress how better coordination of data across the country is necessary for managing health emergencies and coping with emerging challenges. While negotiations are ongoing, the Agency is well situated to put the infrastructural building blocks in place now so that any major initiative can be implemented quickly. Even if a grand strategy is not formally agreed to, there will continue to be on-going advancements of nation-wide data sharing. Infrastructural choices should not unduly constrain would-be partners. Instead, they should support the flow of rich data through decentralized systems. Data mesh and software factory approaches are premised on that level of flexibility and jurisdictional division of powers.

B. **DATA CROSSING BOUNDARIES**

There have been long-standing difficulties exchanging data within the field of health. Notably, clinicians struggle to share patient records and provide seamless care because systems do not talk to each other across facilities and jurisdictions. That lack of flowing data can cause care to suffer, with real-world harms occurring when clinicians fail to get the right patient information in a timely fashion. As a result, the health field has put great efforts in defining what data sharing should look like. **Interoperability** between systems has long been a priority (see sidebar).

Systems engineers are familiar with the idea of interoperability. Technical protocols exist to transport data between facilities. Yet if the systems are unable to interpret that data, the data often has to be re-entered manually, creating new risks of data input errors. Structuring databases to have the same fields and syntax may seem appealing but would involve a level of standardization that is impractical in most cases. It is better for systems to have shared models and vocabularies (*schema*) so that any translation that is required can happen automatically. As discussed, that can be accomplished without having to tinker much with the systems themselves by using a mediating "gateway" called a *data graph*. This acts as a **translator** because it contains the schema necessary for interpretation and repurposing of the data across systems.

A major interoperability challenge is organizational, not technical per se. Facilities and jurisdictions will have different policies and processes that limit the sharing. These are codified in legal documents, stipulated in official policies, or entrenched within formal procedures. Data service teams will have to resolve these **organizational obstacles**. Within a federated data arrangement, data owners ultimately set their own policies. Data-graph management software (such as by Apollo and The Guild) makes preserving those policies easier while sharing data. Yet that does not preclude negotiations to make policies and procedures more compatible across the federation.

INTEROPERABILITY

Interoperability is the ability of different systems to cooperate by exchanging data and putting it to use. Think it as having four-levels:

1. Data files can be transferred but not read by the other system, causing data to be re-entered manually or otherwise translated;
2. Data transferred and reconciled insofar as the same database fields exist in both systems (shared syntax and data structures);
3. Data transferred and automatically used because both systems share models and vocabularies (*schema*) for data interpretation; and,
4. Data can be transferred and used even if crossing organizational boundaries, thanks to common record identifiers (or standards) and compatible non-technical policies and processes (such as those governing privacy, consent, and access).

C. **NEW DATA SOURCES & METHODS**

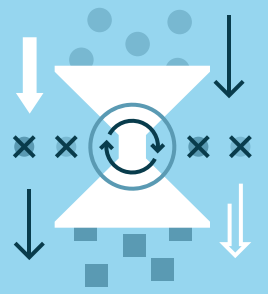
A lot of the data used by the Agency flows from the provinces and territories, Statistics Canada, and a handful of non-governmental organizations that specialize in particular health conditions. There are good reasons for that **dependency** based on jurisdiction and expert capabilities. However, it would be a mistake to limit the data sources to those partners. In our big-data era, new data sources are proliferating at a heady pace.

Many **third-party data sources** have public health implications, even if the ostensible purpose for collection is unrelated. For example, water-quality data gathered from various estuaries for environmental purposes may also contain indicators with health implications. Private companies may collect data as part of their business operations that also shed light on public-health issues. Even if these data sources are not central to public health, they nonetheless serve as important context with which to interpret data that is health related. Lateral thinking and research may be required to identify these alternative sources.

New technologies are also generating new forms of data. For example, artificial intelligence (AI) and machine learning is creating new types of data that can be incorporated into a service (see sidebar). Scientific advances related to genomics is revolutionizing many fields with the help of data and shed light on certain classes of disease. Members of *Data Service Teams* should keep up with these advances to open the mind to new service possibilities.

The **data-mesh infrastructure** allows these data streams to be turned into services with relatively little tinkering. Many already have the APIs necessary to tap in.

The Agency cannot simply be expected to chance upon these sorts of opportunities. Skill, knowledge, and attentiveness are required to spot new data sources. Over time, *Data Service Teams* as a community of practice (see page 34) are expected to become a **centre of excellence** for that sort of opportunity scouting and technological foresight. That partly involves taking an interest in the substantive subject-matter.



AI & MACHINE LEARNING

Artificial intelligence (AI) is a promising source of new data. Specifically, **machine learning** algorithms quickly spot patterns that allude humans; patterns which can then be applied in routinized tasks, such as generate imagery or make recommendations. That ability improves as more and better data trains the algorithm's pattern-spotting. So far, public health officials have only scratched the surface of the potential applications. For example, machine learning can be trained on patient heath records to generate *synthetic data* containing most of the relevant diagnostic patterning while removing person-specific identifiers, thus preserving privacy. Machine-learning diagnosis of diseases from medical scans is fast and accurate, with case definitions becoming more precise, which improves surveillance capabilities. Early-stage disease outbreaks can be spotted more quickly from various data sources. The list goes on.

D. **SUBSTANTIVE SUBJECT-MATTER**

Too often, information- and data-service builders treat the substantive subject-matter of the service as **mere "content."** In other words, they show little interest in what the data is about. They mostly care that the data flows to those who value it. Using the generic term "content" gives a false sense that a service built for one surveillance program can simply be repurposed for another. What difference does it make if, say, one program is about microbial resistance and another is about mental health? Or if one program is about physical injury and another is for a genetic disease? It is true that all the programs can benefit from a shared data infrastructure. However, data pipelines should not just be "big, dumb pipes." The substantive differences in subject make an enormous difference to how the service is designed and implemented.

Every *Data Service Teams* jumps into the deep-end when it comes to gaining familiarity with the relevant public health issues. Obviously, no one will be gaining a doctorate's worth of knowledge in a few weeks. That is why the extended team contains subject-matter experts who are expected to share their knowledge. Indeed, every team members benefits from **taking an interest** in the subject. The intersection between fields is where most innovation happens. Everyone teaches each other and new understandings emerge when team members meet each other half-way. Moreover, a bit of homework goes a long way. It is not a matter of being a "tourist" within a field of expertise, but of developing a *common vocabulary* to have sensible discussions with experts.

Knowledge about the subject is also important to the user of the service. Assuming that users are up-to-date experts in a health field would be a mistake. Within the field of medicine, the amount of practical knowledge is enormous and it can go stale very quickly depending on the subject. That is why **channeling the latest scientific findings** to clients as part of a data service may be an important source of added value (see sidebar).

KNOWLEDGE MOBILIZATION

"Knowledge mobilization" is a fancy term for the practical application of research findings. With 2½ million scientific studies published each year, most research languishes in obscurity. Sometimes, *meta studies* will review a large literature to figure out what the balance of evidence suggests, drawing out the practical implications. Usually it is up to individuals and institutions to curate studies in fields of interest to stay atop the literature. Without the right technical supports, however, such **curation** is impractical for all but the narrowest of fields. Interpretation of data can be misguided without the latest scientific literature, especially during fast-moving health emergencies. Keeping in-house expertise up-to-date happens throughout the Agency. Even so, data services play a key role in bringing knowledge and data together.

INFRASTRUCTURE BUILD-OUT

Insert

BRIDGING TECHNOLOGY

Insert

INFRASTRUCTURE DEVELOPMENT AREAS

UPSTREAM (DATA SOURCES) → DOWNSTREAM (DATA USES)

A DATA SUPPLY-CHAIN INFRASTRUCTURE


Surveys & Inspections


Lab Reports


Case Reports & Samples


Administrative Records


Automated Collection


Synthetic Data & Machine Learning


Open APIs to Related Data


Data As Product (Self-describing & Streaming)


Automated Processes (Fewer Manual Steps)


Networked Databases (Not Static Files)

B BRIDGING TECHNOLOGIES


Legacy Static Data Files


Automated Safe Ingestion of Static Files


Streaming System for Real-time, Event-driven Pipelines


Networked Databases

Bridging technologies help to cope with legacy systems and ongoing friction-points across different systems.

Ideally, most upstream supply-chains will connect directly into the data mesh.

C DATA MESH GATEWAY


GraphQL DataGraph


Schema & Access Management

D DATA SERVICE "SOFTWARE FACTORY"

BACK-END


Scientific & Epidemiology Models


AI Models (e.g., Large Language Models)


Data Fetching Modules & Plug-Ins


Other Data Process Libraries

FRONT-END


Interface Libraries


Graphic Libraries


Data Design Language (Design Tokens)

DATA SERVICES


Interactive Data Services


App-based Services


New Consumer Device Services


Open Accesss

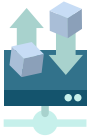
E FLEXIBLE CLOUD INFRASTRUCTURE


Cloud Service Providers


Server-App Containerization


Service Mesh (Container Mgmt Across Providers)


Zero-Trust Authentication Services


System for Safe Building-Block Reuse via Public Repositories

F INHERENT SECURITY DESIGNED INTO EVERYTHING


Zero-Trust Networks (Everything Encrypted)


Automated Checks

Higher Test Thresholds for Less Secure Languages

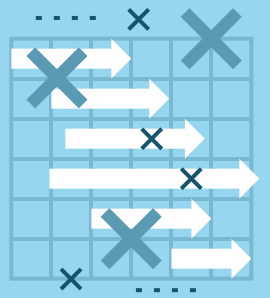
5 HOW DOES **THE PROCESS** UNFOLD?

QUESTION LOGIC

Leadership starts by asking insightful questions that get others to reconsider existing patterns of thinking and acting. Being inquisitive means not presuming too much nor prematurely jumping to conclusions. Questioning can also guide a project along. It is a method, not a form of freewheeling improvisation. A **question logic** is a sequence of questions that build on one another to guide a team through successive bouts of convergent and divergent thinking (see page 10). Progress towards fulfilling goals happens as tentative answers are arrived at and acted upon.

A question logic is an alternative to planning out everything in advance. Contrived planning exercises have a poor track record for technology projects (see sidebar). Following the flow of questions keeps minds open to new influences and nimble enough to adapt to unforeseen difficulties. Tangents may emerge but that does not mean the project meanders aimlessly. The double-diamond process entails a loose **sequence of stages**. The order is not strict; it might make sense to rearrange the stages. Some stages will repeat, as improvements come from iteration. In any case, each stage in the process raises a series of questions that can guide the conversation within the team.

The rest of this section visualizes a hypothetical processes by breaking the double-diamond down into more detailed activities. A panel is devoted to each stage. An accompanying graphic shows the *Data Service Team* involved in particular activities. The objectives of each stage are described. A **generic starter-list of questions** is then listed. Of course, each project will have its own challenges related to the subject-matter. Subject-specific questions are added to the starter-list list as activities proceed. Teams should feel free to adapt these questions as they see fit. Answers are developed through group conversations and exercises. This is where *playbook plays* can come in handy. These are simply ways of structuring the conversation around a group of related questions.



WATERFALL AVOIDANCE

Detailed plans with tightly coupled dependencies inevitably fail to account for everything. Worse, they lock-in tasks and project requirements that prove to be irrelevant later in the process. Planning-based processes are more likely to get bogged down in the "administrivia," low-value paperwork rather that distract teams from achieving progress. Detailed planning of technology projects is often called **"waterfalling,"** a reference to the way Gantt charts stages and dependencies look like a cascading waterfall. Mixing rapid-development and planning processes is often called "agilefall," a term of rebuke for project models that inherent the worst of both worlds.

If there are checklists of requirements and tasks to be implemented, make checklists. It is better to keep track of those items using whatever **task-management** approach the team has adopted. Implementation can then be sequenced in more flexible ways.

“THE SANDBOX” VS. PROBLEM FRAMING

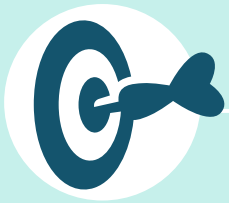
“THE SANDBOX”

Basic project parameters, not plans

Scoping
What substantive topics are in and out of bounds, taking care not to overly constrain the subject early on?

Resourcing
What time, funding, and personnel constraints does the team have to work within, at least initially?

Goal Setting
What is the first-pass description of the unsatisfactory state-of-affairs the team is trying to remedy, with an emphasis on challenges which are considered a priority?



CLEAR AIMS

New problem framing, not solution

Challenge
What are the main challenges that the service will address, from both a client- and public-health perspective?

Roots
What are the underlying causes of the challenge, including major systemic inter-dependencies and potential side-effects?

Perspectives
What are the contested issues and priorities, including myths and misconceptions to be addressed? How do different client- and stakeholder groups see these issues? Which views have been marginalized and why?

Futures
How is the subject area evolving? To what extent is the challenge a moving target that requires a flexible service which adapts?

Obstacles
What factors will likely impede change or present design trade-offs, with an aim towards turning constraints into opportunities?

Goal Setting
What is the unsatisfactory state-of-affairs to be remedied, with reference to the “end game,” or ultimate public good to be achieved?

When a *Data Service Team* is first proposed, an **initial direction** is negotiated between the Intake Coordinator and would-be Design Lead, with consideration given to the priorities of senior leaders in the Agency. Caution is exercised to not overly constrain the project goals before the team has a chance to involve clients and properly explore the substantive issues. A project is set up to fail if it is rigidly committed to solving the wrong problems. The whole point of the first diamond in the double-diamond model is to prevent that from happening. Thus, it is better to talk loosely about the “sandbox” the team is to play in. The sandbox is a first-pass description about what the main concerns are without jumping to solutions. Any initial problematique may be too broad or narrow in scope. For example, if the topic is mental health, is it realistic to expect the service to track the hundreds of conditions recognized by the psychiatric community (in the DSM-5 inventory), many of which are hard to diagnose? Or should a subset of pressing challenges be a priority, such as teen suicide and substance addiction? The point is to establish shared expectations about the general scope, while keeping in mind that direction, goals, resources may have to be adjusted as new understandings emerge.

By the time the team has completed the first diamond in the process, it is expected that the problematique is clearly defined and everyone understands what surveillance challenges the service is expected to address. That **informed framing** of the challenge also does not jump to solutions, even if early deliberations raise some promising options. The point is to develop a concise statement of team’s direction that takes into account the various client and stakeholder perspectives on the issues. The first version of the service may not address every need and want. The project has to have viable aims. Nevertheless, the various demands for data services taken into account and many that do not make it into the service may be added in future iterations.

BUILDING BLOCKS, NOT RULE BOOKS

Before describing the detailed stages, it is worth reiterating a few general principles. One reason to invest in design thinking is so that a *Data Service Team* can **design the design process**. Every project is expected to be different to some extent. However, all teams are expected to account for how all nine of the essential touchstones have been addressed (page 9). For example, designing a services that are not human-centred is not an option. Likewise, good data management principles is not an expendable luxury. The way these touchstones are implemented is expected to vary but all projects are designed around them. The overall logic of the double-diamond process is intended to make sure that happens (see next page).

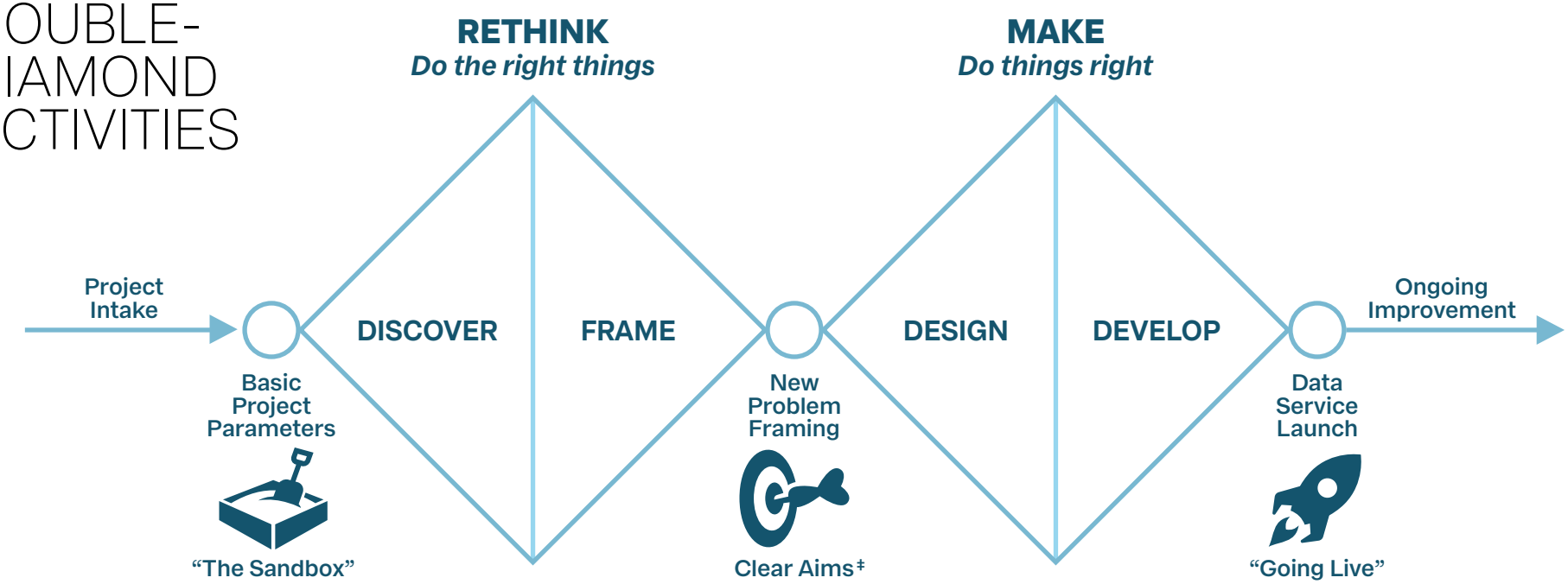
In a similar spirit, it helps to create **building blocks, not rule books**. Anyone can stipulate a set of arbitrary rules that ought to guide the development of services. No one will follow them if they are not reasoned and are helpful. It is better to bake good practices into the actual tools that are in use. That is one reason why the process makes use of playbook plays (see sidebar). The use of reusable software and style libraries is another example of that principle. So is abiding by good standards. As Data Service Teams build up their own toolkits from experience, using the tools makes doing the right thing the intuitive and easy.

None of that means that **discipline and diligence** can be abandoned. An anything-goes attitude can cause teams to lapse back into comfort zones. Anti-patterns seep back into processes. Libraries and standards lose their *raison d'être* if allowed to proliferate. Before long, teams begin to flail and systems begin to lose their coherence. Avoiding such problems is not about encoding everything into rules, which often become obsolete and an end unto themselves. It is about adopting good habits and peer supports as part of an overall culture of continuous learning.

PLAYBOOK PLAYS

Playbooks contain dialogue-based exercises and analytical activities (plays) that can help the project along. An analogy is made with team sports whereby coaching staff compile go-to tactics into a single reference book, which are chosen depending the circumstances. A team does not have to use plays. An experienced team will develop its own repertoire. Even so, off-the-shelf plays can still be helpful, especially for engaging clients who are unfamiliar with design thinking. A list of playbooks can be found in the next section (see page 77). A few words of caution are in order. First, filling out templates should not be a turgid exercise in form-filling. The point is to generate productive dialogue and if the play is not fulfilling that purpose, move on to another one. Second, plays are not recipes, merely foils with which to spark conversation. If a free-ranging dialogue better serves the team, set the plays aside.

DOUBLE-DIAMOND ACTIVITIES



ON-BOARDING HAND-OFF
Sandbox stipulated and project started

CORE TEAM ASSEMBLY
Recruitment of full-time team members

ORIENTATION
Briefings on design and technical issues

STOCK-TAKING
Sorting out current state-of-affairs

CLIENT AWARENESS
Mapping the client base and identifying clients

CLIENT INVOLVEMENT
Inviting clients into design circle as co-creators

USAGE SCENARIOS
Understanding the worlds of clients

THE END-GAME
Understanding public health policy priorities

FIELD WORK
Research to fill in the gaps in knowledge

CLIENT PROFILING
Sorting out clients and promoting inclusion

CLIENT JOURNEYS
Understanding clients' stories at the task level

REFRAMING
Rethinking how data can be used to further goals

FORESIGHT
Thinking about how the subject area is evolving

SYNTHESIS
Developing a concise problematique with clear aims

IDEATION
Brainstorming service ideas

CONSTRAINTS
Understanding limits and design trade-offs

HUMAN-FACTORS
How human diversity is accommodated

SERVICE CHANNELS
Selecting the media and technologies

MOCK-UPS
Making promising ideas more tangible

PROTOTYPES
Test ideas with functional models

USER EXPERIENCE (UX)
Evaluate prototypes from clients' perspective

ARCHITECTURE
Make needed changes to back-end platforms

OPENNESS & SECURITY
Set access policies and safeguards

SHARING & RE-USE
Use, extend, and share software libraries

ALPHA BUILD
Early working version of service

FIELD TESTING
Test with actual clients in realistic setting

BETA BUILD
Fully working version with work to be done

OFF-BOARDING HAND-OFF
Transfer service to program maintaining it

LAUNCH
Allow clients to use and make minor adjustments

CONTINUAL IMPROVEMENT
Iterate service based on usage and feedback

ON-BOARDING HAND-OFF

With the help of the In-take Coordinator, the members of the surveillance program (data suppliers) and the Service Design Lead come together. This is a "warm hand-off" (see page 28) in which everyone involved in project inception takes part. The aim is to set expectations for the project so that everyone shares the same initial understandings. The project's "sandbox" (initial thinking and parameters) is negotiated. Care is taken to not preempt the early exploration of the problem space with parameters that are over-determined.

- *What is the project "sandbox"?* What is the initial scope of the project (topics that are "in" and "out" of consideration, plus sensitivities to be observed)? What resources are available (time, funding, and talent allocations)? What is the general goal, or the first-pass description of the unsatisfactory state-of-affairs the team is to remedy? What are the early aspirations of key stakeholders?

CORE TEAM ASSEMBLY

With the basic project parameters stipulated, the Service Design Lead recruits members of the core team (see page 26). Members of the core team will work full-time on the project (more or less) so that the team can move quickly. A few highly specialized roles may likely require less of a time commitment. Even so, the intent is to assemble everyone early so that they can take part in the initial exploration of the problem space, even if they will mostly be working on late-stage development. As the project proceeds, additional talent needs may emerge and other specialists will be recruited.

- *Who should be involved?* Who should be invited into the core team to create a lean, fast-moving unit that does the project's "heavy lifting"? Who should be involved in the extended team, or those stakeholders necessary for the teams success (suppliers, clients, and platform developers)? Who should be involved within the broader network (outside experts and sectoral stakeholders)?

ORIENTATION

The Data Service Team process will be unfamiliar to many within the design circle. Some may not be aware of service design and rapid development methods. There is a risk that would-be collaborators revert to unproductive work patterns that undermine the project's chances of success. Thus, there is a need to orient everyone to the ways of working that are more common to design studios and lean start-ups.

- *How familiar is everyone to design thinking?* What aspects of behavioral research, service design, and rapid development should be reviewed so that everyone has roughly the same understanding of the *core operating modes* of the team.
- *What learning materials can get everyone up-to-speed quickly?* Are there presentations, hand-outs, and pointers that are particularly helpful? How can these ideas be shared without giving everyone a lot of homework?
- *What technical issues should everyone be made aware of?* For example, technology security issues.

STOCK-TAKING

The project begins by delving into subject area. Issues and controversies are identified. Relevant documents are compiled and reviewed. Important scientific details may need to be understood. The surveillance program will have a history and legacy systems, which should be mapped out. Ideally, everyone gains a common understanding of the program's background.

- *What is known and unknown?* What documents and frameworks exist? What technical concepts and data measurement issues? What history? What has not worked? What gaps in our understanding?
- *Who are the players (stakeholders)?* What are their assumptions, expectations, ambitions, and issue-framings? What are the points of contention?
- *How do we "first do no harm?"* What good work has come before? What laws, regulations, and unspoken rules? What political sensitivities and no-go zones? What myths and misconceptions?
- *What does the existing data pipeline look like?* What partners, dependencies, and technologies? What security vulnerabilities?

CLIENT AWARENESS

The clients of the service are identified. That is partly done to recruit clients into the process. It is also done to get an initial sense for the entire user-base for the data. The various clients can then be mapped out.

- *Who are the direct and downstream clients?* Who uses the data directly? Who are the downstream beneficiaries of the data? What is known about the various clients? How can more be learned? Who can become involved in the project and how?
- *What segments exist within the client base?* How can the clients be grouped for purposes of designing the service? As it relates to the data, what do different groups of clients to think? Do? See? Hear? Say? Feel? How can these perceptions be checked with real clients? What myths about clients need to be debunked?
- *Which clients are of strategic importance?* Who should be engaged to further health objectives? Are they current users of the data? Should they be?

CLIENT INVOLVEMENT

Designers cannot rely on preconceptions about client needs. Real clients have to be engaged to understand their world. The best way to do that is to invite clients into the design circle as genuine co-creators, acknowledging that a small team is not going to capture the full diversity of the client base. A broader array of clients may become involved as research subjects or test subjects. Once clients become involved, care is taken to respect their time, especially those from outside the Agency. As a general rule, clients should be involved in deliberations about subjects that interest them most.

- *How do clients understand the issues?* What problems do they face in their worlds? What roles and responsibilities do they have with relevance to the service?

USAGE SCENARIOS

The clients' context of data use is explored. Data ultimately has to be integrated into the workflows and lifestyles of clients. There will likely be some alteration of existing patterns of activity so that clients can make the most of the data. Sometimes, whole processes may have to be reformed. However, if the data service can complicate existing patterns, it reduces barriers to adoption and learning, plus it makes the data more helpful immediately. All told, the idea is to better understand clients by understanding the worlds?

- *What are the usage scenarios?* What tasks are performed? What jobs are to be done? What time pressures and other contextual constraints? What cultural norms and habits? What ethical implications?
- *What behavioral science insights?* How do clients *actually* think and act within the service scenario? What important cognitive and cultural details may have a bearing on the service but tend to escape notice?

THE END-GAME

The service to be created is client-centred insofar as it takes clients as the starting point of design and works backwards from there. Both direct- and downstream clients have their own goals and agendas. However, there are larger public-health policy goals to serve. What are they? How can client activities be brought in line with these overarching goals?

- *What is the policy end-game?* What are the ultimate policy objectives of the service? What patterns of thinking and behavior should the team try to promote? What would count as a worthwhile contribution to public health outcomes? Who is setting these goals and on what basis are they being set?
- *How can alignment be achieved?* How do existing patterns of client behaviour align or conflict with the desired outcomes? Are there ways to make downstream clients the direct users of data so as to better serve policy goals?

FIELD WORK

Many clients cannot be brought into the design circle for various practical reasons. Thus, the team has to go to them. Field work provides an opportunity to get a larger cross-section of client input. Research data is gathered about clients and the context of data usage and reported to the whole team. Ethnographic analysis picks up on the subtler cultural features of that context, including implicit understandings and socio-technical dynamics of technology usage. Ideally, vivid scenarios are developed so that team members can imagine how the data could actually be used within various settings and circumstances.

- *What clients and sites should be researched?* What types of data should be gathered to fill gaps in the teams understanding? How are service-relevant tasks performed under realistic conditions? What complications can be observed? What sources of information that can be tapped to provide a clearer picture of the clients' worlds? What clients and sites would make ideal test subjects later on?

CLIENT PROFILING

It may make sense to formalize the research on clients by turning it into something handier for design and development work. For example, core clients can be summarized into client personas, which are concise profiles that describe clients and their circumstances. The main benefit of profiling methods is that they put a name and a face to a real-world client that can be a passive presence throughout the project. Everyone is then encouraged to make decisions with such profiles in mind instead of relying on vague, self-serving stereotypes. The client base can also be segmented into different groups now that research is in hand.

- *What client characteristics relate to the service?* For example, what is their data literacy? What technologies do they use? What data visualization methods are familiar to them? What mental models do they use to understand the health issues?
- *Who are the neglected clients?* What are the edge cases? What forms of marginalization? What special needs?

CLIENT JOURNEYS

Any service will involve a sequence of steps, including discovering the service in the first place. It is crucial to get a sense of the mindsets and mental models of clients as they are actually performing tasks that can benefit from data. The point is to identify difficulties and awkwardness in the way things are currently done, with an eye on finding better ways to do things.

- *What are the steps in the service journey?* If there is a preexisting set of routines, how does the flow of tasks work currently? How can the whole journey be streamlined by taking out unnecessary steps and removing obstacles? If a new set of routines is imagined, how would things likely play out? What do actual clients think about that scenario? Are there similar data services that can serve as a guide?
- *What are the main pain points?* What obstacles, frustrations, unnecessary burdens, costs, and so forth? How is data currently be under- and misused? What harmful side-effects accrue to non-clients (externalities)?

REFRAMING

Issues are reinterpreted given the research and insights gathered. The characterization of the problem is reformulated to address underlying root causes, not just surface symptoms, plus clients' needs and challenges. The overall intention is to make sure the team is doing the right things. Often, the initial request will be revised significantly to address a better focused (and often more ambitious) challenge.

- *What is data used for?* Does it shed light on the right public health problems? How are those problems best understood? What are the practical data implications? What problems, clients, and use-cases are being neglected?
- *What are the current data deficiencies?* What is the "nice to know" data and the "need to know" data? Is data streams that should be deprecated because it is of little use? Is there needed data that is not being gathered? Why? Are there methodological changes to be made while developing a new data service?

FORESIGHT

Whatever challenge that the team is addressing is a moving target. For example, if the topic is anti-microbial resistance, that resistance is happening due to usage patterns of anti-fungal and anti-bacterial agents. How are those patterns changing? For example, are farms and ranches becoming more dependent on the agents? Are there viable alternatives? A data service may become prematurely obsolete if the team is not thinking about where the problem space is heading in the future.

- *What are the ongoing and emerging trends?* How are demographics and public attitudes changing, including on the topic of surveillance and privacy? How is the public health space evolving?
- *What future scenarios?* What may happen if the problem is not adequately addressed? What ideal scenarios can be encouraged?
- *What emerging technologies?* Can the project leverage them? Or be adapted once those technologies are sufficiently viable?

SYNTHESIS

The initial understanding of the problem is revisited and a full problem statement is developed. This statement brings together the various threads of discussion to put forward a vivid picture of what is wrong with the current state of affairs and what can be done to improve matters.

- *What are the roots of the problem?* How is the problem understood from the perspective of various clients and stakeholders? What competing priorities and underlying interests have to be managed? How is the problem space evolving and what emerging issues have to be addressed?
- *What are the obstacles to change?* What tensions and paradoxes? How can constraints be reformulated into opportunities?
- *What are the project objectives?* What would success look like? How will the team demonstrate that success has been achieved?

IDEATION

The team brainstorms ideas for services that may address the problem. The aim is to generate as many ideas as possible by temporarily setting aside critical judgement (see page 10). Additional research may be conducted to seek out ideas, such as looking at what is being done in other jurisdictions.

- *What data service would meet clients' needs?* What is the overarching concept, theme, or metaphor that would make the solution recognizable and appealing? What is the elevator pitch? What technologies would be used? What would the solution look like as a basic sketch? What is the value proposition? That is, what is the marquee benefit to the client as they better integrate data into their routines?
- *Who is doing cutting edge work in this area?* What sources of inspiration exist? What are others trying elsewhere? What interesting ideas and concepts are circulating?

CONSTRAINTS

Ideas are made more viable by adapting them to cope with various logistical requirements and constraints. Various tensions and trade-offs emerge that will have to be reconciled. Constraints are not treated as an excuse not to do something but are, instead, treated as an opportunity to think creatively about what would be the most elegant solution. The process of *design critique* involves evaluating proposals based on reasoned criteria, all while constructively generating ideas to improve the design.

- *What are the design constraints?* How would clients be constrained within the context of use? What are the project's resource constraints? What stylistic constraints (such as brand and web standards)? What technical constraints? Ethical constraints? Environmental constraints?
- *What design requirements?* What security and privacy requirements? What accessibility requirements? What technology standards are relevant?

HUMAN FACTORS ANALYSIS

Human diversity is factored into the design proposal given that there is no such thing as an "average" or "normal" user. Various usability features are incorporated into the design to make the service easier to discover, learn, and use. Interfaces are considered with an eye towards adding affordances that complement the way people actually think and move (*ergonomics*). The desired service experience is spelled out from the perspective of clients.

- *How can burdens be minimized?* Cognitive and physical burdens? Contextual demands and distractions?
- *What enablers make the service easier to use and understand?*
- *How are special needs accommodated?* Physical or developmental impairments? Are there ways that the setting or technology "disables" people by being blind to human variation? Are there win-win options, whereby catering to variation makes the service experience better for everyone?

SERVICE CHANNELS

A data service will stream data through one or more channels (technological modalities). For example, the data service could be expressed through an online medium, such as a phone app, messaging notification, on-site digital kiosk, web-based data dashboard, interactive data visualization, and so forth. The digital revolution also provides options for novel forms of physical publishing. Ideally, services are *omni-channel*, meaning they cater to all the media that are popular among clients.

- *What are the priority service channels?* What features and constraints does each offer? Are any channels required because of official policy or law? What are client's preferences?
- What the access implications? Is anyone excluded due to physical, developmental, or socio-economic issues? Do clients have access to the right technologies? What levels of channel familiarity and ability are involved? What learning is involved? How does that fit within the usage scenario?

MOCK-UPS

The vagary of words can make proposals ambiguous. Quickly sketching out options and assembling models can add clarity. The resulting mock-ups can also be used to sell an idea to sceptical stakeholders and clients by showing the art-of-the-possible.

- *What physical forms could each option take?* What medium? What visualization methods? What interface metaphors? What organizing logic? What useful features?
- *What is the signature (or ideal) usage experience associated with each option?* What are the marquee benefits and how are those made apparent to clients? How does the presentation of the service express that appeal?

PROTOTYPES

Once the most promising proposal has been selected, the mock-up is turned into a prototype with some functionality. How functional and realistic depends on the tools available and the time constraints. The sooner the team is able to simulate the experience of using the service (or an aspect of it), the sooner they can test it with real clients. As new insights emerge, the prototype is refined through iteration until it makes sense to proceed to building. The prototype will usually add clarity about the types of behind-the-scenes infrastructure that will be required. Some of these may have to be prototyped too.

- *What forms can the prototype take?* What tools are available? What physical forms? What crucial features are left out because they cannot be simulated adequately? What aspects of the setting should be included too?
- *What experimental features need a proof-of-concept to establish viability?* What conditions must be met to know the feature will work?

USER EXPERIENCE (UX)

As the prototypes and builds advance, the specifics of the user interface are refined. Yet it is not just the display of data that has to be clear. The overall interpretation and analysis has to be meaningful within realistic context of use. Various visual storytelling techniques can make that happen. Those techniques may reveal major errors of omission, such as the lack of benchmarks or comparative data that would give the data meaning. The data story should answer the, “What? So what? Now what?” questions.

- *How will clients interact with the service from start to finish?* How are they on-boarded? How will the service wrap up? What other aspects of the client journey have to be thought through?
- *What affordances could be added?* To make it easier to learn, discover, and use? Where are clients struggling? What unnecessary burdens can be removed?

ARCHITECTURE

Various back-end systems will need to be in place for the full data supply chain to work. As more data platforms are offered within the Agency, it will be easier create to service on top. Often, there will still be a need to develop back-end subsystems to cater to the specifics of a data service. These will have to follow protocols and integrate well within existing platforms. In any case, platform developers should be engaged early and often to make sure the service will function as intended and other services can benefit from whatever new components built. Administrative procedures also make up the back-end and will have to be developed.

- *What is the software stack the service will run off of?* What components are already in place? What standards apply? What emerging technologies should be supported? What reusable components can be repurposed to create underlying capability?
- *How can data flow with minimal delay?* What processes have to be streamlined? What can be automated?

OPENNESS & SECURITY

Even though a lot of hard work went into prefiguring the data service, not every potential use case can be anticipated. Open data streams allow others to experiment with the data and develop alternative applications. The service will also have to be made inherently safe, both in terms of clients' security and the security of government systems.

- *How is openness handled?* How can data be made open for all to use? What APIs are developed? How can software components developed for the service be shared with others? What are the reasons behind any restriction? How can restrictions be less burdensome?
- *How has security been built into the service as a priority?* How does quality control work more generally? What privacy measures have been included to ensure the data will not be used inappropriately?
- *How are you showing your work in progress?* What methods allow colleagues and stakeholders to see what you are doing and offer advice?

SHARING & RE-USE

In order to make the most efficient use of resources, developers will not reinvent the wheel with every module and sub-routine. Various open-source modules will be used. New modules that are built will be uploaded to repositories and made available for others to use. Re-use is a two-way street. Moreover, lessons about the overall project (both successes and failures) will be shared with others doing similar work, all in the spirit of organizational learning.

- *What software components can be reused?* How trustworthy is the source? How mature is the component? Is there a community devoted to ongoing development? Where does the software have to be tweaked to suit project needs?
- *What software can be contributed to the community?* What can be reused within other projects?

ALPHA BUILD

The rapid development process is highly iterative, with the project broken down into smaller chunks that can be worked on in short bursts of effort (“sprints”) according to Agile principles. The aim is to get a “clickable” version up and running as soon as possible. A large share of the innovation in a project is by working through trade-offs and refining functionality during the building stage.

FIELD TESTING

Some testing will happen as would-be clients are brought into the development space. A larger array of clients can get involved. It is crucial that tests happen with actual clients, not test-subjects that are more convenient. There is also no substitute for testing on-site, in the actual setting of use under realistic circumstances. Some aspects of the context may turn out to be major obstacles that will have to be worked around. Lots of minor refinements are to be expected. Some stress testing of the system overall should also happen so that post-launch surges in use do not cause problems.

- *What real-world situations will adequately stress test the service?* What demanding usage scenarios? What edge cases might cause confounds in the system? What users will likely require additional guidance? What scaling challenges?

BETA BUILD

The beta build is a fully working version of the service. There may be non-vital features to add. There will be plenty of bugs to squash. Nevertheless, this version can be released to a larger group of testers and stake-holders for their input. Client requests can be solicited. Major modifications (that are not show-stopping problems) may have to wait to a future release. The goal is to get to the final release candidate as quickly as possible without unduly compromising quality.

OFF-BOARDING HAND-OFF

Once the beta build is completed, the Data Service Team will start to disband. Full control of the project is handed back to the original program. Throughout the process, the intent is build capability within that program. If members have participated as genuine co-creators, many research, design, and development skills have been learned. Even so, the program may still have some technical shortcomings to bring the service to a full release and, as importantly, continue to improve it after launch. It may make sense to let a developer become part of the team for a longer spell until the service has emerged. This is also a period in which the developer can document the system so that successors can easily continue the work.

LAUNCH

The product is launched and a wider variety of clients are able to use the data service. A larger cross-section of clients provide the varied experience to identify more bugs and edge cases. Updates follow quickly to keep the service in working order. The broader client base will also make suggestions for additional features or suggest future projects. Having a passive “build it and they will come” attitude will ultimately cause a data service to go under-used. Some outreach and relationship-building efforts are expected. Communications and educational materials may have to be created. Ideally, the team reaches out to various strategic allies and stakeholders early in the process to to generate demand.

CONTINUAL IMPROVEMENT

Working with “live” code is a different model of development that does not allow for major disruptions. Once a data service is running, it is easier to do A-B testing that simultaneously offers different versions of the service (or features thereof) to see what works best. New features and improvements are added over time as the data service evolves to meet clients’ changing needs.

AN ITERATIVE & EVOLVING PROCESS

This sequence of stages is presented as a through-line to make the various tasks easy to grasp. However, the double-diamond should not be mistaken for a **linear process**. Processes will iterate, as with the refining of ideas, prototypes, and service builds. A team may have to take a step backward to do rework in order to move forward. That is all part of the process. Attempts have been made to show that inherent messiness and improvisation in diagrams but that usually winds up creating confusing picture. Regardless, *Data Service Teams* should always keep the flexible, non-linear, and iterative nature of the process in mind. Indeed, what might seem like chaos to those operating from a planning mindset are in fact methods for managing risk, this guide's keynote (page 2) and capstone (sidebar) message.

It is also crucial to note that this sequence of steps is not set in stone. It is expected that the **process will evolve** as the Agency gains more experience building data services. Accordingly, details in this guide will continue to evolve as lessons are drawn from team experiences. More detailed descriptions of pipelines, infrastructure, and development models will also be added as new pieces are put into place. New learning materials are published all the time and lists of resources are updated. Expect further updates to this field guide along those lines.

All that to say, *Data Service Teams* operate with an ethos of continuous improvement. That is why time should always be set aside to reflect on what is working and what is not, while being attentive to emerging possibilities. These lessons ideas are shared through networks and in repositories such as this one.

RISK MANAGEMENT

Dialogue-base, rapid development processes are fundamentally about **managing risks**. Teams move swiftly to try out promising options through trial and error as part of a *"fail fast" strategy*. Better to discover shortcomings early on. Problems found late in the game pose a *sunk-cost traps*, whereby the team is reluctant to change direction or redo work because that amounts to an admission of wasted effort. The build-up of those shortcomings is called "technical debt" because it creates costly rework later on. Exploratory research and design also looks out for systemic side-effects caused by new service. Failing to do that early is often called an "ethics debt," with unanticipated problems later coming back to bite a service, often in the form of public controversy. Even use of open-source libraries is a form of risk management that reduces bugs, security flaws, and wasted effort.

6 WHAT **ADDITIONAL RESOURCES** CAN HELP?

SERVICE DESIGN RESOURCES

Lou Downe, *Good Services: How to design services that work* (Amsterdam: BIS Publishers, 2020).

If you are looking for a basic overview of service design that is jargon free, this outcome-oriented guide is a good place to start. Several design principles are offered.

Lara Penin, *Designing the Invisible: An Introduction to Service Design* (New York, NY: Bloomsbury, 2018).

This process-oriented guide offers a more comprehensive introduction, including an elaboration of the double-diamond logic model.

Chris Risdon and Patrick Quattlebaum, *Orchestrating Experiences: Collaborative Design for Complexity* (New York, NY: Rosenfeld, 2018).

This medium-oriented guide to service design takes a detailed look at how services are experienced by clients within a context (or usage scenario).

Adam Connor & Aaron Irizarry, *Discussing Design: Improving Communication and Collaboration Through Critique* (Sebastopol, CA: O'Reilly, 2015).

Offering constructive feedback on a design has nothing to do with sharing personal opinions or off-the-cuff "takes." Design critique is a method. Teams stay harmonious by scrutinizing with sensitivity to social dynamics and design goals. Good and bad practices are discussed.

Chris Nodder, *Evil By Design: Interaction Design to Lead Us into Temptation* (Hoboken, NJ: O'Reilly, 2013).

Dark patterns are underhanded, manipulative ways of presenting information on service interfaces. They are common in commercial services. Too often, they find their way into public services through reusable components or the influence of common practices. This guide lists patterns of bad design so as to avoid them.

Jonathan Shariat & Cynthia Savard Saucier, *Tragic Design: The Impact of Bad Design and How to Fix It* (Sebastopol, CA: O'Reilly, 2017).

Service administrators underestimate the extent to which bad design creates genuine harm and suffering, even with the best of intentions. This guide also covers *dark patterns* but also forms of shortsightedness (such as errors and lack of inclusiveness) that can result in tragic outcomes.

Donald A. Norman, *Living with Complexity* (Cambridge, MA: The MIT Press, 2011).

Making a service simple, useful, easy to use, and emotionally rewarding is a worthwhile goal. Yet eliminating all complexity is not the point of design, eliminating confusion and frustration is. Moreover, making services too simple can cause all sorts of harms. This book explains why while reviewing the various forms of complexity that designers have to cope with. Also of interest are Norman's other books, notably *Design for a Better World* (2023).

INCLUSIVE-DESIGN RESOURCES

Kat Holmes, *Mismatch: How Inclusion Shapes Design* (Cambridge, MA: The MIT Press, 2018).

Services frustrate and exclude when the stated goals *mismatch* the realities of users. Inclusive design involves aligning human diversity to: (a.) why we make; (b.) who makes; (c.) how we make; (d.) who uses; and (e.) what we make.

Sara Hendren, *What Can a Body Do? How We Meet the Built World* (New York, NY: Riverhead books, 2020).

Services can be disabling when bodies fall outside what is considered "natural" or "normal." Exclusionary *misfits* occur when designers rely on the *aggregative fallacy*, or mistaken assumption that regularities within a group hold true for any individual.

Ellen Lupton & Andrea Lipps, *The Senses: Design Beyond Vision* (Hudson, NY: Princeton Architectural Press, 2018).

As mentioned (on page 20), visualization is only one way to make data stories better engage the senses. There are many multi-media tools, but how to design for them? This collection of essays explores the role of other sense and the clients who rely on them.

First Nations Information Governance Centre, *The Fundamentals of OCAP* [online course]. (🔗)

Data governance for First Nations, Inuit, and Métis (FNIM) communities involves applying the OCAP principles (Ownership, Control, Access, and Possession).

John Whalen, *Design for How People Think: Using Brain Science to Build Better Products* (Sebastopol, CA: O'Reilly, 2019).

Designing services involves working within the cognitive and emotional constraints of how humans experience the world, such as limits to memory and directional sense. This guide also covers these constraints within the context of double-diamond processes.

Julie Dirksen, *Design for How People Learn* (Hoboken, NJ: New Riders, 2016).

The biggest barrier to designing good services is the assumption that client knowledge and habits. Thus, understanding the client's journey with a service is as much about understanding how they learn about a service and adapt it to their preferred ways of doing things.

Daniel Stalder, *The Power of Context: How to Manage Our Bias and Improve Our Understanding of Others* (Amherst, NY: Prometheus Books, 2018).

Not all patterns of human behaviour are enduring. Temporary circumstances matter a great deal. We intuit that when explaining our own behaviour but are less willing to consider context when judging others. Circumstantial psychological dynamics should be considered when constructing the context of a service. Care should also be taken not to assume too much of clients motivations based on surface attributes (a danger when constructing client personas).

Peter Jones, "Evolutionary Stake-holder Discovery: Requisite System Sampling for Co-creation," *RSD7 Symposium* (2018). (🔗)

Who should be involved in a data services project? It is often necessary to involve people from across the system and various walks of life. This working paper looks at the various stakeholder groups to consider involving so that the project reflects "social requisite variety."

FIELD-RESEARCH RESOURCES

Billy Ehn, Orvar Löfgren, Richard Wilk, *Exploring Everyday Life: Strategies for Ethnography* (New York, NY: Rowman & Littlefield, 2016).

Understanding clients' worlds involves taking an interest in tasks and routines that usually escape designers' attention. Indeed, they usually escape the clients attention too, as we all act on "autopilot" much of the time. Ethnography is a bundle of culture-sensitive methods for understanding the behaviours and mindsets of diverse clients so that services do not frustrate and exclude. This guide is a good place to start.

Sam Ladner, *Mixed Methods: A Short Guide to Applied Mixed Methods Research* (Sam Ladner, 2019).

A former-Microsoft design ethnographer explores using both qualitative and quantitative data (mixed methods) in the development of services within double-diamond processes. Also of interest is Ladner's introduction to design ethnography, *Practical Ethnography* (2014).

Indi Young, *Time to Listen: How Giving People Space to Speak Drives Invention and Inclusion* (San Francisco, CA: Indi Young Books, 2022).

Developing client empathy involves deep listening, often listening to clients with very different life experiences and mental models. The danger is just not recruiting a diverse enough sample of clients to listen and then listening selectively to their accounts. This guide aims to avoid that with listening, recruitment, and synthesis strategies. Also of interest is Young's books on clients' mental models (*Mental Models*, 2008) and empathy (*Practical Empathy*, 2015).

Jan Chipchase, *Hidden in Plain Sight: How to Create Extraordinary Products for Tomorrow's Customers* (New York: Harper Business, 2013).

There is a method to noticing the social-psychology of how people interact with products and services. There is also a duty for service designers to get out of the office and go to where the clients are. This introductory guide covers both. If you do client research in field, Chipchase's advanced guide (*The Field Study Handbook*, 2017) is recommended.

DIALOGUE-BASED DESIGN PLAYBOOKS

Peter Jones & Kristel Van Ael, *Design Journeys Through Complex Systems: Practice Tools for Systemic Design* (Amsterdam: BIS Publishers, 2022).

Many guides fail to account for the entangled, systemic nature of designing services. This guide fills the gap, with thirty plays falling under the topics of framing, listening (co-creation), understanding systems, foresight, possibilities, change, and ongoing adaptation.

Templates: *Systemic Design Toolkit* templates (🔗).

Elizabeth Sanders & Pieter Jans Stappers, *Convivial Toolbox: Generative Research for the Front End of Design* (Amsterdam: BIS Publishers, 2013).

This guide covers creativity and design research with the help of generative "make-tools" (plays), including advice about building your own toolkits (playbooks). The focus is on early exploration of the "opportunity space."

Annemiek van Boeijen, Jaap Daalhuizen, Jelle Zijlstra, & Roos van der Schoor, *Delft Design Guide: Design Methods*–Second Edition (Amsterdam: BIS Publishers, 2020) (🔗).

Originally about product design, the Delft guide has expanded to include services. Useful plays include context mapping, SCAMPER ideation, W5H, cultural probes, the Datum method, strategy wheels, process trees, life-cycle analysis, opportunity areas, and morphological charts, plus a variety of conventional methods for design and management decision-making.

Dave Gray, Sunni Brown, and James Macanufo, *Gamestorming: A Playbook for Innovators, Rulebreakers, and Changemakers* (Sebastopol, CA: O'Reilly, 2010).

In addition to compiling plays, this guide offers advice about dialogue facilitation and visualizing ideas. The plays are a mix of classics and ones generated by the authors (and their colleagues). The guide is best known for introducing the world to empathy mapping, which is the closest thing to a mandatory play for any design process.

Templates: Gamestorming plays wiki (🔗).

Ideo, *The Field Guide to Human-Centred Design–Design Kit* (San Francisco, CA: Ideo.org, 2015) (🔗).

An early version of this toolkit (2009) raised the profile of both human-centred design and playbooks in general. This version includes useful plays about inspiration, ideation, and early-stage making.

Government of British Columbia, *The Service Design Playbook* (Victoria, BC: Government of British Columbia, 2020) (🔗).

This guide provides cursory descriptions of popular plays and adds several traditional public management methods. The emphasis is on government digital services, with several templates in the annexes.

Alex Osterwalder, Yves Pigneur, Greg Bernarda, and Alan Smith, *Value Proposition Design* (Hoboken, NJ: Wiley, 2014).

This guide offers plays about client-centricity and the articulation of value propositions. It part of the Strategyzer series of visual playbooks, which includes business modeling (*Business Model Generation*, 2010), idea testing (*Testing Business Ideas*, 2020), and team dynamics (*High Impact Tools for Teams*, 2021).

Organisation for Economic Cooperation and Development (OECD), Observatory of Public Sector Innovation, *Toolkit Navigator* [wiki] (🔗).

Looking for more playbook plays? The OECD has a wiki of playbooks related to public-sector innovation.

RAPID-DEVELOPMENT RESOURCES

Kathryn McElroy, *Prototyping for Designers* (Sebastopol, CA: O'Reilly, 2017).

Exploring and testing ideas with mock-ups and prototypes is essential for making abstract ideas tangible in short order. This guide covers prototyping for both products and service, with various levels of realism (or *fidelity*, including visual realism, case breadth, case depth, level of interactivity, and data realism).

Marianne Bellotti, *Kill It with Fire: Managing Aging Computer Systems (and Future Proof Modern Ones)* (San Francisco, CA: No Starch Press, 2021).

Legacy systems pose a challenge for ongoing development, especially if they include lots of "technical debt" (awkward work-arounds, sub-par trade-offs, and makeshift patches). Replacing and reworking these systems is a development craft unto itself.

Melissa Perri, *Escaping the Build Trap: How Effective Product Management Creates Real Value* (Sebastopol, CA: O'Reilly, 2019).

Organizations get stuck in a *build trap* when they fixate on project outputs and service features, rather than outcomes. Outcomes include the value to clients (and other stakeholders), but also the side-effects of a service (including harms). This guide explains how to avoid the trap (and related anti-patterns) within rapid development processes.

Alla Kholmatova, *Design Systems: A Practical Guide to Creating Design Languages for Digital Products* (Freiburg: Smashing Media, 2017).

Pattern libraries (with design tokens) help teams reuse visual elements to create a common look and feel without much extra effort. This guide helps build a "design langauge" (or "UX design system") by understanding the foundations of this approach.

Pitfalls to avoid: Cathy Dutton, "The Problem with Patterns" *A List Apart* (2018) (🔗)

James Shore, Diana Larsen, Gitte Klitgaard & Shane Warden, *The Art of Agile Development*–Second Edition (Sebastopol, CA: O'Reilly, 2022).

This guide provides overview of the Agile approach, from general philosophy to the practicalities of organizing a rapid-development team.

Ryan Singer, *Shape Up: Stop Running in Circles and Ship Work that Matters* (Chicago, IL: Basecamp/37Signals, 2019). (🔗)

Agile processes can sometimes loose sight of core rapid development principles by becoming a highly corporate version of the methodology. It is worth consulting other rapid development methods to get a sense of perspective and retain the speed and flexibility. This guide offers a version of rapid development that also respects people's need for balance at work so as to not get burned out with a treadmill of "sprints."

Will Larson, *An Elegant Puzzle: Systems of Engineering Management* (San Francisco, CA: Stripe Press, 2019).

Combining systems thinking and project management can help teams develop quickly without losing sight of bigger-picture issues and adding technical debt. This guide is full of advice for would-be project managers based on hard-won lessons in the trenches of development work.

DATA STORYTELLING RESOURCES

Alberto Cairo, *The Functional Art: An Introduction to Information Graphics and Visualization* (Hoboken, NJ: New Riders, 2012).

Some general principles of information design are reviewed as part of a practical guide on data visualization. Emphasis is placed on making data meaningful to viewers. For an even deeper dive, Cairo's other books are recommended: *The Truthful Art* (2016), *How Charts Lie* (2019), and *The Art of Insight* (2023).

Stephen Few, *Information Dashboard Design*–Second Edition (San Francisco, CA: Analytics Press, 2013)

Good and bad practices involving data dashboards are covered while stressing foundational information-design principles.

- **Brief:** "Why Most Dashboards Fail" (2007) (🔗)
- **Brief:** "Common Pitfalls in Dashboard Design" (2006) (🔗)
- **Brief:** "With Dashboards, Formatting and Layout Definitely Matter" (2008) (🔗)

Drue Barrett, Leonard Ortmann, & Stephanie Larson, eds., *Narrative Ethics in Public Health: The Value of Stories* (Cham: Springer, 2022). (🔗)

Storytelling raises a number of ethical issues that go beyond simply maintaining scientific integrity of the message. This collection explores, from various perspectives, the messiness of how public health stories are received.

Andy Kriebel and Eva Murray, *#Makeover Monday: Improving How We Visualize and Analyze Data, One Chart at a Time* (Hoboken, NJ: Wiley, 2018).

Bad habits get in the way of good data visualization. Many of these habits exist simply because the software makes it easy to make poor choices. This book reviews various examples of poor chart design and while suggesting alternatives.

Jorge Arango, *Living in Information: Responsible Design for Digital Places* (New York, NY: Two Waves Books, 2018).

Data is consumed within a context of use that has implications for interpretation. This guide explains how to design data and information services in a way that is sensitive to various usage scenarios and, indeed, unanticipated contexts.

David C. Evans, *Bottlenecks: Aligning UX Design with User Psychology* (New York, NY: Apress, 2017).

Getting your point across to an audience involves overcoming psychological bottlenecks of various kinds (attentional, perceptual, dispositional, motivational, memory, and social-influence bottlenecks). This book explores the experiences that designers should aim to create to better connect and engage.

Lars-Erik Janlert and Erik Stolterman, *Things That Keep Us Busy: The Elements of Interaction* (Cambridge, MA: The MIT Press, 2017).

Adding interactivity to data interfaces should be done thoughtfully. Indeed, some services reduce demands on users by doing things automatically. This guide explores cases where interaction is engaging and where it is a nuisance.

Neil Richards, *Questions in Dataviz: A Design-Driven Process for Data Visualisation* (Boca Raton, FL: CRC Press, 2023).

This guide to data visualization leads with questions to explore some of the misnomers and esoteric issues associated with the field.

DATA SYSTEM RESOURCES

Donella H. Meadows, *Thinking in Systems: A Primer* (White River Junction, VT: Chelsea Green Publishing, 2008).

Systems are interconnected elements, organized to achieve something, that produce their own patterns of behavior. Systems thinking is the building of mental models about those elements and their relational dynamics to avoid dysfunctions, unpredictable complexity, dynamic traps, and harmful side-effects.

Zhamak Dehghani, *Data Mesh: Delivering Data-Driven Value at Scale* (Sebastopol, CA: O'Reilly, 2022).

A *data mesh* is an architecture for making use of large amounts of data that does not require a central repository. That decentralized arrangement involves: (a.) database ownership stays local and domain-sensitive; (b.) data is treated as a product for others to use; (c.) data is provided in a way that is self-serve; and (d.) bottom-up governance that retains cross-cutting requirements (such as interoperability, security, and so forth).

Brief: "How to Move Beyond a Monolithic Data Lake to a Distributed Data Mesh" (🔗)

Brief: "Data Mesh Principles and Logical Architecture" (🔗)

Hannah Fry, *Hello World: Being Human in the Age of Algorithms* (New York, NY: W. W. Norton & Company, 2018).

Algorithms embed contentious judgements and gimmicky decision-rules that can have profound ethical implications. These get added to services too readily and are trusted too quickly. Hidden judgement calls and unfairness embedded within systems need to be brought into the open and scrutinized.

Vlad Khononov, *Learning Domain-Driven Design: Aligning Software Architecture and Business Strategy* (Sebastopol, CA: O'Reilly, 2022).

It is a challenge to align clients' worlds, business needs, and software architectures while developing data services. Domain Driven Design (DDD) is a development approach to juggling those ideas within decentralized arrangements. This book includes a chapter on EventStorming, one of several plays for bridging client journeys and developer tasks (also see Event Modeling).

Various items: Event Modeling talks and resources (🔗).

Peter Morville, *Intertwingled: Information Changes Everything* (Peter Morville, 2014).

How data categories are constructed often dictates how a system works. Complex systems can also quickly loose the plot; that is, stop working in a way that is sensitive to users. This guide brings an information-architecture perspective to how diverse clients can relate to complex systems without having to muddle.

Evan Gilman & Doug Barth, *Zero Trust Networks* (Sebastopol, CA: O'Reilly, 2017).

This book goes into detail about the architectural difference between zero-trust networks and traditional perimeter defense approaches to security. All the crucial concepts and technologies are covered.

INDEX

Q

R

S

T

U

V

W