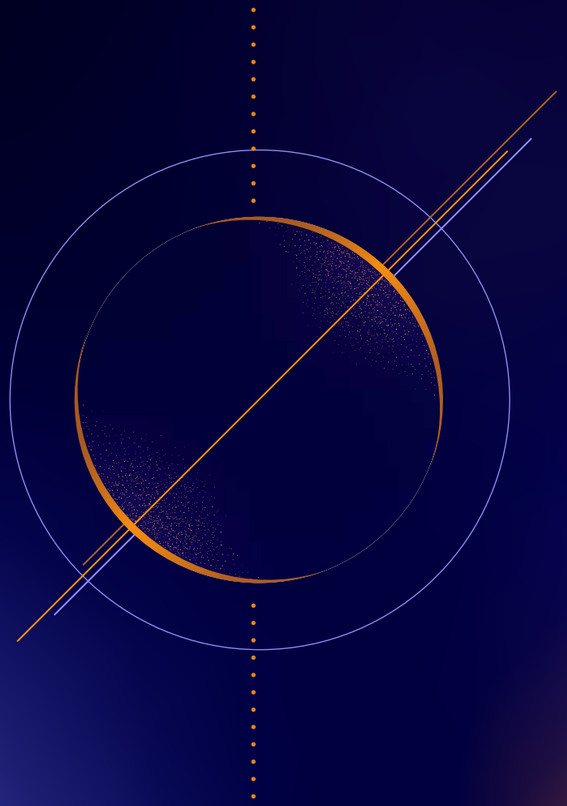

EXPLORING THE PROBLEM SPACE
A GUIDE TO BUILDING THE RIGHT SOLUTION



OUR BUSINESSES ARE ALL UNDER A COMPOUNDING PRESSURE TO ADAPT TO A MARKET THAT KNOWS MORE, WANTS MORE, AND HAS MORE CHOICE.

Technology has changed the traditional cost and time profile for everything we do. Competition is around every corner and it feels like "digital" means "we're all competing with everyone, on every product, all the time".

The lines are blurring. Banks have become mobile operators, mobile operators have become insurers, insurers have become banks. It's possible, and inexpensive, to launch new brands, offer new services, and engage with customers around the world. Exciting, but daunting. This pressure can quickly turn to hype, or failure.

To cope, we need to mix some of our old tricks with some new tools and new thinking. We need to leverage tech with ambition, and act with discipline as we plan our technology changes.

This publication shares **ideas and tools for how to spend time and energy properly exploring a problem prior to committing to the solution ahead**. We describe why and how to completely map out needs from every dimension in a way that results in better long-term IT investments while embracing iterative development.

MODERN TECHNOLOGY-BASED PRODUCTS AND SYSTEMS ARE DEEPLY INTERLINKED WITH DOZENS OF OTHER AREAS OF A BUSINESS, MORE THAN WAS THE NORM JUST 5 YEARS AGO.

The trend is for technology (or IT) to be less of a department of shared services, and more of a skillset within each business. We need business, product, finance, tech, execs and sales all in the same team and operating with the same vision.

Without this 'one-team' philosophy, our products just won't compete because they're too slow to build (or adapt), or worse, incoherent in the eyes of our customers or users.

With this need for relevance, speed and competition in mind, we have to be incredibly careful during the early stages of conceptualising every project and building every team. We need to deeply explore and understand every influence on our ambitions, and the needs arising from each of these lenses.

This can become a complex and exhausting list of demands on every project. We need an analysis approach which offers an effective way to unpack and explore a project concept without taking too much time and without missing any critical perspectives.

This publication shares a set of these tools and tactics which we use in our approach to problem solving.

Each of these tools will add detail to several dimensions in this list. The more you explore, the clearer the problem, and the more meaningful the basis for the solution becomes.

BUSINESS

BUSINESS MODEL

The influence that an operating model might have on how to build or operate technology.

BUDGETS AND TIMING

Is our project an experiment, or are we betting the farm? How much time do we have before we need to make the next critical decision?

PEOPLE AND POLITICS

The unavoidable power dynamics in business, agendas at play and influencers who must be satisfied.

HISTORY AND LEARNINGS

What has been tried before, and how did we arrive at this point?

CUSTOMER

CUSTOMER NEEDS

Is the voice, and the behaviour, of the customer readily available to us and do we know what they really want?

MARKET

INDUSTRY DOMAIN

The nature of the sector which we're operating in.

HISTORY AND LEARNINGS

What has been tried before, and how did we arrive at this point?

STRATEGY

LONG-TERM VISION

How does this project fundamentally move the business forward?

SHORT-TERM INTENTION

What does success look like in 16 weeks? Who are our first users and what critical feedback do we need to validate our project's existence?

BUSINESS STRATEGY

What other themes in the strategy of the business will this project support, or ignore?

TECHNOLOGY

TECHNOLOGY ECOSYSTEMS

How to re-use, improve or accommodate existing tech and infrastructure.

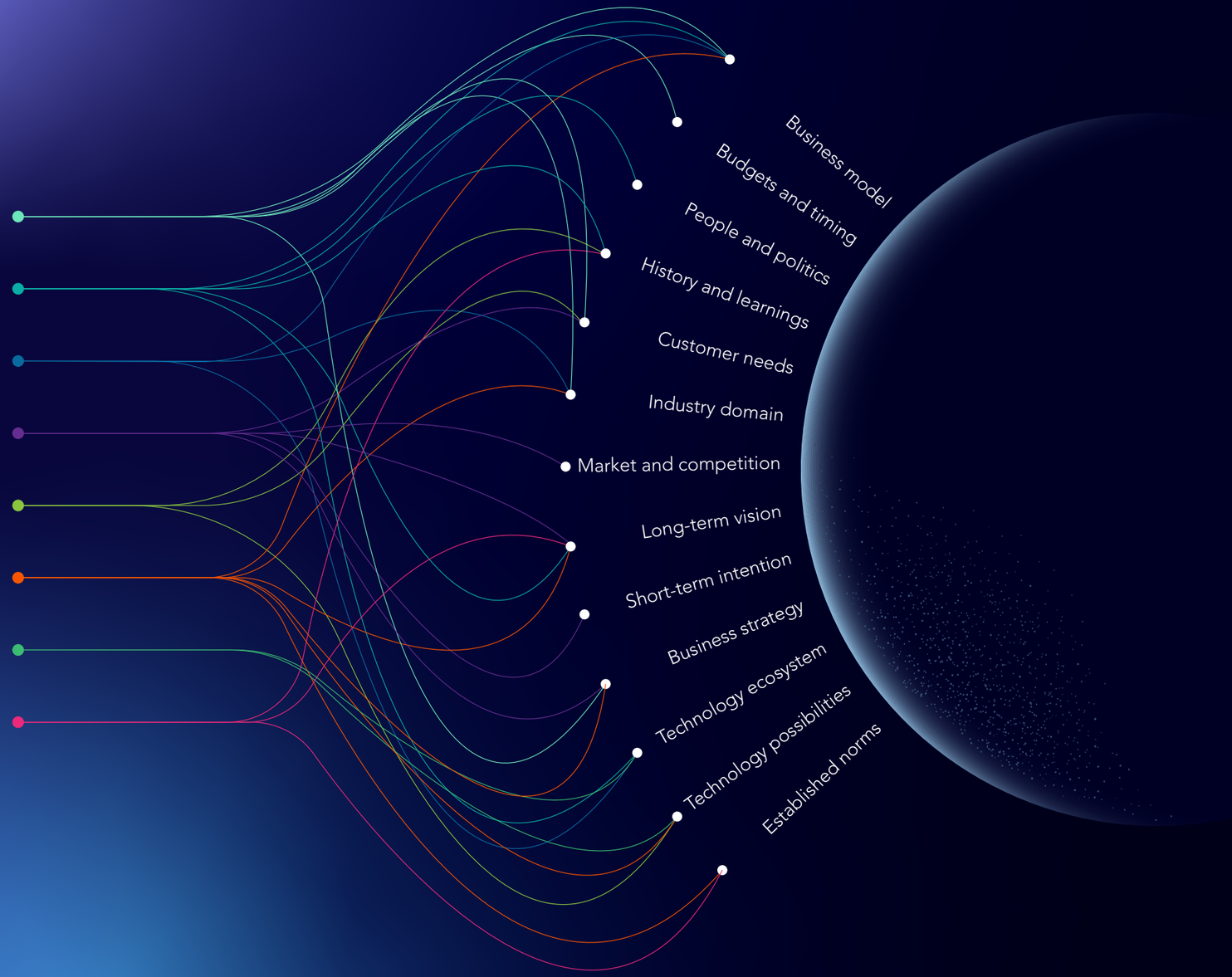
TECHNOLOGY POSSIBILITIES

What new technology could radically change how we solve this problem?

ESTABLISHED NORMS

Can we take advantage of practiced ways of work, or are we also on a mission to change operating procedures? Should this be done at the same time?

CONTENTS



CHAPTER 1

BUSINESS MODELS

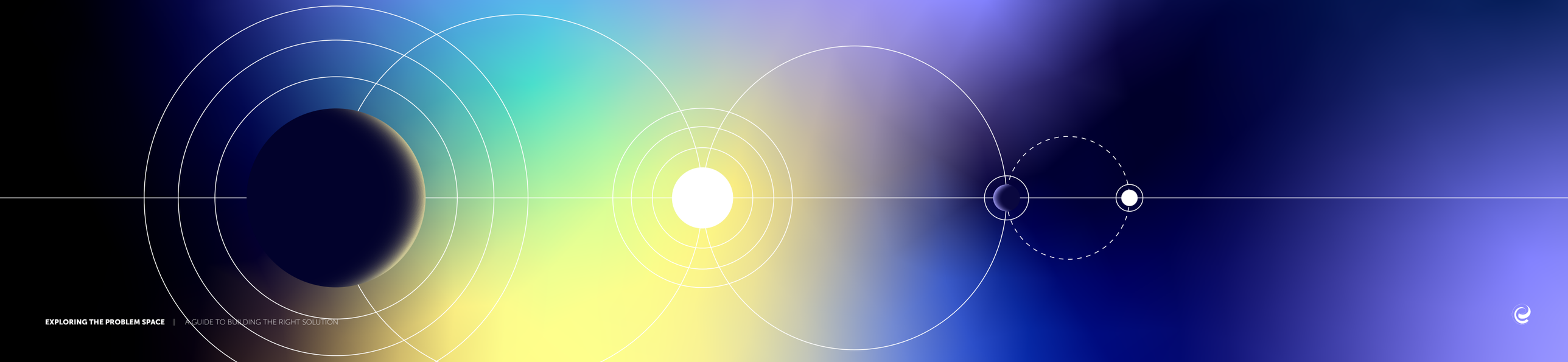
EVERY BUSINESS MAKES AND SPENDS MONEY DIFFERENTLY.

Understanding the fundamental model upon which the business operates is imperative to form a basis for exploring how digital channels and products or services can be leveraged.

Some businesses generate income based on selling services where a profit margin is calculated based on resource cost for providing the service such as personal banking. Others might operate on future risk and opportunity such as personal insurance, and investment-based businesses. It's worthwhile understanding that most established corporations have a multitude of offerings, and many employees and divisions that usually work together towards a shared organisational vision, but each generating revenue in a different way.

However, many modern start-ups have disrupted components of traditional corporations by focusing on a specific product or service and finding alternative or better ways to offer it. Companies like Pineapple, Monzo, Uber, and Airbnb are just a few that exploited a specific service offering in a way that's more aligned with how consumers have evolved.

When embarking on creating a digital offering, we must ground ourselves in the model of how the existing business operates, bring in inspiration from competitors and similar offerings, and formulate a fundamental model that the digital offering will be based on.



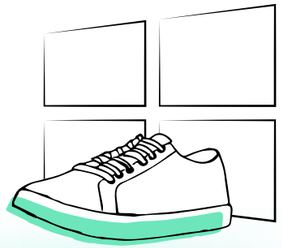
WHAT IS BEING SOLD?

START WITH FIRST-PRINCIPLES. IDENTIFY EXACTLY WHAT IS BEING SOLD, AND WHY THAT DECISION WAS MADE. WHAT VALUE DOES THE PRODUCT OR SERVICE PROVIDE? AND HOW DOES THE TRANSACTION HAPPEN?

A ONCE-OFF PRODUCT

An example of a once-off product sale is selling a pair of sneakers. There is a cost associated with the production and distribution of the sneakers, and the profit is usually a margin on top of the total cost price.

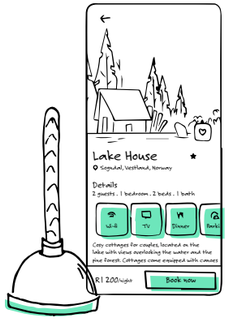
In the digital world a product feels like an object but is not tangible. An example of a digital product is Microsoft's 365 Office software. It's bought as a product that enables the users' productivity. A video game is similar in that you buy the product – however, more and more game publishers are moving towards recurring revenue models.



A ONCE-OFF SERVICE

An example of a once-off service is a plumber coming in to fix a problem or install something new. The plumber offers a service that requires human-hours and specific resources and tools to fulfil.

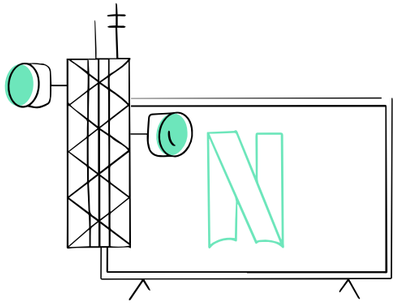
In the digital world, a service can be provided with incurring little or no human-hours and little resource costs. An example of a once-off digital service is the booking of vacation accommodation online. The platform provides a once-off interaction with the consumer, but maintains a longer-term relationship with the renter.



A CONTINUOUS SERVICE

An example of a continuous service is providing internet connectivity to consumers. The service provider is responsible for running and maintaining the infrastructure to support access to the internet.

In the digital world, this is by far the most widely used approach. The idea is to provide value to users via digital offerings. These offerings are usually readily available, cost little to keep running, and require little to no human-labour to fulfil. An example of a continuous service is Netflix. Their business provides continuous on-demand video content to consumers for a continuous monthly fee.



There are other revenue generation models such as licencing and advertising, among others. In large established organisations, there are usually a number of approaches leveraged that interconnect with each other. Understanding the details behind the approach and what it's related to within the organisation creates a solid foundation to start finding problems to solve or generate ideas for innovation.

WHO ARE THE CUSTOMERS?

WHEN WE TALK ABOUT THE PRODUCT OR SERVICE BEING SOLD, **IT'S IMPOSSIBLE TO NOT TALK ABOUT THE INTENDED CUSTOMERS FOR IT.**

Customers come in all shapes and sizes based on what's being sold. Furthermore, defining the market segments within the category of customers is important to ensure that the right thing is being offered to the right people.

END-CONSUMERS

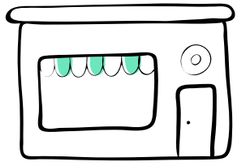
These are generally the public world at-large. Consumers are people like you and me that have needs to satisfy or goals to achieve. The digital product should address these needs or help the person achieve a goal.

For example, Spotify is a music streaming service that's provided directly to consumers. Their potential market is huge since a large portion of the world enjoy listening to music and have access to smart devices.

BUSINESSES

These are other businesses that have a requirement that can be fulfilled by a product or service offering. These offerings might speak to the business's internal processes and help it perform more efficiently and effectively. Or it might be an offering that the business leverages for their own offering to consumers or other businesses.

For example, Salesforce provides tools for customer relationship management, enabling sales, customer service, and more. Their potential market is most businesses since these are fundamental problems and processes.



BOTH END-CONSUMERS AND BUSINESSES

Many businesses provide offerings for both individuals and businesses, but one is usually the primary customer.

For example, Google provides consumers with search capabilities among other things, but also offer business services to interact with consumers and advertise to consumers – making businesses their biggest customers. They also have services aimed at consumers as customers such as cloud file storage and email.

UNDERSTANDING THE COSTS INVOLVED IN PROVIDING A SERVICE OR SERVING A PRODUCT IS IMPORTANT TO KNOW THE VALUE PROPOSITION FOR THE BUSINESS AND ITS CUSTOMERS.
TRADITIONALLY, A PRICE WAS CALCULATED BASED ON THE COST. COST + A MARK-UP = SELLING PRICE.

However, more and more products and services are priced based on value and perceived value, with cost not being a driving factor. With that said, it's important to understand all the expenses at play to make something happen – these drive decision making and inform if something is worth pursuing or not. It also drives the creation of targets and to meet expectations.

COSTS

In a digital world, these are the popular costs that come into play.

Engineering
The cost of development, continuous enhancement, and maintenance of a digital solution.

Infrastructure
The cost of hosting digital solutions and making them available to consumers.

Support
The cost of support staff or support tooling to serve existing customers.

Sales
The cost of acquiring new customers.

PRICING

Pricing strategies are interesting and diverse. Different pricing approaches can be applied to the same product or service and achieve varying levels of success.

Monthly / yearly subscription
Digital products usually have a monthly or yearly price attached to them. Buyers like certainty in fixed pricing, however, many digital solutions provide ways to choose the services required which adjusts the subscription costs.

Usage-based billing
Another popular mechanism for pricing is to base it on usage. This approach provides transparency and seems more fair to the buyer. However, this approach can also cause anxiety due to the uncertainty of incurring unexpected costs.

Once-off cost
Although many digital services and products are provided with a once-off cost, the trend is moving towards recurring revenue with smaller margins to retain customers over the long-term. With that said, once-off costs are attractive to some customers because it removes any long-term commitment.

Cross sales
The goal of cross sales is sometimes the driving factor in features of digital platforms. The intention is to captivate an existing audience and convert them into customers of other products or services within the businesses ecosystem.

Indirect value
Within large established organisations, this is often the case with many features and digital solutions built. There are clear intentions for the single solution, however, it adds to the sum of the parts to achieve different goals. This includes gaining larger user bases, learning more about customers and users, providing more ubiquitous interactions – driving usage and revenue, and more.

CHAPTER 2

MAPPING TOWARDS A VISION

ANY NEW ENDEAVOUR SHOULD WORK TOWARDS A VISION, WHETHER IT INVOLVES A DIGITAL PRODUCT OR NOT.

A wicked problem is one where there are a plethora of ways to work towards the vision with multiple parties, people, systems, processes, and interactions that may impact or be impacted by the solution.

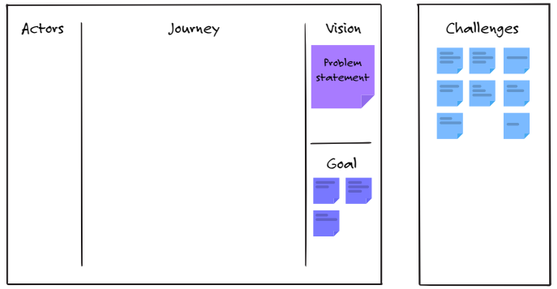
It is important to gain an understanding of how functions in the business operate through existing channels, and empathise with end-users, then determine what needs to be adjusted to achieve the vision for the initiative.

Mapping is a powerful tool that can be harnessed to solve complex problems. It is designed to include people from all levels of the organisation that may impact or be impacted by the initiative. It provides mechanisms to describe the vision clearly, determine the actors that are part of the wicked problem and solution, and plot as-is, and to-be processes for the solution.

Mapping creates a shared brain between people of different roles and expertise. It also creates a democratic way of creating hot spots of focus, while allowing decision makers to decide on what to prioritise by leveraging the input from the group. Although mapping generates rich outputs for next steps, the shared understanding and conversation from participation in the workshop itself is powerful in creating alignment and surfacing ideas that otherwise might be lost.

STATING THE VISION, & CREATING ALIGNMENT

THE FIRST STEP IN MAPPING IS **DEFINING THE PROBLEM STATEMENT**.



This can be an ambitious vision phrased as a problem or a very specific problem that needs to be addressed immediately. Stating the problem statement as a vision in a simple sentence is useful to use as a reference point whenever you ask "what are we trying to achieve again?"

Example of problem statements:

We want flexibility in how we sell our product by optimising our complex sales planning process and systems.

We want to improve our market share by equipping our staff with actionable insights about our customers.

We want to provide a differentiated customer centric experience for our ride sharing customers.

We want to track and trace employees of our farms to ensure the safety of the entire workforce.

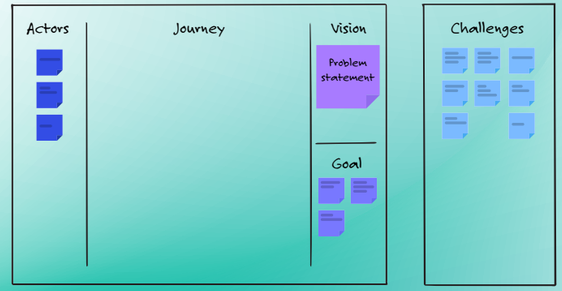
Mapping always starts at the end with the greater vision as a north-star, and everything to achieve it is discovered throughout the rest of the process.

The next step in mapping is identifying and involving the right people to participate. Generally, for big vision ideas or wicked problems, involving people from varying roles and experience levels is hugely beneficial to gain different perspectives. When the problem being solved is narrower, subject matter experts in the problem being solved are useful in surfacing the hidden details and nuances to the group.

The actual mapping processing begins with ensuring that everyone is aligned with the greater problem that is being solved. A good technique to do this is to explain the problem statement, and the reason why it is an important challenge to tackle; then, prompt the group for the challenges that they foresee in achieving it. Each challenge is noted on the map individually. This activity creates a shared understanding of the vision and goals, and uncovers the hurdles that each person might have from their vantage point in the business.

LISTING THE PEOPLE, ORGANISATIONS, & SYSTEMS INVOLVED

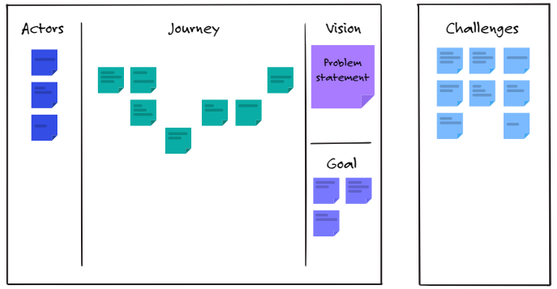
NOW THAT THE GROUP IS RALLIED BEHIND THE SAME GOAL, **IT'S IMPORTANT TO UNDERSTAND THE DIFFERENT ACTORS INVOLVED IN ACHIEVING THE VISION AND GOALS**.



These actors may be individual people, business functions, departments, partner organisations, and technology systems that support existing processes and functions. The actors are sourced similarly to the challenges. Each person in the group should suggest the actors that they think are part of solving the problem. By crowdsourcing the actors, each person in the group is also exposed to actors that they might not have been aware of, or gain a deeper understanding of those actors and how others experience them.

UNCOVERING THE AS-IS JOURNEY

THE AS-IS JOURNEY IS IMPORTANT TO MAP TO **CREATE A SHARED UNDERSTANDING OF EXISTING PROCESSES**.



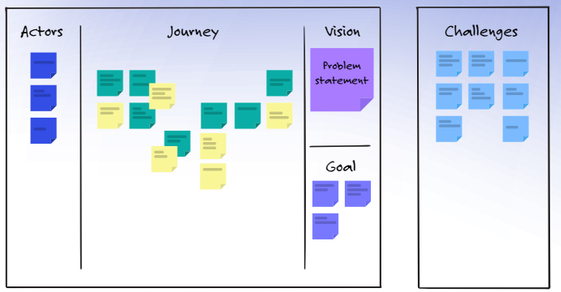
The map now consists of well understood goals, and the different actors involved. Although this is useful knowledge, the "as-is" journey is important to map among the group. This is the existing journey in achieving whatever goal we have listed.

Since there are a number of different actors, the focus should be on one use case for one specific actor. This creates baseline alignment across all stakeholders and often allows people to see the process through other people's eyes, often exposing details they weren't aware of.

An example journey could be: "The journey of a financial advisor for acquiring new customers", or "The journey of a consumer when booking and using a ride sharing service".

PROJECTING A TO-BE JOURNEY

THE JOURNEY THAT WILL EASILY **ACHIEVE ALL ACTORS' GOALS**.



With a strong understanding of the challenges, actors, and as-is journey, ideas for solutions can be generated. Throughout the previous interactions, all the participants would have questions and ideas that come to mind based on their unique perspective of the "shared-brain". These can be phrased as "How might we's". The premise is that we phrase problems as questions to address. For example, "how might we make the checkout process faster", or "how might we provide all important information to our call centre agents". These ideas and questions are placed on the map relative to the area that they are related to.

THE MAP NOW CONTAINS A SHARED BRAIN ENCOMPASSING THE EXPERIENCE AND EXPERTISE FROM VARIOUS PEOPLE.

It contains a collection of shared challenges, an exhaustive list of actors involved, and a clear outline of the existing processes and journeys. The map also contains solutions to problems that will inform what a solution could look like. This artefact is invaluable to teams trying to solve wicked problems and innovate in their digital product and service offerings.

Finally, the experience of creating this map with the different stakeholders involved creates empathy, shared understanding, and alignment between everyone. New relationships and lines of communication are formed, and this ultimately results in improvement in the existing business operations.

CHAPTER 3

UNDERSTANDING EXISTING PROCESSES

USING EVENT STORMING

THE GOAL IS TO NOT JUST SOLVE PROBLEMS WITHIN THE CONSTRAINTS OF DEPARTMENTS, BUT RATHER TO **TAKE A HOLISTIC APPROACH**.

Event storming is a technique that is used to cross the boundaries between multiple departments within an organisation and bring people from various disciplines together to define and map out the things they do that add value to their customer. This technique's sole objective is to bring people from business, UX, development, marketing and other departments together to understand what everyone is doing on a day-to-day basis.

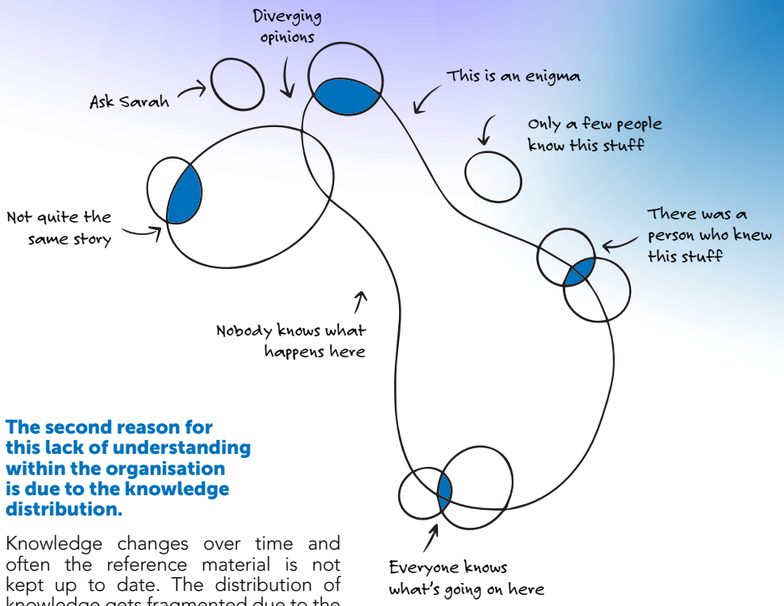
It is not constrained by any system boundaries and enables people to use a simple ubiquitous language to describe events within their specific business domain. Once the full picture is on a large canvas, the teams can collaborate around pain points or hand over points between departments to solve problems which at first appear to be seemingly complex in nature.

WHY USE EVENT STORMING

One of the principles of agile development is that teams self-organise and find the best way to solve problems, since they have the most context and understanding of what needs to happen. People will not self-organise around a system they do not understand. In large organisations there are very few people who know and fully understand the end-to-end value chain that a customer moves through when dealing with the organisation. There are common reasons for this with the first being how organisations are structured.

It is very easy for silos to form between different departments or business units within the organisation, with a lack of focus on integration and working together.

Ultimately the customer is sent from pillar to post, and is left with a feeling of frustration. From the customer's perspective, it is still one company and they expect to receive a unified service, regardless of which department they interact with.



The second reason for this lack of understanding within the organisation is due to the knowledge distribution.

Knowledge changes over time and often the reference material is not kept up to date. The distribution of knowledge gets fragmented due to the silos highlighted above, and political circumstances in the business can lead to people creating knowledge empires in their own departments instead of distributing it to the larger organisation.

Churn in employees is a big risk to any organisation as they take years of knowledge and experience with them that you cannot easily replace by just hiring a new employee. Key person dependencies can compound the knowledge distribution risk within the organisation, so making an effort to create workshops like event storming is a good way to share that knowledge with more employees which lowers the risk of key man dependencies.

HOW TO USE EVENT STORMING

SET THE SCENE
Firstly, identify a few key problems or business processes that need solving or improvement. It is essential to identify stakeholders from every area in the business that are involved as well as ensure that people working at different levels are involved. It is not beneficial to host this just as an executive workshop or just as the IT department because you will not get the full picture and fall into the trap of knowledge distribution fragmentation. This can easily lead to an audience of 25-50 people depending on the size of the organisation. So make sure you have a big enough room for everyone to fit.

GET SUPPLIES
You need to cater for some stationery if the workshop will be in person or use an online virtual whiteboard application. Sticky notes of varying colours are required as well as markers to write on them.

ASSEMBLE A TEAM
When the workshop starts, ensure that you have 2-4 facilitators available depending on the size of the group. Then take some time to explain the agenda for the workshop.

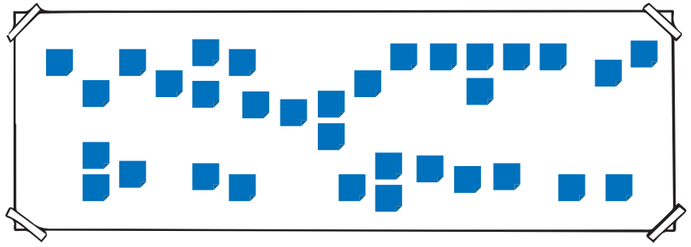
An example of a workshop agenda:
Today we have a canvas depicting a timeline from left to right of how our customer moves through our organisation's value chain.

We will have 3 rounds to map out the process and find some areas where we need to work on improvement.

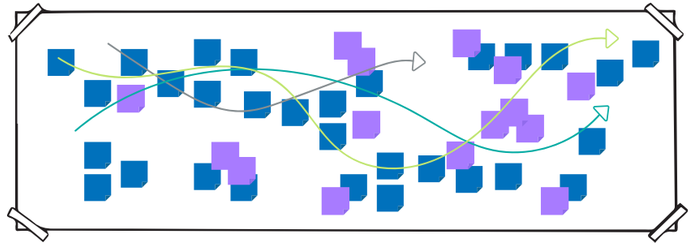
ROUND 1: SET THE SCENE
We will use blue stickies to represent domain events. You will create sequential steps within your department groups and add them to the timeline. Domain events are anything and everything you do to add value to the customer. It is not just executing steps on a software application. Every phone call, email or task happening outside of the system is also a domain event.

This is where the additional facilitators are important as people will struggle to get going with their initial domain event definitions. Helping them with some suggestions and just asking them to talk you through their day is a good way to quickly identify domain events.

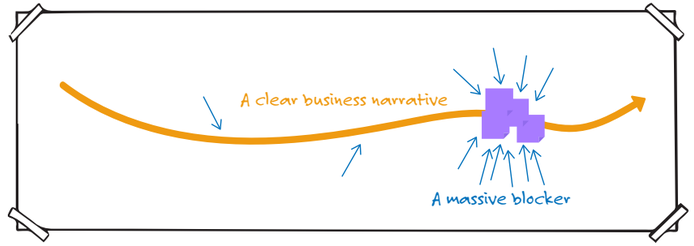
The facilitators are also responsible to ensure that groups of people are not influenced or dominated by the strongest personality in the room. Often, we see teams just going with what their manager says should be happening instead of putting down the events which actually happen. There can be no progress unless all participants feel like they are in a safe space and all participants are working towards the bigger improvement in the organisation.



ROUND 2: PAIN POINTS
We will use purple stickies to add pain points or potential enhancements to areas in the process. Anyone can add pain points anywhere on the canvas. You will start to see where issues occur in the hand over points between departments and what people are struggling with in their own departments. It will also require all participants to review all the other departments and start asking questions.



ROUND 3: FINDING FOCUS AREAS
We use arrow voting to pick the areas where we need to focus first. This will highlight the biggest blockers in the process and where most of the effort should be going in to. A workshop like this can take between 3-4 hours depending on the size of the audience and complexity of the problem. The common understanding it creates is definitely worth the time spent on this.



OUTCOMES & NEXT STEPS

The outcome of the workshops should be to have the whole process visible and really see the big picture. There will be massive learnings for the workshop attendees as they get insight into other departments and really cross the silo boundaries in the organisation. Finally there should be consensus around some core issues identified through the arrow voting.

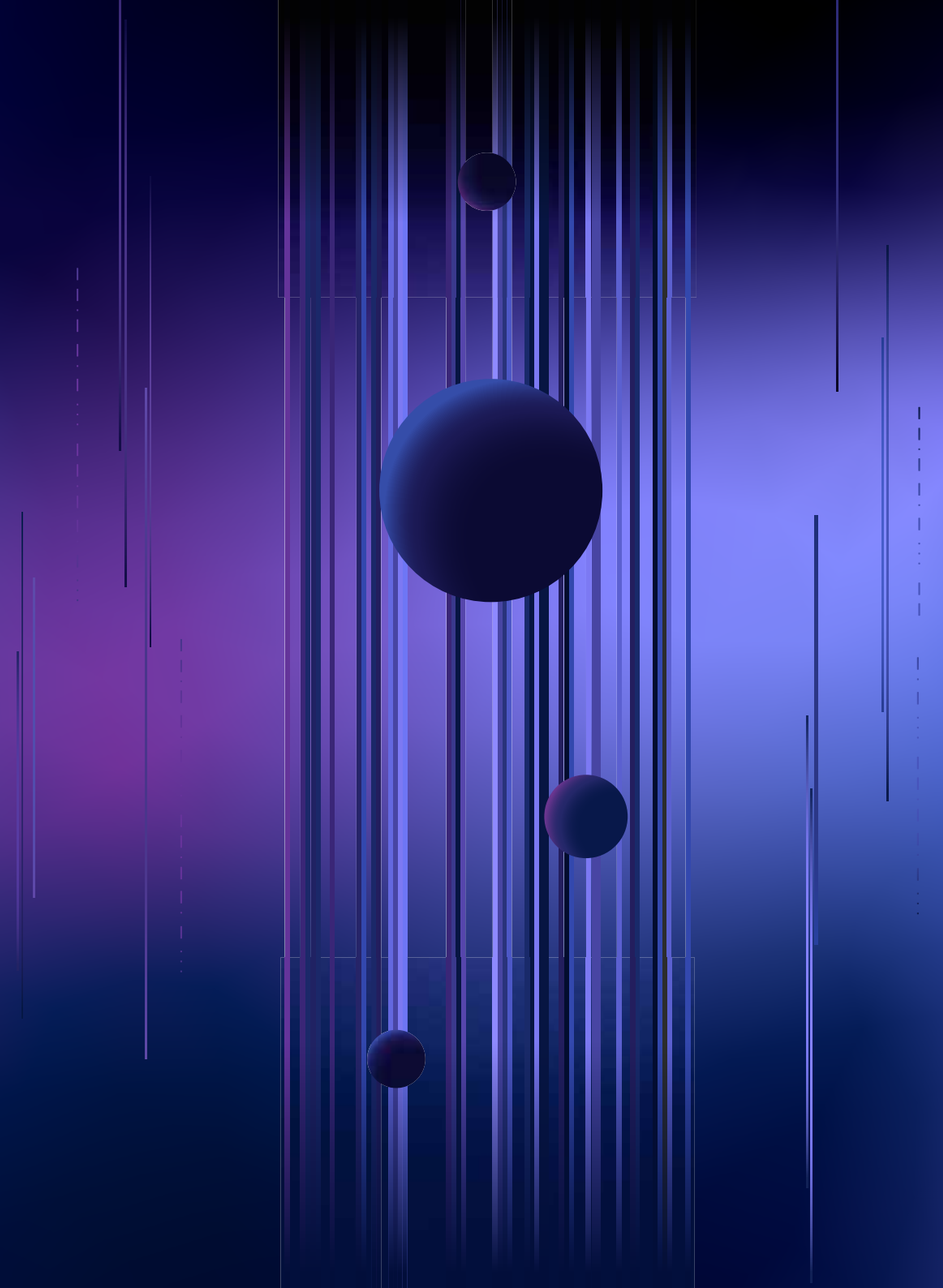
Some of the hidden benefits of the outcome is that you will see different stories evolving from the sequence of events and get different perspectives on the same value chain that a customer moves through.

This technique creates a simple language for all the different tribes of people in the organisation. There is no use of tech jargon or traditional modelling techniques like BPMN or UML and UX specific techniques. The process is mapped using simple English and will be the ubiquitous language that departments need. You will clear up the definitions of concepts as something as simple as "what is a customer?", which can mean different things in different departments.

The next steps are to figure out the highest priority or hardest problems to solve. It is up to the management team to set the backlogs for the different departments to align to the highest priority problem. In essence, the backlog for different teams take a back seat if the highest priority item needs to be worked on. The hardest problem to solve also requires a shift in thinking from having to estimate how long it will take to allowing the teams to experiment for a while and then estimate. By definition these problems are hard because no one has been able to solve them, so asking how long it will take will lead to a lack of innovation due to self-imposed time pressure.

CHAPTER 4

KNOW YOUR USER



FIRST-HAND PERSPECTIVES, BEAT SECOND-HAND GUESSES

In an increasingly competitive market, technology is less and less a differentiator, and experience becomes paramount, so it is crucial to understand your users. Once that understanding is in place through robust research, you need to use your insights to develop empathy that guides your team and product decisions.

Distilling your research and making it come to life is how we create actors and personas. Both tools allow one to understand how people and systems engage with each other. This allows you to analyse incisive points where your product can make a difference.

The empathy generated through these tools creates a shared understanding of customers and systems. This understanding is imperative, as your team needs to know what problems they are solving, and for whom. We now have a unified front where tech, business and user representatives can work together, embedded in a team of problem solvers, not siloed with isolated views of the problem space.

DESIGNING WITH EMPATHY

WHEN IT COMES TO UNDERSTANDING PROBLEMS WE ARE SOLVING, WE NEED TO UNDERSTAND **WHO WE ARE SOLUTIONING FOR**.

There are a few tools we can use to help us understand the *who* behind a problem, which will ultimately help us understand the *why* too. To achieve this, we use actors and personas. A trap one can easily fall into is using actors or personas interchangeably, however, they serve different purposes. To understand these two tools, we'll take a deep dive into each.

ACTORS

An actor in a film plays a role and performs actions that affect the story. In the same way an actor within the world of product development has a role and engages with a system or process – ultimately effecting change.

An actor might be a user or even a system. As an example, an actor might be representative of a customer on an e-commerce platform, who browses products and completes a checkout process.

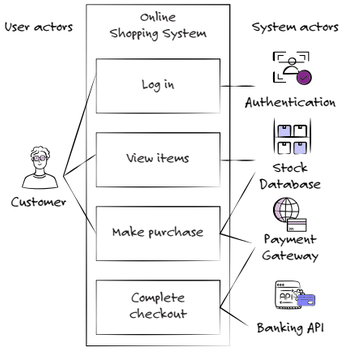
An actor can be defined as an initiator of an action or the recipient of an outcome.

We will often find that an actor has a short descriptor or name. This description does not delve into the detail of the person. Example names for an actor can be the customer, client, stakeholder, employer, or employee.

ACTORS IN ACTION

To understand the problem we are solving for, we need to understand the current or *as-is* flow of the product that users or systems interact with. These interactions can be defined as a use case. A use case diagram can be used to visualise the flow of a system and the necessary actors involved.

Illustrated below is an example of all relevant actors involved in a standard checkout process. Some of the actors are involved in a single use case shared between them, whereas others have use cases that only apply their unique action.



While the details of an actor are high-level, a persona offers more granular details, such as demographics and psychographics. Both personas and actors are effective ways to help us place ourselves in the users' shoes to understand their mindset, frustrations and goals.

PERSONAS

When it comes to product design and development, the user is always at the heart of the solution.

While we need to cater for business needs, it is important to ensure our users' needs are met too, as they will determine the overall adoption of the solution.

The only way we can achieve this is to know the target audience and understand their goals, pain points, motivators, and characteristics in more detail. This is where personas come in.

"EACH PERSONA REPRESENTS A SIGNIFICANT PORTION OF PEOPLE IN THE REAL WORLD AND ENABLES THE DESIGNER TO FOCUS ON A MANAGEABLE AND MEMORABLE CAST OF CHARACTERS, INSTEAD OF FOCUSING ON THOUSANDS OF INDIVIDUALS"

— MO GOLTZ

WHAT IS A PERSONA

A persona is a fictional user with goals, friction points and characteristics created to represent user groups that will interact with a product.

To make a persona effective for its intended purpose, context-specific details are included to tie user behaviour patterns back to the problem space one is solutioning for. Personas also help to prioritise certain features according to existing user needs.

HOW TO CREATE A PERSONA

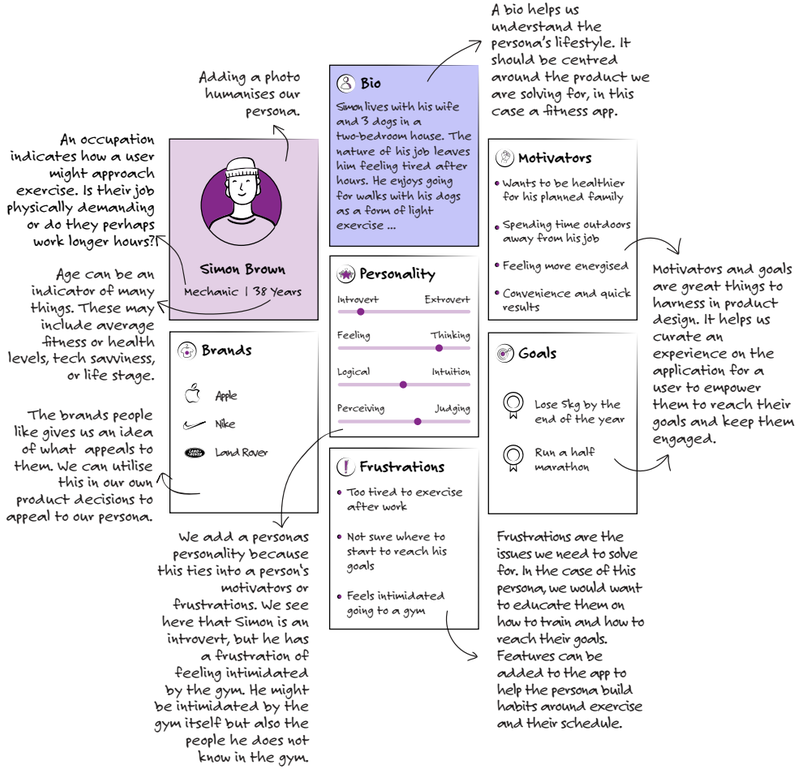
Personas stem directly from customer research and should not be created from a place of assumption or bias. As research is conducted into one's user base, themes often emerge in the user's mindset and behaviour. These themes can then be broken down into user patterns and user mindsets, which will help inform the creation of personas.

While fictional attributes may be added at times to give more substance to the persona and make it more relatable, all elements are still derived from the real data captured during customer research.

HOW TO CREATE EMPATHY THROUGH A PERSONA

As an example in research for a fitness app:
We've discovered a persona that has a goal of running long distance. This research includes how they exercise, their motivators and current frustrations

By understanding all of this we can design a solution that empowers to overcome barriers and reach their personal goals.



WHY DO WE CREATE PERSONAS

The short answer is empathy.

Using empathy to create personas humanises users and creates a clearer picture of who we are solutioning for.

Personas are also an excellent analytical tool. When we don't have access to our users, we can refer back to our personas to analyse their motivators, goals and pain points. Personas can be used to build content strategies or conduct a needs analysis.

Personas are unfortunately often created and forgotten as a project develops so it's important to bring personas to life and keep them top of mind. Adding a name and surname with a photo to a persona,makes them more real and relatable. Personas can be printed out and stuck on walls or used in workshops.

Ultimately, the persona is an important building block in the design process which can be used to inform other design artefacts and inform the overall solution.

Understanding and empathising with a persona's behavioural patterns puts the user at the forefront of the solution, ensuring a well-rounded product catering to both business and user needs.

A persona is a means to an end, and a more empathetic one at that.

USER EXPERIENCE JOURNEY

ENSURING A SEAMLESS AND USER FRIENDLY PRODUCT FROM END-TO-END.

A user journey map is a diagram that indicates all the touchpoints of a user working towards a goal or task, while using a product or service. Journey maps have many layers – they describe a user's behaviour and the emotions the user might feel, as well as how the system or service responds to input or action.

It is a key tool used to communicate to all who are involved in the process, and they are most importantly used for decision-making when it comes to the priority of features being built, and ensuring the journey is user friendly.

THE 6 CRITICAL PARTS TO A USER JOURNEY MAP

Actor
The user of your product or service. The journey map shows all actions for this actor.

Scenario
This is the path of the journey. It could include your current solution or a proposed solution.

Actions
Consecutive behaviours that describe what the user is doing, step by step, when using a product or service.

Emotions/mindset
What the user feels or thinks at each step of the journey.

Touchpoints
The interactions a user will have with a product or service.

Opportunities
The insights gained from the mapping process that can improve the user's journey.

With the respective parts of a user journey in place, we can see the bigger picture. This allows one to discover and solve pain points, and add moments of delight to increase customer engagement and satisfaction.

WHEN SHOULD YOU CREATE A USER JOURNEY MAP

User journey maps are useful when you want to validate the feasibility or assumptions you might have of your solution.

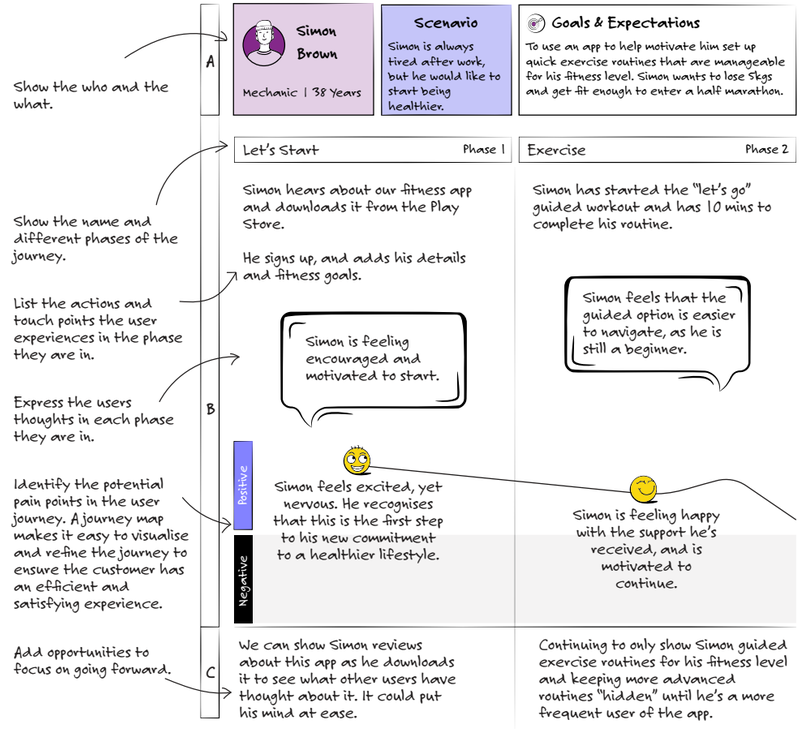
Mapping out the journey is the most effective way to uncover pain points and streamline the users' flow through a product or service.

User journey mapping is a flexible tool. You can create a journey to analyse an existing product or service (as-is), or to map out a new product (to-be).

Creating a map from assumptions will bring no value when trying to solve a problem. You need to have researched and observed your customer to understand their journey before mapping. Research can be done in many ways:

- Send out surveys with questions regarding the processes your users follow
- Run contextual enquiries to observe users and seeing things from their perspective to gain the necessary empathy
- Conduct user interviews to gain further insights

By using personas and the information from data you have obtained through the research, you can then accurately depict what your user does and how they act while using your product or service. Adding this strategic tool to your arsenal, will allow you to analyse what the current state of use is, and move towards an ideal or future state to differentiate you from your competitors.



SUMMING IT ALL UP

As we've mentioned above, the key purpose of creating a user journey map is to create or improve experiences for the user at each touchpoint. What a user journey looks like is less important than what it says about your user.

Don't get bogged down in aesthetics. Remember your why – to map your product or journey to enhance your users' experience, and in turn, improve your bottom line and customer sentiment.

CHAPTER 5

SCALING OMNICHANNEL DESIGN THE RIGHT WAY

NOT TOO LONG AGO CUSTOMERS WERE ONLY ABLE TO ENGAGE WITH A BUSINESS IN A FEW WAYS – IN STORE, TELEPHONICALLY OR THROUGH A LIMITED WEB EXPERIENCE.

Today, the range of technologies have increased to include smartphones, tablets, smart watches and IoT devices, along with ever-expanding digital channels such as social media, chatbots and apps.

This has turned a relatively small customer experience (CX) ecosystem, that was relatively easy to maintain into something that is diverse and at times divergent. How do companies maintain consistency in experience throughout all these systems, taking advantage of omnichannel design but still driving convergence, towards consistent messaging and experiences?

User-centered design methodology worked neatly with desktop or single channel UX but now there has been an exponential rise in complexity due to multi-device ownership.

A single user now has multiple touchpoints and phases in a journey, which require a solid omni-channel digital strategy. A user may start their journey on desktop, leave their home or office, and continue on their phone where they make a purchase and track delivery on their smart watch through alerts, finally answering a call on their phone to receive the purchase, then return to their desktop to post a review or return the product.

CUSTOMERS GO WHERE YOU LET THEM

Customer journey starting and ending channels

A user will simply go wherever they are able, across your channels to achieve their goal using the path of least resistance. This means that you need the following across channels:

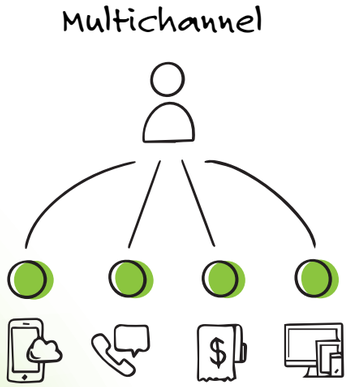
- Seamless integration (backend - API, microservices, cloud)
- Consistency and standards (UI, UX)
- Robust and scalable design system (regardless of channel)
- Strategic selection of relevant channels (do you need an app, chatbot etc.?)
- Robust research to inform channel decisions (validate which channels are appropriate and how to design for them)

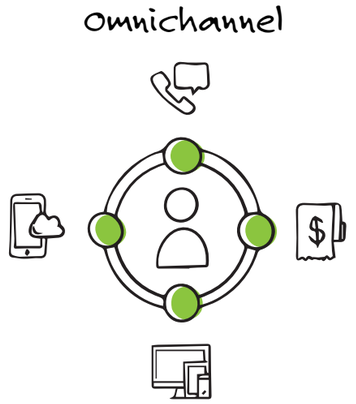
MULTICHANNEL VS OMNICHANNEL

WHAT IS THE DIFFERENCE?

While your business may operate in divisions and your teams may work on different touchpoints or channels within your business, your customer simply isn't concerned or aware of those structures. They experience all channels as a single customer experience. If you provide excellent CX on a single channel but an awful or unintegrated experience on another channel, the user sees this as a failing of the business as a whole.

While a multichannel experience is one that has the user engaging with various but unintegrated touch points, a true omnichannel experience means a user can successfully fulfil a single journey or task moving between various channels.





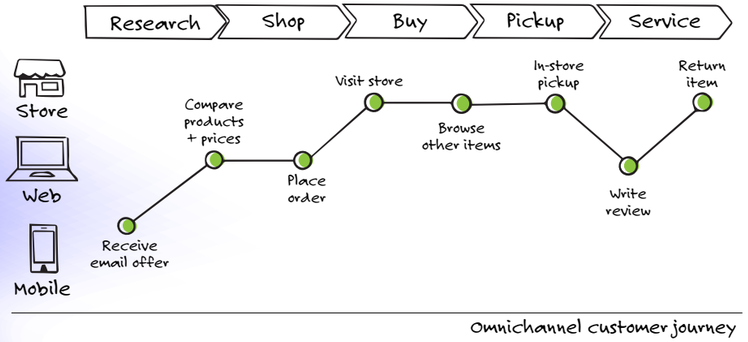
WHAT A TRUE OMNICHANNEL EXPERIENCE PROVIDES

Channel choice
A user may choose to switch between any channel at will, without the risk of losing continuity on their customer journey or a negative experience.

Access to up-to-date and consistent data
A customer's data needs to be consistent and shared across channels.

Contextual CX and UX
A user's experience of a particular channel should be specific to the context in which they're engaging, i.e. a native application should follow best practice for apps on their particular device and not simply replicate a web experience.

Product or service synchronicity across channels
All channels should provide consistent in product catalogue or service offering barring those that are particular to certain devices, i.e. you may have limited features on a smartwatch but those should be directly suited to the device and strategic.



Omnichannel customer journey

A SINGLE TASK ACROSS VARIOUS CHANNELS

IT'S ONE EXPERIENCE FOR YOUR USER

Today's dynamic and interconnected experience means that a user may engage with your business across a range of devices and channels to complete a single task, where customers expect a seamless experience. While transitioning from single or multichannel business towards truly omnichannel it's vital to understand your current landscape.

UNDERSTANDING YOUR BUSINESS AND SELECTING APPROPRIATE CHANNELS

Customer journey
How your users engage with your product or service to achieve a desired task through user research, surveys, and analytics.

Retention rate and drop-off rates
Where do you succeed with your customers and where are your gaps?

App or software flows
Assess the flows of your various systems – how can you integrate your various services and integrate your various data points to achieve seamless integration?

Monitor all your channels
Gain deep insights into each – look at performance, conversions, crashes, APIs, business systems or microservices and clouds.

Get user data
Learn how customers are using application/s or channels in the real world, is app intuitive, common activities and functionality. Find out if your app is intuitive by gaining analytics on common activities and the relative success of functionality across your channel platforms.

Business insight
Gain business insight through analytics tracking the performance of your platform against chosen measures and benchmarks. Retrieve, store and analyse comprehensive details of user actions across channels and the efficiency of the back-end integration of each, as your front-end interaction needs to remain seamless for the user.

By focusing on your customers' journey and analysing analytics provided through monitoring of all channels, you will be able to establish user data that allows for continued business insight. Some useful questions to ask along your omnichannel journey are as follows (it is key to the success of your channel ecosystem that you can answer them):

1. Where are your users? Their devices and preferred channels.
2. Who are your users? User data to create robust personas.
3. What do they need? Relevant channels, devices, products, and services.
4. When do they need you? Points in their customer journey that require engagement.

This will dictate which channels are appropriate for your business, as well as the feasibility of integrating those channels.

CROSS FUNCTIONAL TEAMS

RALLYING AROUND CUSTOMER JOURNEYS INSTEAD OF SEPARATION ACROSS INDIVIDUAL CHANNELS

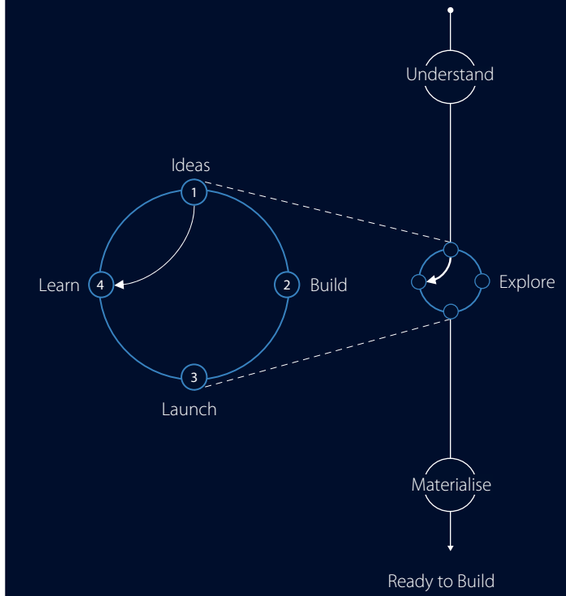
For a business to achieve true omni-channel CX, it's important to step outside of channel-specific thinking and work towards cross-functional teams that structure themselves in terms of customer journeys. A way to start this sort of thinking is through multidisciplinary Design Sprints where various stakeholders are brought together from all areas and levels of the business to solve a wicked problem.

Included in this Design Sprint should be a thorough mapping of the business and customer journey, look at all respective touchpoints, data sources and business rules. The goal here should be to find solutions and understand limitations of true integration.

Innovate, and solve wicked problems

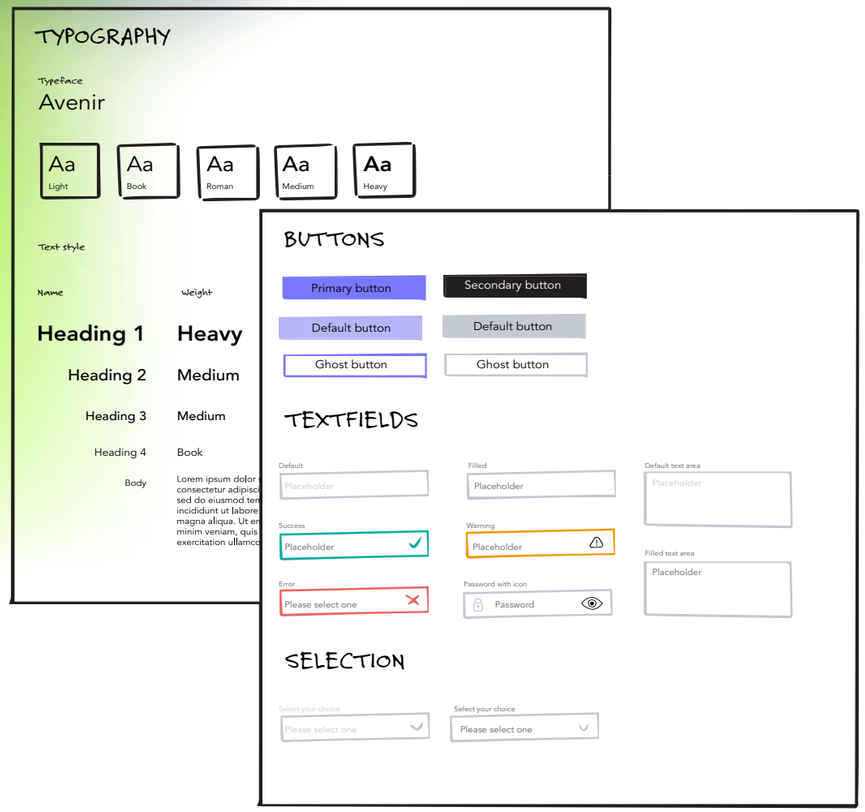
Entelect has reinvented Google Venture's Design Sprints for **solving complex problems effectively and innovating in enterprise environments**.

Design Sprints are useful when problems have no one clear solution, and involve many moving parts in terms of people, teams, systems, and business objectives.



Contact us to innovate and solve your wicked problems.

solutions@entelect.co.za



VALIDATE DECISIONS

DESIGN SYSTEMS AND RAPID PROTOTYPING

Once you've gone through the process of channel selection, you've put together a plan to integrate your various services and have begun structuring cross-functional teams around your customer journey/s it's time to prototype and validate decisions.

You'll need to start off by defining a robust and adaptable UI that caters for various channels and is truly responsive. This design system should define your:

- Branding
- Colours
- Typography
- Form fields and controls
- Data visualisation
- Responsive frameworks
- Channel or device specific design (particular to say a smartwatch or chatbot)

PROTOTYPING AND TESTING

Once you have your design system and components, you'll need to get prototyping. At the early stage of your omnichannel journey, rapid prototyping will suit your purpose.

The aim is to test the hypothesis and fail early, because it's significantly cheaper and more efficient to make potential errors and test in prototyping than in a live solution.

Ensure your prototypes are designed for, and presented to your users on devices that are relevant to your chosen channels. For example, if you're designing for an instore kiosk, use a tablet and emulate the kiosk when testing with users and also test in the context of use, setting up a temporary installation or environment in which the user can truly understand the experience of this channel. You can also test across channels by having various test devices to simulate an omnichannel experience.

JUST ENOUGH TO TEST AND VALIDATE

The aim here is not to get bogged down in extensive quantitative research, rather create just enough to test and validate your hypothesis, and use your user research to rapidly direct your solution and pivot where needed. There is nothing that replaces real world user engagement, and your customers won't be shy to tell you where you got it wrong.

CHAPTER 6

DEFINING YOUR DATA STRATEGY

"DATA IS A PRECIOUS THING AND WILL LAST LONGER THAN THE SYSTEMS THEMSELVES"

— TIM BERNERS-LEE

DATA IS ARGUABLY THE MOST VALUABLE DIGITAL RESOURCE THAT YOUR BUSINESS HAS

Your business likely consists of many systems that collect valuable information from users and business processes.

With the world becoming increasingly aware of data privacy and security, having strategies defined to protect data is becoming increasingly important. Protecting data isn't your only concern, you also need to use your data for insights to help drive decision making.

A good data strategy is defined by:

DATA DEFENCE

Focus your systems on minimising your risk and to ensure safe and compliant use of data.

DATA OFFENCE

Focus your systems on supporting your business objectives with insights and analytics.

Building a new system in your business means finding a balance between these two strategies that matches your regulatory environment and product offering.

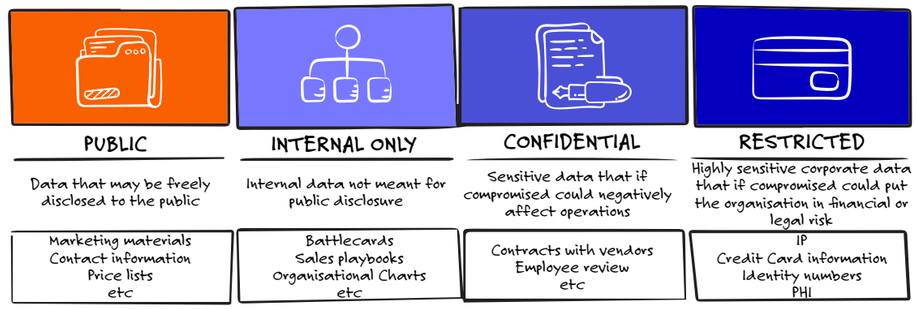
DATA DEFENCE

KEEPING YOUR PRODUCTS' DATA SAFE AND SECURE.

Customers trust your business with their information, and governments are putting pressure on businesses to ensure data is handled with strict privacy controls in mind. Working data in your system and keeping all the security controls in mind can be daunting for your teams but you can simplify the process for everyone involved by making sure you have a well formalised classification policy.

DEVELOP A FORMALISED CLASSIFICATION POLICY

Using the following 4 categories to classify data fields in your system will help you keep your teams aligned with your data strategy, knowing what fields to store and how access to those fields should be controlled.



BAKE IT INTO YOUR PROCESSES

To get the most success out of your data classifications you need to get people thinking about it frequently. Teams must introduce it into their scoping and planning sessions for new features, and architecture reviews should highlight insecure access controls for sensitive fields. Data flow diagrams can also be used to highlight where sensitive information is flowing between systems.

DATA OFFENCE

DATA OFFENCE IS OFTEN USED TO GIVE YOU A STRATEGIC EDGE, AND WHEN USED CORRECTLY, **IT CAN HELP GUIDE YOUR ORGANISATION'S DECISION MAKING PROCESSES.**

With all the data being collected by your system, making the best of that with analytics and business intelligence is a must. Using your data to identify patterns, problem areas, and new opportunities places your business into a data offence mode that gives you an advantage over your competitors.

Large businesses have many complex systems and divisions often have different versions of the same data. This introduces a problem if too many versions of truth for the same entities exist because it reduces your ability to use your data offensively. To make sure we can get the most value from our data for insights, you need a strategy in place to ensure your system will "speak the same language" as the other systems in your business.

3 STEPS TO CREATE ALIGNMENT

THERE ARE 3 STEPS THAT YOU CAN TAKE THAT WILL HELP ENSURE THAT YOUR SYSTEM WILL BE ALIGNED WITH OTHERS IN YOUR BUSINESS.

1. COLLABORATE YOUR NEEDS

Identify the various entities that you will be storing in your system and find other systems in the business that also have these entities as concepts. Have workshops with the stakeholders of these systems to make sure that your system will have the same view of data so that it can be integrated seamlessly into a data warehouse or a business intelligence solution in the future.

2. MAKE YOUR DATA USEFUL

To gain the most from your data, you need to ask the right questions that will define the data that needs to be captured and stored. Some basic ones that can get you started are:

1. Can we capture data that will drive value in other business areas?
2. What metrics do we want to measure and are there existing KPIs that can guide us?
3. How will we visualise the data and are we capturing the information to make that possible?

3. MAKE IT ACCESSIBLE

Humans consume information and identify patterns easily when it is visual. Business Intelligence tools have powerful reporting dashboards that can make your data visual with graphs, maps, and more that can help you see information in your data that only comes to light when viewed visually over time. If your system is used in day-to-day operations you should also provide an operational dashboard for line managers that can help them identify and react quickly to negative patterns that they see.

Systems evolve and change over time - so must our strategies. You can start with a strong defence and as your system grows and you capture more data you can move towards a more offensive stance. Systems can and should be built with their data strategy in mind, keep sensitive information like credit card details in a system that is optimised for data defence and keep all the other information in a separate system optimised for data offence.

CHAPTER 7

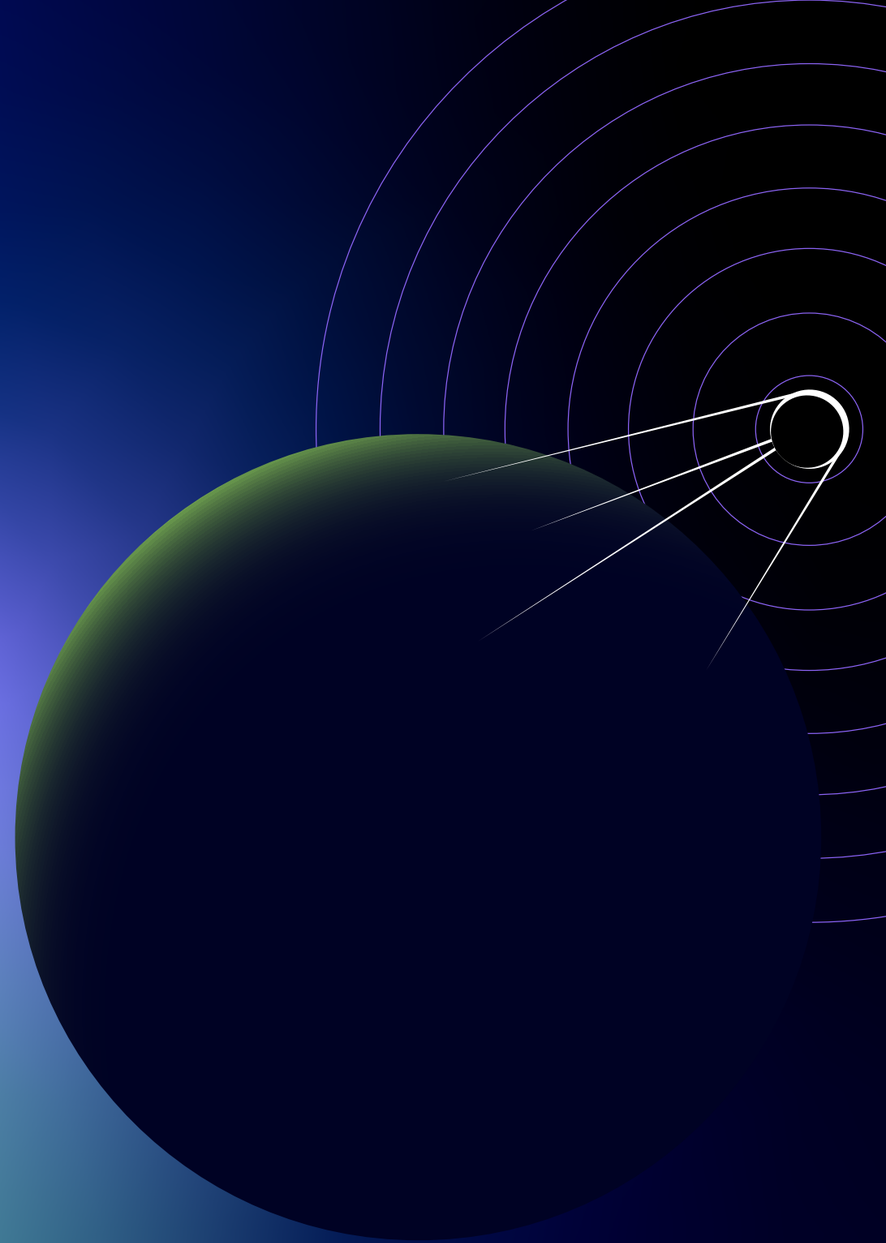
WHAT ARE WE ACTUALLY BUILDING?

IT'S IMPORTANT TO ZOOM IN A BIT, AND **CONSIDER THE TECHNICAL BUILDING BLOCKS AND POSSIBLE ARCHITECTURES.**

This involves breaking a system down into parts and determining how those parts will communicate. It involves identifying user facing components vs background processes.

Layering on the high-level data model allows you to see how the data moves through the system and where it ends up.

As the system is built, a lot of this may change as new information surfaces, but spending the time up front on this analysis helps to validate that the requirements are technically possible and starts to give some idea of the complexity, cost and timelines required to complete the project.

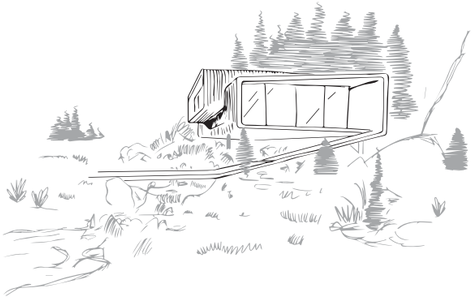


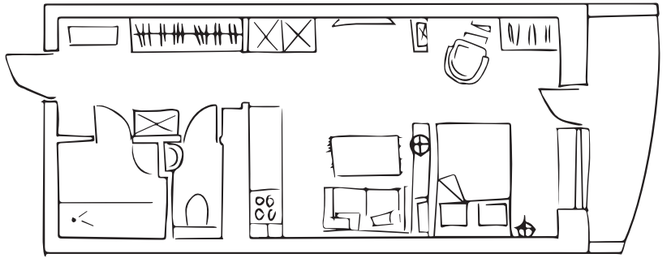
HIGH-LEVEL ARCHITECTURE

SOFTWARE ARCHITECTURE PLAYS OUT AT MANY LEVELS WITHIN A SYSTEM. FROM VERY HIGH-LEVEL ABSTRACT THINKING, ALL THE WAY DOWN TO THE STRUCTURE OF THE ACTUAL CODE AND EVERYTHING IN BETWEEN.

While the desired system is not yet clear, it doesn't make sense to dive too deep into architecture. There is definite value in starting to figure out how the system will be structured at a high-level.

If we were to compare a software system to a physical building, this is the stage where the architect would put together an "Architect's Impression" or concept drawing of what the finished product should look like. This allows stakeholders to visualise how the new building would fit into its surrounds. It also doesn't require much knowledge of construction to understand so it can be used to get buy-in from all stakeholders, while at the same time, those with construction expertise and experience will already start thinking about how to accomplish the vision.





Continuing with this analogy, floor plans provide a bit more detail and require a little bit of understanding to interpret but are also critical for providing detail to all stakeholders.

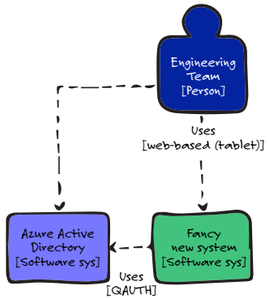
There are many parallels we can draw to software architecture at this stage in the product design. A popular modelling method called C4 is often used to achieve the same goals within a software architecture.

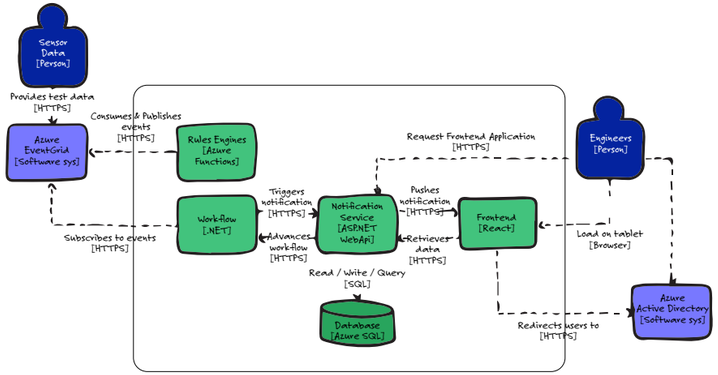
C4model.com →

SYSTEM CONTEXT DIAGRAM

At level 1, an architect models how this new system or product will fit into an existing ecosystem (much like an "Architect's Impression"). This is an extremely zoomed-out view without a lot of detail (in fact, the system itself is normally represented as a single block), but it does show how the system will interact with other systems as well as who (people) will be interacting with it.

Any non-technical stakeholder with a little bit of domain knowledge should be able to follow the diagram and to provide valuable input and feedback at this early stage.





CONTAINER DIAGRAM

Level 2 of the model then zooms in slightly to break the single block down into course-grained chunks. This starts to get a bit more technical but is still high-level enough that most stakeholders would be able to reason about it (like floor plans). This is where we would start to think about frontends, backends, databases and orchestration layers.

At this point we should also have enough information to start making tech stack choices. Technical stakeholders would also have enough context to get their creative juices flowing in terms of actual implementation details.

The C4 model continues to levels 3 and 4, with increasing amounts of detail. These levels are however not beneficial at this stage, while the focus is still on the broader use case and business requirements. Once the implementation phase begins, teams will find it useful to start modelling at the more detailed levels.

DATA FLOWS

While the C4 diagrams provide a good idea of the parts of the system and how they communicate, sometimes it is valuable to drill down into a few common use cases and to focus on how specific sets of data would flow through the system. This detail informs decisions around orchestration within the system and integrations into external systems. A simple flow-chart helps to visualise and to distil these critical details, and while they can get quite technical, they can also be useful to affirm domain understanding from non-technical domain experts.

Shop Floor System

TECH CONSIDERATIONS

We live in a world where we're constantly bombarded with the "next big tech stack". While this is exciting as it opens up so many possibilities, it also brings with it some very difficult decisions around the correct tools for the current scenario. There is rarely a perfect choice for your new venture, but there are some things we consider helpful when making a choice.

PEOPLE

People will have strong opinions and vested interests in particular technologies that will need to be considered.

CONSTRAINTS

There will inevitably be roadblocks that cannot be moved, you'll need to work within these constraints, such as enterprise architecture, sunk investments, existing licenses, etc.

REQUIREMENTS

Will your chosen tech stack help you achieve the actual reason you're building something in the first place? i.e. your functional requirements.

SKILLS

Do you have existing skills that you need to leverage? Are there people in the marketplace you could hire with these skills?

RISK

Will this technology exist in a few years' time (i.e. do the maintainers have a vested interest in keeping it going)? Are there privacy/security concerns around the tech?

CONSISTENCY

Sometimes it makes sense to continue using what has worked in the past, even if it's not the latest and greatest. A consistently maintained and understood tech stack sometimes trumps all else.

PROJECT SPECIFICS

While the previous items point to the greater tech landscape within the organisation, each project will also have very specific needs. Consider deadlines, budget, and team capacity.

CHAPTER 8

PREPARING FOR LONG-TERM SUCCESS

DESIGNING NEW PRODUCTS OR **SOLVING A BUSINESS CHALLENGE CAN BE REALLY, REALLY FUN.**

Whether we're drawing mock-ups on a whiteboard or debating user journeys, it's a creative, stimulating, and collaborative exercise with our colleagues. It's also challenging; pushing the boundaries for you and your business, and the stakes are always high. It's the perfect cocktail for a creator's high when you gain that critical insight, uncover that elegant solution, or discover that golden nugget.

It's incredibly tempting to dive straight into the implementation of solutions once we've figured out how to solve our problem, it's natural as humans, especially when we're on the creator's high. Unfortunately, we come off that high eventually, and we're left with the daunting prospects of seeing our ideas through to fruition, which could be months, or even years away. How do we capitalise on that burst of energy from the creator's high to invest in the long-term success of the project?

We're going to look at how to take the first steps towards a well-oiled SDLC, how to test our ideas quickly and gain feedback, and how to prepare to measure our progress. We'll also break the cardinal rule; we'll talk about Fight Club.

PLANNING IS SACRED. PLANS ARE NOT.

With all the excitement and pressures to get a new digital product launched, it's undesirable (and probably infeasible) to spend months planning out the implementation of the solution in fine grain detail. It's impossible to know in a year's time what features the engineering team will need to be working on as a matter of priority, nor what issues and complexities may have been uncovered during the course of implementation. New insights in the business may also change priorities and requirements of the solution.

So, while it's unrealistic to have a comprehensive and accurate long-term plan for a digital product, it doesn't mean that we can't have one at all. The recommended way forward is to take on the planning in smaller, more iterative steps. You may have heard of this as "Agile Software Development", but we'll avoid labelling this for now and rather focus on the principles.

Whether it's called sprint planning, workshops, backlog grooming, prioritisation meetings, retrospectives, stand up, or a good old fashioned "project catch up", these are all forms of planning. Collaborative activities allow the team to get together and align their vision, debate the issues at hand, and agree a way forward.

Successful teams usually perform planning activities multiple times a week, in a mix of formal and informal settings.

For example, if you follow SCRUM, the recurring sprint planning meeting is a formal setting for the team to form a relatively short-term plan for the next few weeks (depending on your sprint cycle). Sprint retrospective is similarly a formal setting where the team reflects on the past, and plans ways to increase quality and effectiveness. Even the simple daily stand up is an opportunity for the team to plan the next few hours of their day together.

Successful teams don't spend unnecessary time in planning meetings.

You'll know you're getting it right if you're genuinely not using all of the meeting time you've set aside for planning. Everyone loves it when a meeting can end early. Conversely, you'll know that you need to spend time planning more frequently if you're always running overtime in planning meetings. The answer isn't to book longer meetings, but rather have shorter meetings more regularly.

Successful teams don't include every single person in every single planning meeting.

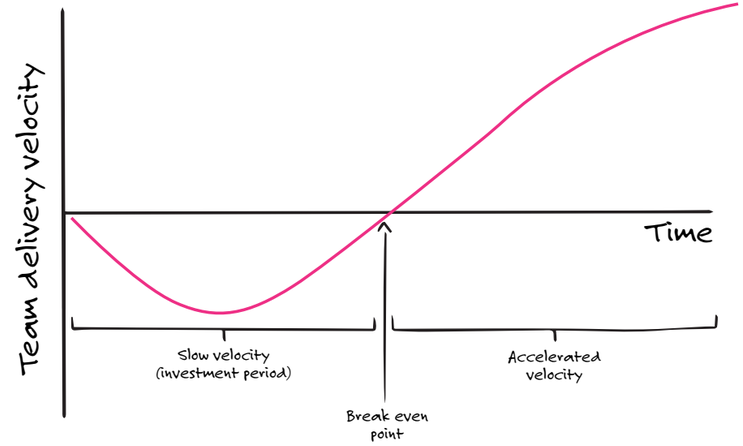
While it's tempting to have everyone from the CTO to the intern in the planning meetings so that "everyone is on the same page", this is a colossal waste of most people's time. Successful teams often co-ordinate who needs to contribute to any given planning activity. They trust in each other that they'll be "roped in" when needed, and they trust each other to reach out for help & information when they're uncertain.

Successful teams take notes and use them.

Whether it's a task created on a digital project management tool, or simple scribblings in a notebook, your team's notes are the raw material that should be feeding into your planning sessions. It's impossible for the team to plan for something that's only in the Product Owner's head.

The characteristics of successful teams highlighted above are manifestations of a deeper principle so elegantly captured in the Agile Manifesto: we value "Responding to change over following a plan".

INVESTING IN THE TEAM EARLY ON UNLOCKS ACCELERATED VELOCITY VERY QUICKLY



Successful teams establish the above characteristics early. They invest time in themselves to establish and continuously improve on their planning activities and habits. This yields more realistic progress tracking of the project, deeper understanding of the trade-offs when making changes to the plan, and having confidence that everyone is pulling in the same direction. Successful businesses give their teams the time they need to make this investment and hold them accountable for the return on investment.

CONTINUOUS INTEGRATION AND CONTINUOUS DELIVERY

HOW DO WE KNOW WHAT WE BUILT BEFORE IS STILL WORKING?

In a post-pandemic world, most industries have come to the realisation that software development is a highly collaborative exercise, not a solo effort. It takes a team of engineers to build the large and complex systems that drive your business. If you think collaborating on a word document is difficult, try a code base with a million lines of code.

Continuous Integration is a software development practice where members of a team integrate their work frequently, usually a few times a day. Each integration is verified by an automated build and testing process to detect integration errors as quickly as possible. Many teams find that this approach leads to significantly reduced integration problems and allows a team to develop cohesive software more rapidly.

WHEN CAN WE USE THE LATEST FEATURES?

We all know that no product survives first contact with real world users. There's always something not quite right, an assumption that was made that's incorrect, or a better potential value proposition.

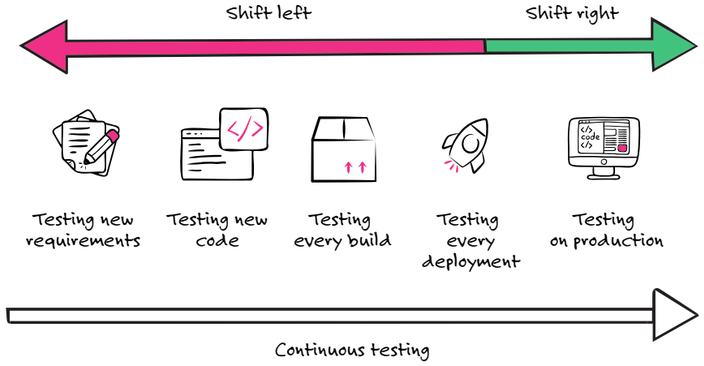
So instead of fearing this first contact, embrace it.

Characteristics of successful teams who perform hands on testing:

- Successful teams test with real users as early as possible
- Successful teams are hungry for feedback
- Successful teams release often, relying on their established Continuous Integration and Continuous Delivery processes for safety and confidence in their releases
- Successful teams "shift defects left"

Shift Left is a practice intended to find and prevent defects early in the software delivery process. The idea is to improve quality by moving tasks to the left as early in the lifecycle as possible. Shift Left testing means testing earlier in the software development process.

HANDS ON TESTING



DEFECT TRIAGE

BUGS ARE NORMALLY FIXABLE IN A FEW HOURS. ANYTHING LONGER AND YOU'RE PROBABLY BUILDING A NEW FEATURE.

WHEN WILL THAT BUG BE FIXED?

A natural side effect of hands-on testing is that defects or bugs are identified. All software has bugs, and no amount of willpower in the world will produce bug-free software on the first attempt. So instead of fighting them, embrace them.

A defect or "bug" actually captures some feedback from a user where they have identified that something doesn't quite work the way they expected, or doesn't satisfy their needs.

It's important that all defects are captured as soon as possible after identification, with as much information as necessary for all stakeholders to be able to readily reproduce and discuss the issue.

In a medical context, Triage is the process of determining the priority of patients' treatments by the severity of their condition or likelihood of recovery with and without treatment.

Likewise, in software, stakeholders and teams must regularly discuss bugs and defects, not only to prioritise them, but also to stay on top of them.

It's easy for the defects to pile up in a backlog somewhere and become insurmountable. It's much more efficient to meet once a week and discuss a few defects, than once at the end of the project for hours at a time to wade through a pile of defects, most of which are no longer relevant.

Systems where defects are swept under the rug, or "will be dealt with at the end of the project" inevitably gain that unshakeable perception of being "buggy", which will harm your digital product, no matter how innovative it is.

Successful teams are honest with each other about what's a bug and what's a feature.

It's very common for users to log new feature requests or enhancements as "defects". They've identified something wrong with the system that prevents them from achieving their objectives in the software, and it must be "fixed", even if that means building new features or changing features that are already working as by design. Regular defect triage creates opportunity for everyone to discuss defects whilst they're still freshly identified; pick them apart, understand them, and identify the root causes.

It's common to see some of the most valuable features in a system start out as a defect or "bug" and evolve into well-defined feature requirements. They've just been given the time and space to be properly triaged.

FIGHT CLUB

A GLOVES OFF TECHNICAL DEBATE

Throughout the development of a project, it's common for the team to come across a technical problem where there's either no obvious solution, or many competing solutions.

The team could gather all stakeholders to try take a decision by committee, however, this is often ineffective for highly technical topics. Alternatively a Technical Lead or Solutions Architect may be brought in to provide guidance, but this too could take valuable time, as they'll need context and to be brought up to speed.

Often an engineering team already has all the knowledge and experience to make a technical decision, they just need to talk it through in a no-holds-barred fashion, free of restrictions or hampering conventions.

Much like the movie from which it gets its name; Fight Club creates a space for "robust" discussion where at the conclusion, everyone can simply walk away knowing that none of it was personal. All team members involved are simply focused on putting forward the best technical solution in their view.

Fight Club is a technical discussion. Everyone should come prepared. The team should not have to pause the discussion every few minutes to explain the basic concepts of the topic.

Whilst this may sound unsettling... non-technical stakeholders are not welcome at Fight Club. This is because an engineering team needs the space to put forward all their ideas; the good, the bad and the ugly. They need to be able to debate the merits and consequences without fear of rebuke.

A team is still accountable for the decisions they take at the end of Fight Club, and must feedback to all stakeholders their rationale and justification. However, during the Fight Club, they must be unencumbered by this responsibility.

CAST YOUR NET WIDE

We have a natural inclination towards spending more of our time and effort in the solution space. It's far more exciting and interesting there as we're innovating and building. In this haste, we often leave unexplored many parts of the problem we're trying to solve.

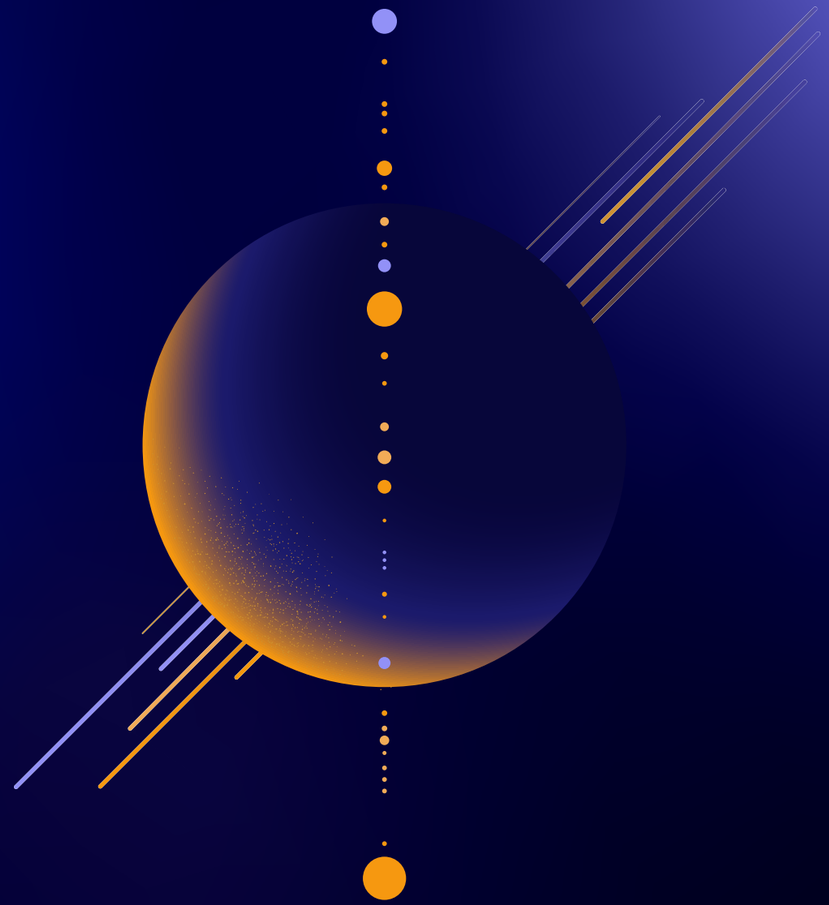
There is clearly a substantial toolset available here, with options for deep and detailed analysis of requirements, business and technical context. This is likely to be overwhelming at first, it is intimidating to browse through such a variety of methodologies and feel any kind of confidence about where to start and what to pick.

The intention is to be pragmatic about the preparation and thinking required before a technology project, and ultimately the risk is not in being too shallow during analysis of one specific dimension, the risk is in missing some dimensions altogether. Frequently, the major issues will jump out as red flags early in the process of looking at each dimension, with diminishing returns as more time is spent in deeper analysis. We're trying to find deal-breakers for the product, and eliminate any dissonance between stakeholders – especially between technology, business and sponsors.

This means the objective for our teams should not be to exhaustively specify, research or design the solution upfront. The goal should be to ensure prudence in that every dimension is considered, and to pick one or more methodologies to cover those bases.

We hope these perspectives and guidelines inspire some ideas, or become useful and practical as you explore the problem space for your next digital initiative.

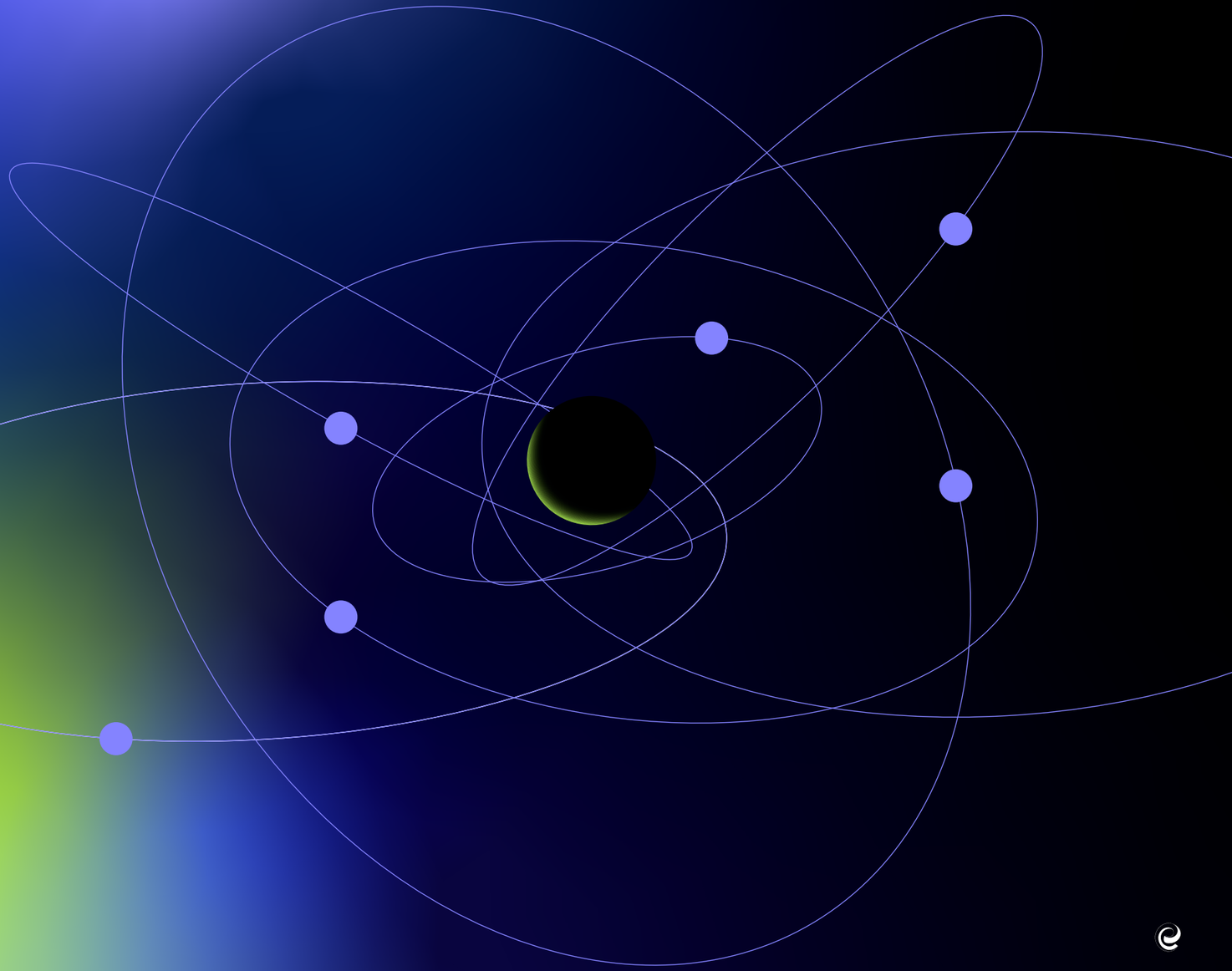
Good luck!

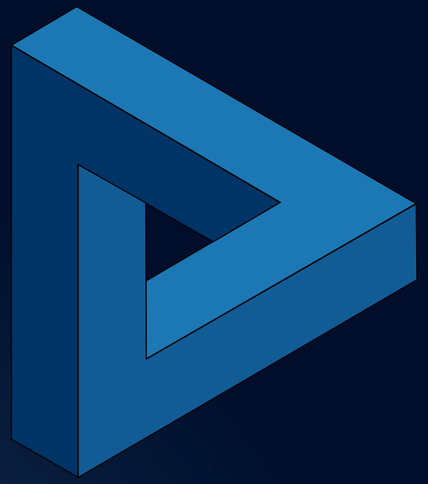


WHO MADE THIS

A LIST OF CONTRIBUTORS

Deirdré Nel
Duane McKibbin
Hennie Brink
Jacques du Preez
Jolene van Heerden
Matthew Butler
Matthew van der Velden
Pierre Leeflang
Rishal Hurbans
Sean Buch
Sophie Laher





START THE CONVERSATION

solutions@entelect.co.za

We explore more, to help businesses become more.


entelect.co.za