

Trajectories for UX designers

By Dan Ramsden

This document contains some of my ideas.

I like having ideas, developing them and sharing them. I don't think having 'half-baked' ideas is all that bad. But the best way to take a 'half-baked' idea and make it better is to share it. It seems the old saying 'too many cooks' doesn't apply to everything.

So I decided to share these ideas. I hope you find them useful. But even if you don't, please let me know.

You can find me online at www.danramdsen.com, on Twitter @danramdsen and email me using dan@danramdsen.com

It's also worth stating that Steve Benford introduced me to most of these ideas. You can find out more about Steve at www.bit.ly/SteveBenford

This booklet is about gaps.

Gaps are where most of the interesting design challenges are lurking, particularly if you're interested in information architecture.

Information architecture is fundamentally interested in gaps – gaps between problems and solutions, between products and audiences, between perfection and pragmatism.

Gaps invite us to think about the differences that cause them and the connections that might bridge them.

Gaps keep us in a job.

But gaps aren't the only things that matter.

I like to think of information architecture as the science of separation, the art of connection and the technology of pointing. The gaps do matter, but only because they're the space between things – and if we're honest, the 'things' usually get most of the attention.

Information architects are fascinated by things and are preoccupied by considering their parts. We analyse and understand the properties of a thing so that we can describe its essence. We're critically interested in simplification. We strive to make complex things more meaningful. So often we break complex things up into smaller, simpler parts. We simplify and separate. We build gaps.

But although there's this necessary commitment to reduction of complexity, I don't think information architecture is a fundamentally reductive activity. Information architecture sets a template for the strategic construction of things. We build stuff – we make places out of information. So, as much as we separate things, we also connect things. We find similarity between and within the classes of things we've identified. We find and form connections.

Once we've identified connections we devise and design ways to exploit these connections. Whether we're designing digital navigation or coherent cross-channel experience, we focus on links. Links often depend on pointing at things – whether we do that with wayfinding aids or technologies that take you somewhere – like the hyperlink.

“**The most effective IAs don't just consider the places they're making, they also consider and design the spaces between the places.**”

This is what an IA does - we simplify, separate and connect things, identifying the 'stuff' we can build places out of, building and bridging gaps and making experiences meaningful. We create places that users can explore and which can be the setting for meaningful experiences.

The space between

So we design places with an understanding that there will always be gaps between the parts of the whole, and that it's our job to identify where and when these gaps need bridging. But there are other gaps that we care about too.

There are always spaces between the places we make. There are always gaps between our intended design and the first iteration. There are always gaps between our vision and the way colleagues and stakeholders might imagine a design.

No matter how good a design is, it's never really finished. There are always gaps between what we release and what it will become.

With any piece of information architecture, the implications of our decisions ripple across the whole experience. IA defines the shapes that can appear in the interface. It feeds back into the strategy for future ideas, iterations and innovations. IA controls and curtails opportunities – it reaches into the future and defines what is possible. IA travels in time.

This is one reason that when you're designing information architecture (or really any type of UX design), you need to make sure that you've got the buy-in of the organisation. You need to have seeded an understanding of the implications of your IA approach, alongside the tangible things you build.

If the success of your IA is predicated on the assumption that the connections will come, in time, then you either need to stick around long enough to build the connections, or seed the culture so that everyone understands that the work is not yet finished.

So how do we do that? This booklet describes one way – Trajectories. It's a design and storytelling technique that should help you design better experience by bridging gaps – whether they're between team members during the design process, between iterations of a design as it evolves or within individual designed experiences.

What's a trajectory?

A trajectory is the path that a moving object follows through space as a function of time.

It's also a line. It has a beginning, middle and end.

It's a type of story.

A trajectory might look like a simple line - and it is. But a simple line can be a very powerful tool.

IAs tend to spend a lot of time thinking about space. Information architecture is preoccupied by the spaces and structures that our design implies. But TIME is just as important as space. Information architecture underpins experiences, and experiences happen in time. If you only ever present your IA as a static structure, you're underplaying the power of your design.

IA is built to be experienced, so you need to find ways to communicate the impact that your IA will have on the temporal and sequential nature of the experiences it will contain.

Architectures also evolve in time. It's an extra dimension we need to keep in mind as we design. So whether we're describing a user experience within our architecture, or the evolution of an information architecture between design iterations, time is just as important to an IA as space.

Trajectories are a way of focusing your attention on the temporal nature of experiences. At their most basic, a trajectory is just a line that describes a path through an experience or information architecture. Their power also lies in their ability to focus your attention on any ‘moment’ or point along that line.

As we design experiences, it can be tempting to tell stories in rapid cuts - as if all user experiences are really unfilmed music videos. Whenever you hear a sentence that begins ‘and then the user...’ think about how well the sentence is connected to the one that preceded it and the one that follows. This kind of storytelling can sometimes suspend the ‘user’ in a mythical hinterland where the motivation or context for their understanding and the action they’re taking has not been thought through. Storyboards are a great way to describe experiences. But sometimes the secrets of a design are lost in the gutter between the cells.

Something as simple as drawing a line allows you to trace the path of that action forwards and backwards to determine the motivation for the action, as well as the trajectory that the action suggests. A line has continuity – it’s connected. You can still describe the spaces, both between places and times, but a line encourages you to imagine the coherence and possible connections between the gaps.

We’ll start off by describing this simplest sort of trajectory – a line that follows a path between A–B. I’ll describe a set of transitions, points along that path that are of particular interest to the UX designer and information architect.

Once we’re comfortable using trajectories to trace the experience of an end-user of a designed experience, I’ll describe how the same technique can be applied to increase the effectiveness of design teams.

Design is about telling the story of a world that could exist. I think telling stories is at the heart of iterating and sharing design including information architecture.

By considering the path that you’re creating for your users, and the path that you’re on as an organisation, you can better understand the opportunities and obstacles that stand ahead of you.

Every design decision has consequences, whether you build a dominant organisational paradigm based on taxonomy, or you focus on modelling domains and assume connections will be formed organically, for you... there are always consequences

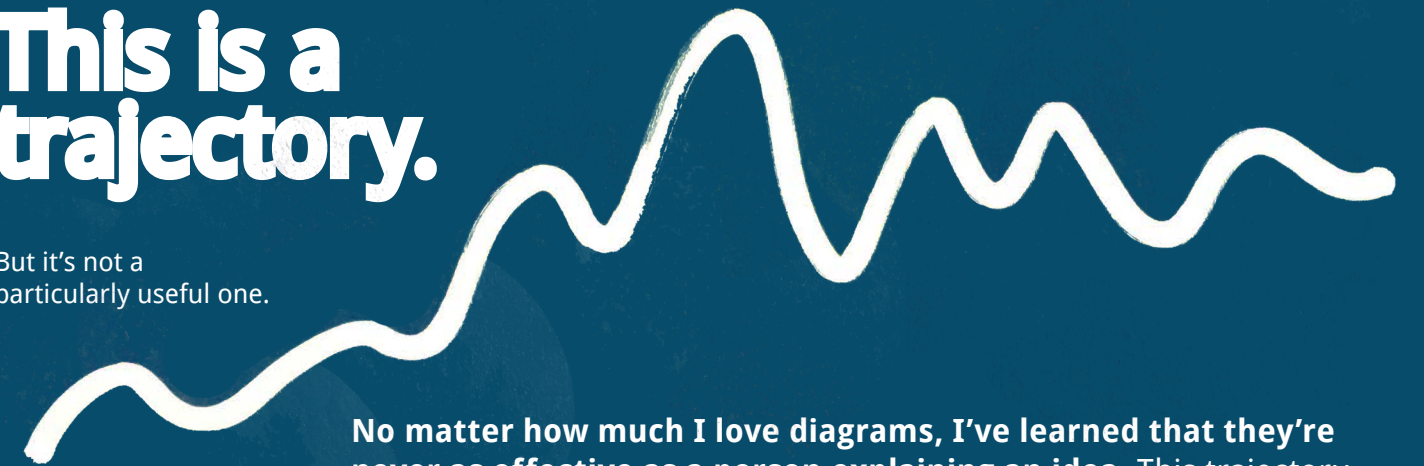
Trajectories can be used to reflect on the direction that you’re travelling in, explore other direction, judge the energy needed to connect the elements of your designs. It’s a tool that can be used to judge a design, as well as the processes that will be required to realise a design.

What’s the trajectory of the information architecture you create – how is it transformed from an idea into a reality?

Stories, whether you use trajectories or not, are another abstraction you can use to zoom out and judge your design. They’re helping us to identify the challenges that stand in front of us as we bind together the smaller pieces of the larger ecosystem we’re continually making.

This is a trajectory.

But it’s not a particularly useful one.



No matter how much I love diagrams, I’ve learned that they’re never as effective as a person explaining an idea. This trajectory is pretty useless. Even if you add some annotation (always a useful addition to a trajectory or any form of diagram) it will still be limited in effectiveness. But trajectories can be brought to life, they can be ‘animated’ by telling the story of the path that the trajectory describes.

Imagine me running my finger along the line, describing the purpose of the curves. Trajectories can be used to document a design, but I don’t think that’s where they’re most useful. Trajectories are most effective when they’re more than just a line – they should be the basis for a design conversation.

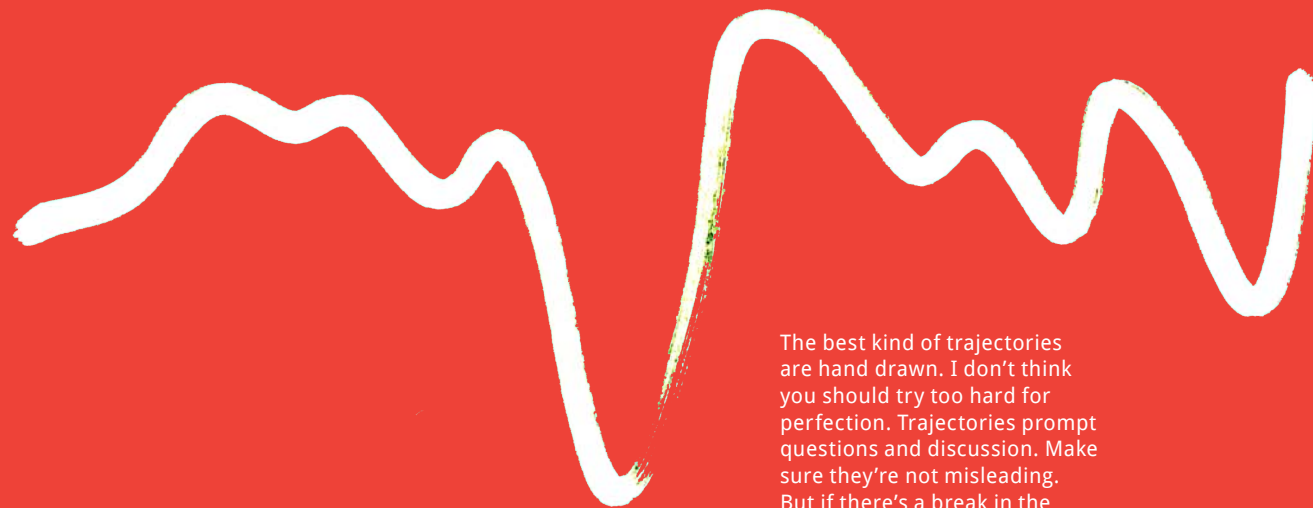
Think of the line as a path that your conversation might follow. When you need to deviate from the path you’ve imagined, that reveals something about the design – work hard to make the implicit assumptions baked into the abstract trajectory really clear.

A trajectory line can describe more or less any aspect of an experience as it occurs in time. As with any 2 dimensional line, you have access to a couple of axes.

‘X’ is usually time.

But we don’t have to use the Y axis to describe a movement through space. It could be a measure of attention or engagement. It could be used to denote a user moving between different contexts or technologies or platforms. Or it could mean nothing at all. You might not even have two axes, a trajectory might snake its way through a domain model, bringing to life the relationships, connections and properties that stand between objects and entities.

Some examples



The best kind of trajectories are hand drawn. I don't think you should try too hard for perfection. Trajectories prompt questions and discussion. Make sure they're not misleading. But if there's a break in the line, or you make a mistake, don't worry too much. The discussion that they prompt is where the real value lies.



I'll soon talk about different types of trajectories. The intersection of different journeys and the variation in experiences is easily noted using trajectories.

This reminds me of the graphs I sometimes see on my smartphone once I return from a bike ride. The phone detects gradient and plots it against my speed. Seeing how going up hill slows me down is not a huge surprise. But using trajectories to explore the effects of the features of your experiences is a powerful tool, and usually leads to insights.



You can use trajectories to show movement through different devices or platforms. They're a powerful tool for describing cross-channel experiences. I've used variations in contextual studies to show how editorial workflows are dependent on multiple tools and processes.

The designed experience.



This is the ideal experience. I suspect it doesn't exist. But if it does, it looks a bit like a fat snake, waiting to contain things.

We've found it helpful to think about three types of trajectory.

One is the designed experience.

Whenever you're designing something, there's an optimal path through the thing you create. We've all sat at a desk imagining a sunny day when everything goes right and the user flows through the experience in exactly the way we always imagined.

This is the ideal experience. I suspect it doesn't exist.

Experiences are always unique. A user will always contribute something to the equation of the experience – they always bring something. Most trajectories are a line, but I think it's a good idea to make the line that describes this 'designed experience' a little fatter than the others.

We should always be designing resilience into the experiences we give to users and audiences. I like to describe this tolerance in the trajectories I design.

If I know that there will be different levels of familiarity with an interface, I might describe a bulge in the trajectory at this point. This allows me to describe that although there are different levels of competency in my users, I've designed for a range. It also allows me to describe what might fall outside of the range of the designed experience. This will prompt consideration and often discussion as you tell the story of the design.

Is there enough tolerance in the design? Is there too much?

The trajectory of the designed experience describes the range of different user experience that your design is capable of supporting. It can also be used to denote the boundaries at which you have accepted the fact that the experience might fail or break down – which in turn can be used to design measures to pull individuals back on track.

The individual trajectory.

Every user will have their own 'individual experience' of the design you've provided.

If you've designed tolerance within your product or service the second type of trajectory will hopefully lie somewhere within the region of the designed experience.

This second type of trajectory describes the path that an actual person might take through the experience.

Because we acknowledge that each user will have their own experience, we can describe multiple individual trajectories for the same designed experience. This is one of the most powerful tools that trajectories give us. It allows us to think about the different ways that the stimulus of our design can be interpreted and used by different individuals.

Information architecture often makes me think about the 'Choose your own adventure' books that I read as a child. These books have multiple narratives running simultaneously within the same book.

At the bottom of key pages the reader is given a choice, "To go in the cave go to page 42, to run away go to page 66." I like to think of information architecture as providing the same kinds of structure.

The structure remains the same. The choices that each individual user can experience differ. They're still usually based on an acceptable range of responses for each individual choice we offer, but the user is mainly responsible for making the experience meaningful, based on the stimulus we provide.

Thinking in these terms allows us to focus on a broader range of supported experience, expanding outward from the 'designed experience' to explore the 'individual experience' in more depth and richness.



The historical trajectory.

The historical trajectory is the story that participants will be able to tell about the experience that they've had.

It might be memories of the experience or it might even be physical artefacts. It might even be a social share that they create to pull people into having a version of the experience for themselves.

It's important to consider the historical trajectory for a number of reasons. After an experience is over, the historical trajectory is all that the user is left with, so if you want the experience to remain meaningful and shareable it's important that at least some elements can be remembered and described.

Thinking about the description that a user could give at the end of an experience is often a good way to identify possible points of confusion or pain.

Transitions

So, three kinds of lines to describe designed experiences - is that it?

Well, kind of.

But within any story there are punctuation points – chapters, sections, pages. The same is true of all experiences. Experiences are made up of parts and in the language of Trajectories the gaps between the parts are called transitions.

Transitions are points in your trajectory where you need to pay the most attention. They mark points at which there might be a change of direction or where the continuity of the experience you're describing is in jeopardy. These are the moments of risk and reward for a designer. And because they represent both risks and opportunities, it's important to understand them.

Information architects know the importance of naming things. Giving something a name helps you to recognise things of the same type. So over the next few pages I'll describe a few different types of transition that it's useful to think about, and give each one a name.

If you're successful in adopting and seeding Trajectories as a way to explore user experiences in your organisation, you should find that these Transitions become a checklist or a set of heuristics that anyone can use to investigate and analyse the success of a design or experience.

Beginnings provide context and help us describe the motivation of the user.

Considering beginnings can help you to explore whether your 'designed experience' reaches back far enough. How do people enter or begin their experience with you? What are the foundations you're building on?

Thinking about beginnings will also encourage you to consider the range of routes into a designed experience. Where do your users come from? Do you prefer any particular inbound referrer or source?

Could you enhance the entire experience by reconsidering where and what you consider as the beginning?

Types of things you might learn by thinking about beginnings:

- Does the user have all the information they need to take the first required action in the experience?
- How might you enhance the motivation of the user at this stage to ensure the experience begins?
- Is the required first action obvious across the full range of 'beginnings'?

Let's start at the beginning.

Roles

People play different roles during experiences.

Recognising the roles we want users to adopt, as well as the roles they might just 'fall into' will help you to design better experiences.

Search behaviour is one good example of the way a users intention and motivation might shift throughout an experience. At times users might be tightly focused on finding a specific item or resource – you might describe this role as 'motivated and informed seeker' – they know what

they want, they just don't know where it is. At other times the same user might be looking for distraction and be open to browsing – this 'curious browser' would probably appreciate different features and functionality than when they inhabited the other role.

Another example of role might be 'beginner', 'intermediate' and 'expert' users. They all experience the same stimulus, but they are likely to perceive and use the affordance you provide very differently.

Role is one of the most fundamental aspects contributing the context that your experience will take place in.

I like to think that good design and information architecture can orchestrate experiences. By considering the roles that a user might shift between, and the optimal roles required to deliver delightful experiences, we've got more information and inspiration to nudge users into the desired states and mindsets.

Suggestion:

Consider devising a controlled vocabulary of roles that a user might inhabit during their experience.

Interface

The interface is the means by which we meaningfully communicate with our audience.

An interface could be any means of communication – the user interface of a website, app or other digital product. It could be physical signs or way-finding devices, printed instructions or just the language we use when we talk to them.

Is the same interface appropriate throughout the whole trajectory and experience?

There are times when changes to the interface can have a profound effect. When we think about immersion and optimising experiences for engagement we'll often want to make the interface as invisible as possible – maybe the content and the functional interface are fused together. Maybe the interface disappears when it's not needed – think of most video players – but is easily recalled when required.

The interface will often be how you manifest the 'giver of the experience' to the

user. There will often be a bond of trust, so interfaces should be meaningful and reliable. But they needn't remain static. Considering how an interface might react to different contexts, devices or user needs might unlock otherwise unseen potential.

If you're designing cross-channel experiences you might also want to consider how your interfaces complement each other. How will you support switching between interfaces to ensure that your experiences have coherence and continuity as well as flexibility and resilience.

Remember:

Language is an interface. Clarifying and controlling the terms you use as you design your experience will make the process easier and more meaningful.

Temporal episodes

Information architects are preoccupied with the spaces we create using information and classification, but time is arguably the more important feature of experience design.

Experiences happen in time. Experiences have duration and sometimes they can even be spread over multiple episodes or sessions. Considering how your design responds to this will always result in a more resilient and meaningful design.

Does each episode mark a new beginning? How might this beginning differ for each episode? Why will an episode or session end? How might you prolong it? How might you encourage a new session to begin?

Asking these questions will reveal opportunities and challenges. But thinking about the answers will ensure that your designs reflect the actual flow of real experiences. Experiences are often interrupted, and they always occur over time. Always consider time and the evolution of the experiences you're designing.

Using the transitions discussed so far.

Using the transitions discussed so far.

Consider the impact on the user experience of beginning a 'journey' at different points.

In the old days we'd construct a structure imaging a journey would flow from A to B. Search engines enable users to jump into journeys right in the middle of the A to B routes. More complex models place beginnings at many different points. What sort of beginnings does your UX need to support?

Experiences can stretch across sessions. How might you enhance an experience by making it easier to pick up where you left off?

Some experiences might require a user to switch role or mindset. How could you support a user to transition between active and passive states? How might you define and support multiple roles?

Sometimes a user might switch interfaces. Think about how Google Chromecast asks a user to split their focus while they move from one device to another. The experience can be satisfying, but the seam between the interfaces needs to be considered.

Switching domains.

In the original conception of trajectories there was a transition called 'physical/virtual traversals'. It encourages designers to consider the important transition that occurs when someone switches from a virtual interface and digital experience into one rooted in the real world.

This is still a useful focus, particularly when you're considering how you might augment real-world experiences with digital enhancements, the transition between the digital and the real-world domain is a seam that should be considered in the design.

However, I think for most information architects we can extend our focus to consider

any switching between domains – whether this is a virtual/physical traversal or moving between differently oriented information architectures.

Even within entirely virtual experiences we can move users and audiences through different domains and sections. There might be continuity in the interface and temporal episode, but the underpinning architecture of the 'space' they inhabit can shift.

Consider the points at which a user might leave one domain and enter another - this might signal a new beginning or a shift in role too. But there will definitely be opportunity to make the transition smoother and more motivated and meaningful for the user.

Access to resources.

Where your design or experience is predicated on the assumption of access to a set of physical resources, identifying this dependency is hugely important. Any core dependency that your experience rests upon should be identified and understood. But it's likely that there are points in your Trajectory or experience where progress is entirely dependent on this access. Think of these moments as gate points, beyond which the experience or design cannot proceed unless the necessary resource is in place.

Access to physical resources provides a chance to consider where scarcity might be an issue. For example, 'moderating comments' might not always be considered a physical resource. But if we're not able to support 'user commenting' fully, experiences and products could be undermined.

Considering this type of transition should help you to design more resilient experiences. What redundancy could you design into the experience to increase resilience? What alternative resources might be able to serve the function once the optimal design is compromised?

Seams

Seams describe any other type of weak points in technology. Of course, language is a technology, so you can consider a broad range of seams when you design an experience.

Considering the technical barriers that could undermine an experience is vital whenever you design or deliver an experience.

Caching and off-line access to content is an obvious example of where considering seams can help ensure that participants remain inside our optimal designed experiences.

Seams are points of weakness. Rips can occur when you haven't tightly coupled the sides of a seam, so designing for seams is really a case of designing for resilience. What is the most efficient way you might bridge a point of weakness in your design? What error messages might you provide when everything goes wrong and the user or audience does find a point at which the experience breaks down?

Encounters

Considering the social aspect of the experience you're designing will enhance any experience.

Lots of experiences are social, and even when a user experiences a design in isolation, there might be multiple users experiencing the same space or domain at any one time.

How many concurrent users is the right number? How might you support asynchronous cohorts? What is the social aspect of your design?

'Encounters' can encompass other types of thinking too. How might the isolation of a user enhance or detract from an experience? How could you use pacing to bring different users into contact or avoid encounters?

Encounters allow you to focus on the inherently social aspect of any experience design – this can extend your attention to consider the social aspect of design. What are the social interactions you need to consider as you design?

Endings

Most experiences come to an end – for example, we’re now at the end of this list.

Considering the ending of experiences will help you create better experiences. If you know the ending that each user or audience member should reach, you have a quantified point you can use to judge whether your design or experience has been a success.

Designing Endings help you to consider the historical trajectory that a user will emerge with from the experience. What do they have at the end of the experience? Can they use this to share the experience or shape the future behaviour?

**This is not
the end.**

Trajectories, co-design and design research

Trajectories are a pretty simple idea to understand. It's really just a language to describe experiences – combining some specific terms and a rough style of sketching a diagram. Because it's so easy to grasp, it's also a good technique to use in design research and co-design sessions.

Once you've described the basic idea of tracing the story of an experience across a two-dimensional line, everyone can do it. If you avoid using the technical jargon of Trajectories with the participant, you can still identify transitions and explore ways to manage and enhance them.

I like to think of this application as being similar to the real-time polling you sometimes see in political debates. Graph-based visualisations describe audience reaction during a speech. It's easy to encourage users to provide the same kind of feedback as they walk through a design or prototype – Trajectories give a simple, standard way to capture these insights.

Domain modelling

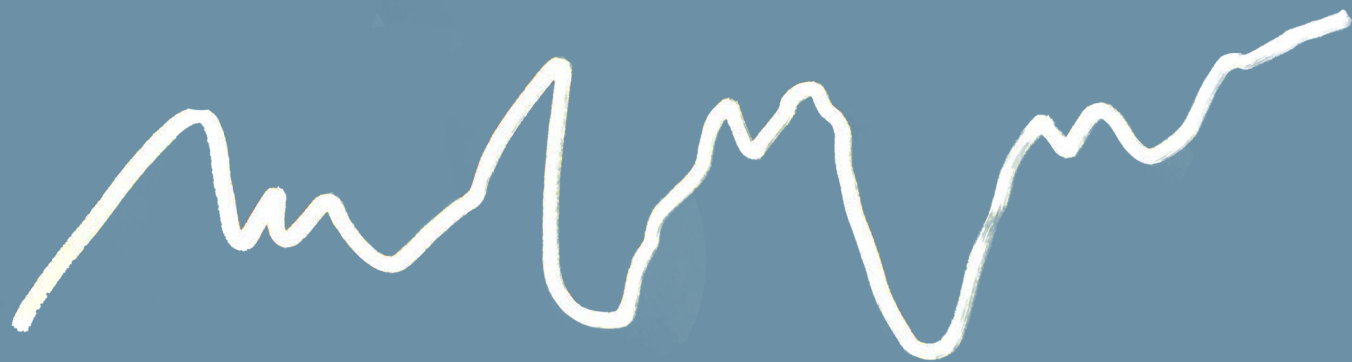
Trajectories provide another type of abstraction you can use to zoom out and judge the success and the overall shape of an experience. They're useful as you think about any type of experience, and I suspect they can be used to navigate through information architectures – and domain models in particular.

Domain modelling is an incredibly valuable way for IAs to think about the design challenges and opportunities that they face. It focuses an IA on thinking about the domain they're designing for. It asks us to think about the classes and types of thing that populate that domain, as well as the properties that those things have.

Most domain models are graphs – nodes and edges – boxes and arrows. Using some of the ideas of Trajectories should enable you to tell stories through the models you create, validating the models you create with domain experts and sharing the insights you've acquired with other designers and team members.

I'm going to be writing in more depth about domain modelling and using Trajectories to interrogate them - 'storytelling' through a domain model is vital to doing it right, and I suspect that Trajectories are another useful filter to apply as we build and tell those stories.

Organisational trajectories



There's one other application of Trajectories that I'd like to introduce - I've called it 'organisational trajectories'.

I've already mentioned that I think 'time' is the often overlooked aspect of lots of IA. Very often we design information architecture imagining we're living in a perfect world. We design vocabularies imagining that they'll be used consistently and will remain relevant and useful. We design workflows and processes based on

the best information we have at the time. We design solutions, but we're not magic and we're not time travellers – so how do you design IA for an uncertain future.

One way to handle uncertainty is to document it. What are the assumptions that your design rests on? What sort of changes might affect your design? What features of the design are dependent on certain factors?

Documenting assumptions and dependencies is one way to make your

design more resilient. With a shared understanding of the features that make a design work, there's a better chance that the design will be used sympathetically and as you'd imagined – giving it the best chance to succeed.

This is another example of why Trajectories are so important. If you've used stories to describe the requirements of the design and the imagined experiences that it will power, there will be a more detailed shared understanding of how the design works.

Trajectories can also be used to play out possible futures. Any design team or organisation can project trajectories into the future to begin to describe the shape of things to come. What possibilities might stand in the future? What threats could undermine your design?

The transitions we've used for trajectories can be used to trace 'organisational trajectories' – but the focus might be a little different.

Beginnings

What is the pre-existing context for this stage of design work? What constraints and historical 'debt' might affect your ability to begin a project or design?

Switching domains

What constitutes the boundaries of the current design? How many channels are required to design a successful solution? Does the current project have responsibility for all dependent domains?

Role

Are roles and responsibilities clearly defined in this project? Who will do what?

Access to resources

All digital products, services and experiences have dependencies. There are also dependencies throughout the design and development process. Understand the foundations which underpin your trajectory.

Interface

How will you communicate the design process, deliverables and recommendations? What communication and documentation will this process and design require?

Encounters

How will you communicate the design process, deliverables and recommendations? What communication and documentation will this process and design require?

Temporal episodes

How long will the current design last? What's the projected life span for the design? Are scheduled reviews required to check the design is still appropriate?

Seams and endings

How will you ensure that the design is resilient? Where are the points of weakness? Which are the most vulnerable dependencies? How might you ensure permanence for the elements of the design which require it?

Who is Dan Ramsden?



I am Dan Ramsden. I design experiences.

Of course, everyone creates experiences. We all do things and lots of the time other people experience those things. I like to think that when I do a thing that someone is going to experience, I do it thoughtfully. Professionally, most of the things I do are on the web. But even though they're therefore often virtual, I still treat them with love and respect.

I'm currently the Creative director for User experience architecture at the BBC. I lead a team of information architects who are committed to making the BBC's tools, content, experiences and platform more meaningful and connected.

I'm based in Salford, at MediaCity UK. And I live just outside Sheffield with a wife, a child and a cat called Rosa.

Ubiquitous language

A ubiquitous, shared language is the most powerful tool any organisation or design team can develop. Eric Evans stresses the importance of discovering and developing a shared language in software development teams, through domain driven design. Information architects stress the importance of creating and facilitating shared meaning through ontology.

Design deals in uncertainty, it's an unavoidable consequence of any design endeavour.

But designers, and IAs in particular have a responsibility to build meaning and trust through the processes they follow. Trajectories are one way to explore meaning and standardise language.

But they're not magic.

Stories only exist while they're being told. Language only has meaning while it is being shared or used. This book provides a quick overview of some of the ideas I've had around Trajectories. But I encourage you to experiment, develop and share your own approaches to using these ideas as you design.



My IA inspiration cards are about the size of a business card and each has a suggestion to help you unlock a solution to the problem you face – ‘slanted strategies’ to help you see differently.

They're now available at
www.danramsdn.com/inspiration



This work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-sa/4.0/>.