

# DATIUM DATA AS MATERIAL FOR CONTEXTUALLY RESPONSIVE DESIGN

Hi, my name's Andrew Fisher and I'm an interaction researcher. Tonight I'd like to talk to you about data as a design material for the next 40 minutes.

Be Responsive, September 8, 2015

Andrew Fisher @ajfisher

# THE WEB WAS ALWAYS RESPONSIVE

What I find most interesting about responsive design is that the web actually started fully responsive - fluidity was inherent in the way it was designed.

## World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

### What's out there?

Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

### Help

on the browser you are using

### Software Products

A list of W3 project components and their current state. (e.g. [Line Mode](#) ,X11 [Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) , [Mail robot](#) , [Library](#) )

### Technical

Details of protocols, formats, program internals etc

### Bibliography

Paper documentation on W3 and references.

### People

A list of some people involved in the project.

### History

A summary of the history of the project.

### How can I help ?

If you would like to support the web..

### Getting code

Getting the code by [anonymous FTP](#) , etc.

## World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

### What's out there?

Pointers to the world's online information, [subjects](#) , [W3 servers](#), etc.

### Help

on the browser you are using

### Software Products

A list of W3 project components and their current state. (e.g. [Line Mode](#) ,X11 [Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) , [Mail robot](#) , [Library](#) )

### Technical

Details of protocols, formats, program internals etc

### Bibliography

Paper documentation on W3 and references.

### People

A list of some people involved in the project.

### History

A summary of the history of the project.

### How can I help ?

If you would like to support the web..

### Getting code

Getting the code by [anonymous FTP](#) , etc.

# THEN WE BROKE IT

But for 20 years after that we basically systematically broke that inherent responsiveness - mostly because of table based design but generally because we made things work for the monocultureis of the day - first Netscape and then IE.

# THEN WE FIXED IT



Then mobile came along, and we spent a long time fixing stuff that was only really broken because we stuffed it in the first place and over the last 2 years we have all become converts to the idea of responsive design. So the sum total of our last 20 years of effort has been to largely bring us back to the point where we started, albeit with better typography and fewer beavers in hardhats welding shovels.

Continue on, we come back to the light soon

That's me being a little  
disgenerous obviously,  
because we do have a  
better platform for design  
than we had 20 years ago -  
but the point is, we broke  
things due to our focus on  
execution for what was in  
front of us, and didn't think  
of what was coming next  
even when it was apparent.  
So in the last couple of  
months we've probably  
transitioned past the point  
of no return on mobile  
though we're not quite sure  
if we're there or not. More  
traffic is now coming from  
mobile web than desktop  
web and if it hasn't quite  
made it yet then we are  
damned close. Basically if  
you don't have mobile in  
your business now then it's  
too late. The question is  
really one of how bad it's  
going to be for you. With an  
audience like you all that's  
probably not a problem but  
it will be for others.

# LOSING FOCUS ON WHAT MATTERS

However even if you are fully mobile and you've done a good job I think many organisations have missed a critical feature about what underpins what's happening in behaviour right now because we're a little fixated on screens.

CGA  
320 x 200

WVGA  
800 x 480

VGA  
640 x 480

SIF\*\*  
384 x 288

VGA (NTSC\*)  
640 x 480

PAL\*  
768 x 576

SVGA  
800 x 600

XGA  
1024 x 768

WSVGA  
1024 x 600

1024 x 576

1152 x 768

1366 x 768

WXGA  
1280 x 768

WXGA  
1280 x 800

1440 x 900

1600 x 900

1440 x 960

WSXGA+  
1680 x 1050

HD 1080  
1920 x 1080

2K‡  
2048 x 1080

WUXGA  
1920 x 1200

UWUXGA  
2560 x 1080

WQHD  
2560 x 1440

21:9

16:9

17:9

8:5  
(16:10)

5:3

3:2

4:3

5:4

4K‡  
4096 x 2160

# WE'RE STILL (MOSTLY) OBSESSED WITH SIZE

I think we're largely through the size debates and should we support mobile and we're now in the mode of "I'm designing for a small screen or a large screen". But we're also still a bit stuck in the "look, see how this all refows, see - it works". That's awesome and all but we're missing the underlying behaviour going on here. These devices have allowed computation to move into different contexts and as humans we are changing our behaviour as a result of this change in tech,

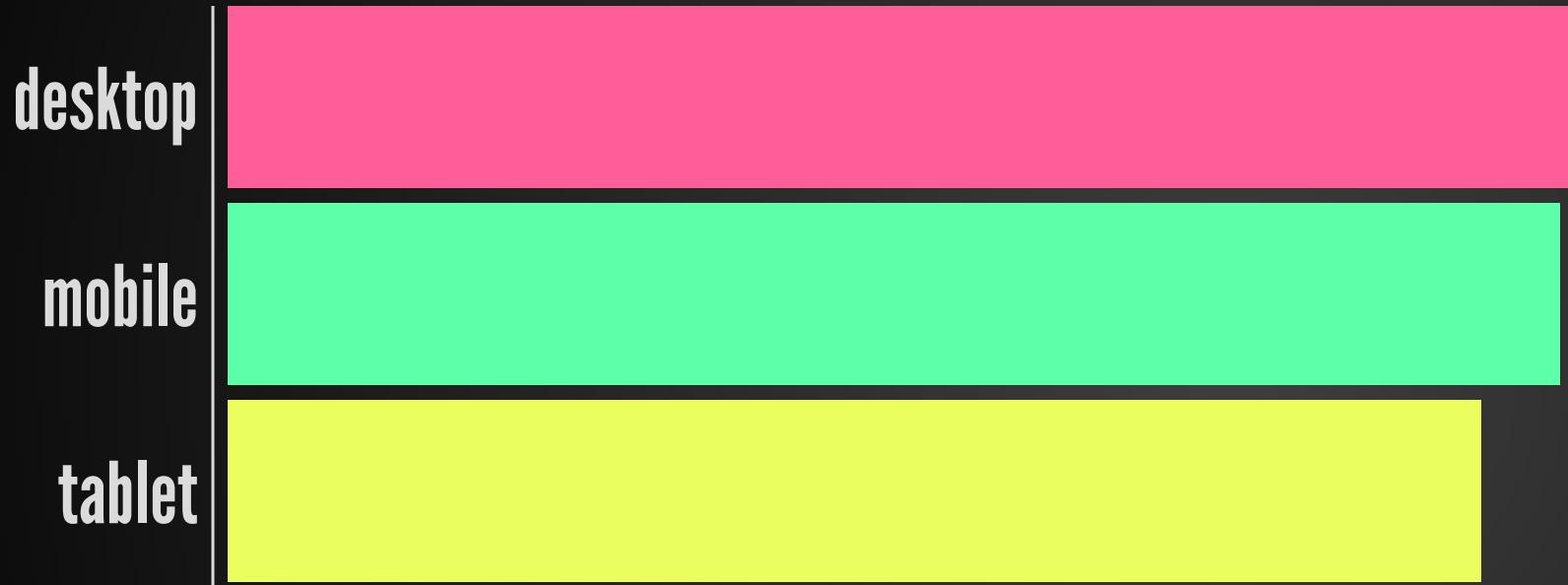
and PAL/SECAM are analog video standards with no fixed horizontal resolution. The resolutions listed here for 4:3 aspect ratio assume square pixels, but the actual horizontal resolution (in non-square pixels) ranges from 320 (VHS/Betamax) to 720 (DVD) for both standards.  
ough computer industry defines SIF as 384x288 for PAL countries (for NTSC countries, it is equivalent to 352x288), MPEG-1 defines it as 352x288 (CIF) or 360x288 (for NTSC: 352x240 or 360x240).  
olutions depicted here for 16:9 widescreen NTSC and PAL/SECAM assume square pixels, but the horizontal resolution (in non-square pixels) ranges from 520 (PALplus) to 720 (DVD).  
ough Digital Cinema System specifies 2K at the depicted resolution in square pixels, in some cases it can assume non-square pixels and go as high as 2048 x 1536. Aspect ratio 17:9 is approximate.

# NEW FORMS OF COMPUTING



Yes, a phone may be a small screen but it's a small screen that is not on a desk in a fixed location. It's a small screen attached to an incredible computer that can access just about any information in the world within a few seconds and a taps on the screen. It's also a small screen that we can put away and pull out at a moment's notice. With the transition to mobile web overtaking desktop web the predominate interaction method is probably now touch, it could easily also be voice but it sure as hell isn't sitting down with a keyboard and mouse any more. We have moved from sit down computing to wherever you are computing and that's very very different in terms of the behaviour it affords.

# MOBILE'S DIRTY SECRET



And this is the thing that we as an industry have missed and it's already hurting and will hurt even more. Take a look at this. This is some aggregated data I pulled together which looks at normalised conversion rates by device class. You can see how poorly mobile performs, only about 2/3 as well as desktop and tablet isn't much better. Even the best in mobile only just approach what would be considered "below" average on desktop. I've seen data from adobe that looked at every analytics client they have and their numbers are the same (and what spurred me to look at ours). Mobile traffic is going up yet ability to turn that traffic into actual outcomes is poor. Yes, you might have upped your conversion by 5 times from what it was on mobile but when the majority of your traffic comes from mobile and it's performing worse questions are going to be asked and I can tell you CEOs and CMOs memories are short.

# RESPONSIVE DESIGN SHOULD BE ABOUT ME



Responsive design is still all about my device, not about me and what I want. We spend more time reacting to whether I have a mobile and what orientation I'm holding it or what fancy new framework is in fashion this month than we do thinking about the service the web application is delivering to me and the context I'm in when I'm using it.

# DATATIUM

Da

To truly create interesting experiences. To truly take mobile from being just a different screen. To truly realise the idea of a web that is responsive, we need to think beyond pixels and start thinking about context and behaviour. Tonight, I want to talk to you about contextually responsive design and how data can be used to facilitate it. In my work I think about data as a material - something I call datatium, and I'd like to share this with you so you can go away and start designing responsive experiences as well.

# AGENDA

To do that we'll cover off these three areas - What is contextual design - Data as a material and how we get it - Creating contextually responsive experiences  
Let's get cracking

- 1 What is contextually responsive design?
- 2 Using data as a material.
- 3 Creating contextually responsive experiences.

# WHAT IS CONTEXTUAL DESIGN?

Contextual design embraces the chaotic nature of behaviour. We think about intent a lot - what is this person trying to achieve, rather than does this design layout the right way on this device. These experiences constantly evolve and are shaped by the data available to the experience you're creating.

# A MULTITUDE OF FACTORS

- Explicit data (what do I tell you?)
- Implicit data (how do I behave?)
- Extrinsic factors (what can you infer?)

Contextual design uses a combination of explicit information - what they've told me such as preference information. We use implicit data such as behavioural information but we also use external information as well - information we can infer or derive from other sources and factors. You need all of these components in order to be able to create a contextually responsive experience.

# MISSING ONE PIECE



A Heads up on your data use for 61450403 [REDACTED]. You have less than 1 GB of your monthly data allowance left. Your billing month ends on 6/11/2014 Check your usage in MyVodafone by clicking <http://vfau.co/usage>

6 Nov

## Message details

Type: Text message  
From: AUVodafone  
Sent: 6 Nov 11:07  
Received: 6 Nov 11:07

Here's an example of an okay experience. This came from vodafone warning me I was approaching my data limit and when it would start again. There was one point of failure with it and the eagle-eyed amongst you may spot it. It hits two of the three factors: - Explicitly I've told vodafone to contact me if I approach a threshold. So I've expressed preference - Implicitly I've been using my data allowance so the system knows I've exceeded that threshold. This is the point the message gets sent. The missing part is the extrinsic factors - what else can be derived from knowledge about me or my behaviour is missing and is what turns this into only an okay experience rather than a great one. We'll return to this example later and I'll show you how to make it better.

# CONTEXTUAL DESIGN IN ACTION

So here's what a contextually responsive experience looks like in my opinion and how seamless it can be from an experience perspective. Here's a clip from the first avenger's movie.

# CONTEXTUAL DESIGN IN ACTION



Notice here how the HUD changes from a general display on what's going on to full threat tactical display in a heartbeat. Stark doesn't activate it at all and it becomes perfectly tuned to the tactical situation around him, responding contextually to the threats nearby. Most interestingly, this interface was designed with consultation from an A-10 pilot who worked with the designer on the UI and their focus on was on contextual tactical information - notably because that's NOT how current heads up systems work in military aircraft. Now clearly this HUD doesn't exist in anything other than a fictional sense but this completely sums up how I want the web to behave every day - and I'll settle for this without the flying suit of armour and pulse cannons in my hand.

# ODD EXPERIENCES



(CC) Flickr Joe Goldberg

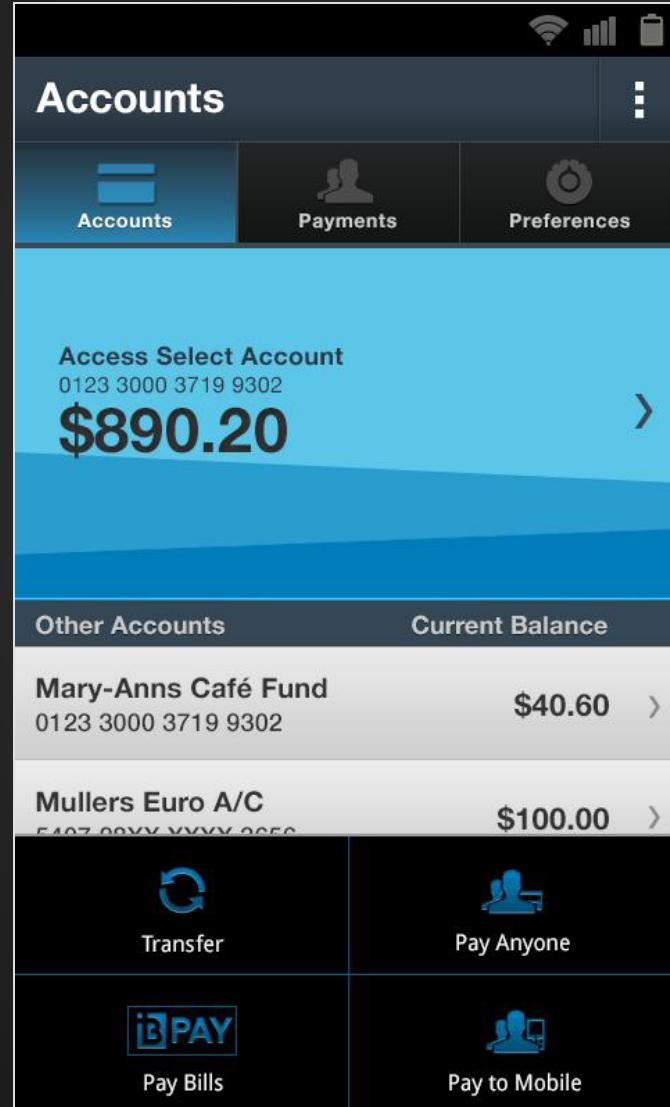
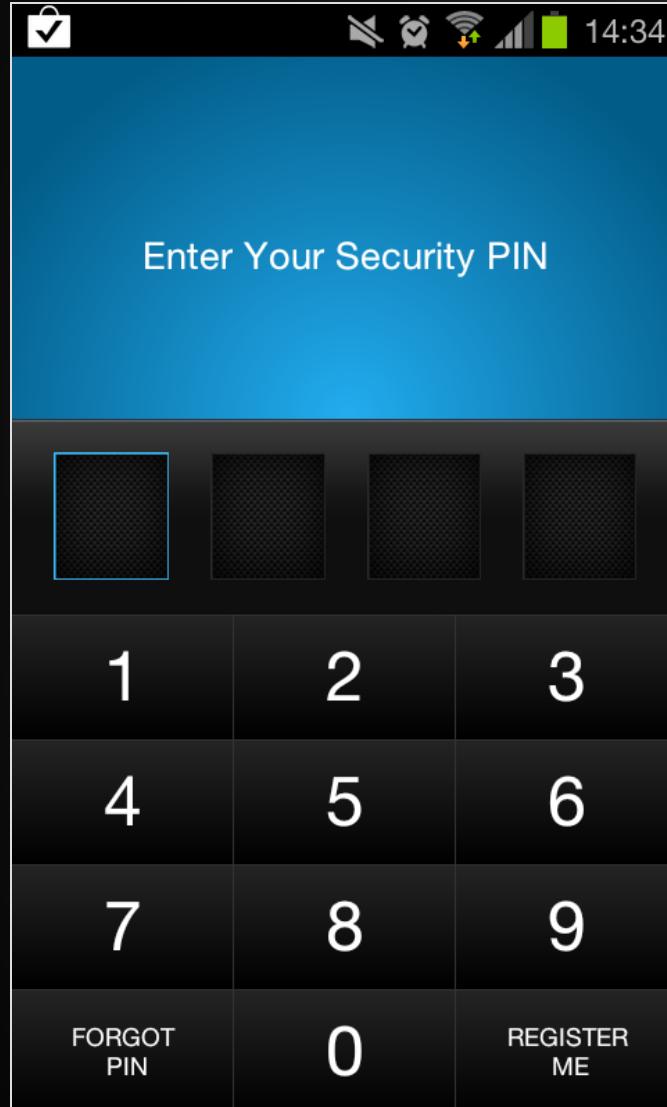
In a forum like this everyone would be practicing the core responsive design techniques. But I bet you're still coming across some issues - probably more frequently now everything you're doing is inherently responsive. Here's the situation. You design a wonderfully responsive site everything reflows beautifully - it's a work of responsive art. But you then use the web application on mobile - or maybe you go from mobile to desktop - and something just "feels" wrong. You can't put your finger on it, but there's something odd about the experience when you move from one device to another. Instagram going from mobile to desktop feels like this - it's just not right. Has anyone experienced this? Good - a couple.

# CONTEXT SHIFT

What you are noticing is the effect caused by context shift. Sometimes in our services the shift to a different device causes the whole idea of how you use that service to change fundamentally. By shifting the device interaction we've actually forced an entirely different context. This is where we need to start accounting for how context is driving the requirements rather than simply device. Here's an example of this.



# CONTEXT DRIVES REQUIREMENTS



I've been told anecdotally that when ANZ came to build their GoMoney application for mobile devices it was based around the insight that the majority of users logged in to the existing web application on their phone, looked at their account balance and exited again. As such in their mobile app they put it front and centre and then tacked on making a payment as well as that was the number two task people did. Context drives need and requirements. When I'm out and about and I'm really mobile, I don't need the full capabilities of the entire ANZ online banking platform. I just want to see whether I've been paid and whether I have enough money to buy that new game or not.

# IS THIS A BROWSER?



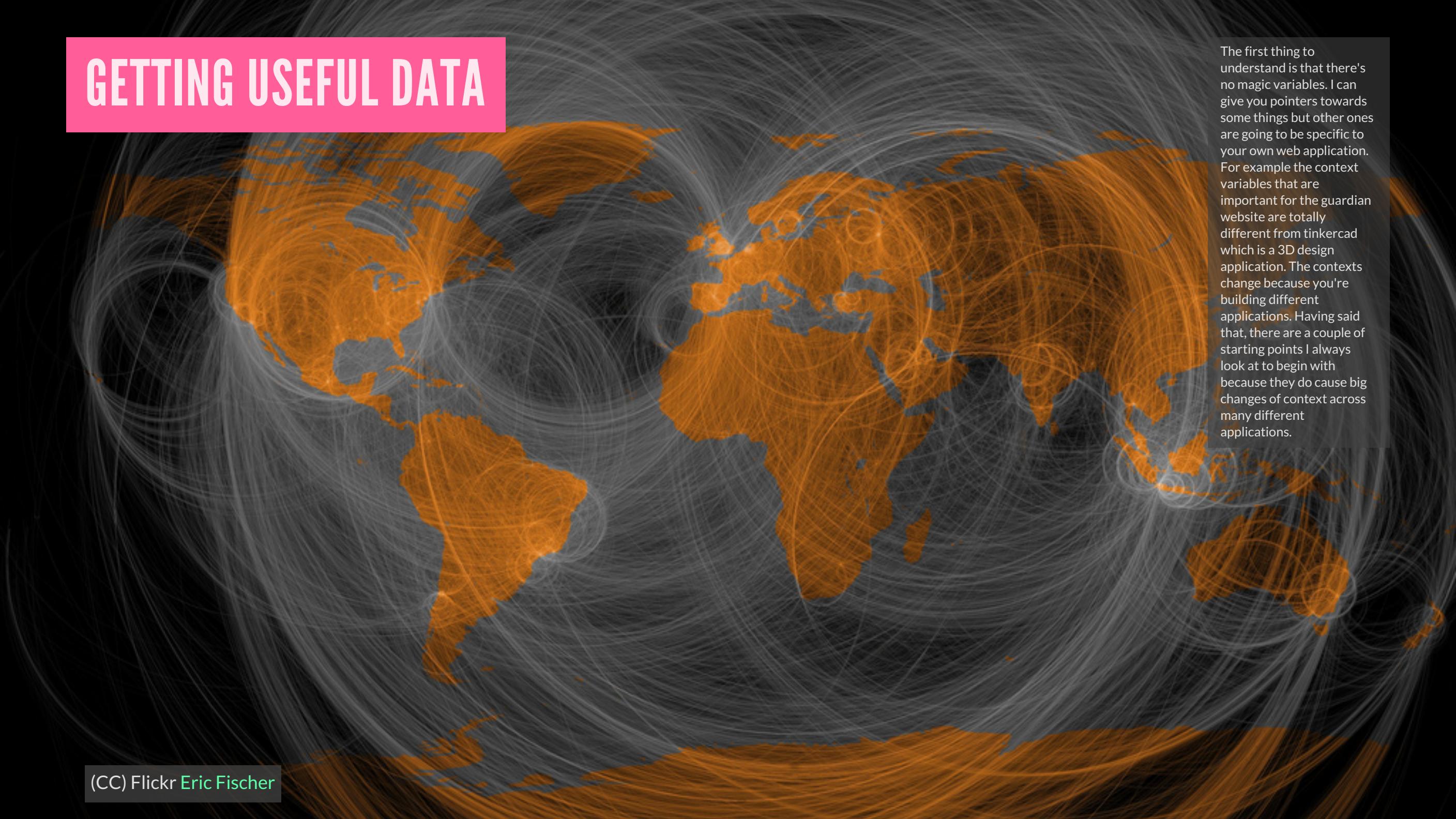
This is my final point for the moment about why you should be thinking about context. Designing this way is important because our notion of what is a browser is starting to break down. Our services are starting to transcend the browser we've become accustomed to after these 25 years. The web is scalable. Far more scalable than a thousand apps installed for every person on every device they own. But the idea of how we experience that web is going to change fundamentally over the next few years as computing becomes more embedded into our environments and gets closer to our bodies.

# DATA AS MATERIAL



Now you have a bit more of an understanding about what I mean when I talk about context and contextual design, I'll give you some understanding about how we get to use it. The critical component is data and specifically, starting to think about how we use data as material during our design process - just like we do any other component of design such as type, animation, colour and light. I'll start with what things matter, how we get that data, and how we plan our experiences around this.

# GETTING USEFUL DATA



The first thing to understand is that there's no magic variables. I can give you pointers towards some things but other ones are going to be specific to your own web application. For example the context variables that are important for the guardian website are totally different from tinkerad which is a 3D design application. The contexts change because you're building different applications. Having said that, there are a couple of starting points I always look at to begin with because they do cause big changes of context across many different applications.

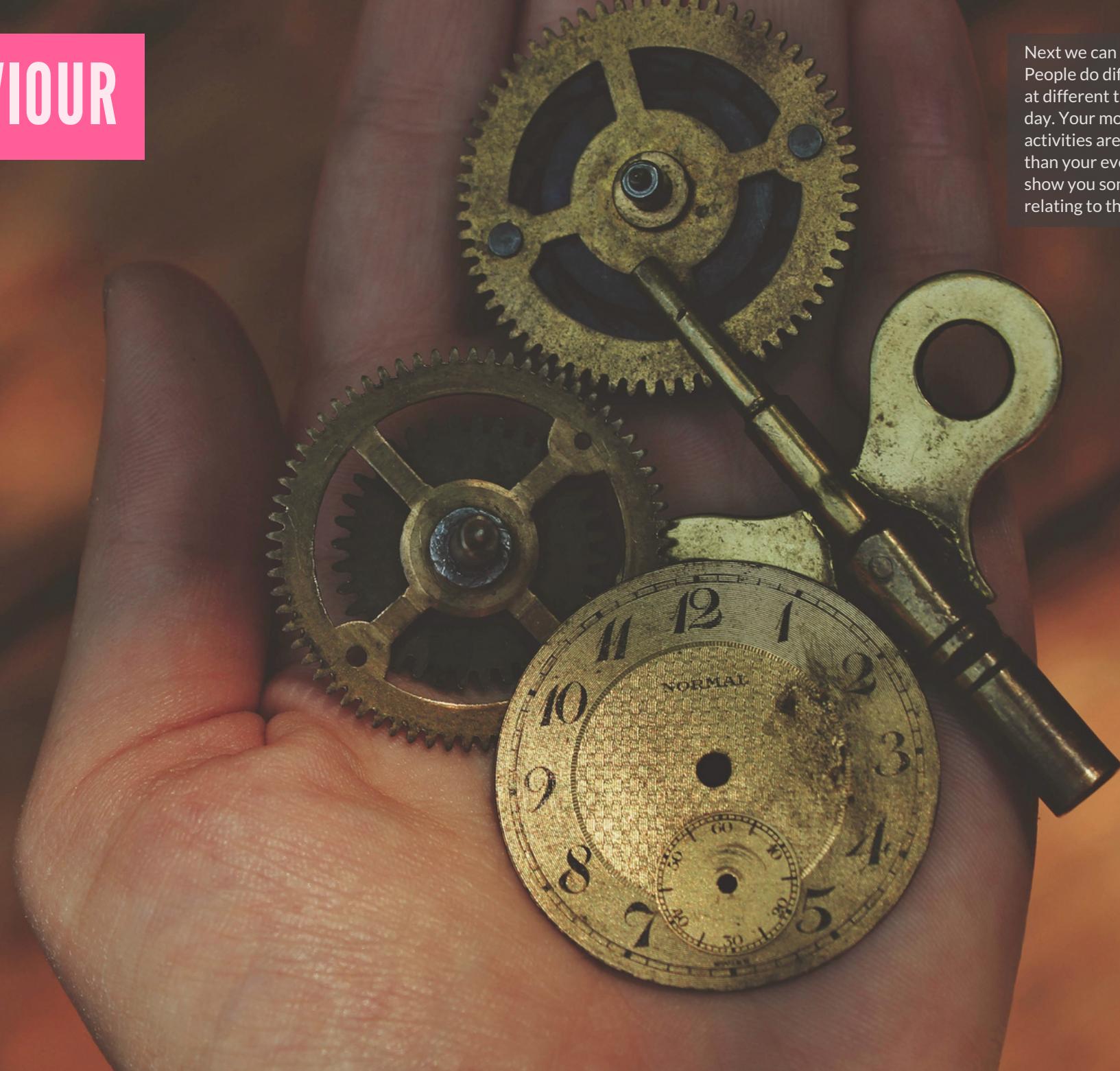
# DEVICE SHAPES BEHAVIOUR



This is the obvious place to start and where all of you are right now. The reality is that this only really gives you a sense of what technical capabilities a person is attached to. However, in some cases device might be really important. For example if I'm a telco like telstra, I can determine the rough age of your device and infer whether you might be looking to upgrade. Similarly if you're an android user on the iPhone microsite at Apple you could highlight some of the benefits of iOS and how you can still use all your usual google apps.

# TIME SHAPES BEHAVIOUR

Next we can look at time. People do different things at different times of the day. Your morning activities are very different than your evening ones. I'll show you some data relating to this in a minute.



# LOCATION SHAPES BEHAVIOUR

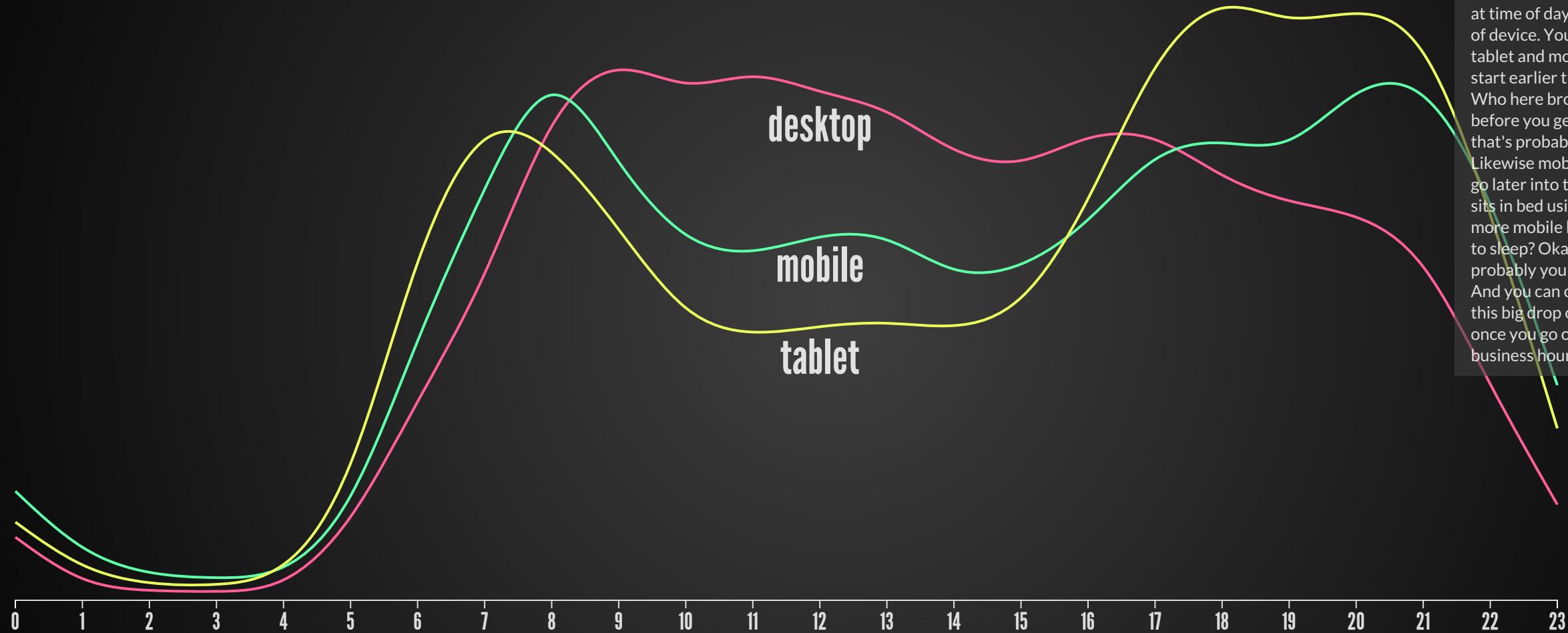
We can also consider location - is the person moving or stationary? You can infer location but you can also ask a user for their location if you think it will make it a better experience. Does the person's location overlap with known places like a train station, a train line or say you were a retailer one of your stores (or close to it). These are just a few points that can create a whole bunch of different contexts that the user may be in.

# CONTEXT SHAPES BEHAVIOUR

(CC) Flickr Nana B Agyei

If I visit a site on my phone at about 8am every morning through the week and I'm moving chances are I'm on my commute. This is very different to me using a tablet in landscape mode at 9pm where I'm connected via a high speed wifi link from a static location.

# TIME AND DEVICE



Here you can see some data that was composed as an aggregate of visitation across many clients I work with. This is a few months old now but you can see some clear behavioural differences when you look at time of day against class of device. You can see that tablet and mobile devices start earlier than desktops. Who here browses in bed before you get up? Okay so that's probably you. Likewise mobile and tablet go later into the day. Who sits in bed using a tablet more mobile before going to sleep? Okay so that's probably you up this end. And you can obviously see this big drop off of desktop once you go outside of business hours.

# CONTEXT VARIABLES

Context starts creating a significant number of combinations of variables and this can quickly spiral out of control but start by just considering device, location, time and speed and you'll get a surprising number of contexts from them to consider in your design.

# PLANNING FOR THIS DATA IN OUR DESIGNS

- When can I tell what someone is going to do?
- Why did they do that at that point?
- What else could I know that will help the user to their end goal?

When I'm working on an experience, I'm constantly looking for data about what people are doing and what more can I know that will help take this experience from good to great. There's always pragmatism involved obviously, but we can use these variables to start the process. I often start by just asking these questions and go into the process from there.

# CREATING CONTEXTUALLY RESPONSIVE EXPERIENCES

Now you understand what data is important and how to get it it's time to look at the types of experiences you can go and create. Unfortunately at the moment the examples are a bit thin on the ground because we're only starting to think about this type of design. My general approach is to consider desire lines, links between the physical and the digital and response to failure.

# UNDERSTANDING DESIRE LINES

A desire line is when you can see from someone's behaviour what they are trying to achieve before they get there. So We can ask questions about a user's intent. What are they after? Are they exhibiting any sorts of behaviour where we can start predicting what they are ultimately wanting to do? Are they using our system in a particular way that allows us to tune the experience for that.



# SURFACING CONTENT / FEATURES CONTEXTUALLY

If we understand a desire line properly then it means we can surface content or features contextually without the user even being aware of it. Google NOW is probably the best example of this. Particularly to do with travel.

# GOOGLE NOW



## Trip to Bengaluru

16 – 22 Sep



- Melbourne to Singapore – VA 5516**  
16 Sep, 00:05 – 06:00

- Singapore to Bengaluru – VA 5704**  
16 Sep, 07:00 – 08:50

- Bengaluru to Singapore – VA 5533**  
21 Sep, 23:05 – 22 Sep, 06:10

- Singapore to Melbourne – VA 5519**  
22 Sep, 10:45 – 20:10

Being able to show just as you need it the information to do with your flights a day before to remind you and then just before your flight, showing gate and departure information is a very good example. Other areas of google now are a little weak - such as suggesting that every place I search I want to go to afterwards - however flight details are exceptionally well handled. To the point that I just archive all my travel details now as soon as they arrive as I know Google Now will sort them out when I need them. This is what contextual content looks like - we just need that surfaced to the browser now too.

Related messages

# LINKING TIME AND SPACE



Time and space go together almost always. It's rare I need to worry about my shopping list when I'm not at the supermarket for example. Being able to link someone's location to the time they are in is a powerful mechanism for responsive design. If you trap details about these two things then you can surface actions or improve the design once you know where and when someone is at.

# INBOX

Re: [johnny-five] refactor LedControl (#665)

[VIEW ISSUE](#)

R

**Rick Waldron** to rwaldron/johnny-five, me ↗

Here's a copy of my reply on gitter, just to keep it all on record...

| is there a reason why 7-seg stuff and matrix stuff are so interwoven together?

The original `LedControl` class was actually an "almost" 1-to-1 port of the `LedControl` C library. That Segment and Matrix displays, specifically with the MAX7219 and MAX7221 chips. The port was done when Johnny-Five existed (much like the Wii classes), which obviously predates the component = controller paradigm. I attempted to update the class, but was wary of breaking changes, which resulted in a sub-par update. I ended up taking ownership of the class and do a comprehensive re-write (similar to [@dtex](#)'s work in Motor or [Sensor](#) sensor classes).

| I get that you can control them using the same chip but just wondering why the controller interface is so interlinked when the use cases are so different?

You're not alone here—this was sort of my motivation for making the aliases `LedControl => [ Led.Digits, Led.Matrix ]`. I wanted to treat them as different entities so they could have documentation that clearly defined them as unique component classes.



Snooze until...

Tomorrow Wed, 8:00 AM

Next week Mon, 8:00 AM

Someday

Pick date & time

Pick place

Another good example of this is Google's Inbox. If you haven't used it yet and you're a gmail user I highly recommend you request access as this is one of the strongest examples of contextual design in existence at the moment.

The main feature as shown here is the ability to snooze a message until a time such as later, to a place such as home or a combination of both. Being able to remind you to take action on something when you're in the right time and place to do it is very interesting and seeing how this develops beyond just email will be good to watch.

# DESIGNING FOR FAILURE

we need to consider designing in methods to allow people to correct a mistake. If you embrace the chaos of contextually responsive design you ARE going to make mistakes in what you put to the user but they are good opportunities to learn. So in your design process you need to do two things.

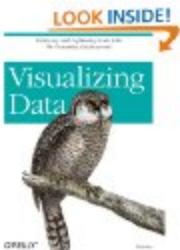
# BE TRANSPARENT

These recommendations are based on [items you own](#) and more.

view: [All](#) | [New Releases](#) | [Coming Soon](#)

The first is to make it transparent to the user about WHY they are being provided with the information you're providing them. Sometimes this can be hard to articulate easily so you might need to work at it. Amazon recommendations work quite well here because it tells you what you purchased or looked at in order to get the recommendation so you can see how it's affecting what you're seeing.

1. [Visualizing Data: Exploring and Explaining Data with the Processing Environment](#)



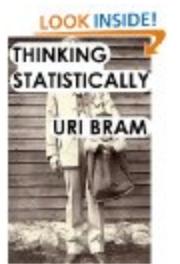
by Ben Fry (December 17, 2008)  
Average Customer Review: ★★★★☆  (18)  
Auto-delivered wirelessly

**Kindle Price:** \$17.59

I own it  Not interested  Rate this item

Recommended because you purchased **Processing** and more (Fix this)

2. [Thinking Statistically](#)



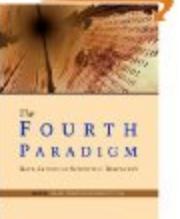
by Uri Bram (October 22, 2011)  
Average Customer Review: ★★★★☆  (9)  
Auto-delivered wirelessly

**Kindle Price:** \$2.99

I own it  Not interested  Rate this item

Recommended because you purchased **Data Jujitsu: The Art of Turning Data into Product** and more (Fix this)

3. [The Fourth Paradigm: Data-Intensive Scientific Discovery](#)



by Tony Hey (October 16, 2009)  
Average Customer Review: ★★★★☆  (6)  
Auto-delivered wirelessly

**Kindle Price:** \$0.99

I own it  Not interested  Rate this item

4.



[\*\*The Little Book of BIG DATA, 2012 Edition\*\*](#)

by Noreen Burlingame (February 14, 2012)

Average Customer Review:  (5)

Auto-delivered wirelessly

**Kindle Price: \$3.95**

# CAPTURE FAILURE DATA

The second is allowing the user to correct things. Facebook ads are a good example of this as they let you fix a problem and in fact they look at it as an opportunity to collect some additional data about WHY it was no good for you which helps them with their targeting ongoing. Don't pass up the opportunity to learn from your mistakes. Contextually responsive design is inherently fuzzy so you will make mistakes. Always remember to let the user take action to correct things if needed.

# DESIGNING A CONTEXTUALLY RESPONSIVE EXPERIENCE



A Heads up on your data use for 61450403 [REDACTED]. You have less than 1 GB of your monthly data allowance left. Your billing month ends on 6/11/2014 Check your usage in MyVodafone by clicking <http://vfau.co/usage>

6 Nov

## Message details

Type: Text message  
From: AUVodafone  
Sent: 6 Nov 11:07  
Received: 6 Nov 11:07

Remember this? The SMS I got from Vodafone. If you were looking closely you would have noticed the date. I had 1GB left on the day I was going to switch to my new billing month and I got a warning about it. To make this experience better Voda should be checking the trajectory of my usage and predicting when I'm going to fall short. If I'm highly unlikely such as having 25% of my monthly plan left with only a day to do then don't even warn me - it's burning contact for no reason. Likewise though, if I'm going to fall short by a tiny amount then make a call on just providing grace. Why bother charging for what may be a 1% overage for a month. Indeed send a text message like this:

# A BETTER MESSAGE

Now that's a more interesting experience for the end user I think you'll agree.

*"You just went over your monthly limit. Don't worry - anything you use within reason is on us until tomorrow when you reset."*

# SUMMARY

- Design for context from the outset
- Use location, time, speed, device to start
- Learn from failure

To finish off then. We've looked at what is contextually responsive design, how to get the data we need to facilitate it and looked at some examples of how to do it in practice. If you take nothing more away, these three things will serve you well. Build context into your process from the start, work with some core variables and build up familiarity and make sure you learn from your fails.

Thanks very much for listening. All my details are here, including the link to the html slides and if you have any questions I'd be happy to answer them now, after or ping me via twitter.

# DATATIUM - DATA AS MATERIAL FOR CONTEXTUALLY RESPONSIVE DESIGN

Be Responsive, 8 September, 2015

Andrew Fisher @ajfisher

Slides: <http://datatium.ajf.io/>