Interusability:
designing a coherent system UX

Experience Design for the
Internet of Things

Claire Rowland - @clurr
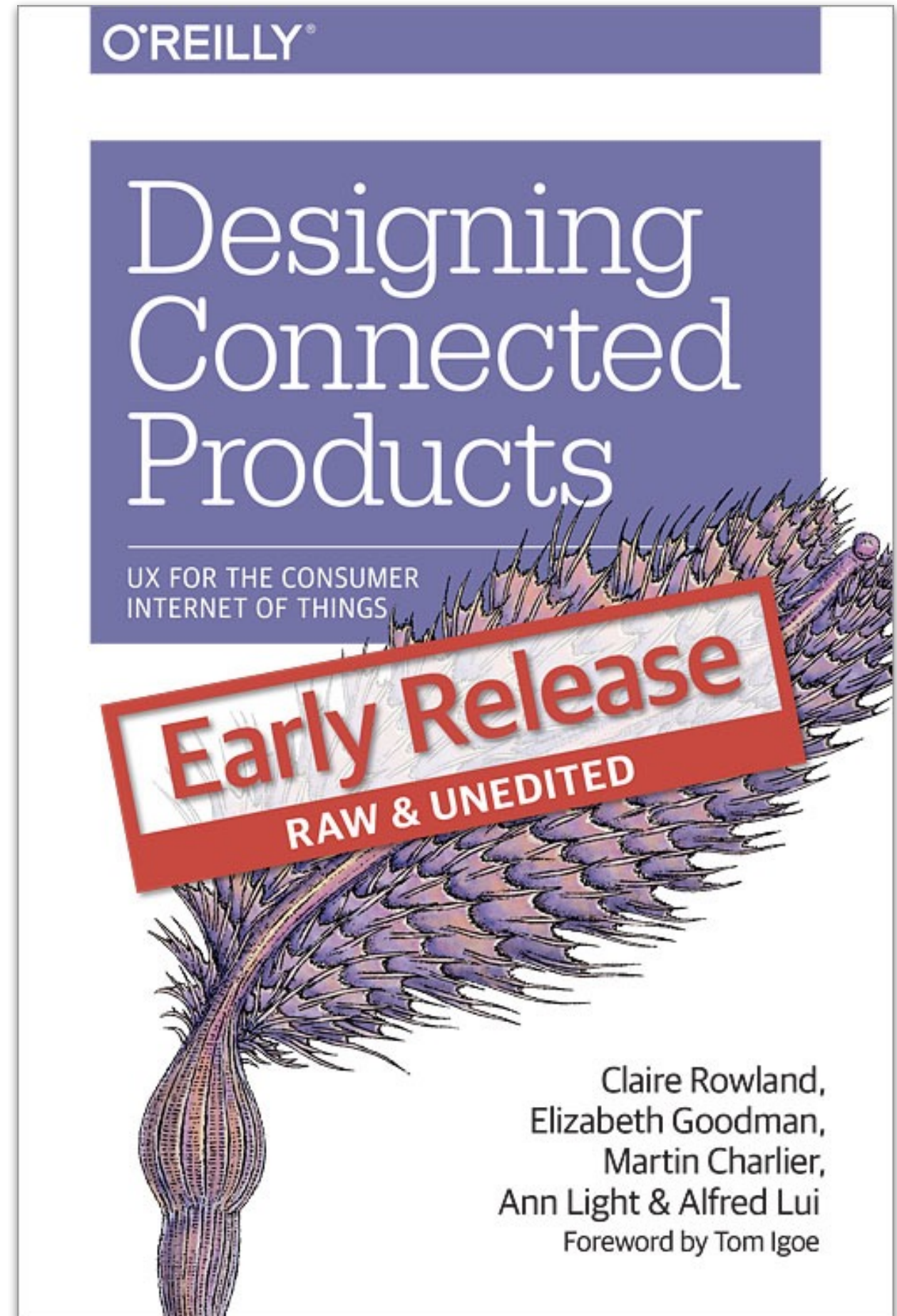
Hello :)

- Independent UX and product consultant
- Lead author: "Designing Connected Products: UX design for the consumer internet of things" (due end of May 2015)

UX for IoT: not just UI & industrial design

Images: Nest

Users have to understand systems

Images: Withings

- Functionality and interactions are distributed across multiple devices, often with different capabilities
- Systems are inherently harder to understand
- We are much better at thinking about things than about relationships between things
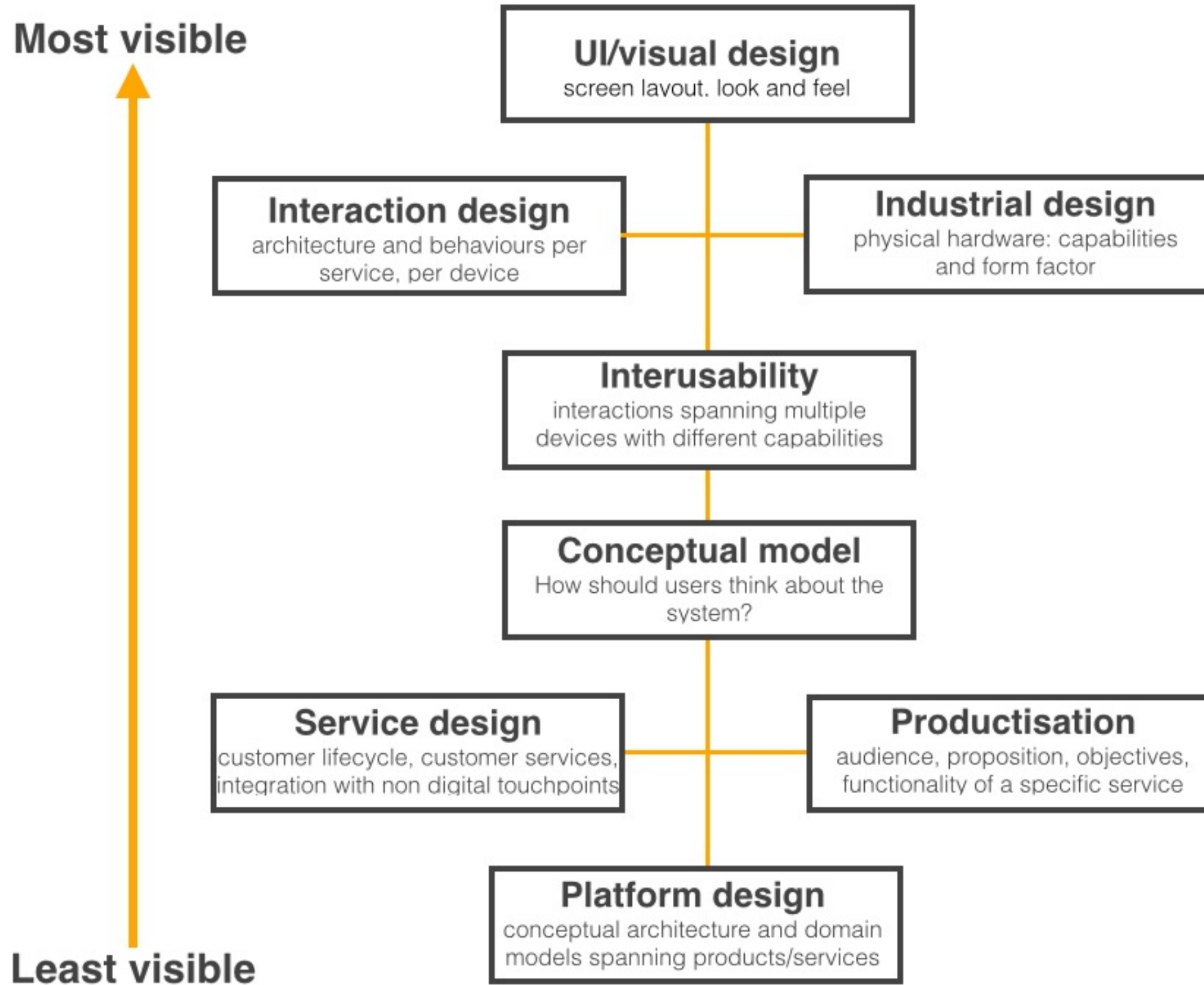
We don't (yet) expect Things to behave like the Internet

The average consumer is going to find it very strange when objects take time to respond, or lose instructions.
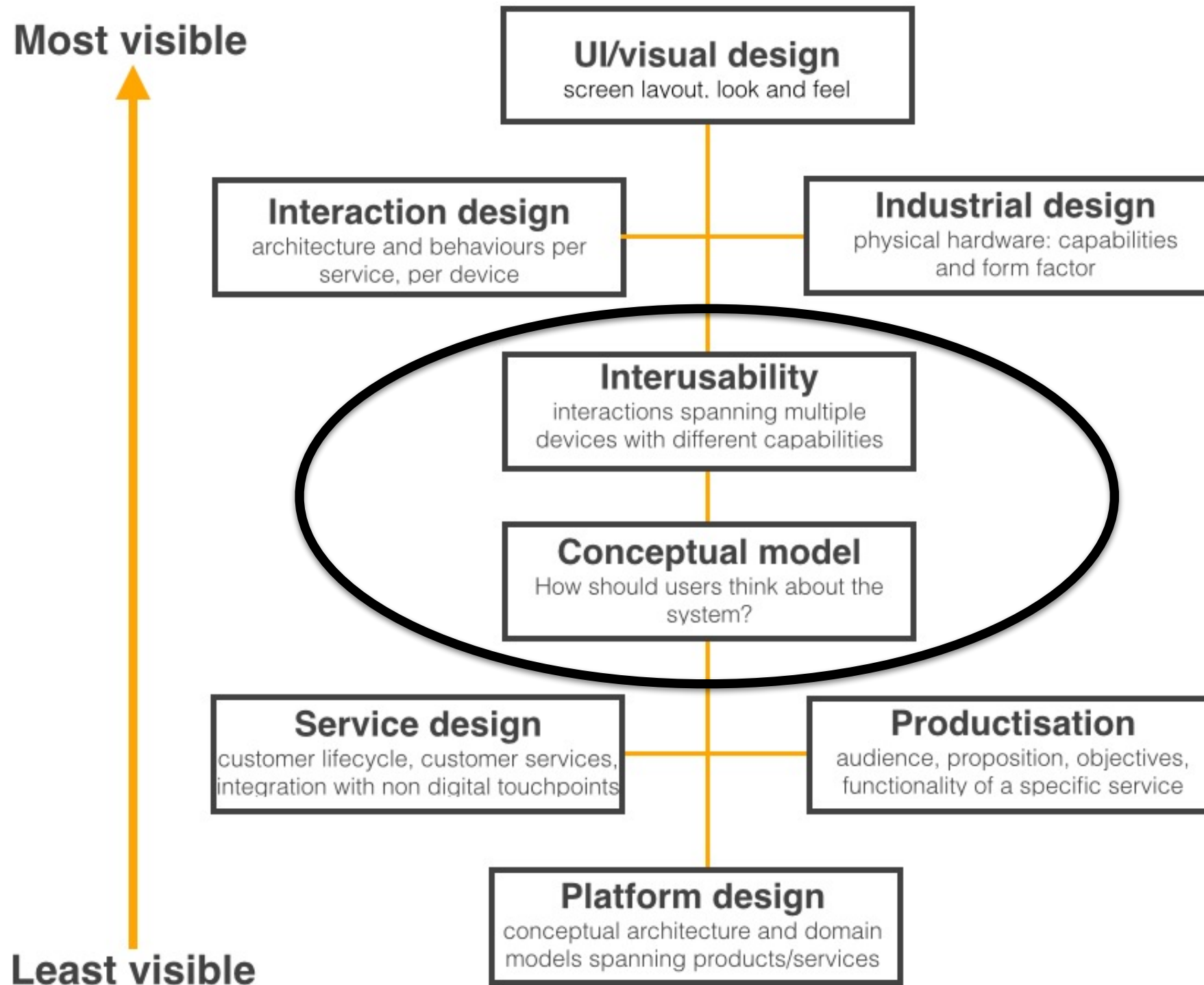
Image: Nissim Farim

Facets of IoT UX

Most visible

Least visible

UI/visual design
screen layout, look and feel

Interaction design
architecture and behaviours per
service, per device

Industrial design
physical hardware: capabilities
and form factor

Interusability
interactions spanning multiple
devices with different capabilities

Conceptual model
How should users think about the
system?

Service design
customer lifecycle, customer services,
integration with non digital touchpoints

Productisation
audience, proposition, objectives,
functionality of a specific service

Platform design
conceptual architecture and domain
models spanning products/services

Today we'll look at this part:

Most visible

UI/visual design
screen layout. look and feel

Interaction design
architecture and behaviours per
service, per device

Industrial design
physical hardware: capabilities
and form factor

Interusability
interactions spanning multiple
devices with different capabilities

Conceptual model
How should users think about the
system?

Service design
customer lifecycle, customer services,
integration with non digital touchpoints

Productisation
audience, proposition, objectives,
functionality of a specific service

Platform design
conceptual architecture and domain
models spanning products/services

Least visible

Conceptual models

Understanding how it works

Image: Instructables

Image: How It Works Daily

Value proposition

Conceptual model

Interaction model
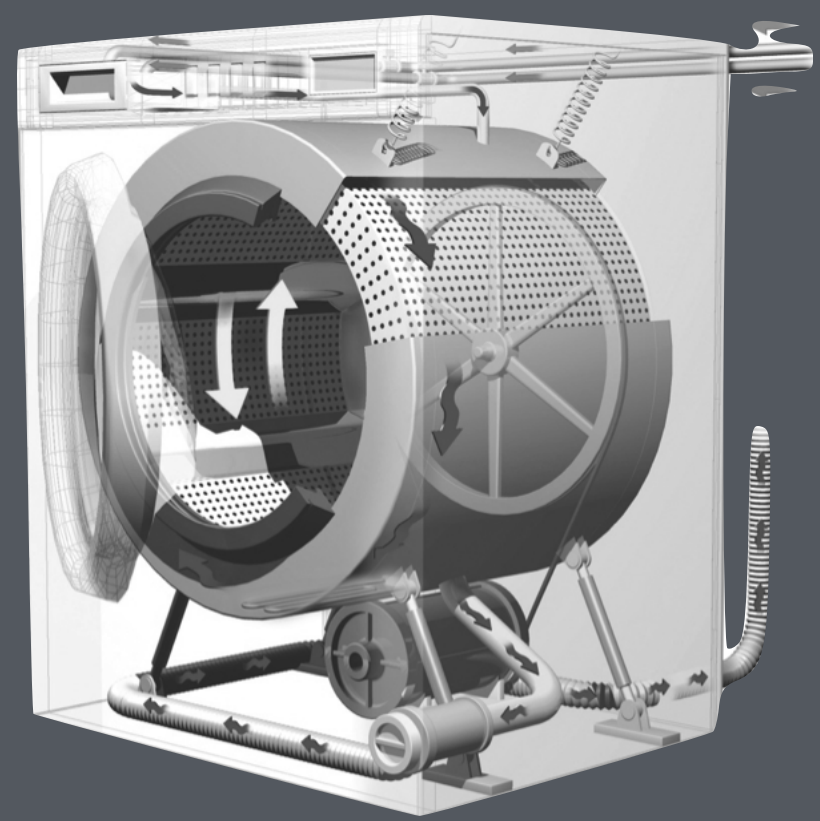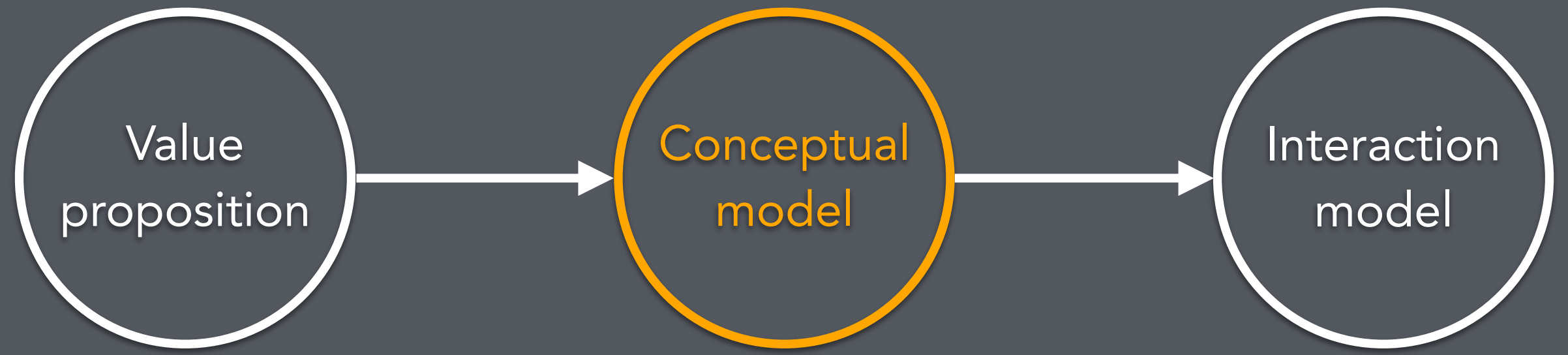
What does it do?

How does it work?

How do I use it?

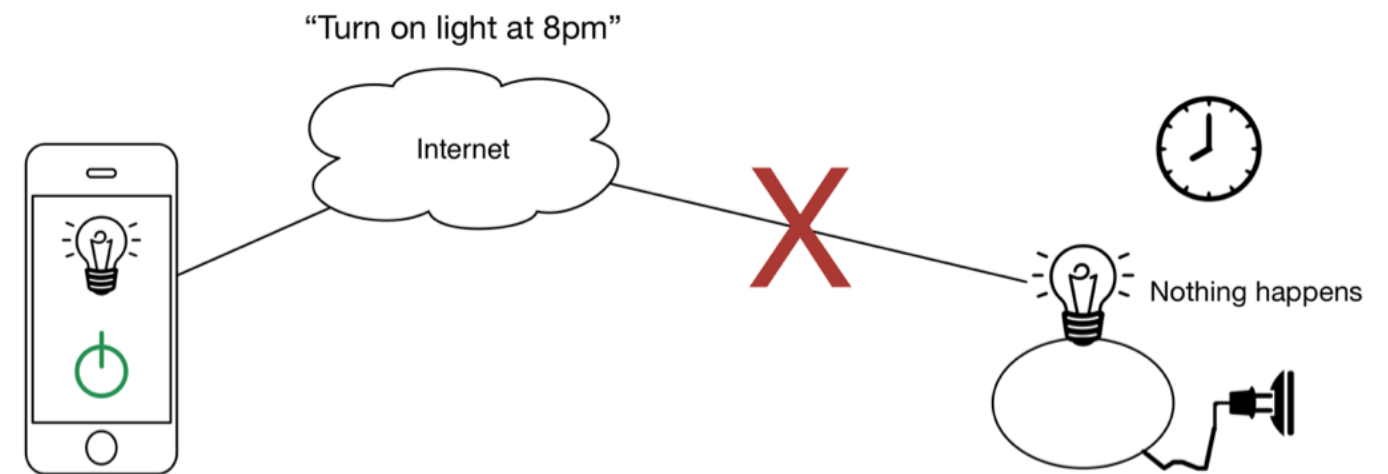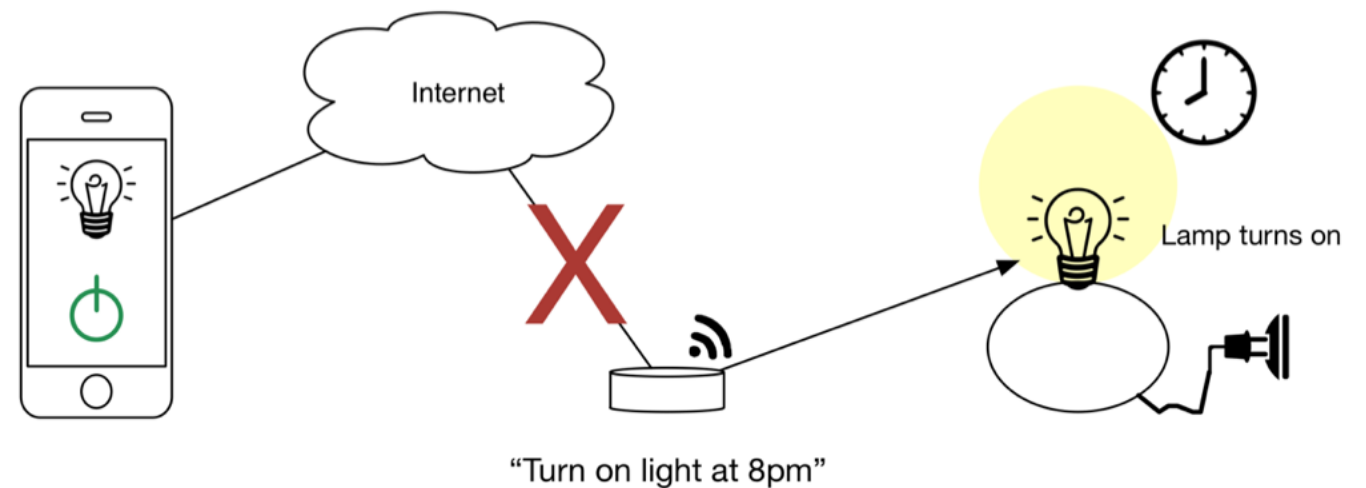# Non-connected products are often conceptually quite simple

Bulb lights up

Press switch for electricity

Connected products are more complex

Sends instructions to internet service

Passes instructions to gateway, may run rules

Internet

Passes instructions to lightbulb, may run rules locally

Talks directly to gateway

Bulb lights up

Product images: Philips
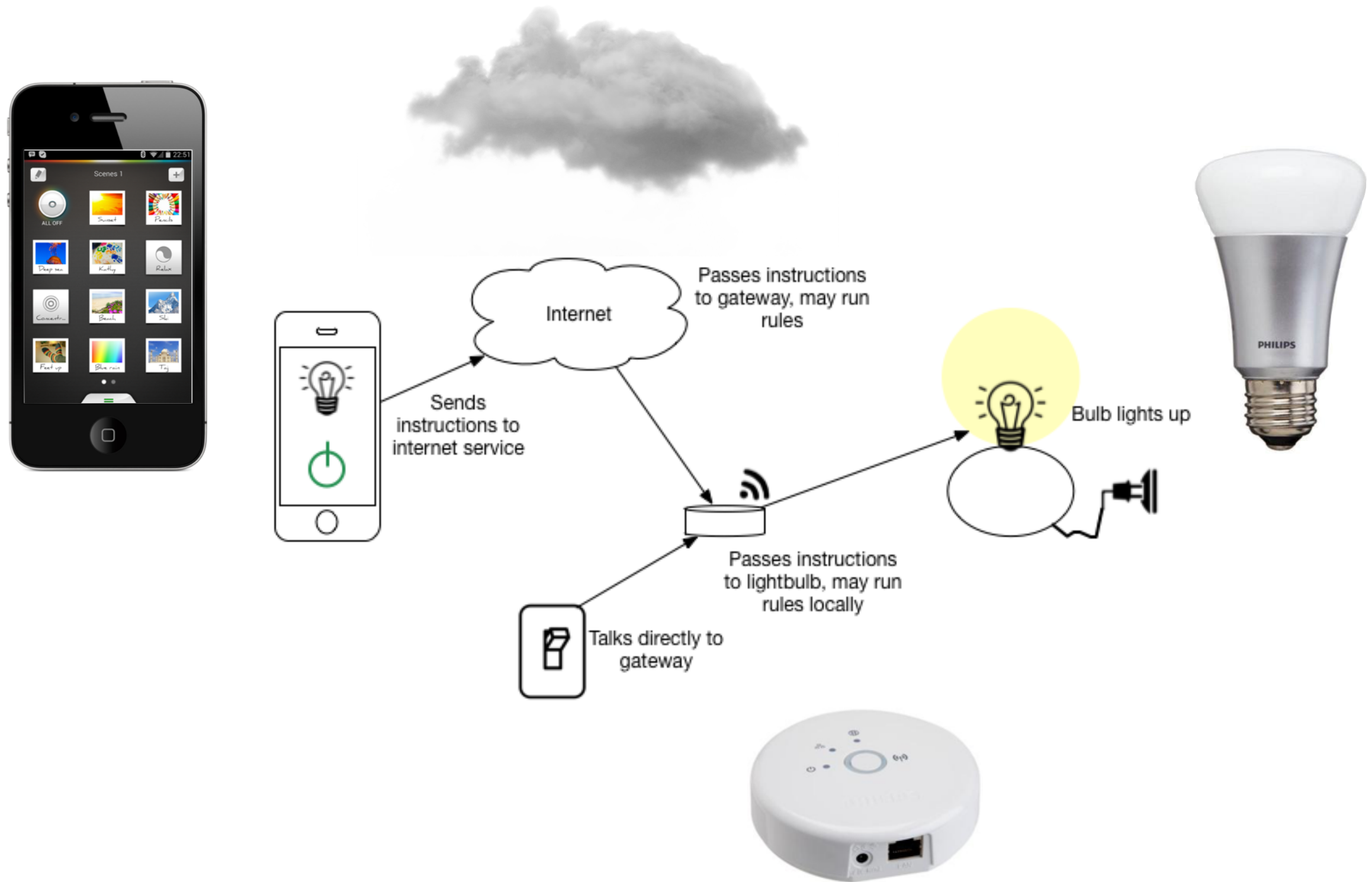
Connectedness requires users to think about system models

- Which bit does what?
- Where does code run?
- What fails/still works if connectivity is lost?

Rules in cloud

"Turn on light at 8pm"

Internet

Nothing happens

Local rules

Internet

Lamp turns on

"Turn on light at 8pm"

It's extra stuff to think about

In addition to price, aesthetics and features, customers have to understand how a product connects and whether that meets their needs.

Product images: LIFX, Philips, Cree

You can explain the system model...
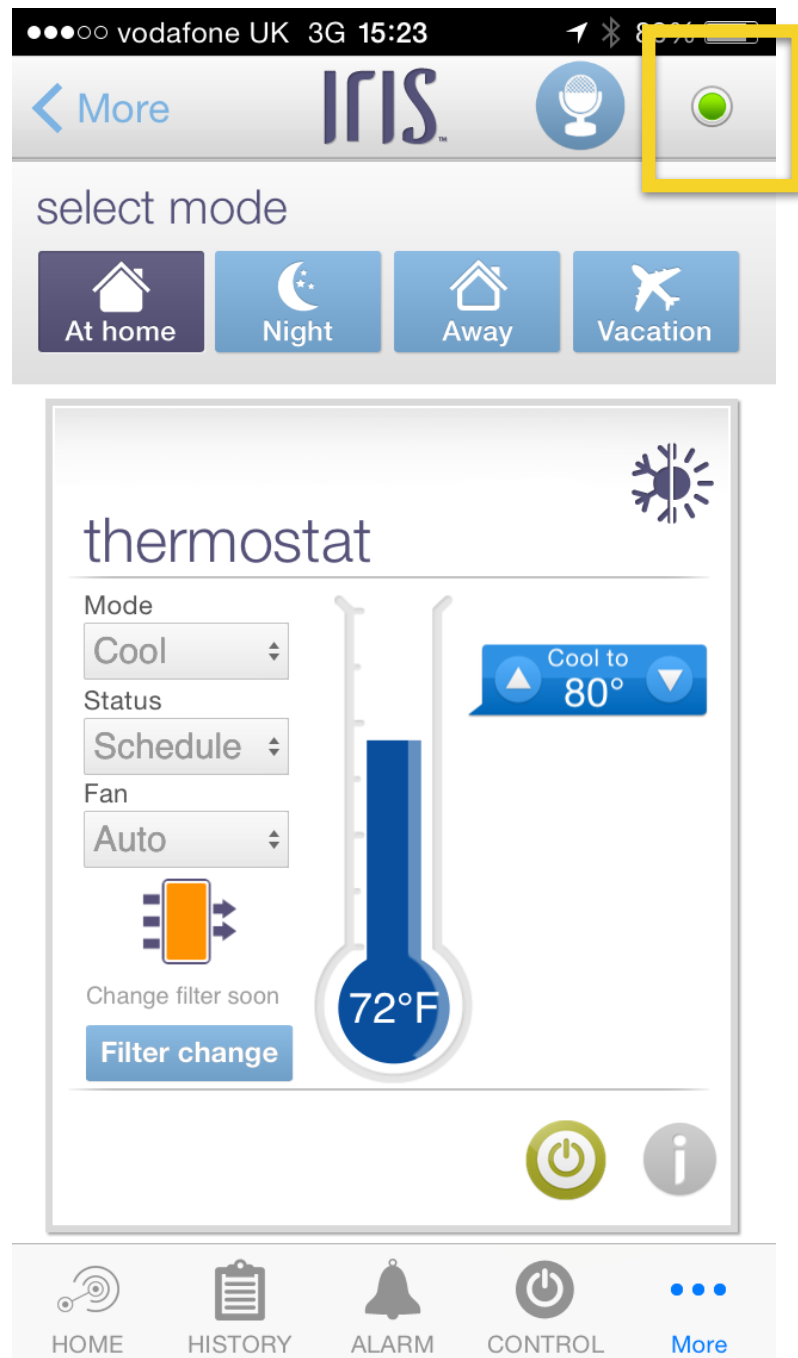
Image: Lowes

Lowes Iris: showing the connection to the hub

Image: BERG

BERG Cloud bridge: transparent network comms

…or simplify the conceptual model

Image: Estimote

Automatic gearboxes…   …and iBeacons

iBeacons

beacon sends special offer notification

Free coffee!

beacon broadcasts unique ID

phone detects ID, which wakes up the associated app

app may check in to internet service for updated special offers

Free coffee! special offer is available, notification displayed on phone
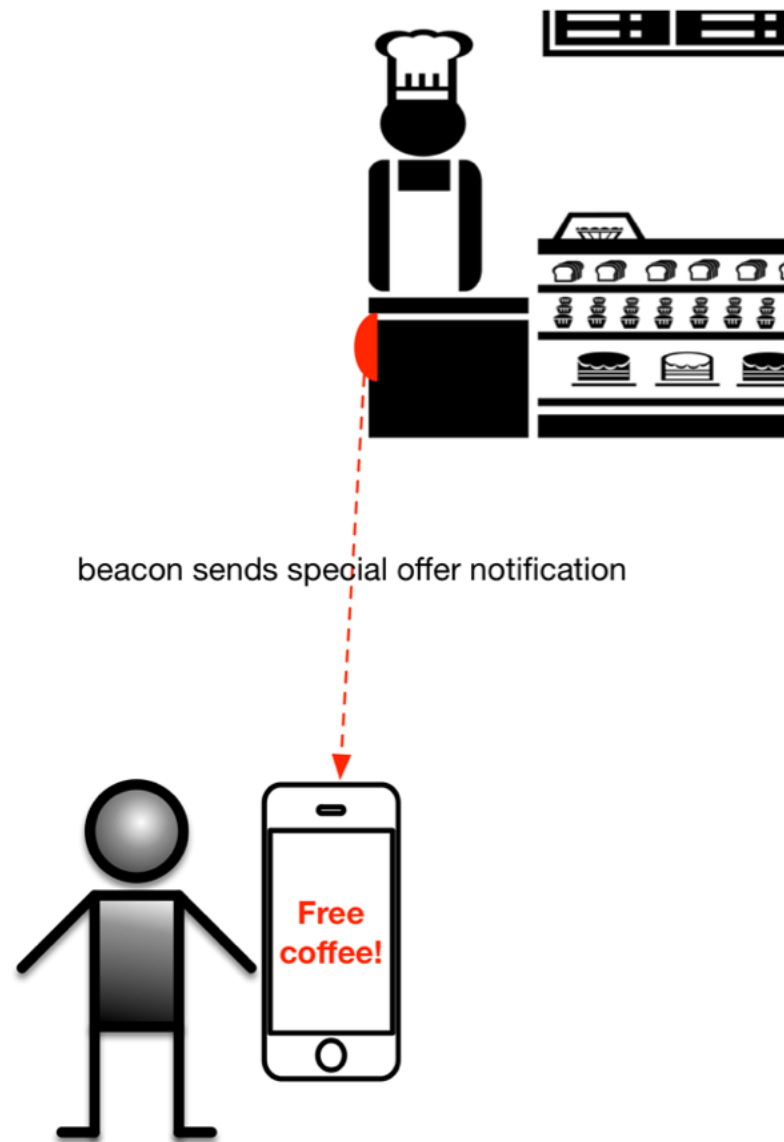
What the user needs to know   **What actually happens**
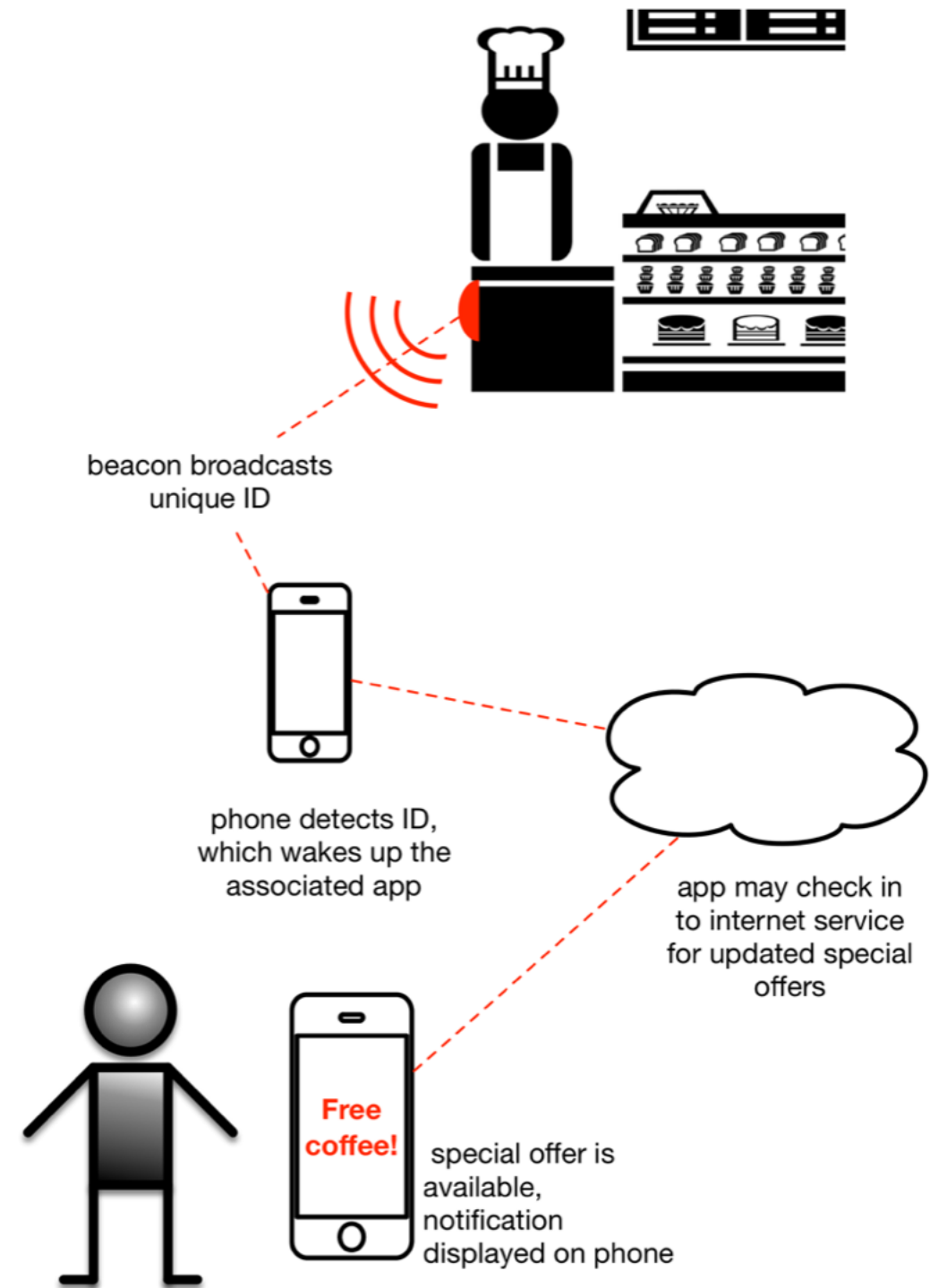
Interusability

Creating a coherent system UX

Cross-Platform Service User Experience: A Field Study and an Initial Framework. Minna Wäljas, Katarina Segerståhl, Kaisa Väänänen-Vainio-Mattila, Harri Oinas-Kukkonen MobileHCI'10: http://bugi.oulu.fi/~ksegerst/publications/p219-waljas.pdf

Composition

How functionality is distributed across devices

Distribute functionality to suit the context of use

Image: Tado

(Nearly) all interactions via phone app

Images: British Gas

Interactions mirrored on phone and thermostat

Another example:

Product images: BlueSpray, skydrop

Determining the right composition

- What best fits the context of use? What do users expect?
- What devices do users already have and what can they do?
- How much should the hardware cost?
- How much do you need to upgrade the system or change features over time?
- Do you need local control if connectivity is unavailable?
- Does the system need to work if some devices are unavailable?
- How accurate does sensing need to be?

Consistency

Appropriate consistency across UIs and interactions

What is it...

"Users should not have to worry whether different words, situations or actions mean the same thing. Follow platform conventions."

- Words, data and actions
- Aesthetic/visual design
- Interaction architecture: how functionality is organised
- Interaction logic: how tasks are structured, the types of control used

http://www.nngroup.com/articles/ten-usability-heuristics/

Consistency != making everything the same

This:

"Users should not have to worry whether different words, situations or actions mean the same thing."

...may be in tension with this:

"Follow platform conventions."

Image: Made by Many

Top priority: terminology

However different the UIs, identical functions must have the same name

Images: British Gas

Follow device platform conventions…
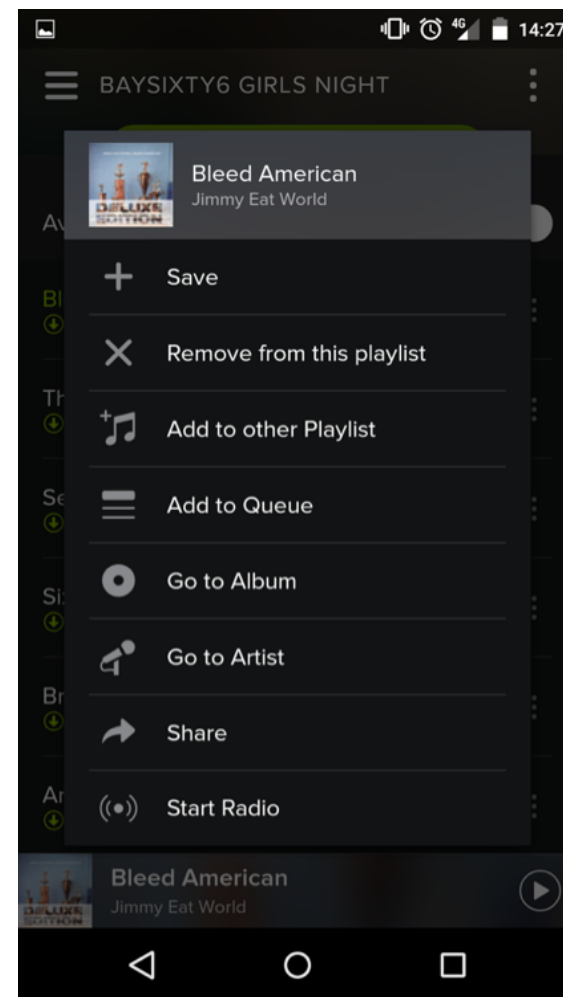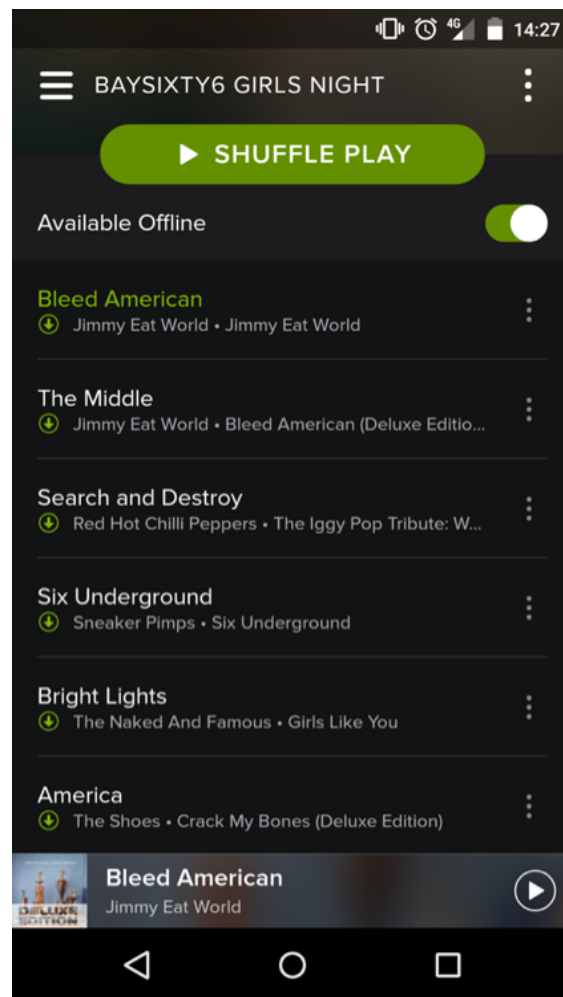
Android contextual menu
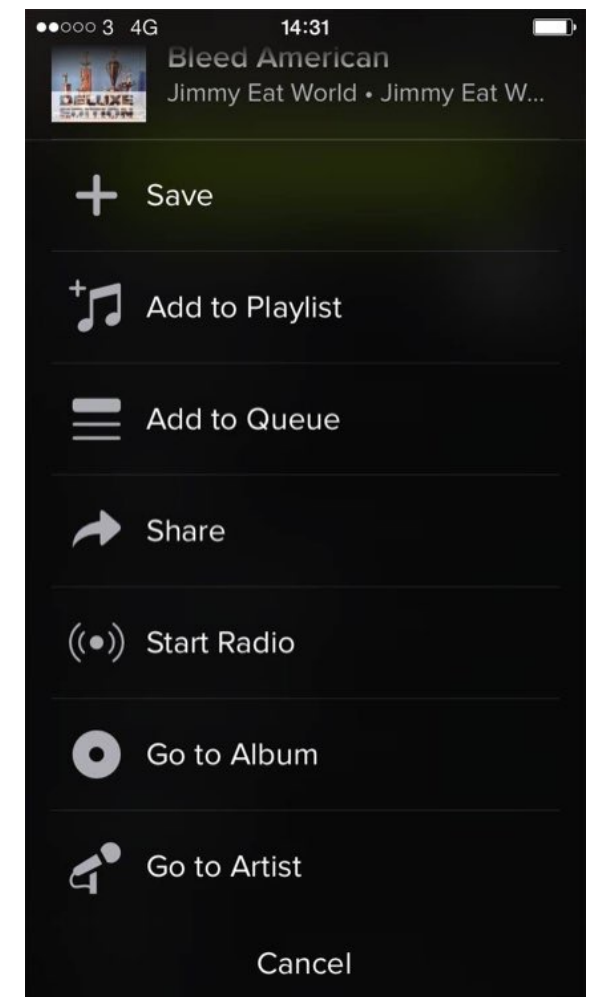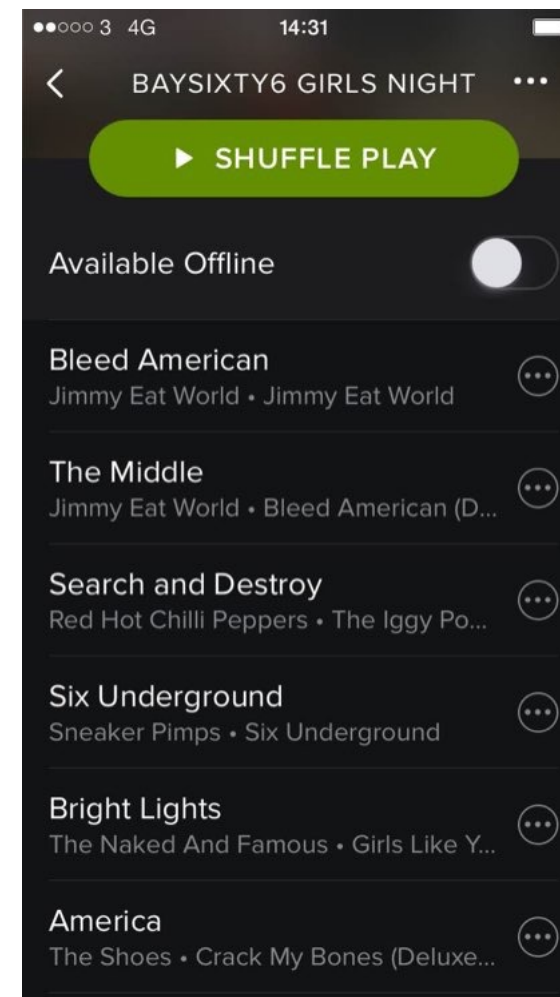
iOS separate screen

...be true to the device

Images: Nest

A touchscreen does not need a fake bezel

A thermostat does not have to pretend to be an iPhone

Aesthetic styling

"Click"

Images: Nest

Nest use visual and audio cues to tie the thermostat and phone app together

Interaction architecture need not be the same

- The logical structure of UI features and controls is likely to be platform dependent
- Different features may be prioritised on different devices
- Devices with limited UIs may need deeper hierarchies
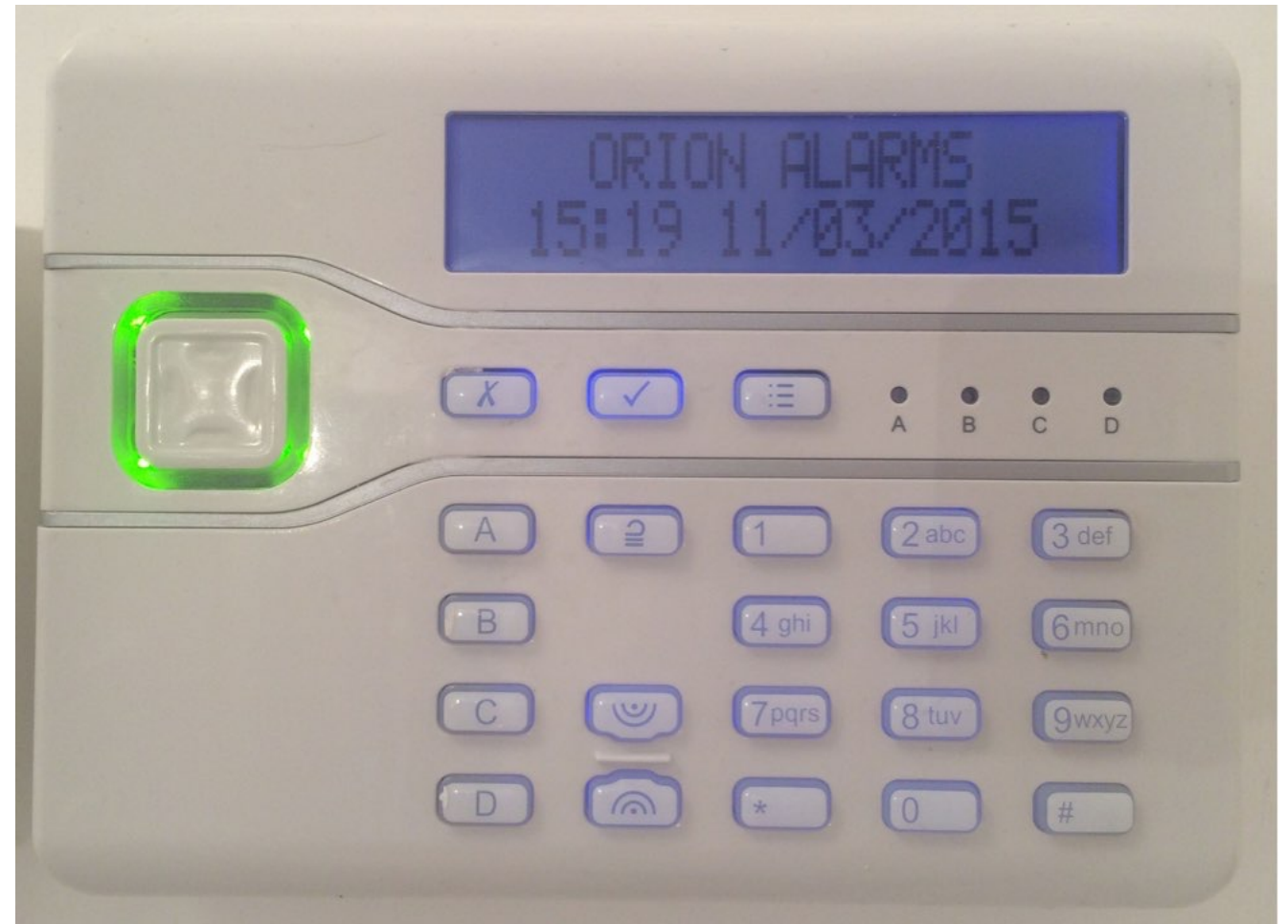
Legacy hardware UIs may be less than ideal (e.g. confusing modes) but that need not restrict other device UIs

Continuity

Fluent cross-device interactions

What is it...

- The flow of interactions and data in a coherent sequence across devices
- Continuity helps the user feel as if they are interacting with the service, not a bunch of separate devices
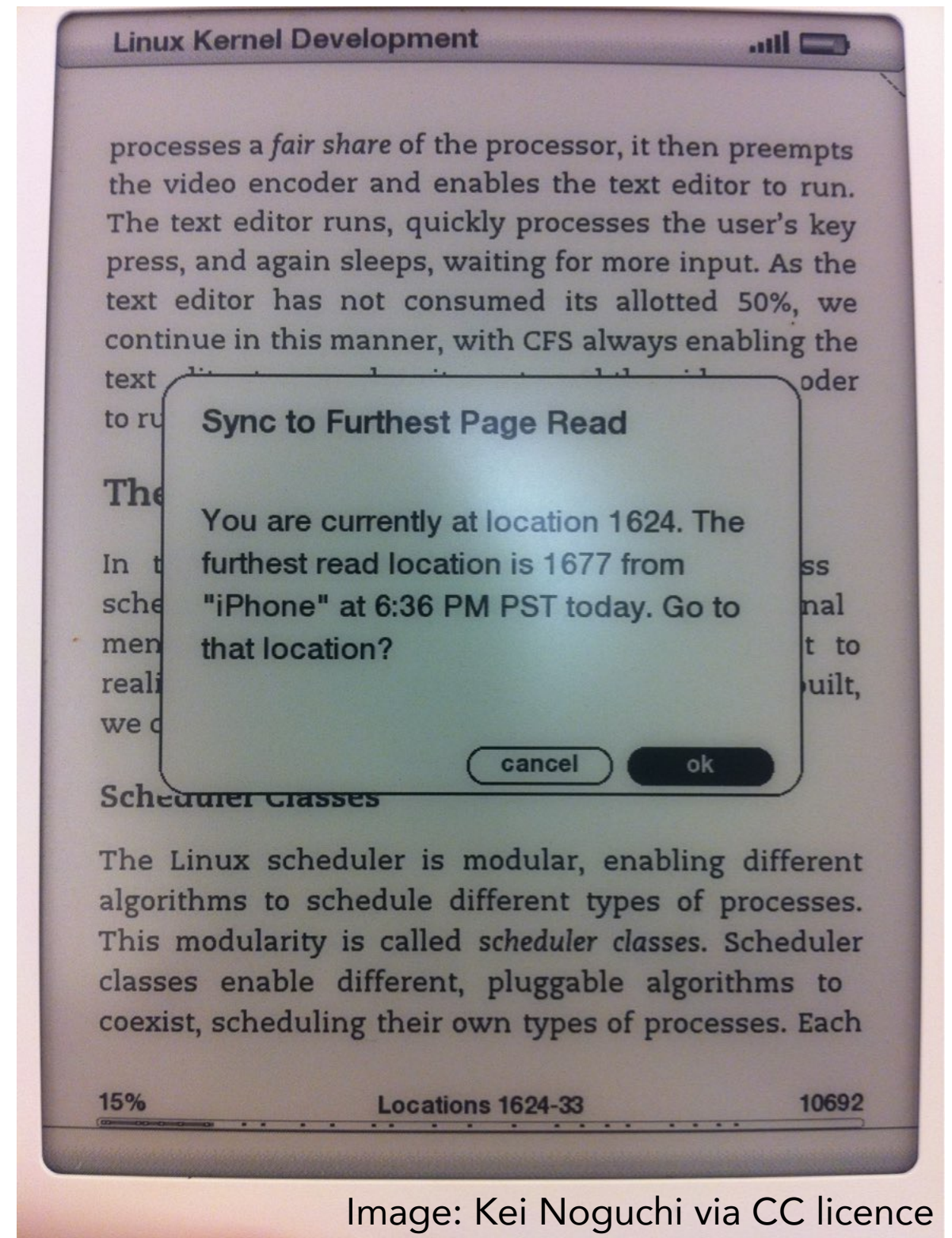
Image: Kei Noguchi via CC licence

Continuity is not always about seamlessness… it often means handling interstitial states gracefully

Some technical context:

- Some IoT devices have batteries and only connect intermittently to conserve power. In conventional UX we assume devices are mostly connected, but many IoT devices may spend more time offline

- Networks are subject to latency (esp. the internet) and reliability issues. People have mental models that help them understand this online, but delays and failures might feel strange in physical objects

Image: New Wave DV

Latency and reliability

Interactions won't always be smooth and immediate

BERG Cloudwash prototype

We expect switches to work like this

- The switch both confirms the user action and shows the state of the lamp
- But in reality, latency and reliability issues mean this can't be guaranteed over a network
- The user can't tell whether their action has been executed or whether it's in progress

Plug, and lamp, are off

User presses 'on'

Switch and lamp are on
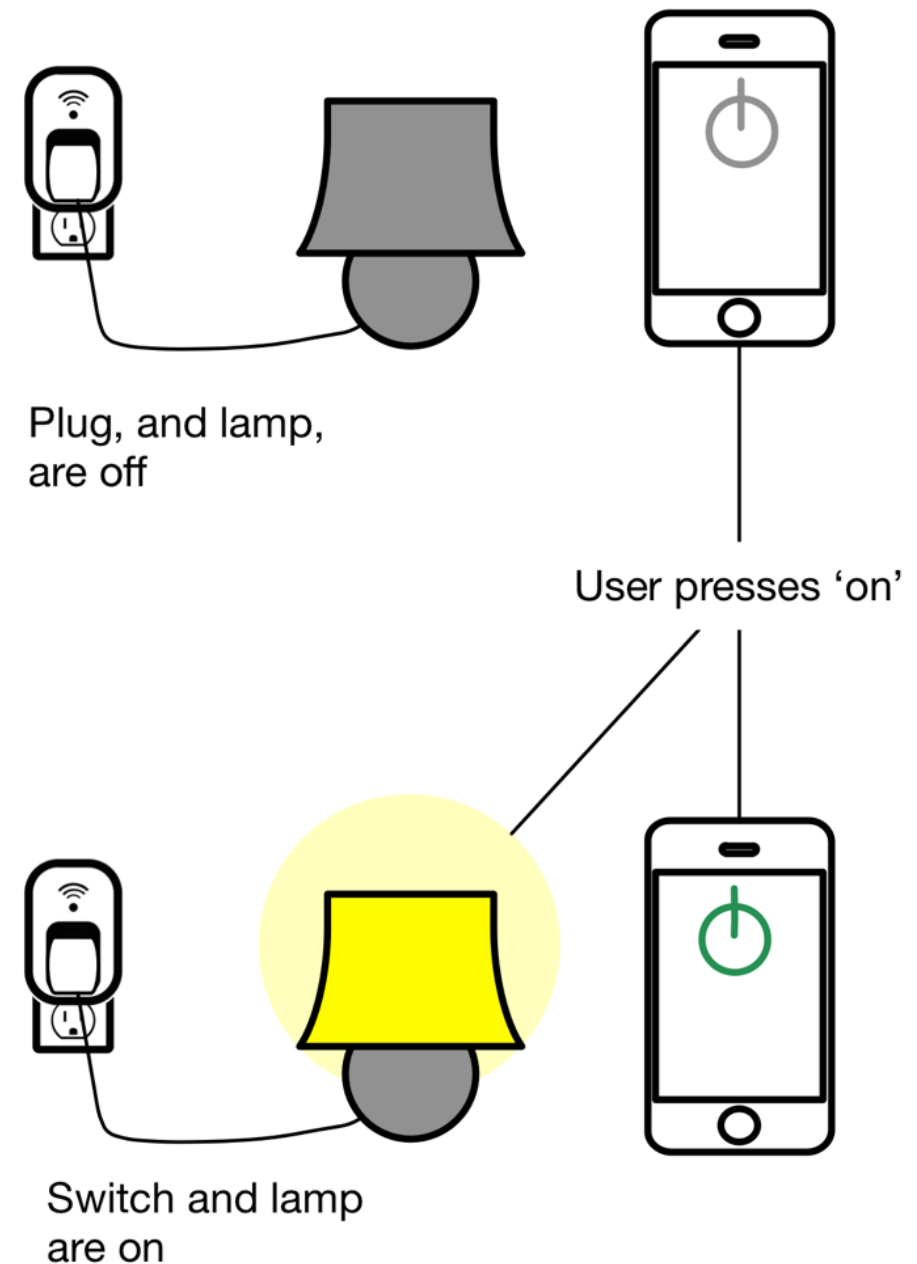
Option 1: the white lie

Confirm action, backpedal if something goes wrong

Plug, and lamp, are off

User presses 'on'

The UI shows the switch as on.. but the plug might still be off

Instruction sent to plug

Was action successful?

Yes

No

UI shows error

Plug, and lamp, are turned on
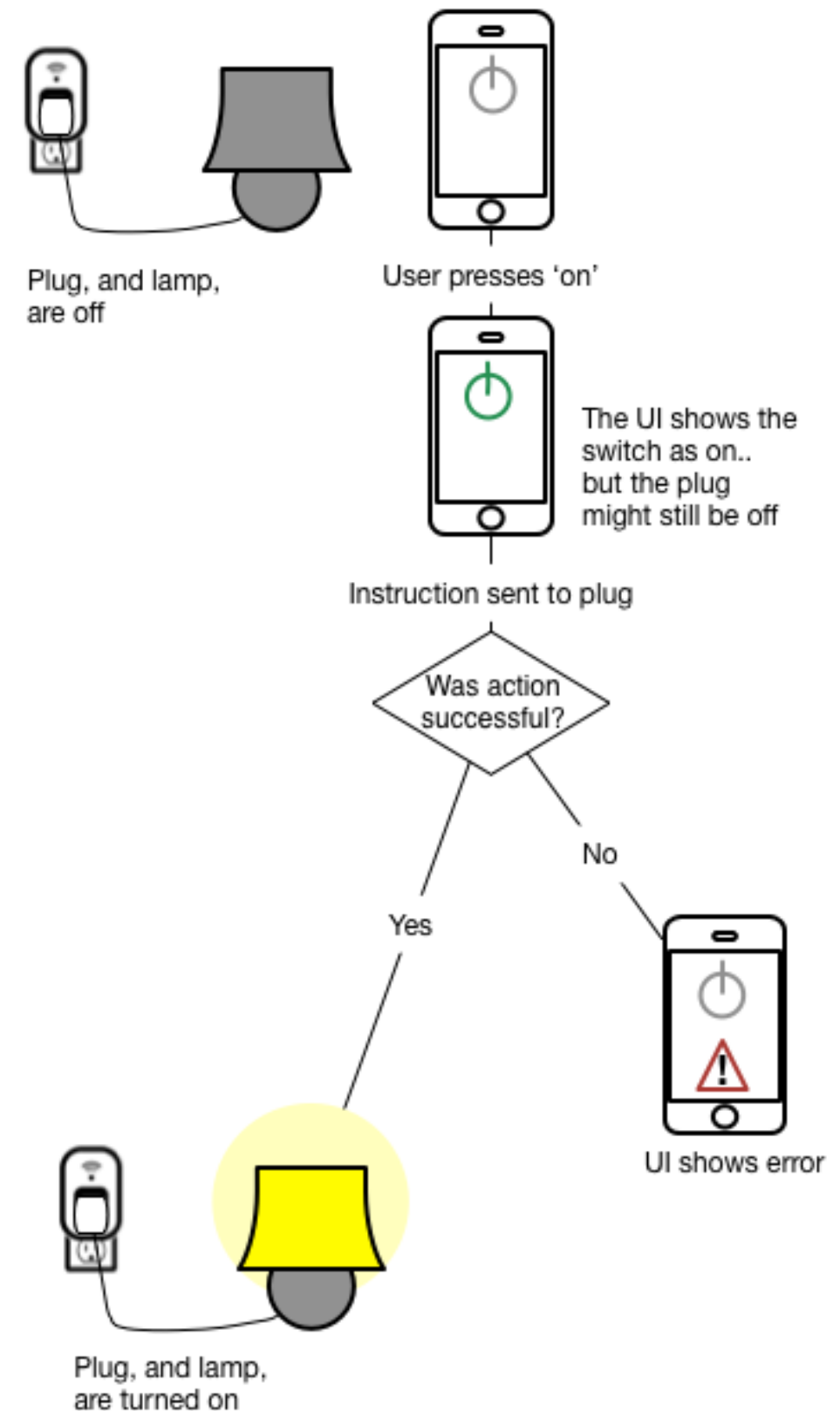
Instagram do this

The photo is already shown as 'liked', even though the instruction is still being sent
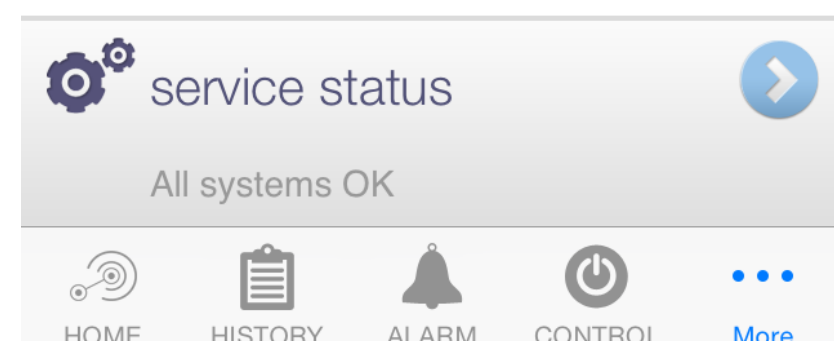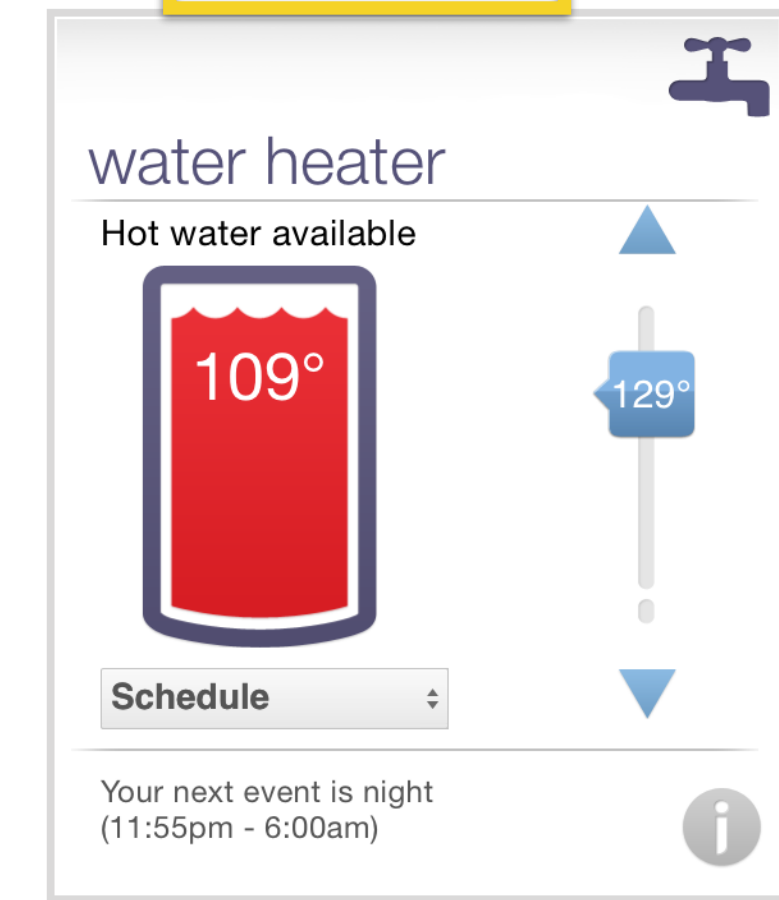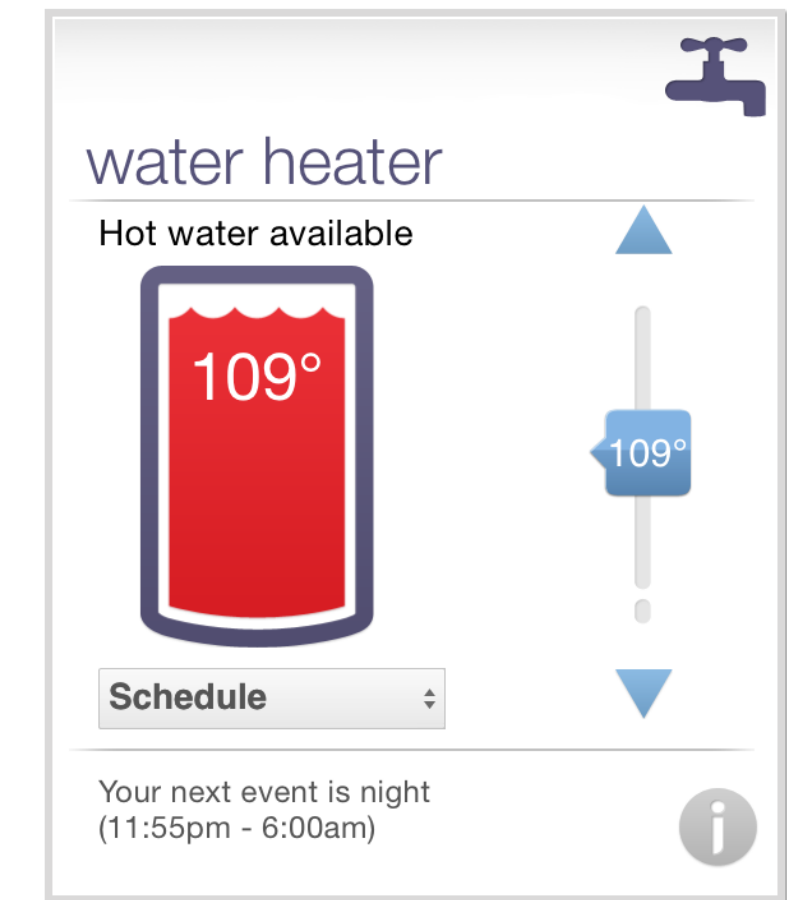
Option 2: be transparent

- Acknowledge action, show that it is in progress
- Confirm only once it's done

Plug, and lamp, are off

User presses 'on'

UI shows 'sending'

Instruction sent to plug

Was action successful?

No — UI shows error

Yes

Plug, and lamp, are turned on

UI shows 'done'
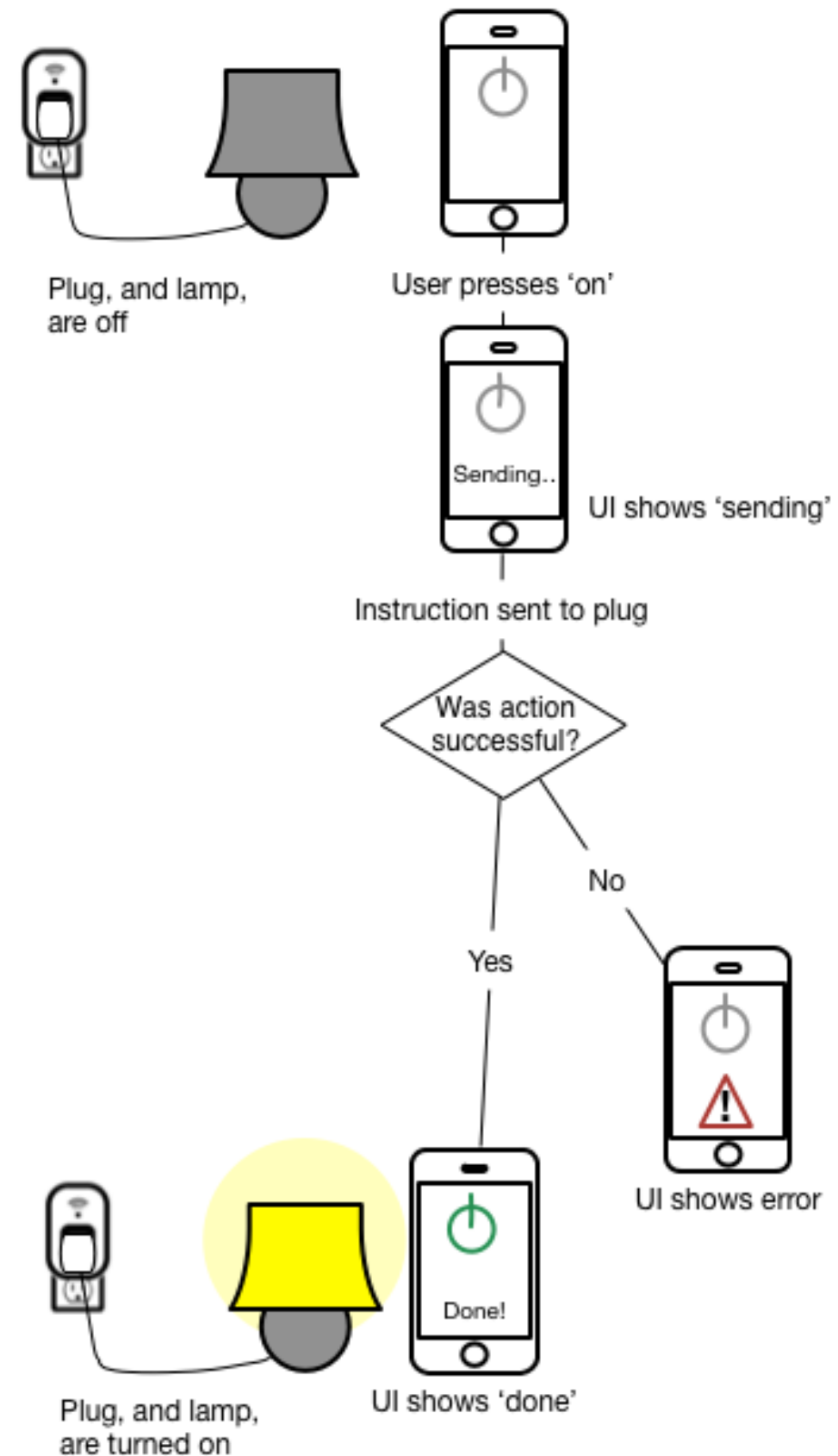
WeMo Switch does this subtly
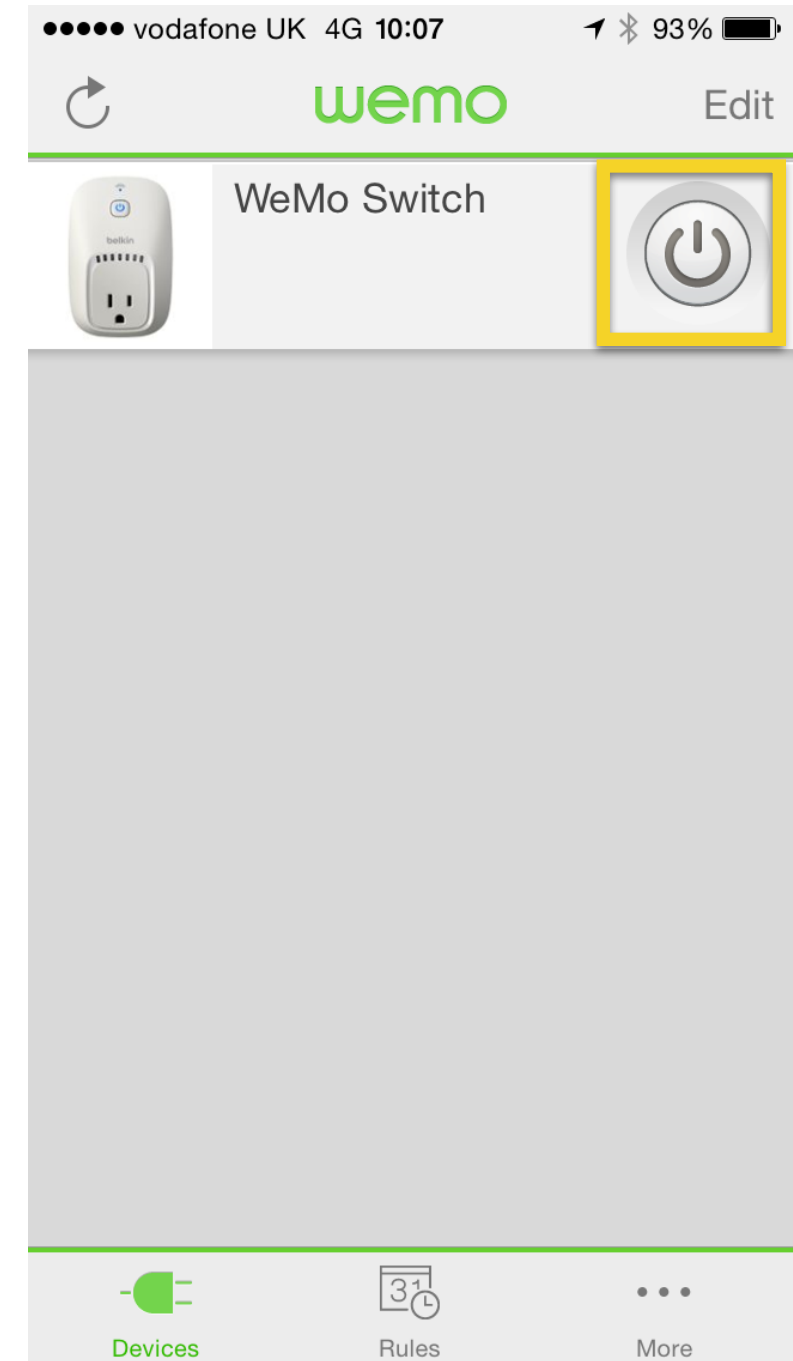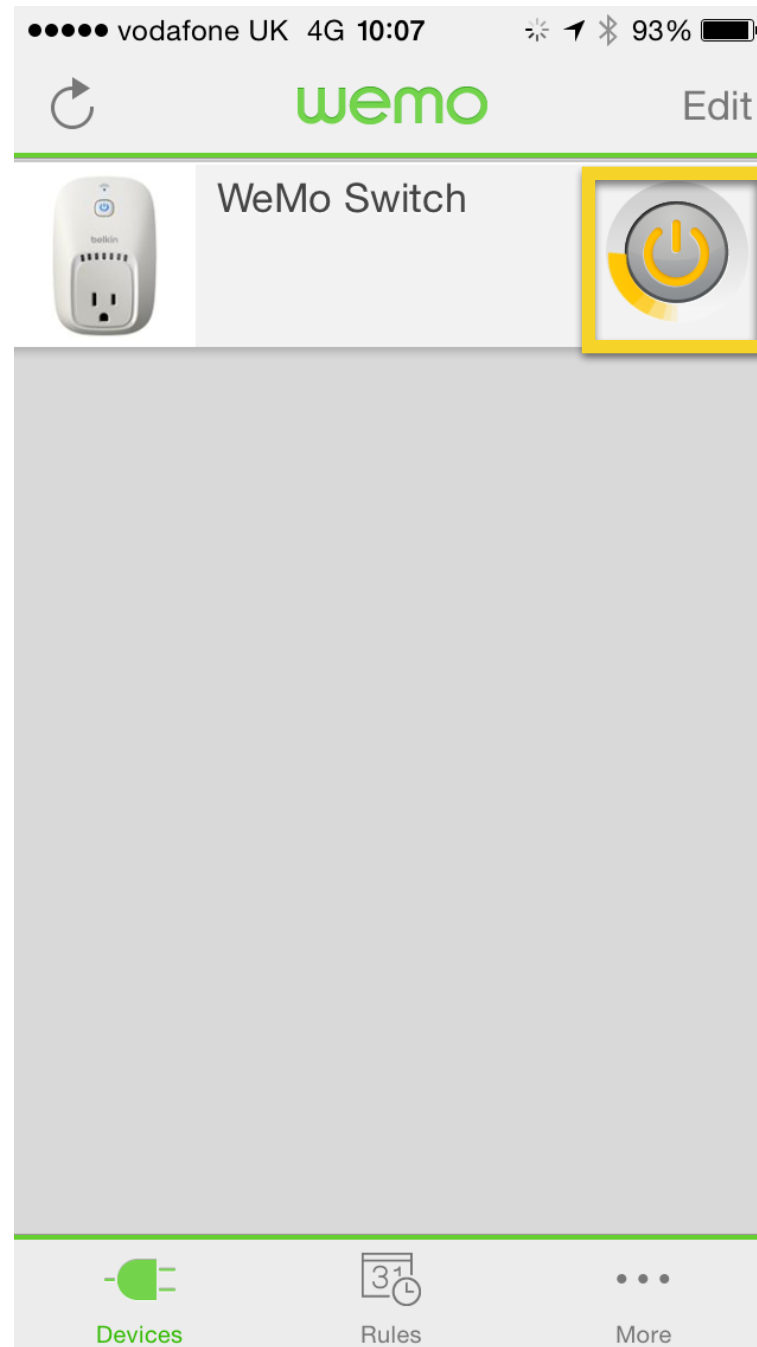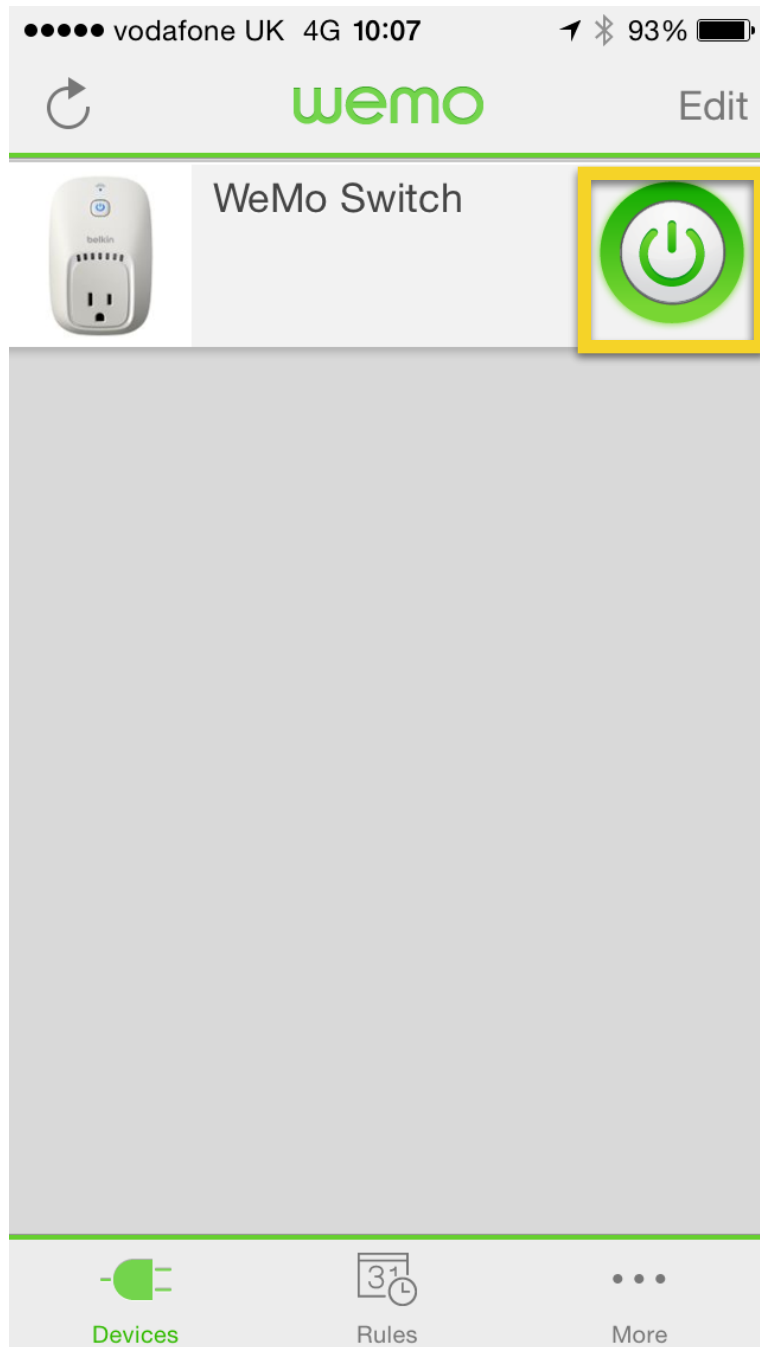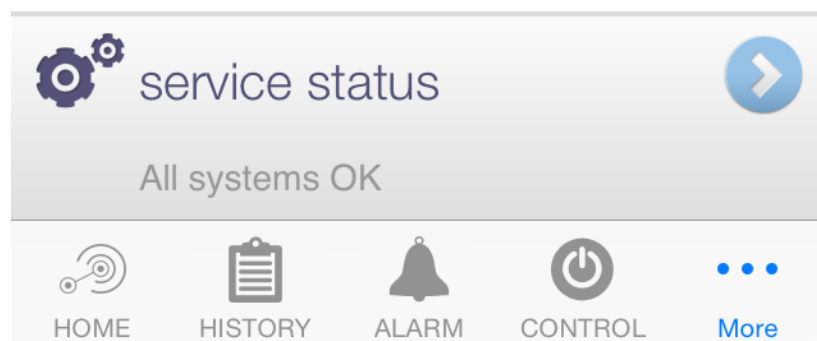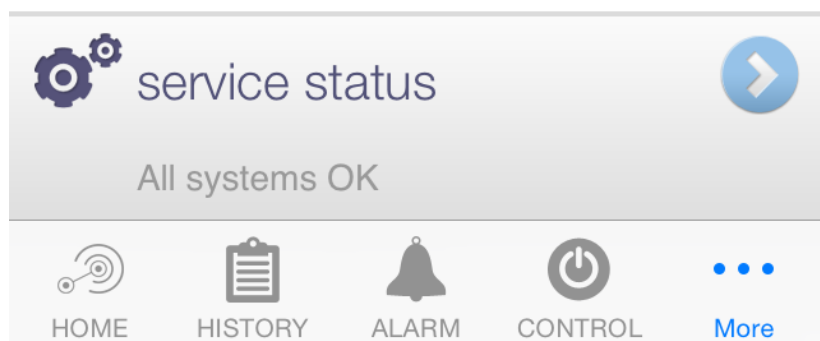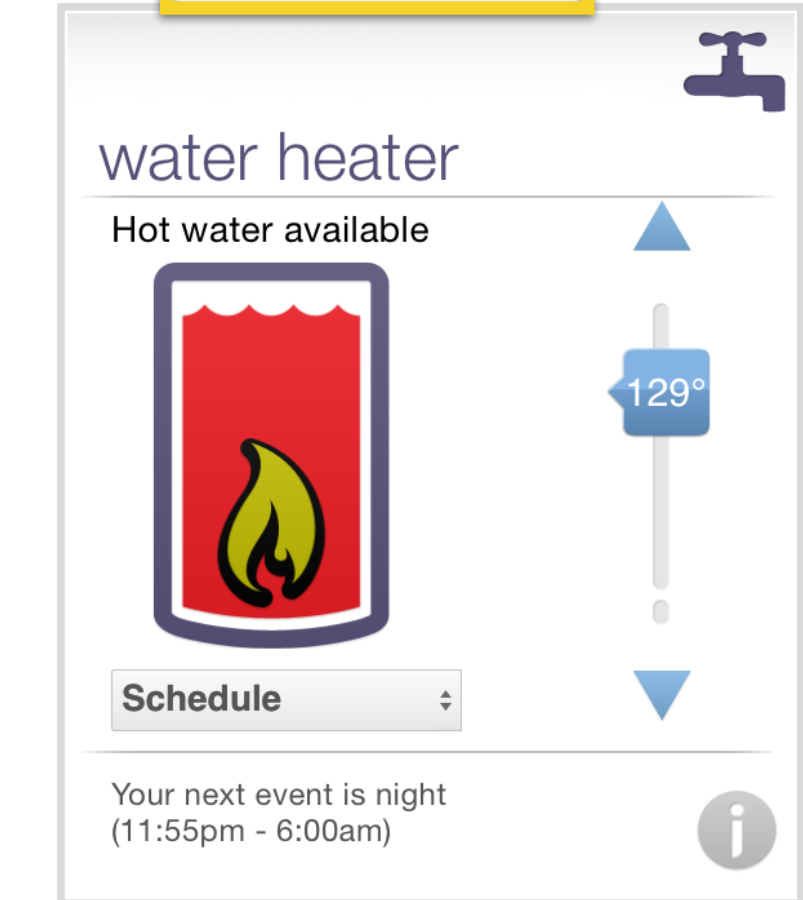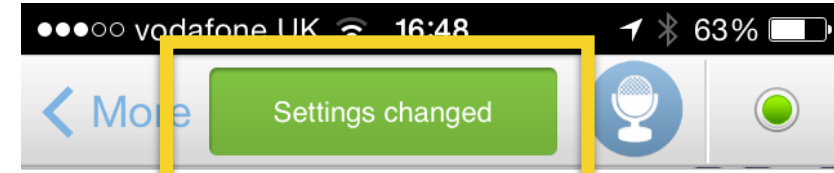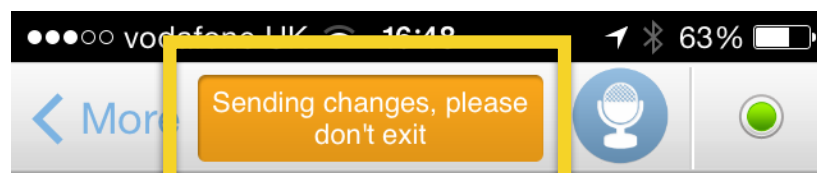
Images: Lowes

Lowes Iris is more explicit

Intermittency

21 ▲▼

2 min delay

19 ▲▼
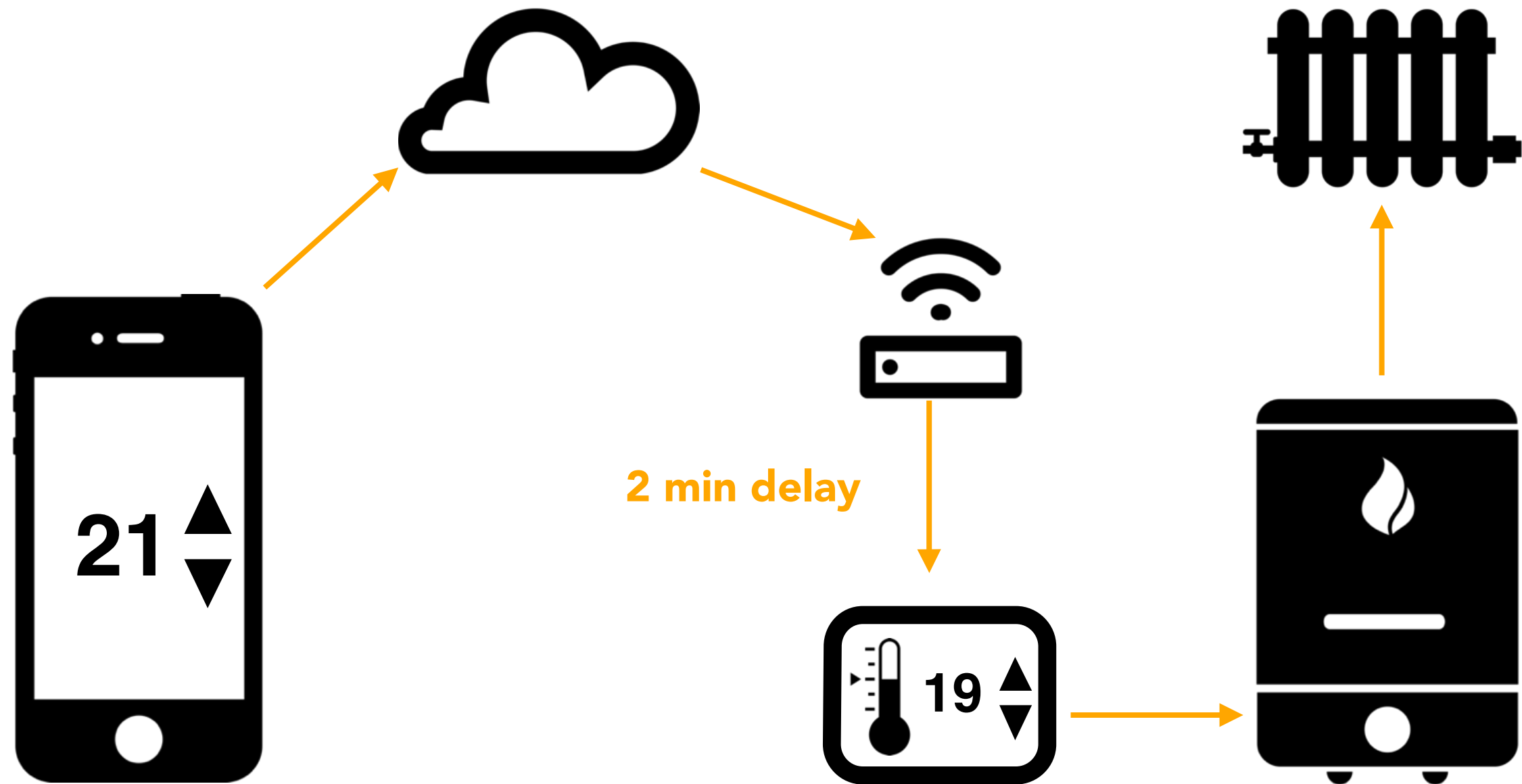
When some devices that only check into the network occasionally, there may be conflicting information about the status of the system. Data/actions may need to be timestamped.

The 'right' approach depends on context

Safety critical/urgent

Messages must get through quickly

Status information needs to be updated frequently, and clearly indicate how old it is

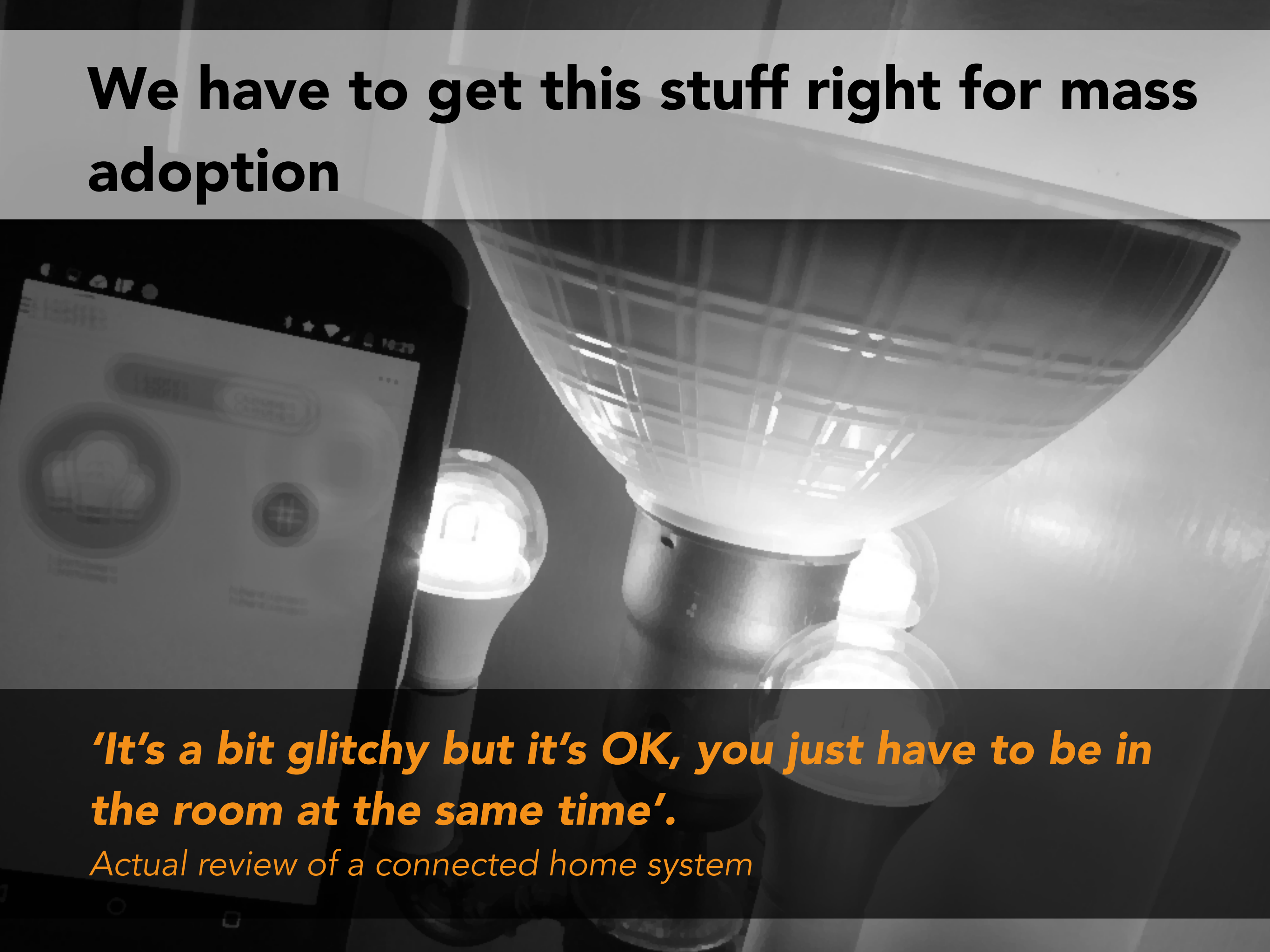Need to know when instructions have been received and acted upon

Low touch/non-critical:

Assume it's working unless notified of a problem

OK if data or instructions take time to get through (as long as they are timestamped)

Images: MyLively, Efergy

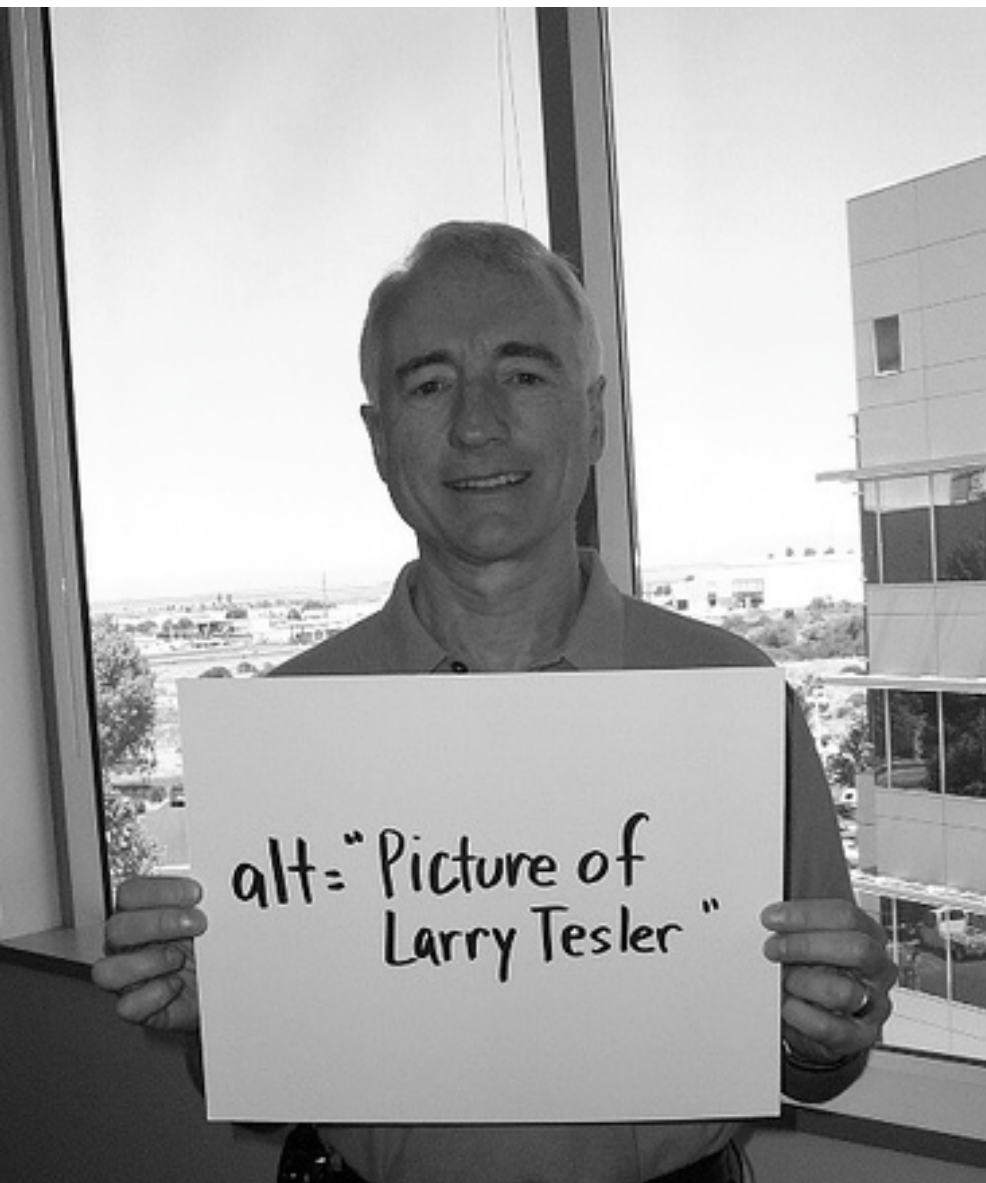A final thought

Good consumer UX for IoT is surprisingly hard

We have to get this stuff right for mass adoption

'It's a bit glitchy but it's OK, you just have to be in the room at the same time'.

Actual review of a connected home system

Tesler's law of the conservation of complexity:

As you make the user interaction simpler you make things more complicated for the designer or engineer

Larry Tesler, former VP of Apple

Thank you

@clurr

claire@clairerowland.com

O'REILLY®

Designing Connected Products

UX FOR THE CONSUMER
INTERNET OF THINGS

Early Release

RAW & UNEDITED

Claire Rowland,
Elizabeth Goodman,
Martin Charlier,
Ann Light & Alfred Lui
Foreword by Tom Igoe