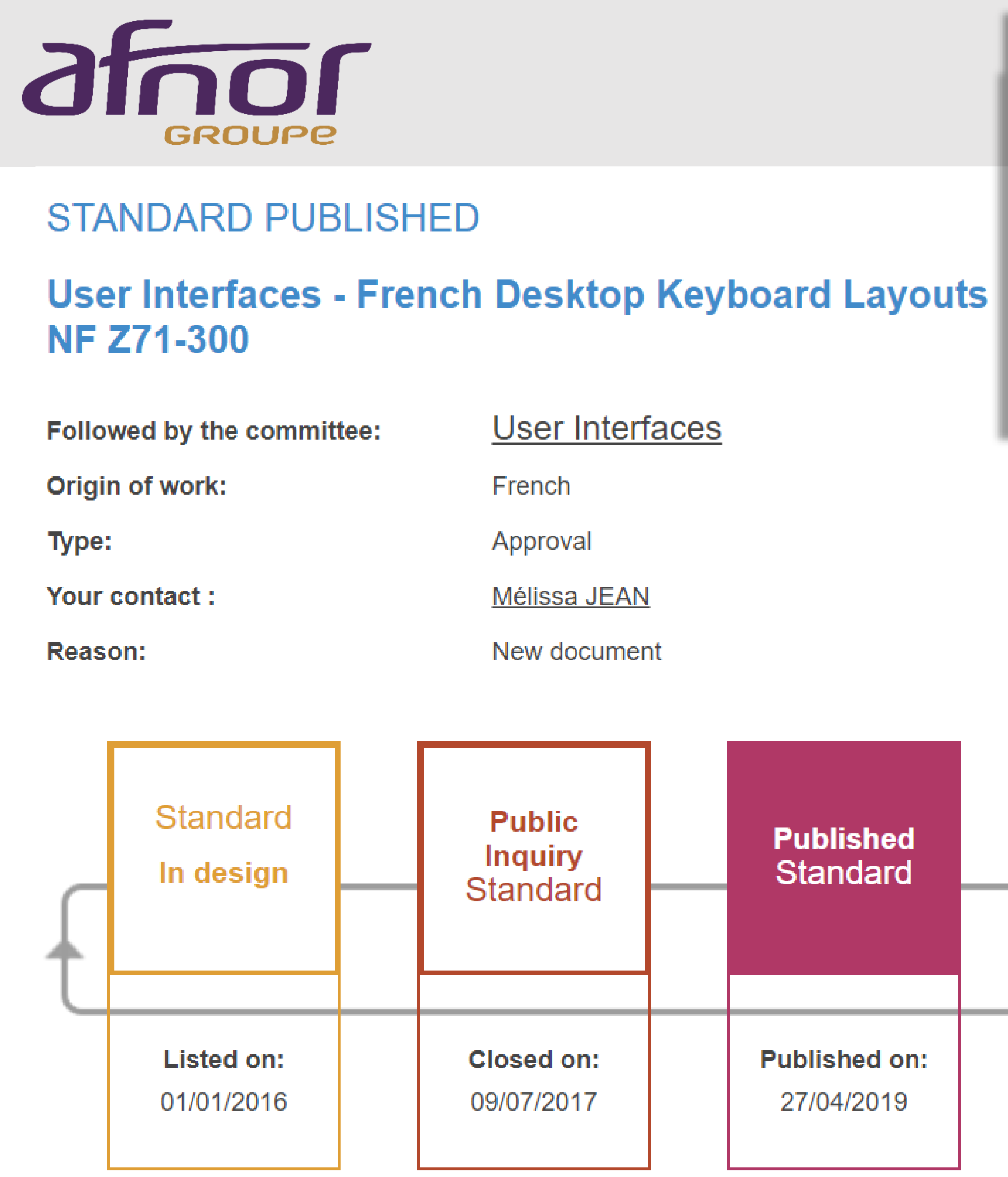
Two Research Examples



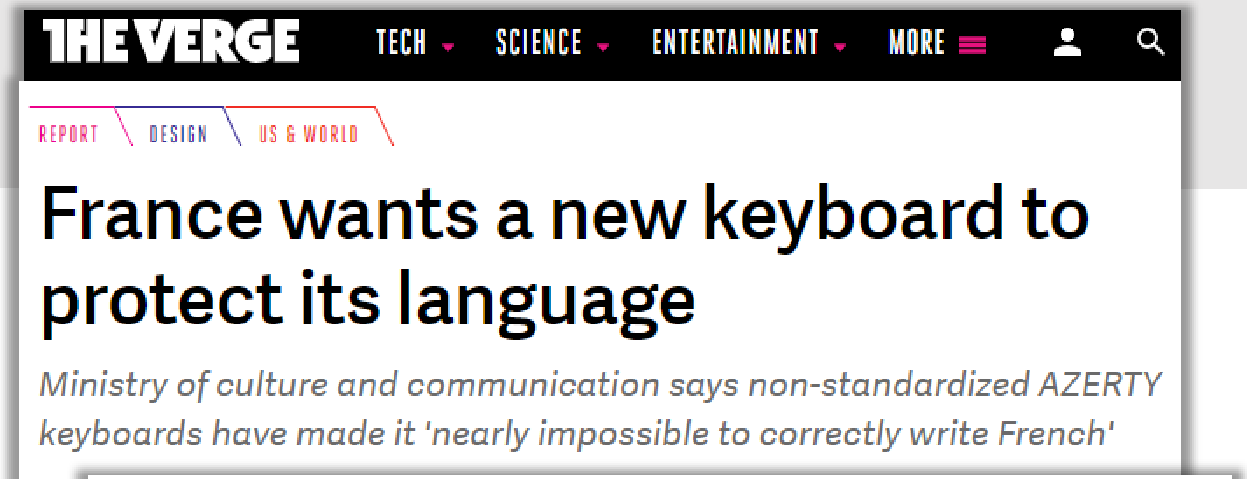
afnor GROUPE

STANDARD PUBLISHED

User Interfaces - French Desktop Keyboard Layouts NF Z71-300

Followed by the committee:	User Interfaces
Origin of work:	French
Type:	Approval
Your contact :	Mélissa JEAN
Reason:	New document

Standard In design	Public Inquiry Standard	Published Standard	Pl
Listed on: 01/01/2016	Closed on: 09/07/2017	Published on: 27/04/2019	2

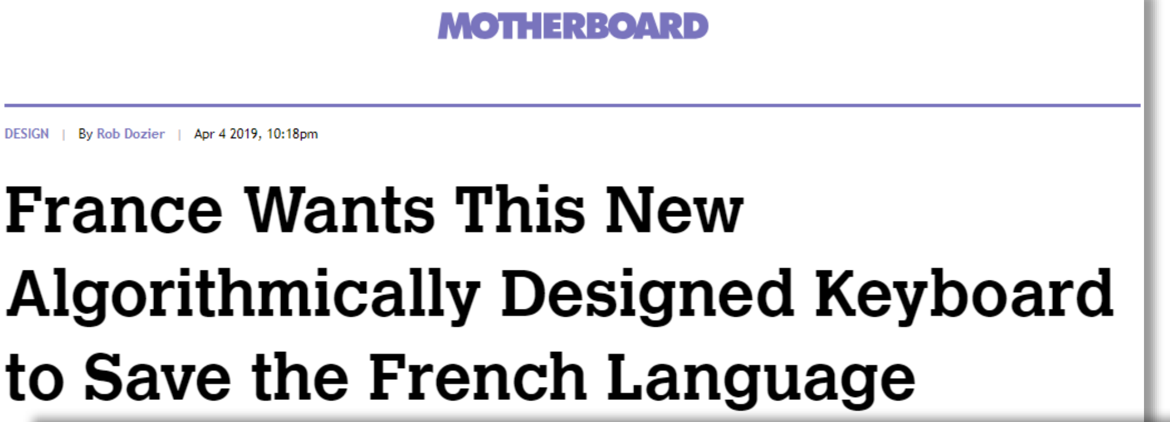


REPORT DESIGN US & WORLD

France wants a new keyboard to protect its language

Ministry of culture and communication says non-standardized AZERTY keyboards have made it 'nearly impossible to correctly write French'

By A

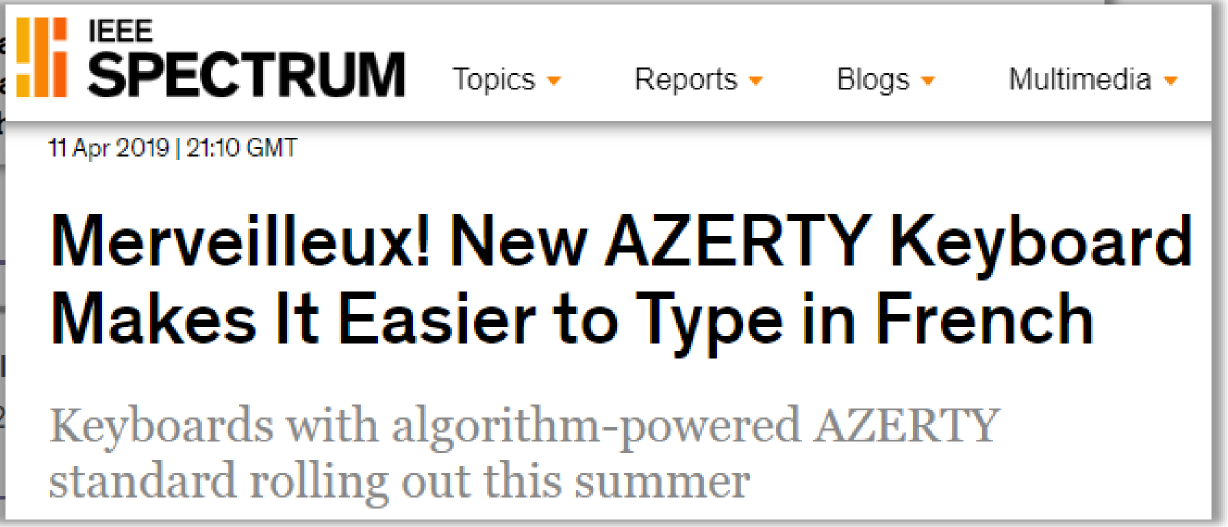


MOTHERBOARD

DESIGN | By Rob Dozier | Apr 4 2019, 10:18pm

France Wants This New Algorithmically Designed Keyboard to Save the French Language

Fra
gra
oth



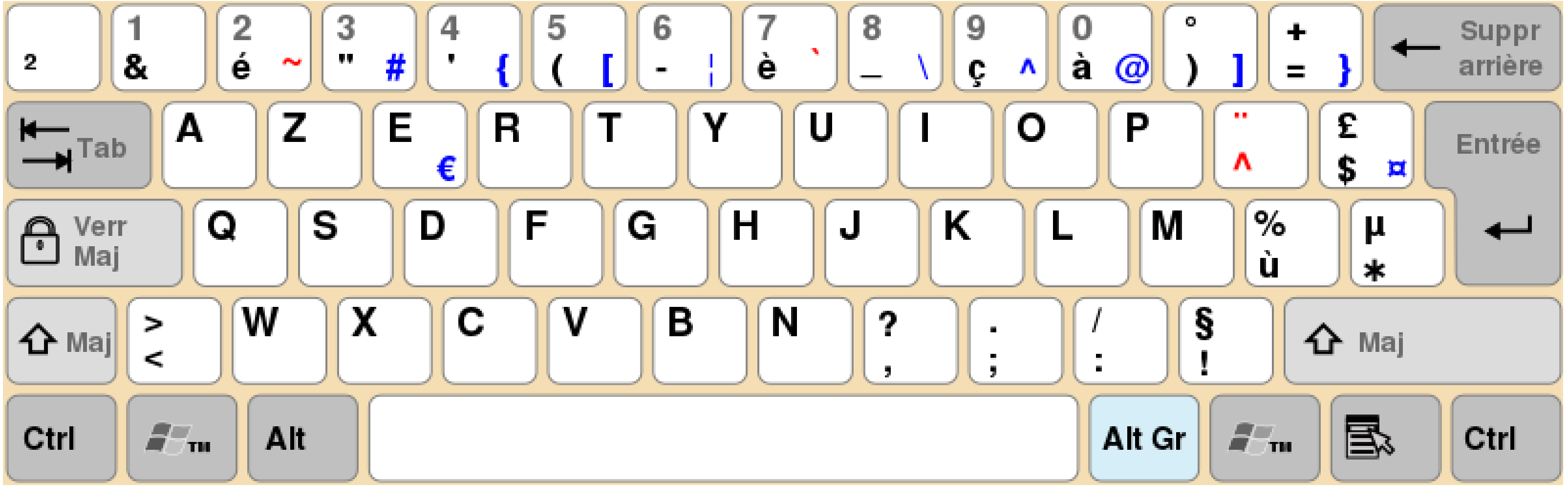
11 Apr 2019 | 21:10 GMT

Merveilleux! New AZERTY Keyboard Makes It Easier to Type in French

Keyboards with algorithm-powered AZERTY standard rolling out this summer

” It is almost impossible to write correctly French with a keyboard marketed in France”

French Ministry of Culture and Communications



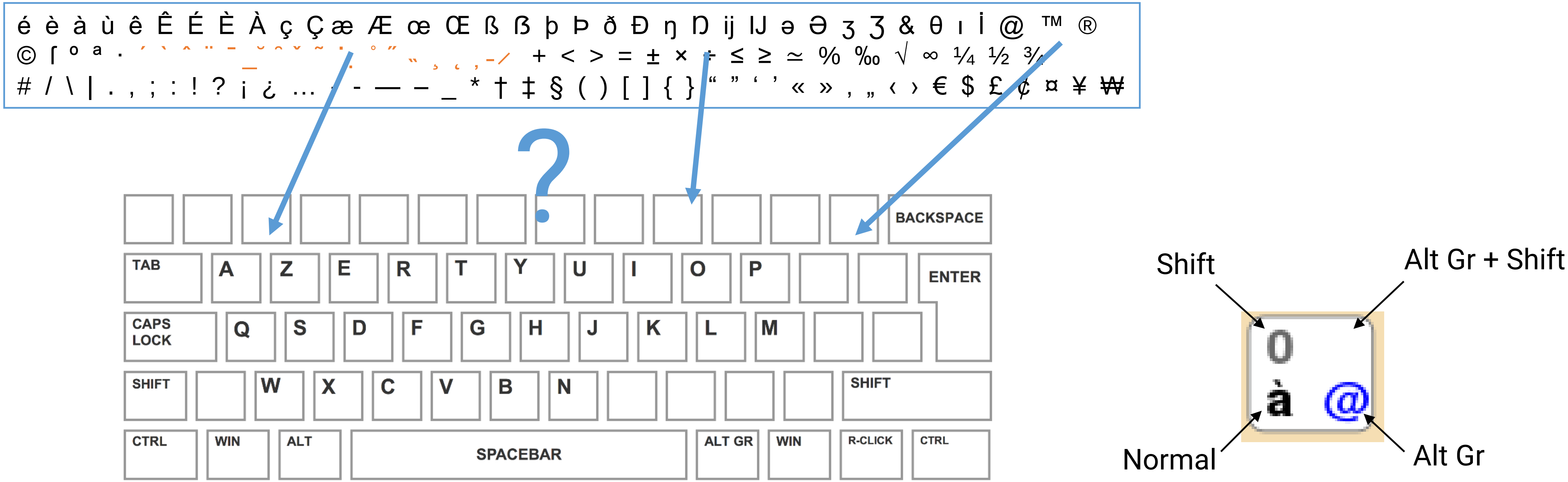
Fun facts about the French keyboard

- There is no way to type **œ**uf, **É**té or FRAN**Ç**AIS (*egg, Summer, FRENCH)*
- où (*where*) is the only word that contains **ù**, yet it has it’s own key
- No french quotation marks « »
- Typing . requires Shift, so do all numbers, and try finding the @ key…

Design Goal

Standardization committee:

”The new keyboard should facilitate typing of correct french, should be easy to learn and intuitive to use”



é è à ù ê Ê É È À ç Ç æ Æ œ Œ ß ẞ þ Þ ð Đ ŋ Ŋ ĳ Ĳ ə Ə ʒ Ʒ & θ ı İ @ ™ ®

© ʃ º ª · ´ ` ˆ ¨ ¯ _ ˘ ̑ ˇ ˜ ˙ ̣ ˚ ˝ ̏ ¸ ˛ ̦ - ⁄ + < > = ± × ÷ ≤ ≥ ≃ % ‰ √ ∞ ¼ ½ ¾

/ \ | . , ; : ! ? ¡ ¿ … ‐ - — – _ * † ‡ § ( ) [ ] { } “ ” ‘ ’ « » , „ ‹ › € $ £ ¢ ¤ ¥ ₩

[Feit, Nancel, Weir, John, Karrenbauer, Oulasvirta, upcoming]

Special character entry

- Many more special characters than letters
- Frequency varies with task but mostly infrequent in comparison to letters
- Mostly entered before or after a letter (which are fixed)

We can partially simplify the QAP to a **linear** assignment problem (makes it solvable with some tricks from optimization experts)

Design Space

- n characters – $i, j \epsilon \Sigma$
- m keyslots – $k, l \epsilon S$

- $x_{ik} = 1$ if character i is assigned to keyslot k,
- $x_{ik} = 0$ otherwise

- Goal: Find the assignment of characters to keyslots that minimizes the cost of typing (special characters)

Objectives

Motor Performance: minimize time to move between special character and letter

$$\min \sum_{i=1}^{N} \sum_{k=1}^{M} \sum_{c=1}^{27} (p_{ci}\mathbf{t_{ck}} + p_{ic}\mathbf{t_{kc}})x_{ik}$$

Note: linear function

Ergonomics: minimize frequent extreme movements of wrist and fingers when entering a special character in combination with a letter

$$\min \sum_{i=1}^{N} \sum_{k=1}^{M} \sum_{c=1}^{27} (p_{ci}\mathbf{e_{ck}} + p_{ic}\mathbf{e_{kc}})x_{ik}$$

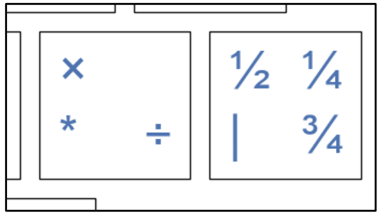
Note: linear function

Don't need to understand this in detail

Objectives

Learnability (Intuitiveness): minimize the distance between similar characters to ensure a consistent and intuitive layout that is easy to use and learn.

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{m} \sum_{l=1}^{m} \mathbf{s_{ij}} d_{kl} x_{ik} x_{jl} + \sum_{i=1}^{n} \sum_{k=1}^{m} \sum_{c=1}^{27} \mathbf{s_{ic}} d_{kc} x_{ik}$$



Learnability (Familiarity): minimize the distance of a character to its position on the standard keyboard layout.

$$\min \sum_{i=1}^{N} \sum_{k=1}^{M} p_i \mathbf{d_{k\mathcal{A}(i)}} x_{ik}$$

Note: linear function

Don't need to understand this in detail

Optimization Problem

$$\begin{aligned}
\min\ w_P &\sum_{i=1}^{N}\sum_{k=1}^{M}\sum_{c=1}^{27}(p_{ci}t_{ck} + p_{ic}t_{kc})x_{ik} \\
+\ w_E &\sum_{i=1}^{N}\sum_{k=1}^{M}\sum_{c=1}^{27}(p_{ci}e_{ck} + p_{ic}e_{kc})x_{ik} \\
+\ w_I &\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m}\sum_{l=1}^{m} s_{ij}d_{kl}x_{ik}x_{jl} + \sum_{i=1}^{n}\sum_{k=1}^{m}\sum_{c=1}^{27} s_{ic}d_{kc}x_{ik} \\
+\ w_F &\sum_{i=1}^{N}\sum_{k=1}^{M} p_i d_{k\mathcal{A}(i)}x_{ik}
\end{aligned}$$

Assign each letter to exactly one key

$$s.t. \sum_{k=1}^{M} x_{ik} = 1 \quad \forall i \in \{1, \ldots N\}$$

*Assign each key to **at most** one letter*

$$\sum_{i=1}^{N} x_{ik} \leq 1 \quad \forall k \in \{1, \ldots M\}$$

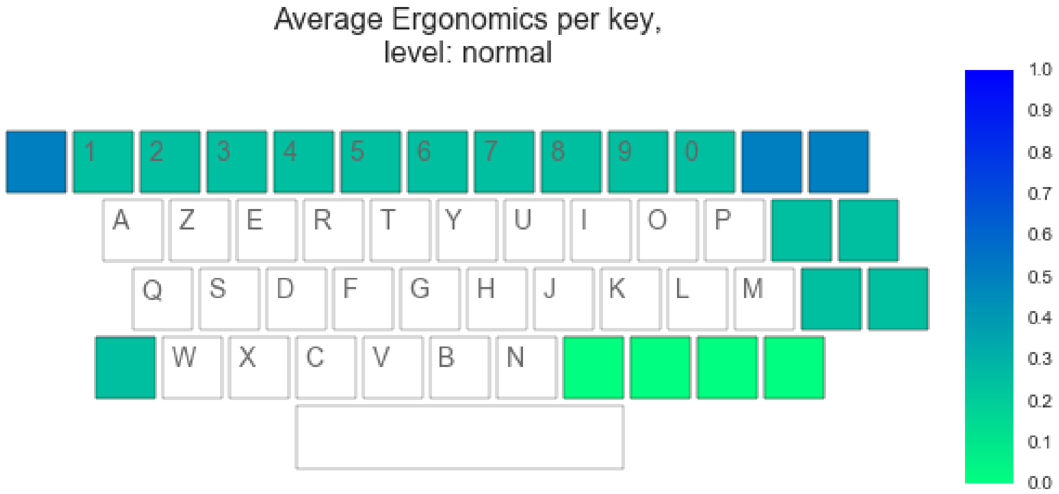
$$x_{ik} \in \{0, 1\} \quad \forall i \in \{1, \ldots N\}, k \in \{1, \ldots M\}$$

Input data

- **Performance**: crowdsourced movement times of 7555 different key pairs embedded in nonsense words and repeated 10 times.

Input data

- **Performance**: crowdsourced movement times of 7555 different key pairs embedded in nonsense words and repeated 10 times.
- **Ergonomics:** heuristic cost based on literature, equally punishing movement toward outer right and left, wrist and finger extension, use of one or even two modifier keys



Input data

- **Performance**: crowdsourced movement times of 7555 different key pairs embedded in nonsense words and repeated 10 times.
- **Ergonomics:** heuristic cost based on literature, equally punishing movement toward outer right and left, wrist and finger extension, use of one or even two modifier keys
- **Similarity:** heuristic score based on multiple similarity criteria such as visual similarity (à and a), semantic or functional similarity ($ and £), phonetic similarity (s and ß), etc.
- **Distance:** Manhattan distance weighted by letter frequency
- **Character and letter pair frequencies:** legal texts, newspapers, wikipedia, emails, facebook, twitter, code (Java, C++, Python, Html, CSS, javascript)

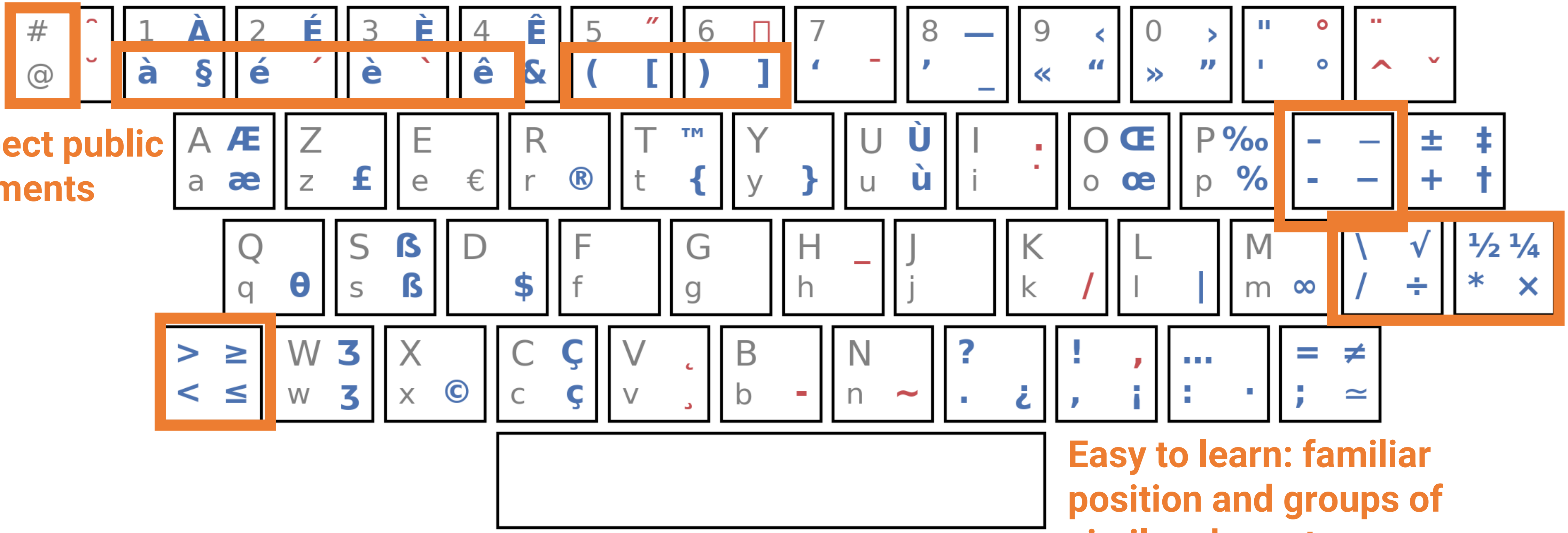
The new French keyboard standard

norme-azerty.fr

Feit, Nancel, John, Weir, Karrenbauer, and Oulasvirta, *under review*

French characters easily and quickly accessible

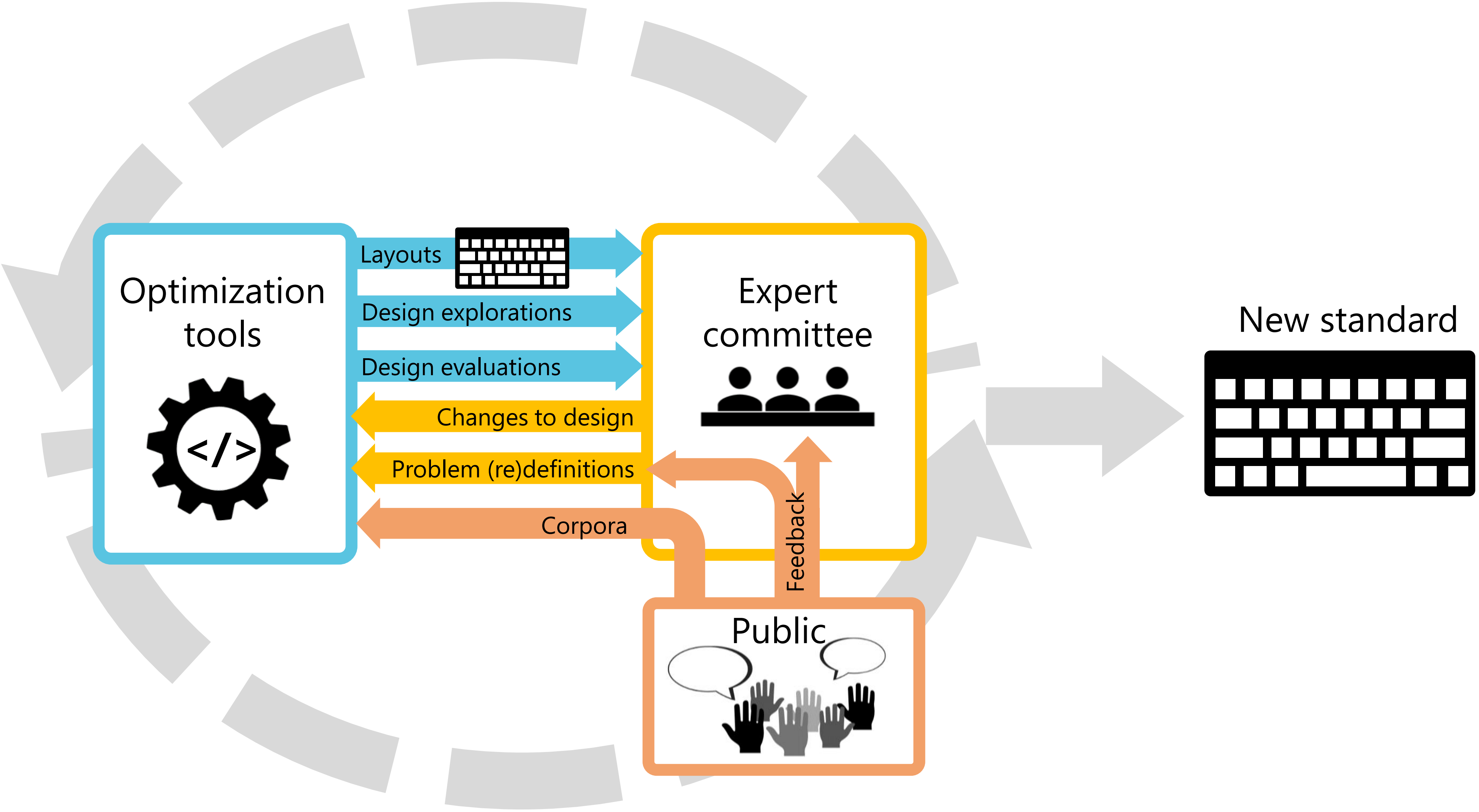
Respect public comments



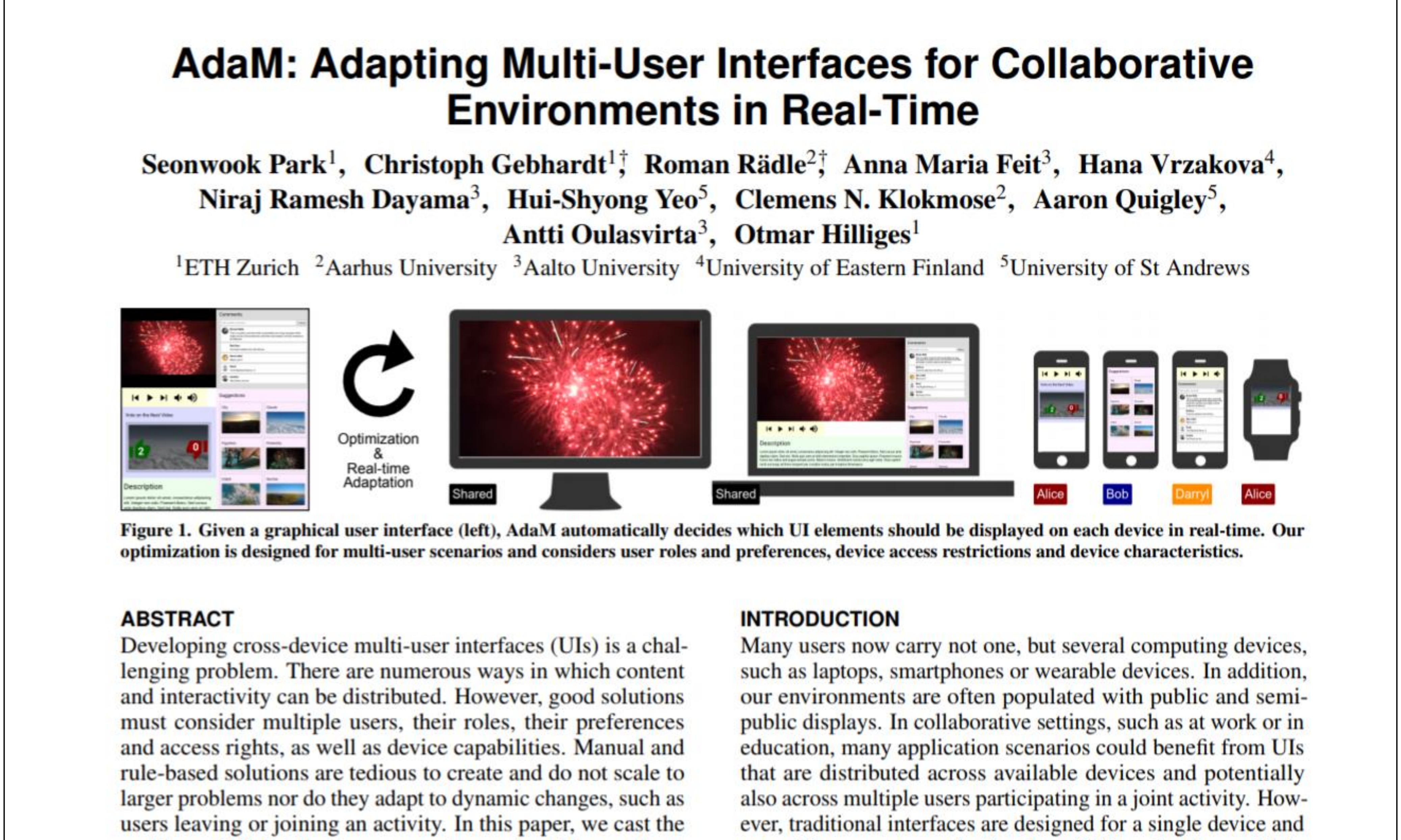
Easy to learn: familiar position and groups of similar characters

Based on a mapping within 2% of global optimum

Iterative & Participatory Optimization Process



Outcome of the 2017 Hackathon!



AdaM: Adapting Multi-User Interfaces for Collaborative Environments in Real-Time

Seonwook Park[1], Christoph Gebhardt[1]†, Roman Rädle[2]†, Anna Maria Feit[3], Hana Vrzakova[4], Niraj Ramesh Dayama[3], Hui-Shyong Yeo[5], Clemens N. Klokmose[2], Aaron Quigley[5], Antti Oulasvirta[3], Otmar Hilliges[1]

[1]ETH Zurich [2]Aarhus University [3]Aalto University [4]University of Eastern Finland [5]University of St Andrews

Figure 1. Given a graphical user interface (left), AdaM automatically decides which UI elements should be displayed on each device in real-time. Our optimization is designed for multi-user scenarios and considers user roles and preferences, device access restrictions and device characteristics.

ABSTRACT

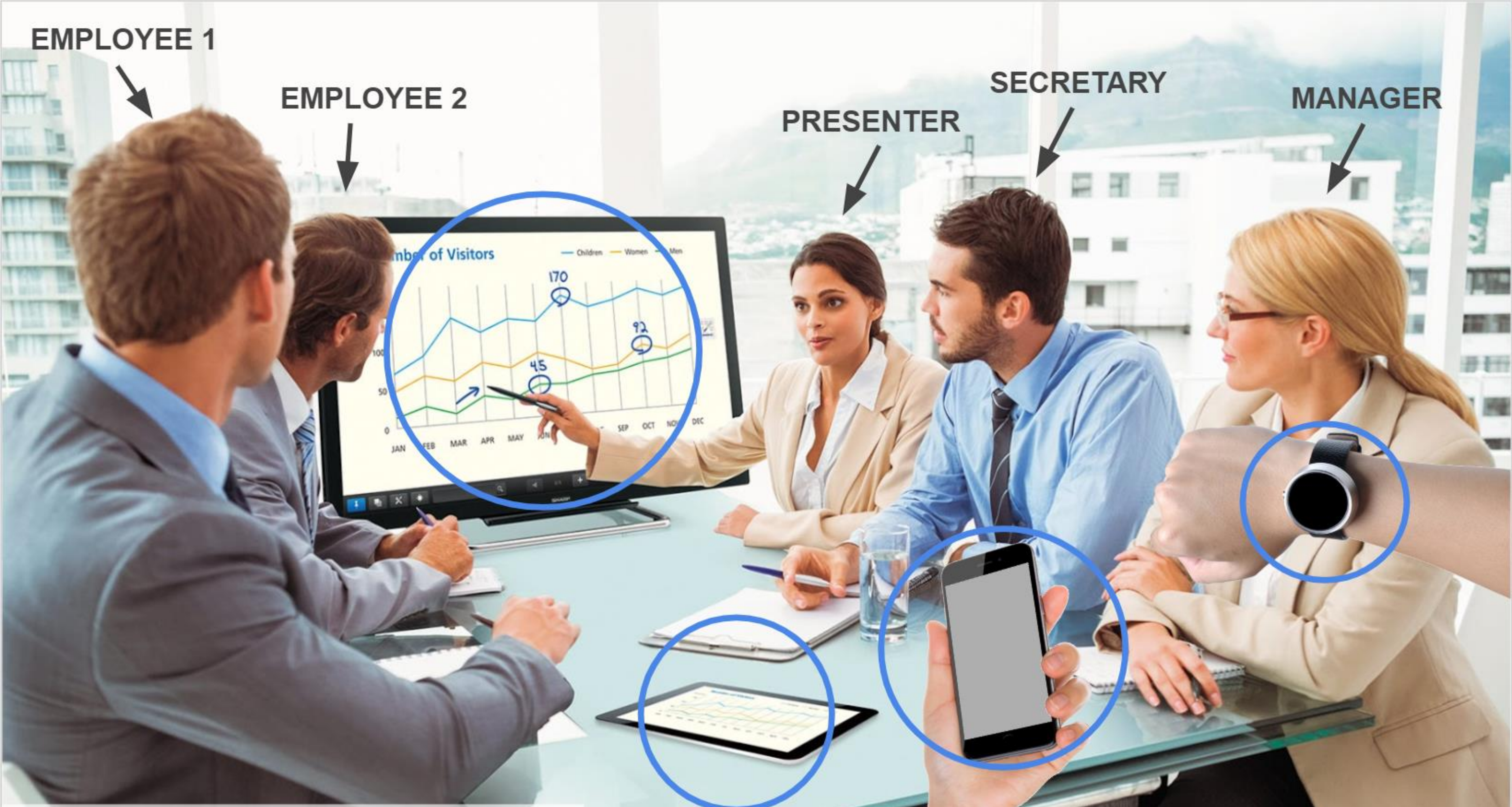
Developing cross-device multi-user interfaces (UIs) is a challenging problem. There are numerous ways in which content and interactivity can be distributed. However, good solutions must consider multiple users, their roles, their preferences and access rights, as well as device capabilities. Manual and rule-based solutions are tedious to create and do not scale to larger problems nor do they adapt to dynamic changes, such as users leaving or joining an activity. In this paper, we cast the

INTRODUCTION

Many users now carry not one, but several computing devices, such as laptops, smartphones or wearable devices. In addition, our environments are often populated with public and semi-public displays. In collaborative settings, such as at work or in education, many application scenarios could benefit from UIs that are distributed across available devices and potentially also across multiple users participating in a joint activity. However, traditional interfaces are designed for a single device and

Multi-device User interfaces

https://www.ofcom.org.uk/__data/assets/image/0016/13039/modern-family-blank-lo-res.jpg



EMPLOYEE 1

EMPLOYEE 2

PRESENTER

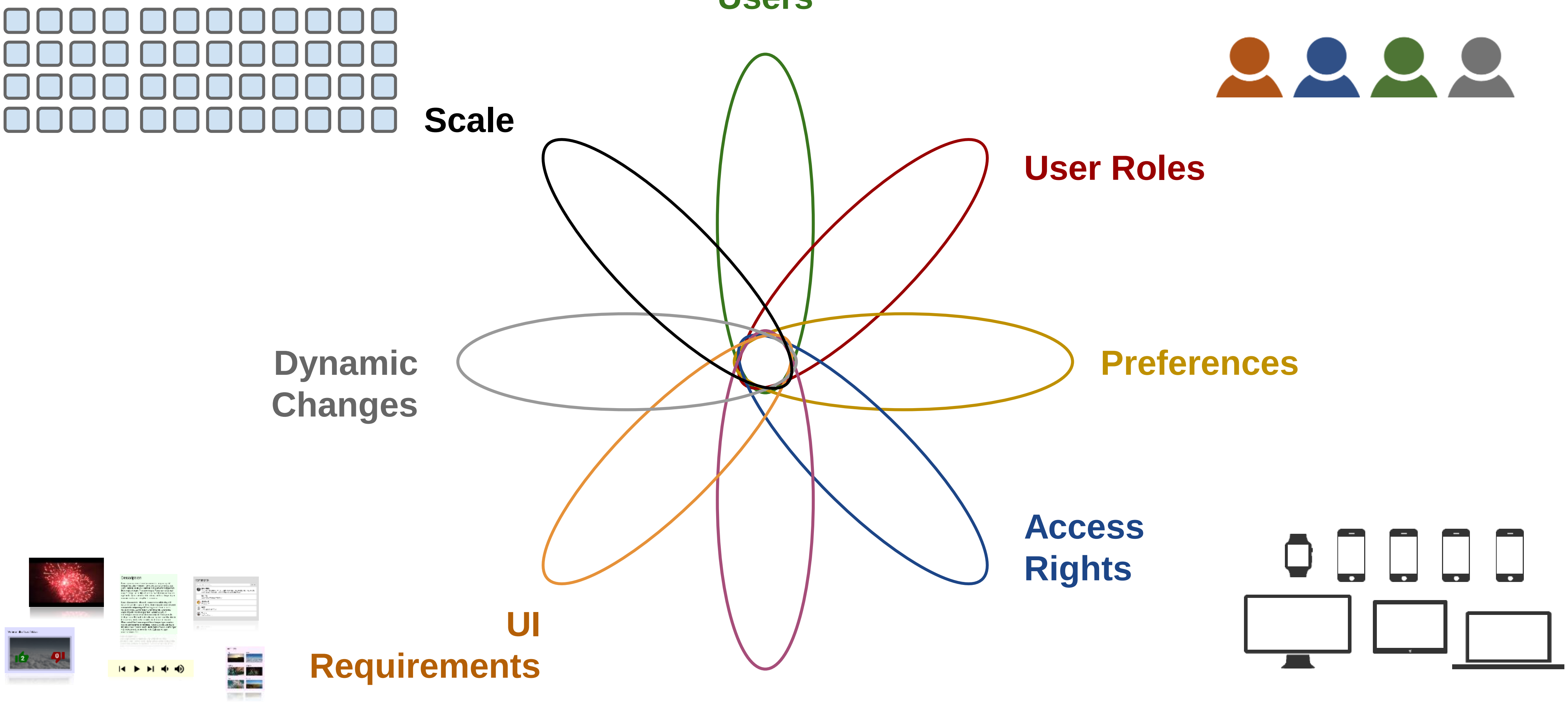
SECRETARY

MANAGER

https://www.kpms.co.uk/wp-content/uploads/2017/03/PN-40TC1_huddle-meeting1_i522258271.jpg

What to show on which screen?

Many requirements



Multiple Users

User Roles

Preferences

Access Rights

Device Capabilities

UI Requirements

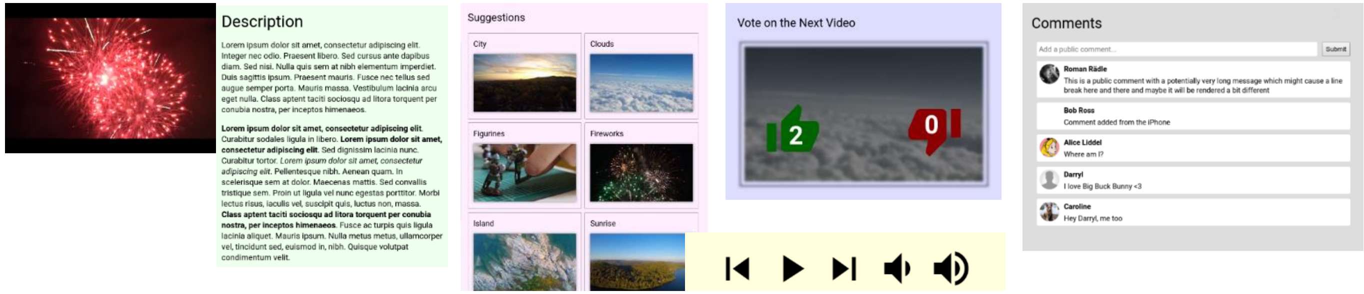
Dynamic Changes

Scale

Design space

Elements

$e \in \mathcal{E}$



Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer nec odio. Praesent libero. Sed cursus ante dapibus diam. Sed nisi. Nulla quis sem at nibh elementum imperdiet. Duis sagittis ipsum. Praesent mauris. Fusce nec tellus sed augue semper porta. Mauris massa. Vestibulum lacinia arcu eget nulla. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur sodales ligula in libero. **Lorem ipsum dolor sit amet, consectetur adipiscing elit**. Sed dignissim lacinia nunc. Curabitur tortor. *Lorem ipsum dolor sit amet, consectetur adipiscing elit*. Pellentesque nibh. Aenean quam. In scelerisque sem at dolor. Maecenas mattis. Sed convallis tristique sem. Proin ut ligula vel nunc egestas porttitor. Morbi lectus risus, iaculis vel, suscipit quis, luctus non, massa. **Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos**. Fusce ac turpis quis ligula lacinia aliquet. Mauris ipsum. Nulla metus metus, ullamcorper vel, tincidunt sed, euismod in, nibh. Quisque volutpat condimentum velit.

Suggestions

Vote on the Next Video

Comments

Add a public comment... Submit

Roman Rädle
This is a public comment with a potentially very long message which might cause a line break here and there and maybe it will be rendered a bit different

Bob Ross
Comment added from the iPhone

Alice Liddel
Where am I?

Darryl
I love Big Buck Bunny <3

Caroline
Hey Darryl, me too

Devices

$d \in \mathcal{D}$

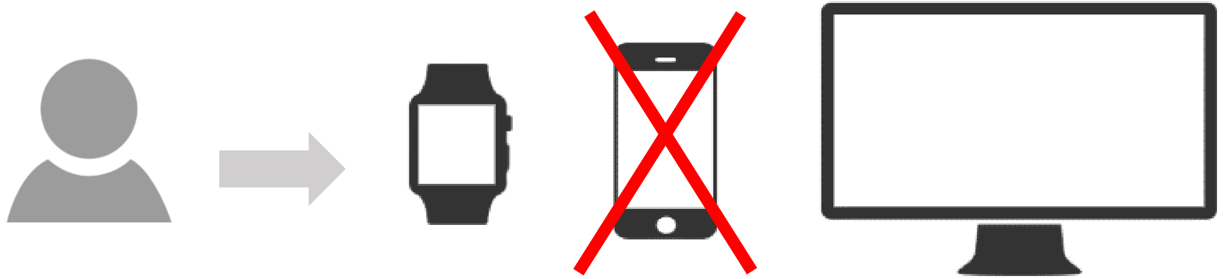
Users

$u \in \mathcal{U}$

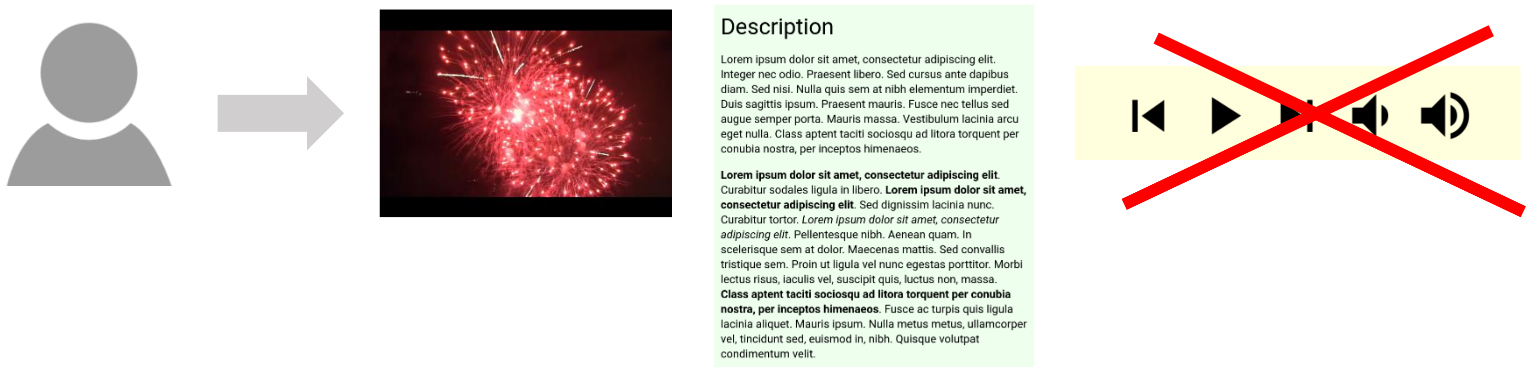
$$x_{ed} = \begin{cases} 1, \text{if } e \text{ is assigned to } d \\ 0, \text{otherwise} \end{cases} \qquad o_{eu} = \begin{cases} 1, \text{if u has access to e} \\ 0, \text{otherwise} \end{cases} \qquad s_{ed} \in \mathbb{Z}^{+}$$

Constraints

- Device access



- Element permission

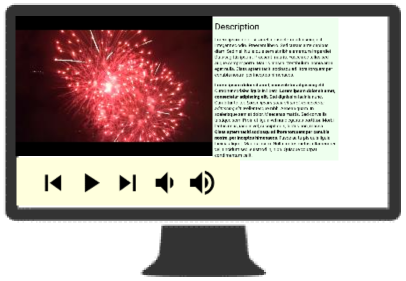


Description

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer nec odio. Praesent libero. Sed cursus ante dapibus diam. Sed nisi. Nulla quis sem at nibh elementum imperdiet. Duis sagittis ipsum. Praesent mauris. Fusce nec tellus sed augue semper porta. Mauris massa. Vestibulum lacinia arcu eget nulla. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Curabitur sodales ligula in libero. **Lorem ipsum dolor sit amet, consectetur adipiscing elit**. Sed dignissim lacinia nunc. Curabitur tortor. *Lorem ipsum dolor sit amet, consectetur adipiscing elit*. Pellentesque nibh. Aenean quam. In scelerisque sem at dolor. Maecenas mattis. Sed convallis tristique sem. Proin ut ligula vel nunc egestas porttitor. Morbi lectus risus, iaculis vel, suscipit quis, luctus non, massa. **Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos**. Fusce ac turpis quis ligula lacinia aliquet. Mauris ipsum. Nulla metus metus, ullamcorper vel, tincidunt sed, euismod in, nibh. Quisque volutpat condimentum velit.

- Device capacity

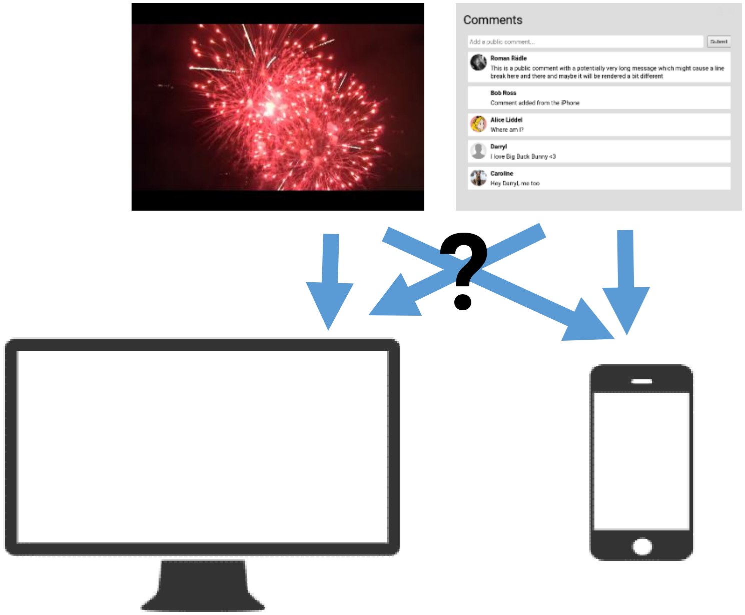


Objectives

Assignment Quality (Usability)

More important elements should:

- better match element requirements and device characteristics
  (visual quality, text input, touch pointing, mouse pointing)
- be displayed larger



Roman Rädle
This is a public comment with a potentially very long message which might cause a line break here and there and maybe it will be rendered a bit different

Bob Ross
Comment added from the iPhone

Alice Liddel
Where am I?

Darryl
I love Big Buck Bunny <3

Caroline
Hey Darryl, me too

Objectives

Assignment Quality (Usability)

More important elements should:

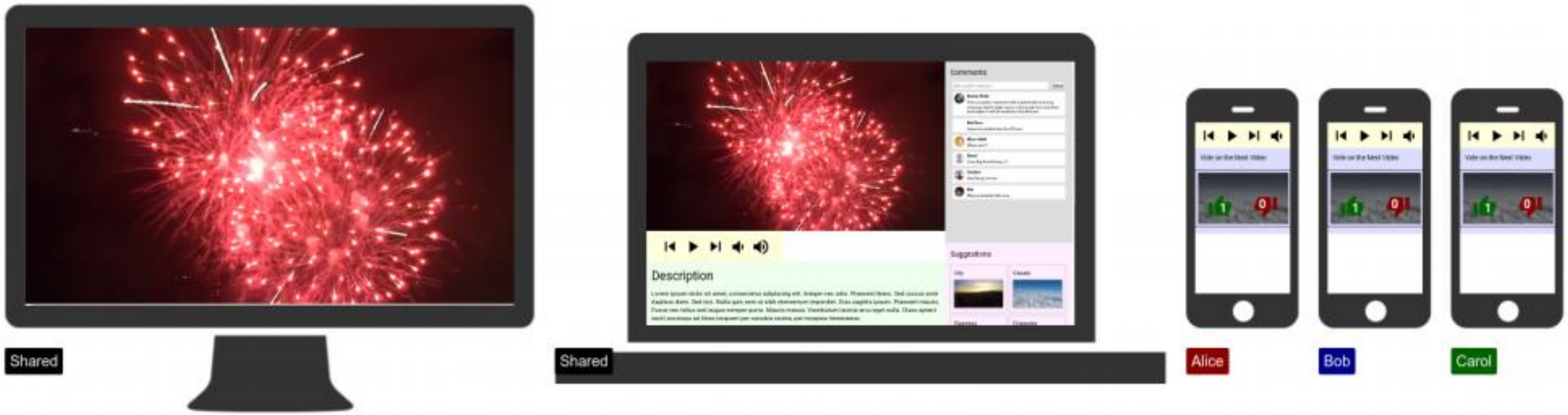
- better match element requirements and device characteristics
  (visual quality, text input, touch pointing, mouse pointing)
- be displayed larger

Completeness (Usefulness)

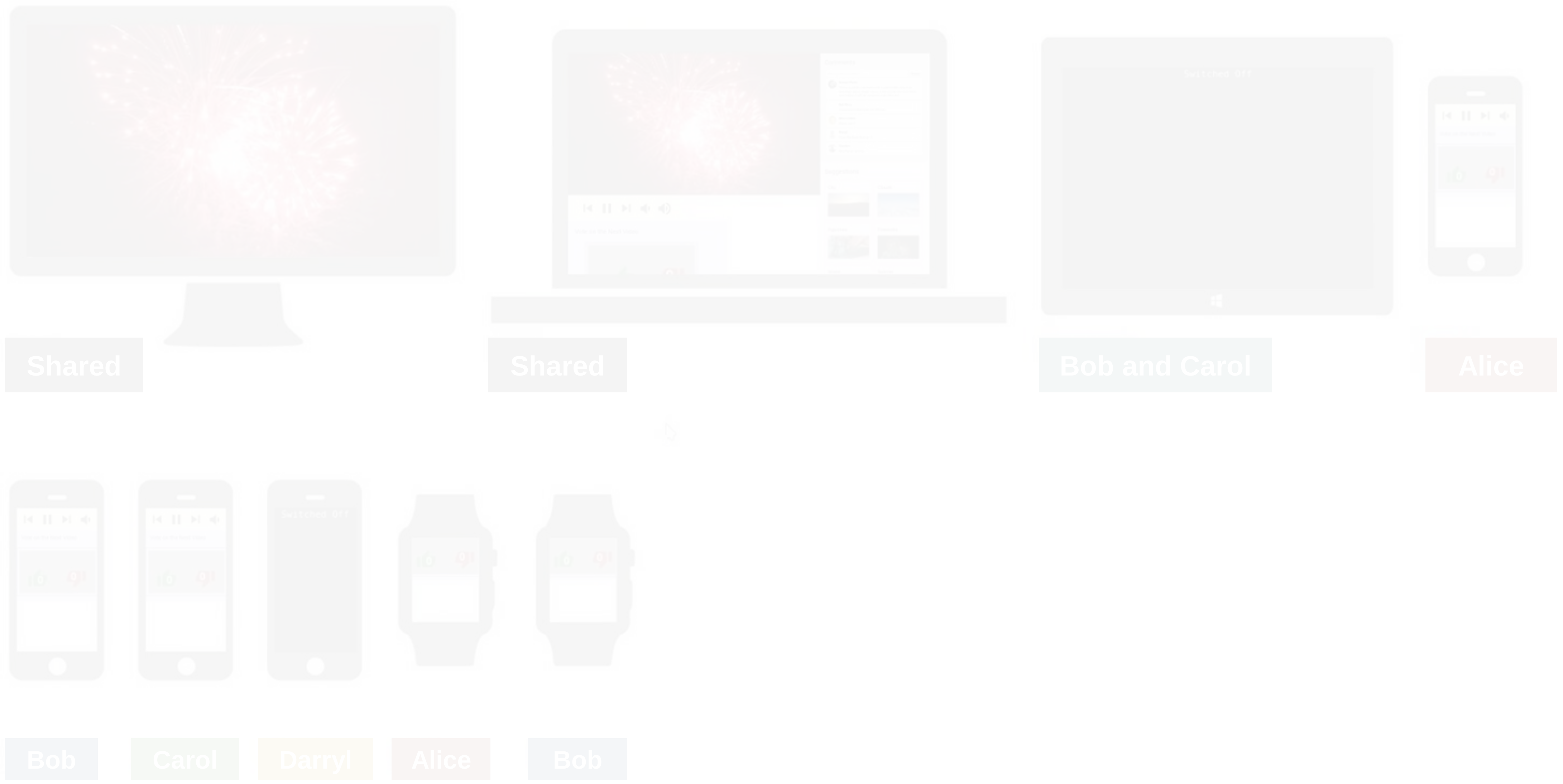
Maximize

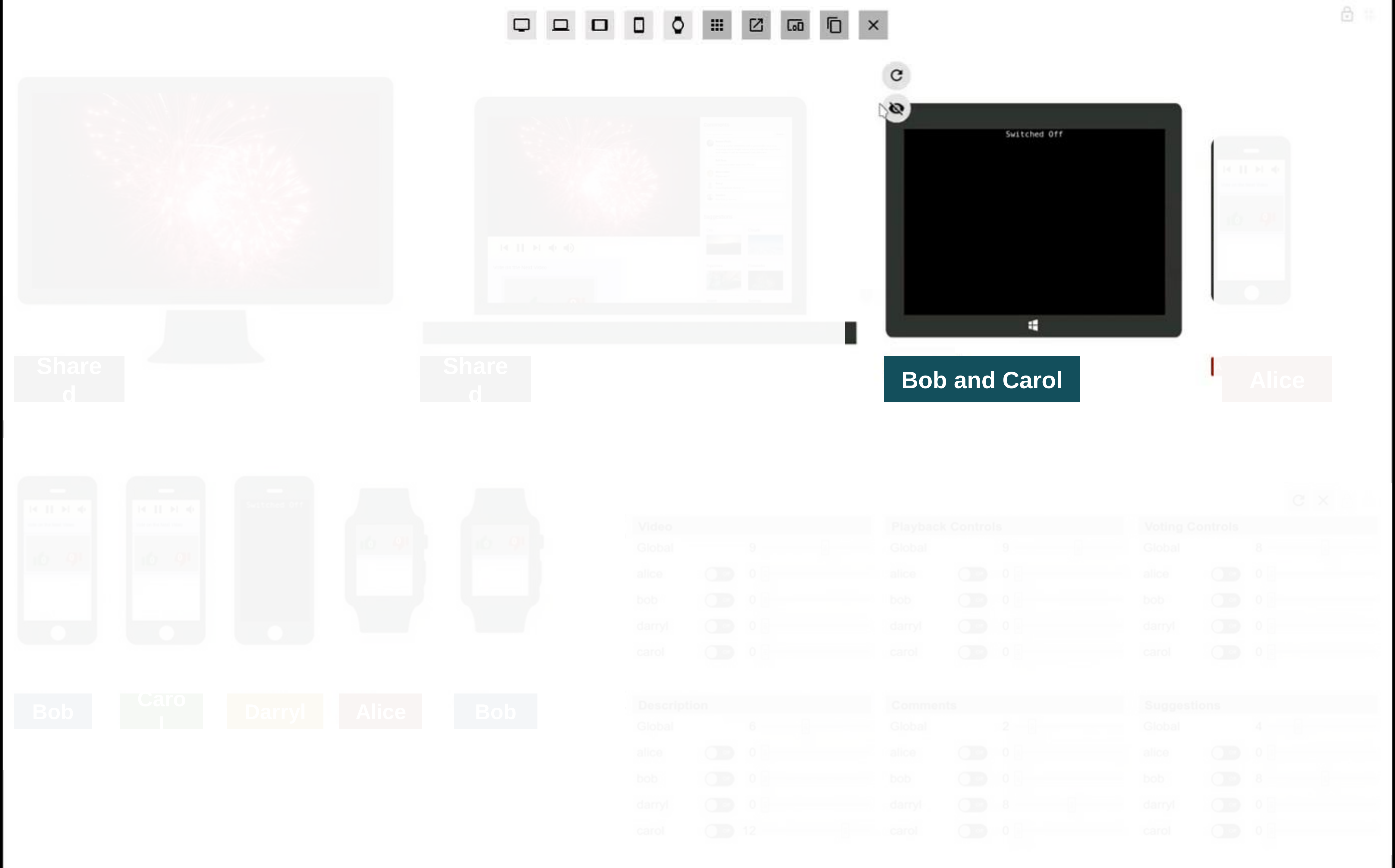
- the number of elements a user has access to and
- the access of users to elements they need (have permission to use)

AdaM: Automatic distribution of elements



9 DEVICES (2 PUBLIC, 1 SHARED, 6 PRIVATE)





Video

Global		9
alice	○	0
bob	○	0
darryl	○	0
carol	○	0

Playback Controls

Global		9
alice	○	0
bob	○	0
darryl	○	0
carol	○	0

Voting Controls

Global		8
alice	○	0
bob	○	0
darryl	○	0
carol	○	0

Description

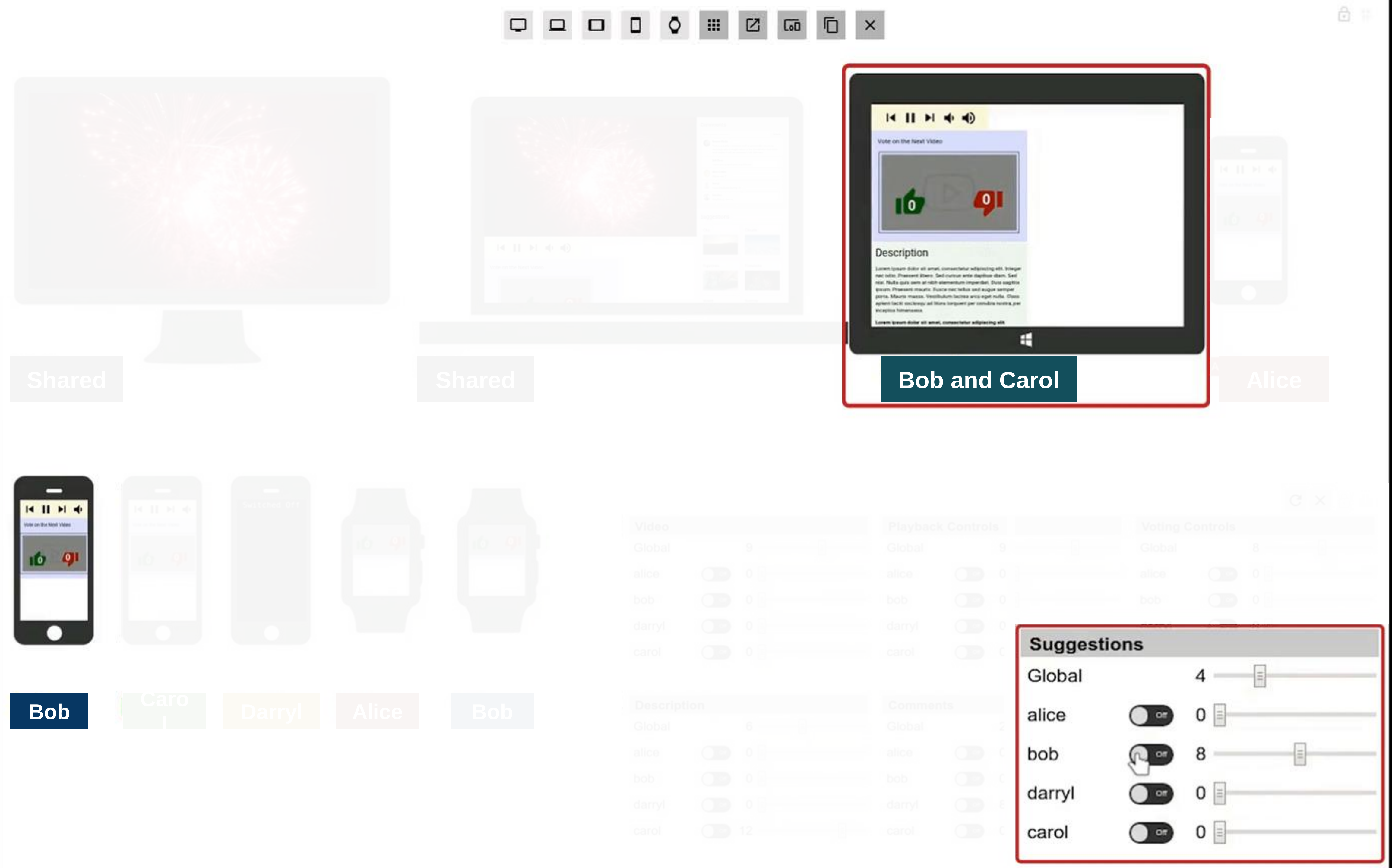
Global		6
alice	○	0
bob	○	0
darryl	○	0
carol	○	12

Comments

Global		2
alice	○	0
bob	○	0
darryl	○	8
carol	○	0

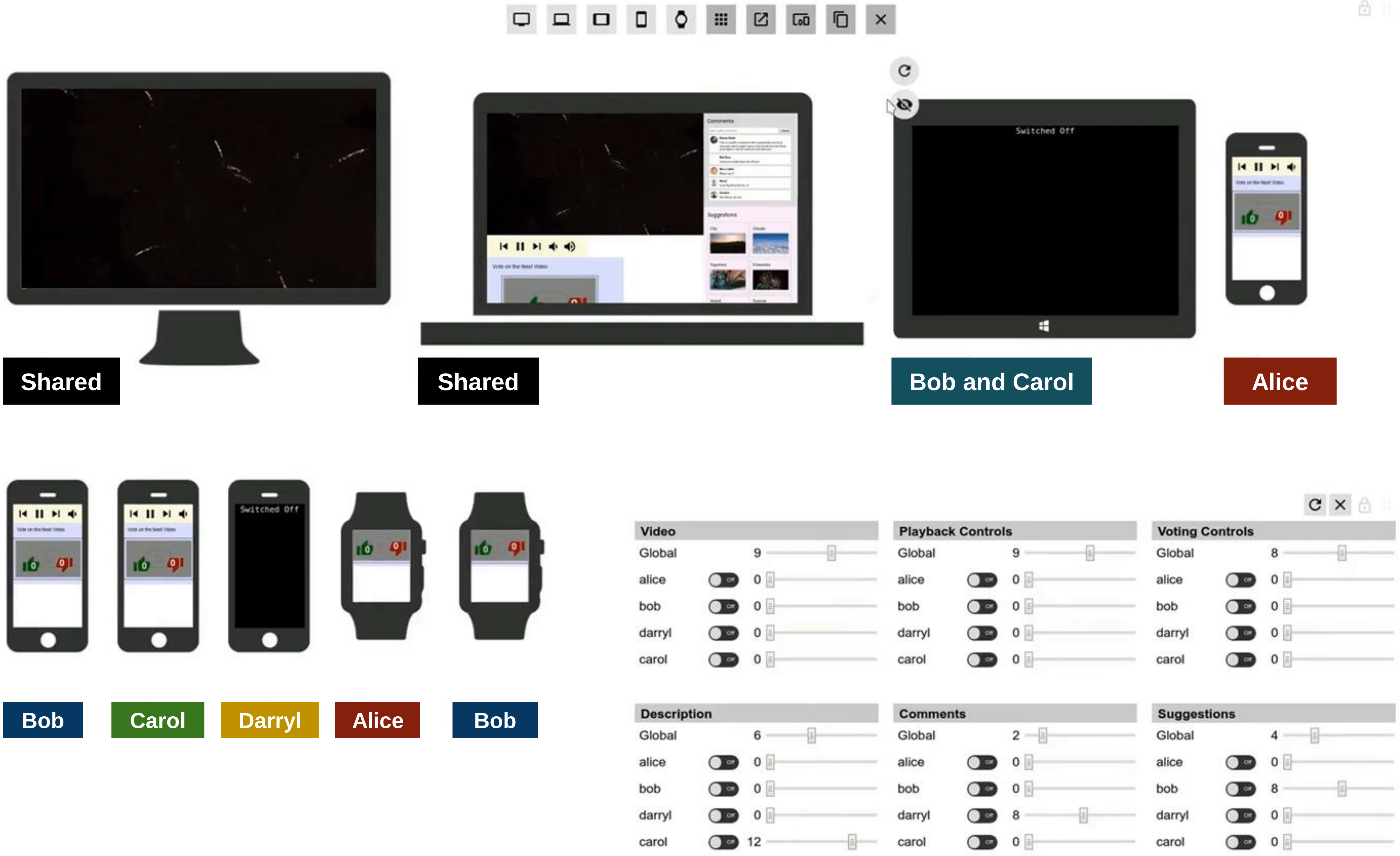
Suggestions

Global		4
alice	○	0
bob	○	8
darryl	○	0
carol	○	0



Suggestions

Global		4	
alice	Off	0	
bob	Off	8	
darryl	Off	0	
carol	Off	0	



Shared

Shared

Bob and Carol

Alice

Bob

Carol

Darryl

Alice

Bob

Video			Playback Controls			Voting Controls		
Global		9	Global		9	Global		8
alice	Off	0	alice	Off	0	alice	Off	0
bob	Off	0	bob	Off	0	bob	Off	0
darryl	Off	0	darryl	Off	0	darryl	Off	0
carol	Off	0	carol	Off	0	carol	Off	0

Description			Comments			Suggestions		
Global		6	Global		2	Global		4
alice	Off	0	alice	Off	0	alice	Off	0
bob	Off	0	bob	Off	0	bob	Off	8
darryl	Off	0	darryl	Off	8	darryl	Off	0
carol	Off	12	carol	Off	0	carol	Off	0

AdaM

ait.ethz.ch/projects/2018/adam
Including code and tool

- Complex interaction of many constraints and objectives
- Scales to thousands of users and devices
- Flexible and adaptive (to new users, devices, roles etc.)
- Easy customization for end users