Ability-Based Design: Concept, Principles, and Examples

Jacob O. Wobbrock, Shaun K. Kane, Krzysztof Z. Gajos, Susumu Harada, and Jon Froehlich ACM Trans. Access. Comput., 3:9:1-9:27, April 2011

JACOB O. WOBBROCK, SHAUN K. KANE The Information School, University of Washington

KRZYSZTOF Z. GAJOS Harvard School of Engineering and Applied Sciences

SUSUMU HARADA, JON FROEHLICH Department of Computer Science & Engineering, University of Washington

Ability

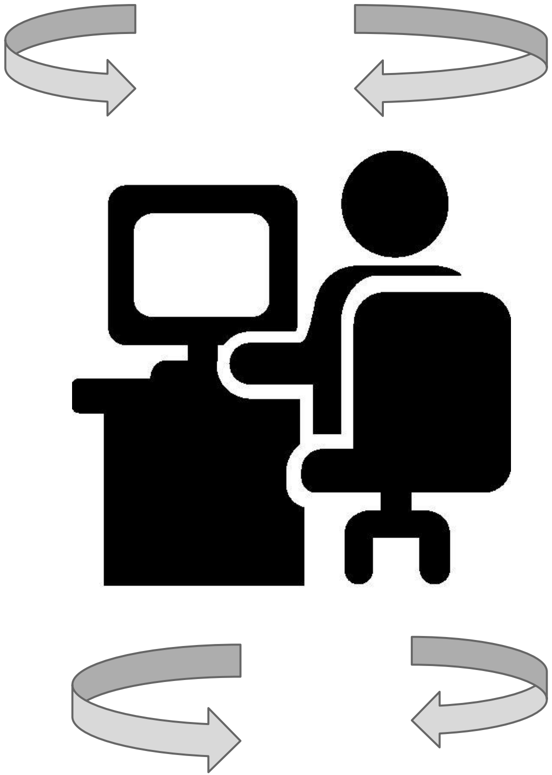
Ability is context-dependant for everyone

Environment

- Temperature
- Light
- Noise
- Weather
- Terrain

User

- Preferences
- Goals and Tasks
- Physical state
- Orientation
- Emotional state



Social

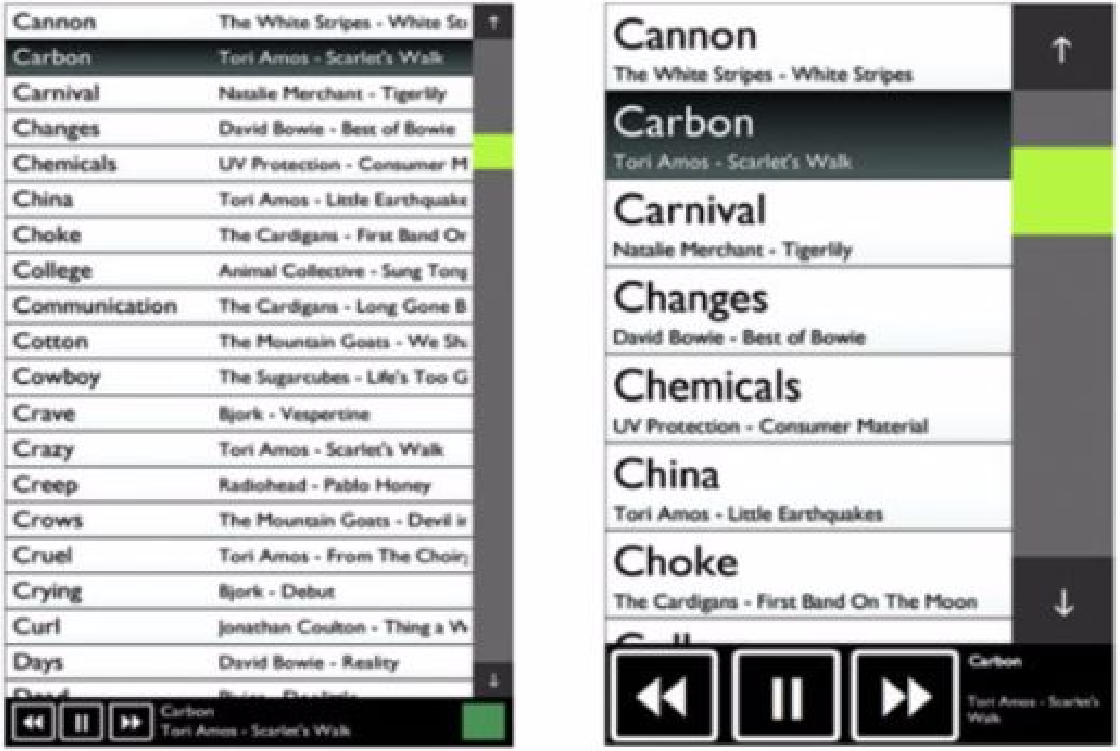
- Distraction
- Collaboration
- Confinement
- Clothing

Technology / Device

- Screen resolution
- Connectivity
- Speed
- Browser
- Battery

Walking User Interfaces

Users are “situationally impaired” while walking (Yambe and Takahashi 2007) - *Re: “getting off the treadmill”*



Cannon	The White Stripes - White St
Carbon	Tori Amos - Scarlet's Walk
Carnival	Natalie Merchant - Tigerlily
Changes	David Bowie - Best of Bowie
Chemicals	UV Protection - Consumer M
China	Tori Amos - Little Earthquake
Choke	The Cardigans - First Band On
College	Animal Collective - Sung Tong
Communication	The Cardigans - Long Gone B
Cotton	The Mountain Goats - We Sh
Cowboy	The Sugarcubes - Life's Too G
Crave	Bjork - Vespertine
Crazy	Tori Amos - Scarlet's Walk
Creep	Radiohead - Pablo Honey
Crows	The Mountain Goats - Devil i
Cruel	Tori Amos - From The Choir
Crying	Bjork - Debut
Curl	Jonathan Coulton - Thing a W
Days	David Bowie - Reality

Carbon
Tori Amos - Scarlet's Walk

Cannon
The White Stripes - White Stripes

Carbon
Tori Amos - Scarlet's Walk

Carnival
Natalie Merchant - Tigerlily

Changes
David Bowie - Best of Bowie

Chemicals
UV Protection - Consumer Material

China
Tori Amos - Little Earthquakes

Choke
The Cardigans - First Band On The Moon

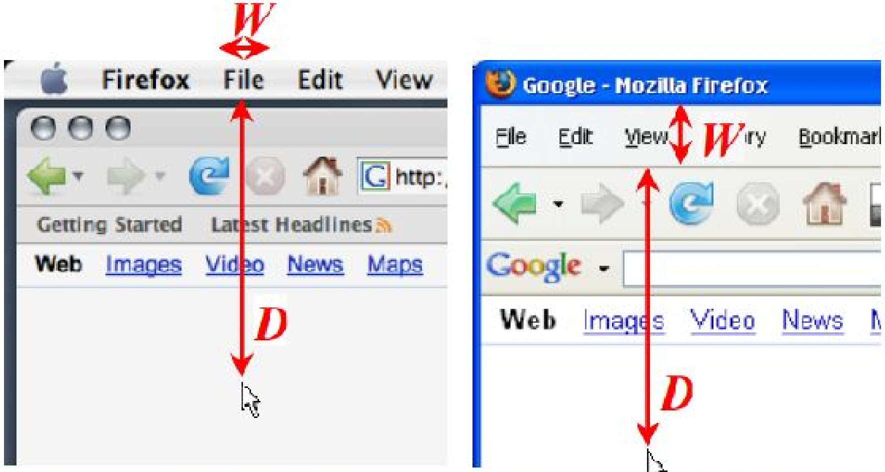
Carbon
Tori Amos - Scarlet's Walk

Figure 2. Our music player user interface in two sizes. (*left*) The player while standing; (*right*) the player while walking.

Barrier Pointing

[Froehlich et al. 2007)

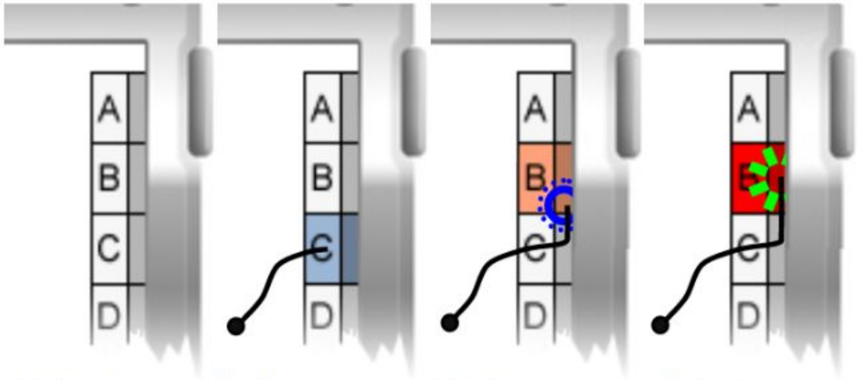
How to provide stability for a user to press a stylus against a screen, particularly **during motion**?



(a) The Apple Macintosh file menu bar is flush against the screen edge.

(b) The Microsoft Windows file menu bar is below the title bar and away from the screen edge.

Barrier Targets / Widgets



(a) *Normal*

(b) *Hover* – Stylus over target *C*

(c) *Select* – Stroke into *B*'s selection trough

(d) *Confirm* – Lift stylus while within selected target *B*.

Barrier Interaction Techniques

“Design for One”

Adaptivity: How systems and interfaces are made user-customizable to to incorporate user’s preferences

Adaptability: How systems and interfaces are made to adapt to our abilities. For example through sensing and performance modeling.

Discussion:

In what ways does designing with an ability-based approach help all users, not just those with impairments?

POSITIVES

TUSHAR
"Situation aware" systems. [...] might be helpful for the unimpaired in special cases where "Ability" changes drastically under stress or duress.

MICHELLE
Designing with an ability-based approach helps all users because it changes the mindset of designers.

NEGATIVES

FRANCESCO

I think that in the majority of the cases the best design always includes a “perfect match” between hardware and software. In the past [..] there was a battle between hardware engineers that want to fix all the problems by improving the hardware and software engineers that want to solve them by pushing on the software.

CALVIN

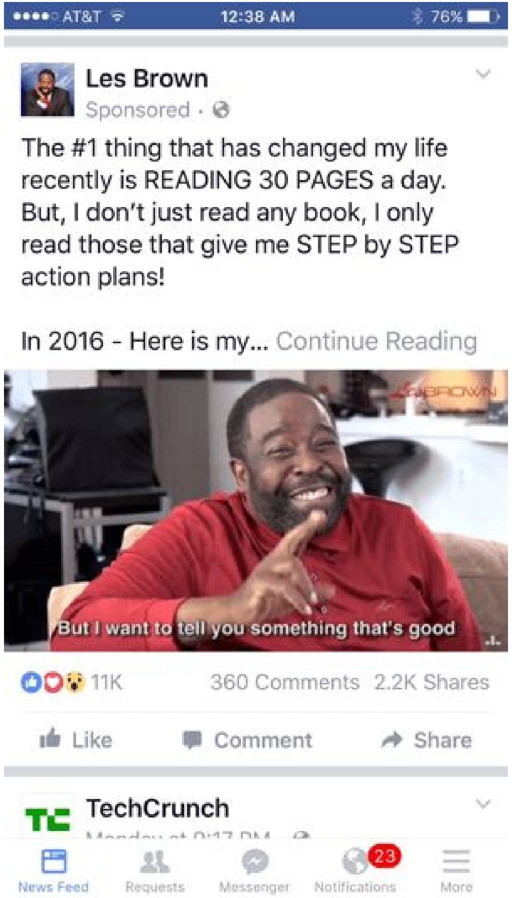
A problem I noticed with this "design-for-one" methodology is that it [...] is not always feasible from a technology standpoint. While computation is only getting cheaper by the day, the speed of the process of capturing, cleaning, analysing and spitting out insights is of concern.

Win-win: Muting and Captioning Advertising Videos

80% of people **react negatively when mobile video ads play sound** in the feed

Facebook will **automatically caption video ads**, so videos can be played **but with the sound off**

<u>Captioning ads</u> **<u>makes them more accessible for everyone</u>,** while increasing view time by 12% in a Facebook study (!)



Les Brown
Sponsored ·

The #1 thing that has changed my life recently is READING 30 PAGES a day. But, I don't just read any book, I only read those that give me STEP by STEP action plans!

In 2016 - Here is my... Continue Reading

But I want to tell you something that's good

TechCrunch
Monday at 9:17 PM

Dustin: “...an interface designed with blind users in mind can also be useful for users with able vision who, at the end of a long day of staring at a screen, may benefit from an interface which does not require sight…”

Ability-based systems may both adapt to user’s abilities, or be adapted or customized directly by the user

(Edwards 1995).

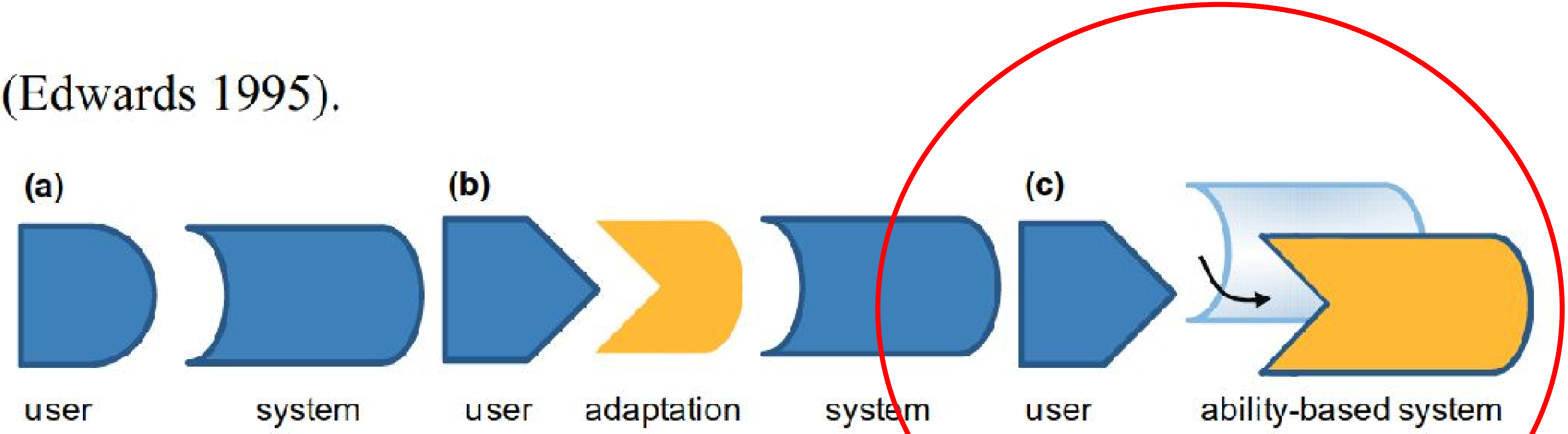


Figure 2. (a) A user whose abilities match those presumed by the system. **(b)** A user whose abilities do not match those presumed by the system. Because the system is inflexible, the user must be adapted to it. **(c)** An ability-based system is designed to accommodate the user’s abilities. It may adapt or be adapted to them. Our symbols are based on those from prior work (Edwards 1995).

“No more than 60% of people who indicate a need for access technologies actually use them…”

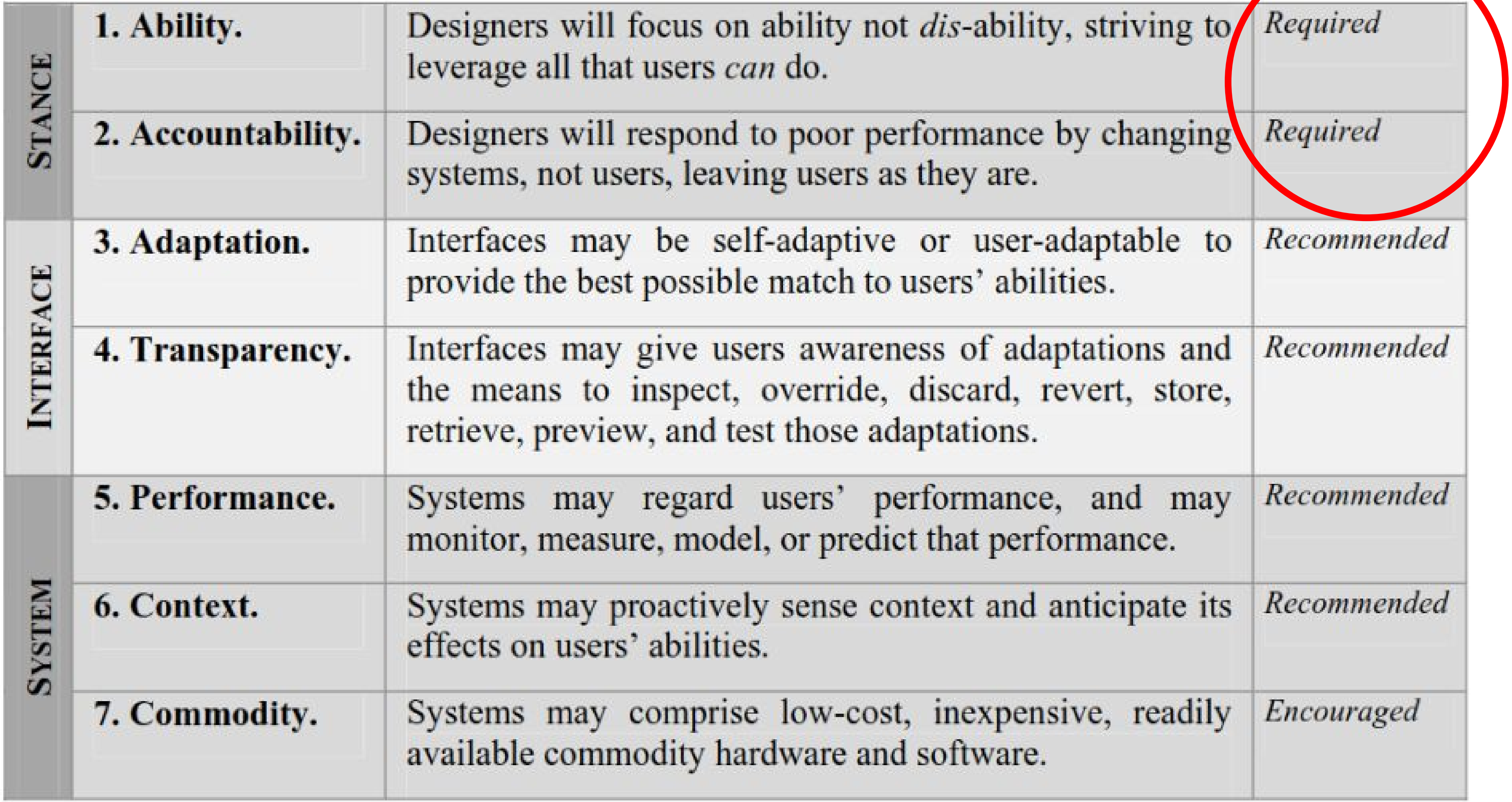
(Fichten et al. 2000)

Why? . . .

Reasons for abandonment include:

- Procurement and maintenance costs
- Configuration time
- Compatibility (or lack of)
- Stigmatizing or humiliating aspects
- Painstaking effort of use (learning cost)

Seven Principles of Ability-Based Design

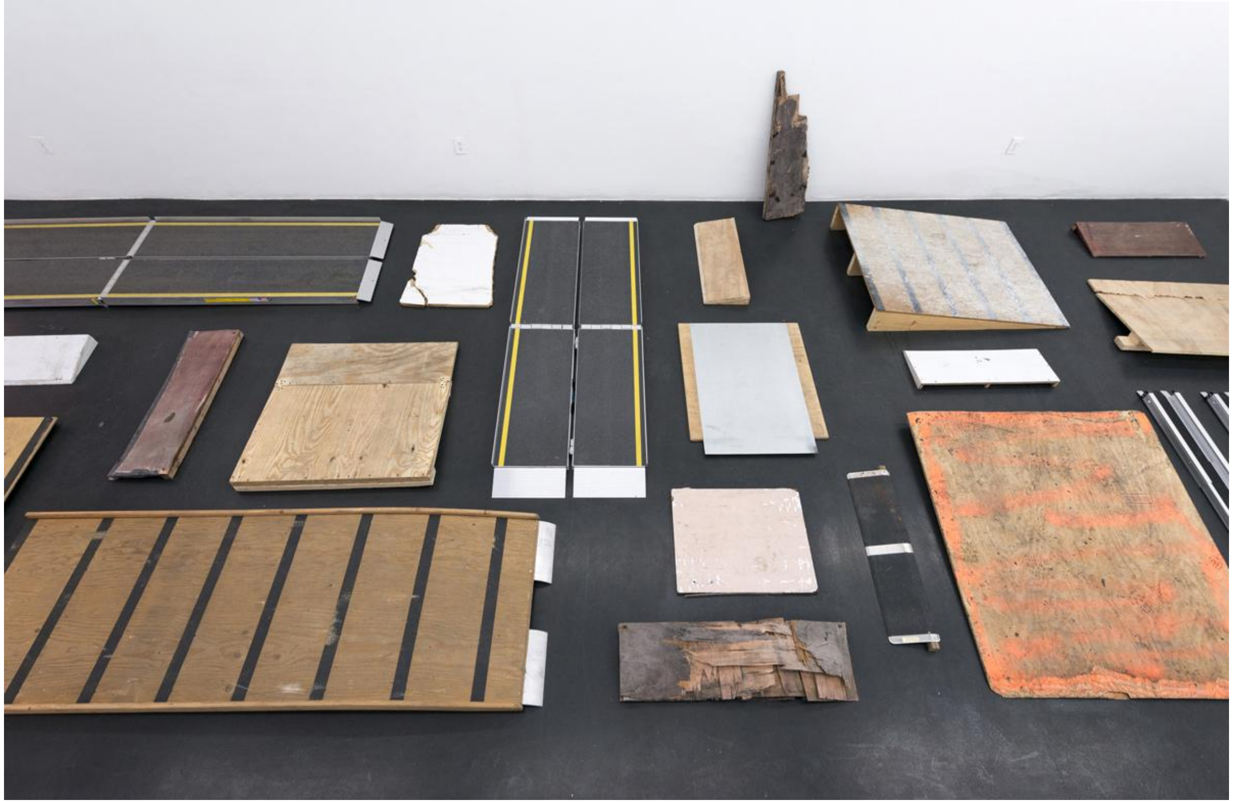


STANCE	**1. Ability.**	Designers will focus on ability not *dis*-ability, striving to leverage all that users *can* do.	*Required*
	2. Accountability.	Designers will respond to poor performance by changing systems, not users, leaving users as they are.	*Required*
INTERFACE	**3. Adaptation.**	Interfaces may be self-adaptive or user-adaptable to provide the best possible match to users' abilities.	*Recommended*
	4. Transparency.	Interfaces may give users awareness of adaptations and the means to inspect, override, discard, revert, store, retrieve, preview, and test those adaptations.	*Recommended*
SYSTEM	**5. Performance.**	Systems may regard users' performance, and may monitor, measure, model, or predict that performance.	*Recommended*
	6. Context.	Systems may proactively sense context and anticipate its effects on users' abilities.	*Recommended*
	7. Commodity.	Systems may comprise low-cost, inexpensive, readily available commodity hardware and software.	*Encouraged*

Prior Approaches

Assistive Technology
Rehabilitation Engineering (non HCI specific)
Universal Design
Universal Usability
Design for All
User Interfaces for All
Inclusive Design
Extra-Ordinary Human-Computer Interaction

Assistive Technology



Park MacArthur.
Ramps (2013)

How the SUPPLE system works

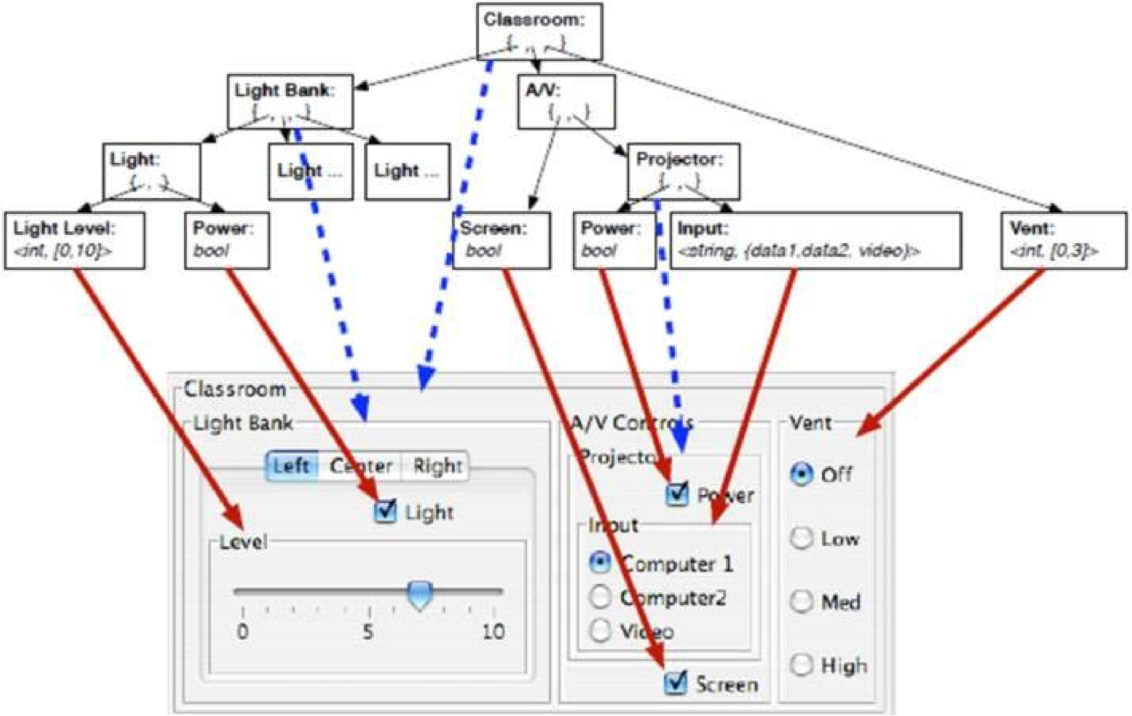
[Gajos and Weld, IUI ‘04]

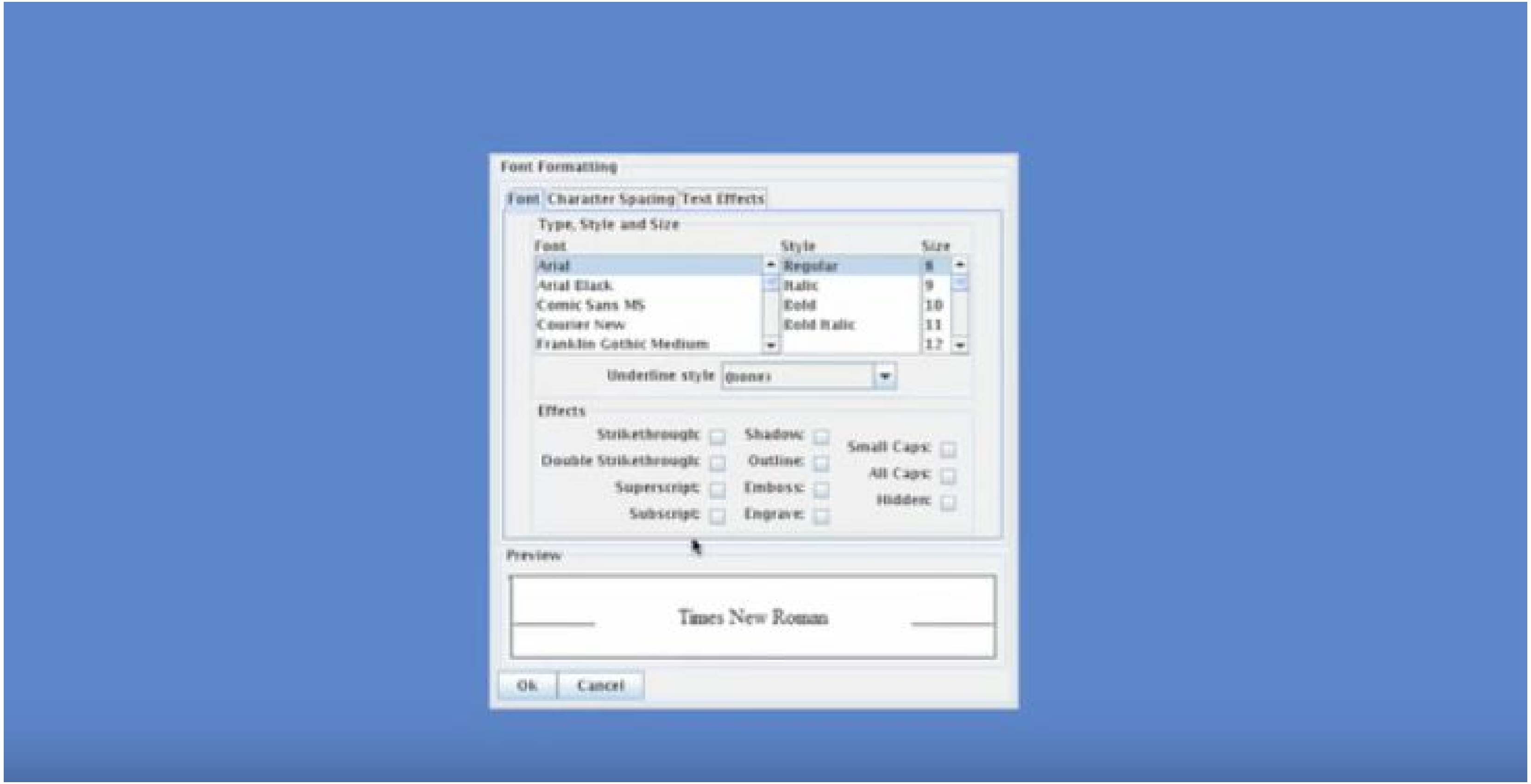
The optimization algorithm **automatically adapts user interfaces** using a predictive model **based on a user’s performance**, and dexterity in tasks such as pointing, dragging, list selections, and multiple clicks

SUPPLE

[Gajos and Weld, IUI ‘04]

User behavior is measured and then used to automatically customize interfaces which render to the user's usage pattern to make accurate typing easier





Font Formatting

Font | Character Spacing | Text Effects

Type, Style and Size

Font: Arial
- Arial
- Arial Black
- Comic Sans MS
- Courier New
- Franklin Gothic Medium

Style: Regular
- Regular
- Italic
- Bold
- Bold Italic

Size: 8
- 8
- 9
- 10
- 11
- 12

Underline style: (none)

Effects

Strikethrough: □ Shadow: □ Small Caps: □
Double Strikethrough: □ Outline: □ All Caps: □
Superscript: □ Emboss: □ Hidden: □
Subscript: □ Engrave: □

Preview

Times New Roman

https://www.youtube.com/watch?v=B63whNtp4qc :45 second

Discussion:

If we are able to generate predictive models based on a user’s performance, is it better to make **local or global adaptations** to the interface? **What are the trade-offs?**

To what extent is it important **to ask the user to approve a customization** (perhaps in the form of a preview), that has been generated based on performance assessments?

Takeaways

- Interface generation treated as an optimization problem
- Performance evaluation is a form of customization
- Usefulness of heuristics to find a solution

Outstanding questions

- How do we model abilities and impairments?
- How often are abilities measured and how?
- How do we map certain abilities to interface changes?
- How can the automatic generation of interfaces based on performance evaluation also be customized by users?

Example-Centric Programming: Integrating Web Search into the Development Environment

...

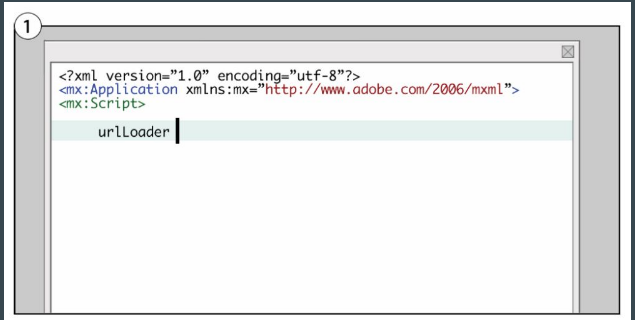
Joel Brandt, Mira Dontcheva, Marcos Weskamp, Scott R. Klemmer

Scenario

Suppose you need to retrieve power-usage data from a Web service, using Adobe Flex builder.

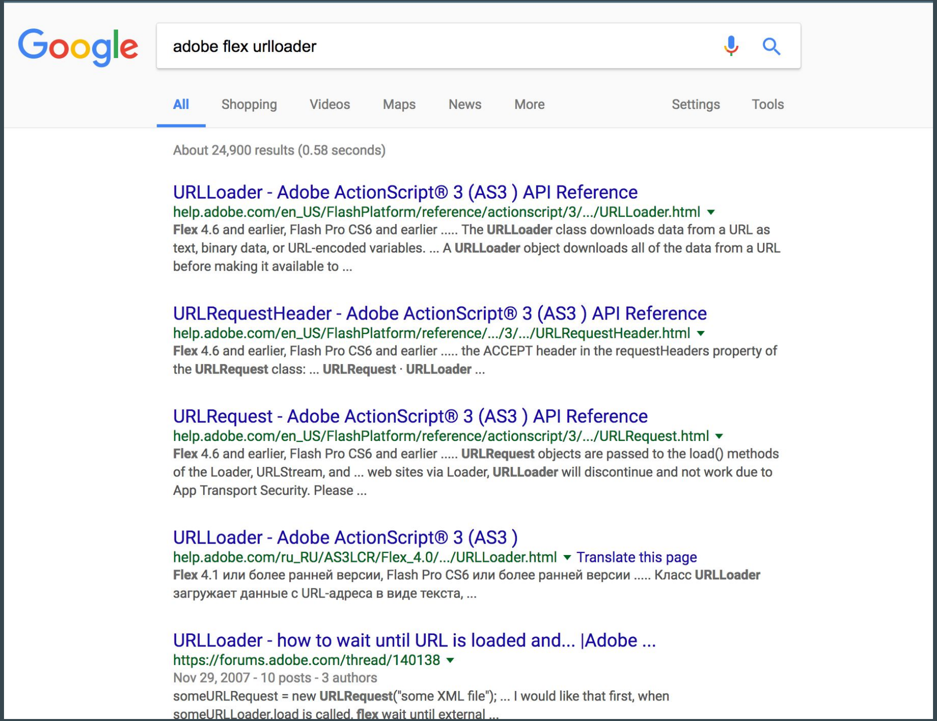
Method contains “URL” ? => urlLoader => Can’t remember the usage

Problems with this approach?



```
<?xml version=”1.0” encoding=”utf-8”?>
<mx:Application xmlns:mx=”http://www.adobe.com/2006/mxml”>
<mx:Script>

      urlLoader
```



adobe flex urlloader

About 24,900 results (0.58 seconds)

URLLoader - Adobe ActionScript® 3 (AS3) API Reference
help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/.../URLLoader.html ▾
Flex 4.6 and earlier, Flash Pro CS6 and earlier ..... The **URLLoader** class downloads data from a URL as text, binary data, or URL-encoded variables. ... A **URLLoader** object downloads all of the data from a URL before making it available to ...

URLRequestHeader - Adobe ActionScript® 3 (AS3) API Reference
help.adobe.com/en_US/FlashPlatform/reference/.../3/.../URLRequestHeader.html ▾
Flex 4.6 and earlier, Flash Pro CS6 and earlier ..... the ACCEPT header in the requestHeaders property of the **URLRequest** class: ... **URLRequest** · **URLLoader** ...

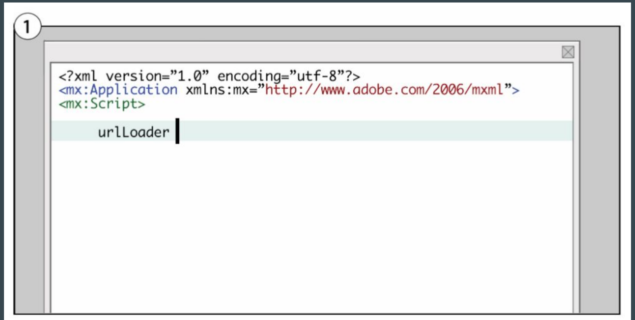
URLRequest - Adobe ActionScript® 3 (AS3) API Reference
help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/.../URLRequest.html ▾
Flex 4.6 and earlier, Flash Pro CS6 and earlier ..... **URLRequest** objects are passed to the load() methods of the Loader, URLStream, and ... web sites via Loader, **URLLoader** will discontinue and not work due to App Transport Security. Please ...

URLLoader - Adobe ActionScript® 3 (AS3)
help.adobe.com/ru_RU/AS3LCR/Flex_4.0/.../URLLoader.html ▾ Translate this page
Flex 4.1 или более ранней версии, Flash Pro CS6 или более ранней версии ..... Класс **URLLoader** загружает данные с URL-адреса в виде текста, ...

URLLoader - how to wait until URL is loaded and... |Adobe ...
https://forums.adobe.com/thread/140138 ▾
Nov 29, 2007 - 10 posts - 3 authors
someURLRequest = new **URLRequest**("some XML file"); ... I would like that first, when someURLLoader.load is called, **flex** wait until external ...

Scenario

Code editor assumes all code is typed by hand

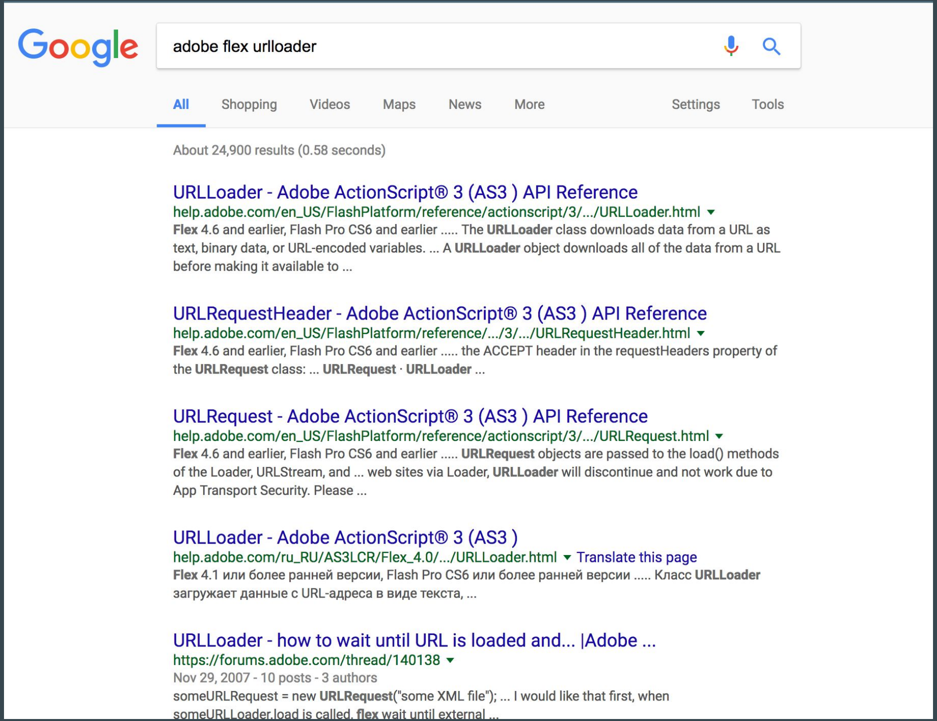


```
<?xml version=”1.0” encoding=”utf-8”?>
<mx:Application xmlns:mx=”http://www.adobe.com/2006/mxml”>
<mx:Script>

    urlLoader
```

Web browser is independent of other tools in the development chain

Search engine has no notion of a user’s development context



adobe flex urlloader

About 24,900 results (0.58 seconds)

URLLoader - Adobe ActionScript® 3 (AS3) API Reference
help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/.../URLLoader.html ▾
Flex 4.6 and earlier, Flash Pro CS6 and earlier ..... The **URLLoader** class downloads data from a URL as text, binary data, or URL-encoded variables. ... A **URLLoader** object downloads all of the data from a URL before making it available to ...

URLRequestHeader - Adobe ActionScript® 3 (AS3) API Reference
help.adobe.com/en_US/FlashPlatform/reference/.../3/.../URLRequestHeader.html ▾
Flex 4.6 and earlier, Flash Pro CS6 and earlier ..... the ACCEPT header in the requestHeaders property of the **URLRequest** class: ... **URLRequest** · **URLLoader** ...

URLRequest - Adobe ActionScript® 3 (AS3) API Reference
help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/.../URLRequest.html ▾
Flex 4.6 and earlier, Flash Pro CS6 and earlier ..... **URLRequest** objects are passed to the load() methods of the Loader, URLStream, and ... web sites via Loader, **URLLoader** will discontinue and not work due to App Transport Security. Please ...

URLLoader - Adobe ActionScript® 3 (AS3)
help.adobe.com/ru_RU/AS3LCR/Flex_4.0/.../URLLoader.html ▾ Translate this page
Flex 4.1 или более ранней версии, Flash Pro CS6 или более ранней версии ..... Класс **URLLoader** загружает данные с URL-адреса в виде текста, ...

URLLoader - how to wait until URL is loaded and... |Adobe ...
https://forums.adobe.com/thread/140138 ▾
Nov 29, 2007 - 10 posts - 3 authors
someURLRequest = new **URLRequest**("some XML file"); ... I would like that first, when someURLLoader.load is called, **flex** wait until external ...

Blueprint

- A Web search interface for accessing online example code from within the development environment and thus enable programmers to write better code more easily.

Adobe Flex Builder

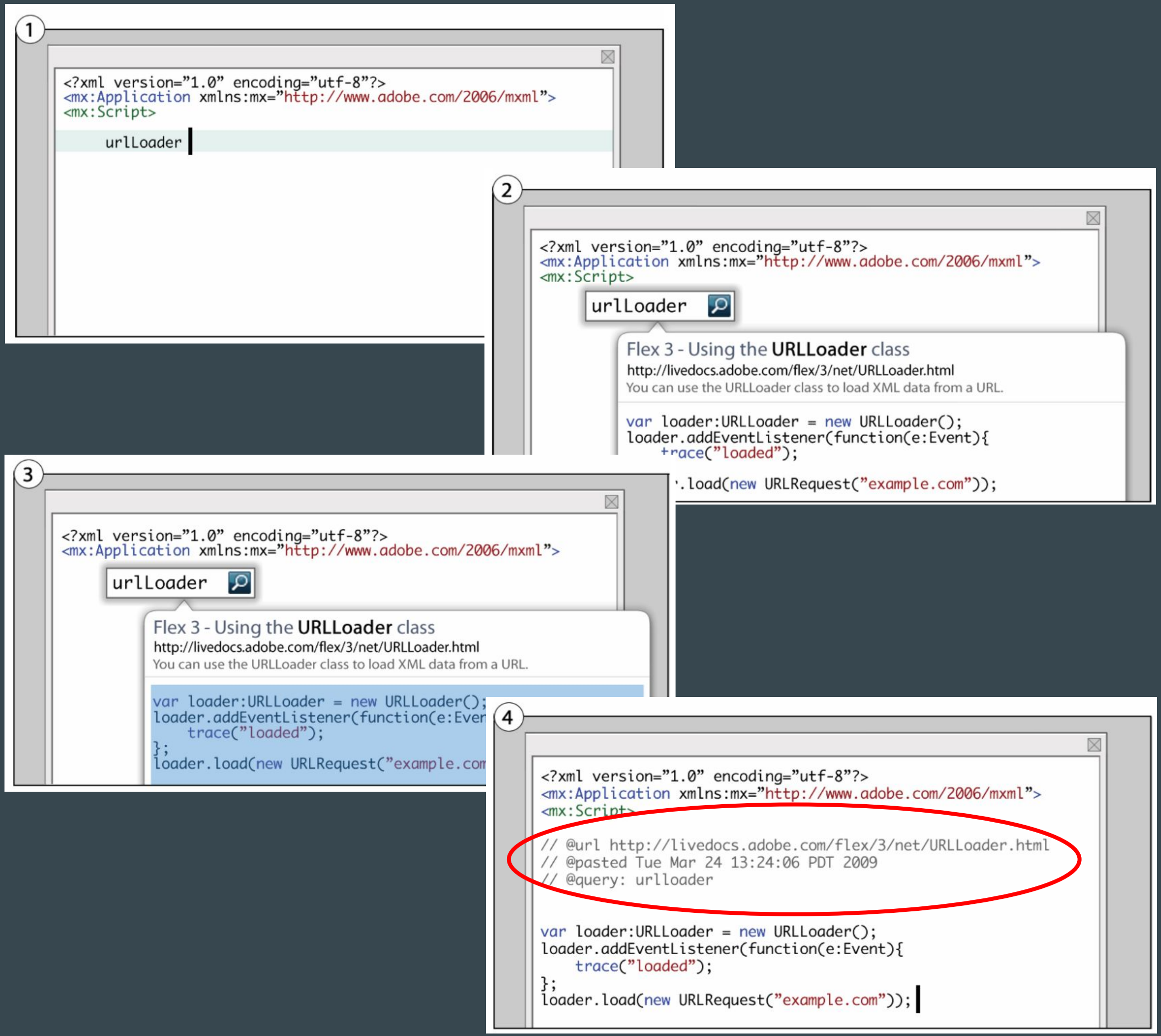
- Provides a UI for search queries and results.
- Automatically augments queries with code context.
- Focused on getting code examples.

Scenario

Suppose you need to retrieve power-usage data from a Web service, using Adobe Flex builder.

Method contains “URL” ? => urlLoader => Can’t remember the usage

Benefits of this approach?



1

```
<?xml version=”1.0” encoding=”utf-8”?>
<mx:Application xmlns:mx=”http://www.adobe.com/2006/mxml”>
<mx:Script>
        urlLoader
```

2

```
<?xml version=”1.0” encoding=”utf-8”?>
<mx:Application xmlns:mx=”http://www.adobe.com/2006/mxml”>
<mx:Script>
        urlLoader
```

Flex 3 - Using the **URLLoader** class
http://livedocs.adobe.com/flex/3/net/URLLoader.html
You can use the URLLoader class to load XML data from a URL.

```
var loader:URLLoader = new URLLoader();
loader.addEventListener(function(e:Event){
    trace(”loaded”);
'.load(new URLRequest(”example.com”));
```

3

```
<?xml version=”1.0” encoding=”utf-8”?>
<mx:Application xmlns:mx=”http://www.adobe.com/2006/mxml”>
        urlLoader
```

Flex 3 - Using the **URLLoader** class
http://livedocs.adobe.com/flex/3/net/URLLoader.html
You can use the URLLoader class to load XML data from a URL.

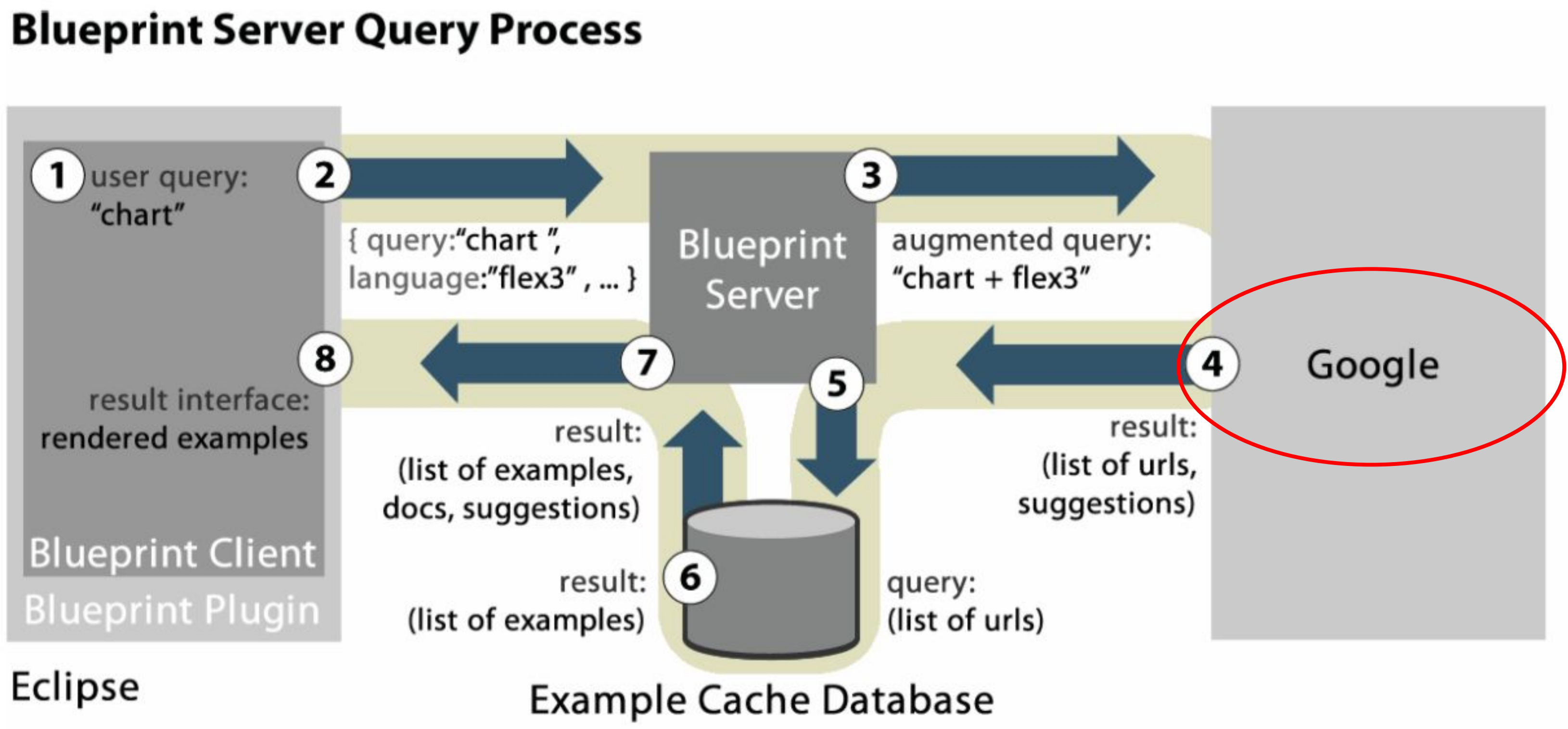
```
var loader:URLLoader = new URLLoader();
loader.addEventListener(function(e:Ever
    trace(”loaded”);
};
loader.load(new URLRequest(”example.com
```

4

```
<?xml version=”1.0” encoding=”utf-8”?>
<mx:Application xmlns:mx=”http://www.adobe.com/2006/mxml”>
<mx:Script>
// @url http://livedocs.adobe.com/flex/3/net/URLLoader.html
// @pasted Tue Mar 24 13:24:06 PDT 2009
// @query: urlloader

var loader:URLLoader = new URLLoader();
loader.addEventListener(function(e:Event){
    trace(”loaded”);
};
loader.load(new URLRequest(”example.com”));
```

Implementation


Blueprint Server Query Process
1 user query: "chart"
2
{ query:"chart ", language:"flex3" , ... }
Blueprint Server
3
augmented query: "chart + flex3"
4
Google
result: (list of urls, suggestions)
5
query: (list of urls)
6
result: (list of examples)
Example Cache Database
7
result: (list of examples, docs, suggestions)
8
result interface: rendered examples
Blueprint Client
Blueprint Plugin
Eclipse

Evaluation

- Lab study:
  - 20 participants
  - Control: Firefox + Adobe Community Help Search Engine
  - Treatment: Blueprint
  - Measures: Time, Rank
    - Tutorial task
    - Directed programming task
    - Exploratory programming task

- Hypotheses:
  1. Programmers will complete direct tasks more quickly because they will find example code faster
  2. Code produced will have the same or higher quality as code written with traditional means
  3. Programmers will produce better designs on an exploratory design task

small-scale

Evaluation

- Longitudinal study:
  - 2,024 Blueprint users + 13,283 Community Help users
  - Control: Users who used the Community Help search engine over same duration
  - Treatment: Blueprint users
  - Measures: Click-throughs, syntax, queries etc.

- Hypotheses:

1. If additional context is not necessary, Blueprint queries should have a significantly lower click-through rate
2. If users are using Blueprint with other IDE features, queries should contain more correctly formatted code
3. If Blueprint is used for reminders, users should repeat queries more frequently across sessions

large-scale

Lab Study - Results

1. **Programmers will complete direct tasks more quickly because they will find example code faster**

- Time to first copy/paste of example (57s vs. 121s)
- Time to completion (346s vs. 479s)
- Paste time strongly correlated with task completion time

✓

Lab Study - Results

2. **Code produced will have the same or higher quality as code written with traditional means**

- Professional software engineer external to the project rank-ordered participants code
- Treatment produced significantly higher- rated code

✓

Lab Study - Results

3. **Programmers will produce better designs on an exploratory design task**

- Professional software engineer external to the project rank-ordered participants charts
- Treatment produced higher-rated designs but the result was not statistically significant

❌

Longitudinal Study - Results

1. **If additional context is not necessary, Blueprint queries should have a significantly lower click-through rate**

- Number of click-throughs to source pages on search (~43,000 queries measured)
- Treatment is much lower (0.38 vs. 1.32)

✓

Longitudinal Study - Results

2. **If users are using Blueprint with other IDE features, queries should contain more correctly formatted code**

- Check for camelCase in queries
- Treatment contains much more camelCase (49.6% vs. 16.2% of queries)

✓

Longitudinal Study - Results

3. **If Blueprint is used for reminders, users should repeat queries more frequently across sessions**

- Repeat queries during sessions
- Treatment has much more queries issued again by the same user (12.2% vs. 7.8% of queries)

✓

Small scale vs. Large scale

Other example:

- Joel Brandt et al. “Two studies of opportunistic programming: interleaving web foraging, learning, and writing code”
- Lab: 20 programmers to rapidly prototype a Web application in the lab.
- Longitudinal: quantitatively analyzed a month-long sample of Web query data. 24,293 programmers produced the 101,289 queries in the sample.

Discussion - Groups of 2 (1 min)

- What are the pros and cons of small-scale and large-scale studies?
- How would you apply them into your own projects?

Overarching discussion

How would you relate Blueprint to Ability-Based Design?

Thank you!