

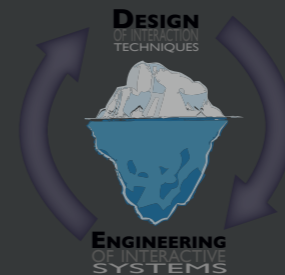
DESIGNEERING INTERACTION

A MISSING LINK IN THE EVOLUTION
OF HUMAN-COMPUTER INTERACTION

STÉPHANE HUOT

Université Paris-Sud
Laboratoire de Recherche en Informatique
Inria - in|situ| group

Habilitation à Diriger des Recherches
May 7th, 2013

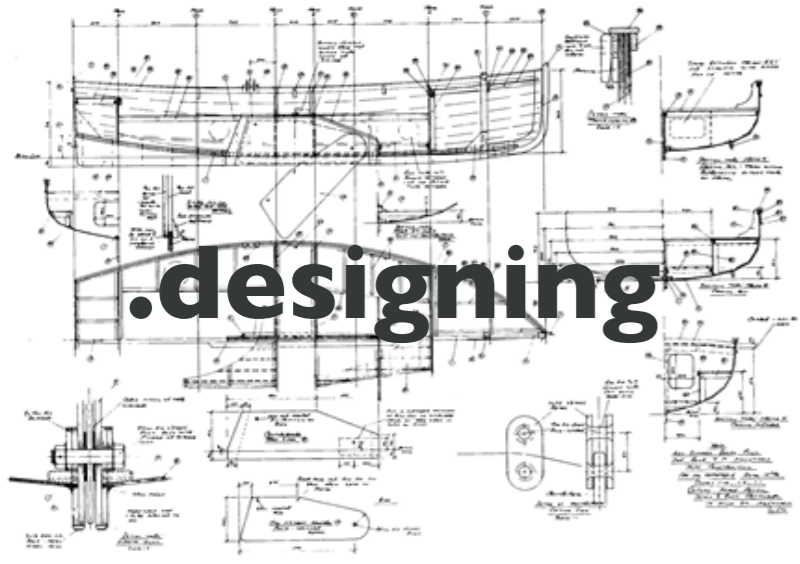


research only real when shared

Caroline Appert
Michel Beaudouin-Lafon
Olivier Chapuis
Fanny Chevalier
Jonathan Diehl
Pierre Dragicevic
Cédric Dumas
James R. Eagan
Guillaume Faure
Jean-Daniel Fekete
Émilien Ghomi
Sean Gustafson
Clemens N. Klokmoose
Eric Lecolinet
Can Liu
Wendy E. Mackay

Mathieu Nancel
Emmanuel Pietriga
Clément Pillias
Romain Primet
Anne Roudaut
Quentin Roy
Jean-Marc Vézien
Julie Wagner

Alexandra Merlin

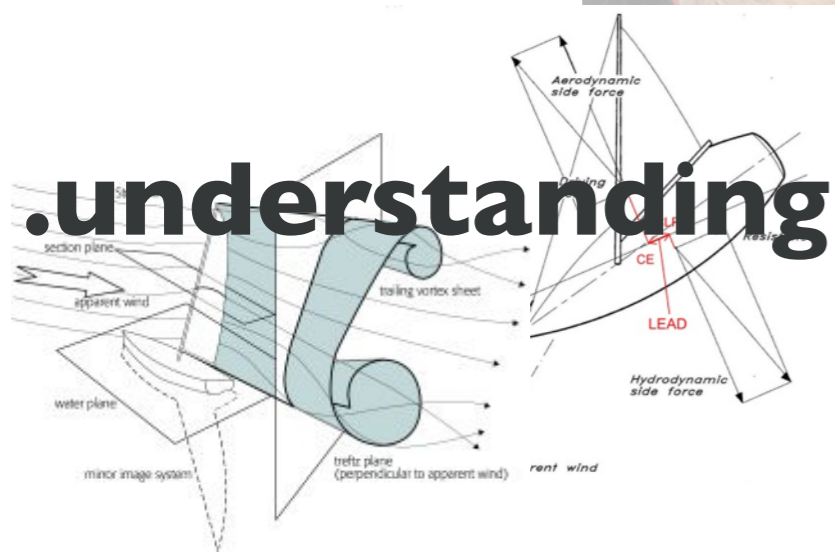


.designing



.crafting

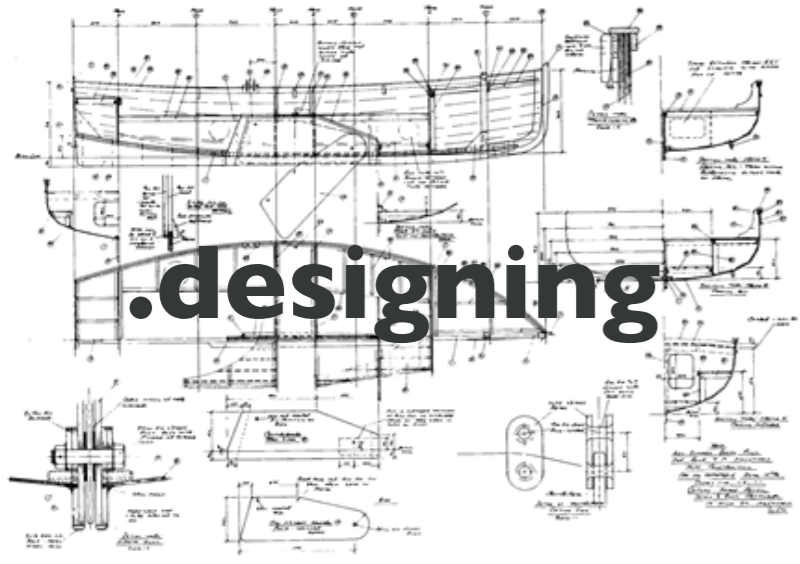
**design
theories
technology
craftsmanship**



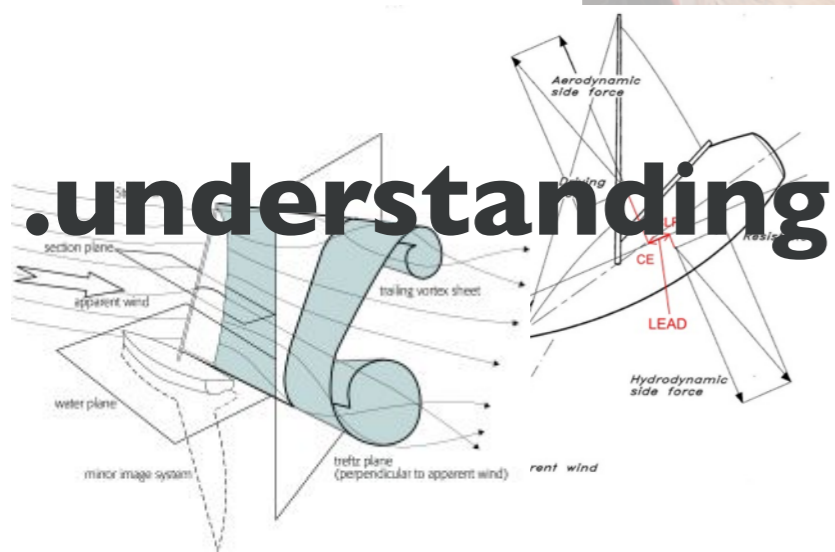
.understanding



.using



**design
theories
technology
craftsmanship**



the golden age of a young research field



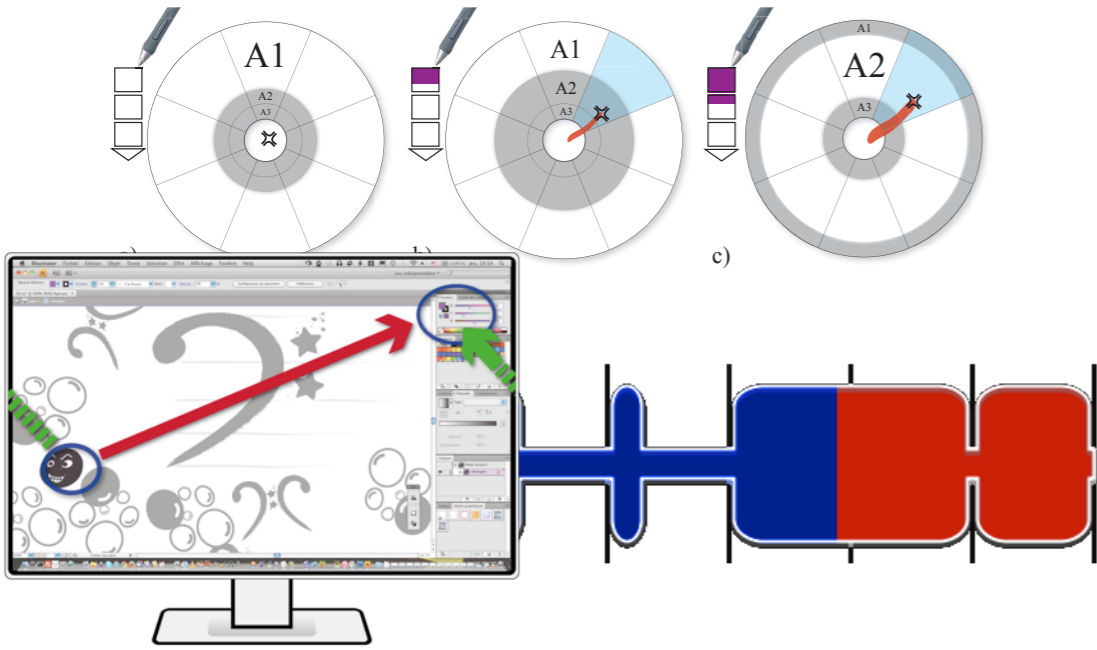
Stuart Card - keynote @ CHI '12

“how to ground the field, accelerate its progress, and make it cumulative by fashioning theories and incorporating them into practice”

↳ Technology develops by combinatoric evolution and we need to understand how HCI works with technology, to focus on technological and theoretical progress all together in order to successfully take on the new “golden age” that HCI is entering now.

interaction design and software technology

designing interaction techniques

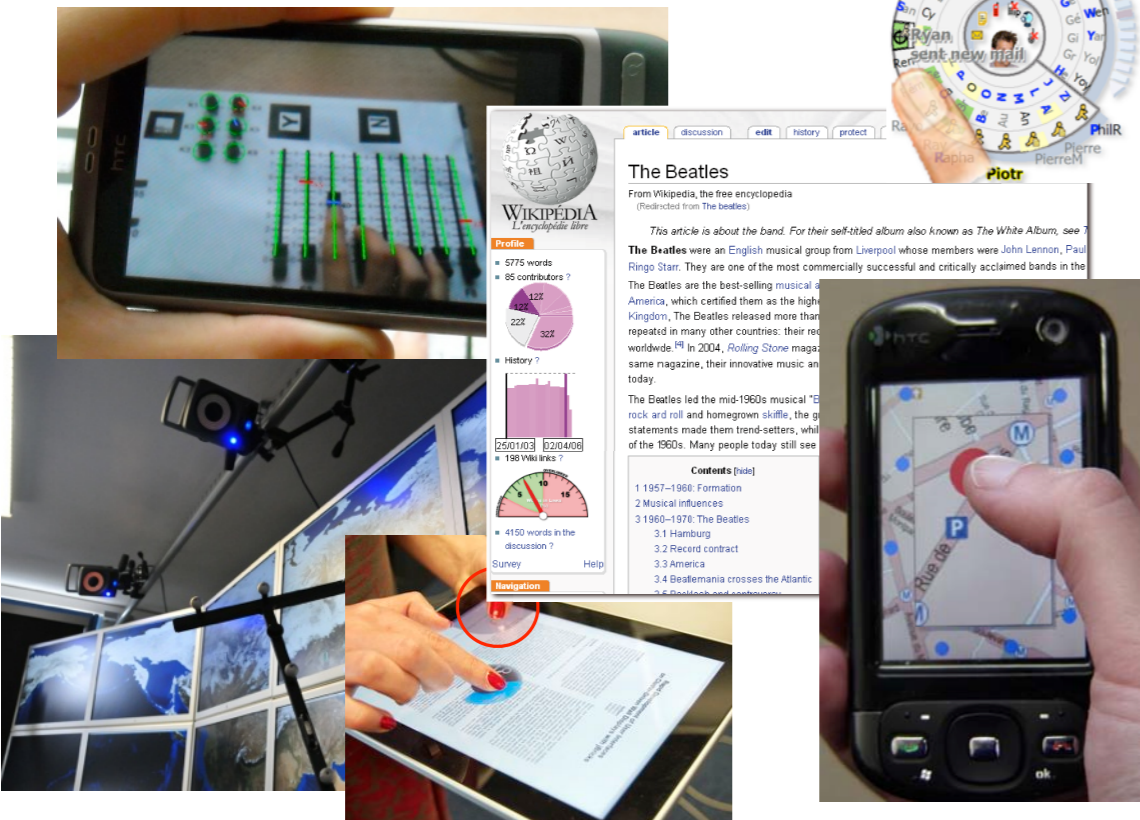


Vice Versa Vice Versa and Vice Versa

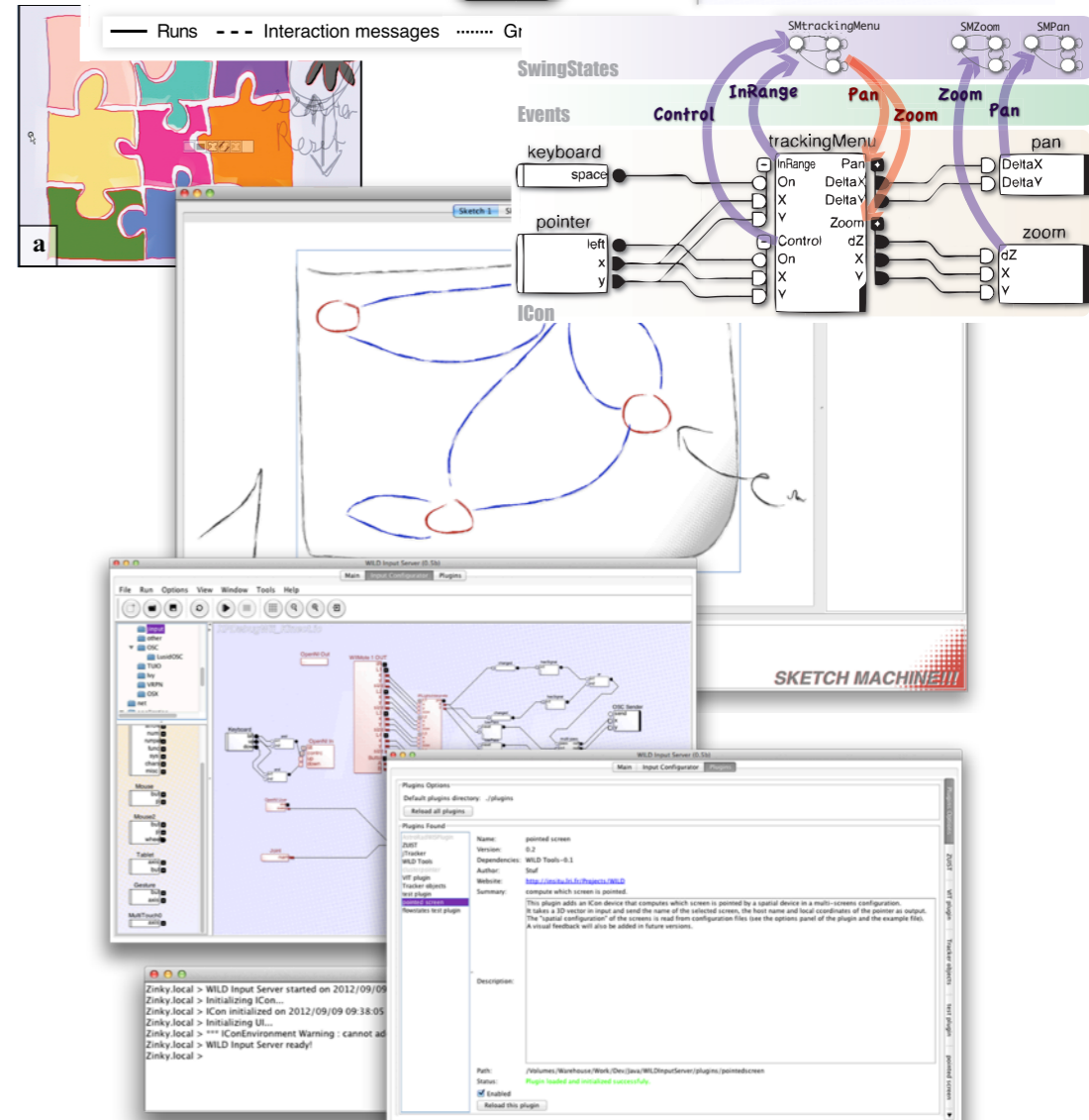
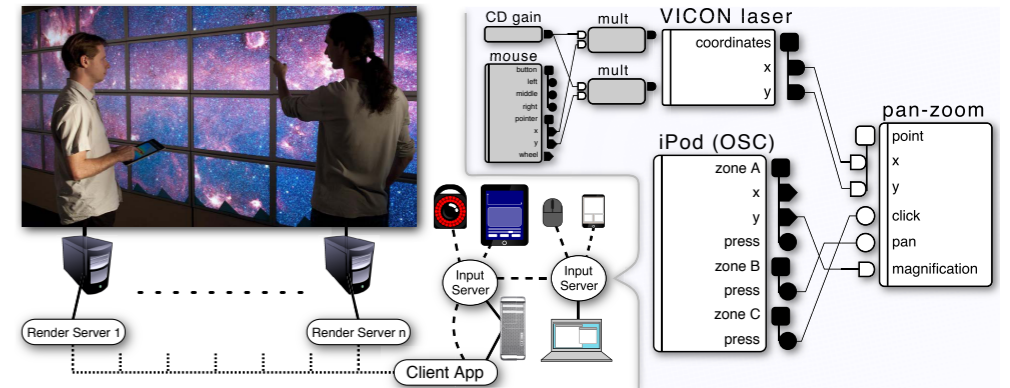
Markup Code to Rendered Documents and Vice Versa

Animating from Markup Code to Rendered Documents and Vice Versa

We present a quick preview technique that smoothly transitions between document markup code and its visual rendering. This technique allows use the code they are ed the text editor.



engineering interactive systems



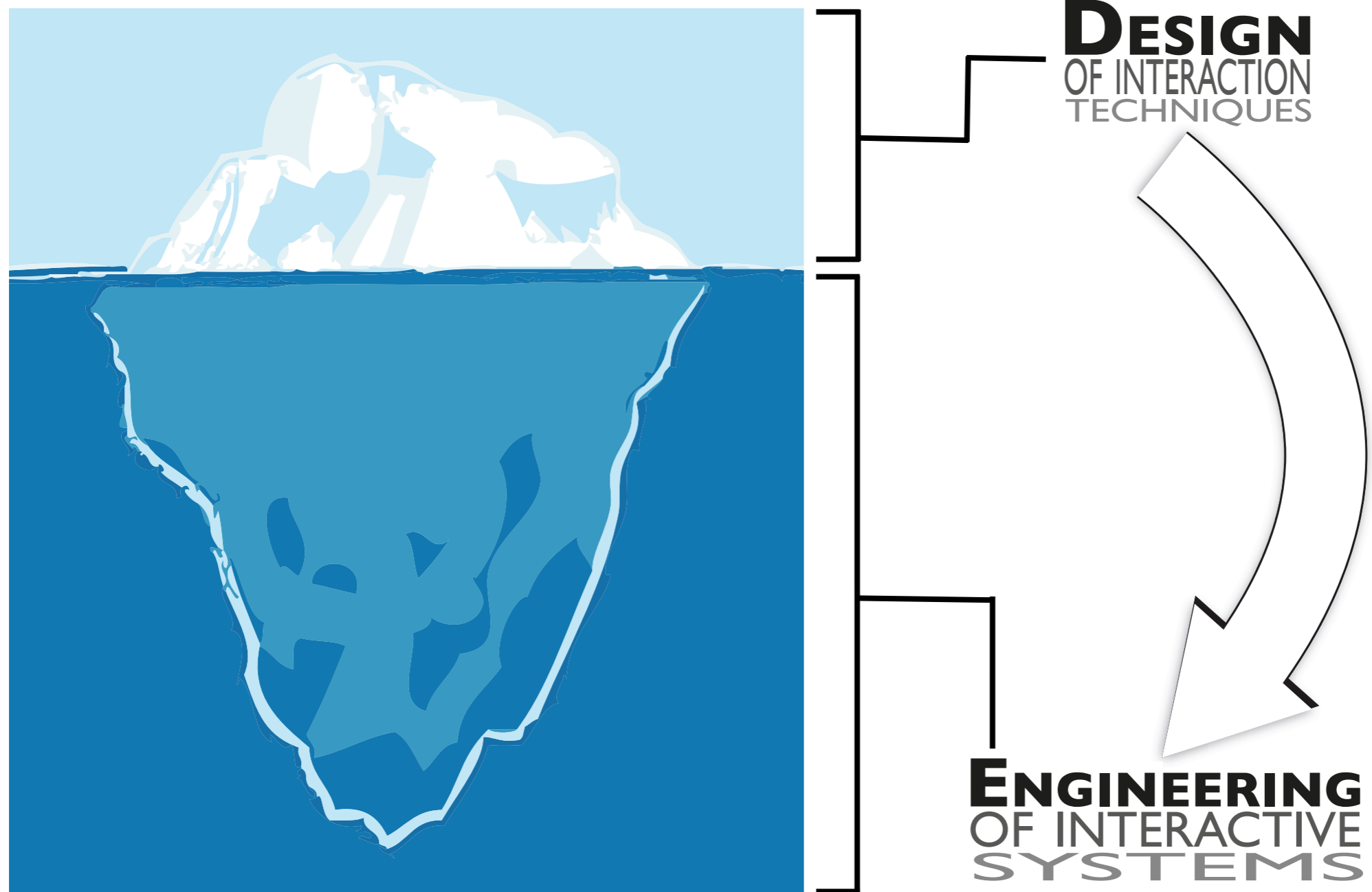
interaction design challenges technology

DESIGN
OF INTERACTION
TECHNIQUES



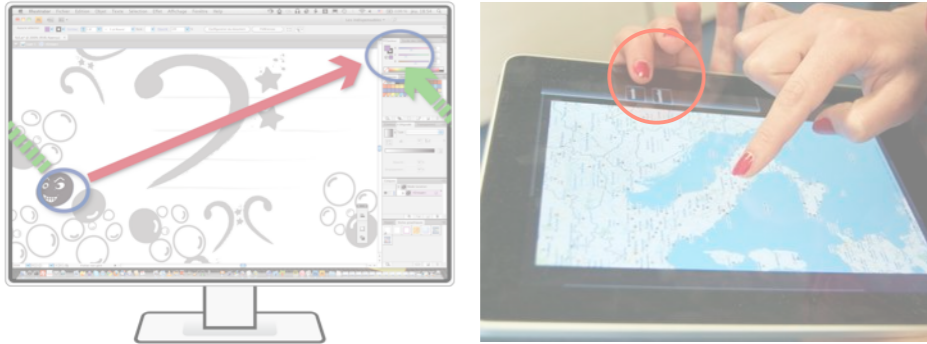
ENGINEERING
OF INTERACTIVE
SYSTEMS

interaction design challenges technology

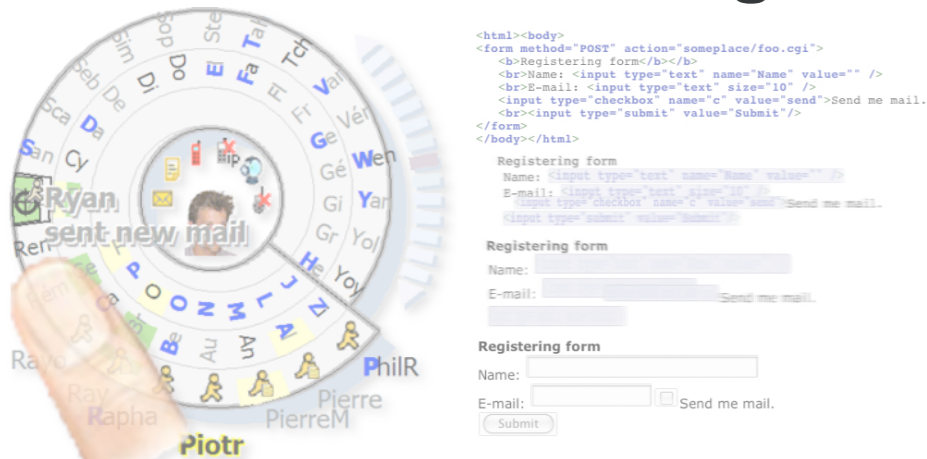


interaction design challenges technology

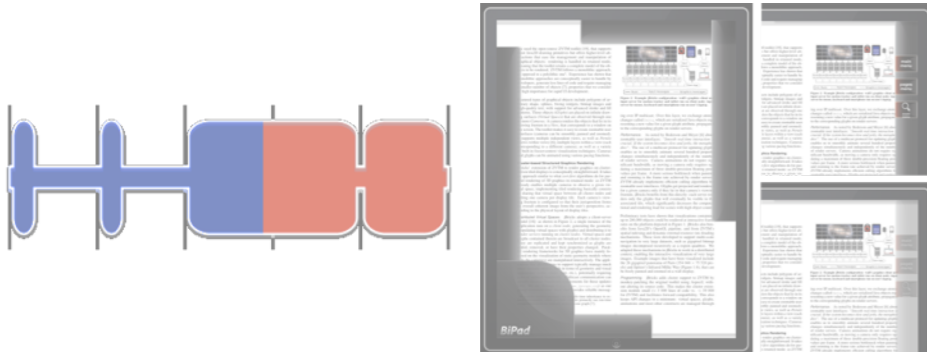
.when interaction design is driven by technology



.when interaction design is constrained by technology



.when interaction design improves technology



DESIGN
OF INTERACTION
TECHNIQUES

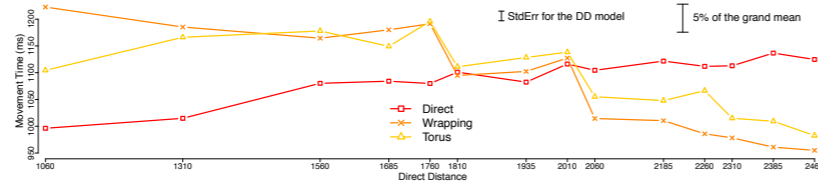
ENGINEERING
OF INTERACTIVE
SYSTEMS

when interaction design is driven by technology

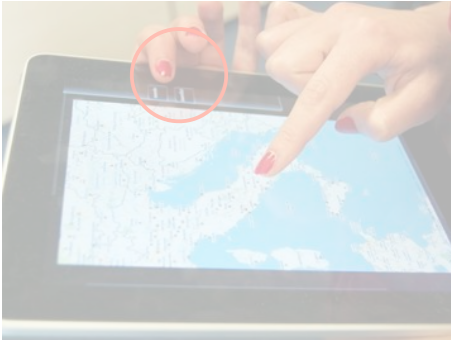
.designing with limitations



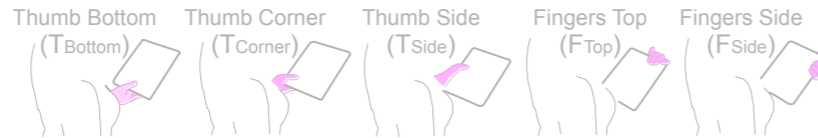
[TorusDesktop - CHI'11]



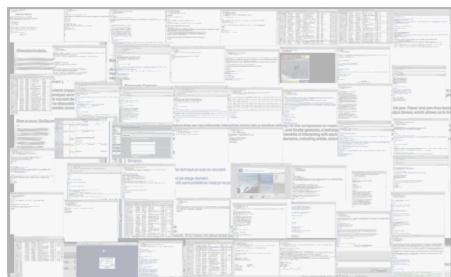
.designing for features



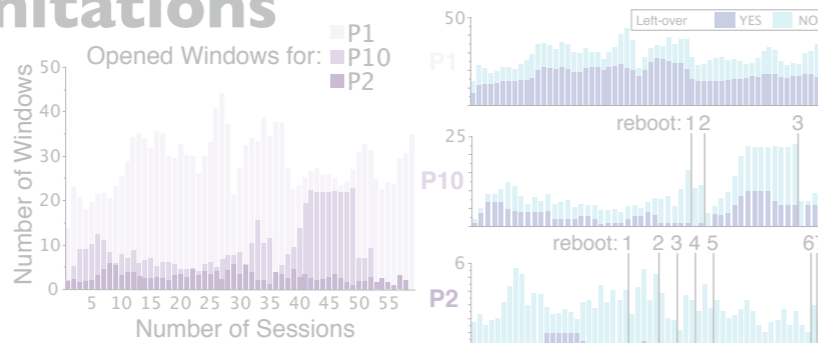
[BiPad - CHI'12]



.identifying limitations



[left-over windows - IHM'12]



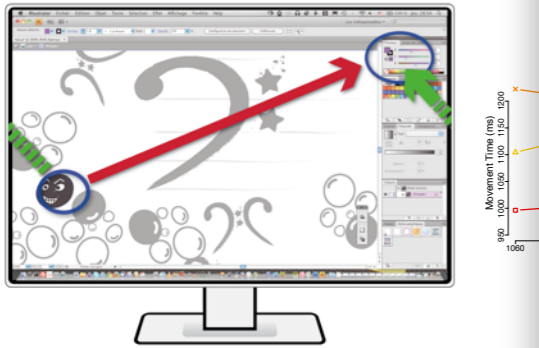
DESIGN
OF INTERACTION
TECHNIQUES

ENGINEERING
OF INTERACTIVE
SYSTEMS

TorusDesktop

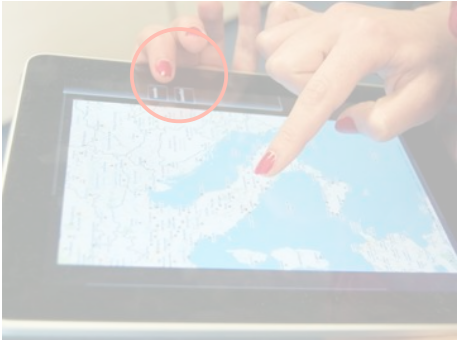
with O. Chapuis & P. Dragicevic

.designing with I



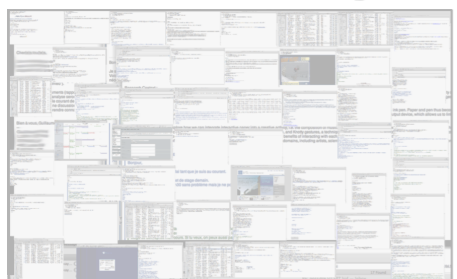
[TorusDesktop - CHI'11]

.designing for fe



[BiPad - CHI'12]

.identifying limit



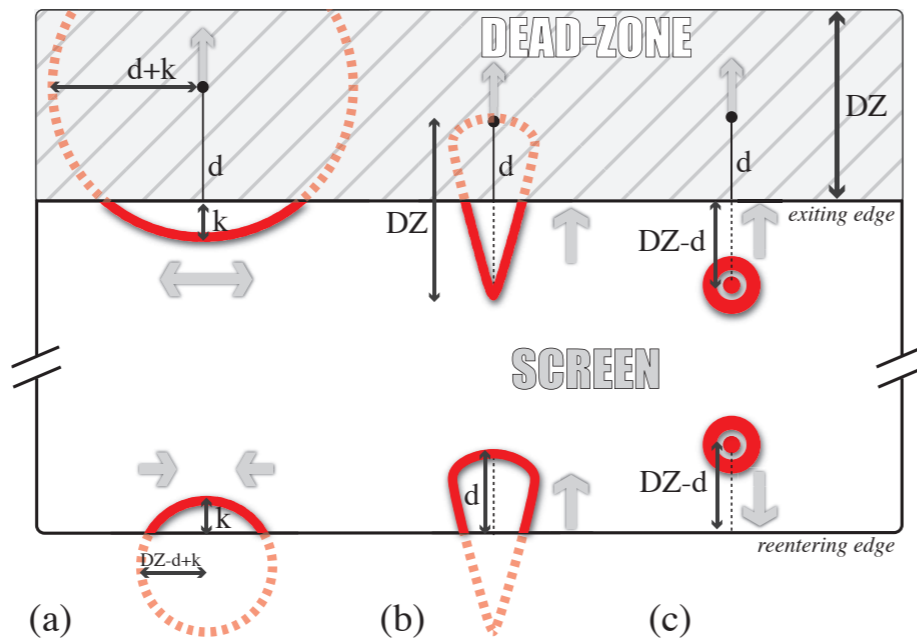
[left-over windows - IHM'11]

.designing with limitation

- ↳ accessing targets in real systems
- detecting user's intentions

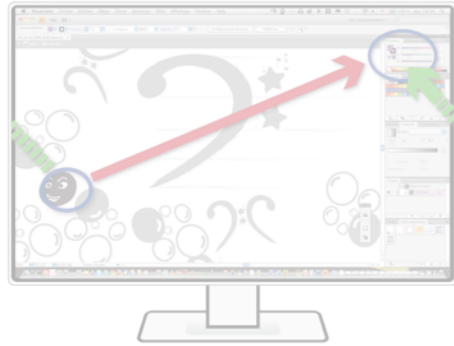
objective : target agnostic pointing technique

solution : revisit cursor wrapping

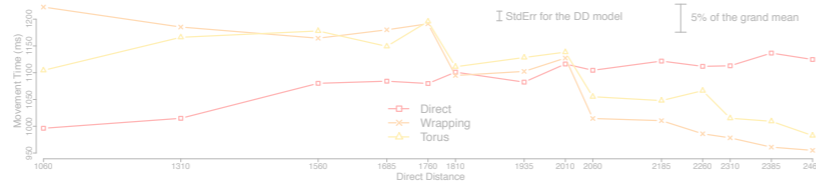


when interaction design is driven by technology

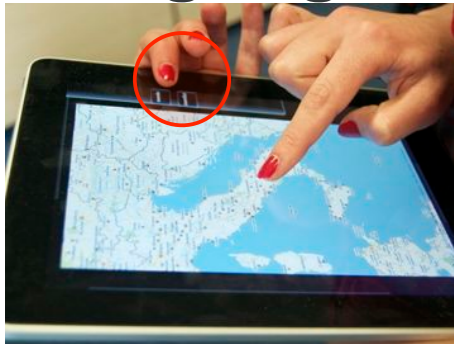
.designing with limitations



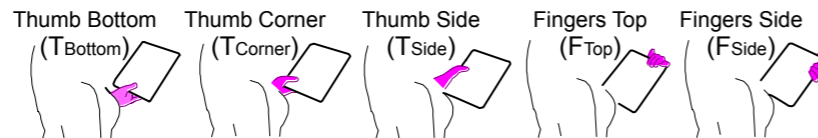
[TorusDesktop - CHI'11]



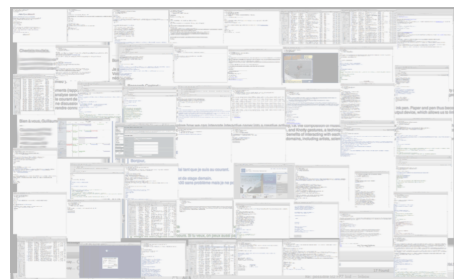
.designing for features



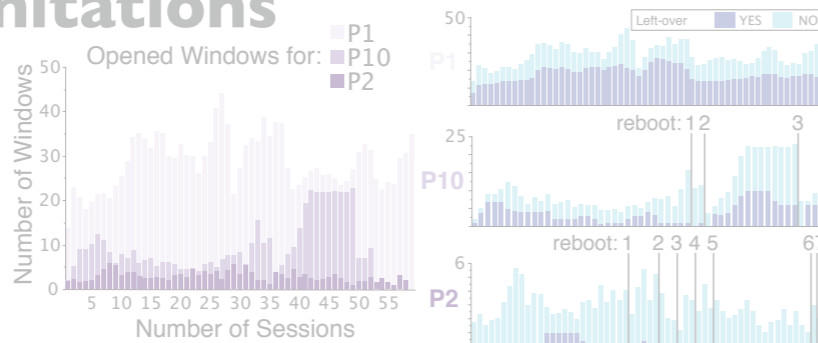
[BiPad - CHI'12]



.identifying limitations



[left-over windows - IHM'12]



DESIGN
OF INTERACTION
TECHNIQUES

ENGINEERING
OF INTERACTIVE
SYSTEMS

when interaction design is driven by technology

BiPad

with J. Wagner & W. E. Mackay

.designing for features

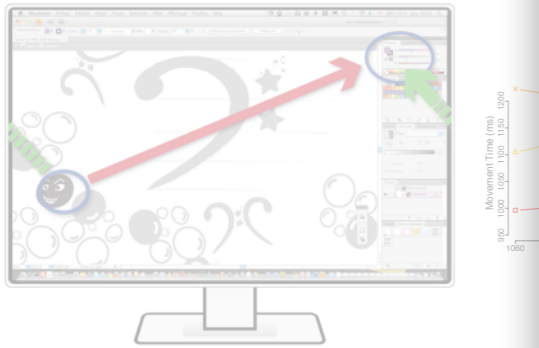
↳ handheld devices

multitouch capabilities

objective : bimanual interaction in mobility

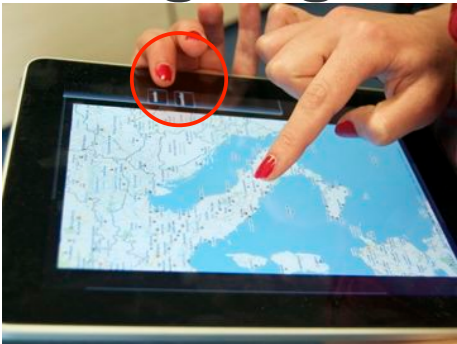
solution : design space and multiple designs

.designing with I



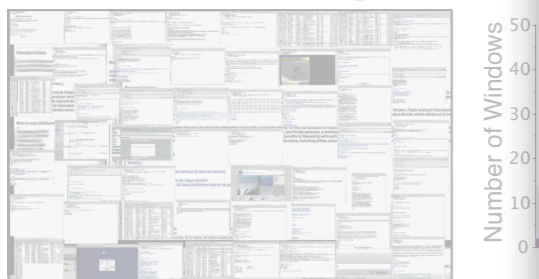
[TorusDesktop - CHI'11]

.designing for fe

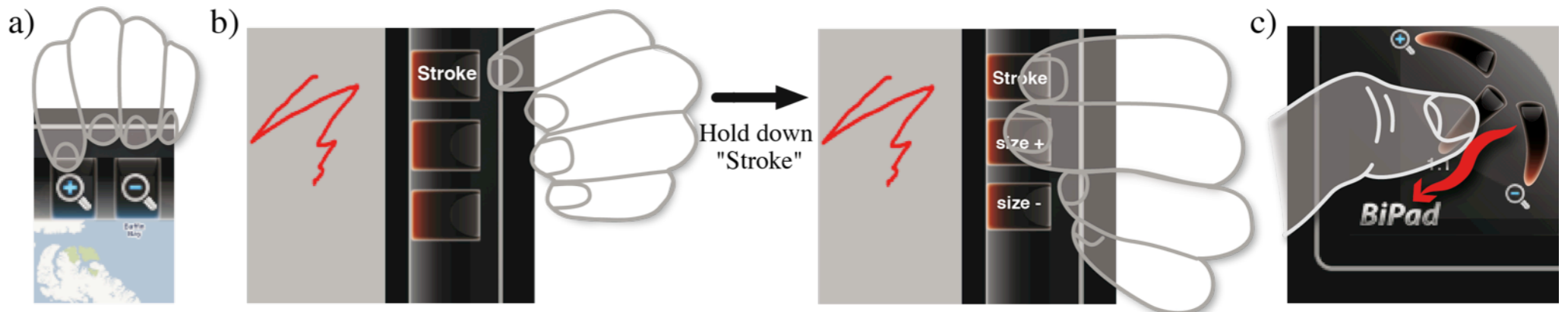
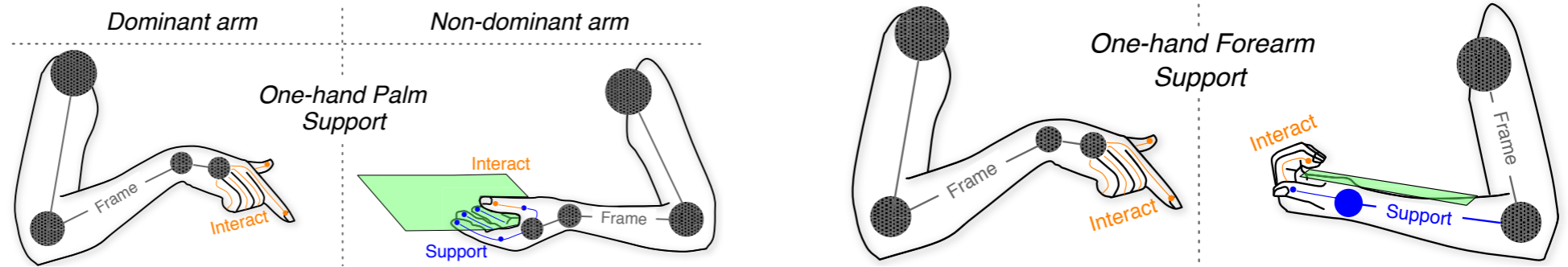


[BiPad - CHI'12]

.identifying limit

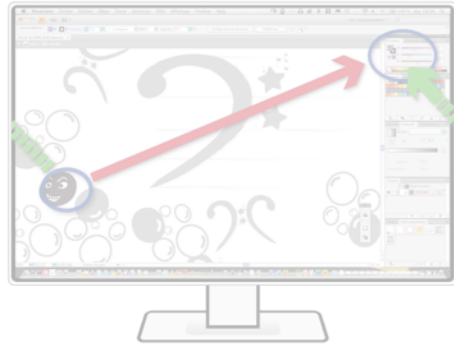


[left-over windows - IHM'11]

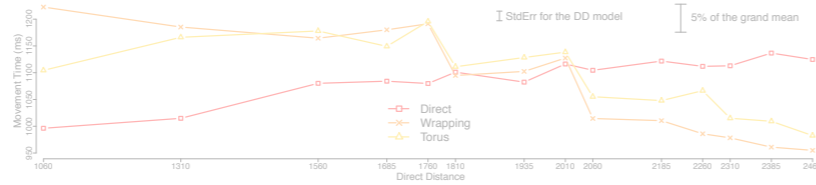


when interaction design is driven by technology

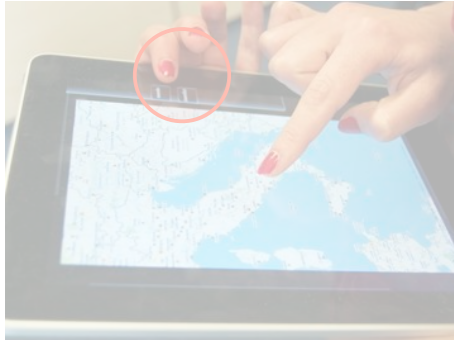
.designing with limitations



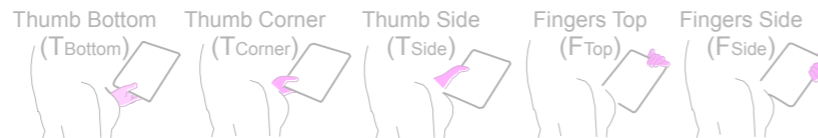
[TorusDesktop - CHI'11]



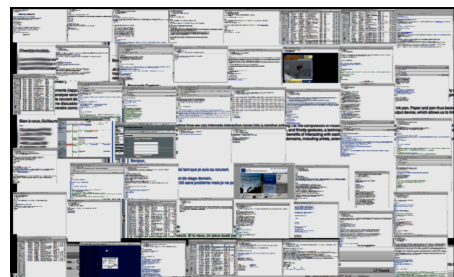
.designing for features



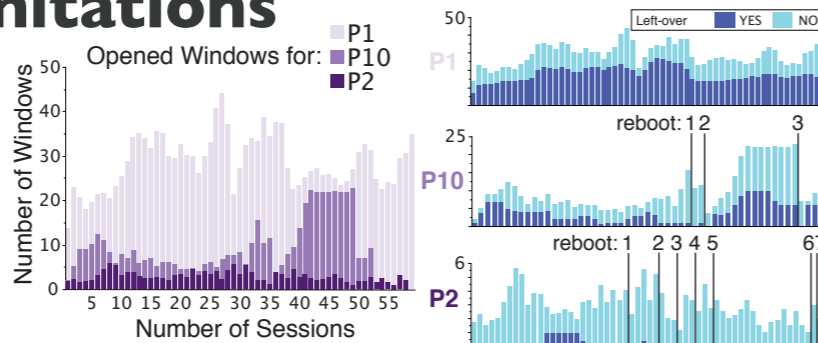
[BiPad - CHI'12]



.identifying limitations



[left-over windows - IHM'12]

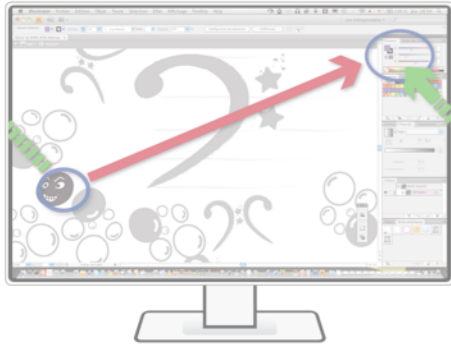


DESIGN
OF INTERACTION
TECHNIQUES

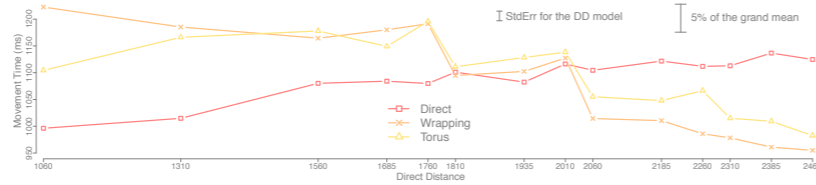
ENGINEERING
OF INTERACTIVE
SYSTEMS

when interaction design is driven by technology

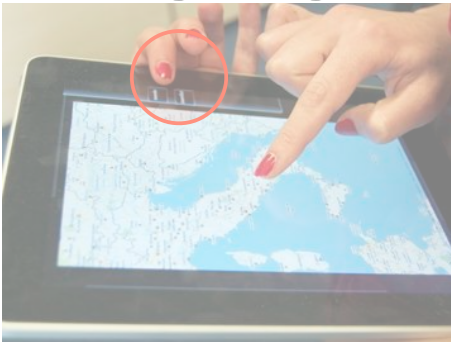
.designing with limitations



[TorusDesktop - CHI'11]



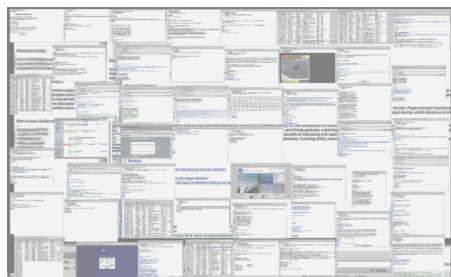
.designing for features



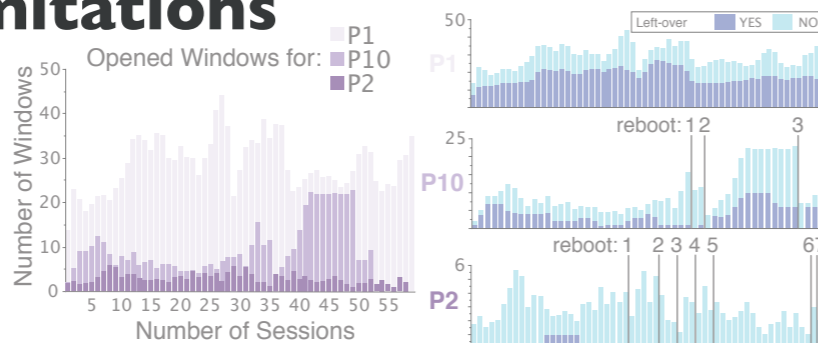
[BiPad - CHI'12]



.identifying limitations



[left-over windows - IHM'12]



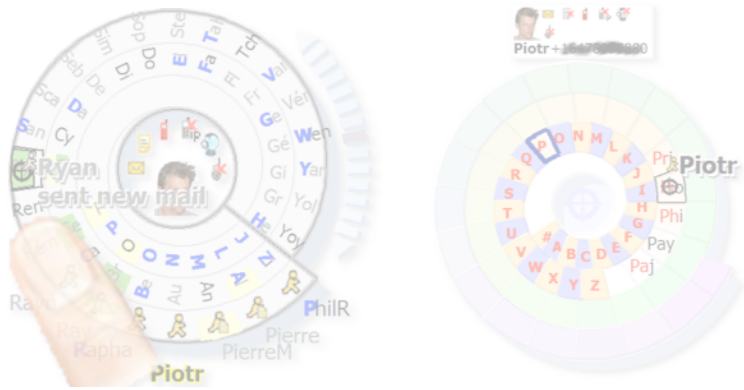
DESIGN
OF INTERACTION
TECHNIQUES

Evaluation

ENGINEERING
OF INTERACTIVE
SYSTEMS

when interaction design is constrained by technology

.performance issues



[SpiraList & SnailList - NordiCHI'06 & Interact'07]

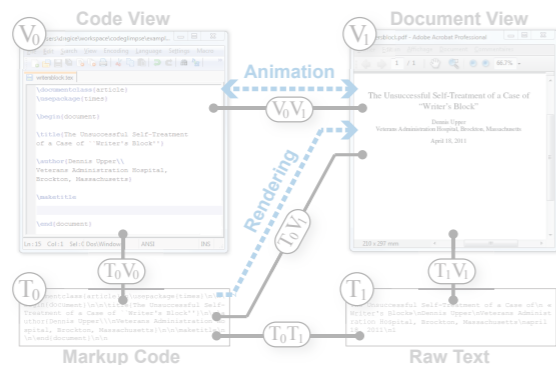
.openness and interoperability issues

```
<html><body>
<form method="POST" action="someplace/foo.cgi">
  <b>Registering form</b></b>
  <br><input type="text" name="Name" value="" />
  <br><input type="text" size="10" />
  <br><input type="checkbox" name="c" value="send">Send me mail.
  <br><input type="submit" value="Submit"/>
</form>
</body></html>

Registering form
Name: 
E-mail:   Send me mail.


Registering form
Name: 
E-mail:   Send me mail.

```



[Glimpse - UIST'11]

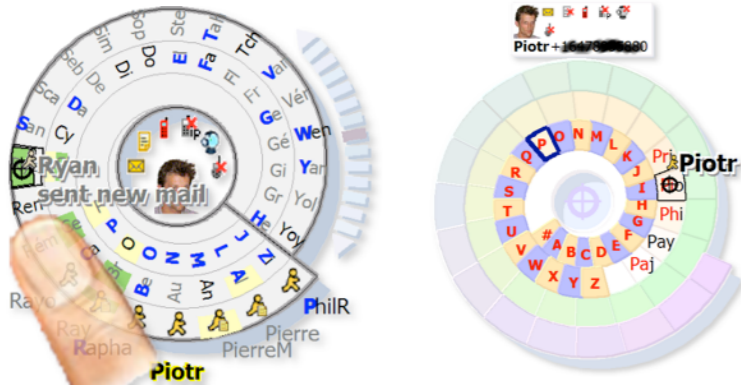
**DESIGN
OF INTERACTION
TECHNIQUES**

Evaluation

**ENGINEERING
OF INTERACTIVE
SYSTEMS**

when interaction design is constrained by technology

.performance issues



[SpiraList & SnailList - NordiCHI'06 & Interact'07]

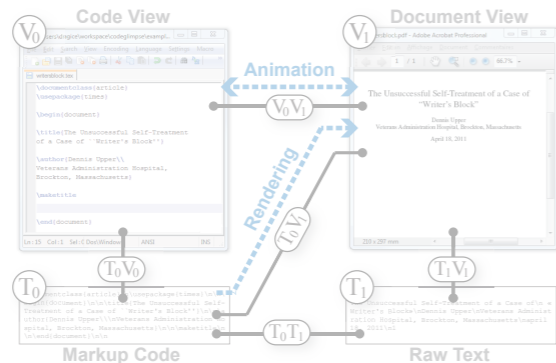
.openness and interoperability issues

```
<html><body>
<form method="POST" action="someplace/foo.cgi">
  <b>Registering form</b></b>
  <br>Name: <input type="text" name="Name" value="" />
  <br>E-mail: <input type="text" size="10" />
  <br><input type="checkbox" name="c" value="send">Send me mail.
  <br><input type="submit" value="Submit"/>
</form>
</body></html>

Registering form
Name: _____
E-mail: _____ Send me mail.


Registering form
Name: _____
E-mail: _____  Send me mail.

```



[Glimpse - UIST'11]

DESIGN
OF INTERACTION
TECHNIQUES

Evaluation

ENGINEERING
OF INTERACTIVE
SYSTEMS

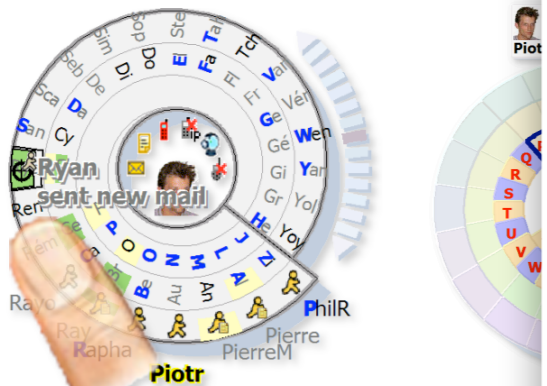
when interaction design is constrained by technology

SpiraList & SnailList

with E. Lecolinet

- objective** : manipulating large lists on mobile devices
- solution** : new layout and F+C data presentation

.performance iss



[SpiraList & SnailList - No

.openness and in

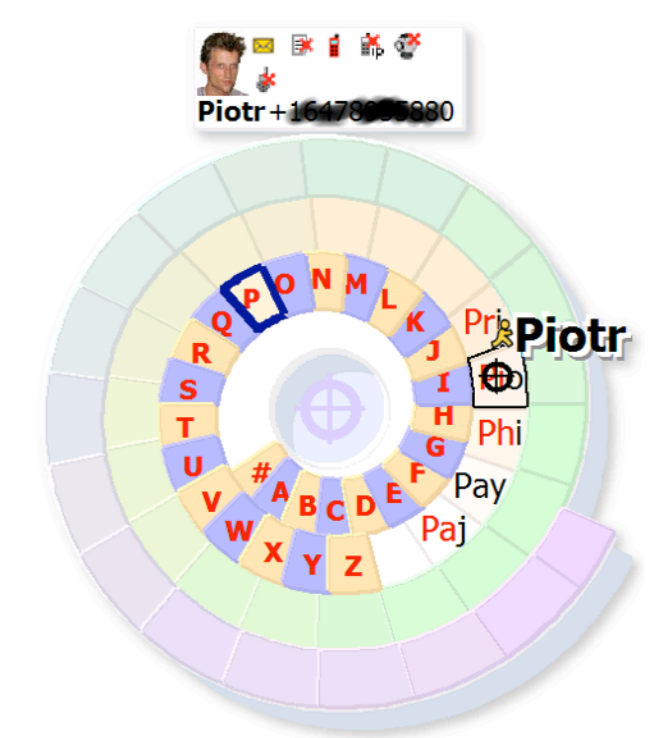
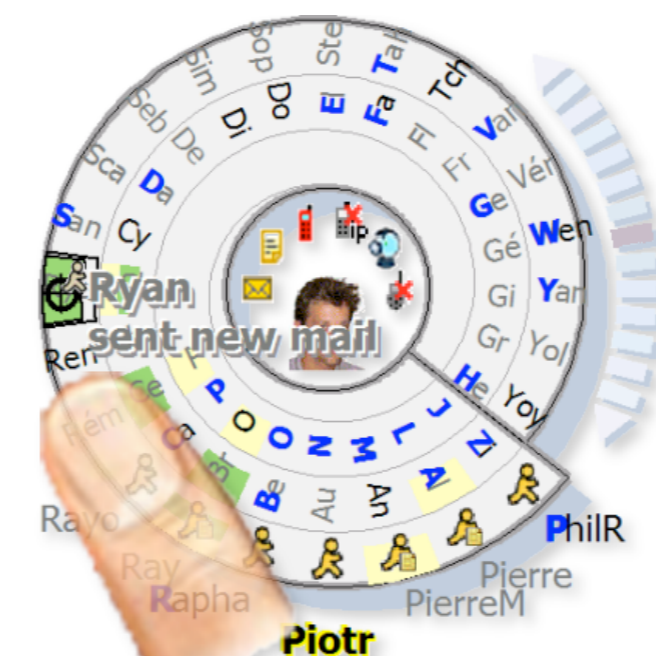
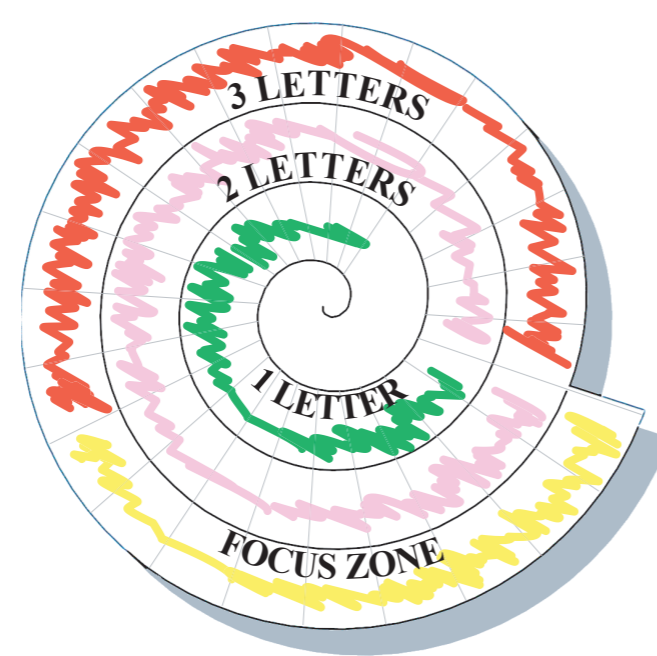
```
<html><body>
<form method="POST" action="someplace/foo.cgi">
  <b>Registering form</b></b>
  <br><input type="text" name="Name" value="" />
  <br><input type="text" name="E-mail" value="" />
  <br><input type="checkbox" name="c" value="send">Send me mail.
  <br><input type="submit" value="Submit"/>
</form>
</body></html>

Registering form
Name: 
E-mail: 
 Send me mail.


Registering form
Name: 
E-mail: 
 Send me mail.

```

[Glimpse - UIST'11]

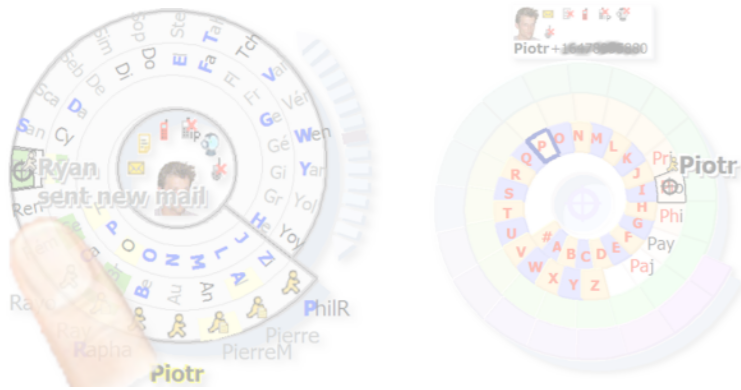


.performance issues

- ↳ input hardware
- advanced graphics vs limited hardware/libraries

when interaction design is constrained by technology

.performance issues



[SpiraList & SnailList - NordiCHI'06 & Interact'07]

.openness and interoperability issues

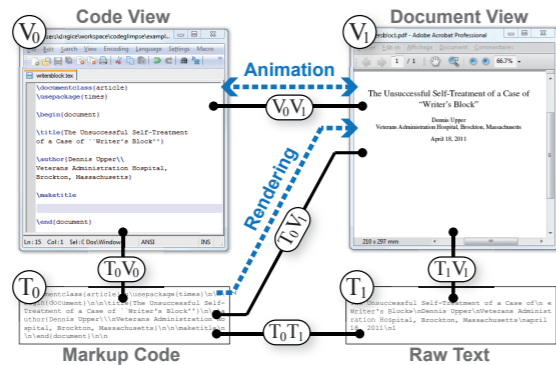
```
<html><body>
<form method="POST" action="someplace/foo.cgi">
  <b>Registering form</b></b>
  <br>Name: <input type="text" name="Name" value="" />
  <br>E-mail: <input type="text" size="10" />
  <br><input type="checkbox" name="c" value="send">Send me mail.
  <br><input type="submit" value="Submit"/>
</form>
</body></html>

Registering form
Name: 
E-mail: 
 Send me mail.


Registering form
Name: 
E-mail: 
 Send me mail.

```

[Glimpse - UIST'11]



DESIGN
OF INTERACTION
TECHNIQUES

Evaluation

ENGINEERING
OF INTERACTIVE
SYSTEMS

when interaction design is constrained by technology

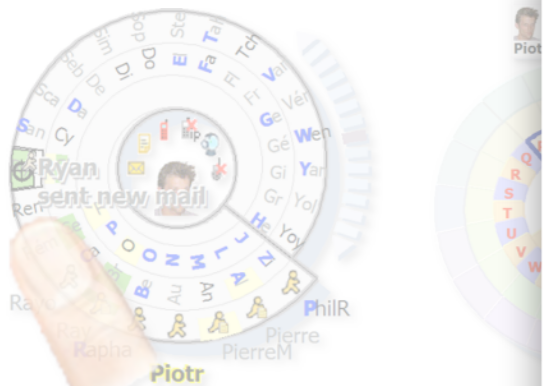
Gliimpse

with P. Dragicevic & F. Chevalier

objective : improve markup language editing

solution : in-place & on-demand animations

.performance iss



[SpiraList & SnailList - No

.openness and in

```
<html><body>
<form method="POST" action="someplace/foo.cgi">
  <b>Registering form</b></b>
  <br>Name: <input type="text" name="Name" value="" />
  <br>E-mail: <input type="text" size="10" />
  <br><input type="checkbox" name="c" value="send">Send me mail.
  <br><input type="submit" value="Submit"/>
</form>
</body></html>
```

```
Registering form
Name: <input type="text" name="Name" value="" />
E-mail: <input type="text" size="10" />
<input type="checkbox" name="c" value="send">Send me mail.
<input type="submit" value="Submit"/>
```

```
Registering form
Name: _____
E-mail: _____ Send me mail.
```

```
Registering form
Name: _____
E-mail: _____  Send me mail.
Submit
```

[Gliimpse - UIST'11]

.openness and interoperability issues

↳ components inspection

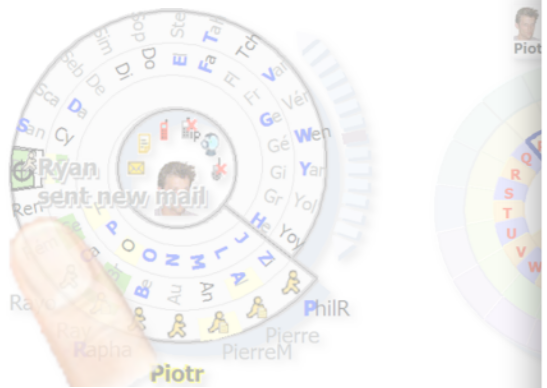
linking components

when interaction design is constrained by technology

Glimpse

with P. Dragicevic & F. Chevalier

.performance iss



[SpiraList & SnailList - No

.openness and in

```
<html><body>
<form method="POST" action="someplace/foo.cgi">
  <b>Registering form</b></b>
  <br>Name: <input type="text" name="Name" value="" />
  <br>E-mail: <input type="text" size="10" />
  <br><input type="checkbox" name="c" value="send">Send me mail.
  <br><input type="submit" value="Submit"/>
</form>
</body></html>

Registering form
Name: 
E-mail: 
 Send me mail.


Registering form
Name: 
E-mail:   Send me mail.

```

[Glimpse - UIST'11]

Glimpse:

What you glimpse is what you get

Pierre Dragicevic
INRIA

Stéphane Huot
LRI - Université Paris-Sud & CNRS, INRIA

Fanny Chevalier
OCAD University

when interaction design is constrained by technology

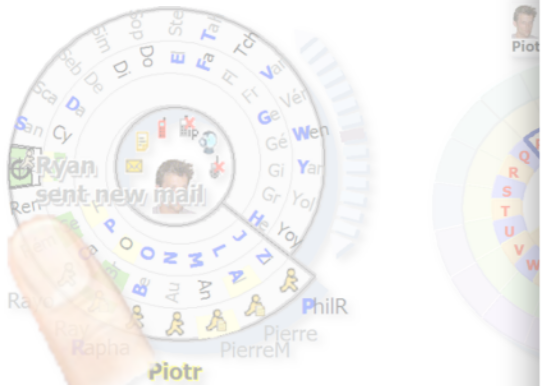
Gliimpse

with P. Dragicevic & F. Chevalier

objective : improve markup language editing

solution : in-place & on-demand animations

.performance iss



[SpiraList & SnailList - No

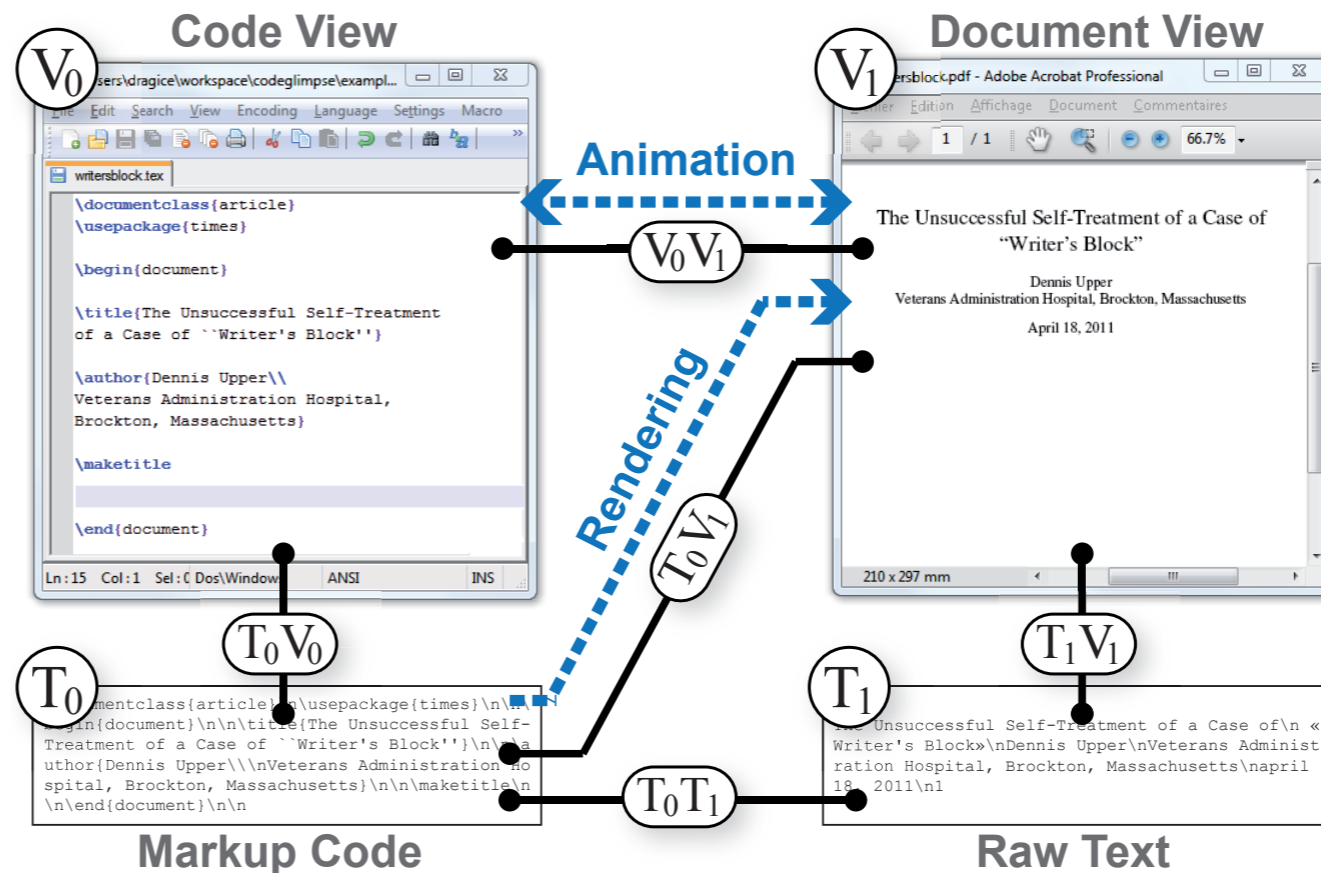
.openness and in

```
<html><body>
<form method="POST" action="someplace/foo.cgi">
  <b>Registering form</b></b>
  <br>Name: <input type="text" name="Name" value="" />
  <br>E-mail: <input type="text" size="10" />
  <br><input type="checkbox" name="c" value="send">Send me mail.
  <br><input type="submit" value="Submit"/>
</form>
</body></html>

Registering form
Name: 
E-mail: 
 Send me mail.

```

[Gliimpse - UIST'11]

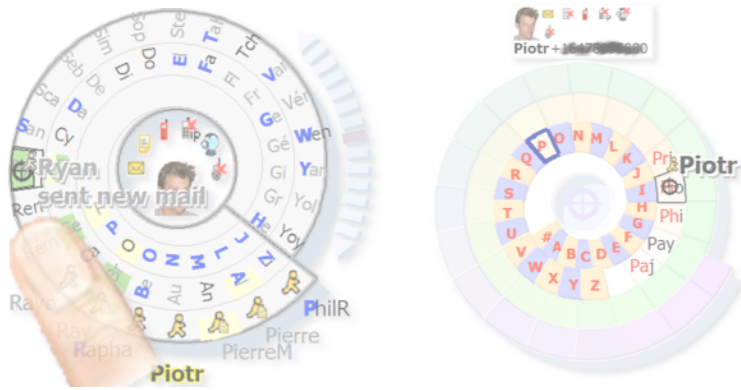


.openness and interoperability issues

- ↳ components inspection
- linking components

when interaction design is constrained by technology

.performance issues



[SpiraList & SnailList - NordiCHI'06 & Interact'07]

.openness and interoperability issues

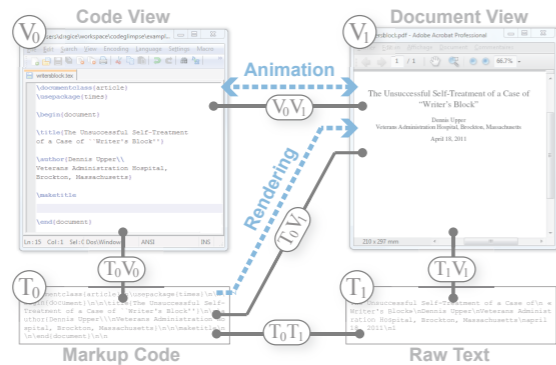
```
<html><body>
<form method="POST" action="someplace/foo.cgi">
  <b>Registering form</b></b>
  <br>Name: <input type="text" name="Name" value="" />
  <br>E-mail: <input type="text" size="10" />
  <br><input type="checkbox" name="c" value="send">Send me mail.
  <br><input type="submit" value="Submit"/>
</form>
</body></html>

Registering form
Name: 
E-mail: 
 Send me mail.


Registering form
Name: 
E-mail: 
 Send me mail.

```

[Glimpse - UIST'11]



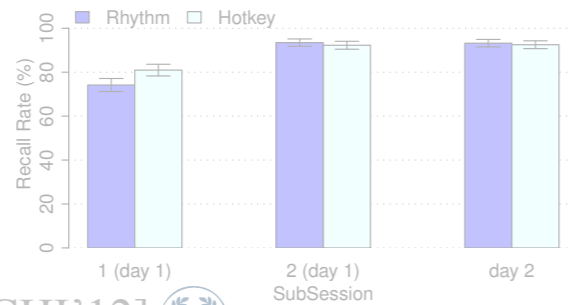
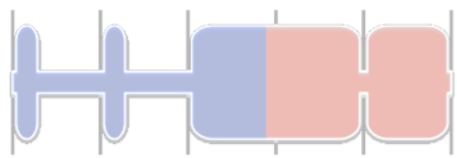
DESIGN
OF INTERACTION
TECHNIQUES

Evaluation
Revision

ENGINEERING
OF INTERACTIVE
SYSTEMS

when interaction design improves technology

.integration of new designs

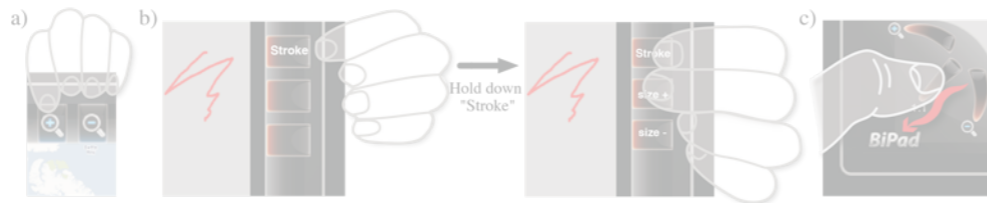


[Rhythmic Interaction - CHI'12]

.replication and new design opportunities



[BiPad - CHI'12]



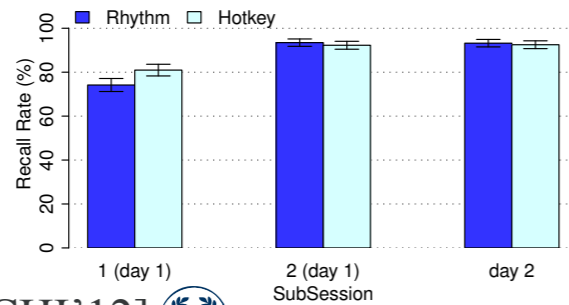
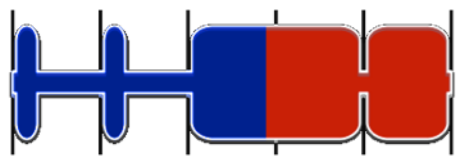
DESIGN
OF INTERACTION
TECHNIQUES

Evaluation
Revision

ENGINEERING
OF INTERACTIVE
SYSTEMS

when interaction design improves technology

.integration of new designs

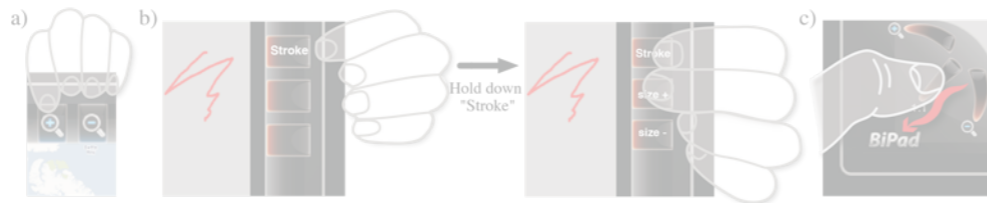


[Rhythmic Interaction - CHI'12]

.replication and new design opportunities



[BiPad - CHI'12]



DESIGN
OF INTERACTION
TECHNIQUES

**Evaluation
Revision**

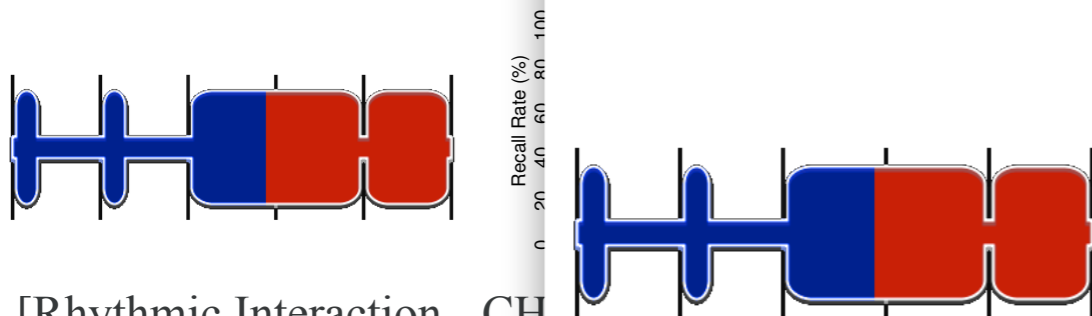
ENGINEERING
OF INTERACTIVE
SYSTEMS

Rhythmic Interaction

with E. Ghomi, G. Faure,
O. Chapuis & M. Beaudouin-Lafon

- objective** : explore rhythmic patterns as an input method
- solution** : design, feasibility & technology

.integration of n



.replication and

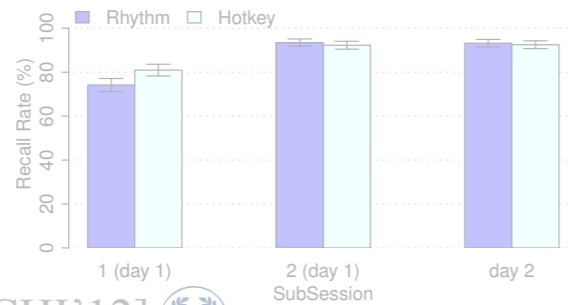
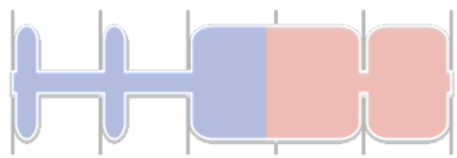


.integration of new designs

- ↳ standard input devices
- universal recognizer

when interaction design improves technology

.integration of new designs

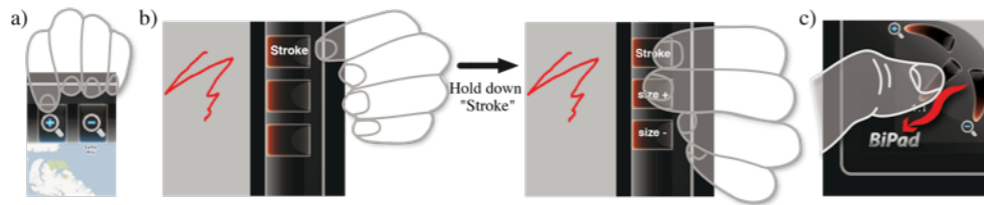


[Rhythmic Interaction - CHI'12]

.replication and new design opportunities



[BiPad - CHI'12]



DESIGN
OF INTERACTION
TECHNIQUES

Evaluation
Revision

ENGINEERING
OF INTERACTIVE
SYSTEMS

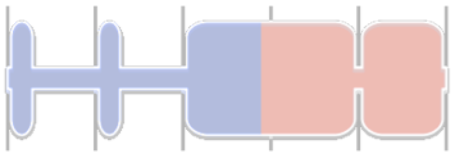
when interaction design improves technology

BiPad (again)

with J. Wagner & W. E. Mackay

objective solution : bimanual interaction in mobility
 : design space, design & **toolkit**

.integration of n



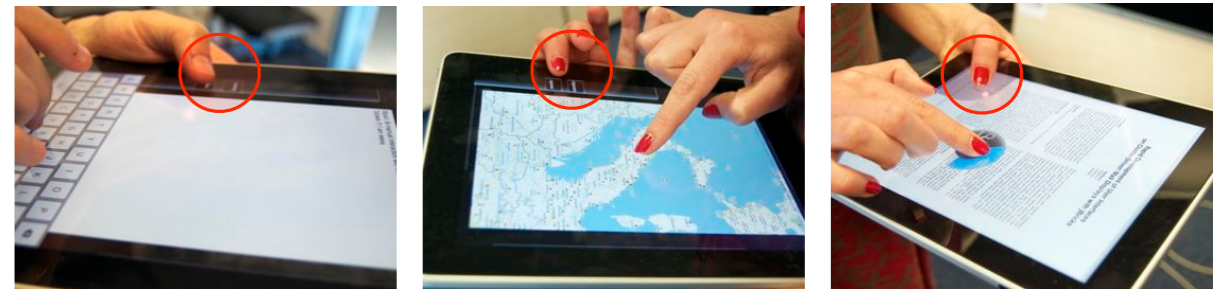
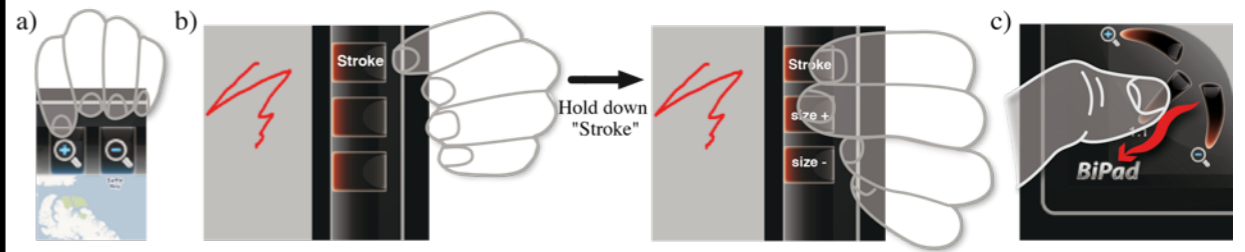
Recall Rate (%)

[Rhythmic Interaction - CHI

.replication and



[BiPad - CHI'12]

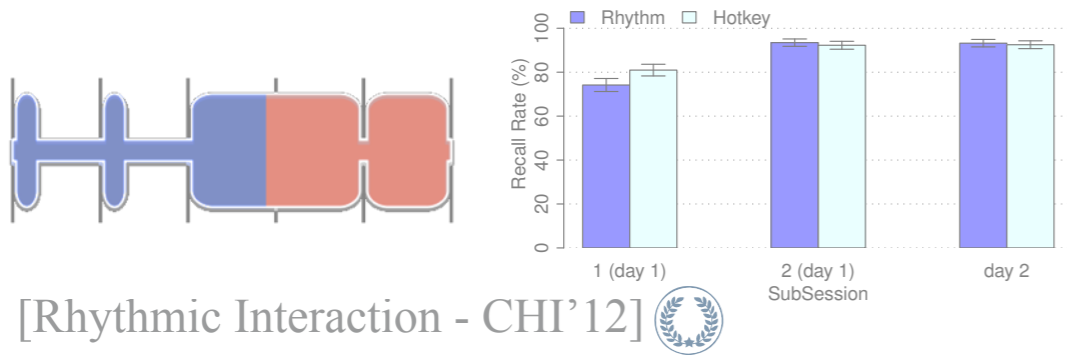


.replication and new design opportunities

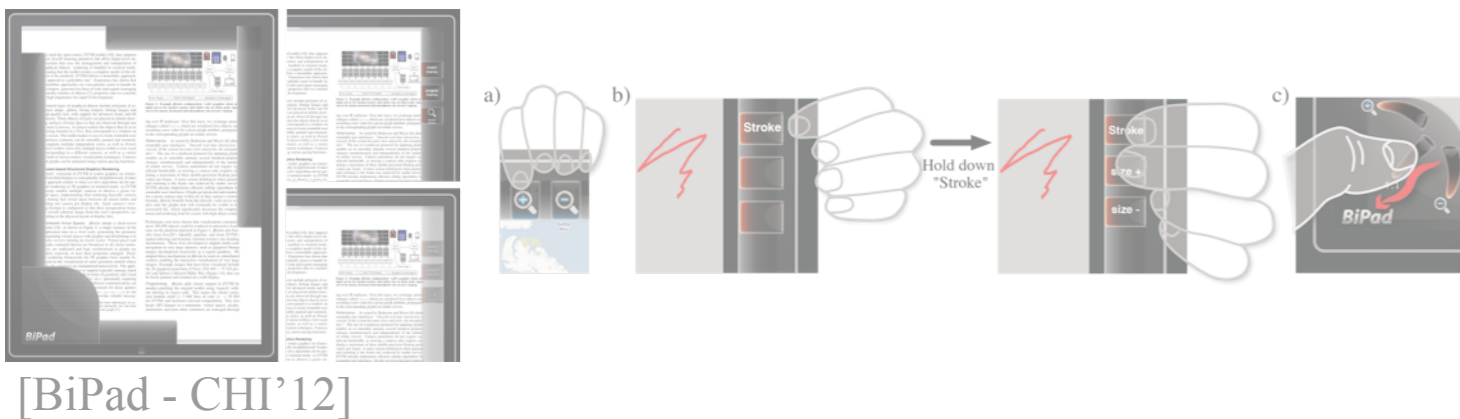
↳ “real-world” applications
 toolkit

when interaction design improves technology

.integration of new designs



.replication and new design opportunities



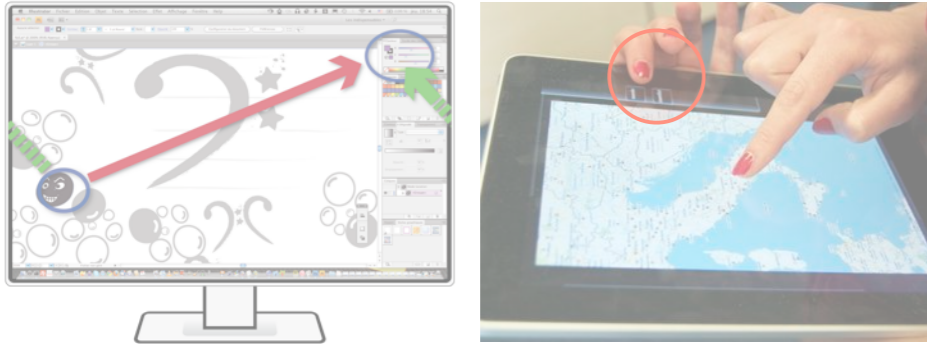
DESIGN
OF INTERACTION
TECHNIQUES

Evaluation
Revision
Extension

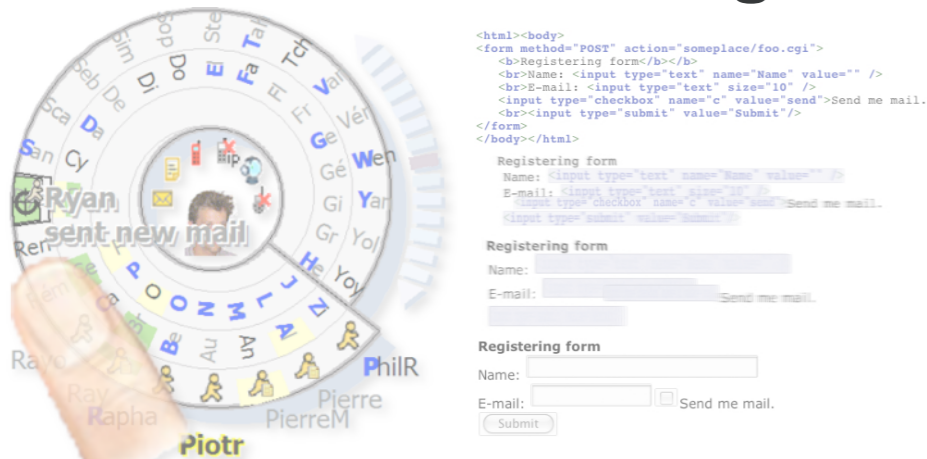
ENGINEERING
OF INTERACTIVE
SYSTEMS

interaction design challenges technology

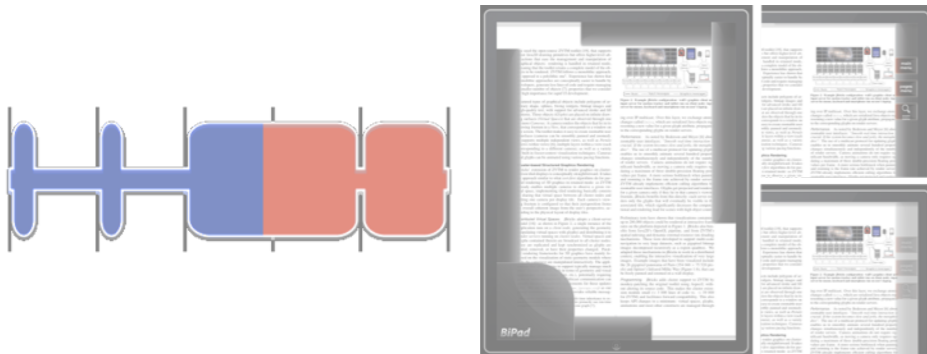
.when interaction design is driven by technology



.when interaction design is constrained by technology



.when interaction design improves technology

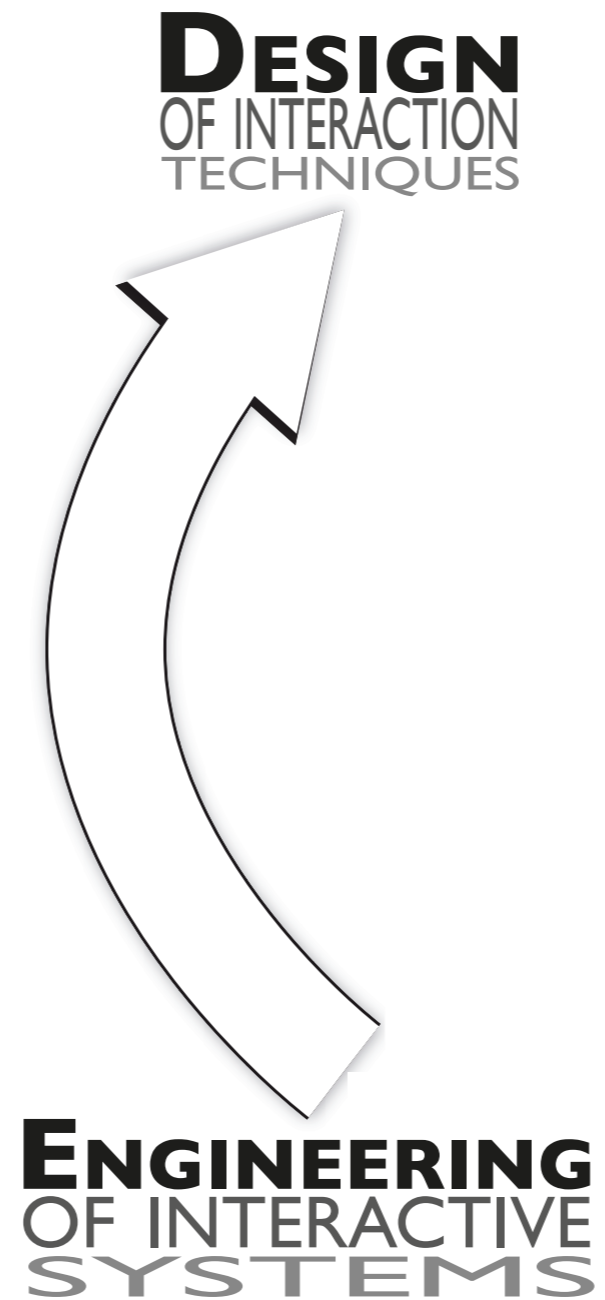


**DESIGN
OF INTERACTION
TECHNIQUES**

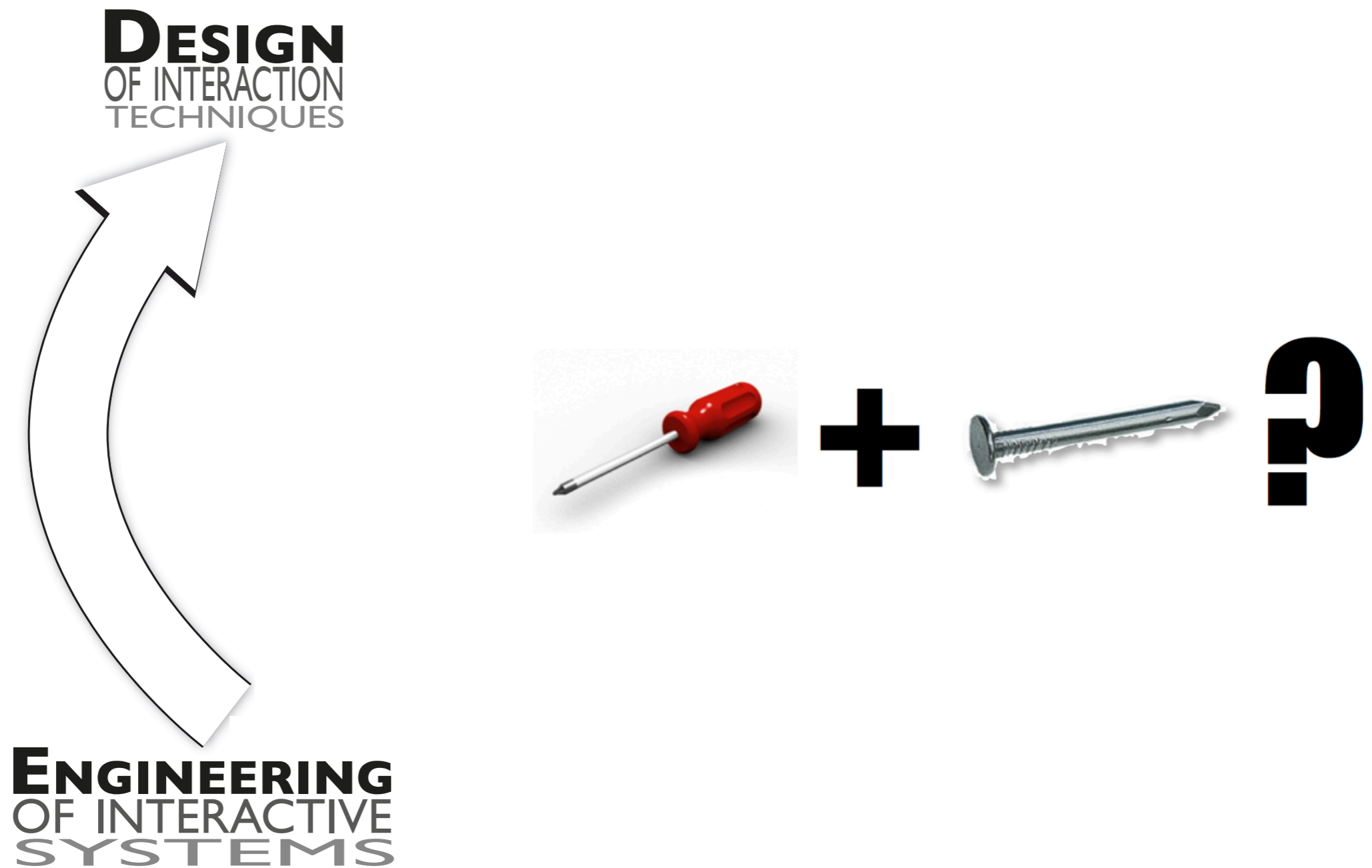
**Evaluation
Revision
Extension**

**ENGINEERING
OF INTERACTIVE
SYSTEMS**

engineering unleashes interaction design

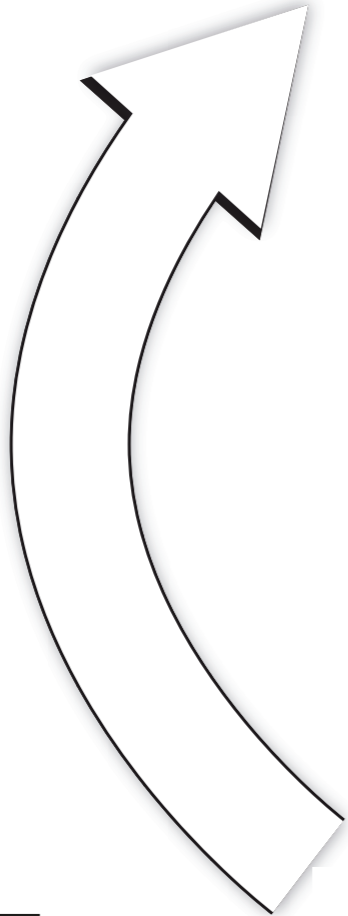


engineering unleashes interaction design



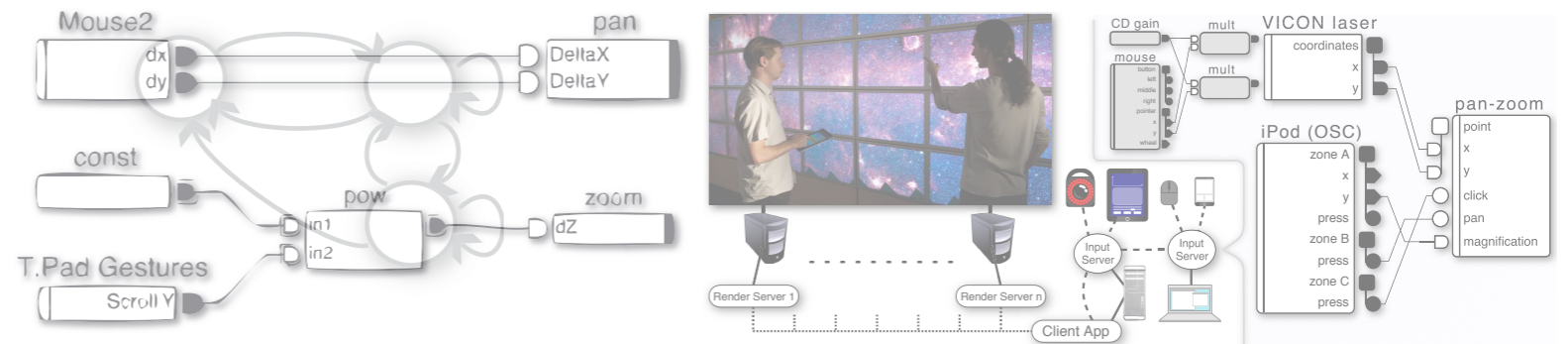
engineering unleashes interaction design

DESIGN
OF INTERACTION
TECHNIQUES

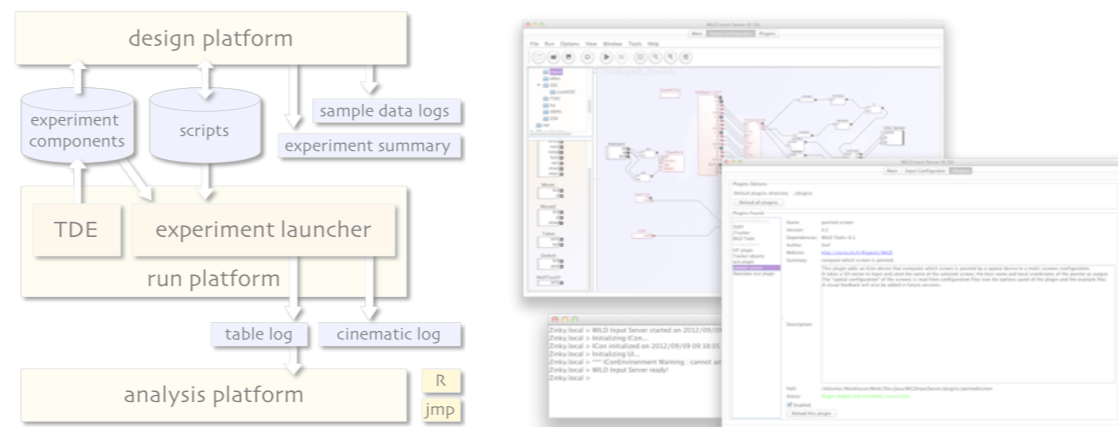


ENGINEERING
OF INTERACTIVE
SYSTEMS

.when technology defines possible designs



.when technology enables the evaluation of designs



.when technology integrates designs

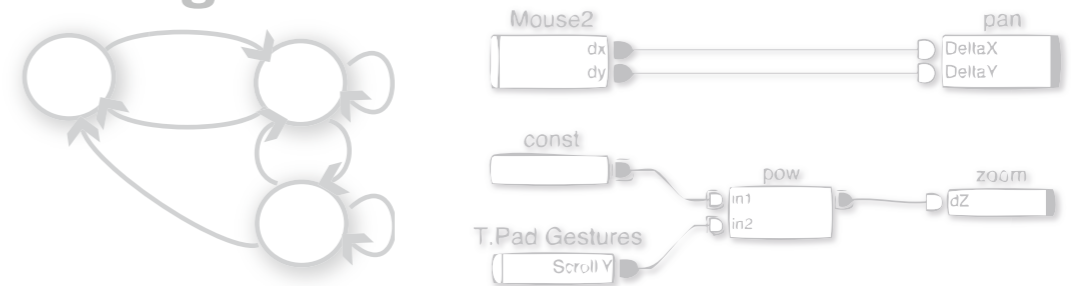


when technology defines possible designs

DESIGN
OF INTERACTION
TECHNIQUES

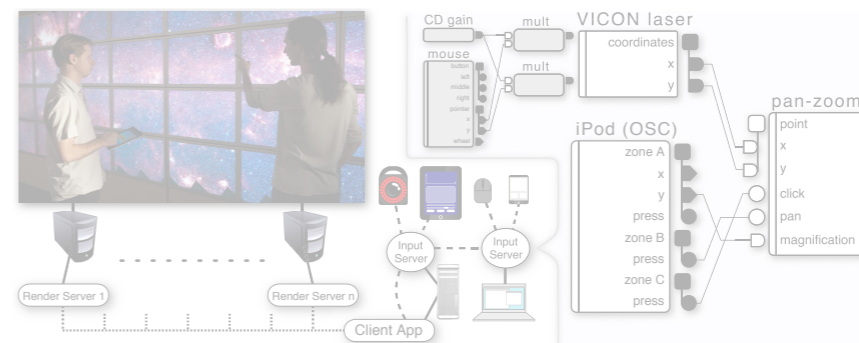
ENGINEERING
OF INTERACTIVE
SYSTEMS

.building blocks and combination



[FlowStates - IHM'09] 

.interoperability and extension



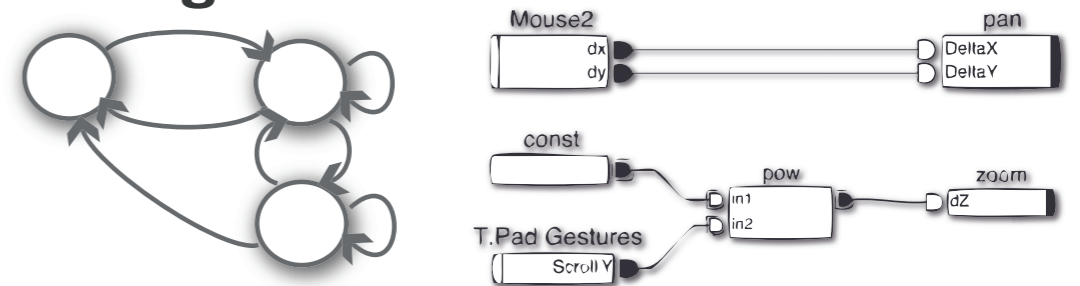
[jBricks - EICS'11]

when technology defines possible designs

DESIGN
OF INTERACTION
TECHNIQUES

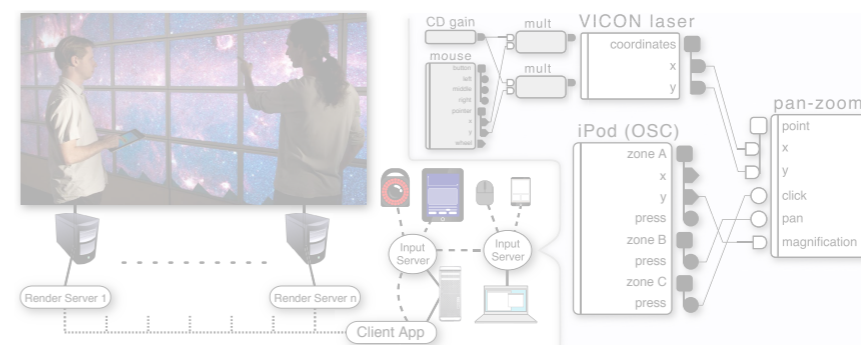
ENGINEERING
OF INTERACTIVE
SYSTEMS

.building blocks and combination



[FlowStates - IHM'09] 

.interoperability and extension



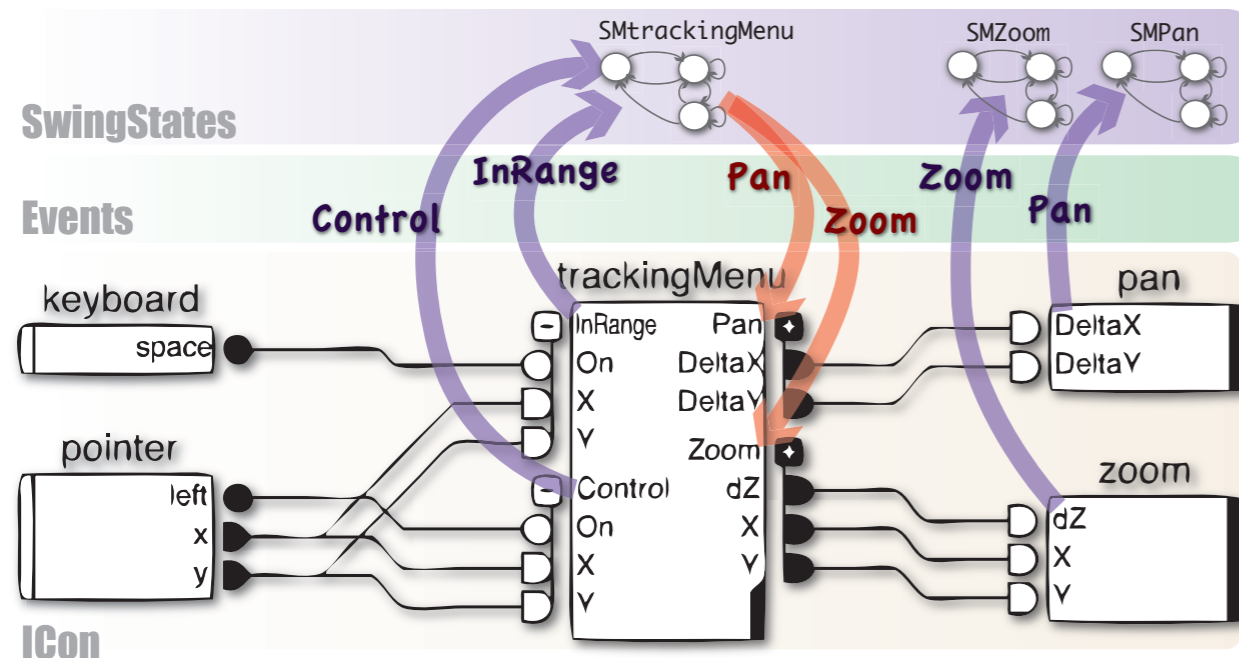
[jBricks - EICS'11]

FlowStates

with C.Appert, P. Dragicevic & M. Beaudouin-Lafon

objective : prototyping advanced interaction

solution : state-transition (SwingStates)
& data-flow (ICon)



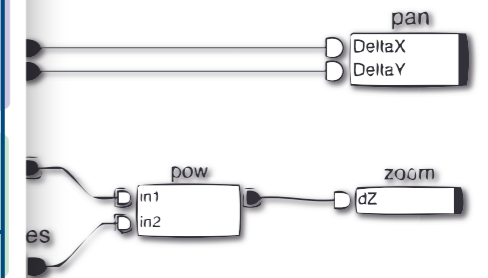
```

new IConStateMachine("zoom", canvas) {
    State idle = new State() {
        Transition zoom = new Event(Zoom.class) { ... };
    };
}
new IConStateMachine("pan", canvas) {
    State idle = new State() {
        Transition pan = new Event(Pan.class) { ... };
    };
}

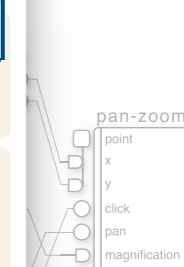
public class Zoom extends IConEvent {
    protected double dZ;
    public void setSlotDZ(double dz) { dZ = dz; }
    public double getSlotDZ() { return dZ; }
    public boolean occurs() { return dZ > 0; }
}

public class Pan extends IConEvent {
    private double dX;
    private double dY;
    public double getSlotDeltaX() { return dX; }
    public double getSlotDeltaY() { return dY; }
    public void setSlotDeltaX(double dx) { dX = dx; }
    public void setSlotDeltaY(double dy) { dY = dy; }
}
    
```

combination



extension



.building blocks and combination

↳ describing = programming
adaptability and dynamicity

FlowStates

with C. Appert, P. Dragicevic & M. Beaudouin-Lafon

objective solution

```

new IConStateMachine("zoom", canvas) {
  State idle = new State() {
    Transition zoom = new Event(Zoom.class) { ... };
  };
}
new IConStateMachine("pan", canvas) {
  State idle = new State() {
    Transition pan = new Event(Pan.class) { ... };
  };
}

public class Zoom extends IConEvent {
  protected double dZ;
  public void setSlotDZ(double dz) { dZ = dz; }
  public double getSlotDZ() { return dZ; }
  public boolean occurs() { return dZ > 0; }
}

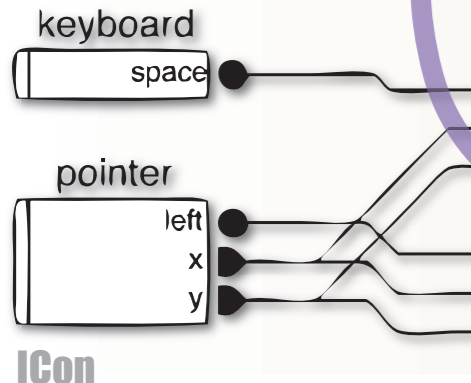
public class Pan extends IConEvent {
  private double dX;
  private double dY;
  public double getSlotDeltaX() { return dX; }
  public double getSlotDeltaY() { return dY; }
  public void setSlotDeltaX(double dx) { dX = dx; }
  public void setSlotDeltaY(double dy) { dY = dy; }
}

```

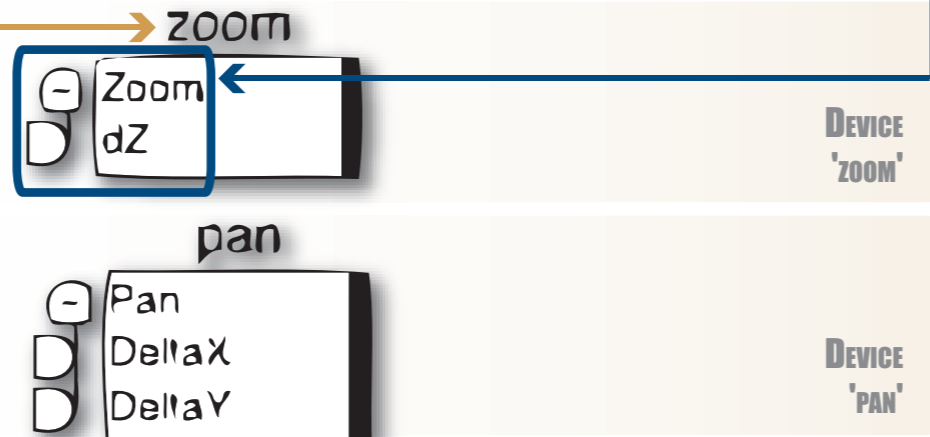
SwingStates

Events

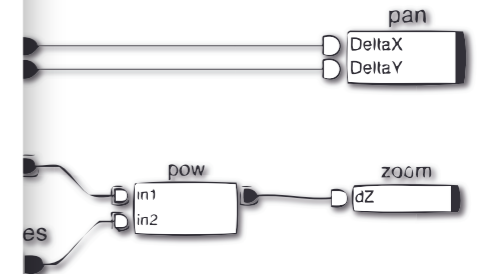
Control



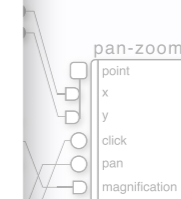
.building t
↳ describ
adaptat



combination

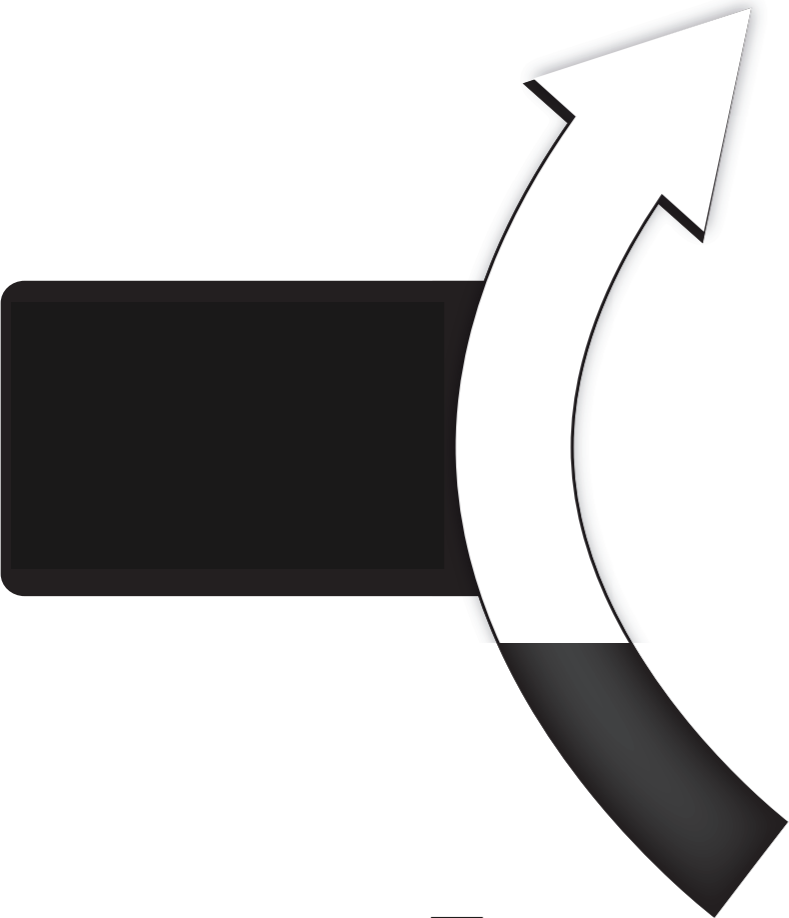


extension



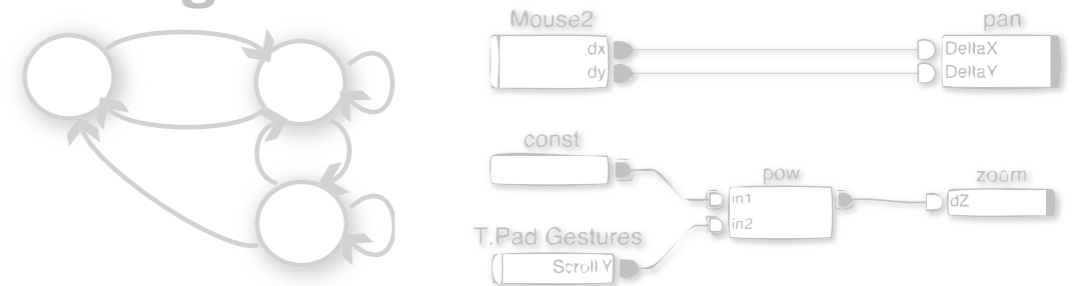
when technology defines possible designs

DESIGN
OF INTERACTION
TECHNIQUES



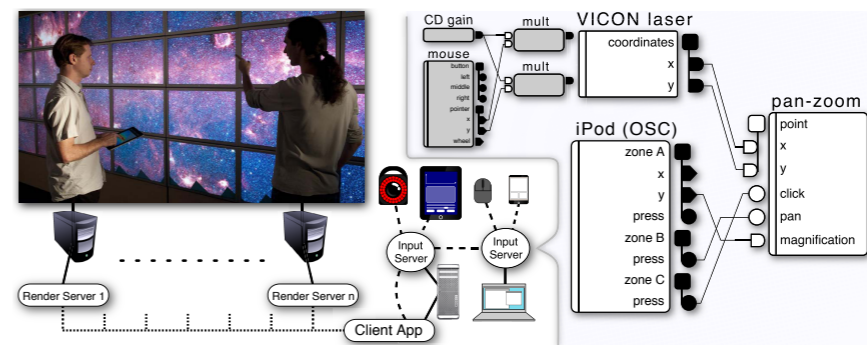
ENGINEERING
OF INTERACTIVE
SYSTEMS

.building blocks and combination



[FlowStates - IHM'09]

.interoperability and extension



[jBricks - EICS'11]

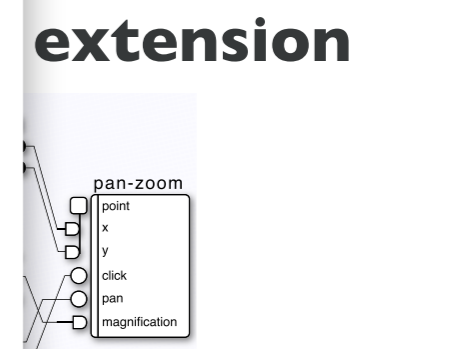
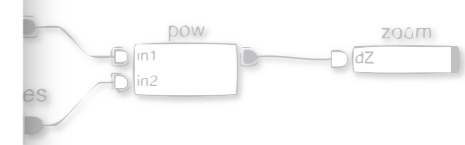
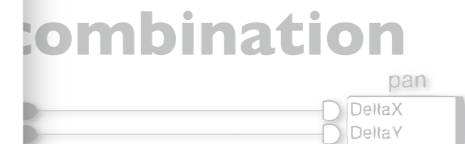
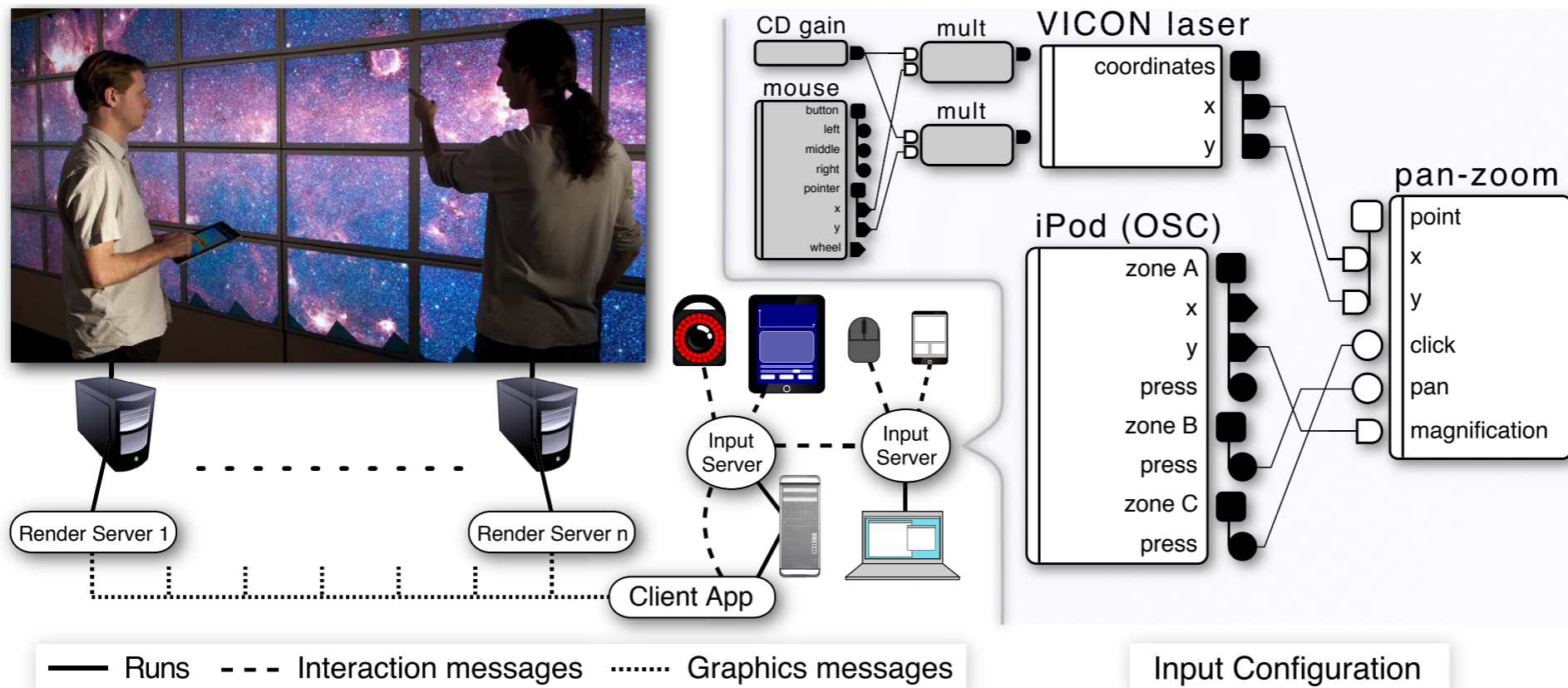
when technology defines possible designs

jBricks

with E. Pietriga, R. Primet & M. Nancel

objective : rich graphics & advanced interaction in multi-surface environments

solution : separation of concerns



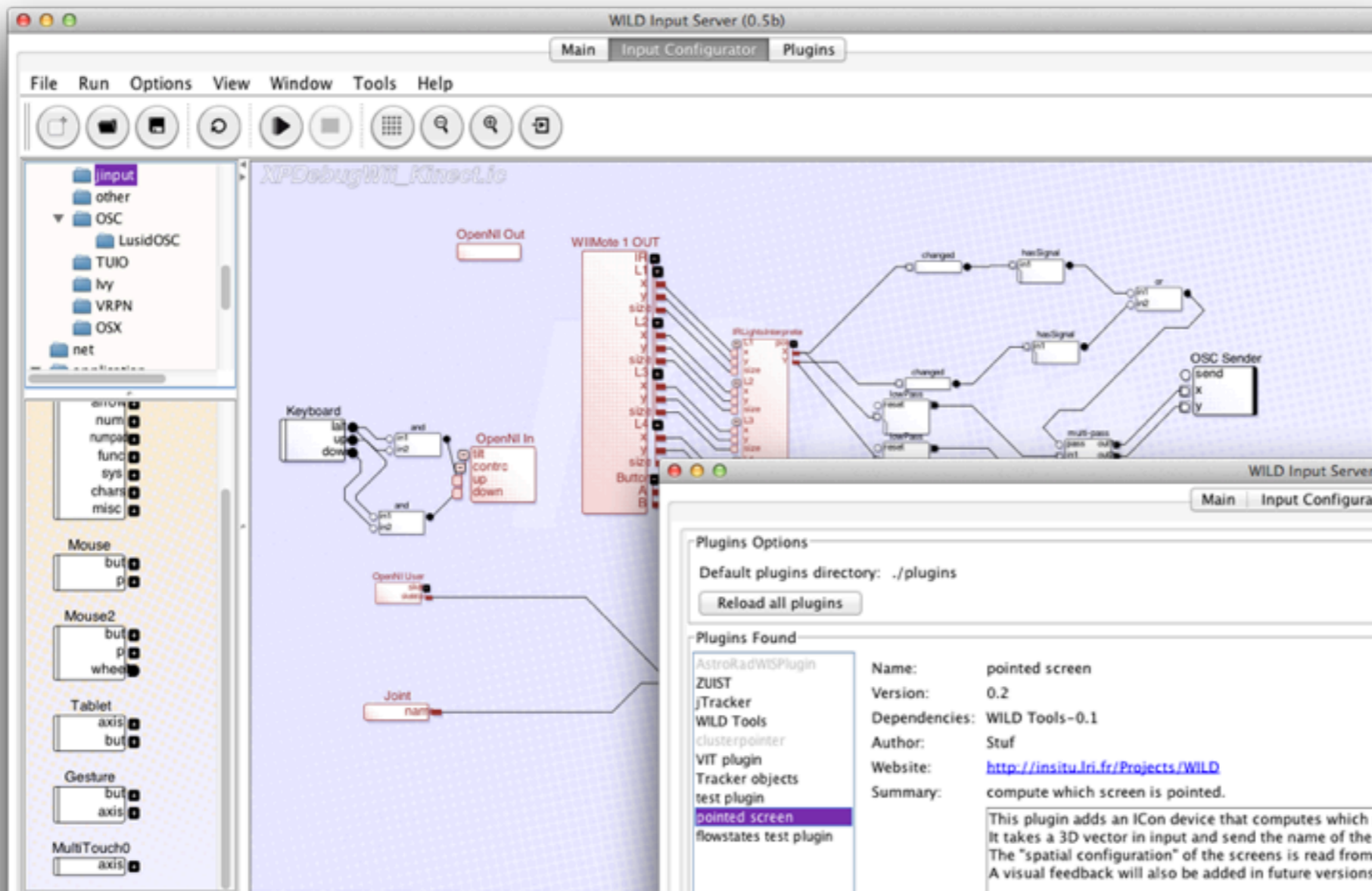
.interoperability and extension

- ↳ separate graphics and interaction
- several levels of combination

Possible designs

M. Nancel

on in



Plugins Options

Default plugins directory: ./plugins

Reload all plugins

Plugins Found

AstroRadWSPlugin	Name:	pointed screen
ZUIST	Version:	0.2
jTracker	Dependencies:	WILD Tools-0.1
WILD Tools	Author:	Stuf
clusterpointer	Website:	http://insitu.lri.fr/Projects/WILD
VIT plugin	Summary:	compute which screen is pointed.
Tracker objects		
test plugin		
pointed screen		
flowstates test plugin		

Description:

This plugin adds an ICon device that computes which screen is pointed by a spatial device in a multi-screens configuration. It takes a 3D vector in input and send the name of the selected screen, the host name and local coordinates of the pointer as output. The "spatial configuration" of the screens is read from configuration files (see the options panel of the plugin and the example file). A visual feedback will also be added in future versions.

Path: /Volumes/Warehouse/Work/Dev/Java/WILDInputServer/plugins/pointedscreen

Status: Plugin loaded and initialized successfully.

Enabled

Reload this plugin

(Render Server 1) (Render Server n)

```
Zinky.local > WILD Input Server started on 2012/09/09
Zinky.local > Initializing ICon...
Zinky.local > ICon initialized on 2012/09/09 09:38:05
Zinky.local > Initializing UI...
Zinky.local > *** IConEnvironment Warning : cannot ad
Zinky.local > WILD Input Server ready!
Zinky.local >
```

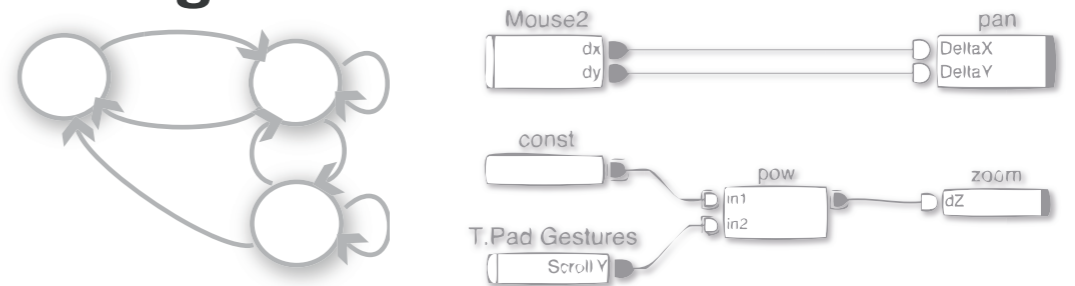

when technology defines possible designs

DESIGN
OF INTERACTION
TECHNIQUES

Prototyping

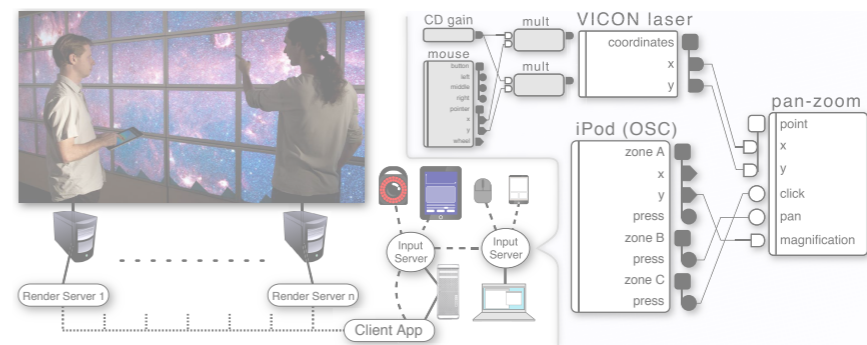
ENGINEERING
OF INTERACTIVE
SYSTEMS

.building blocks and combination



[FlowStates - IHM'09]

.interoperability and extension



[jBricks - EICS'11]

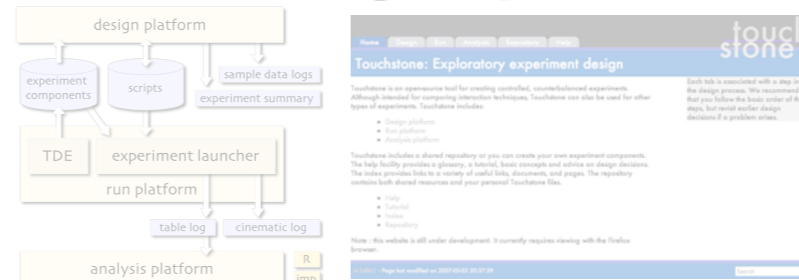
when technology enables the evaluation of designs

**DESIGN
OF INTERACTION
TECHNIQUES**

Prototyping

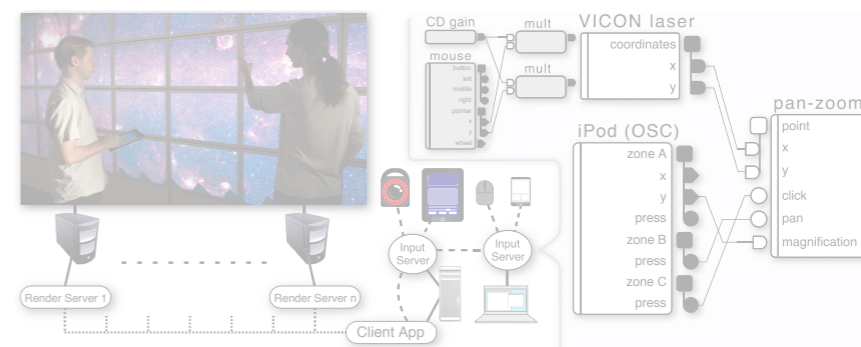
**ENGINEERING
OF INTERACTIVE
SYSTEMS**

.conducting experiments



[Touchstone - Mackay et al., CHI'07]

.software quality



[jBricks - EICS'11]

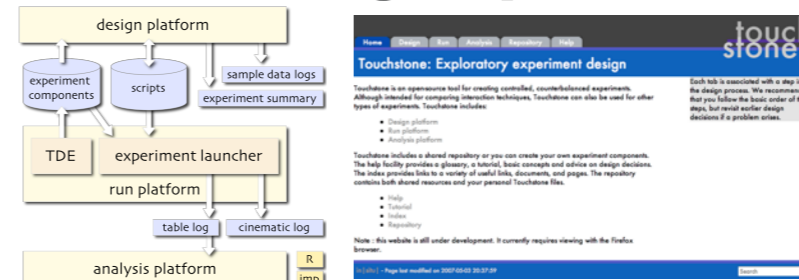
when technology enables the evaluation of designs

DESIGN
OF INTERACTION
TECHNIQUES

Prototyping

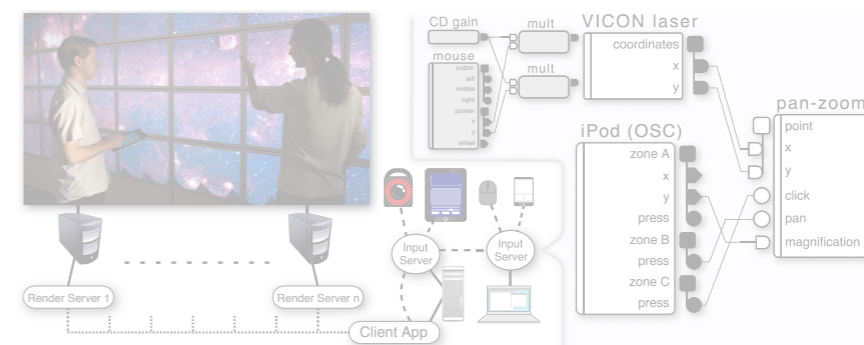
ENGINEERING
OF INTERACTIVE
SYSTEMS

.conducting experiments



[Touchstone - Mackay et al., CHI'07]

.software quality



[jBricks - EICS'11]

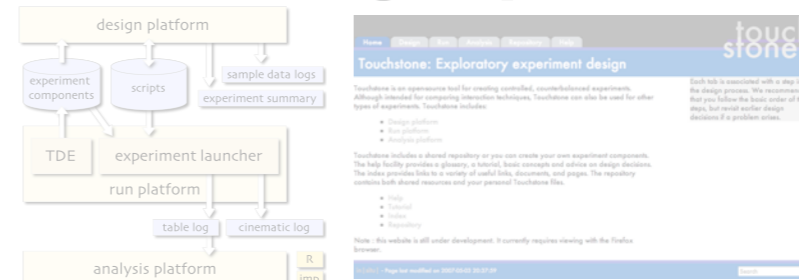
when technology enables the evaluation of designs

DESIGN
OF INTERACTION
TECHNIQUES

Prototyping

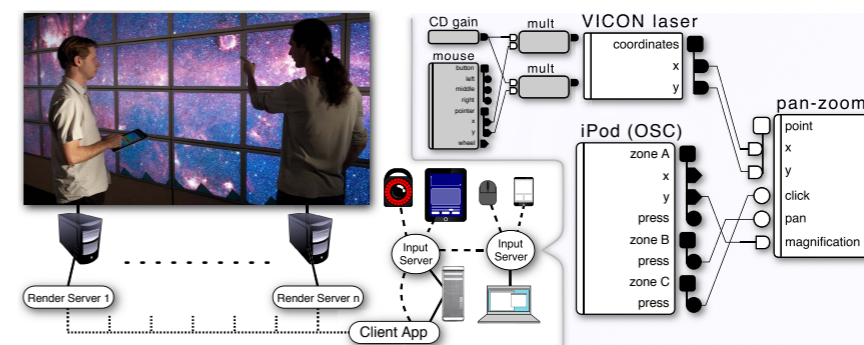
ENGINEERING
OF INTERACTIVE
SYSTEMS

.conducting experiments



[Touchstone - Mackay et al., CHI'07]

.software quality



[jBricks - EICS'11]

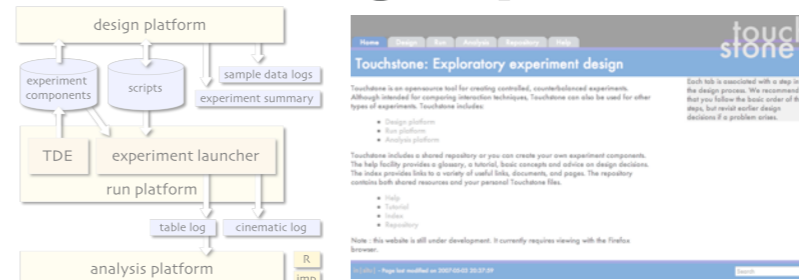
when technology enables the evaluation of designs

DESIGN
OF INTERACTION
TECHNIQUES

**Evaluation
Prototyping**

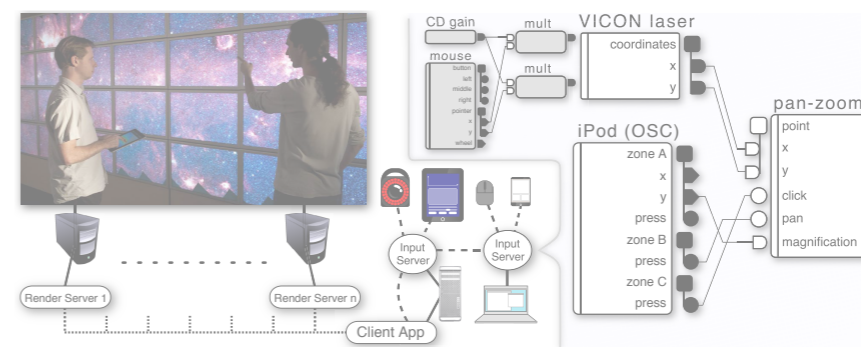
ENGINEERING
OF INTERACTIVE
SYSTEMS

.conducting experiments



[Touchstone - Mackay et al., CHI'07]

.software quality



[jBricks - EICS'11]

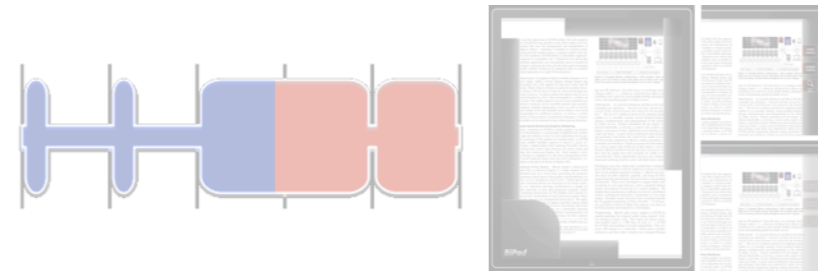
when technology integrates designs

DESIGN
OF INTERACTION
TECHNIQUES

Evaluation
Prototyping

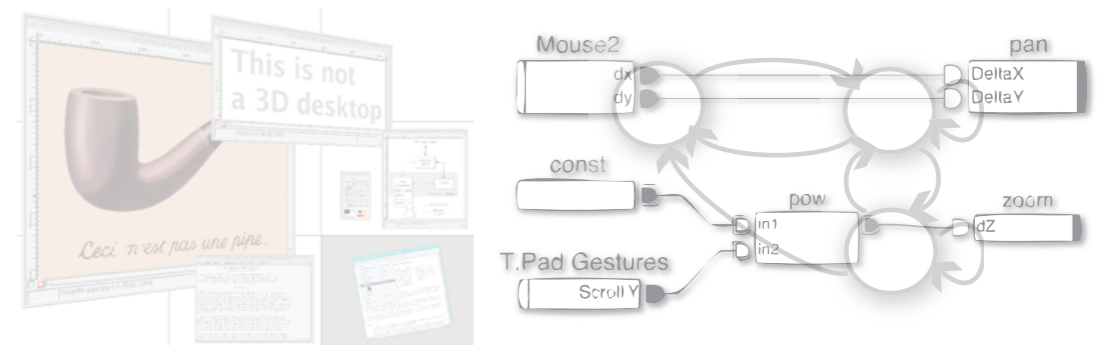
ENGINEERING
OF INTERACTIVE
SYSTEMS

.toolkit or application level



[Rhythmic Interaction, BiPad - CHI'12]

.opening closed systems



[Metisse - Chapuis and Roussel, UIST'05] [FlowStates - IHM'09]

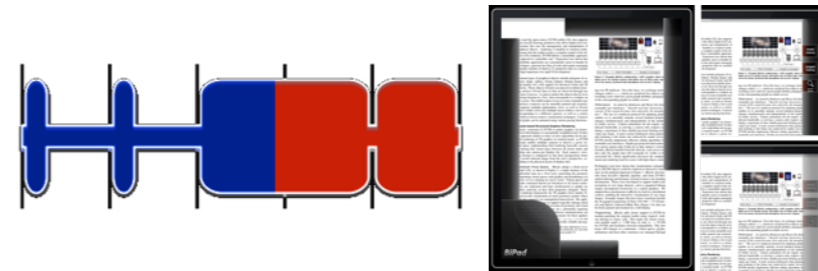
when technology integrates designs

DESIGN
OF INTERACTION
TECHNIQUES

Evaluation
Prototyping

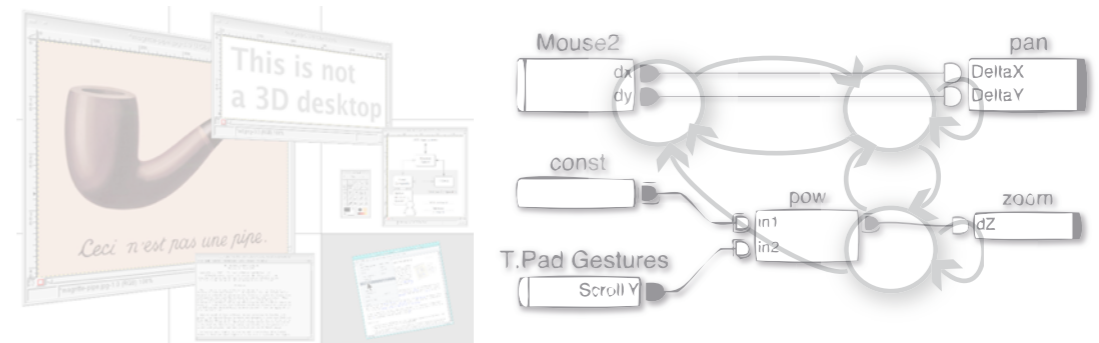
ENGINEERING
OF INTERACTIVE
SYSTEMS

.toolkit or application level



[Rhythmic Interaction, BiPad - CHI'12]

.opening closed systems



[Metisse - Chapuis and Roussel, UIST'05] [FlowStates - IHM'09]

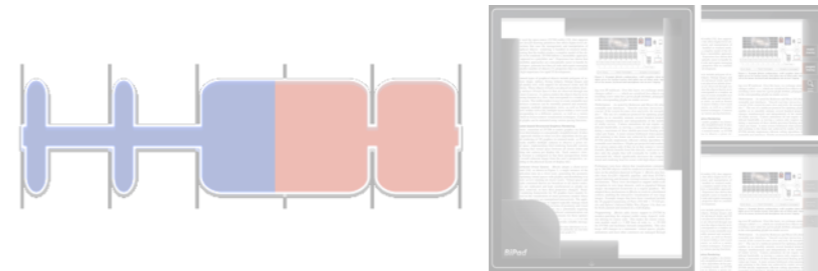
when technology integrates designs

DESIGN
OF INTERACTION
TECHNIQUES

Evaluation
Prototyping

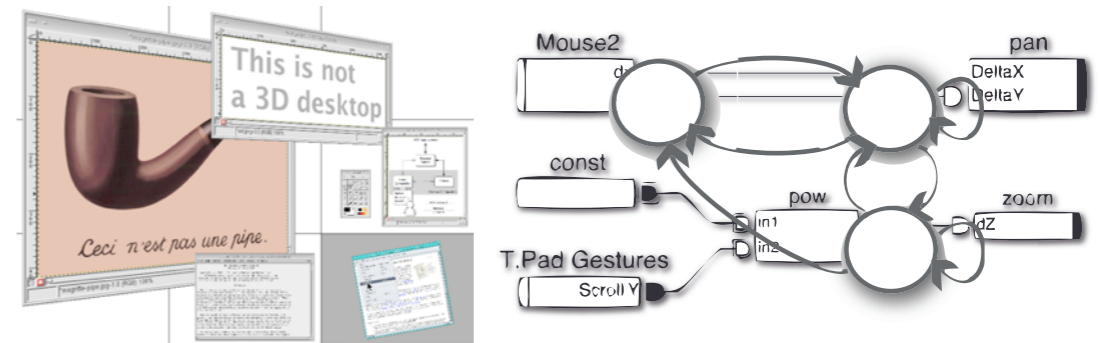
ENGINEERING
OF INTERACTIVE
SYSTEMS

.toolkit or application level



[Rhythmic Interaction, BiPad - CHI'12]

.opening closed systems



[Metisse - Chapuis and Roussel, UIST'05] [FlowStates - IHM'09]

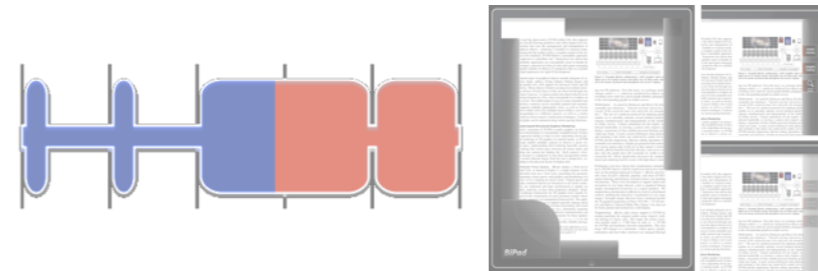
when technology integrates designs

DESIGN
OF INTERACTION
TECHNIQUES

Integration
Evaluation
Prototyping

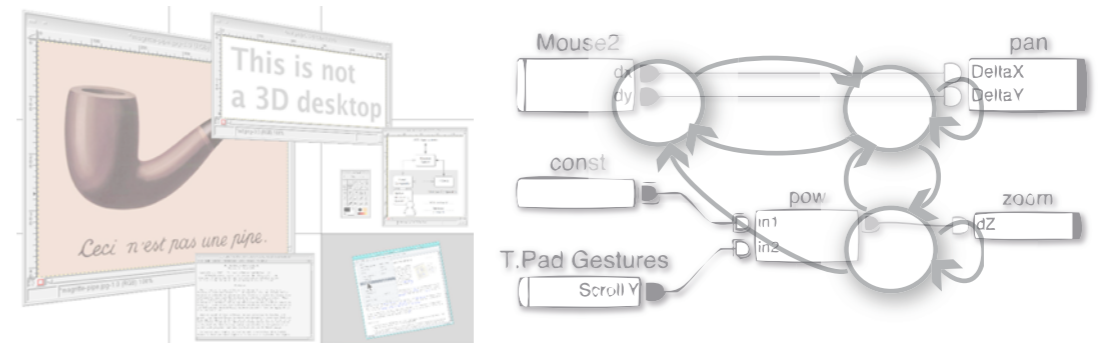
ENGINEERING
OF INTERACTIVE
SYSTEMS

.toolkit or application level



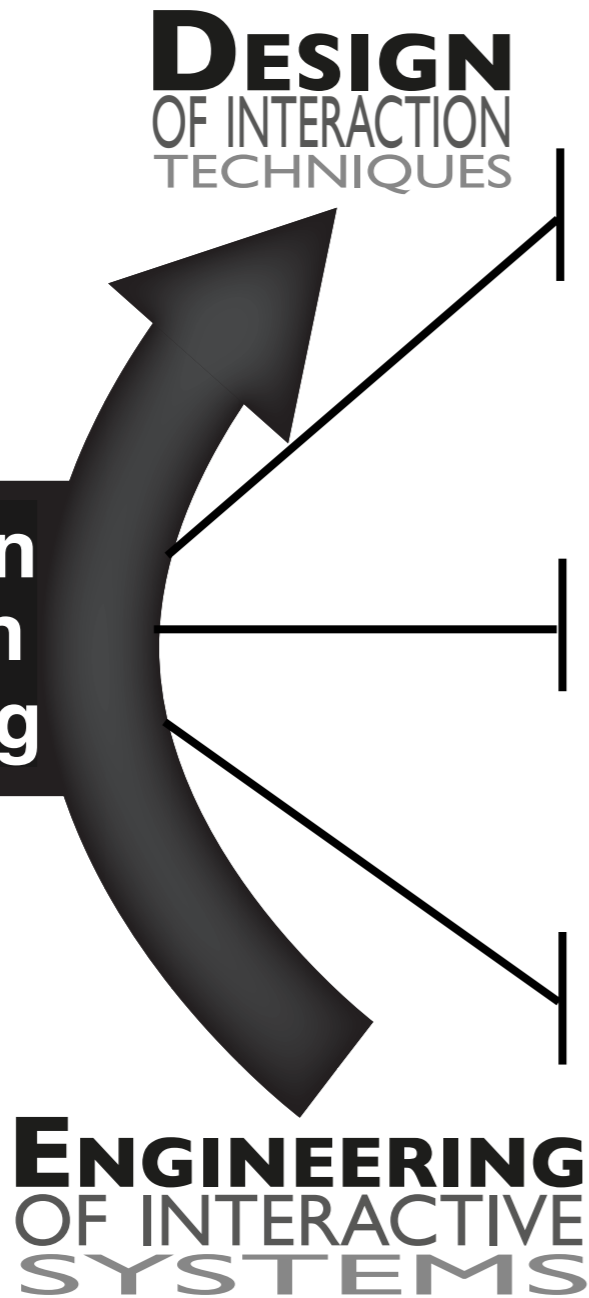
[Rhythmic Interaction, BiPad - CHI'12]

.opening closed systems

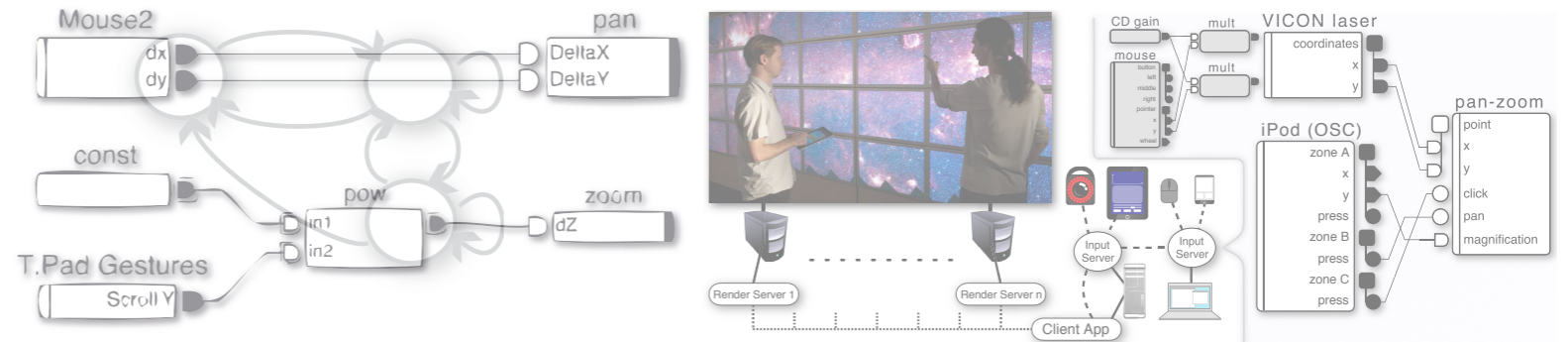


[Metisse - Chapuis and Roussel, UIST'05] [FlowStates - IHM'09]

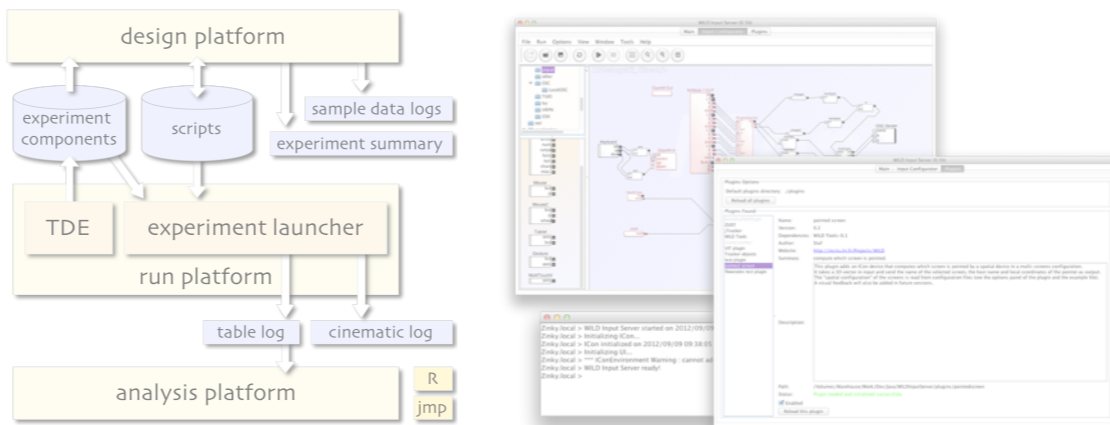
engineering unleashes interaction design



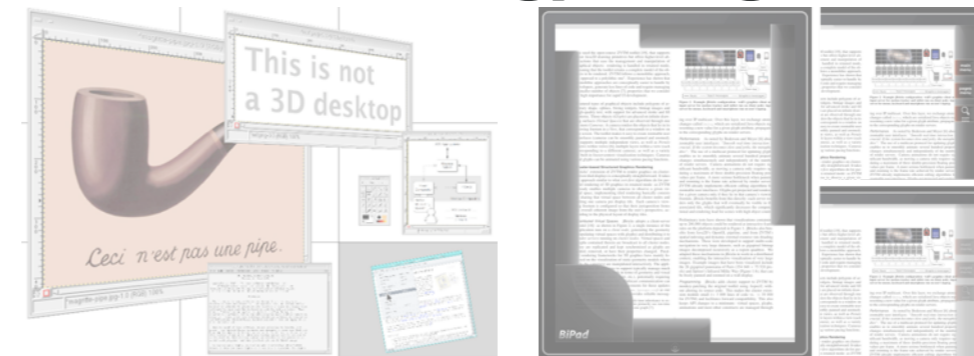
.when technology defines possible designs

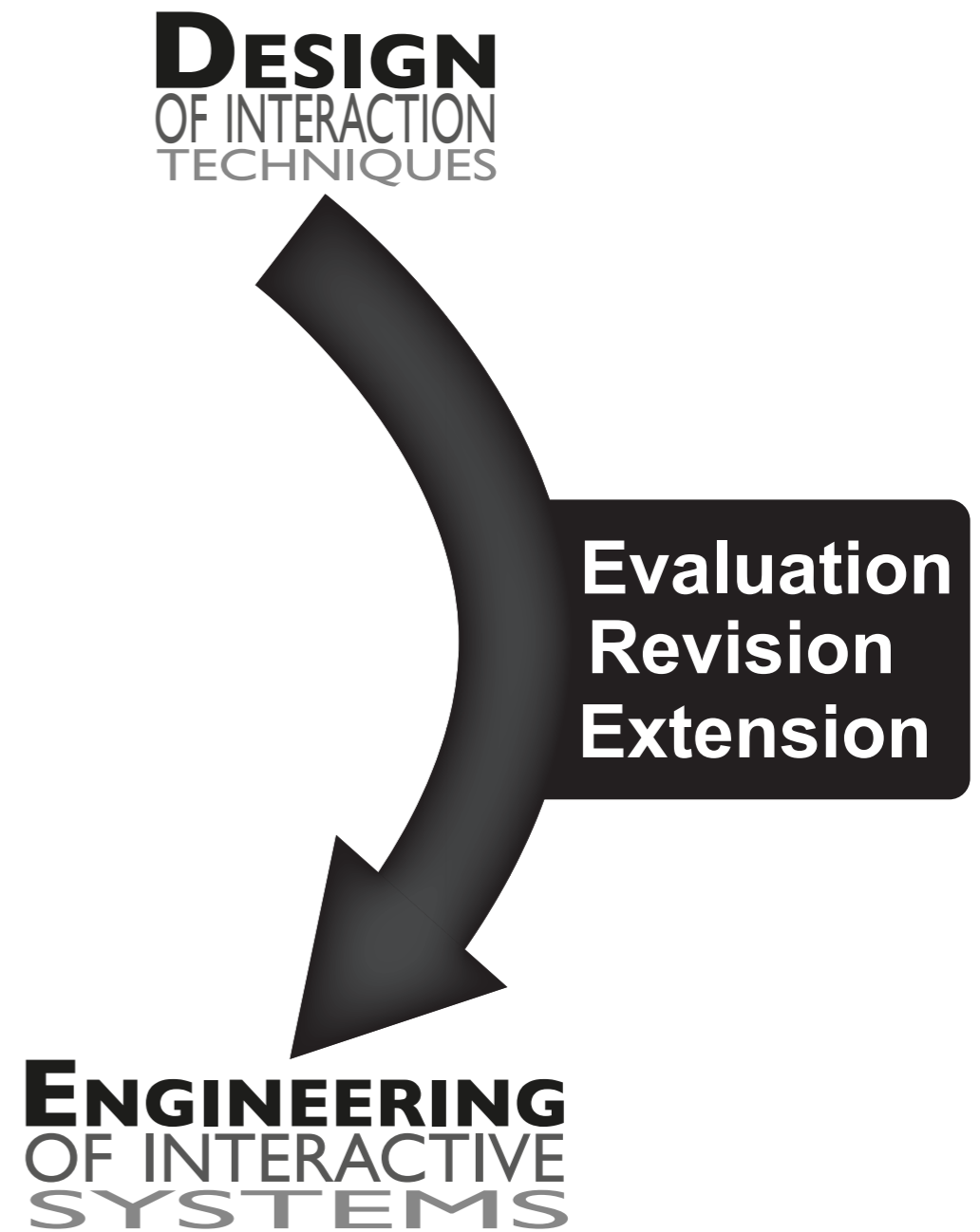
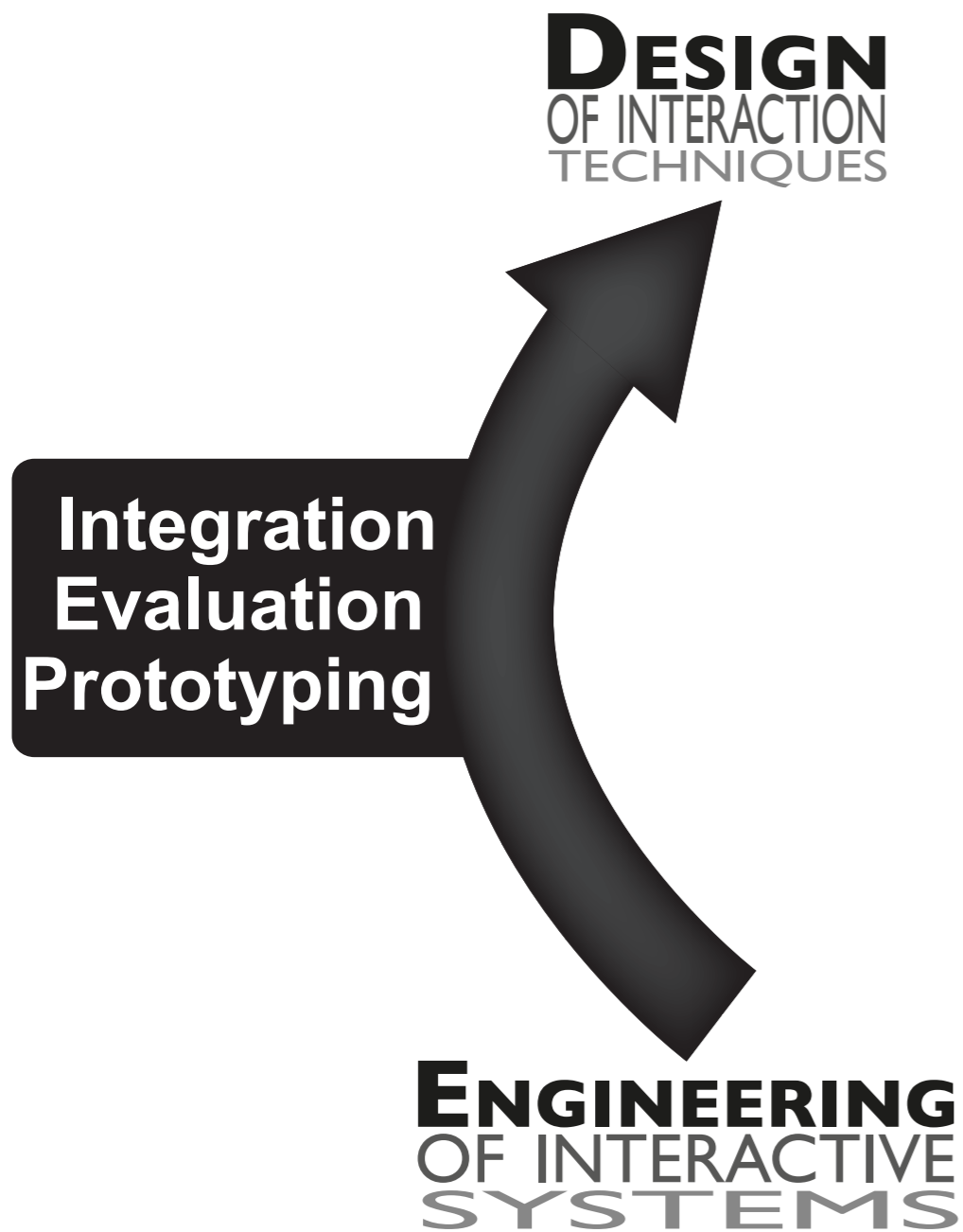


.when technology enables the evaluation of designs

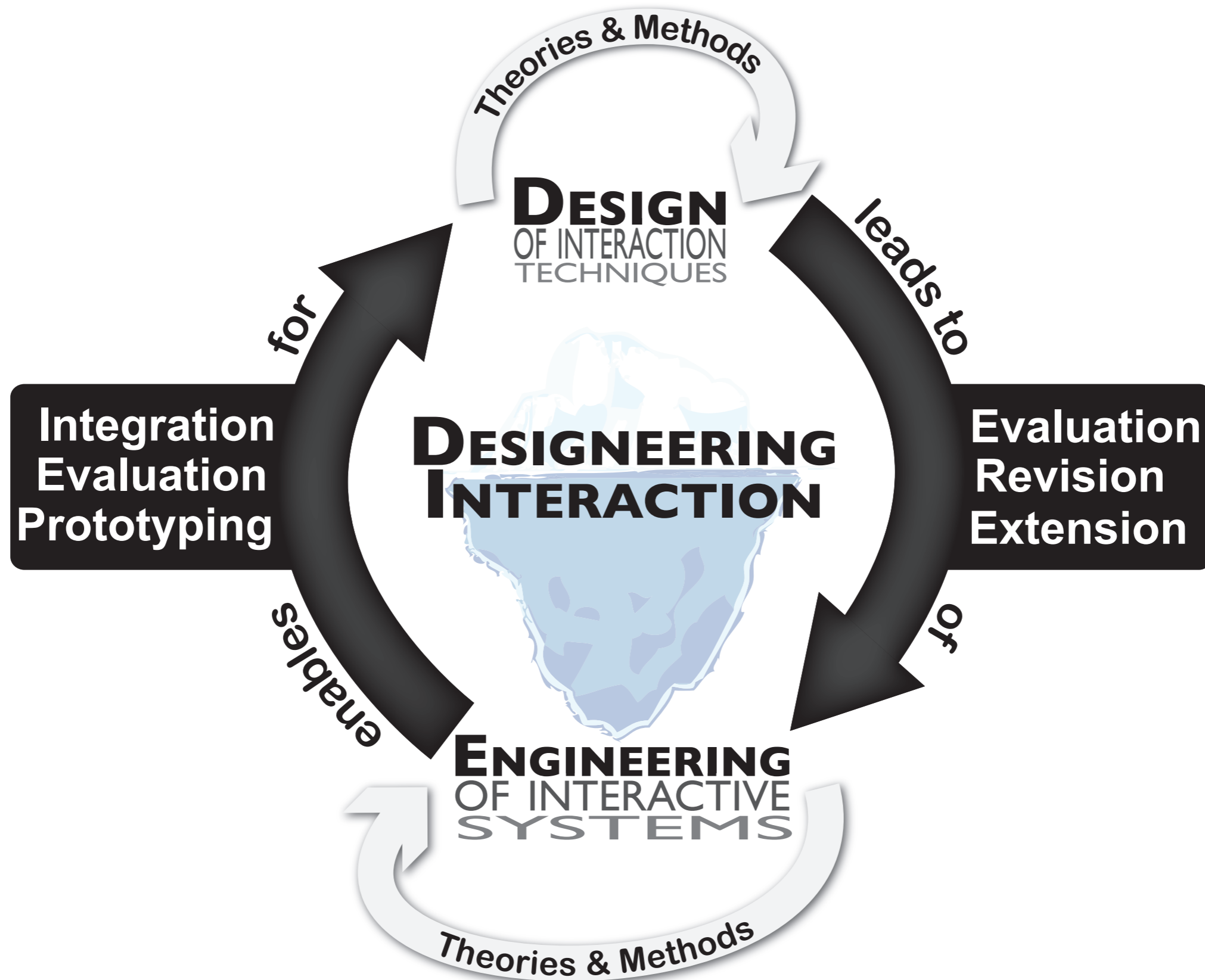


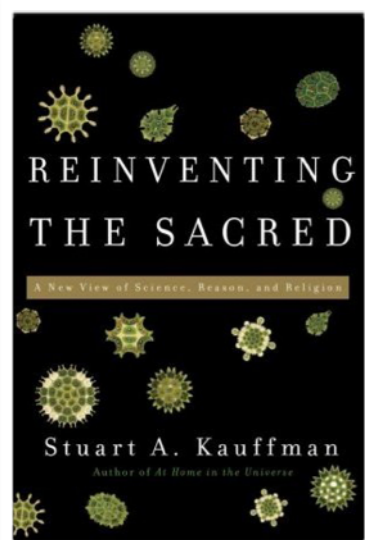
.when technology integrates designs





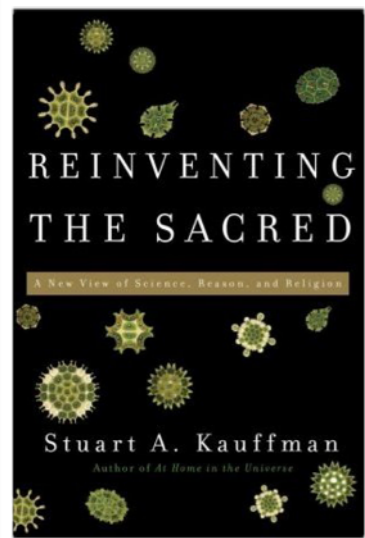
the cycle of Designeering Interaction





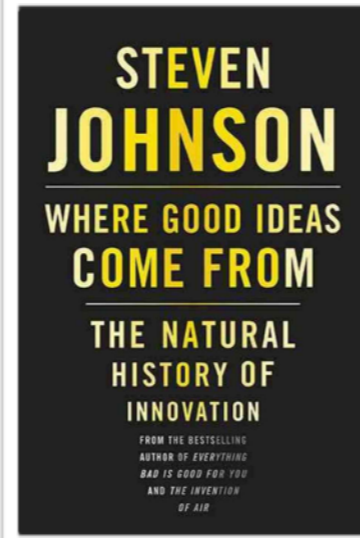
“Consider a reaction graph with N molecular species, polymer sequences of A , and B monomers of diverse lengths. **Call**

this initial N the actual. Now ask the organic chemist to draw all the reactions that these N species might undergo [...]. It may well be that the products of some of these single-step reactions will not be among the initial N in the “actual” but will be new molecular species. **Call the set of new molecular species reachable in a single-reaction step from the actual, the adjacent possible.** [...] The initial plus its adjacent possible can be considered a new actual, which will then have a new adjacent possible.”



“Consider a reaction graph with N molecular species, polymer sequences of A , and B monomers of diverse lengths. **Call**

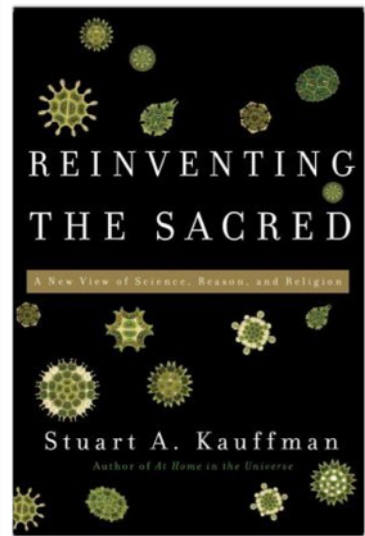
this initial N the actual. Now ask the organic chemist to draw all the reactions that these N species might undergo [...]. It may well be that the products of some of these single-step reactions will not be among the initial N in the “actual” but will be new molecular species. **Call the set of new molecular species reachable in a single-reaction step from the actual, the adjacent possible.** [...] The initial plus its adjacent possible can be considered a new actual, which will then have a new adjacent possible.”



“[...] we like to think of breakthrough ideas as sudden accelerations on the timeline, where a genius jumps ahead fifty years and invents something that normal minds [...]

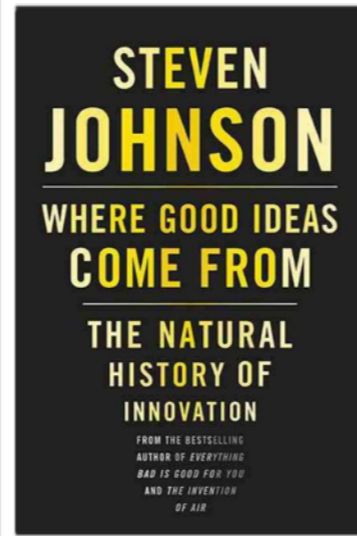
couldn't possibly have come up with. But the truth is that **technological (and scientific) advances rarely break out of the adjacent possible;** the history of cultural progress is, almost without exception, a story of one door leading to another door, exploring the palace one room at a time. But of course, human minds are not bound by the finite laws of molecule formation, and so **every now and then an idea occurs to someone that teleport us forward a few rooms,** skipping some exploratory steps in the adjacent possible. But **those ideas almost always end up being short-term failures,** precisely because they have skipped ahead.”

inspiration



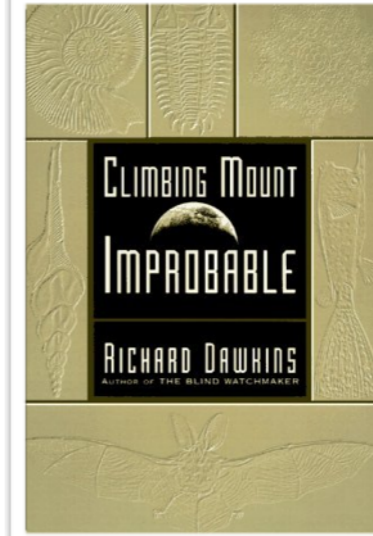
“Consider a reaction graph with N molecular species, polymer sequences of A , and B monomers of diverse lengths. **Call**

this initial N the actual. Now ask the organic chemist to draw all the reactions that these N species might undergo [...]. It may well be that the products of some of these single-step reactions will not be among the initial N in the “actual” but will be new molecular species. **Call the set of new molecular species reachable in a single-reaction step from the actual, the adjacent possible.** [...] The initial plus its adjacent possible can be considered a new actual, which will then have a new adjacent possible.”

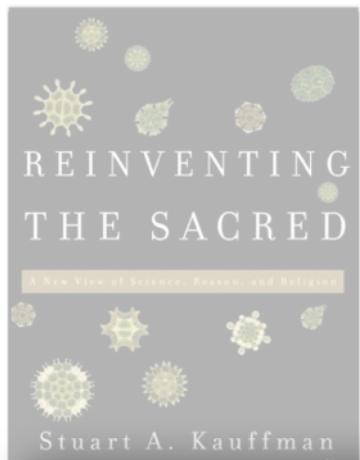


couldn't possibly have come up with. But the truth is that **technological (and scientific) advances rarely break out of the adjacent possible**; the history of cultural progress is, almost without exception, a story of one door leading to another door, exploring the palace one room at a time. But of course, human minds are not bound by the finite laws of molecule formation, and so **every now and then an idea occurs to someone that teleport us forward a few rooms**, skipping some exploratory steps in the adjacent possible. But **those ideas almost always end up being short-term failures**, precisely because they have skipped ahead.”

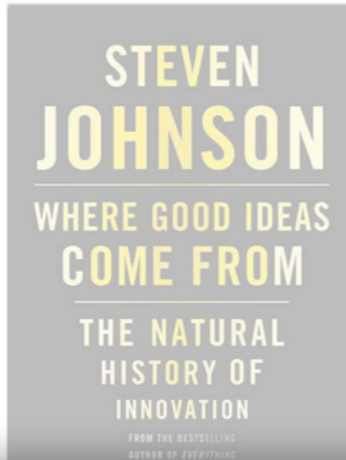
“[...] we like to think of breakthrough ideas as sudden accelerations on the timeline, where a genius jumps ahead fifty years and invents something that normal minds [...]



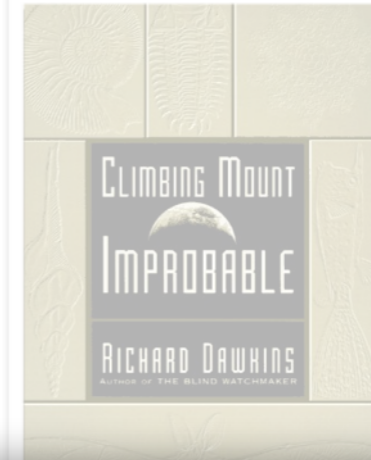
“There can be **no sudden leaps upward** – no precipitous increases in ordered complexity. Second, there can be **no going downhill** – species can't get worse as a prelude to getting better. Third, **there may be more than one peak** – more than one way of solving the same problem, all flourishing in the world.”



“Consider a reaction graph with N molecular species, polymer sequences of A , and



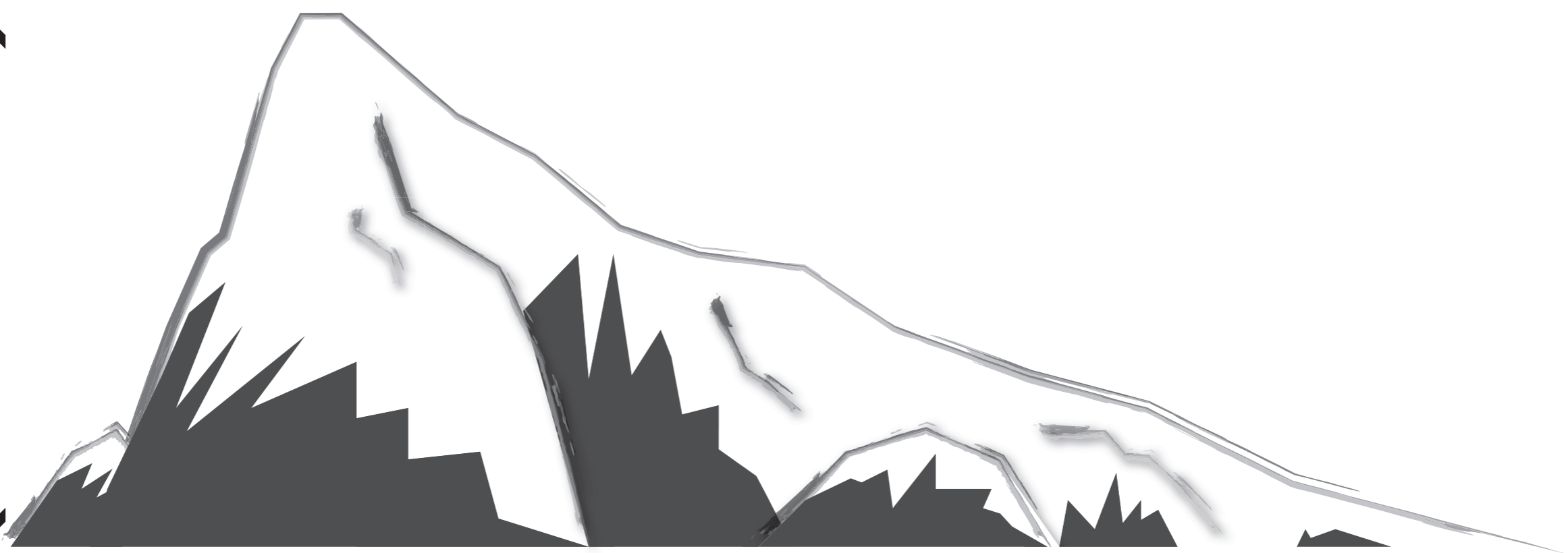
“[...] we like to think of breakthrough ideas as sudden accelerations on the timeline, where a genius jumps ahead fifty years and invents



“There can be no

The Adjacent Possible

Mount Improbable



because they have skipped ahead.

the actual of HCI

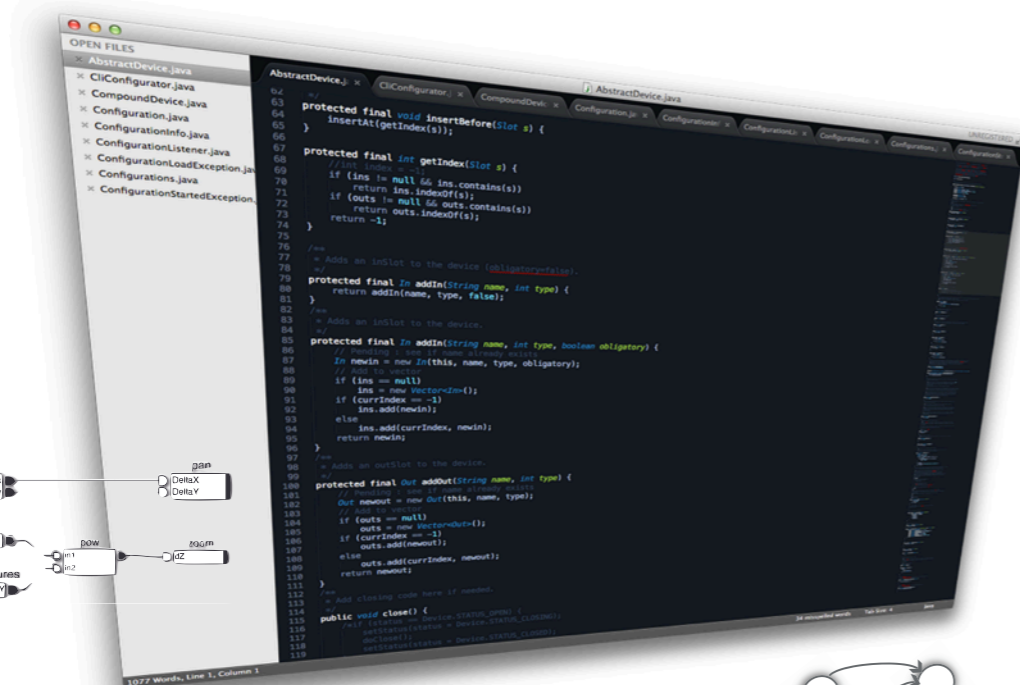


FDominec, Wikimedia Commons

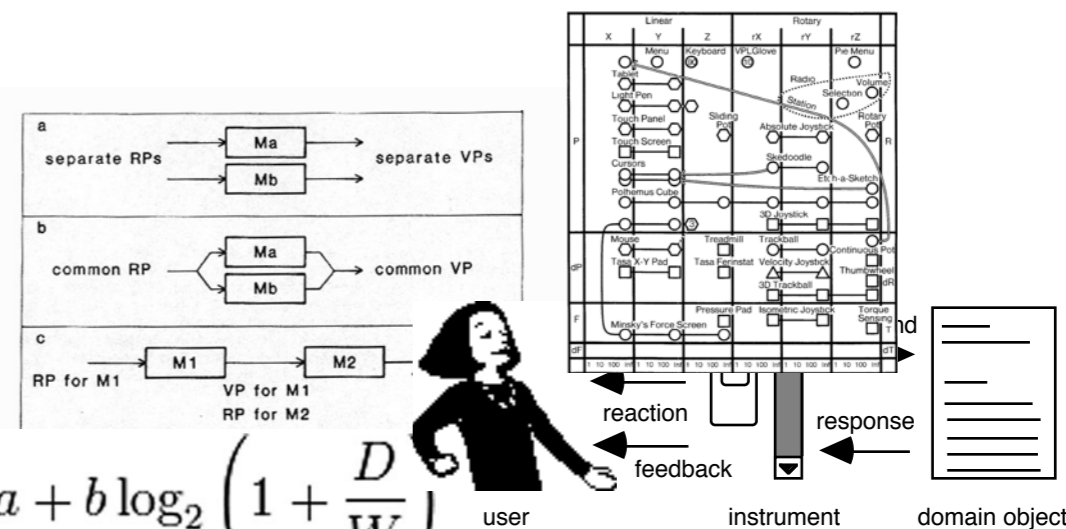
.electronics



.input/output devices



.software models, languages & toolkits



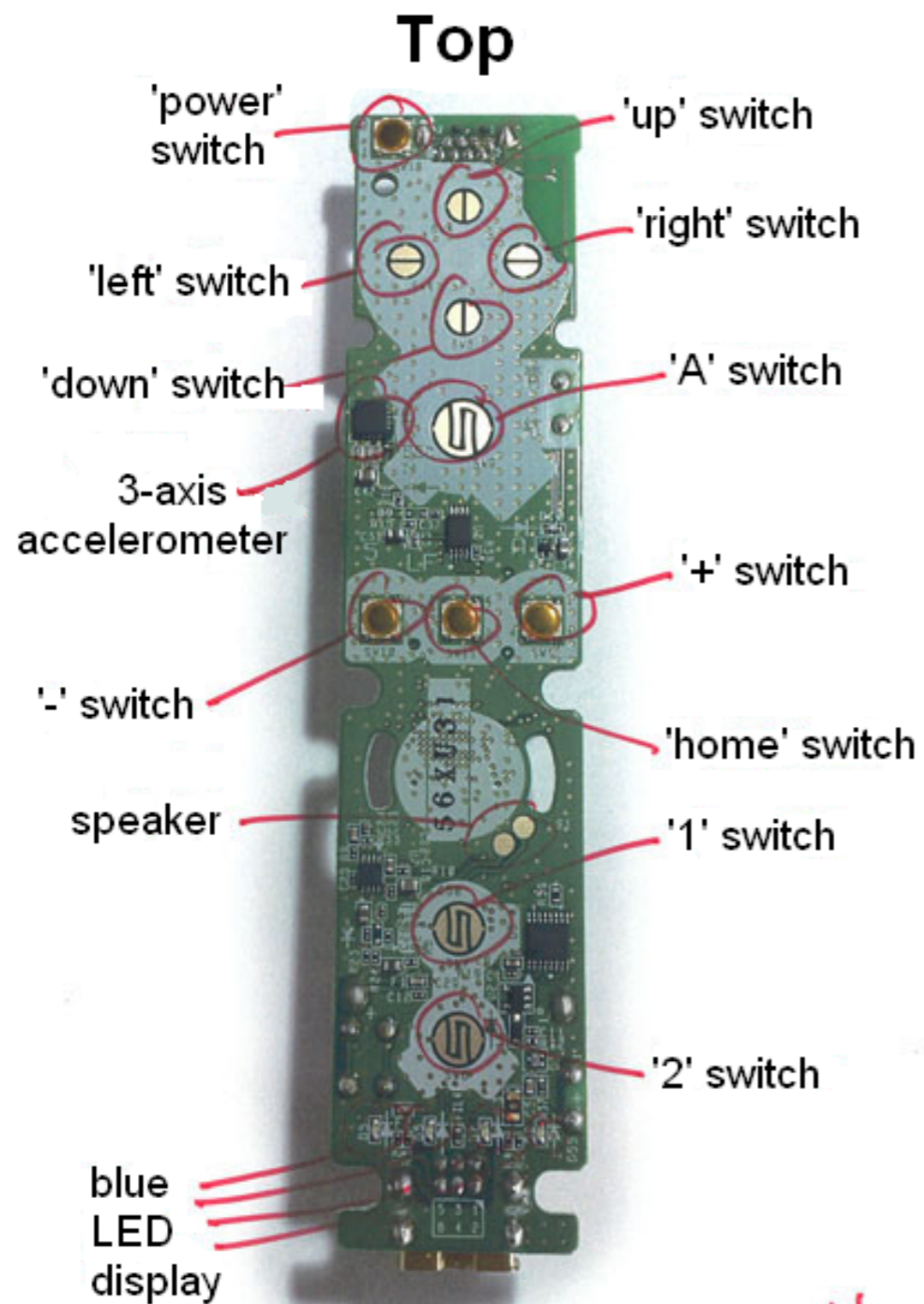
$$T = a + b \log_2 \left(1 + \frac{D}{W} \right)$$

.theories & models

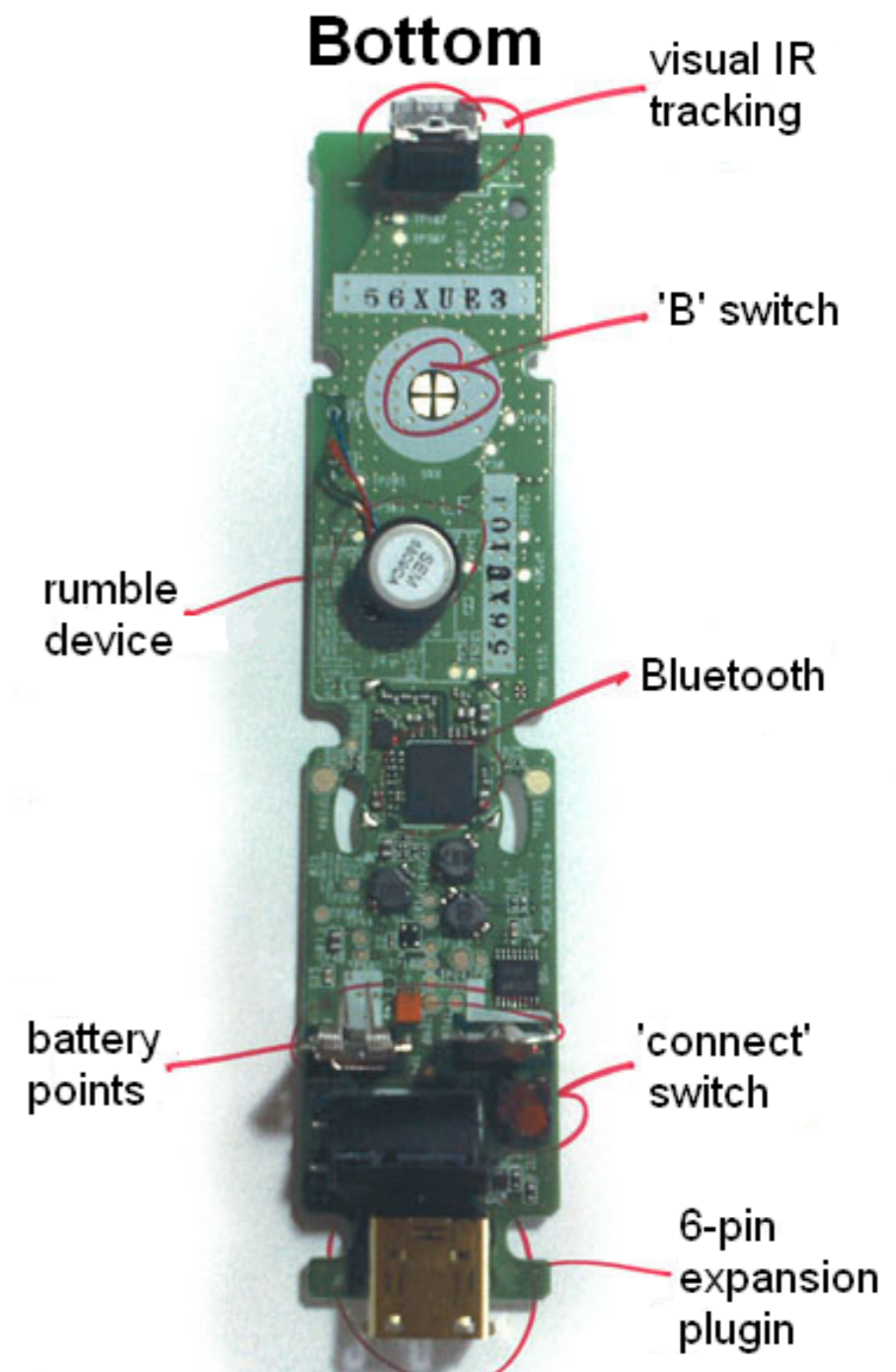
example of adjacent possible: the Wiimote



example of adjacent possible: the Wiimote



Wiimote



J. Brindza & J. Szweda, netscale.cse.nd.edu

example of adjacent possible: the Wiimote

Low-cost Multi-Point Interactive Whiteboard using the Wiimote

Johnny Chung Lee
Human-Computer Interaction Institute
Carnegie Mellon University

famous visions

.Engelbart, augmenting human intellect (60s)



monday afternoon

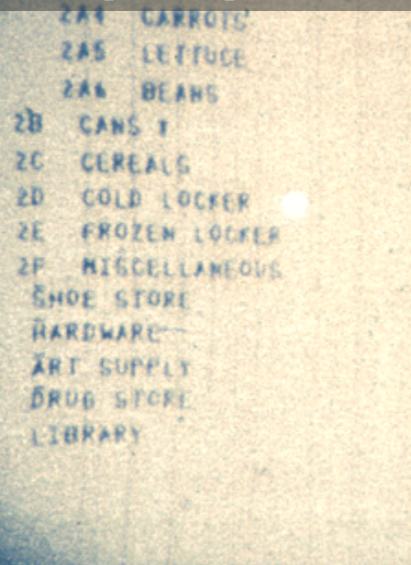
december 9

3:45 p.m. / arena

Chairman:
DR. D. C. ENGELBART
Stanford Research Institute
Menlo Park, California

a research center for augmenting human intellect

This session is entirely devoted to a presentation by Dr. Engelbart on a computer-based, interactive, multiconsole display system which is being developed at Stanford Research Institute under the sponsorship of ARPA, NASA and RADC. The system is being used as an experimental laboratory for investigating principles by which interactive computer aids can augment intellectual capability. The techniques which are being described will, themselves, be used to augment the presentation. The session will use an on-line, closed circuit television hook-up to the SRI computing system in Menlo Park. Following the presentation remote terminals to the system, in operation, may be viewed during the remainder of the conference in a special room set aside for that purpose.



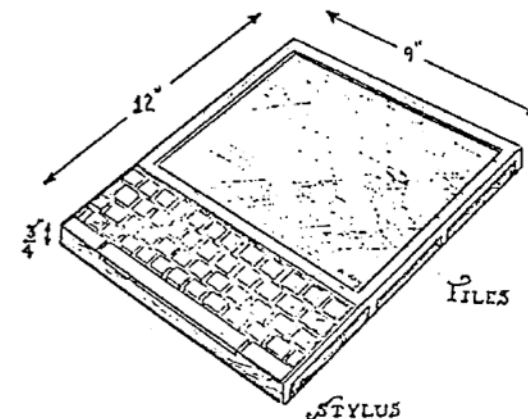
.Kay, a personal computer for children of all ages (70s)



The DynaBook

"I wish to God these calculations were executed by steam!"
Charles Babbage (age 19)
ca. 1803

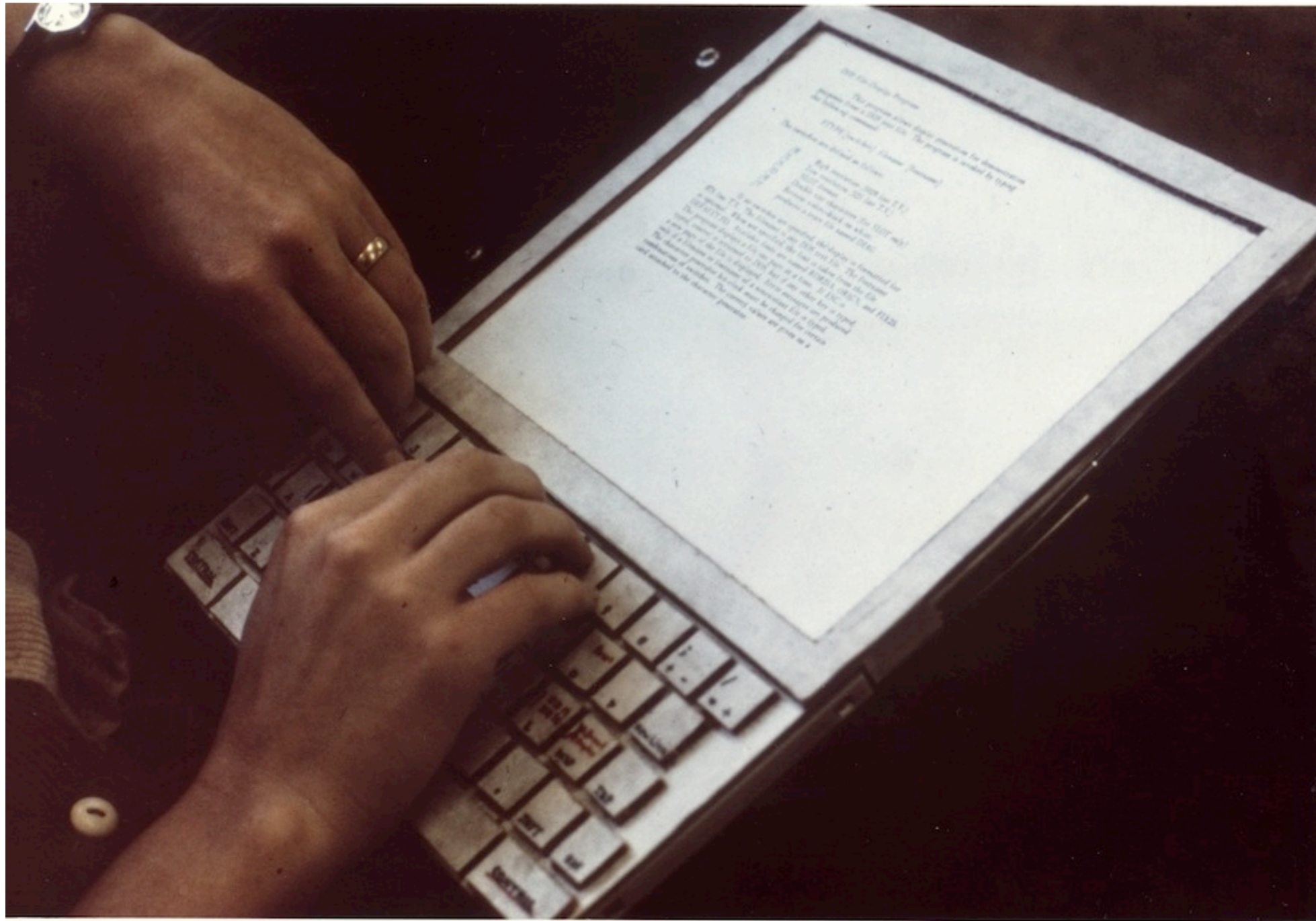
"The Analytical Engine weaves algebraic patterns, just as the Jacquard Loom weaves patterns in silk."
-Ada Augusta
Countess of Lovelace



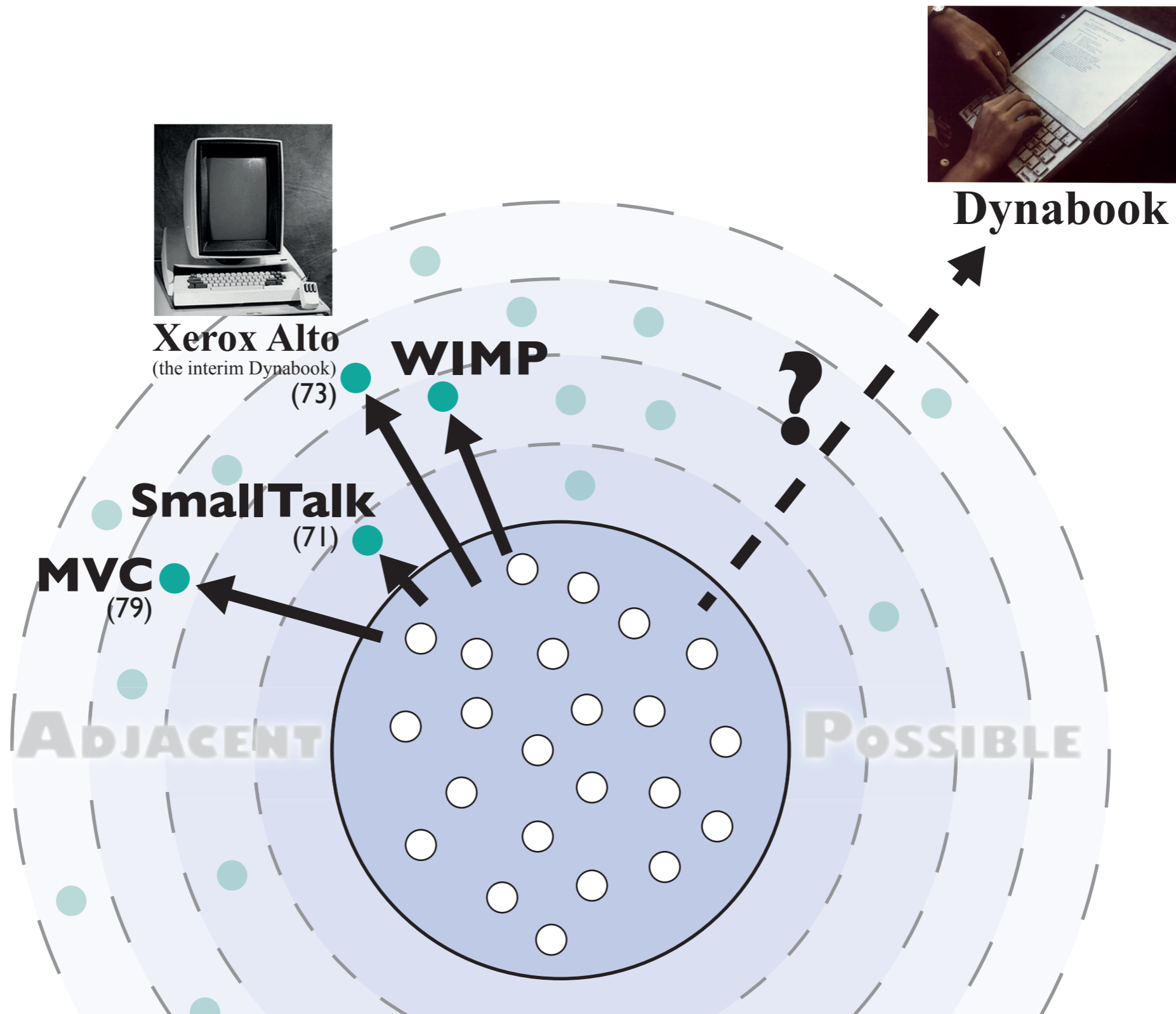
.Weiser, integrating computers seamlessly into the world (late 80s)



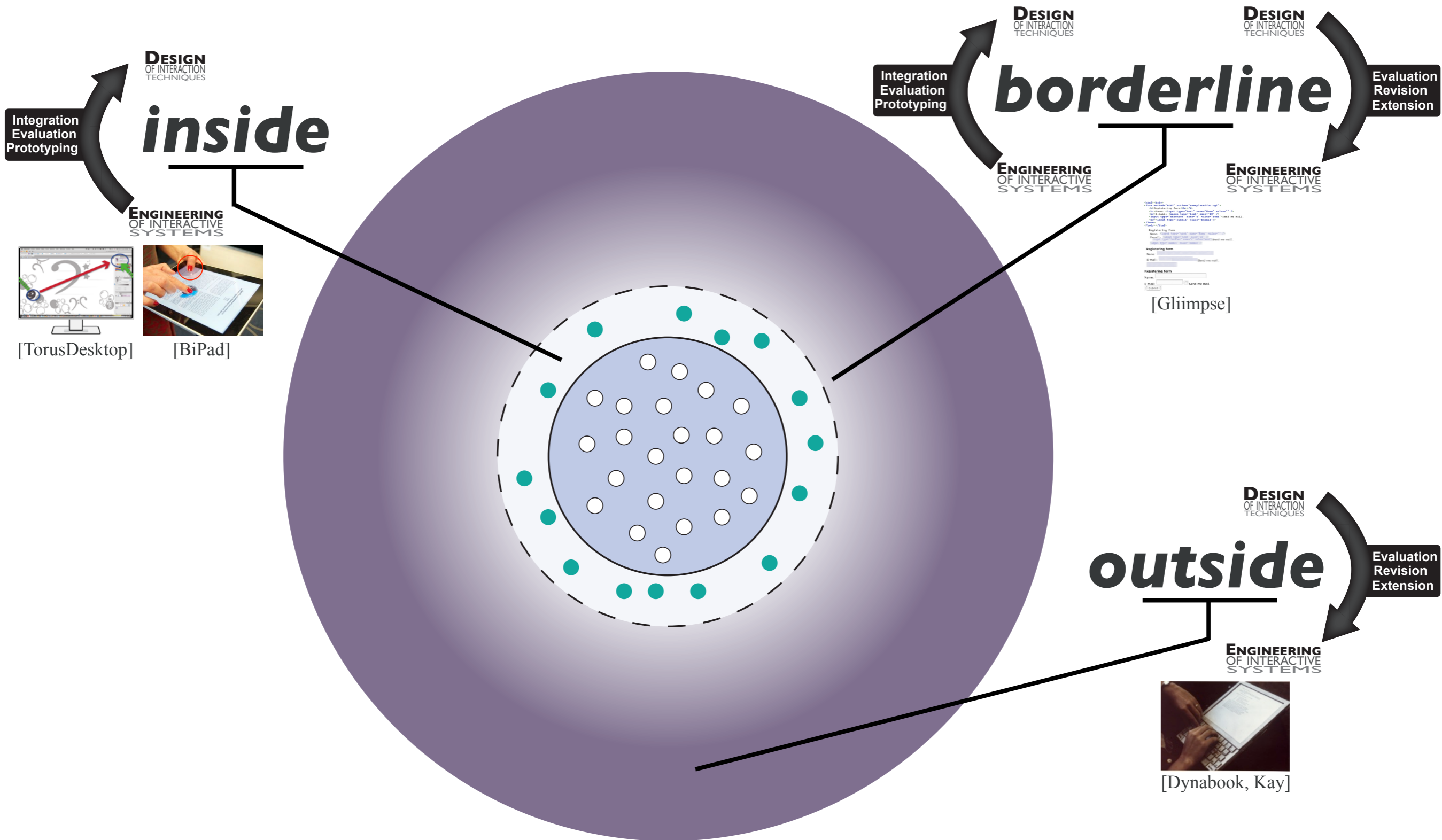
ex: Dynabook [Kay, 70s]



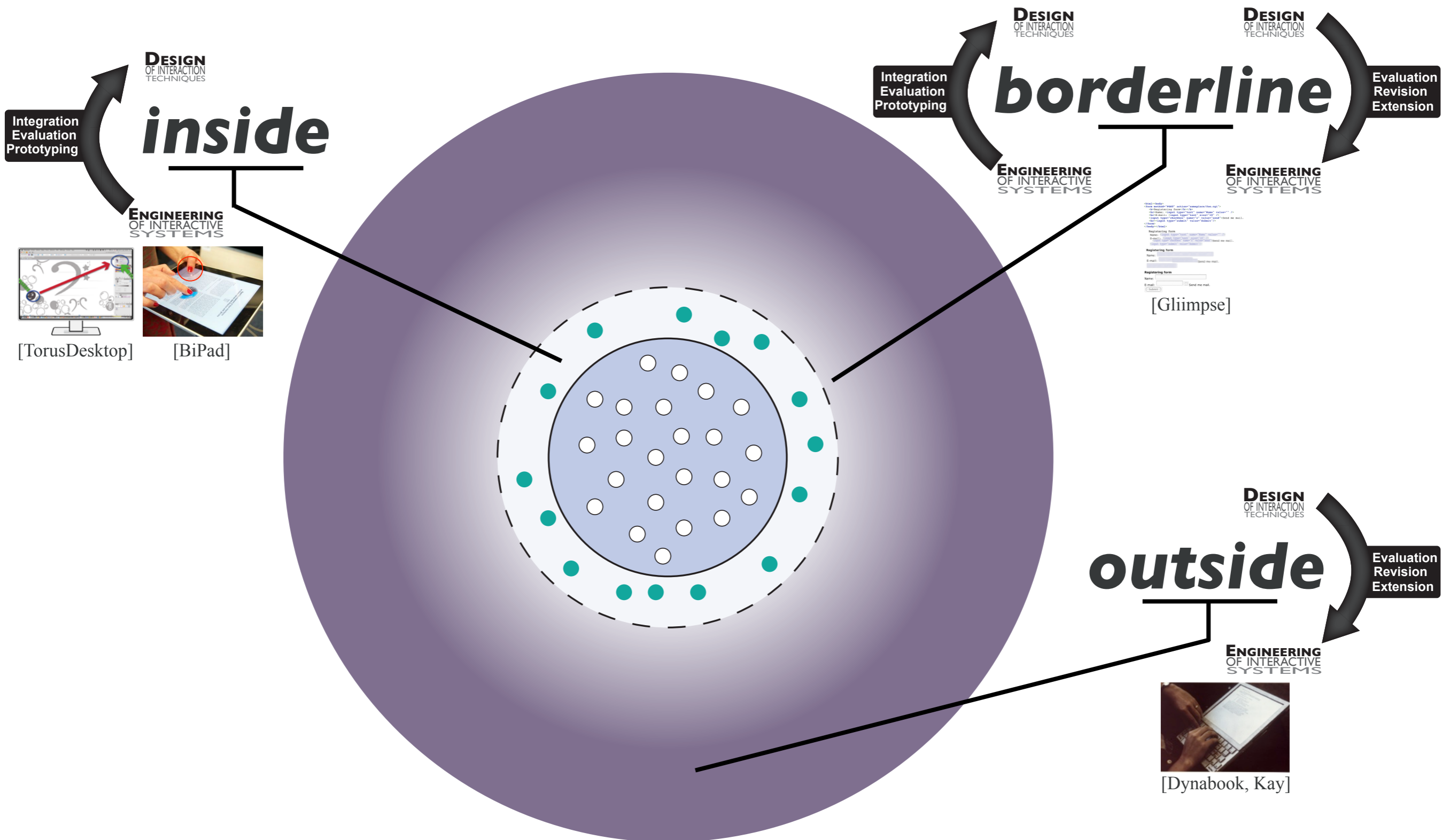
ex: Dynabook [Kay, 70s]



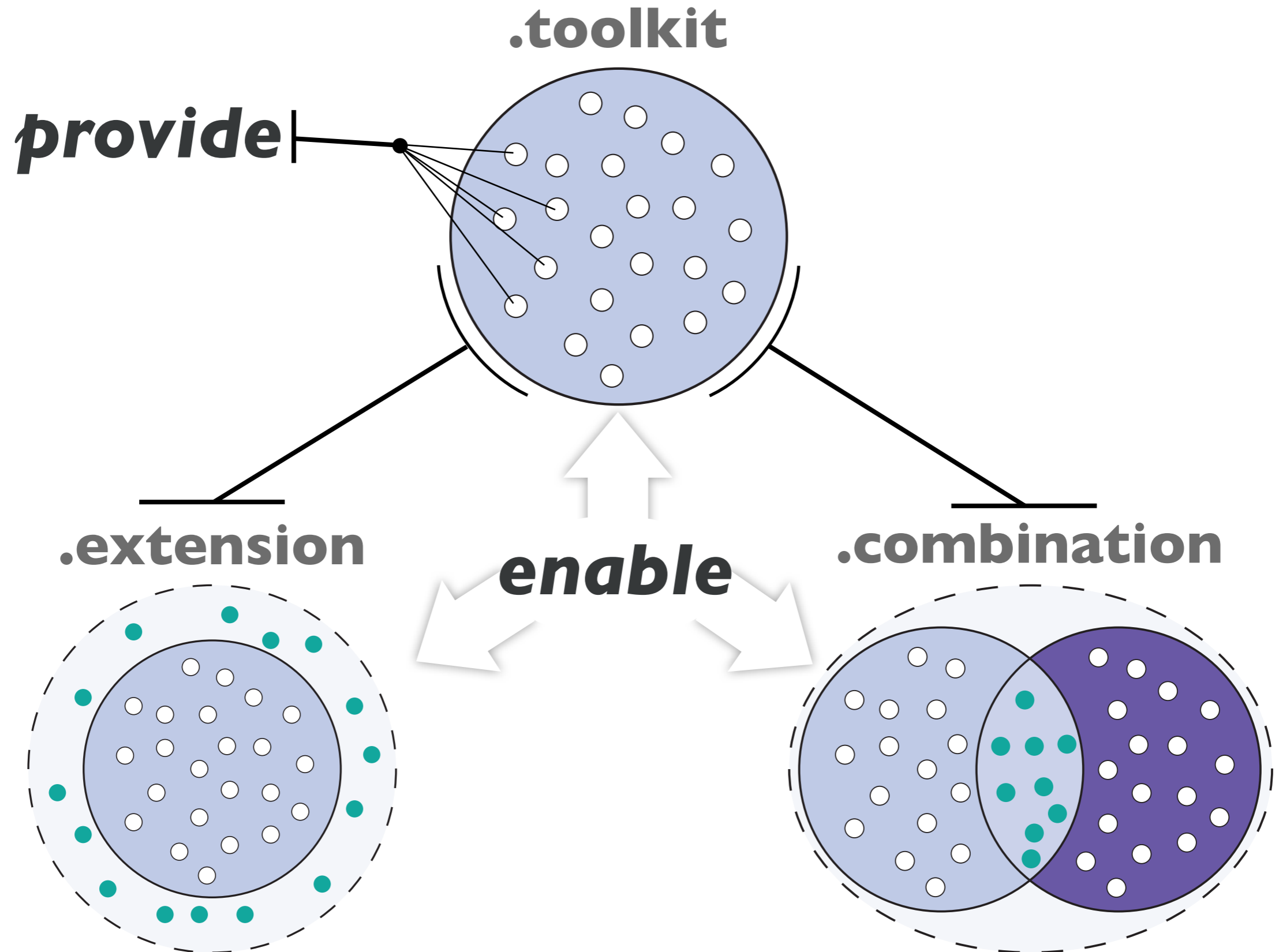
ideas and the adjacent possible



ideas and the adjacent possible

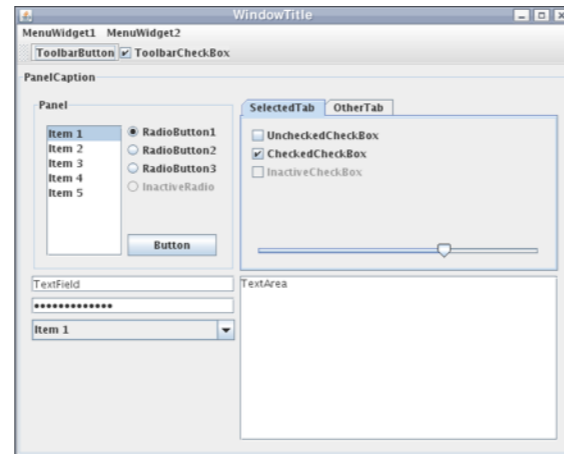


the adjacent possible of a toolkit



provide = atomic parts

WIMP

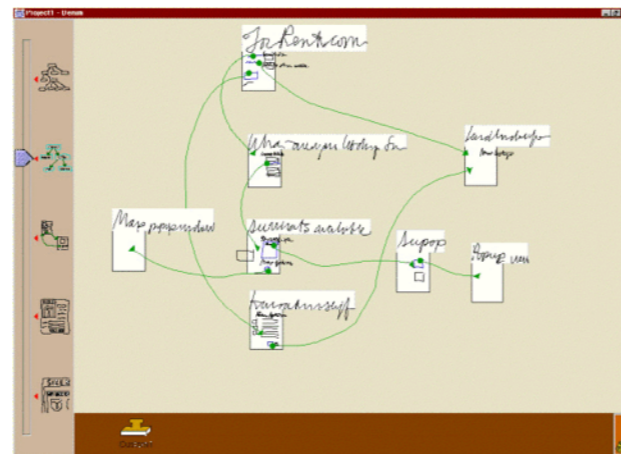


.widgets

.callbacks

SATIN

[Hong & Landay - UIST'00]



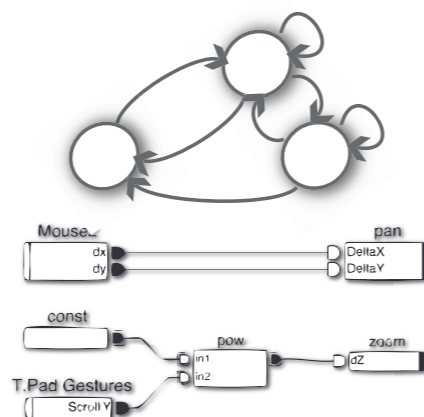
.strokes interpreters

.gestures recognizers

.strokes

.gestures

FlowStates



.state machines

.data-flow devices

.abstract events

enable = extension, combination, reuse and interoperability

extension

.inheritance

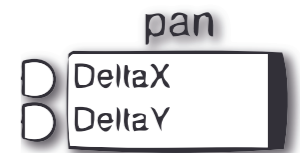
Ich bin das JMenuItem



Ich bin das JCheckBoxMenuItem

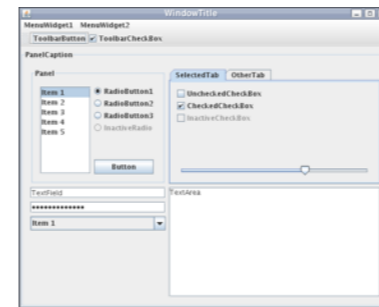
combination

.creation



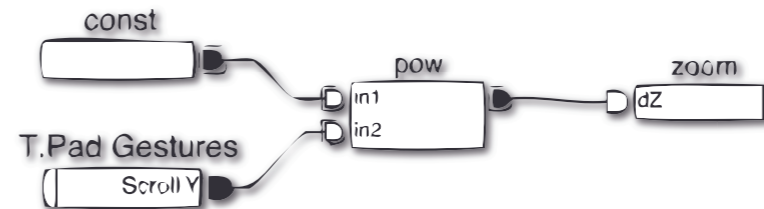
reuse

.structural



interoperability

.logical



.extensions

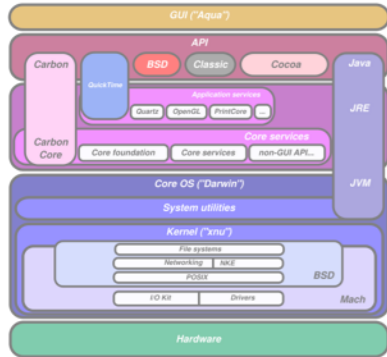
.combinations

.internal

.external

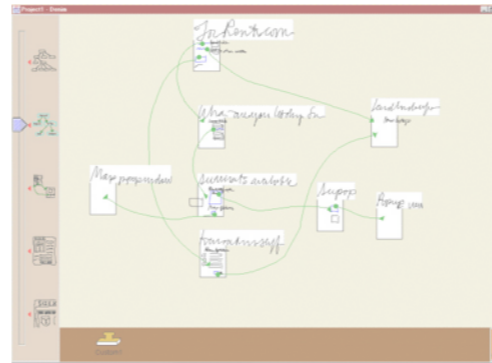
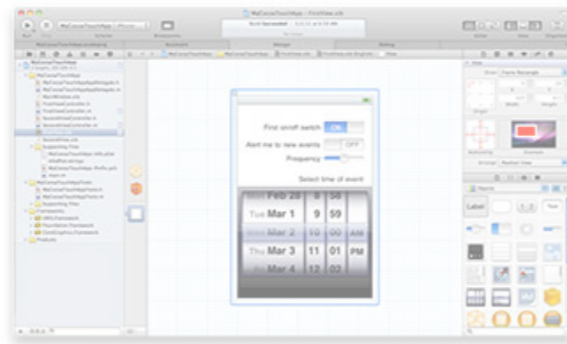
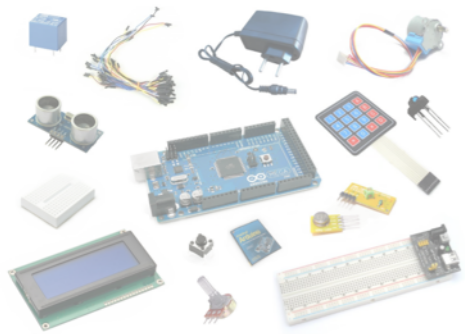
perspectives: tools for designeering interaction

.system and programming languages

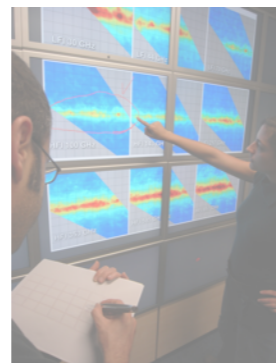
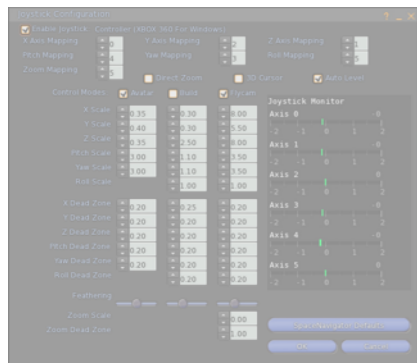


```
File Edit Run Compile Options Debug Break/watch
Line 15 Col 39 Insert Indent Unindent * 0:NONAME.PRS
program KenLovesTurboPascal;
uses
  crt;
var
  age: Integer;
  name: String;
  message: String;
begin
  ClrScr;
  name := 'Ken Egozi';
  age := 30;
  if age < 10 then
    message := ' loves Turbo Pascal'
  else
    message := ' loved Turbo Pascal';
  write (name);
  writeln (message);
end.
```

.creative prototyping: sketching interaction, not interfaces

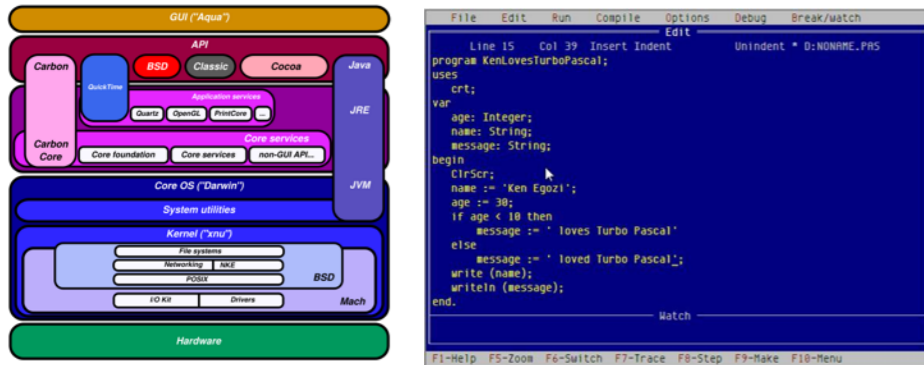


.adaptability for end-users

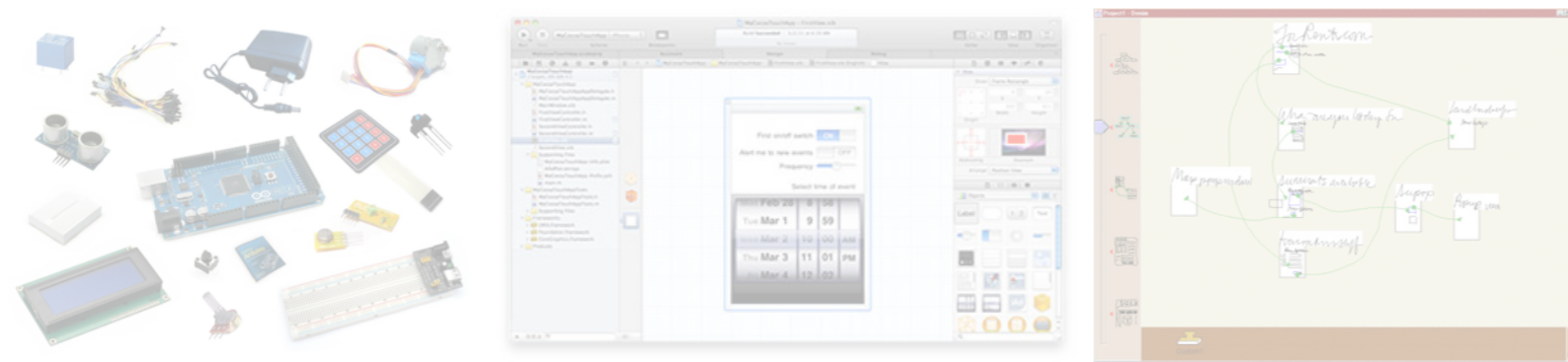


perspectives: tools for designeering interaction

.system and programming languages



.creative prototyping: sketching interaction, not interfaces



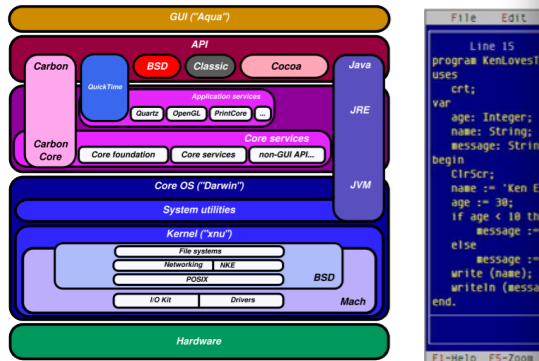
.adaptability for end-users



perspectives: tools for design engineering interaction

system & programming languages

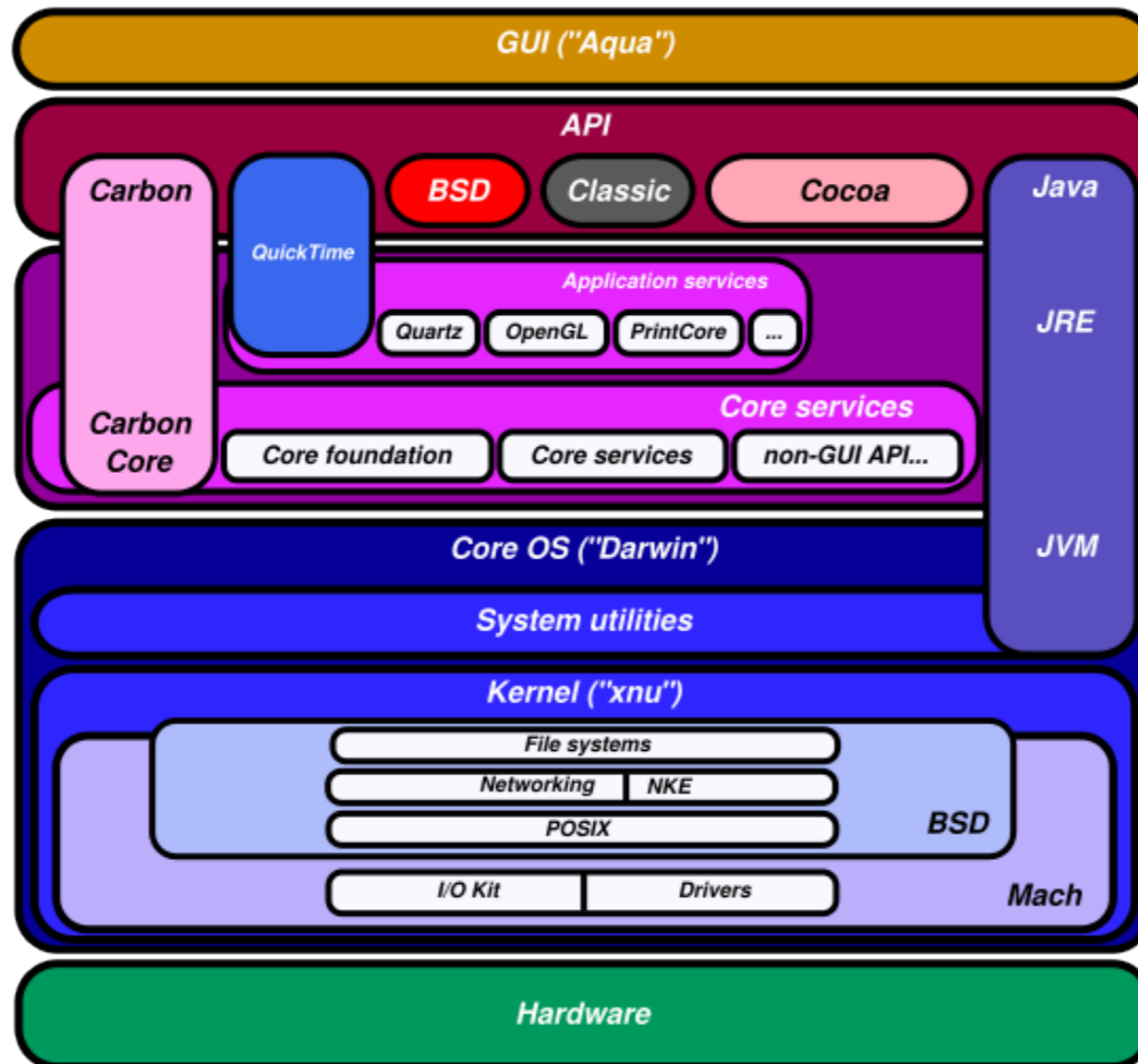
.system and prog



.creative prototy



.adaptability for

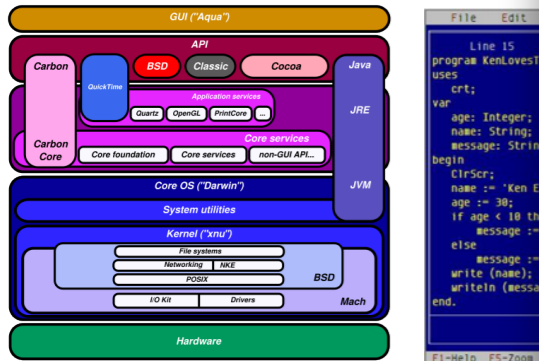


.low-level language/libraries

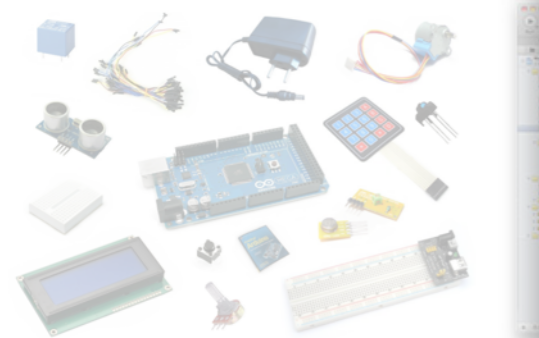
↳ combination and interoperability
unifying the levels

system & programming languages

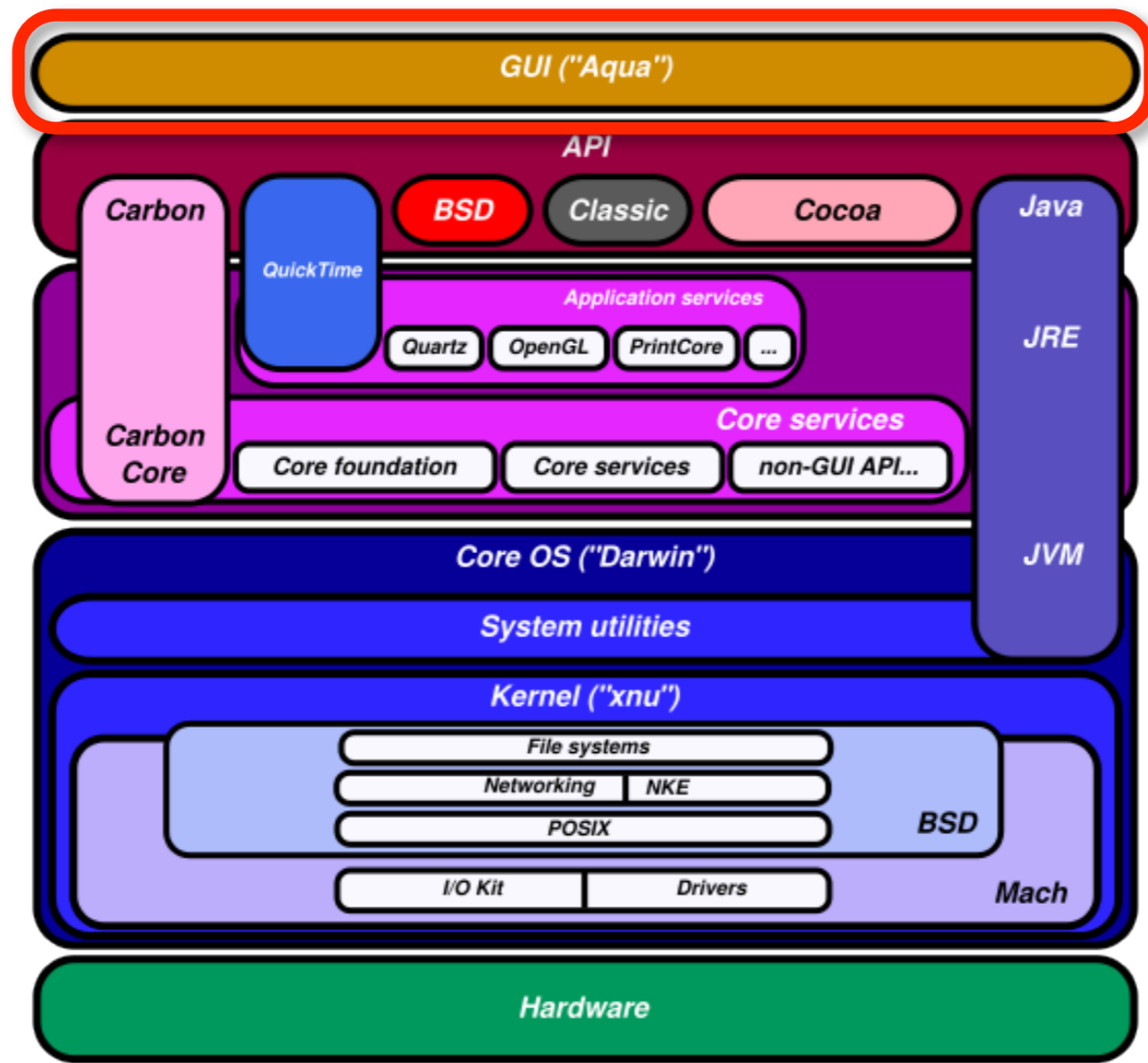
.system and prog



.creative prototy



.adaptability for

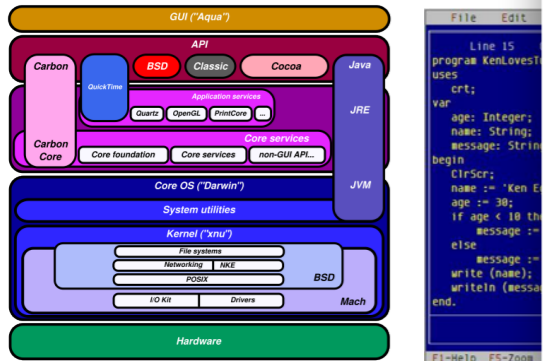


.low-level language/libraries

↳ combination and interoperability
unifying the levels

system & programming languages

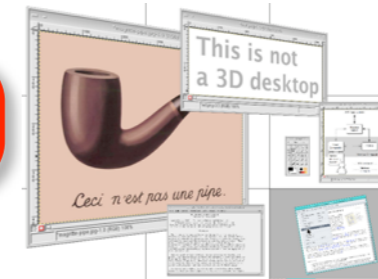
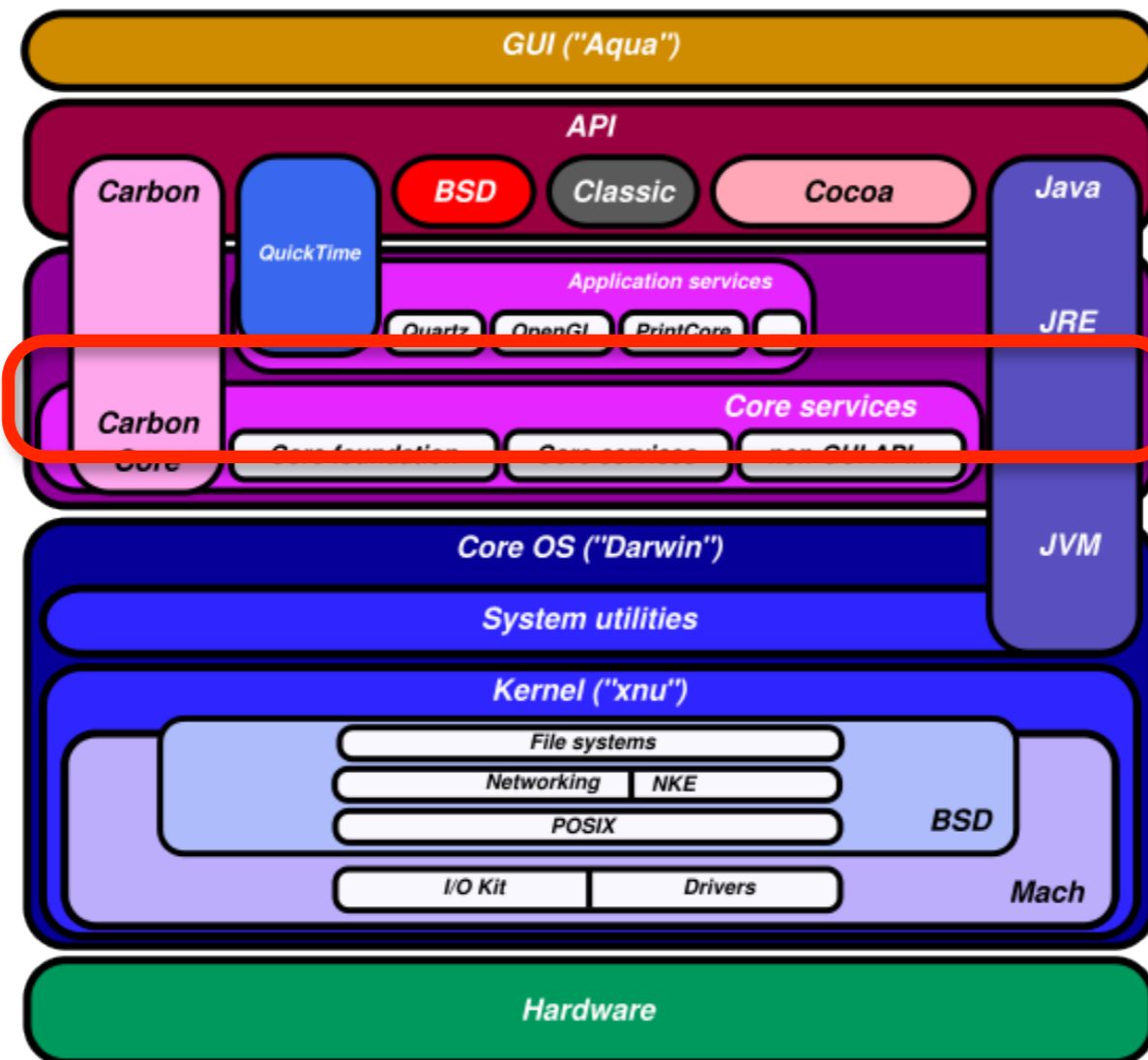
.system and prog



.creative prototy



.adaptability for



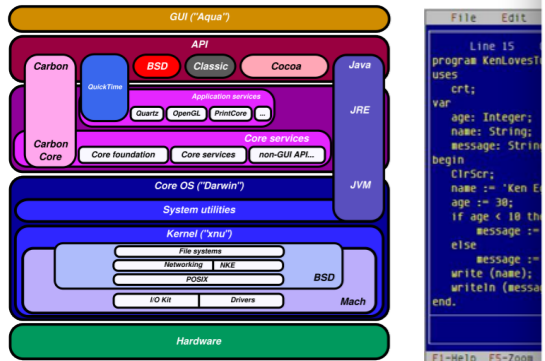
[Metisse - Chapuis & Roussel, UIST'05]

.low-level language/libraries

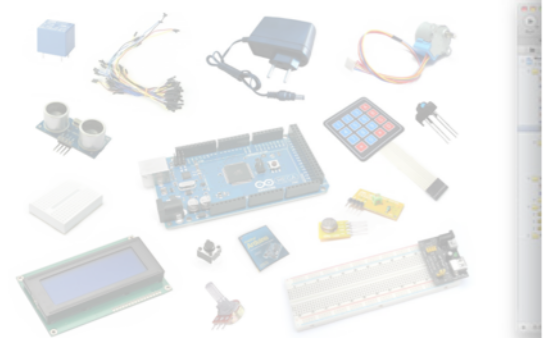
↳ combination and interoperability
unifying the levels

system & programming languages

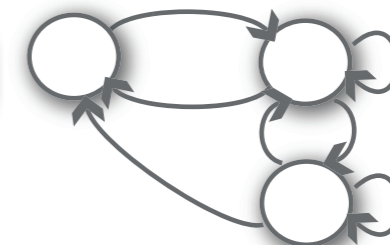
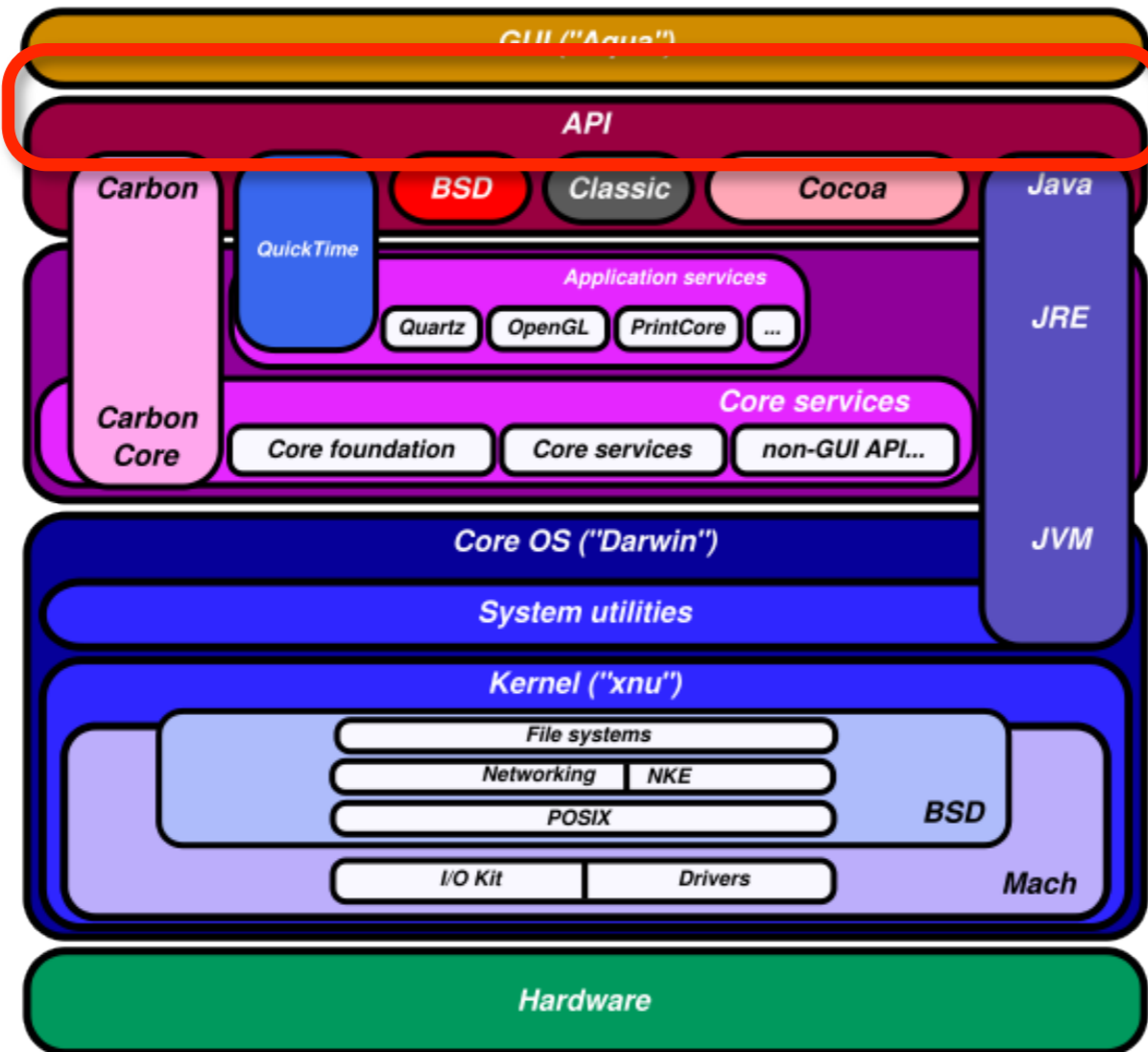
.system and prog



.creative prototy



.adaptability for



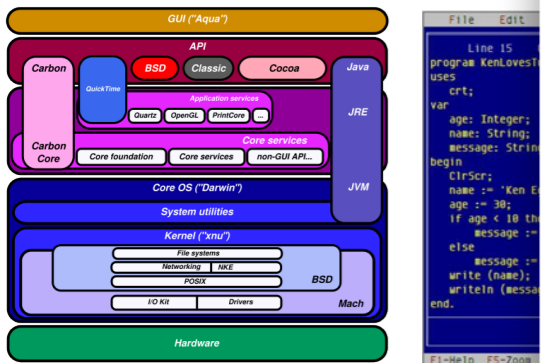
[SwingStates]

.low-level language/libraries

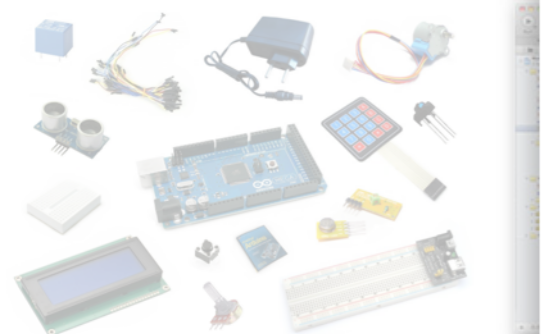
↳ combination and interoperability
unifying the levels

system & programming languages

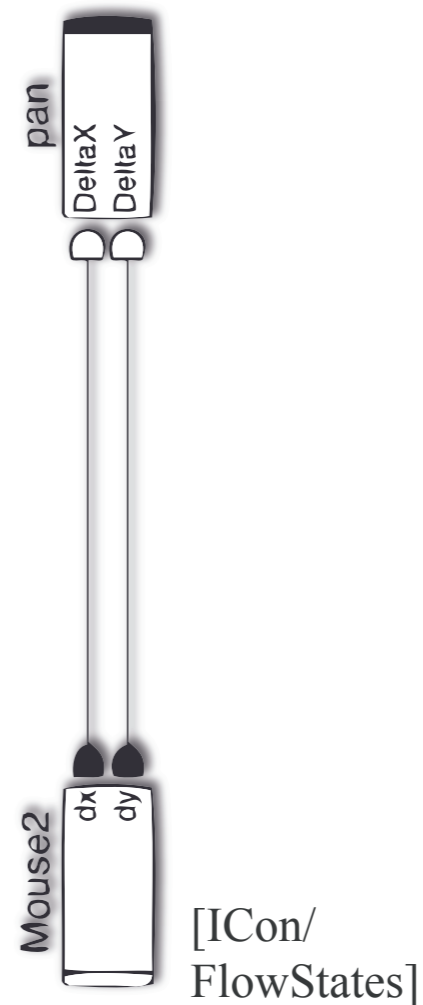
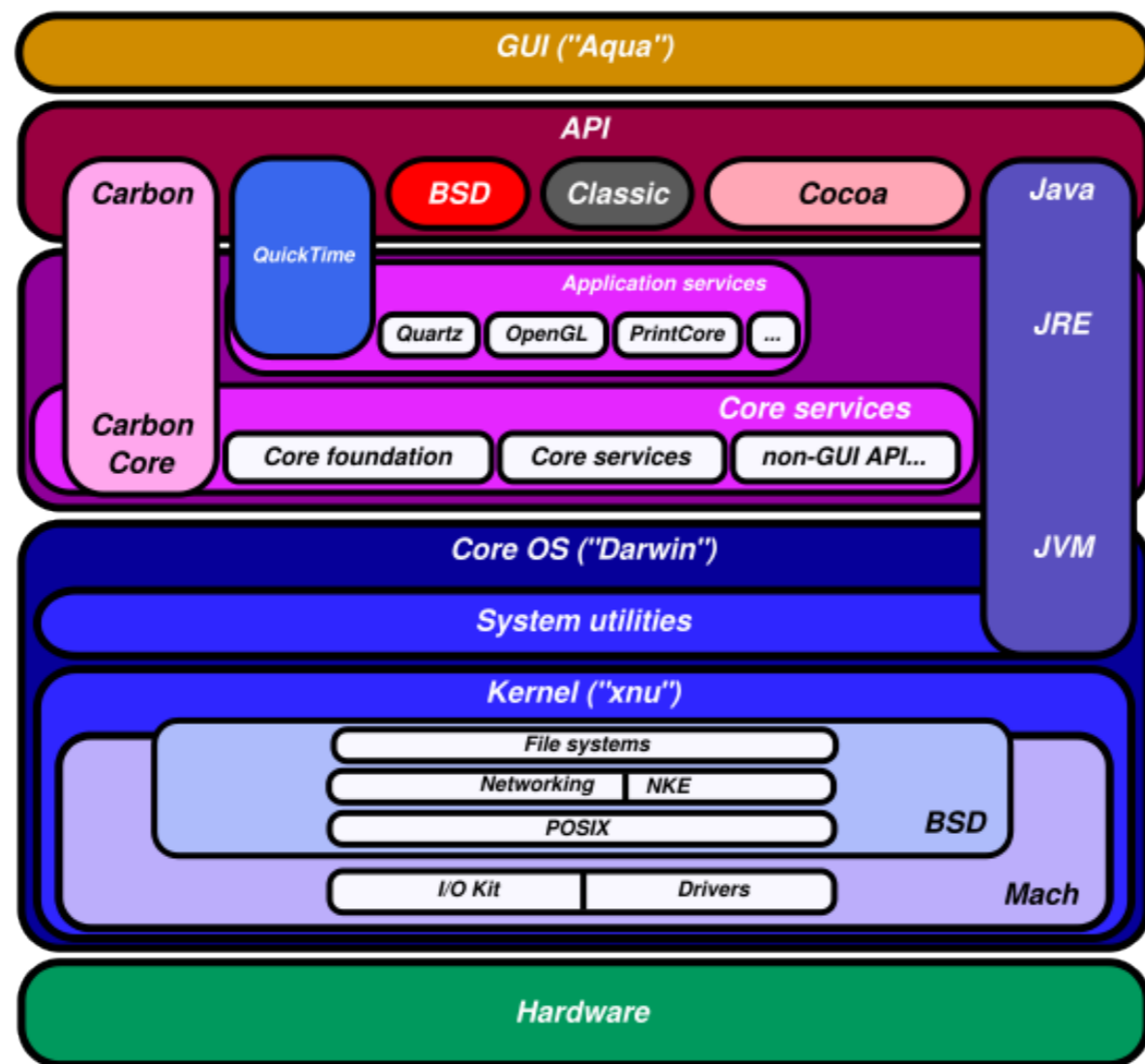
.system and prog



.creative prototy



.adaptability for



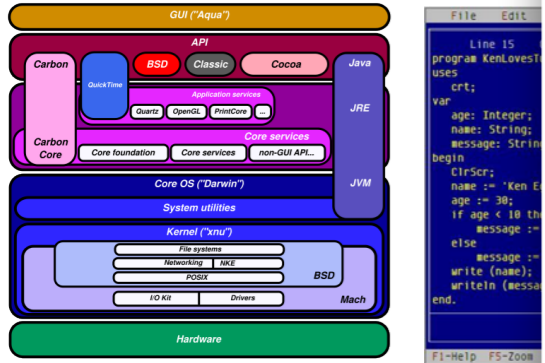
[ICon/
FlowStates]

.low-level language/libraries

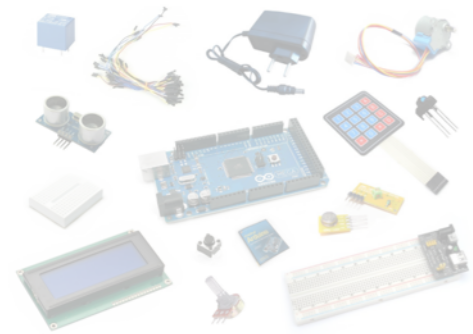
↳ combination and interoperability
unifying the levels

system & programming languages

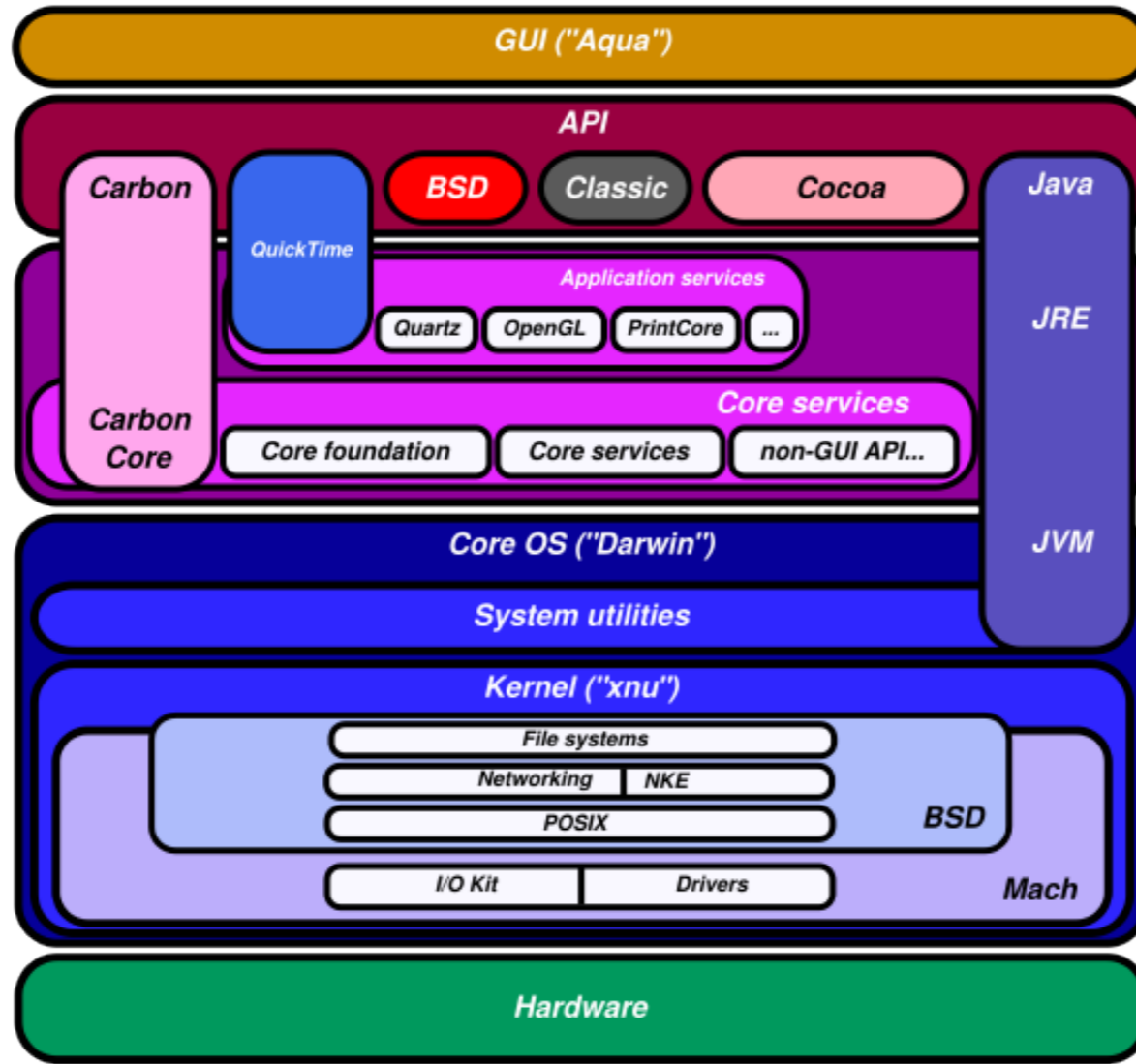
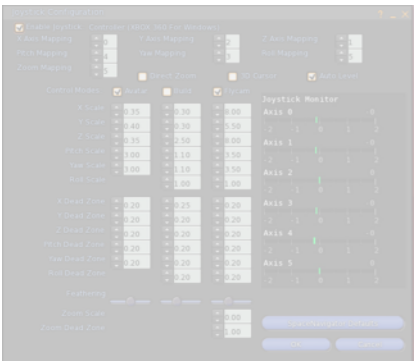
.system and prog



.creative prototy



.adaptability for



```
Welcome to the Distel Interactive Erlang Shell.
% C-j evaluates an expression and prints the result in-line.
% C-M-x evaluates a whole function definition.

test1:area({square,10}).
--> 100

TempConvert = fun({c,C}) ->
  {f, 32+(C*9/5);
  ({f,F}) ->
  {c,(F-32)*5/9}
end.
--> #Fun<erl_eval.6.13229925>

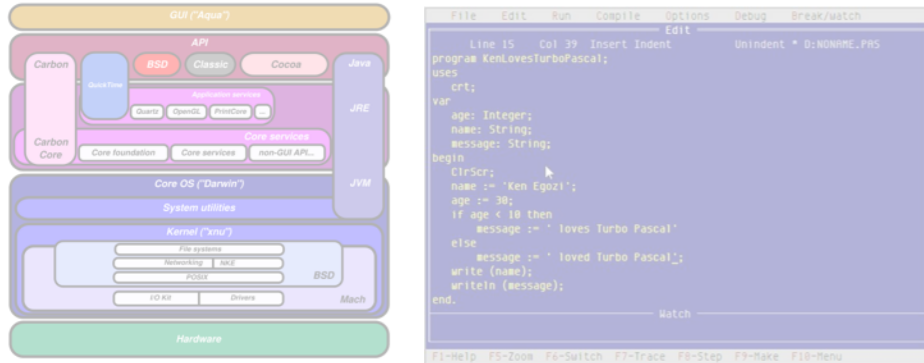
TempConvert({f,100}).
--> {c,37.77777777777778}
```

.low-level language/libraries

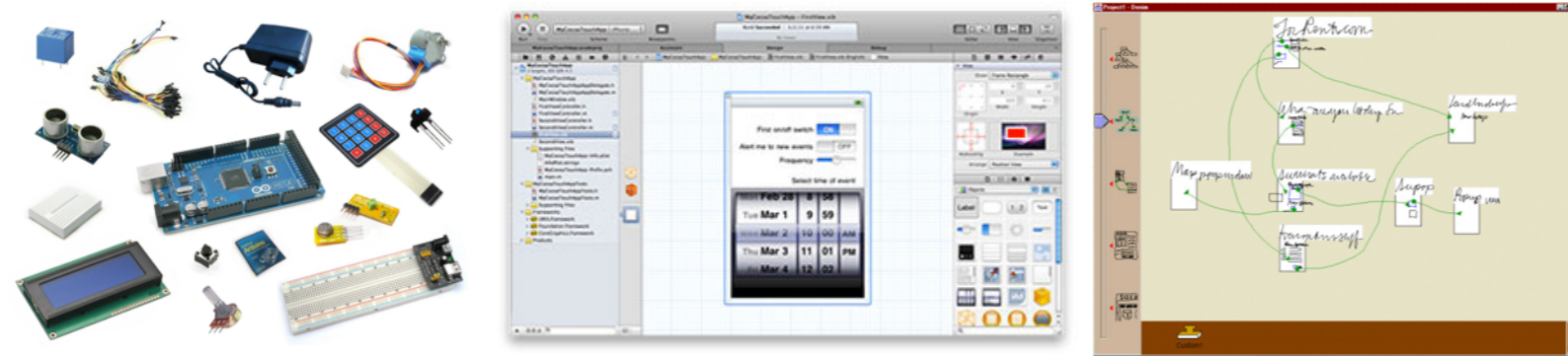
↳ combination and interoperability
unifying the levels

perspectives: tools for designeering interaction

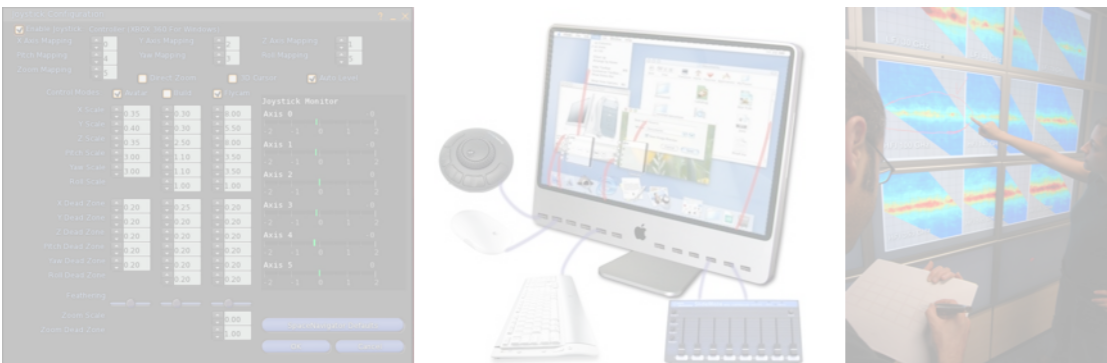
.system and programming languages



.creative prototyping: sketching interaction, not interfaces



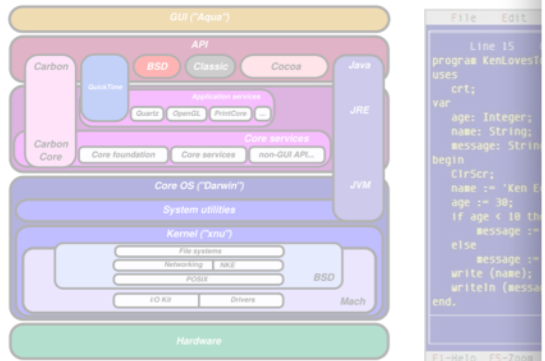
.adaptability for end-users



perspectives: tools for designing interaction

sketching interaction, not interfaces

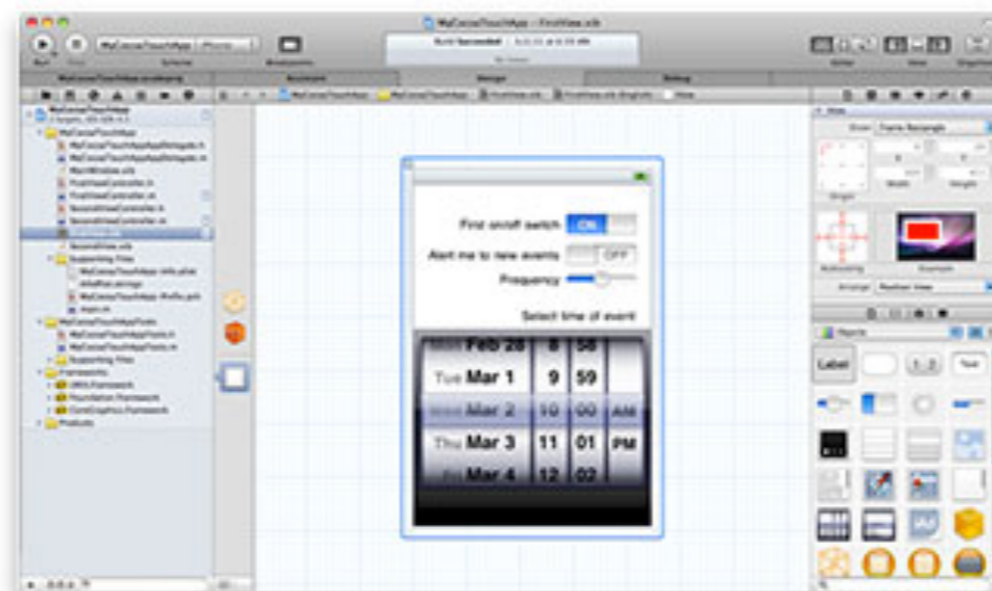
.system and prog



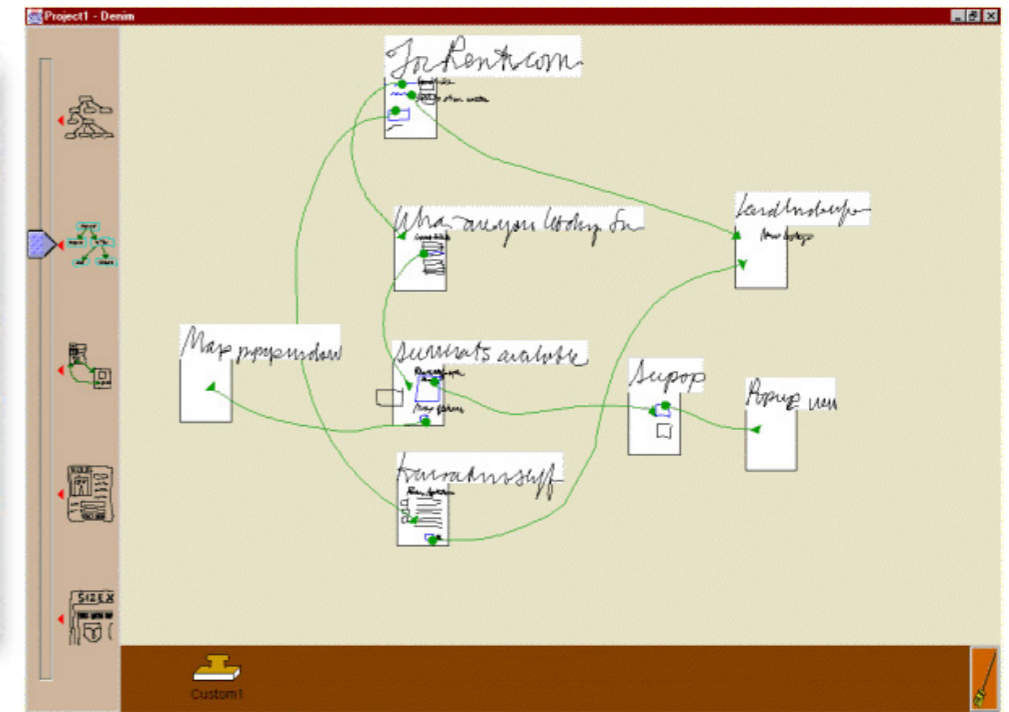
.creative prototy



.adaptability for



[Apple XCode]



[Denim - Lin et al, CHI'00]

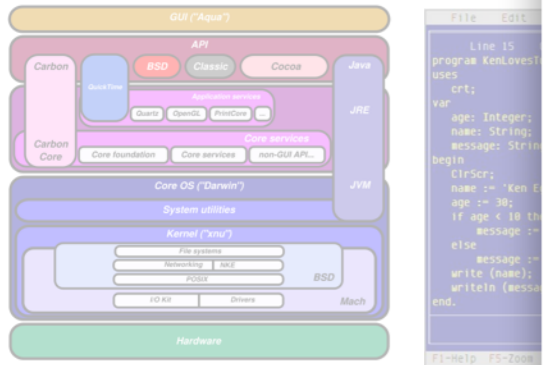
.rapid and iterative design

↳ structural **and logical** combinations
unifying the levels of prototyping

perspectives: tools for design engineering interaction

sketching interaction, not interfaces

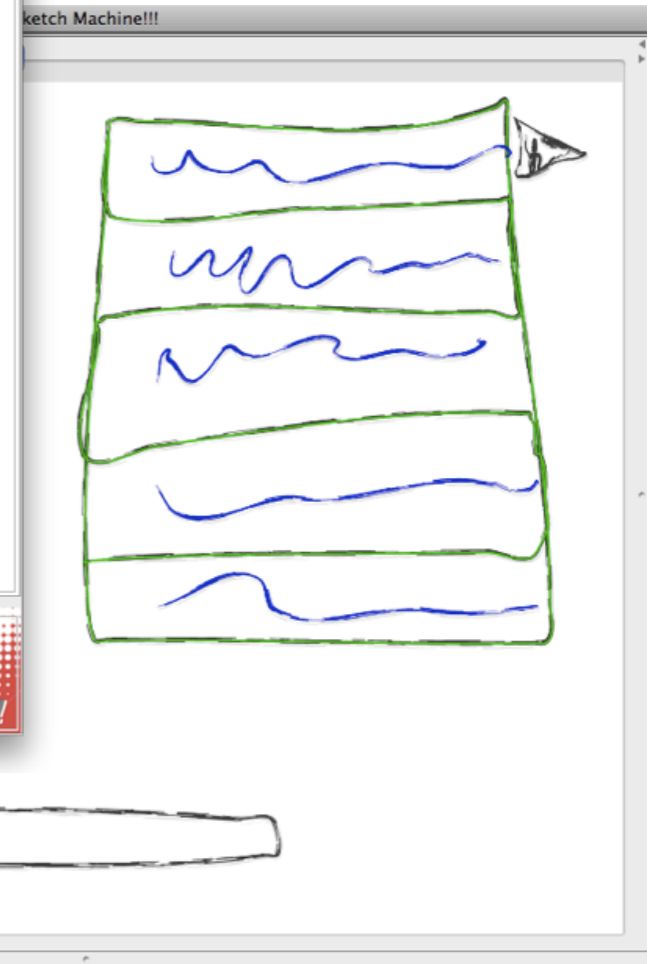
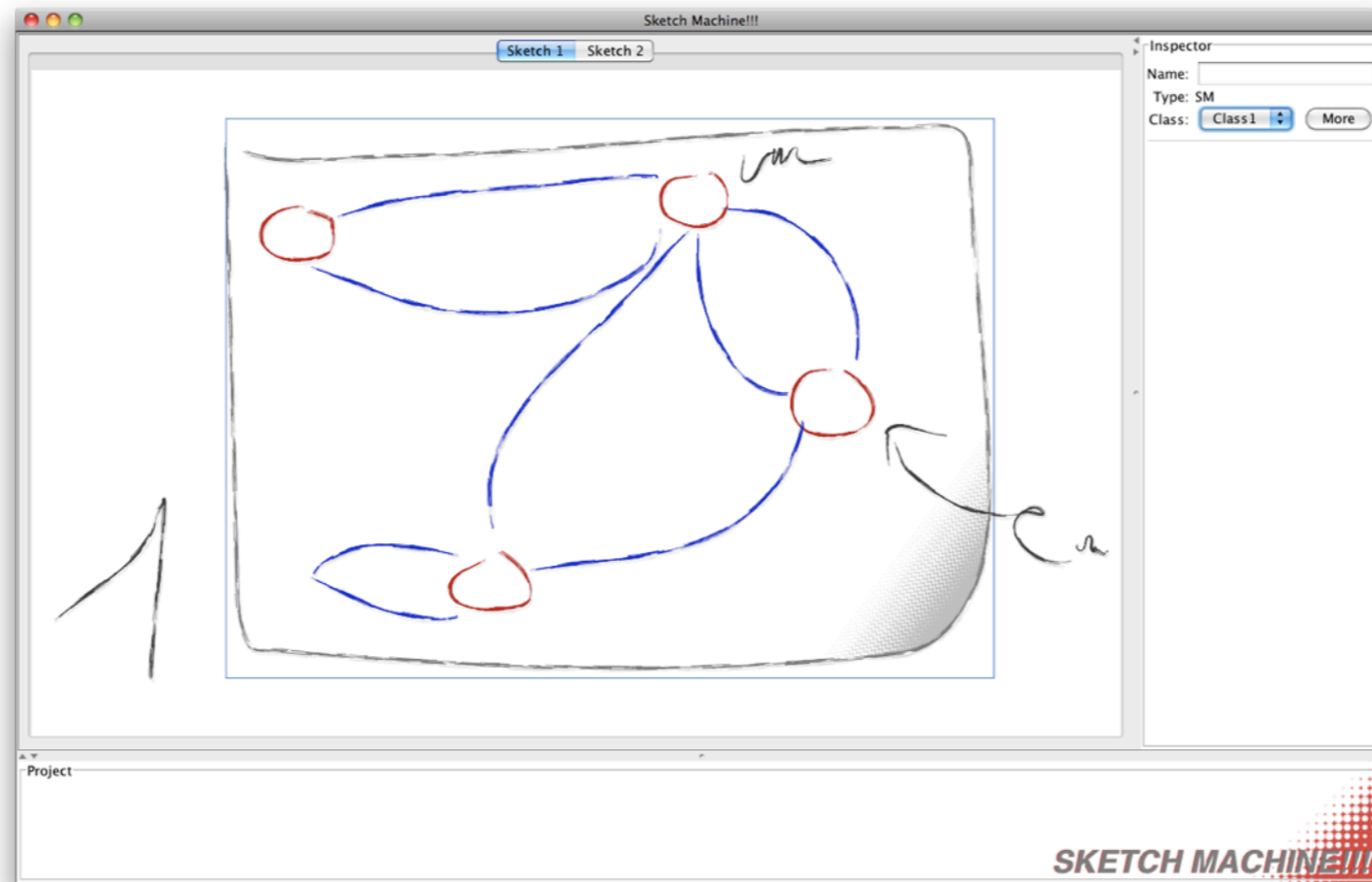
.system and prog



.creative prototy



.adaptability for

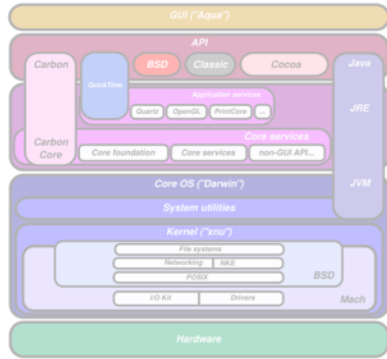


.rapid and iterative design

↳ structural **and logical** combinations
unifying the levels of prototyping

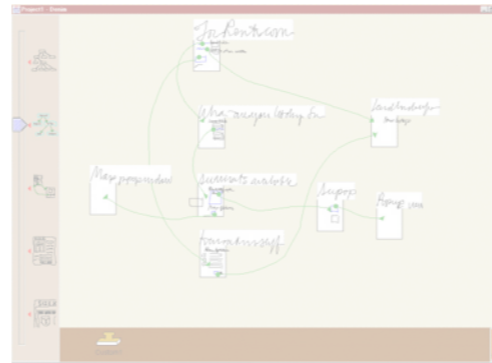
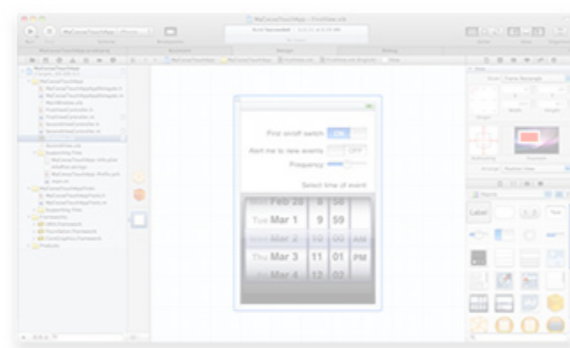
perspectives: tools for designeering interaction

.system and programming languages

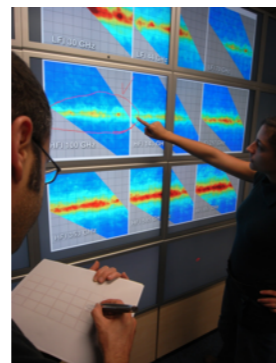
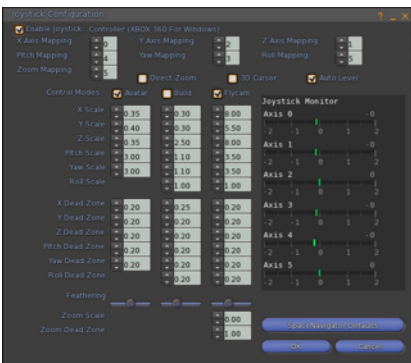


```
File Edit Run Compile Options Debug Break/watch
Edit
Line 15 Col 39 Insert Indent Unindent * 0:NONAME.PAS
program KenLovesTurboPascal;
uses
  crt;
var
  age: Integer;
  name: String;
  message: String;
begin
  ClrScr;
  name := 'Ken Egozi';
  age := 30;
  if age < 10 then
    message := ' loves Turbo Pascal'
  else
    message := ' loved Turbo Pascal';
  write (name);
  writeln (message);
end.
```

.creative prototyping: sketching interaction, not interfaces

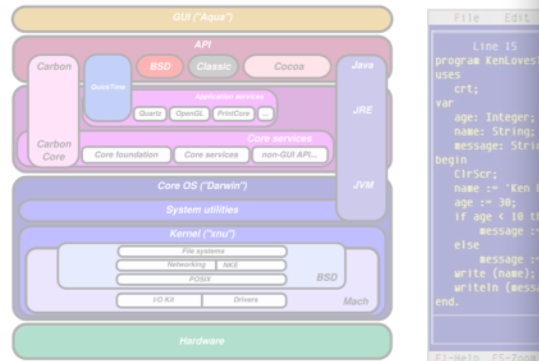


.adaptability for end-users

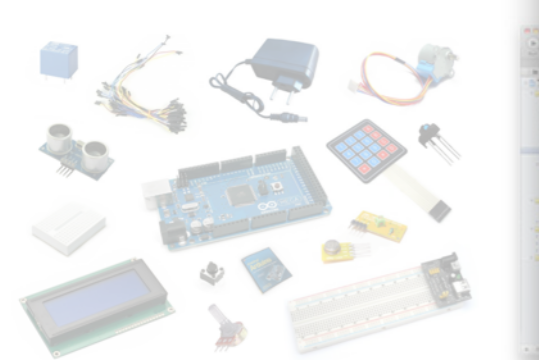


adaptability for end-users

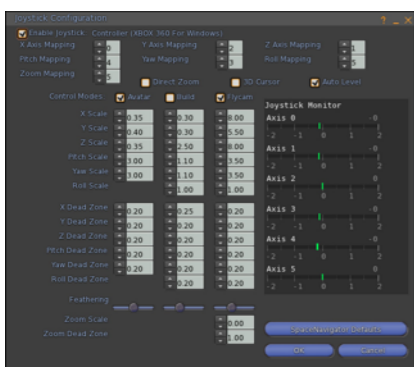
.system and prog



.creative prototy



.adaptability for



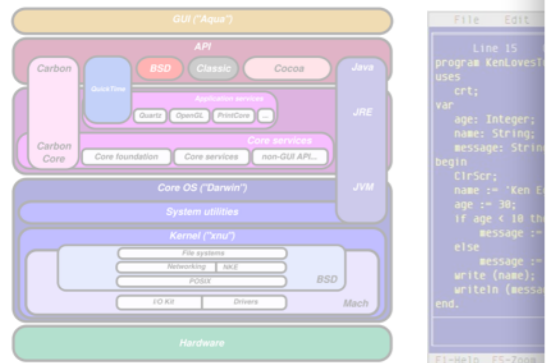
.interaction as a first-class object

↳ interaction manipulation

interaction transformation

adaptability for end-users

.system and prog



.creative prototy



.adaptability for



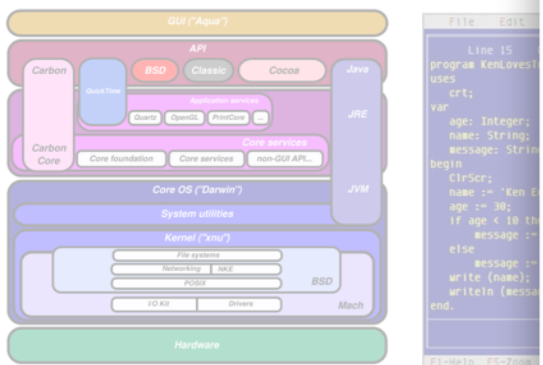
.interaction as a first-class object

↳ interaction manipulation

interaction transformation

adaptability for end-users

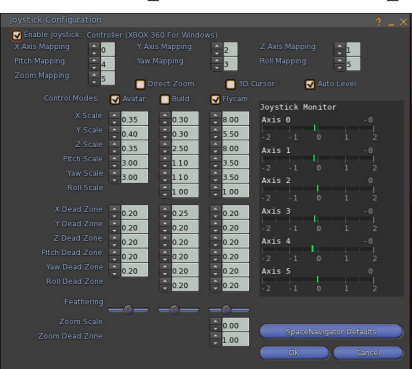
.system and prog



.creative prototy



.adaptability for



The main screenshot of the Joystick Configuration window. It features a dark theme and includes the following sections:

- Enable Joystick:** Controller (XBOX 360 For Windows) - checked.
- Axis Mapping:** X Axis Mapping (0), Y Axis Mapping (2), Z Axis Mapping (1), Pitch Mapping (4), Yaw Mapping (3), Roll Mapping (5), Zoom Mapping (5).
- Options:** Direct Zoom (unchecked), 3D Cursor (unchecked), Auto Level (checked).
- Control Modes:** Avatar (checked), Build (unchecked), Flycam (checked).
- Scales:** X Scale (0.35, 0.30, 8.00), Y Scale (0.40, 0.30, 5.50), Z Scale (0.35, 2.50, 8.00), Pitch Scale (3.00, 1.10, 3.50), Yaw Scale (3.00, 1.10, 3.50), Roll Scale (1.00, 1.00, 1.00).
- Dead Zones:** X, Y, Z, Pitch, Yaw, Roll (all set to 0.20).
- Feathering:** Three sliders for X, Y, and Z axes.
- Zoom Scale:** 0.00
- Zoom Dead Zone:** 1.00
- Joystick Monitor:** Six axes (0-5) with numerical values and visual indicators.
- Buttons:** SpaceNavigator Defaults, OK, Cancel.

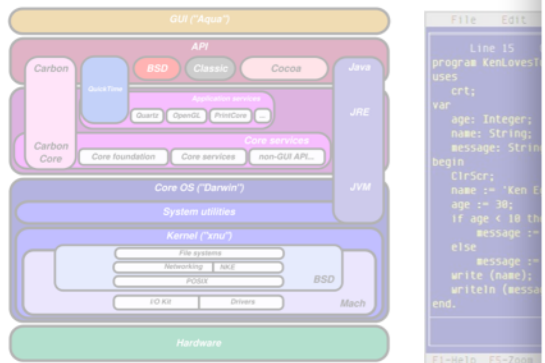
.interaction as a first-class object

↳ interaction manipulation

interaction transformation

adaptability for end-users

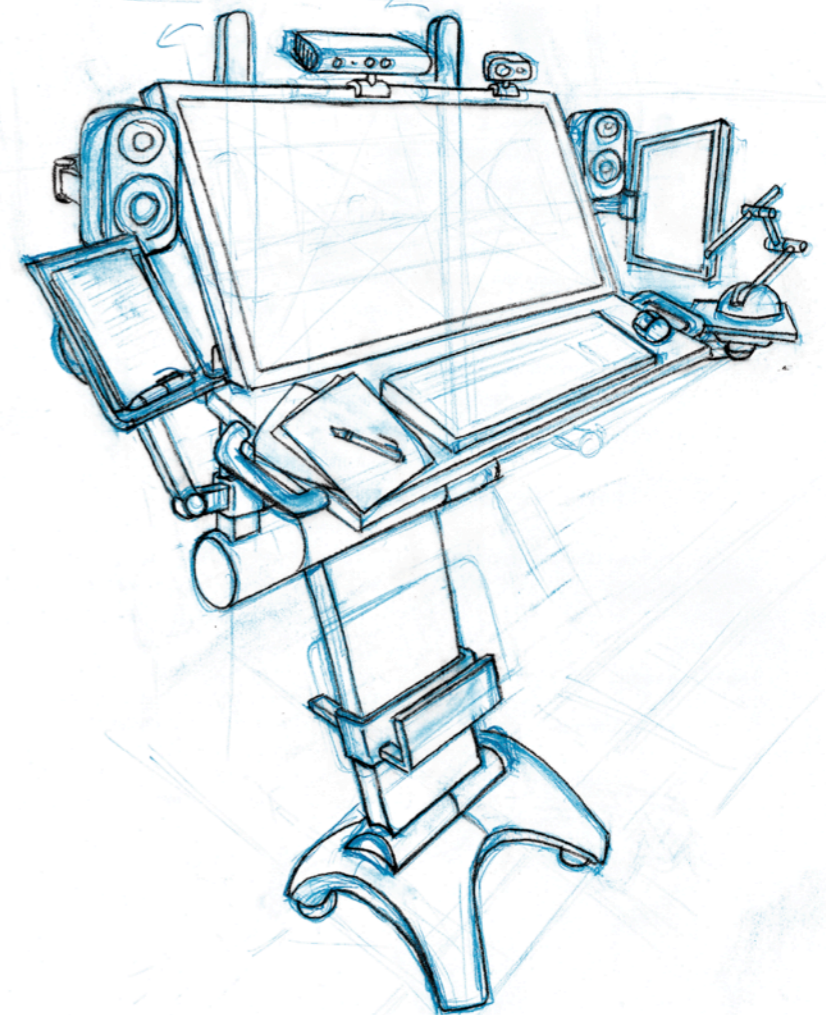
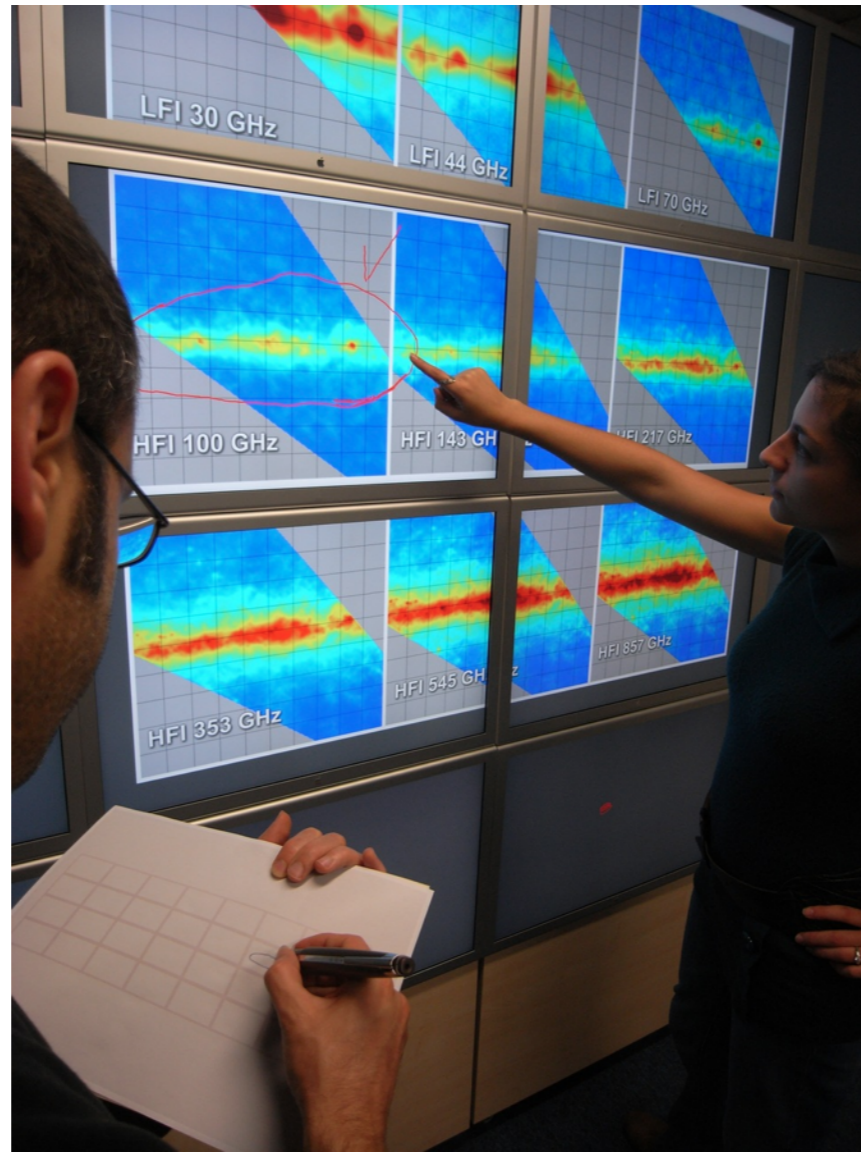
.system and prog



.creative prototy



.adaptability for

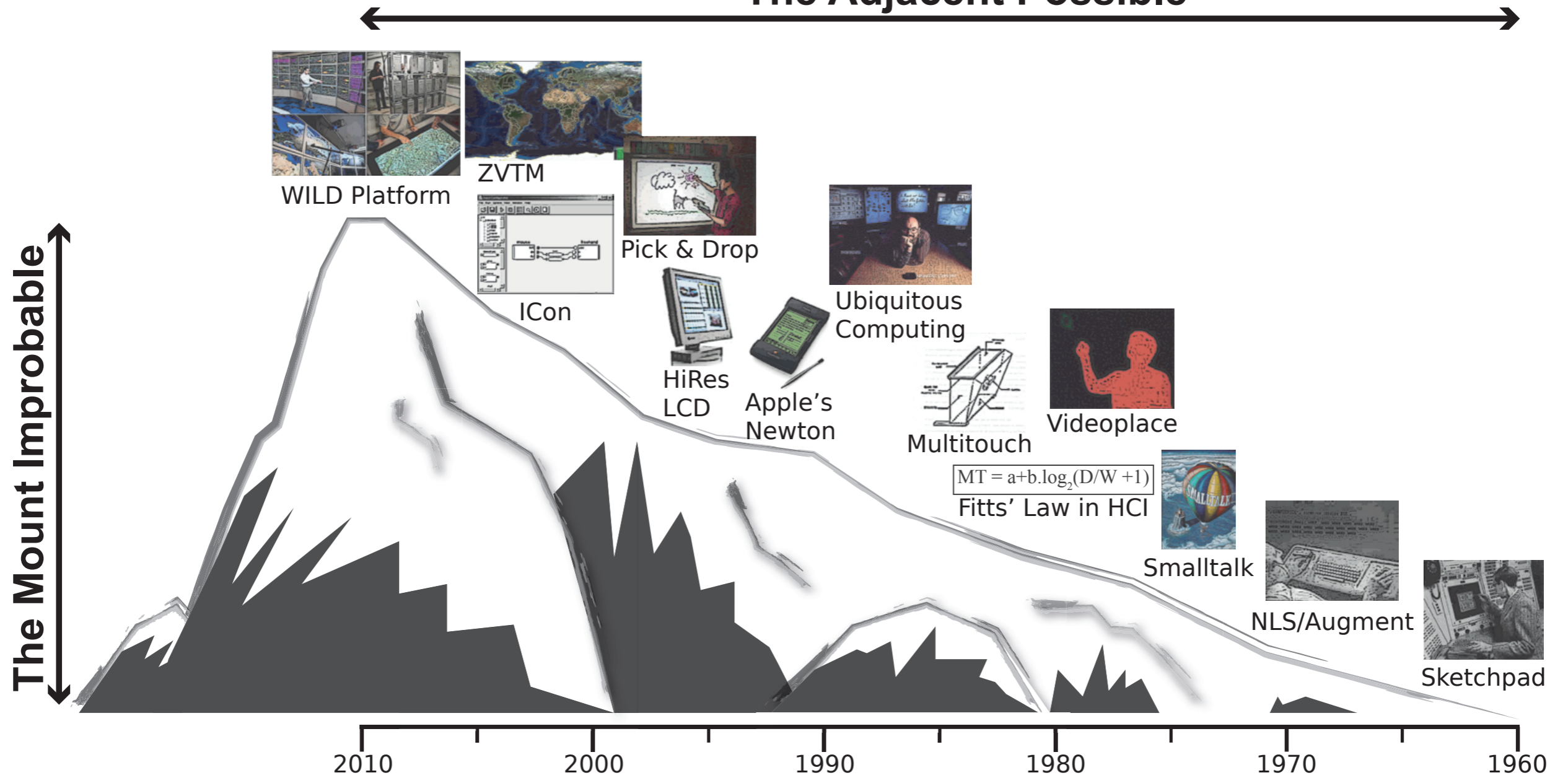


.interaction as a first-class object

- ↳ interaction manipulation
- interaction transformation

to be continued...

The Adjacent Possible



to be continued...

Double-click to edit