

Wendy E. Mackay

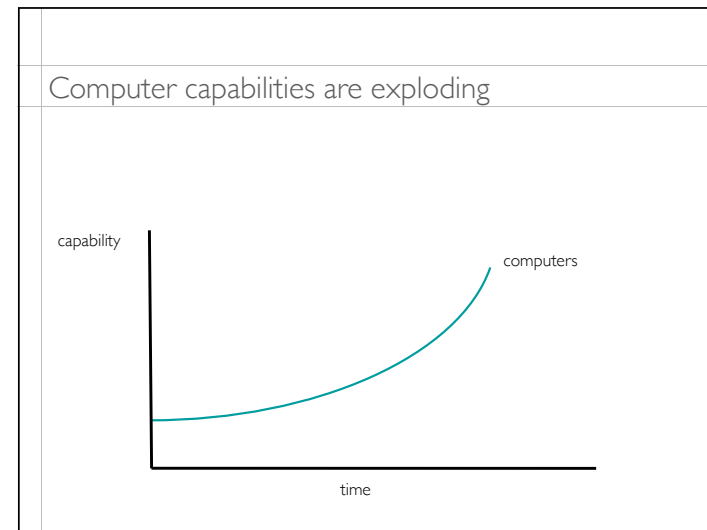

Fundamentals of
Situated Interaction

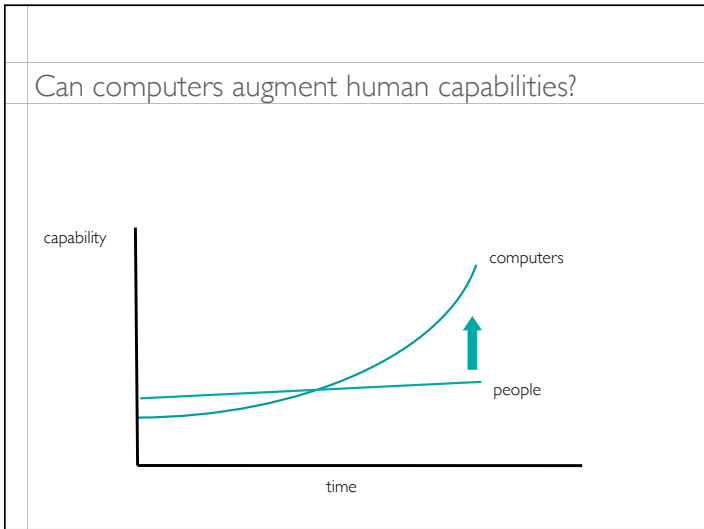
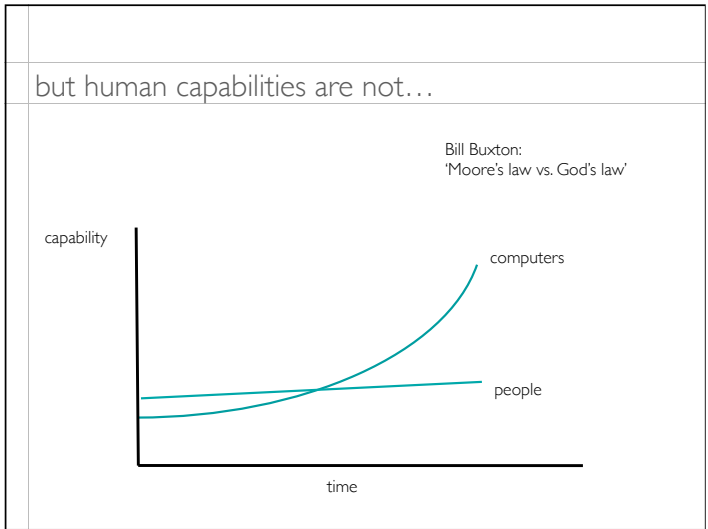
Wendy Mackay & Michel Beaudouin-Lafon
27 September 2019

mackay@iri.fr *mbl@iri.fr*

Human-Computer
Partnerships
or
Co-Adaptive Instruments

Computer hardware
has changed dramatically over the past
40 years ...





Key Challenge

How can we improve interactive systems, given today's ever-increasingly complex computational environment?

A collage of images showing people in various settings interacting with computers and data. On the left, a person in a yellow hard hat looks at a large screen. In the center, a person in a white shirt works at a computer. On the right, a person in a white shirt works at a computer with a large screen. The background is a busy office or control room.

We have multiple relationships with computers

- Computer as a *tool*
I accomplish the task myself
- Computer as a *servant*
It accomplishes the task for me
- Computer as a *medium*
It lets me communicate with other people

Three small images illustrating the relationships: 1. A hammer, representing a tool. 2. A person in a white shirt standing next to a computer, representing a servant. 3. A person in a white shirt sitting at a computer, representing a medium.

Graphical User Interfaces

Designed for executive secretaries to process documents in a completely different technology environment

Dates back to the 1970s to:
copy hand-written notes
check for mistakes
format on letterhead

Problem:
Brilliant then,
out-moded today



GUIs are a vindication ... and a challenge

Human-Computer Interaction research
fought hard to make interfaces easier to use

Today, novices easily accomplish simple tasks

GUIs are a vindication ... and a challenge

Human-Computer Interaction research
fought hard to make interfaces easier to use

Today, novices easily accomplish simple tasks

Yet ...
advanced research in interaction techniques
is rarely adopted in commercial systems

Today, experts use inefficient techniques and are
constantly forced to change their behavior


Desktops, the web and apps ...

Require constant relearning:

- each new version introduces arbitrary changes
- each system requires slightly different interaction

Require high visual attention
Do not scale
Depend on specific devices

Smartphones are easy ... but not powerful



The diagram illustrates four basic smartphone gestures: Tap (a single finger), Drag (a finger moving across the screen), Swipe (a finger moving quickly across the screen), and Pinch (two fingers moving towards each other). To the right is a photograph of a hand holding a smartphone.

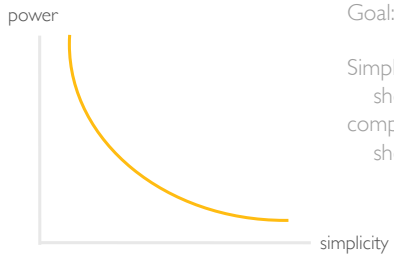
Smartphones are easy ... but not powerful

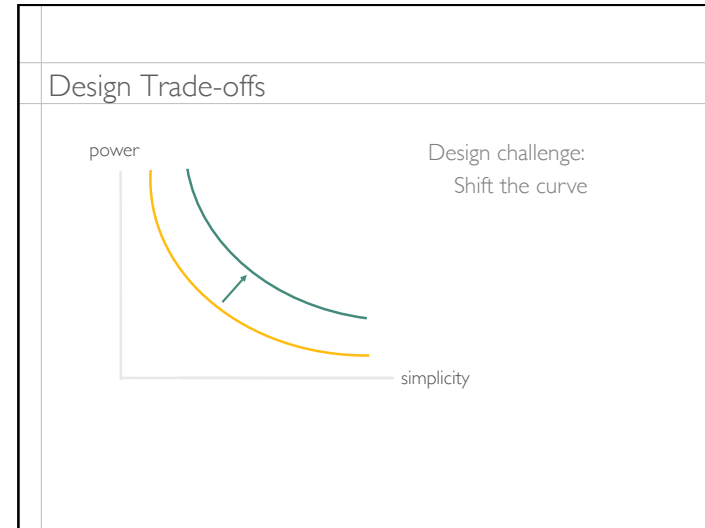
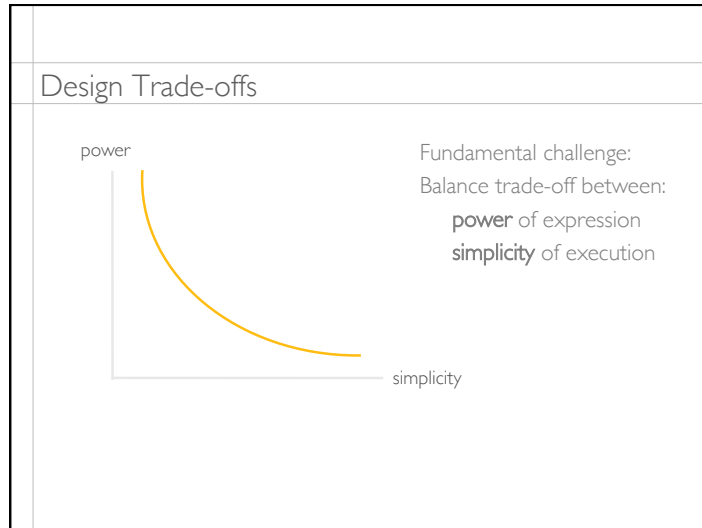
What about creativity and expression?



Three small images are shown below the text: a violinist playing a violin, a person wearing a mask and holding a violin, and a chef working at a kitchen counter.

We need to reassess human-computer interaction
Early assumptions about graphical user interfaces no longer hold
Everyone, not just experts manages increasing quantities of data faces information overload constantly relearns the details of interaction
Redefine what we mean by "computer literacy"

Design Trade-offs
 <p>A graph with 'power' on the vertical axis and 'simplicity' on the horizontal axis. A yellow curve starts high on the power axis and curves downwards towards the simplicity axis, indicating that as power increases, simplicity decreases.</p>
Goal: Simple things should be simple ... complex things should be possible



Strategy: Combine two key concepts

Instrumental interaction
(Michel Beaudouin-Lafon)

and

Co-adaptive phenomena
(Wendy Mackay)

Human-Computer Relationships

Between **people** and **physical** tools:
follow well-known physical principles
users can learn them
users can appropriate them

Human-Computer Relationships
<p>Between people and physical tools: follow well-known physical principles users can learn them users can appropriate them</p> <p>Between people and computer tools: follow arbitrary constantly changing rules users must learn, and relearn, and relearn them users break them when they try to appropriate them</p>

Focus on interaction, not interfaces
<p>How can we let users control interaction in a flexible, reusable way, developing expertise without constantly relearning skills?</p> <p>Solution: Co-adaptive Instruments Separate interaction from data and functionality Interaction becomes a first-class object</p>

Co-adaptive phenomena
<p>Inspired by co-evolution in biology Organisms create their environment even as they adapt to it</p> <p>Anaerobic bacteria change the atmosphere making it possible for aerobic bacteria to emerge</p> <p>Users change spreadsheets from an addition tool to a tool for exploring 'what if' scenarios</p>

<p>Key phenomenon: <i>Co-adaptation</i></p> <p>Users <i>adapt</i> to a new system they learn to use it</p> <p>Users <i>adapt</i> the new system to their own needs they appropriate and change it</p>

Co-adaptive instruments
<p>Creative activities require both especially when integrating physical and digital information</p> <p>Create digital tools that are as intuitive, and learnable, as physical tools</p>

<i>Reciprocal Co-adaptation</i>
<p>People adapt their behavior to technology ... they learn it</p> <p>People adapt the technology for their own purposes ... they appropriate it</p> <p>Computers adapt their behavior to people ... machine learning</p> <p>Computers adapt human behavior ... training</p>


Co-adaptive Instruments

Worthwhile spending time and energy learning them

Complex tools become accessible
can learn cognitive and sensori-motor skills
can adapt to new situations

Move beyond graphical user interfaces to expert instruments

To do this:
Extract widgets from applications to create personal instruments



Human-Computer Partnerships

Wendy E. Mackay

People have rich cognitive and sensory motor capabilities

increasingly,
so do computers

Why is the interface so limited?


Physical tools
follow the laws of physics
we learn them
we appropriate them

Computer tools
follow the whims of programmers
we learn, and relearn and relearn
and then we break them!

Learning to play a musical instrument
—from novice to virtuoso—
the instrument becomes part of the body



Compare to learning software:
every 'upgrade' changes the interface
tools belong to the application, not the user




Our vision:

- Software tools should be incrementally learnable
- People should choose and control their own tools
- Software tools should be easy to appropriate


Octopocus: Learning complex gestures

Dynamic partnership:
Progressive algorithms reveal intermediate recognition states




Octopocus: Learning complex gestures

Experts *just do it*



Octopocus: Learning complex gestures

Experts *just do it*
Novices hesitate ... which activates:
feedforward shows current available gestures
feedback shows what the recognizer sees



Wendy E. Mackay



Physical tools are easy to appropriate — software tools are not

Arpege: Learning chords on a multi-touch surface

Beyond one- and two-finger gestures :
novice to expert transition
feedforward and feedback

A 2x2 grid of diagrams illustrating hand gestures on a multi-touch surface. The top-left diagram shows a hand with fingers touching several points. The top-right diagram shows a hand with fingers touching a single point. The bottom-left diagram shows a hand with fingers touching two points. The bottom-right diagram shows a hand with fingers touching three points. To the right of the grid is a photograph of a person's hands interacting with a large, multi-touch surface displaying a map.

Arpege: Design and learning of multifinger chord gestures

Submitted for review
CHI 2010

Dynachord: Combining chords and gestures

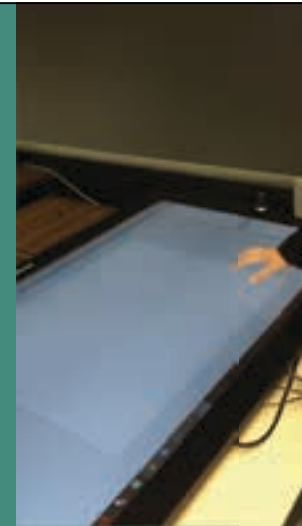
Chord sequences for a larger chord vocabulary
Dynamic adjustment of parameters



Dynachord

Enter a chord with one hand
to choose a color

Continuously adjust the color
with the other hand



How can we help users choose
and control their own tools ?



Appropriation

Interaction designers usually assume that users will focus
on their system and use it as intended

Users often use systems in different ways

They may have a different mental model of the system

They may turn 'mistakes' into opportunities

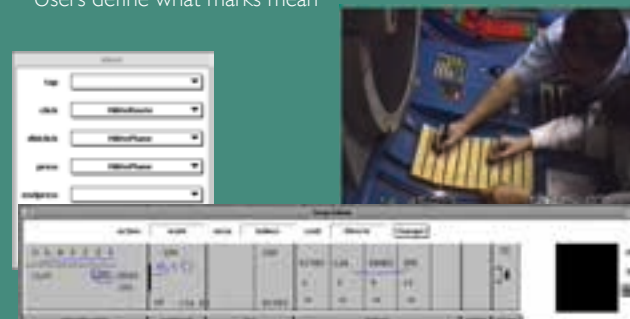
'Bugs' become 'features'

Anything that involves communication among people
is usually adapted for new purposes

How can we help users appropriate technology ?	
<p>Creating a partnership in which the user defines the semantics of the interaction with the computer</p>	
Interaction Browser :	Linking marks to actions
Knotty Gestures :	Interacting while writing
Musink :	Creating a user-defined language
Façades :	User-reconfigurable interfaces

Interaction browser: User-defined commands

Air traffic controllers annotate flight strips
 Marks can be linked to RADAR and other computer functions
 Users define what marks mean




Striptic

Flights in my Hands: Coherence Concerns in Designing a Tangible Space for Air Traffic Controllers, (Letondal et al., CHI'14)




Knotty Gestures

Draw a dot, define a command
 Interact while writing
 Interact with command later




Knotty Gestures

Interactive Paper
Users interact as they write
or define their own gestures
and interact with them later




Knotty Gestures: Creating an interactive controller

Draw a line with a 'knotty gesture' at the end




Choose "recording" to define the type of line

Knotty Gestures: Creating an interactive controller



Define where the recording will start

Knotty Gestures: Creating an interactive controller



Define an end point for the recording

Knotty Gestures: Creating an interactive controller

Slide the pen along the line to move forward or backward on the recording

Drawing a Math Calculator

This line acts as a base for attaching mathematical value sliders

The knotty gesture at the end defines the type

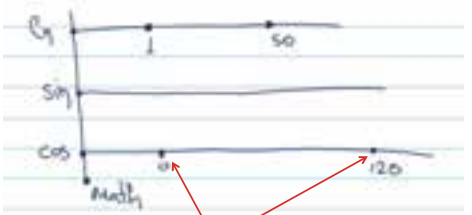
Drawing a Math Calculator

Any knot drawn on line lets the user select a mathematical function

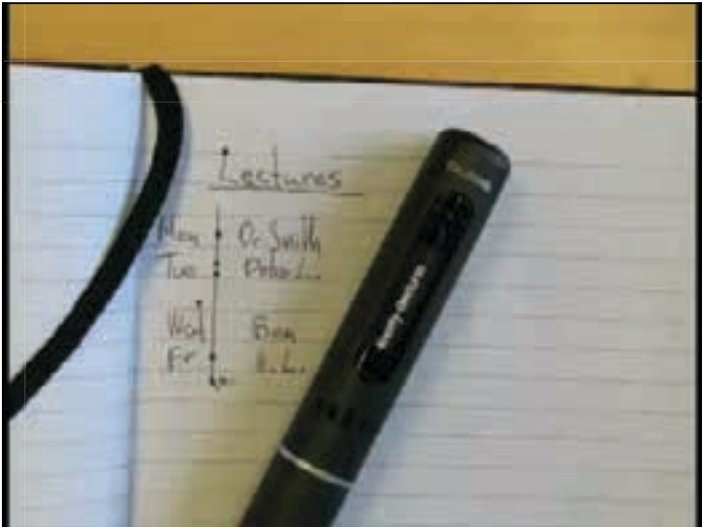
Drawing a Math Calculator

The extensions act as value controllers
Sliding the pen over the line moves through range of function values, shown on the pen display

Drawing a Math Calculator



Knots may define ranges or act as traces of past interactions with specific values



But recognition is not the only problem ...

Recognition must be *good enough*
but users override and reinterpret
no single 'correct' interpretation
recognized and non-recognized gestures co-exist

Real question:
Can Musink support the creative process?
What are the design implications for Musink v2?

Semi-Structured Delayed interpretation

Key insights:

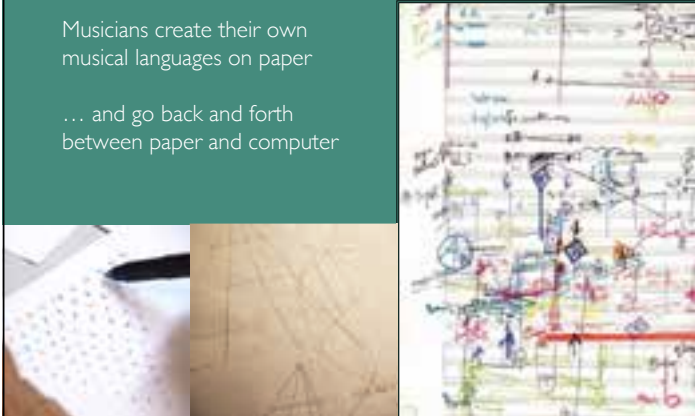
- Spatial structure on paper
improves recognition
under user's control
- Recognition need not be immediate
users decide *when* to interpret
interpretation *changes over time*

Wendy E. Mackay

MusInk



Musicians create their own musical languages on paper

... and go back and forth between paper and computer




MusInk Define meaning of gestures over time

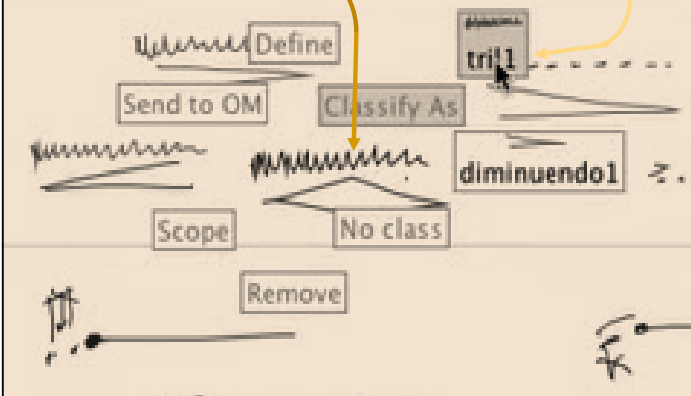
User decides if and when to interpret each gesture



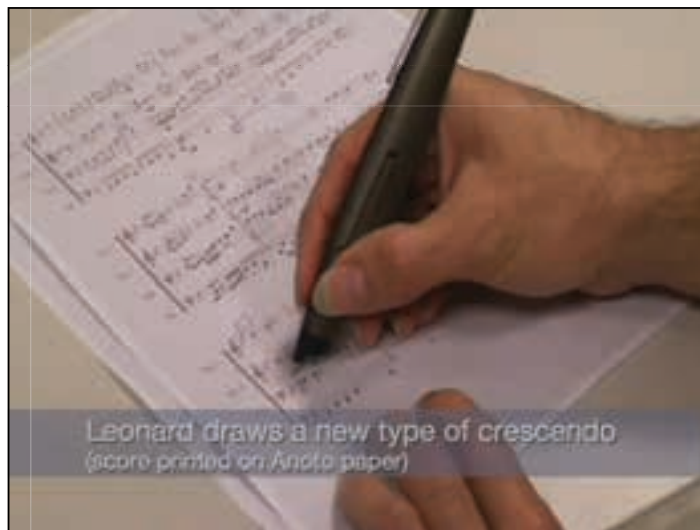
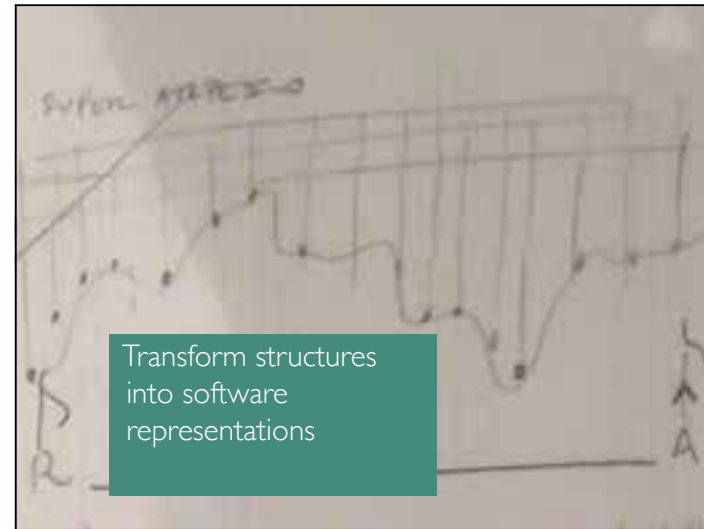
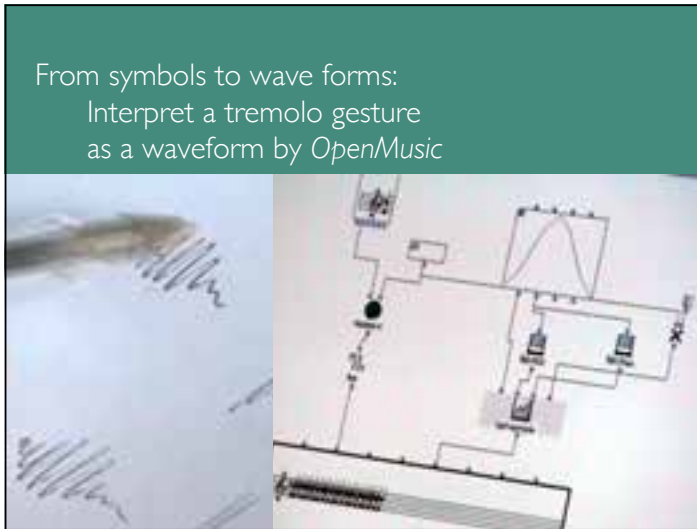
Create interactive annotations



Reclassify a 'squiggle' and turn it into a trill



Wendy E. Mackay



Musink: Semi-structured, delayed interpretation

Users decide when and how each annotation should be interpreted by the computer

score pointers

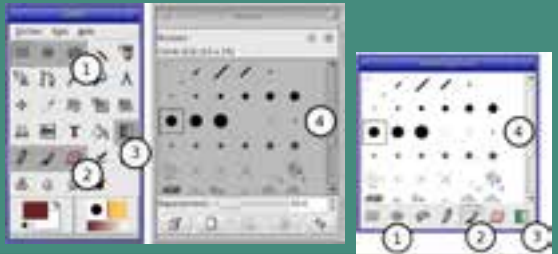
scoping gestures

textual elements

connectors

Façades: Reconfiguring interfaces

Users can adopt parts of **any** Linux interface and reconfigure it for specific needs
Grab three selections from GIMP and choose a brush and create a new, custom-made palette

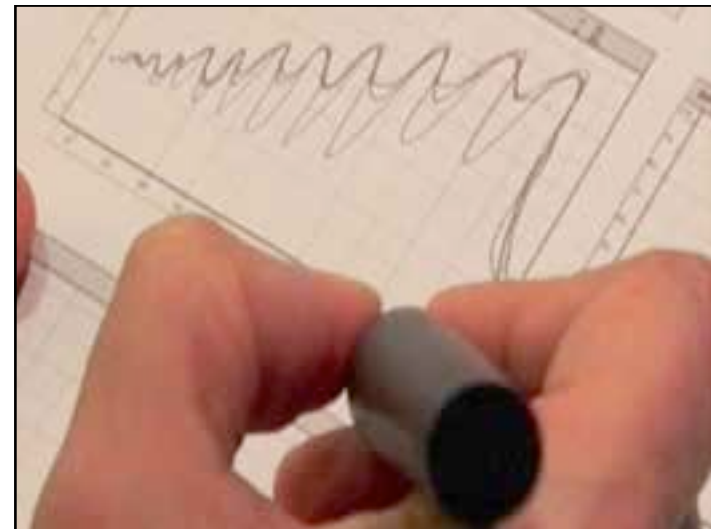


Substrates

Define the structures and rules
Ways to interpret the data

Different structures

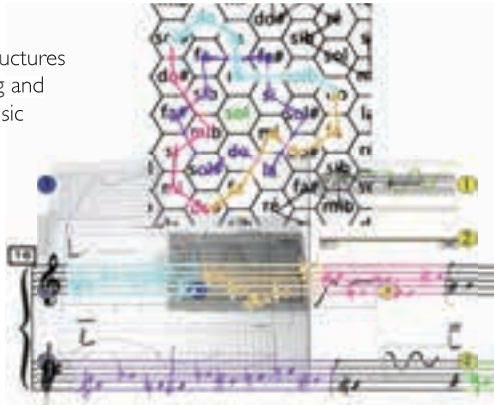
to facilitate interpretation



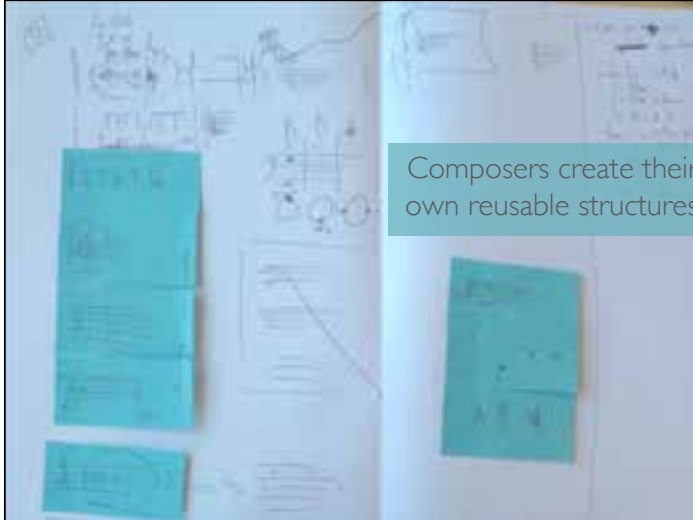
Wendy E. Mackay

Paper Substrates: create own language & structure

Composers create new structures for interpreting and composing music

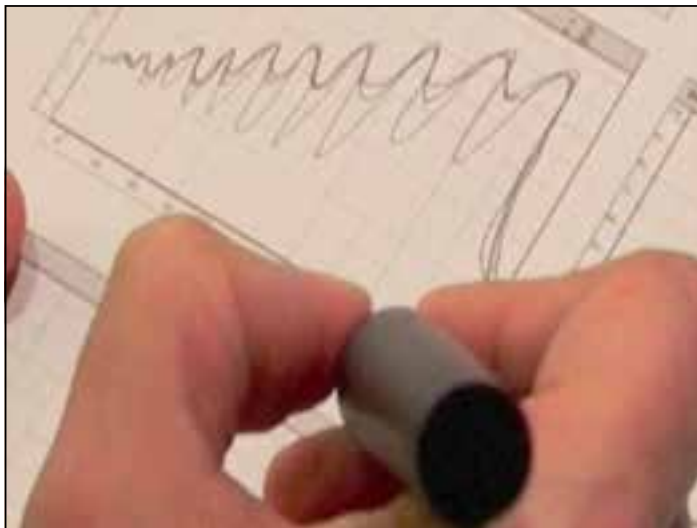


The image shows a collage of musical notation and a hexagonal grid. The grid contains letters and numbers, possibly representing a musical language. Below the grid are several staves of musical notation with notes and clefs.



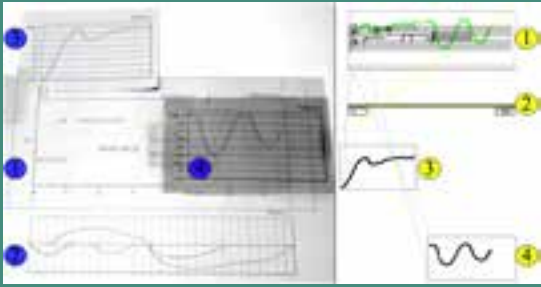
Composers create their own reusable structures

The image shows a wall covered with hand-drawn diagrams and sticky notes. The diagrams appear to be flowcharts or structural diagrams. There are several blue sticky notes attached to the wall, some with text and some with diagrams.



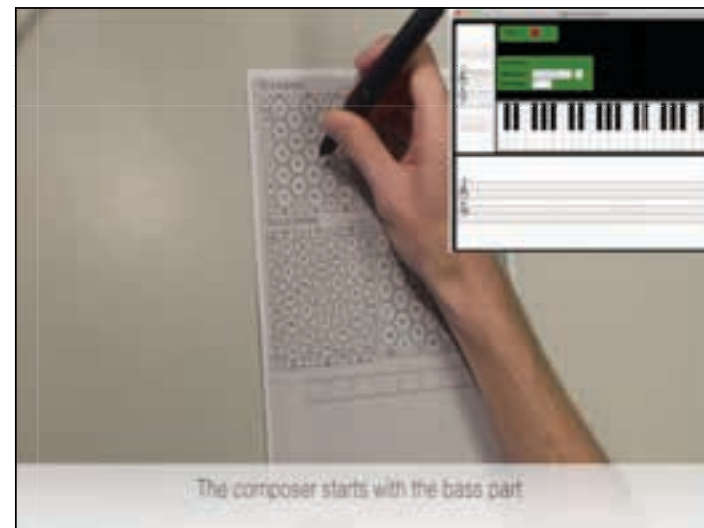
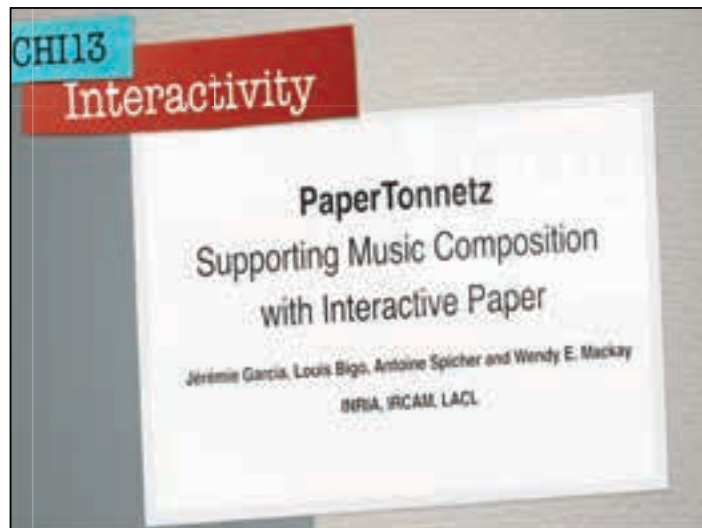
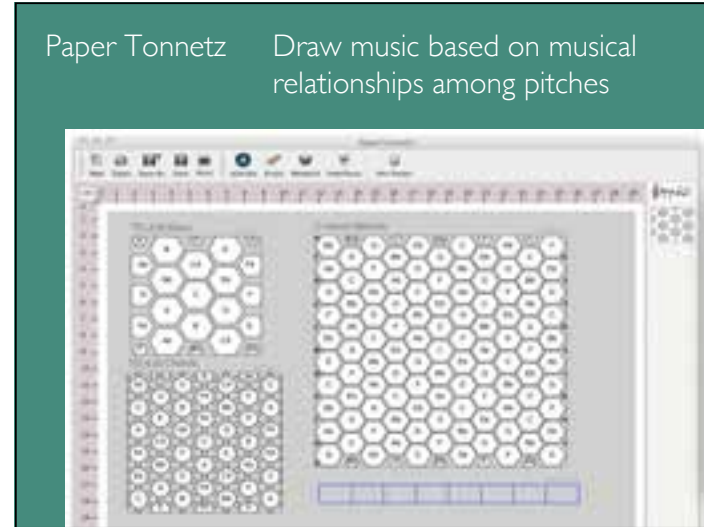
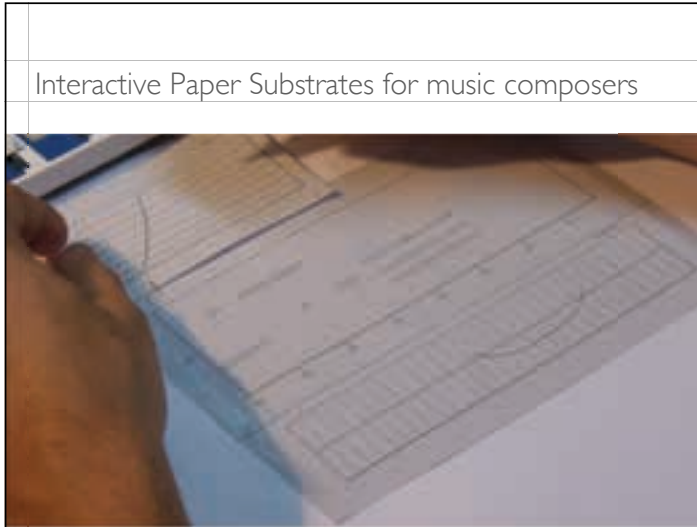
Paper Substrates

A substrate is both an instrument for interpreting a personalized language and an object in its own right



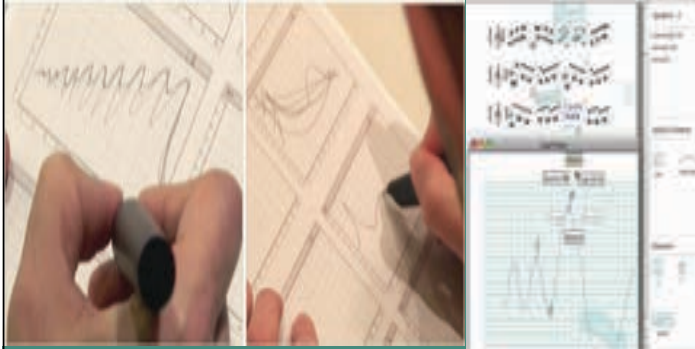
The image shows a collage of musical notation and diagrams. It includes several staves of musical notation with notes and clefs, and a central diagram with a grid and a wavy line. There are also some smaller diagrams and notes scattered around.

Wendy E. Mackay



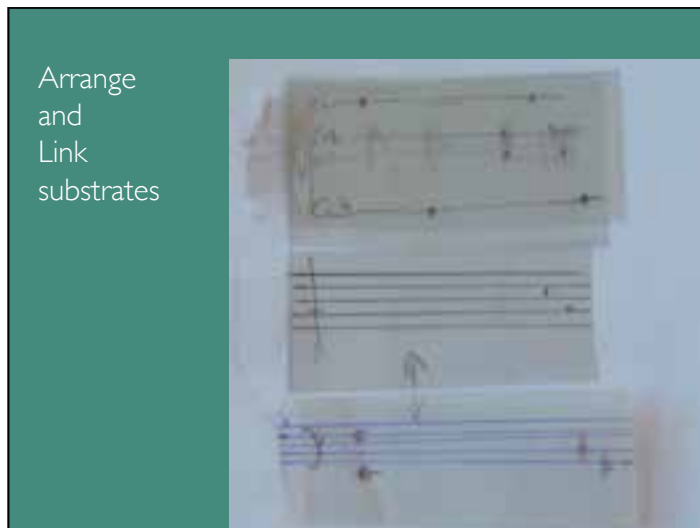
Wendy E. Mackay

Paper Substrates Composer create their own reusable musical structures



establish relationships among them

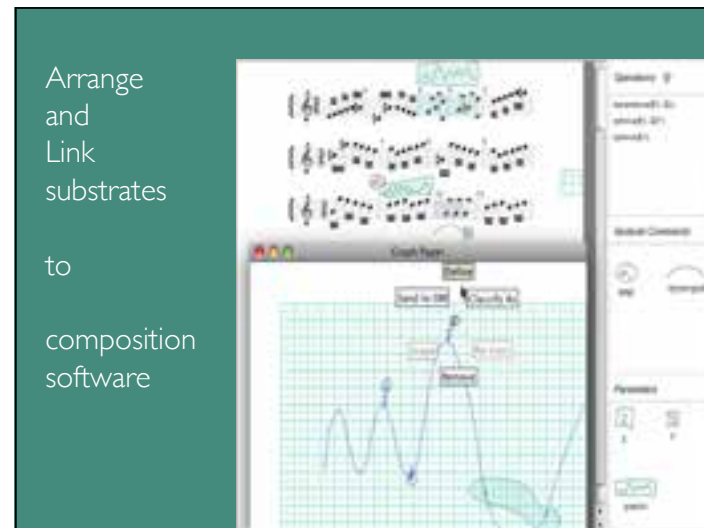
Arrange and Link substrates



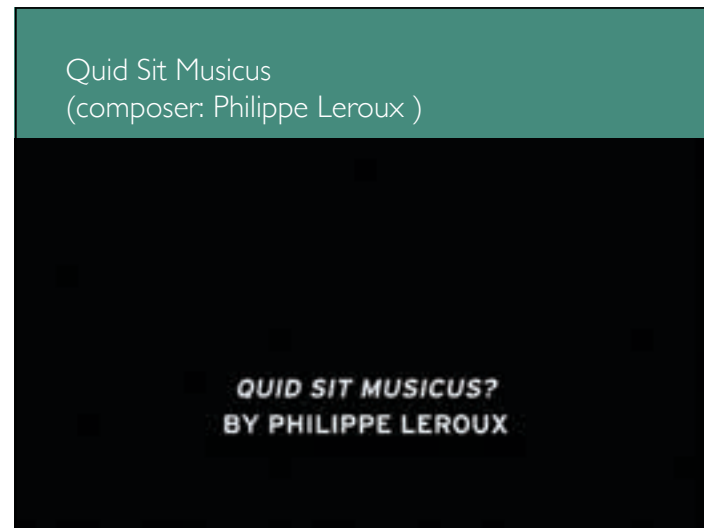
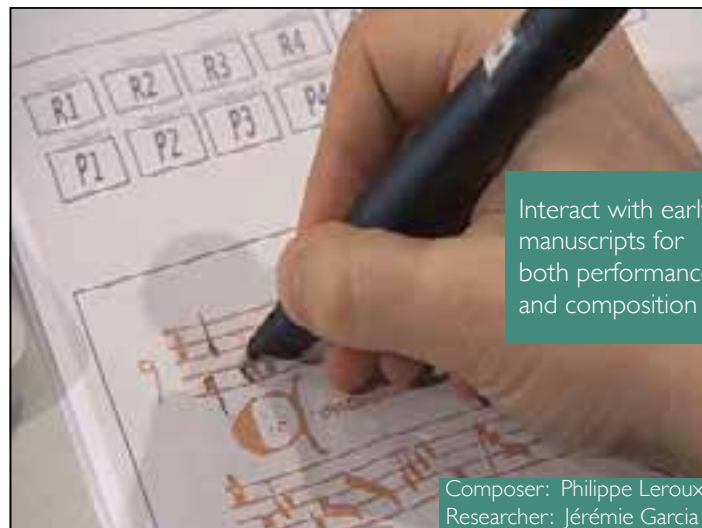
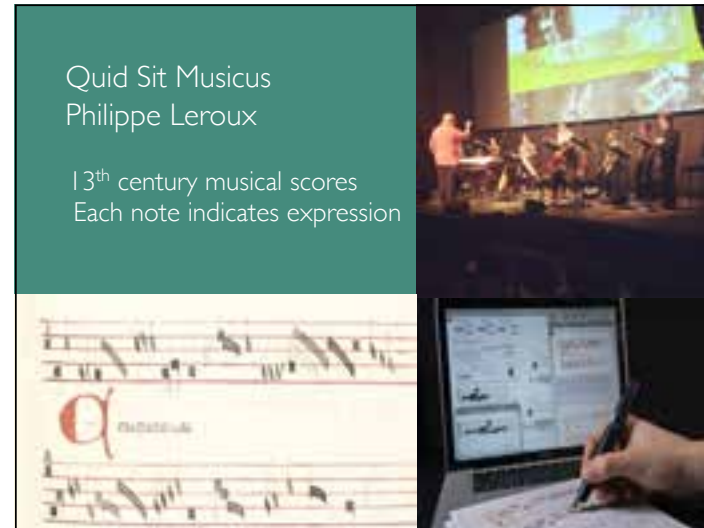
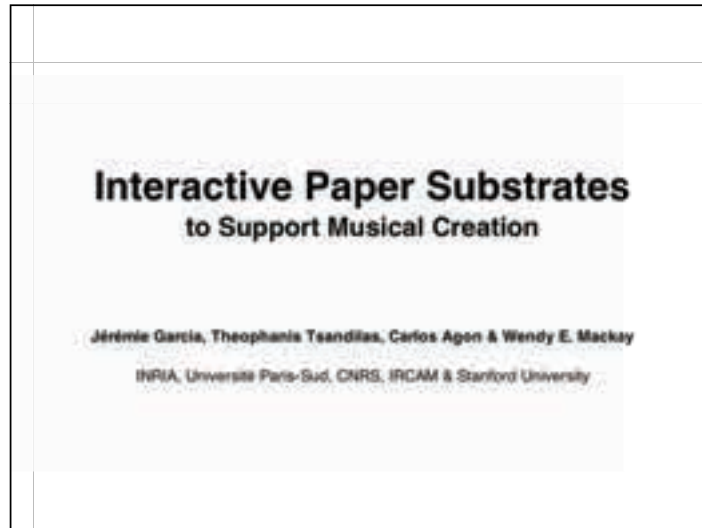
Arrange and Link substrates

to

composition software



Wendy E. Mackay



Create human-computer partnerships

People

- adapt to technology – they learn it
- adapt the technology – they appropriate it

Computers should

- adapt to people – they learn (AI)
- adapt people's behavior – they teach


Reciprocal Co-adaptation

People adapt their behavior to technology
... they learn it
People adapt the technology for their own purposes
... they appropriate it

Computers adapt their behavior to people
... machine learning
Computers adapt human behavior
... training

What do we mean by 'partnership' ?


Take a taxi
Driver in control



What do we mean by 'partnership' ?


Take a taxi
Driver in control

Drive a motorcycle
User in control



What do we mean by 'partnership' ?

- Take a taxi
Driver in control
- Drive a motorcycle
User in control
- Ride a horse
Shared control



- Computer as *tool*
Empower users
- Computer as *servant*
Delegate tasks
- Computer as *medium*
Communicate



- Human-Computer Interaction
- Artificial Intelligence
- Mediated Communication

Human-Computer Partnerships

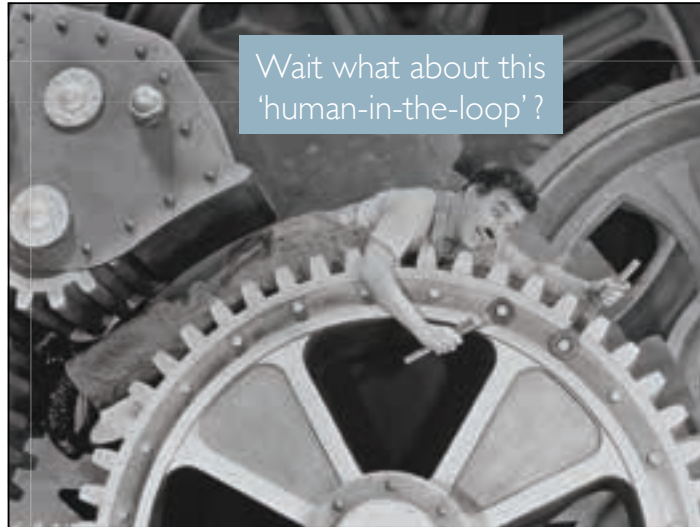
Human-in-the-loop

A 'simple' human-computer partnership

User types – Google suggests – User chooses

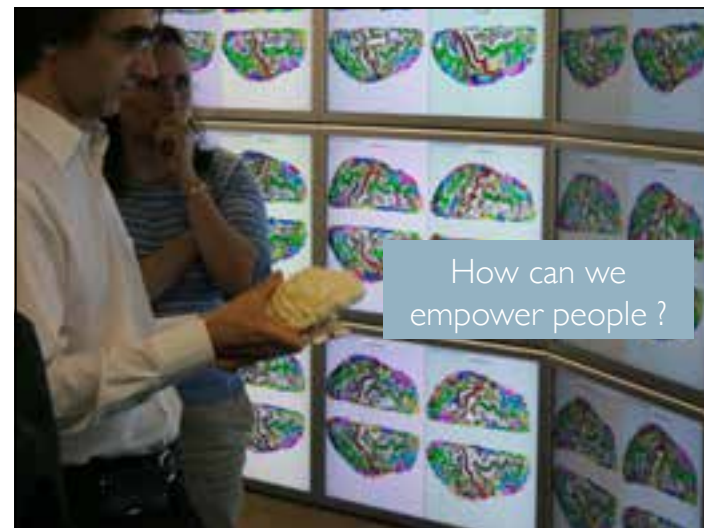


Wendy E. Mackay



Human-Computer Partnerships
<i>Human-in-the-loop</i> Machine learning perspective: Human helps <i>improve the algorithm</i>

Human-Computer Partnerships
<i>Human-in-the-loop</i> Machine learning perspective: Human helps <i>improve the algorithm</i>
<i>Computer-in-the-loop</i>



Human-Computer Partnerships

Human-in-the-loop
Human helps *improve the algorithm*

Computer-in-the-loop
Human-computer interaction perspective:
Computers *empower the human user*

Unified principles of interaction

Two complementary perspectives:
System: How to build it ?
Instrumental Interaction and Substrates

with Michel Beaudouin-Lafon

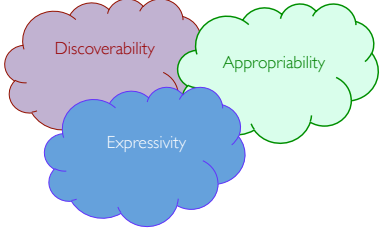
Unified principles of interaction

Two complementary perspectives:
System: How to build it ?
Instrumental Interaction and Substrates

Human: How to interact with it ?
Co-adaptive Systems
Human-computer partnerships

Human-computer partnerships


People can
adapt to technology they learn it
adapt the technology they appropriate it





Discoverability


Expressivity

Appropriability

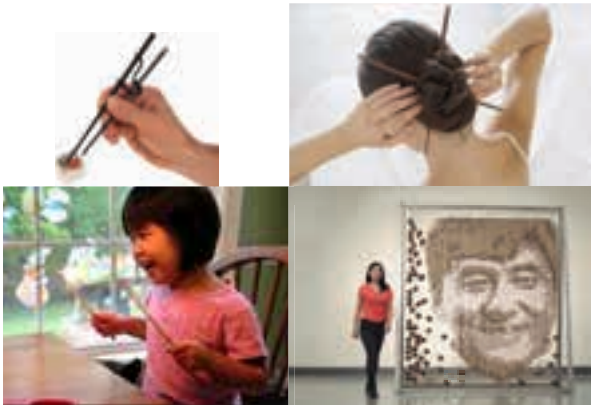
What can we learn from physical tools ?


What can we learn from physical tools ?


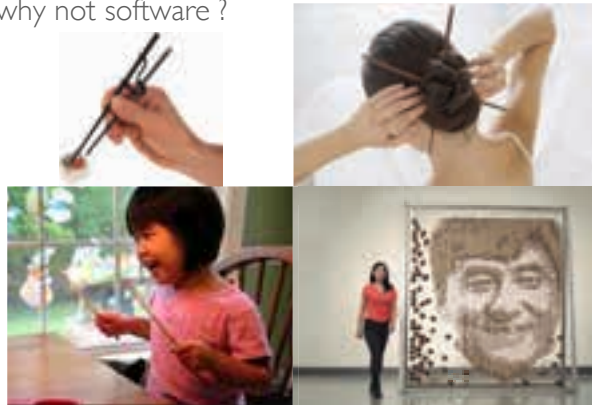
	Physical tools are easy to reinterpret (or make mistakes)
--	--

We can appropriate physical tools

108


We can appropriate physical tools



We can appropriate physical tools
... why not software ?

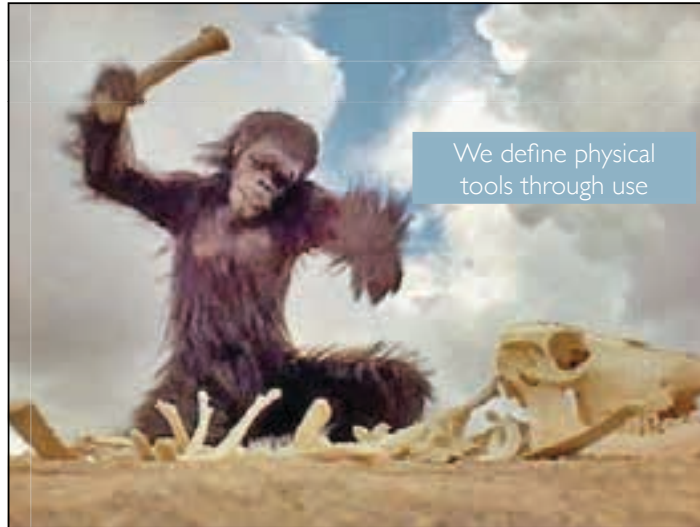


Why do software applications choose for us ?







Imagine needing the manufacturer's stick to hit a drum !

Wendy E. Mackay

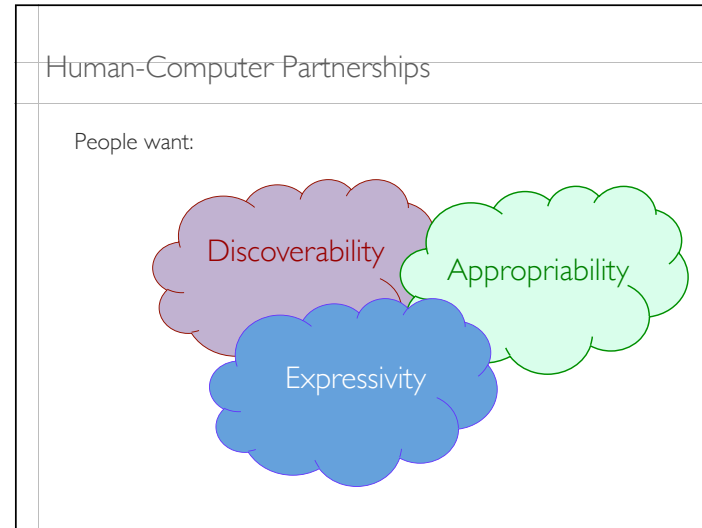


Why can't we learn to 'play' software tools ?
without relearning the interface
with every software upgrade ?



Computer as <i>tool</i> Empower users		Human-Computer Interaction
Computer as <i>servant</i> Delegate tasks		Artificial Intelligence
Computer as <i>medium</i> Communicate		Mediated Communication

Human-computer partnerships	
People can <i>adapt to</i> technology <i>adapt</i> the technology	they learn it they appropriate it
Computers can <i>adapt to</i> people <i>adapt</i> people's behavior	they learn (AI) they teach




How can I learn new gestures and Discoverability



Octopus

Learn gestures that issue commands


- Progressive feedforward
What gestures are available?
- Progressive feedback
What did the system recognize?



Olivier Bau

Dynamic partnership

Experts just perform the gesture
Novices pause ... the guide appears




Inking the 'Help' command

How can I define my own gestures?
Appropriability

Create personal gesture commands
Choose easy-to-remember gestures
Progressive feedforward reveals whether


- command exists
- it is recognizable



Joe Malloch, Carla Griggio & Joanna McGrenere

Fieldward: create personal gesture commands

Fieldward
Shows a color gradient indicating optimal directions to make a recognizable gesture



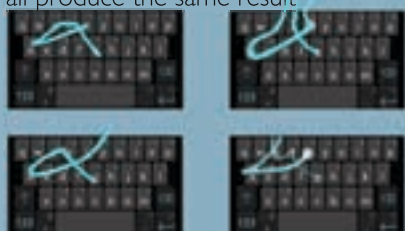
How can I express myself?
Expressivity

Expressive Keyboard CHI'16

Redefining gesture-typing keyboards
to support user expression

Four ways to type the word
"great"


all produce the same result



Jessalyn Alvina, Joe Malloch

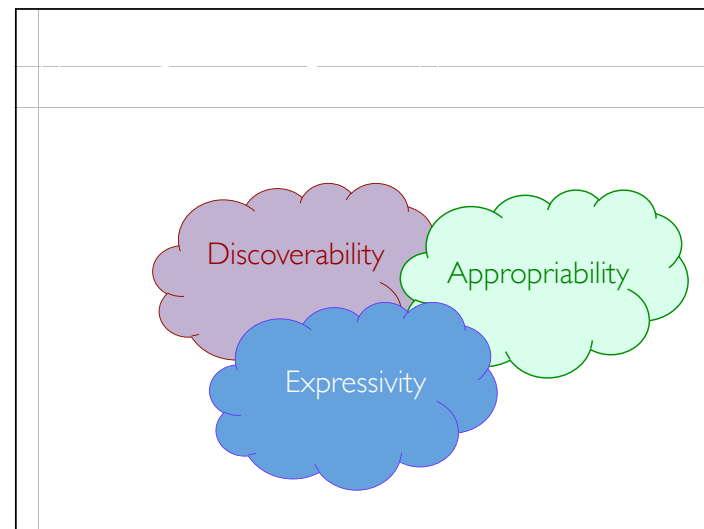

Machine learning guesses the correct word
Gesture variations creates expressive text

Users control
text color
font style
and emojis



Expressive Keyboard

The recognition algorithm is highly tolerant
of input variation ... these are all 'great!'

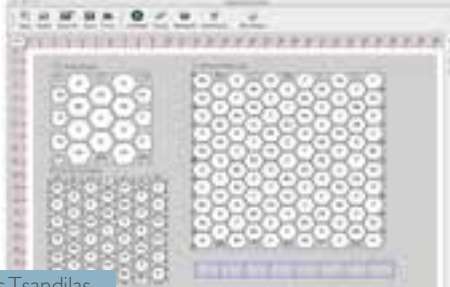


Wendy E. Mackay

How can I learn musical relationships?


Discoverability

Paper Tonnetz



Jérémie Garcia, Fanis Tsandilas

Paper Tonnetz




The composer starts with the bass part

How can I define my own commands?

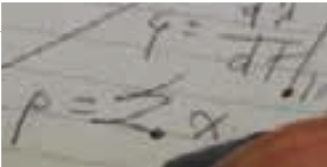
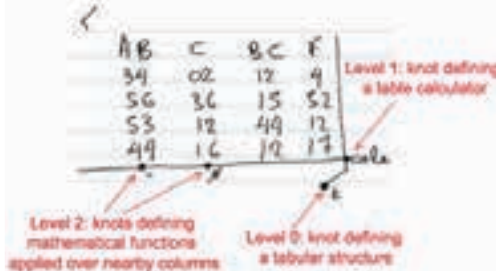
Appropriability

Draw a knot to define a command
Interact now or later



Fanis Tsandilas


Knots can define mathematical or other relationships

A	B	C	D	E
34	02	12	4	
56	36	15	52	
53	12	49	12	
44	16	19	17	


Knotty Gestures

Draw a line with a knot
Choose "recording" to define the type of line



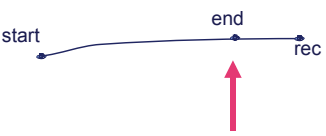
Knotty Gestures

Add another knot to define the start of the recording



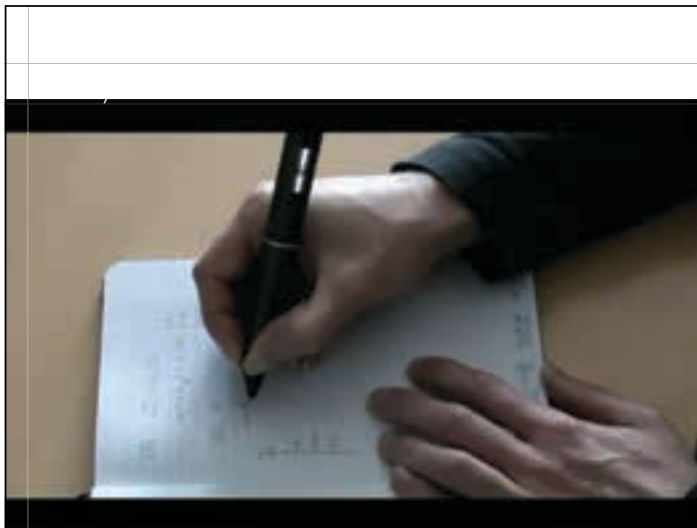
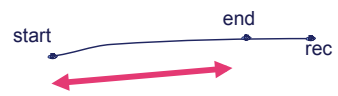
Knotty Gestures

Add a third knot to define the end of the recording



Knotty Gestures

Slide the pen back and forth to play the recording



How can I express myself?
Expressivity



