Fundamentals of Situated Interaction

Wendy Mackay & Michel Beaudouin-Lafon
23 September 2016

Seminar format

Discovering the principles of situated interaction:
- Instrumental Interaction
  - Reification
  - Polymorphism
  - Reuse
- Substrates

- Human-computer partnerships
  - (Reciprocal co-adaptation)

Course Schedule

Building 640 (PUIO)		Friday 1:30 – 4:30 pm
Room B109		Monday 9:30 – 12:30 am
Friday	16 Sept.	Instrumental Interaction
Friday	23 Sept.	Human-Computer Partnerships Choosing articles
Monday	26 Sept.	Presentations (Ph.D.) / Group exercise
Friday	30 Sept.	Presentations / Group exercise
Monday	03 Oct.	Presentations / Group exercise
Friday	07 Oct	Presentations / Group exercise
Friday	24 Oct.	Final Presentations & discussions
Friday	18 Oct.	Final Master reports due

Class activities

Lectures on key concepts (Michel & Wendy)

Seminar presentations (30 min.) (Masters)
- Prresent key concept from 3 papers
- Lead discussion

Seminar presentations (30 min.) (Ph.D.)
- Present recent research
- Lead discussion

Generative Deconstruction (Groups, Ph.D. lead)
- Deconstruct systems
- Generate novel design ideas

Grades

Class participation	= 20%
Seminar Presentation	= 30%
Report & iMuseum entry	= 50%

Additional activity: iMuseum

Read a paper with a novel interaction technique
- Find one key image
- Create an animated .gif
- Create a 3-5 step storyboard
- Write a summary of the novel aspect of the technique
- Identify related techniques
- Identify related references
- Identify commercial uses

Link: https://hci-museum.lri.fr/WWW/
or hci-museum.org

Assignment #1
Readings

Friday, 23 September 13h30

Read:

Beaudouin-Lafon, M. (2000). Instrumental Interaction: an Interaction Model for Designing Post-WIMP User Interfaces. *Proc. ACM Human Factors in Computing Systems*, CHI 2000, The Hague (The Netherlands), CHI Letters 2(1):446-453, ACM Press.

Beaudouin-Lafon, M. & Mackay, W. (2000). Reification, Polymorphism and Reuse: Three Principles for Designing Visual Interfaces. *Proc. Advanced Visual Interfaces*, AVI 2000, Palermo (Italie), ACM Press, pp 102-109.

Assignment #2
Prepare Presentations

Friday, 30 September 13h30

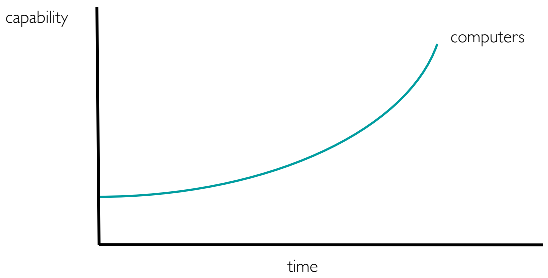
1. Prepare a 15-minute talk
   - choose three papers
     - choose one or two from InSitu/ExSitu
     - choose one or two from ACM/Digital Library
   - describe each technique (show video, if possible)
   - compare to each other and analyze:
     - how do they represent instruments? or not?
     - how are they co-adaptive? or not?

2. Prepare a 15-minute class discussion
   Suggest ideas for revising the techniques
   to become co-adaptive instruments

Human-Computer
Partnerships
or
Co-Adaptive Instruments

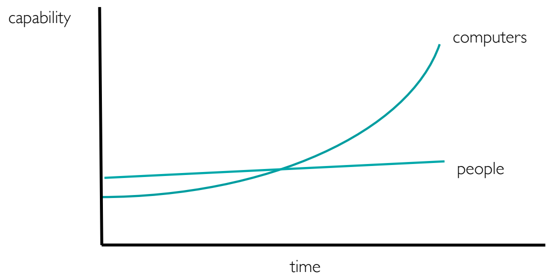
Computer hardware
has changed dramatically over the past
40 years …

Computer capabilities are exploding

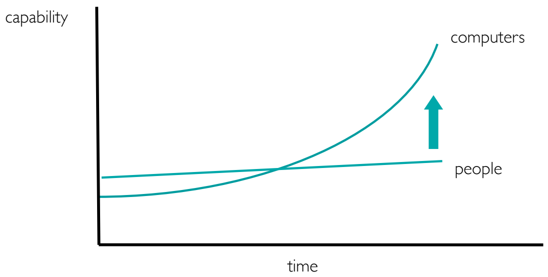


but human capabilities are not…

Bill Buxton:
'Moore's law vs. God's law'



Can computers augment human capabilities?



Key Research Challenge

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

Situated Interaction

Focus on *interaction*
we cannot effectively model user behavior
without taking context into account

Human behavior is planned, but action is situated:
plans are a resource for action, not the action itself

Differentiate between measuring data
and understanding the complexities of context

Methodology

Generative theory
- understand co-adaptive interaction
- principles for creating co-adaptive instruments

Participatory design with creative professionals
- develop novel prototypes
- real-time interaction
- personal language creation
- long-term reusable patterns of interaction

Empirical studies
- controlled laboratory studies
- extended field studies

ExSitu focuses on:
reinventing interactive software
to support creative activities …

based on two key ideas

in|situ| research projects

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

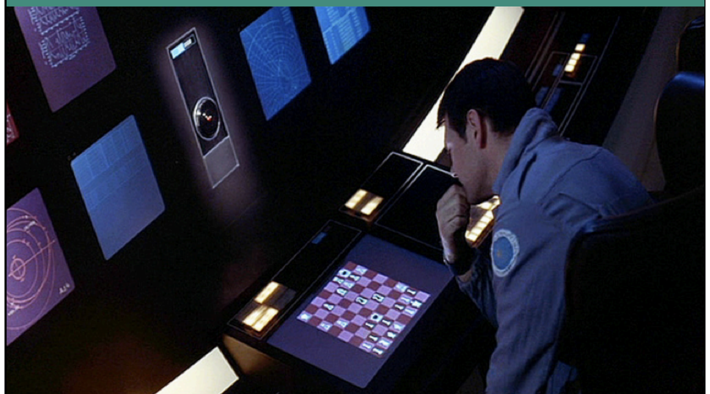
Competing views of the future

Artificial Intelligence

or

Human-Computer Interaction

Hollywood's view of Artificial Intelligence



2001 A Space Odyssey

Contrast with Human-Computer Interaction view

Iron Man

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

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

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

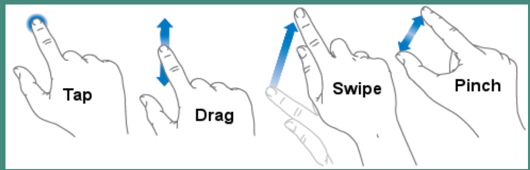
Today, experts use ineffecient techniques and are
constantly forced to change their behavior

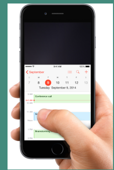
Desktops, the web and apps …

Require constant relearning:
- each new version introduces arbitrary changes
- each system requires slightly different interaction

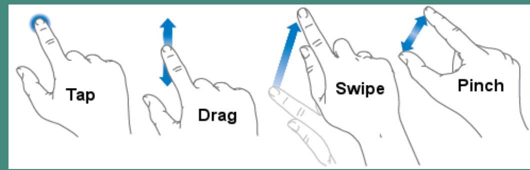
Require high visual attention
Do not scale
Depend on specific devices

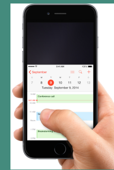
Smartphones are easy … but not powerful





Smartphones are easy … but not powerful





What about creativity and expression?

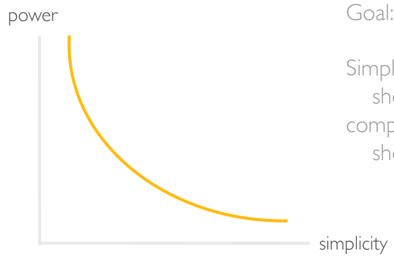
We need to reassess human-computer interaction

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"

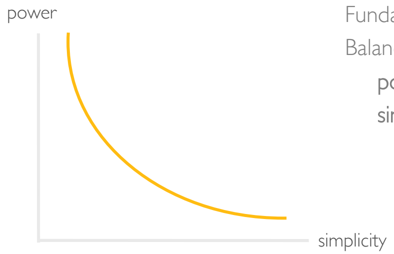
Design Trade-offs



Goal:

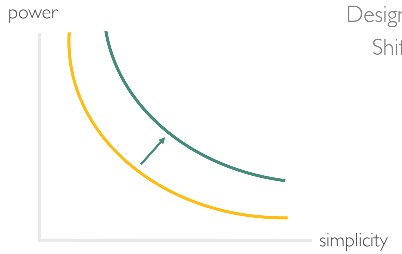
Simple things
should be simple …
complex things
should be possible

Design Trade-offs



Fundamental challenge:
Balance trade-off between:
- power of expression
- simplicity of execution

Design Trade-offs



Design challenge:
Shift the curve

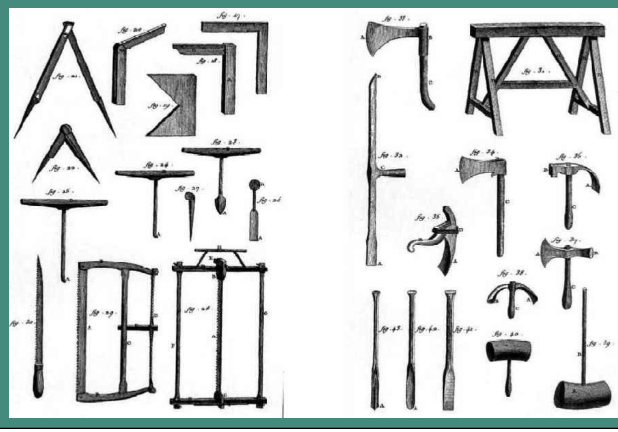
Strategy: Combine two key concepts

Instrumental interaction
(Michel Beaudouin-Lafon)

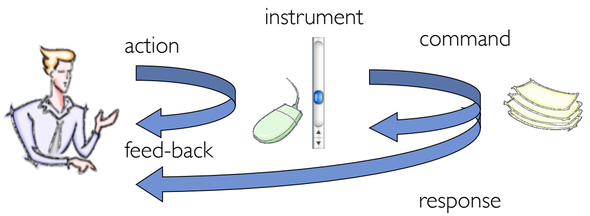
and

Co-adaptive phenomena
(Wendy Mackay)

Instrumental Interaction in the physical world



Instrumental interaction



Two levels of interaction:
- interaction with the instrument
- mediation with the system

Michel Beaudouin Lafon (2000)

Magnetic guidelines

Reification of the alignment command

Power and simplicity
- Align command vs Align object:
- Align (now) vs Align (and keep aligned)

Multiple shapes
- Horizontal, vertical, diagonal, circular, rectangular
- Distribute objects

Decomposition

- Create / Move / Add object / Remove object

What makes an object an instrument?

Relationship between the user and the object

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

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

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

Focus on interaction, not interfaces

How can we let users control interaction
in a flexible, reusable way,
developing expertise without constantly relearning skills?

Solution: **Co-adaptive Instruments**
Separate interaction from data and functionality
Interaction becomes a first-class object

Co-adaptive phenomena

Inspired by co-evolution in biology
Organisms create their environment
even as they adapt to it

Anaerobic bacteria change the atmosphere
making it possible for aerobic bacteria to emerge

Users change spreadsheets from an addition tool
to a tool for exploring 'what if' scenarios

Key phenomenon: *Co-adaptation*

Users *adapt* to a new system
  they learn to use it

Users *adapt* the new system to their own needs
  they appropriate and change it

Co-adaptive instruments

Creative activities require both
  especially when integrating physical and digital information

Create digital tools that are as intuitive, and learnable,
  as physical tools

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

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

Human-Computer
Partnerships

People have rich cognitive and
sensory motor capabilities

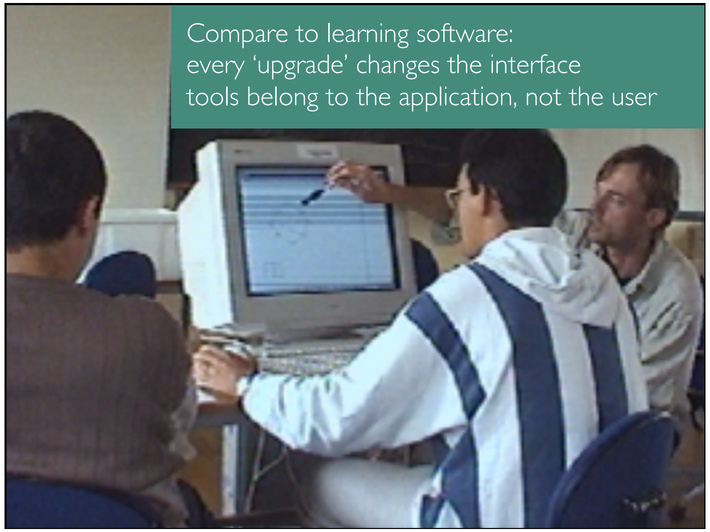
increasingly,
so do computers

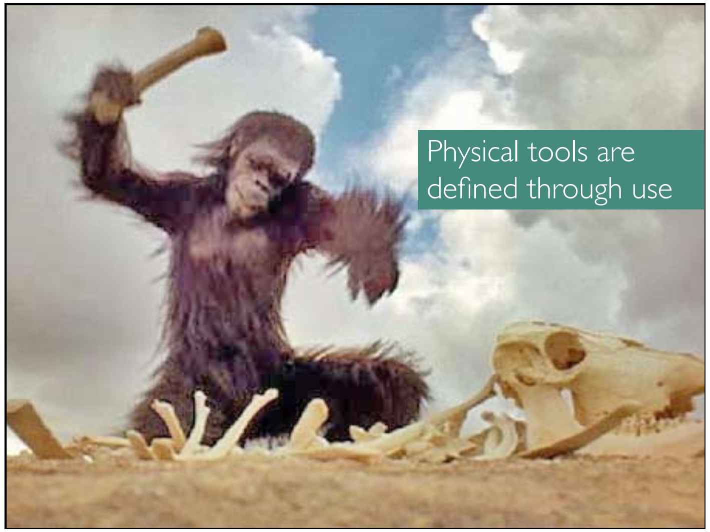
Why is the interface so limited?

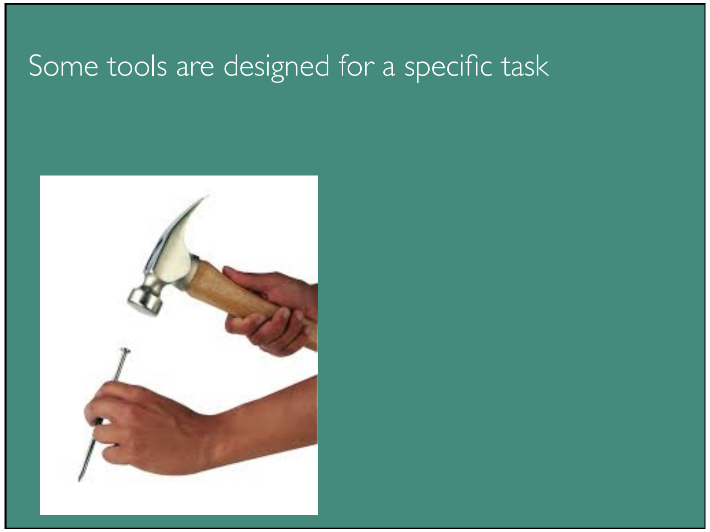
Physical tools
follow the laws of physics
we learn them
we appropriate them

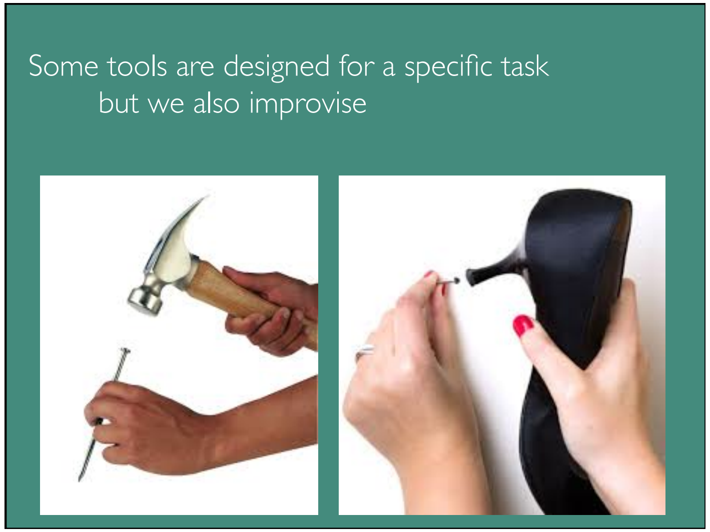
Computer tools
follow the whims of programmers
we learn, and relearn and relearn
and then we break them!

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


Physical tools are
defined through use


Some tools are designed for a specific task


Some tools are designed for a specific task
but we also improvise

People redefine physical tools
… why not software?

Our vision:

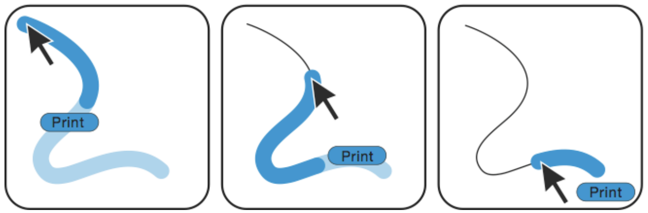
Software tools should be
incrementally learnable

People should choose and
control their own tools

Software tools should be
easy to appropriate

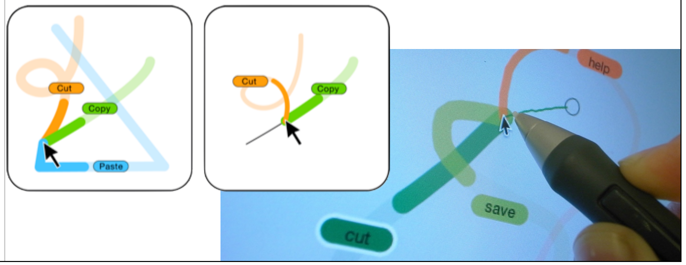
Octopocus: Learning complex gestures

Dynamic partnership:
Progressive algorithms reveal intermediate recognition states



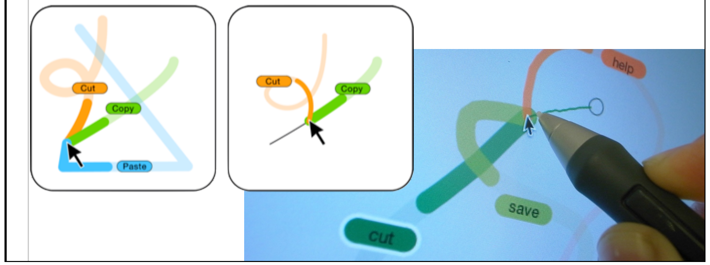
Octopocus: Learning complex gestures

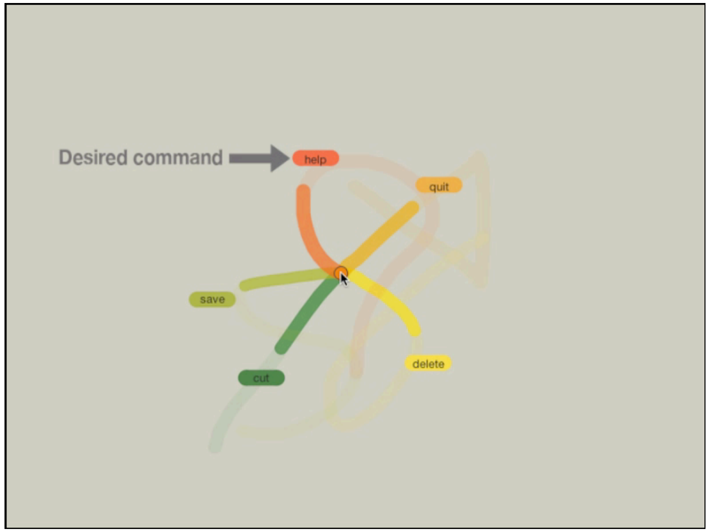
Experts *just do it*



Octopocus: Learning complex gestures

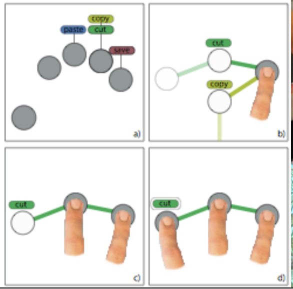
Experts *just do it*
Novices *hesitate* … which activates:
feedforward shows current available gestures
feedback shows what the recognizer sees





Arpege: Learning chords on a multi-touch surface

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

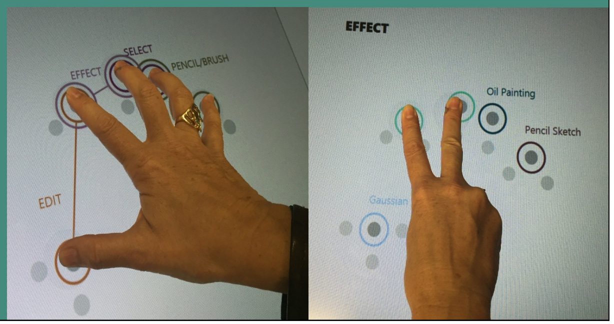


Arpege: Design and learning
of multifinger chord gestures

Submitted for review
CHI 2010

Dynachord: Combining chords and gestures

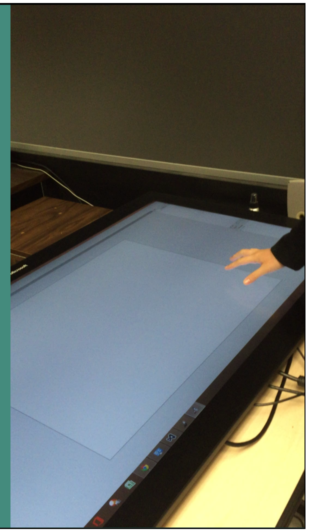
Chord sequences for a larger chord vocabulary
Dynamic adjustment of parameters



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

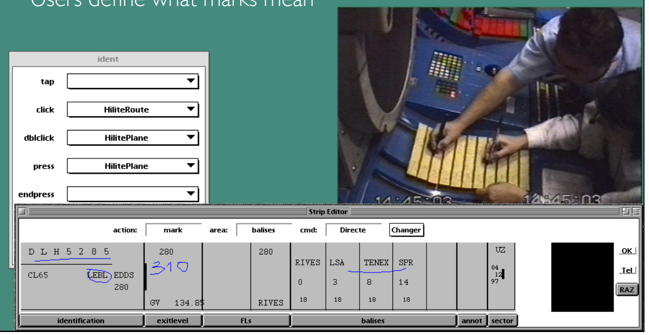
How can we help users appropriate technology ?

Creating a partnership in which the user defines the **semantics** of the interaction with the computer

Interaction Browser :	Linking marks to actions
Knotty Gestures :	Interacting while writing
Musink :	Creating a user-defined language
Façades :	User-reconfigurable interfaces

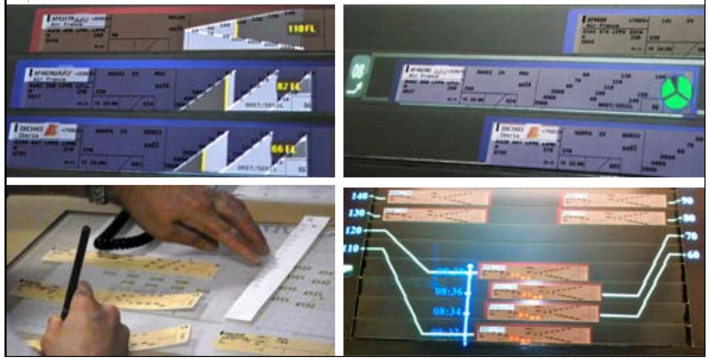
Interaction browser: User-defined commands

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



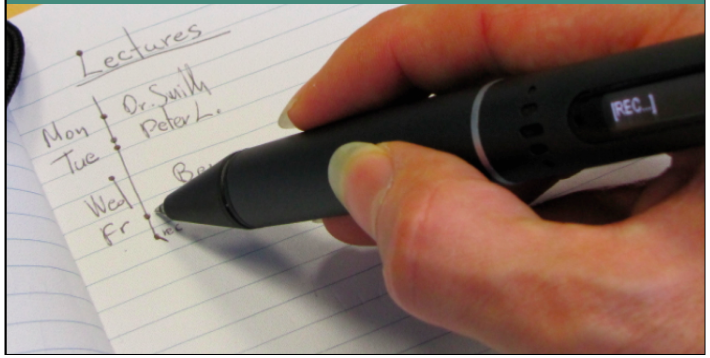
Striptic

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



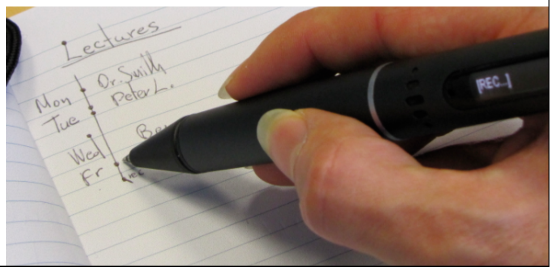
Knotty Gestures

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



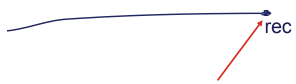
Knotty Gestures

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



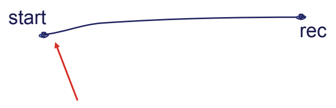
Knotty Gestures: Creating an interactive controller

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



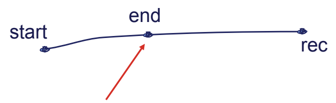
Choose "recording" to define the type of line

Knotty Gestures: Creating an interactive controller



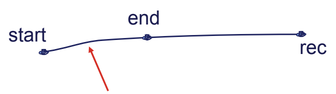
Define where the recording will start

Knotty Gestures: Creating an interactive controller



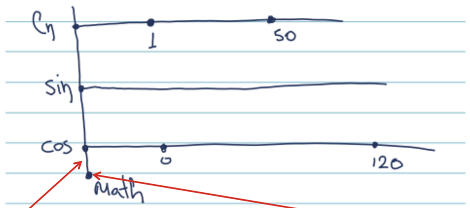
Define an end point for the recording

Knotty Gestures: Creating an interactive controller



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

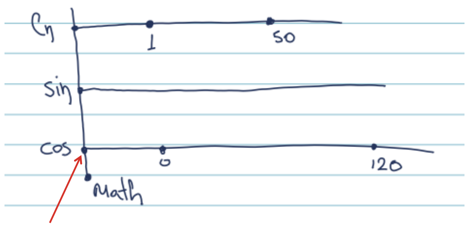
Drawing a Math Calculator



This line acts as a base for attaching mathematical value sliders

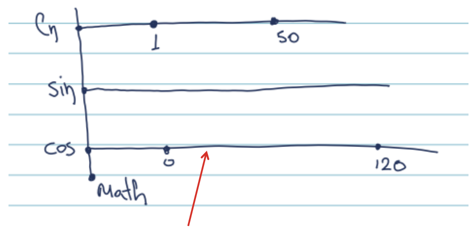
The knotty gesture at the end defines the type

Drawing a Math Calculator



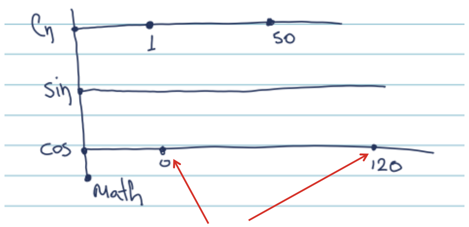
Any knot drawn on line lets the user select a mathematical function

Drawing a Math Calculator

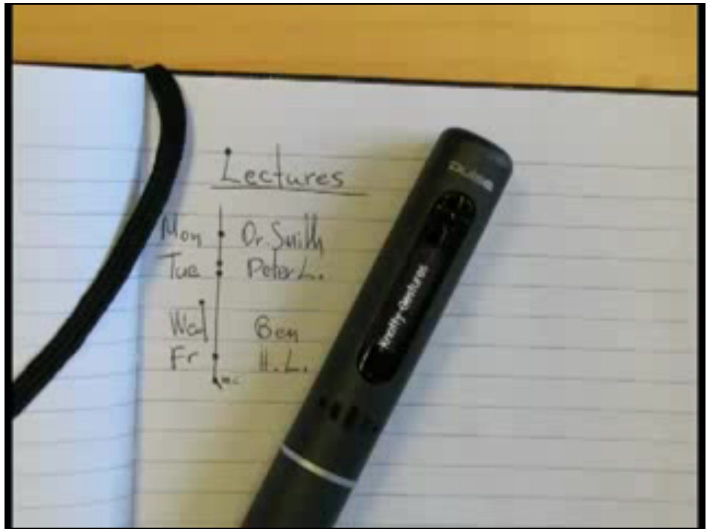


The extensions act as value controllers
Sliding the pen over the line moves through range of function values, shown on the pen display

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 Mus*ink* support the creative process?
- What are the design implications for Mus*ink* v2?

Semi-Structured Delayed interpretation

Key insights:

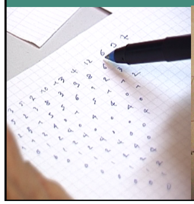
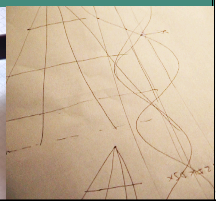
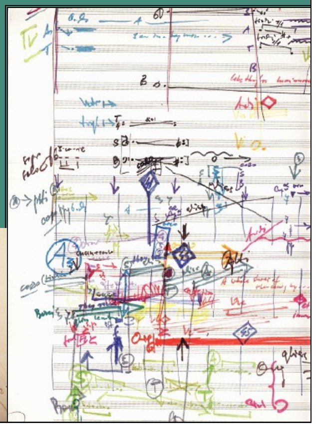
Spatial structure on paper
- improves recognition
- under user's control

Recognition need not be immediate
- users decide *when* to intrepret
- interpretation *changes over time*

Musink

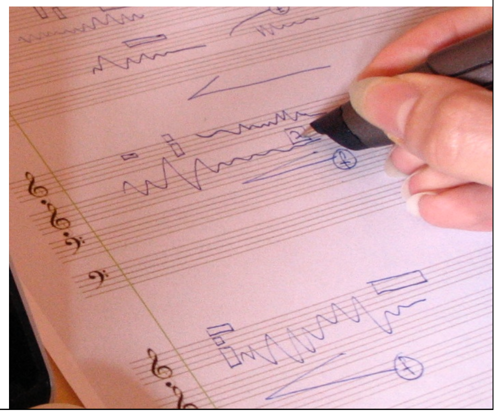
Musicians create their own musical languages on paper

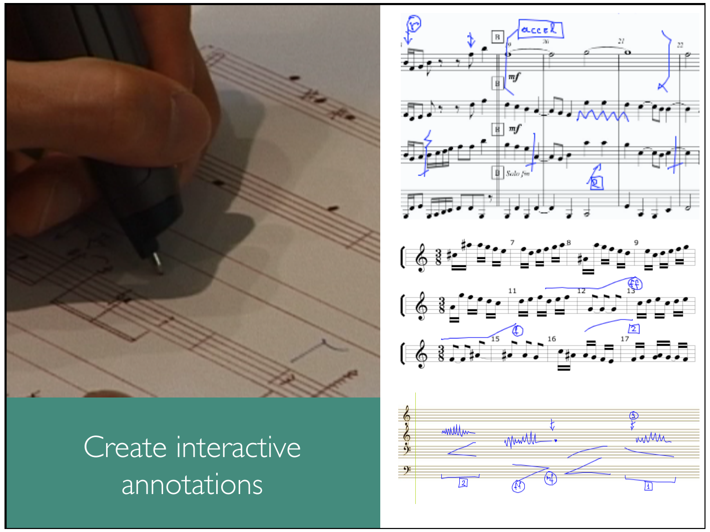
… and go back and forth between paper and computer

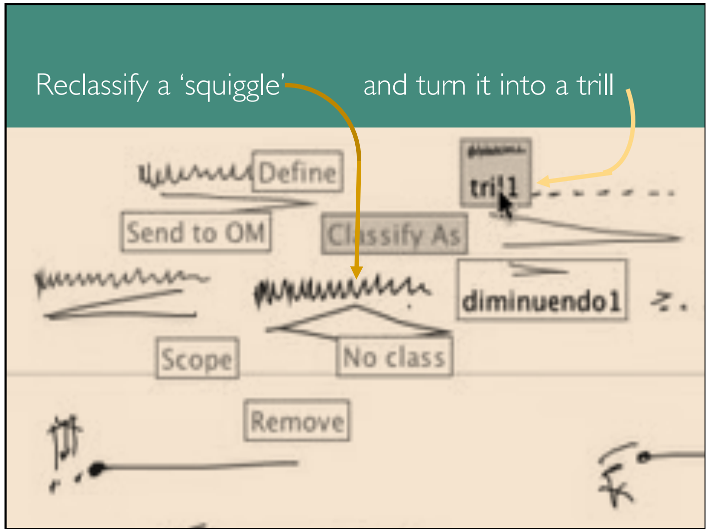




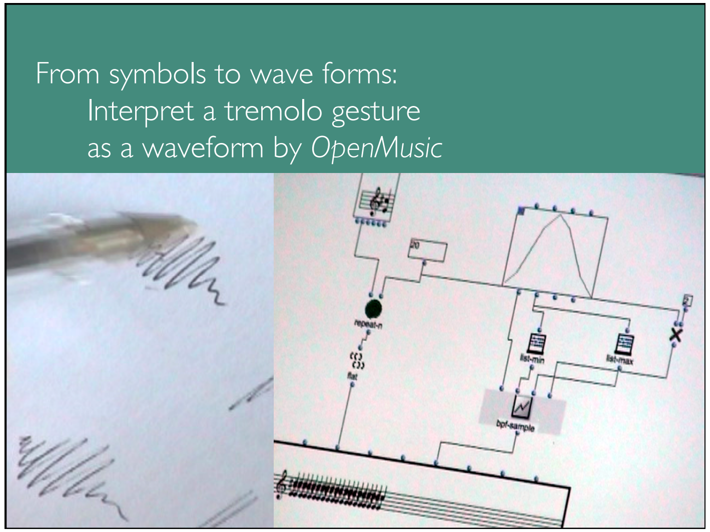
Mus*Ink* Define meaning of gestures over time

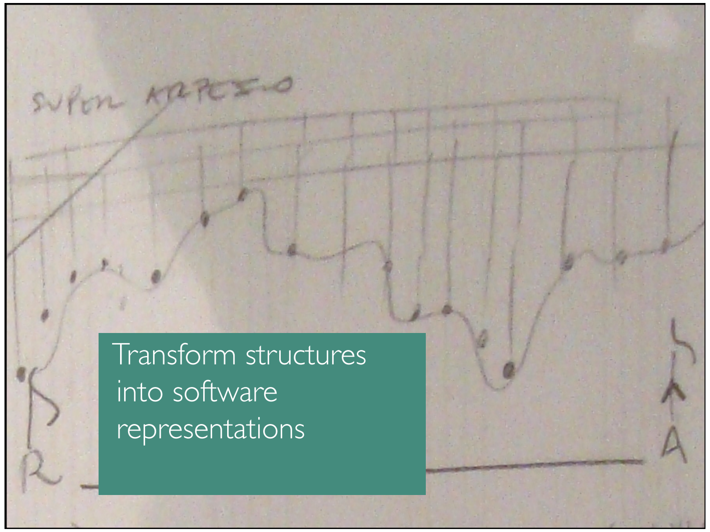
User decides if and when to interpret each gesture

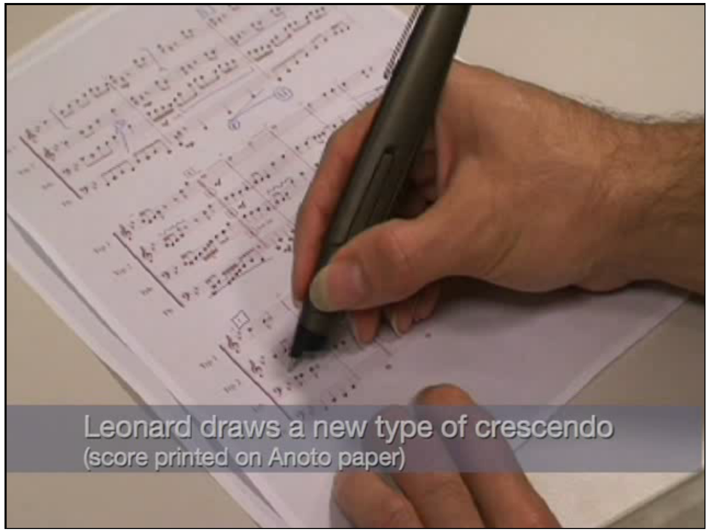



Create interactive
annotations


Reclassify a 'squiggle'
and turn it into a trill
Define
Send to OM
Classify As
trill1
diminuendo1
Scope
No class
Remove

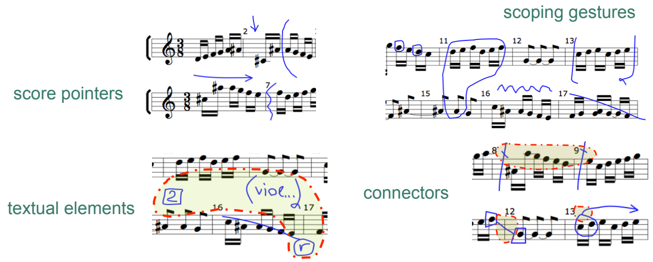

From symbols to wave forms:
Interpret a tremolo gesture
as a waveform by *OpenMusic*


Transform structures
into software
representations



Musink: Semi-structured, delayed interpretation

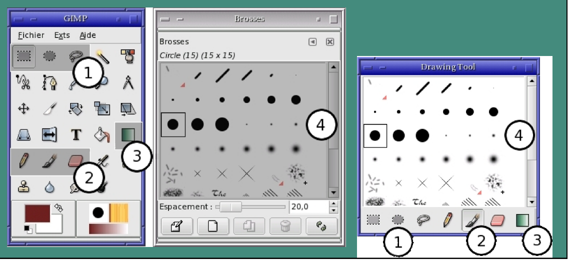
Users decide when and how each annotation should be interpreted by the computer



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



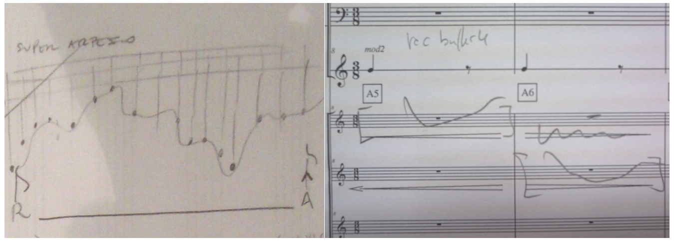
Substrates

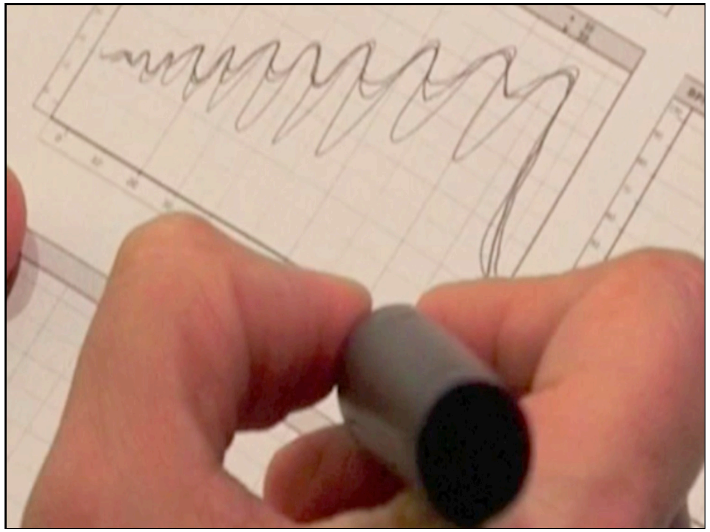
Define the structures and rules

Ways to interpret the data

Different structures

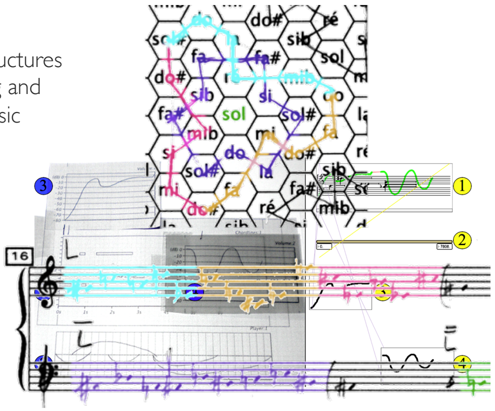
to facilitate
interpretation

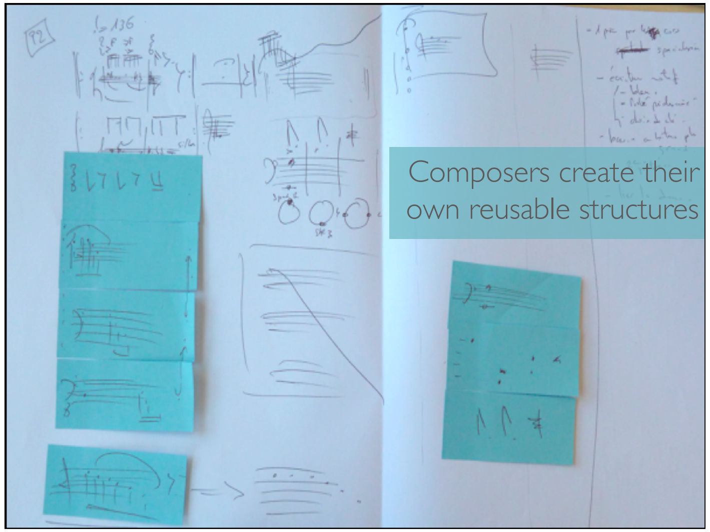


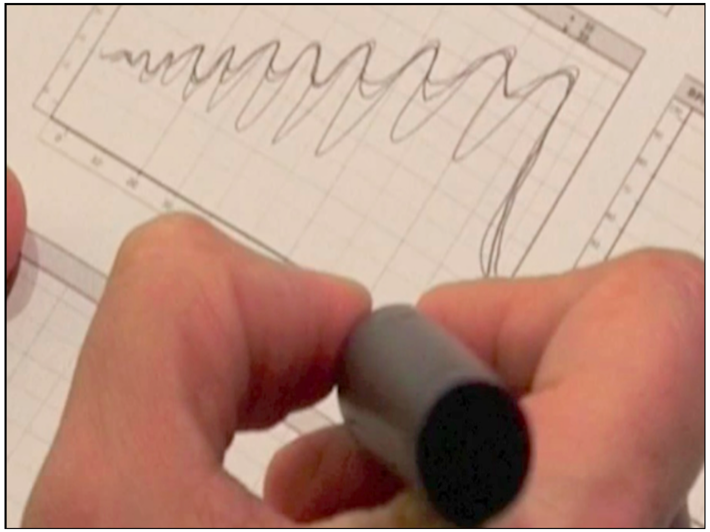


Paper Substrates: create own language & structure

Composers
create new structures
for interpreting and
composing music

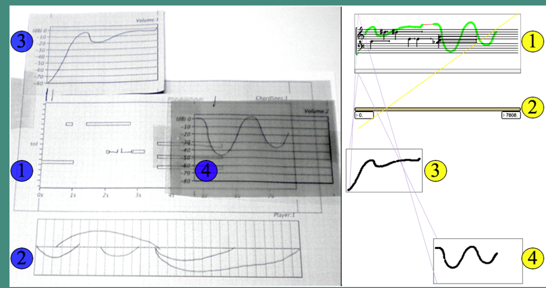




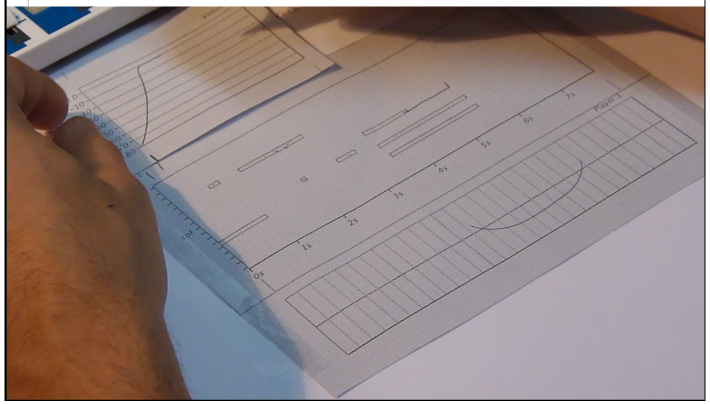


Paper Substrates

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

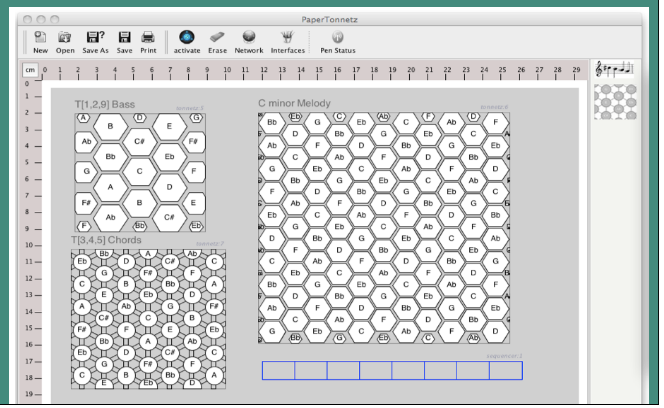


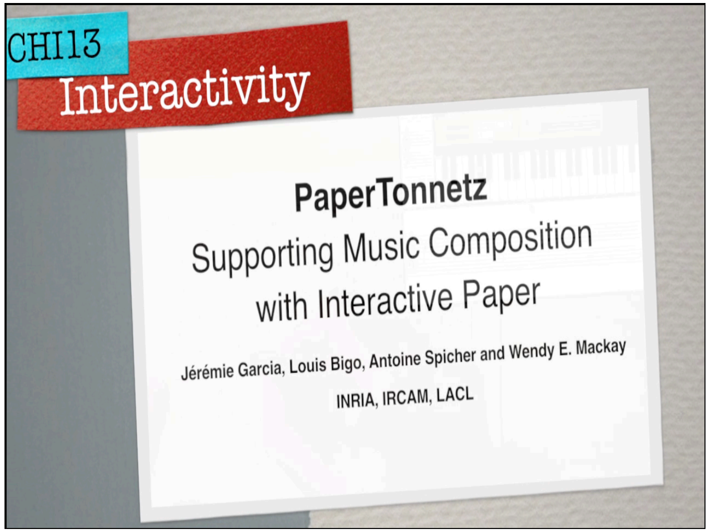
Interactive Paper Substrates for music composers

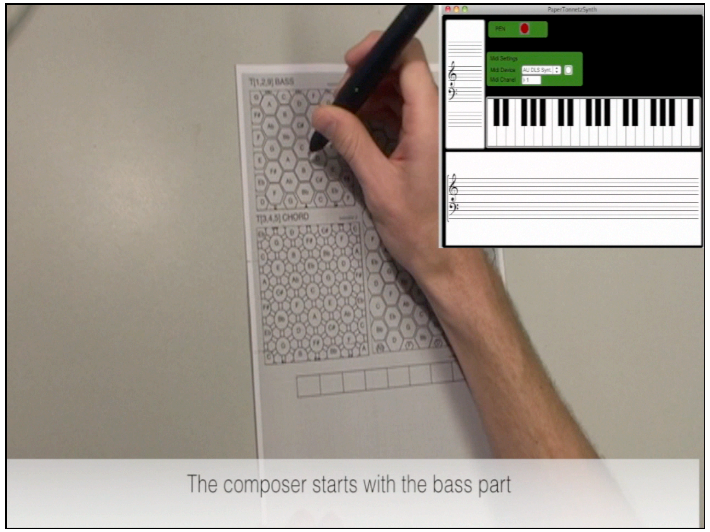


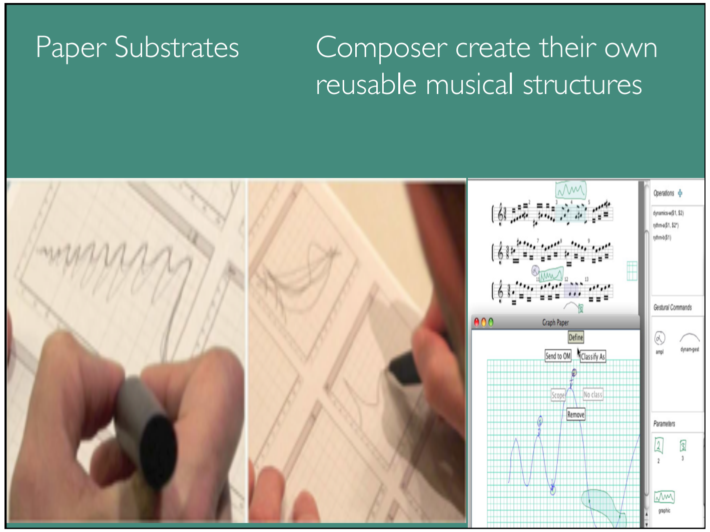
Paper Tonnetz

Draw music based on musical relationships among pitches



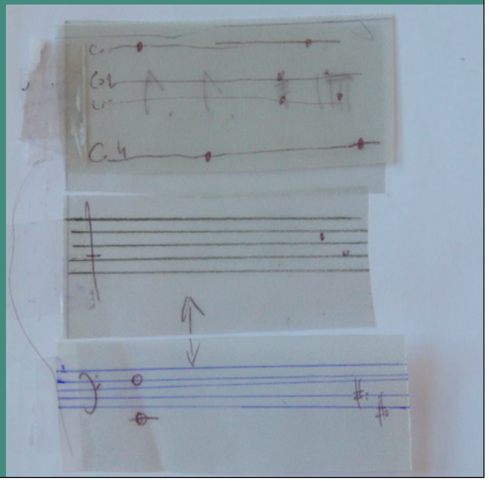

CHI13
Interactivity
PaperTonnetz
Supporting Music Composition
with Interactive Paper
Jérémie Garcia, Louis Bigo, Antoine Spicher and Wendy E. Mackay
INRIA, IRCAM, LACL


The composer starts with the bass part


Paper Substrates
Composer create their own
reusable musical structures


RULE
establish
relationships
among them

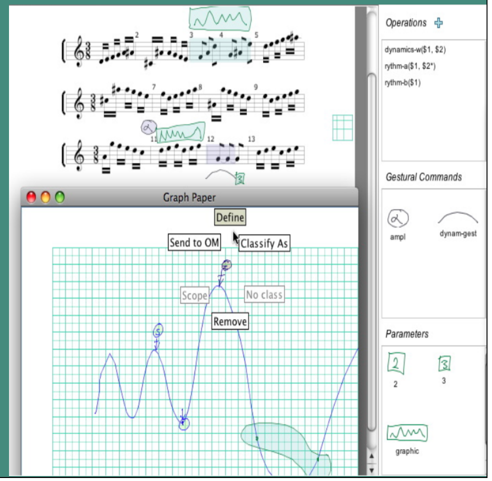
Arrange
and
Link
substrates



Arrange
and
Link
substrates

to

composition
software



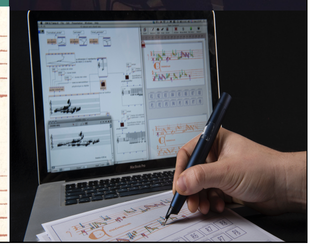
Interactive Paper Substrates
to Support Musical Creation

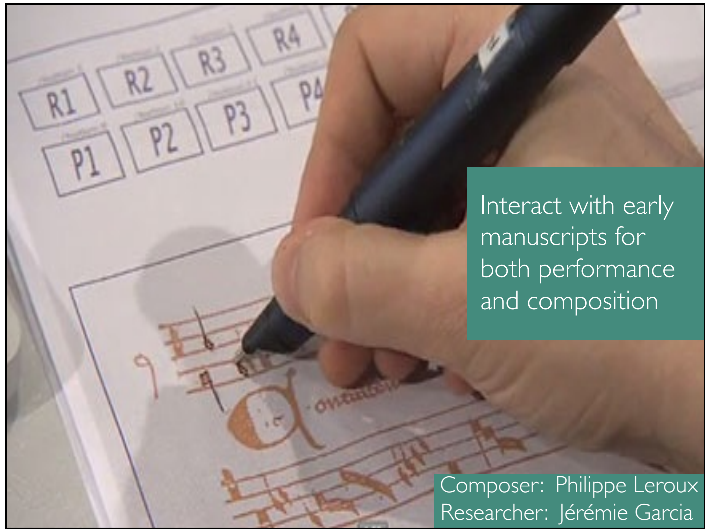
Jérémie Garcia, Theophanis Tsandilas, Carlos Agon & Wendy E. Mackay

INRIA, Université Paris-Sud, CNRS, IRCAM & Stanford University

Quid Sit Musicus
Philippe Leroux

13th century musical scores
Each note indicates expression





Interact with early manuscripts for both performance and composition

Composer: Philippe Leroux
Researcher: Jérémie Garcia

Quid Sit Musicus
by Philippe Leroux

QUID SIT MUSICUS?
BY PHILIPPE LEROUX

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