Computational Instruments:

Concept-level Tools for Collaboration with Intelligent Interactive Systems
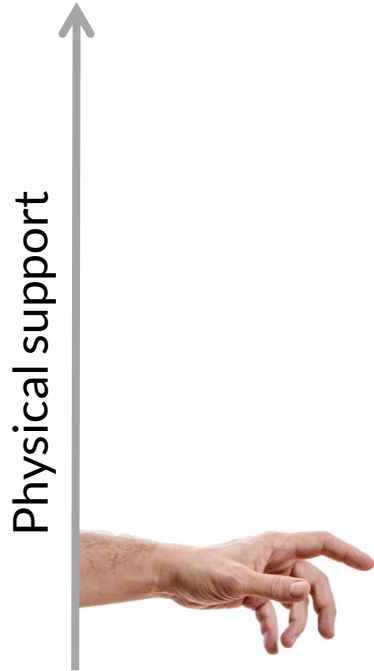
Daniel Buschek
LMU Munich
April, 2018

Computational Instruments

Tools, which reify roles of contributing intelligence to a task
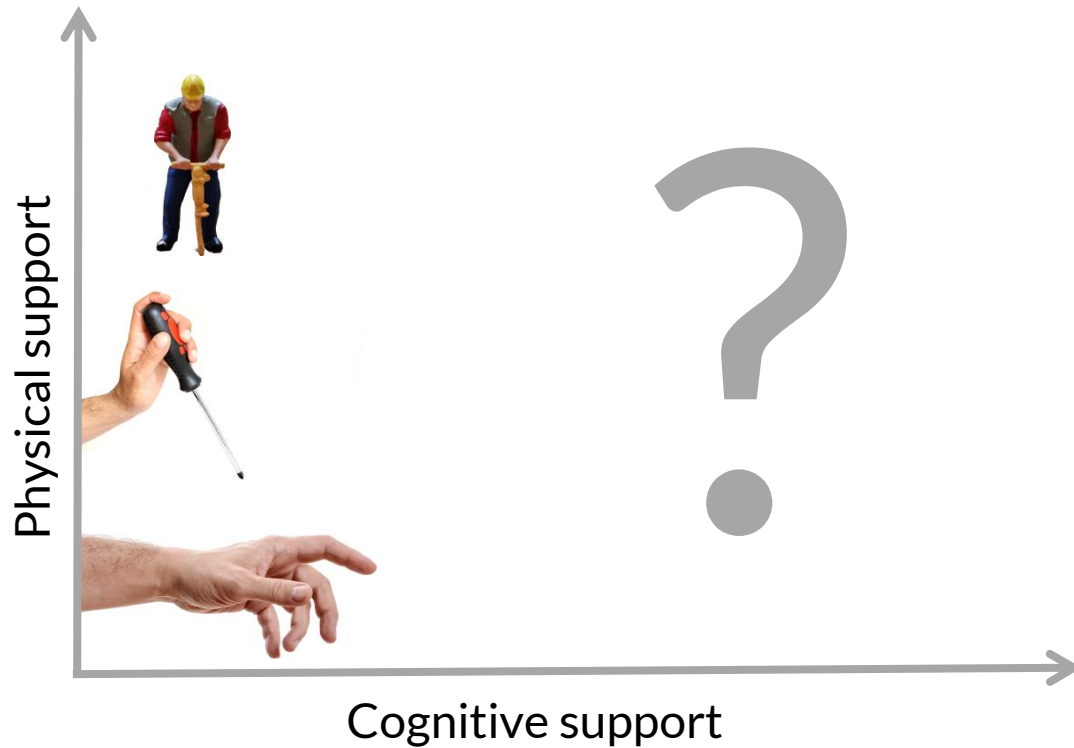
A very brief history of tools

Physical support

A very brief history of tools

Physical support

A very brief history of tools

Physical support

A very brief history of tools

Physical support (vertical axis)

Cognitive support (horizontal axis)

From task to concept-level tools
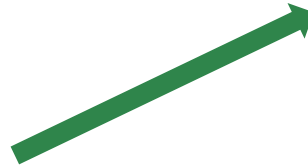
A tool for **hitting**
used to execute subtasks
when building something

From task to concept-level tools

A tool for **building**
used to execute the
"role of the builder"

A tool for **hitting**
used to execute subtasks
when building something

https://pixabay.com/de/hand-hammer-werkzeug-h%C3%A4nde-halten-2620237/

Roles of contributing intelligence to a task

Divergence

Inspiration
What to work on?

Proposition
How might we do it?
Concrete ideas?

Roles of contributing intelligence to a task

Divergence

Inspiration
What to work on?

Proposition
How might we do it?
Concrete ideas?

Iteration

Feedback
Where are we?
Right direction?

Refinement
What more to do?

Roles of contributing intelligence to a task

Divergence

Inspiration
What to work on?

Proposition
How might we do it?
Concrete ideas?

Iteration

Feedback
Where are we?
Right direction?

Refinement
What more to do?

Convergence

Evaluation
How good is it?

Decision
What to accept?

Example

Filler CI

A general purpose auto-complete tool
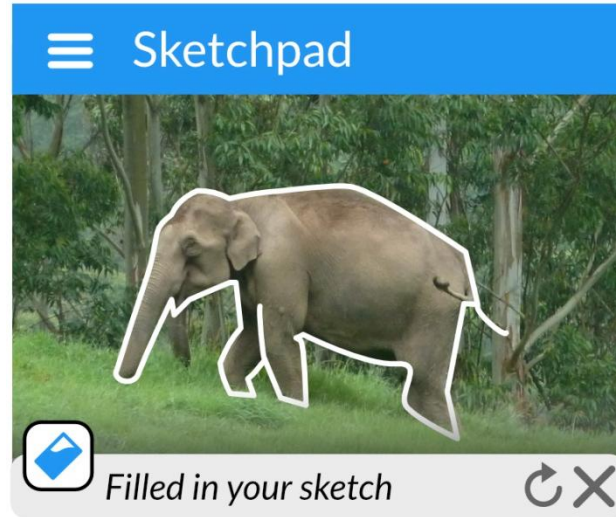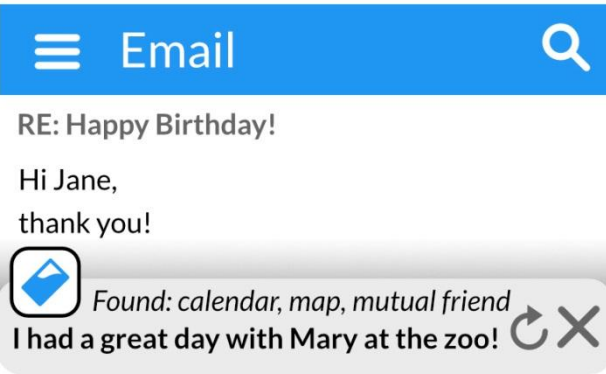
Example: A general purpose auto complete tool

Example: A general purpose auto complete tool

Example: A general purpose auto complete tool
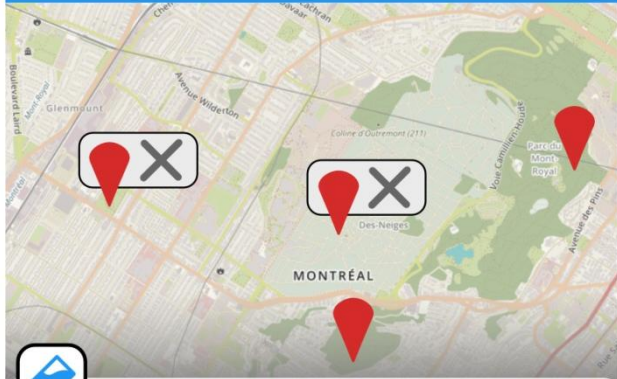
Example: Settings & control?

Questions & Challenges

1. How can users **anticipate** what a CI might do to an object?

Questions & Challenges

1. How can users **anticipate** what a CI might do to an object?

2. How can they **control/guide** CIs beyond one-click "magic sauce" implementations?

Questions & Challenges

1. How can users **anticipate** what a CI might do to an object?

2. How can they **control/guide** CIs beyond one-click "magic sauce" implementations?

3. How can CIs be **integrated** into GUIs, in particular w.r.t. appropriation?

Questions & Challenges

1. How can users **anticipate** what a CI might do to an object?

2. How can they **control/guide** CIs beyond one-click "magic sauce" implementations?

3. How can CIs be **integrated** into GUIs, in particular w.r.t. appropriation?

4. How can CIs **learn** with continued use?

CIs support:

- **Appropriation**, by representing recurring abstract concepts

- **Partnership**, by contributing intelligence via collaboration in specific limited roles

- **Varying degrees of control**, by allowing users to flexibly choose and chain CIs with different roles
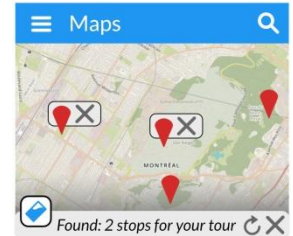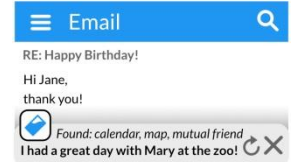
Computational Instruments: Concept-level Tools for Collaboration with Intelligent Interactive Systems

daniel.buschek@ifi.lmu.de