Design Tools

CS 347

Maneesh Agrawala

Announcements

Quiz 2 is a week from today—covers four lectures

- Design Cognition
- Design Process
- Design Tools (today)
- Social Media (Monday)

Last time

The design process is a set of structured activities meant to address problems in how we generate and develop ideas

These activities are well attuned to solving **wicked problems**, which feature contradictory or conflicting goals

Participatory design is a movement to decenter the designer's power in these activities. We still struggle to achieve our goals here.

Design patterns help us avoid reinventing the wheel.

We can intervene on these activities to address shortcomings in how we practice design: e.g., **parallel prototyping** and **comparing multiple designs** (to reduce **design fixation** and **demand characteristics**)

Today

Schön's reflective practitioner

Empowering design reflection through tools and technology

- **Ideation:** low-fidelity input
- **Implementation:** rapid construction
- **Evaluation:** feedback

…with examples of each

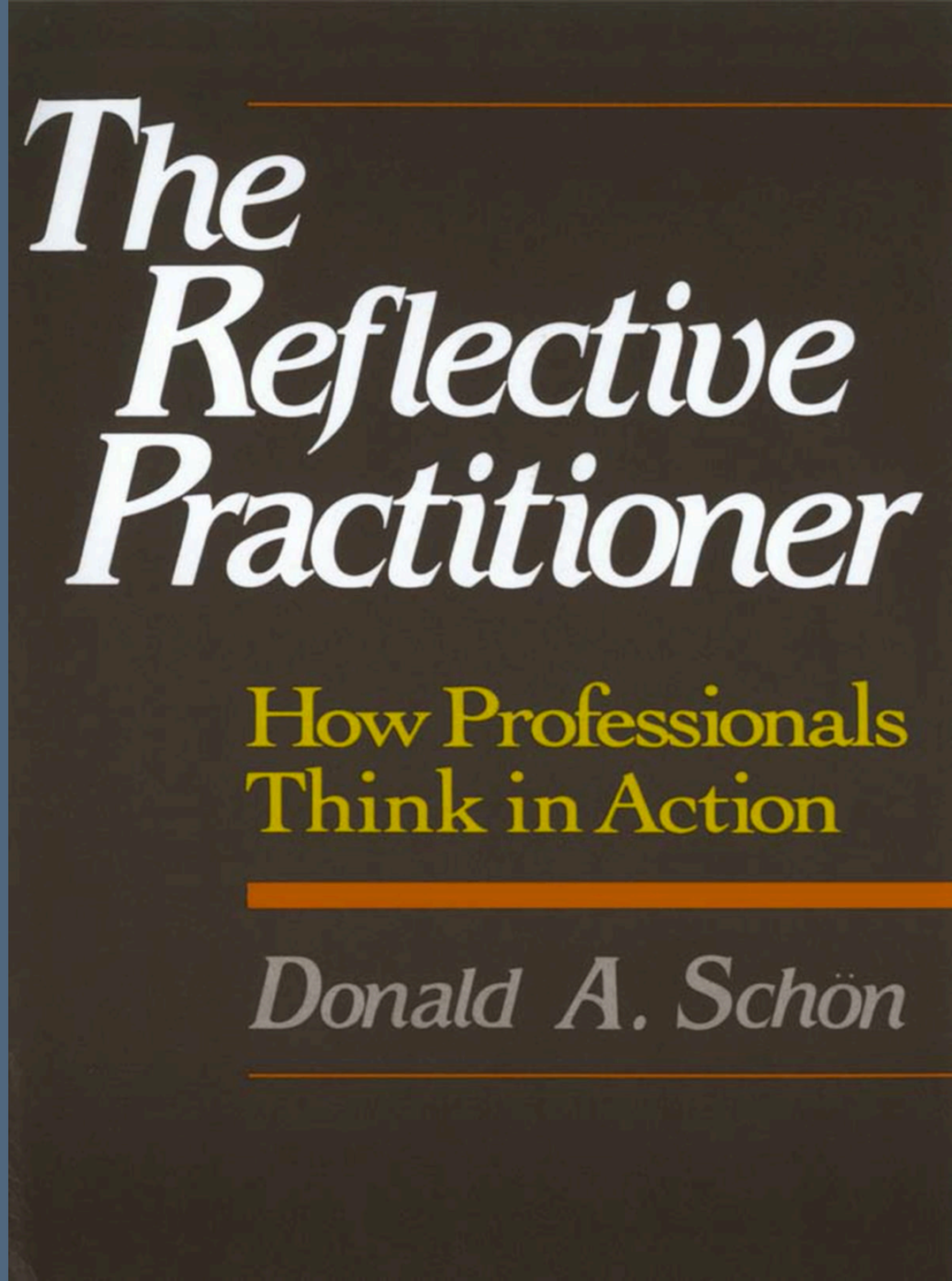
YOU READ THIS

The Reflective Practitioner

Reflective practitioner

How does design work? Why does it work?

Donald Schön [1984] studied a variety of professionals, including designers, and articulated a theory of the how and the why that has remained influential.

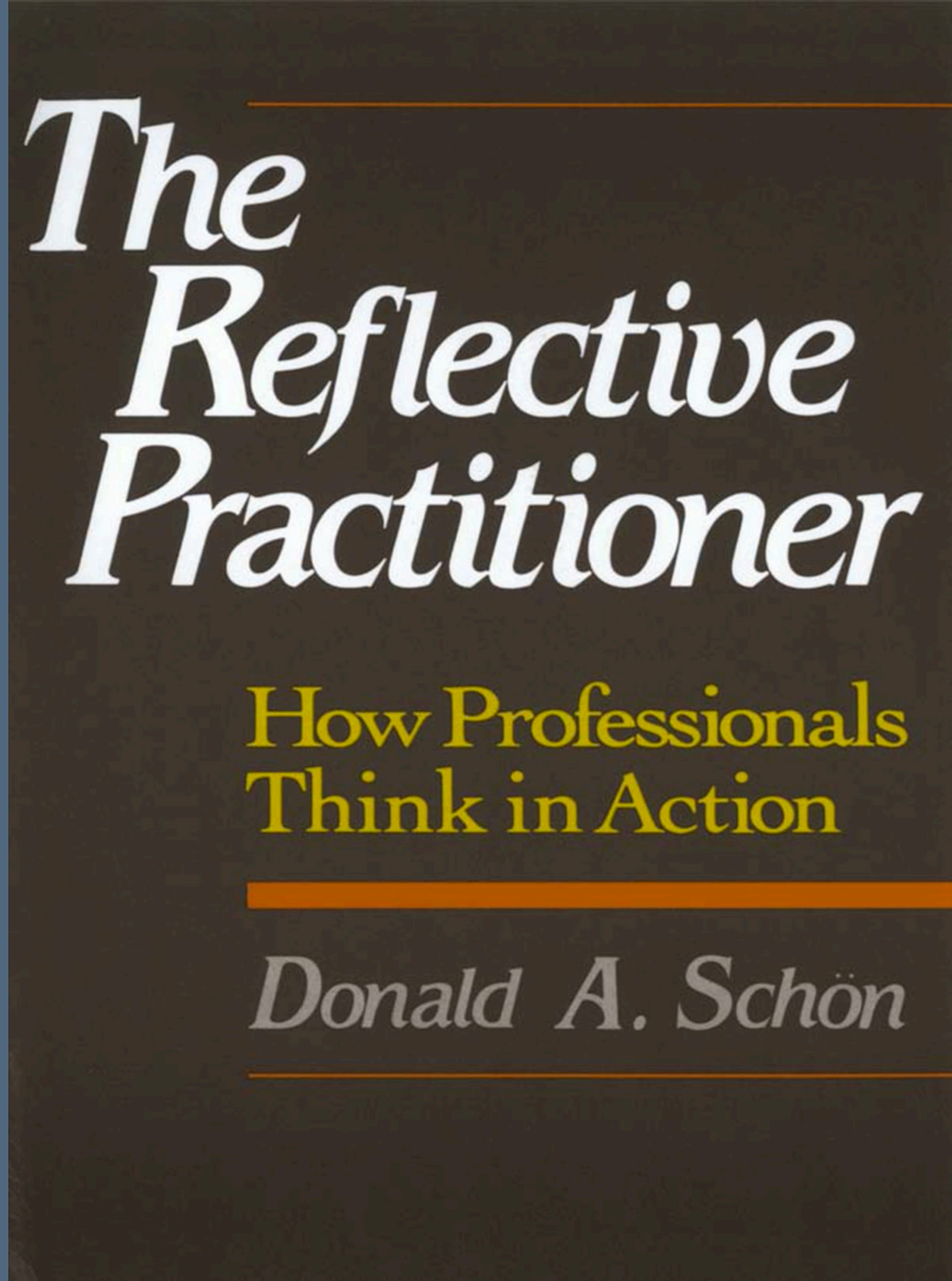


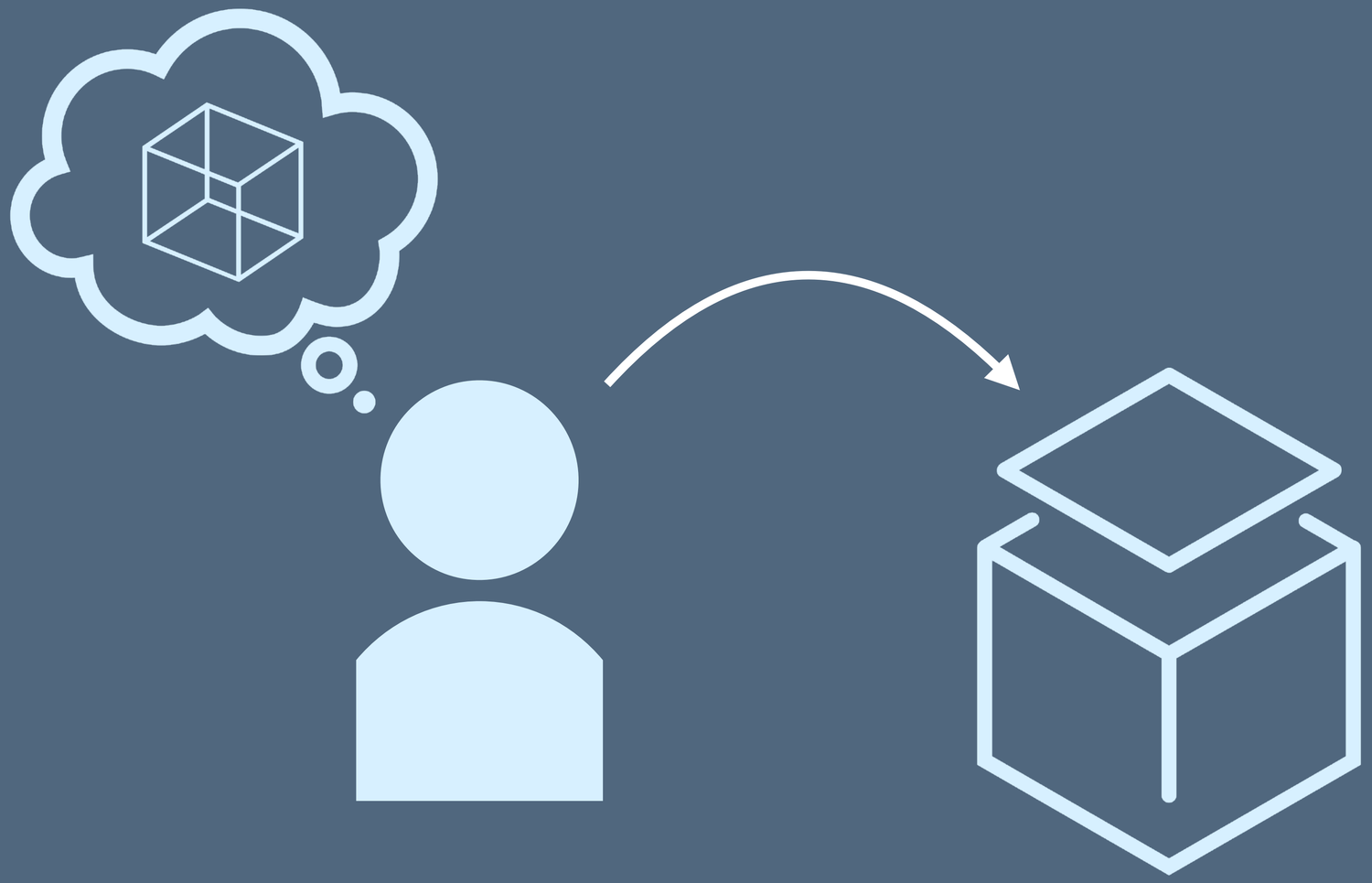
Reflective practitioner

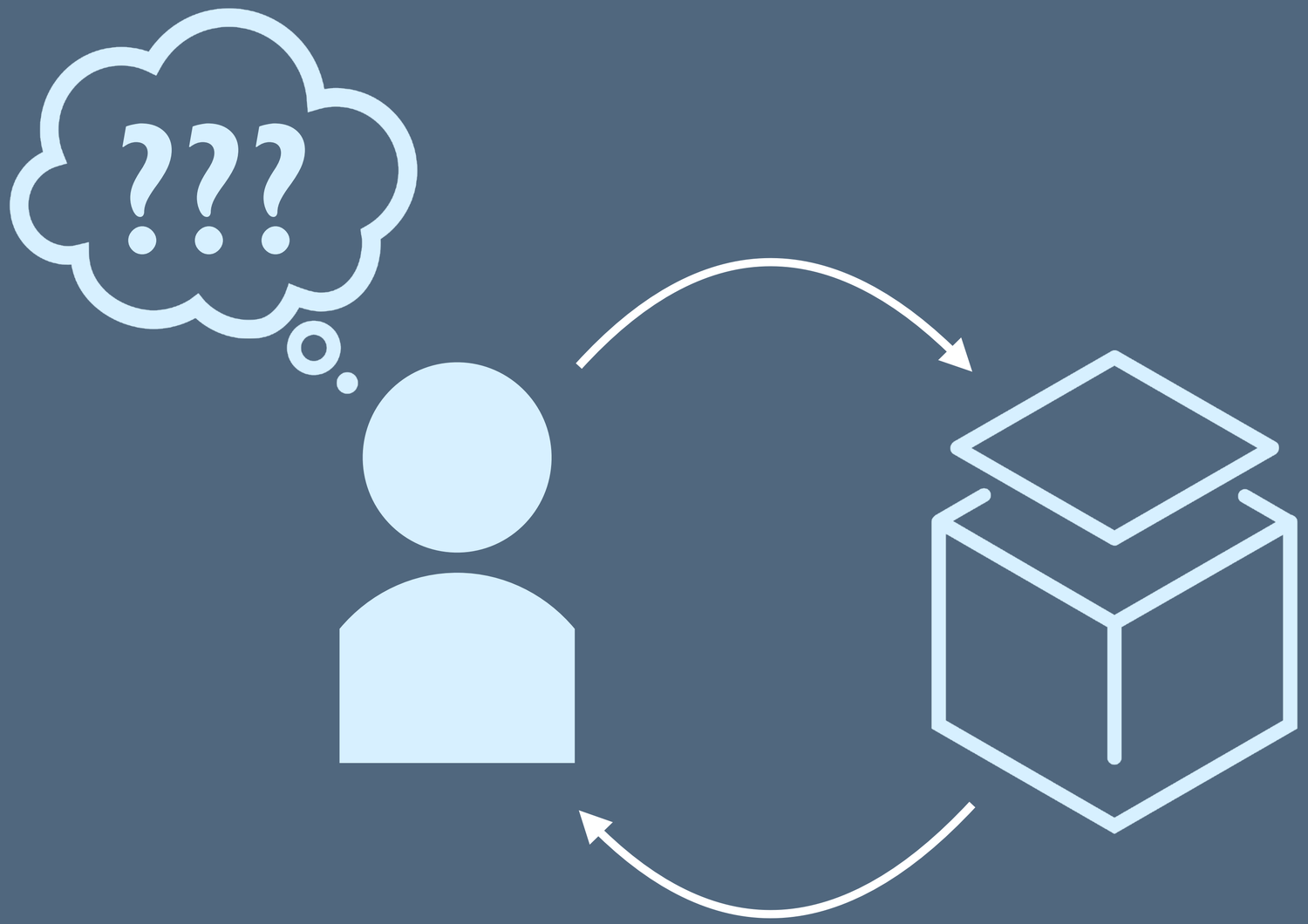
Design is not a "plan, then do" praxis

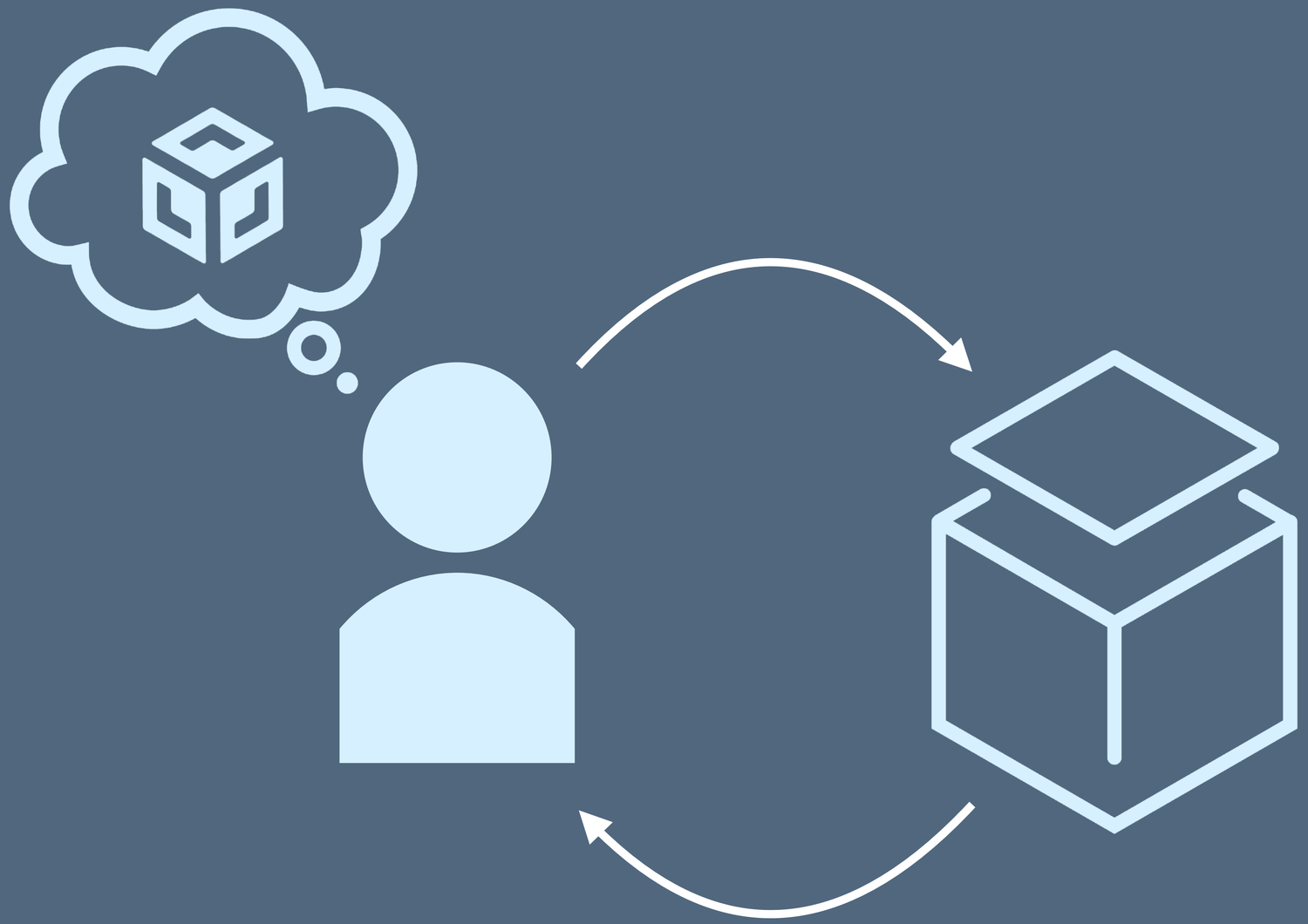
Instead, the designer is engaged in an ongoing conversation with the design

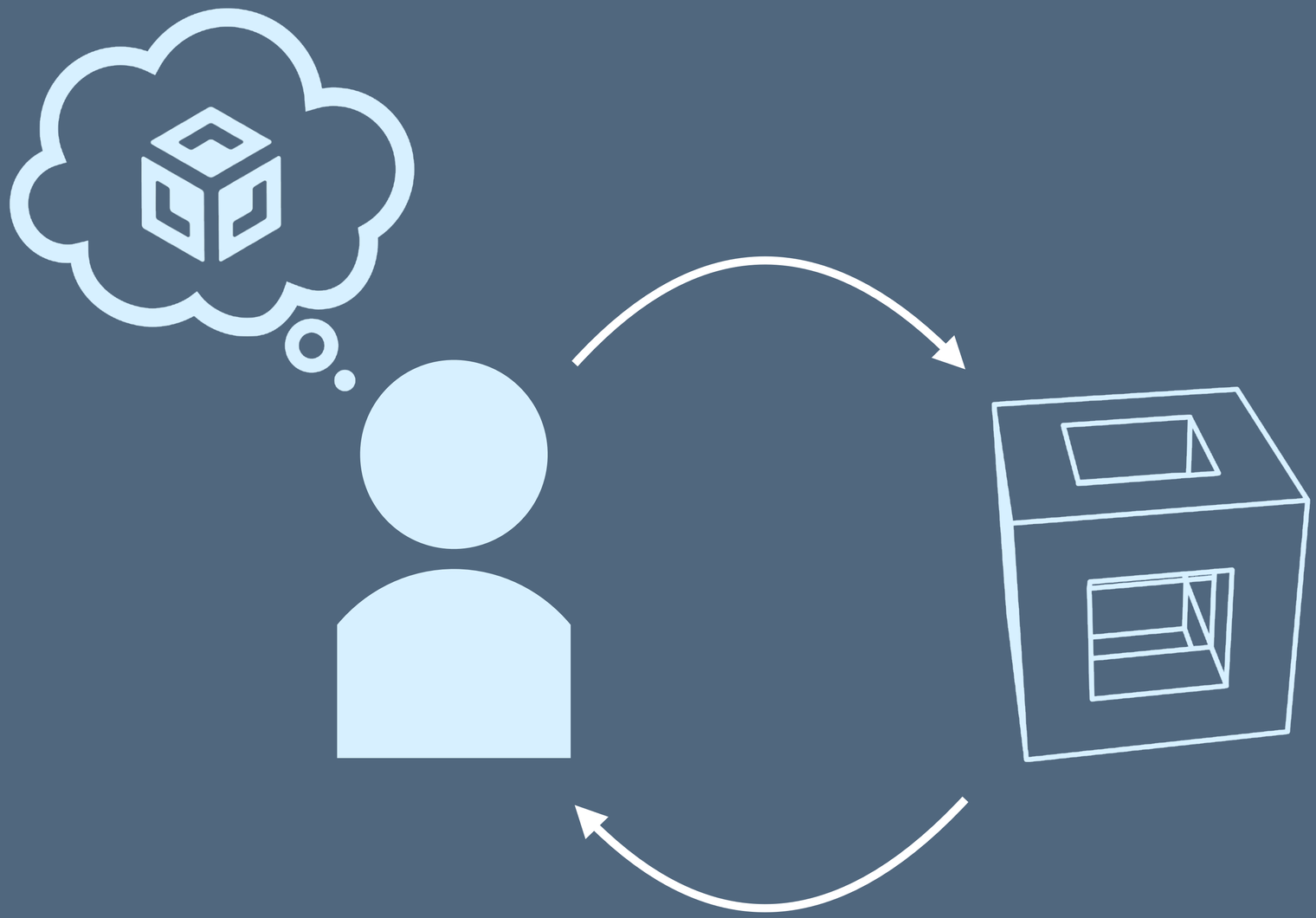
Critically, it's only by **observing the result of the doing can the designer engage in reflection**, allowing them to improve



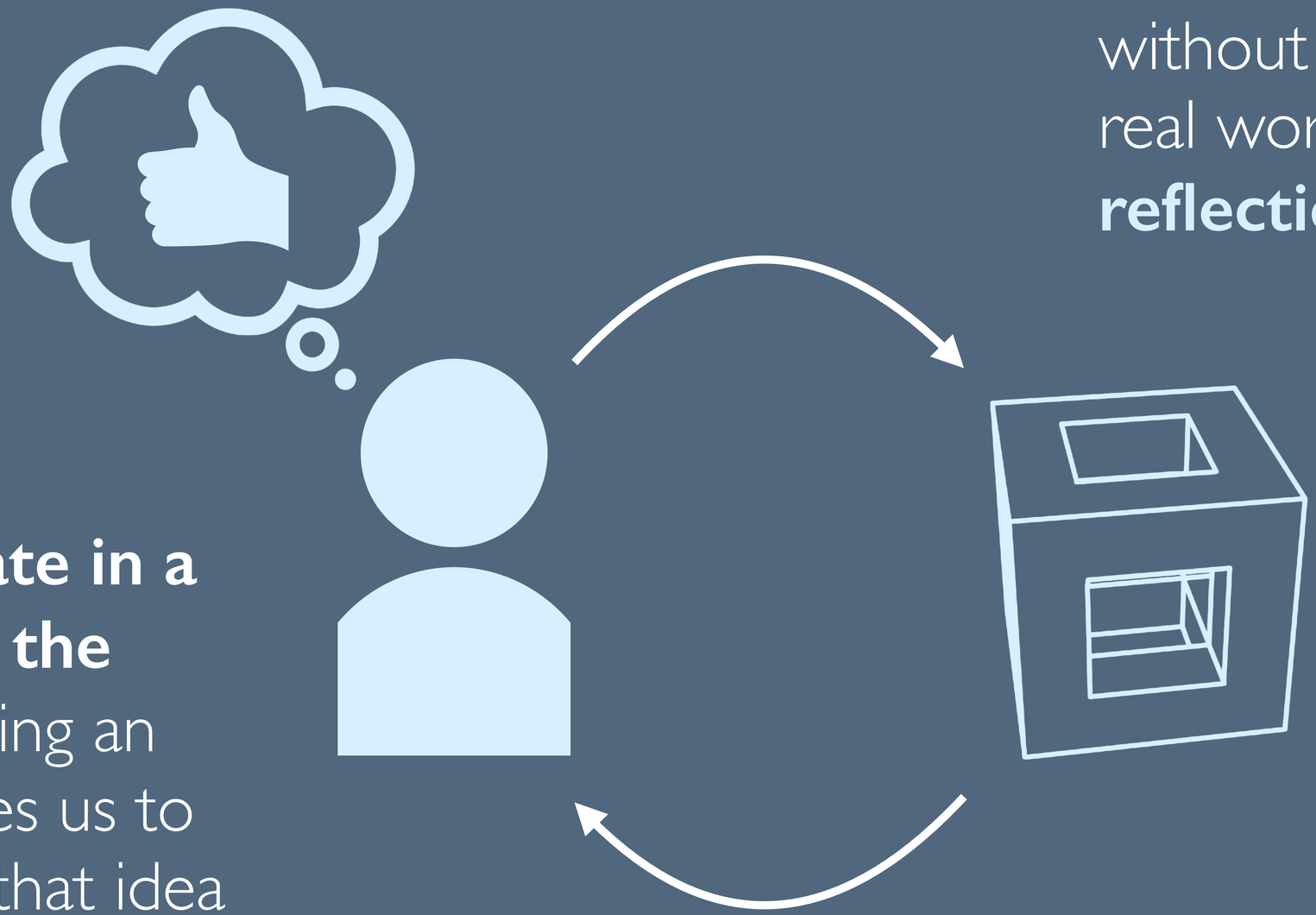








We learned something that we couldn't have without testing it in the real world. Schön calls this **reflection-in-action**

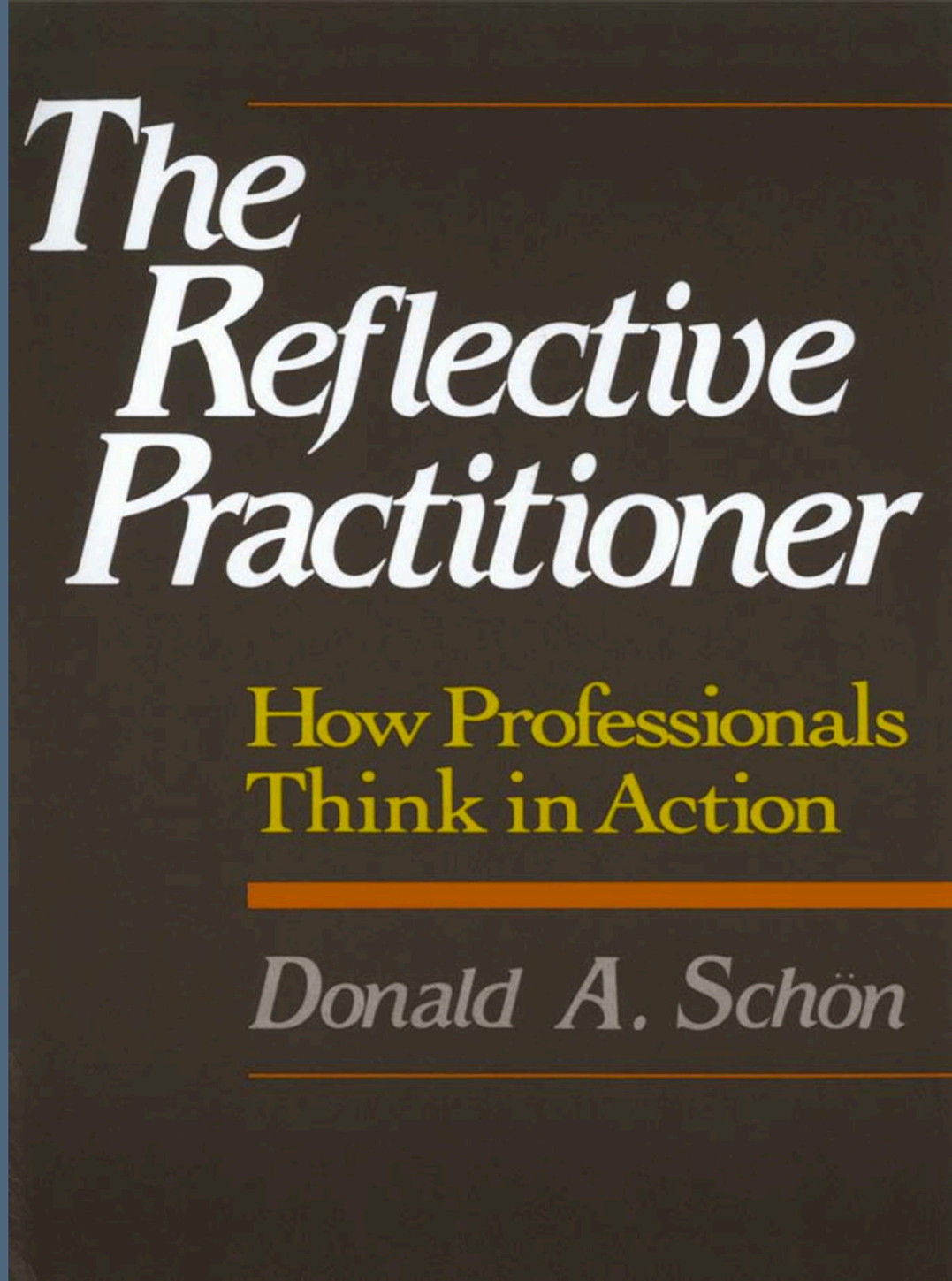


We operate in a loop with the world: trying an idea enables us to reflect on that idea and improve it

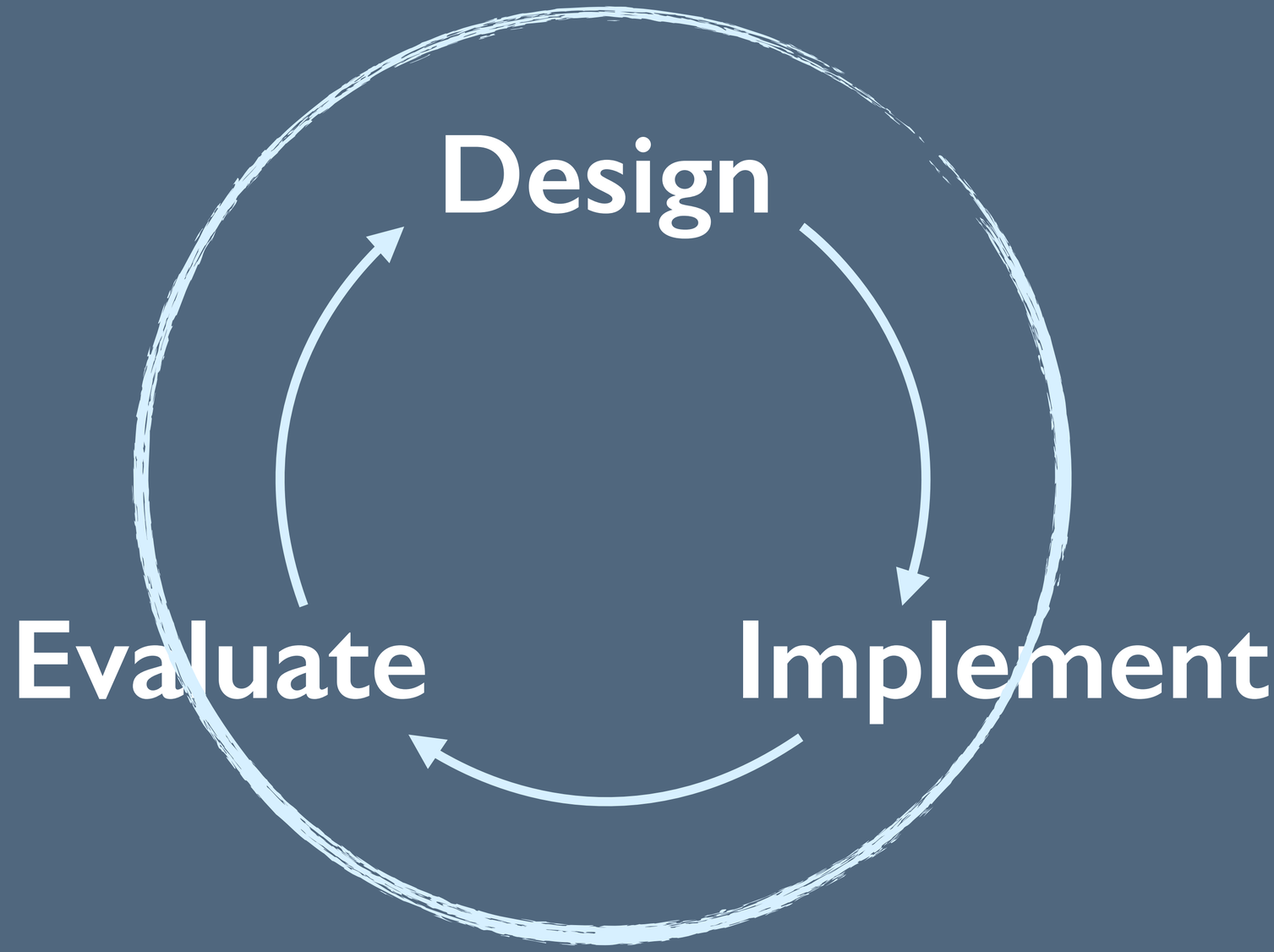
Implication

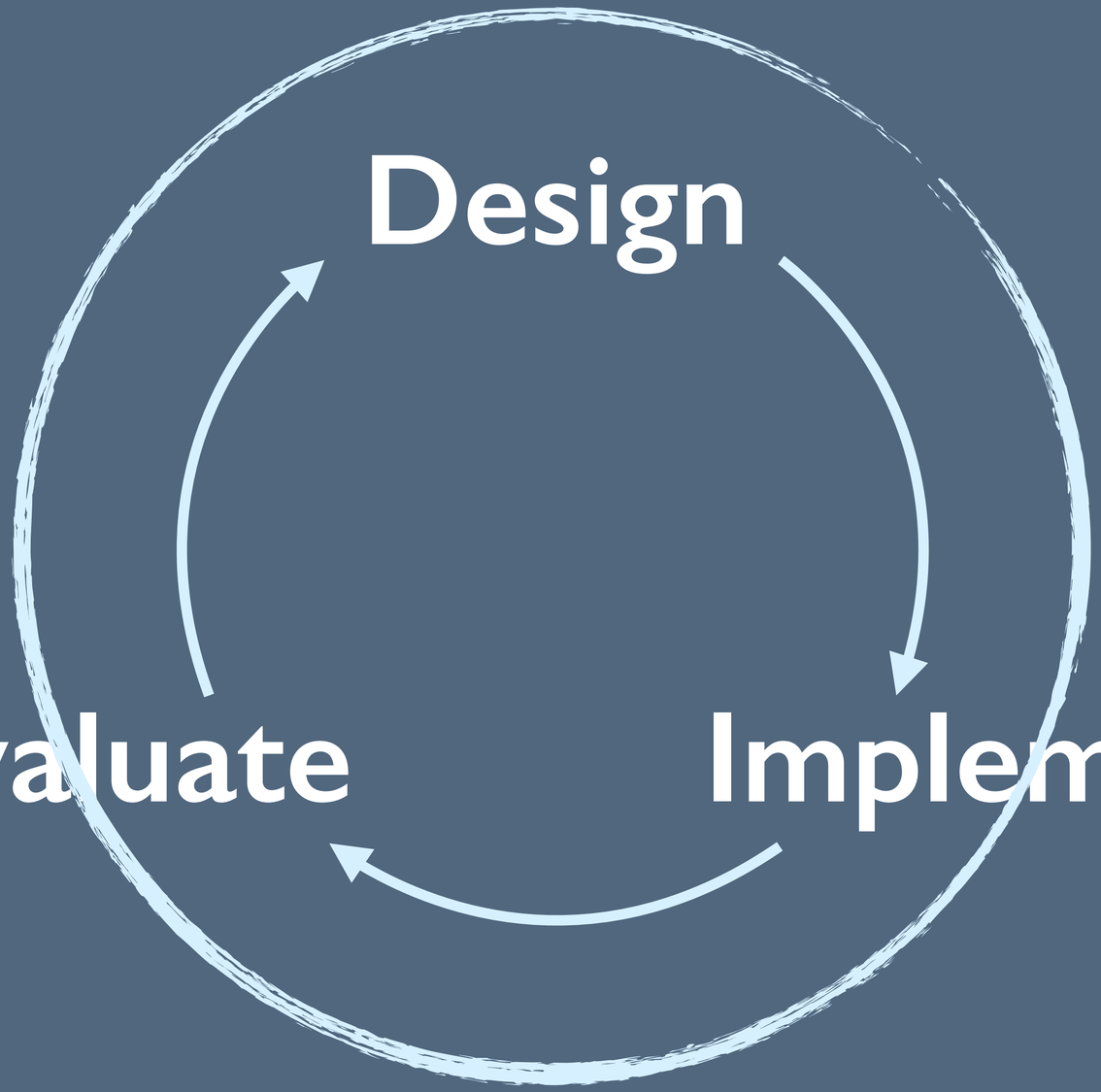
To improve the process:
encourage more rapid reflection, or
improve the quality of the reflection

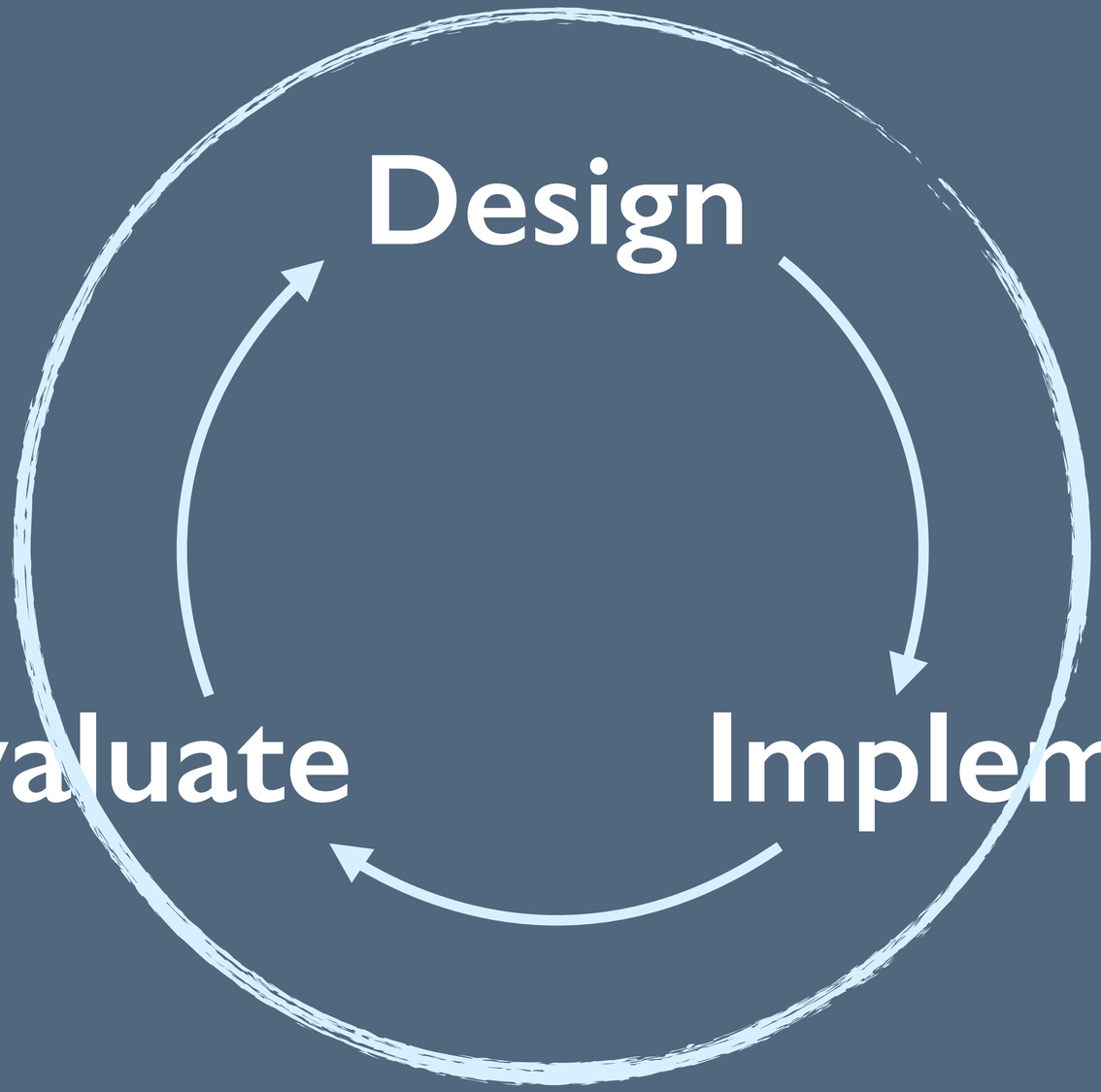
To improve the tools:
create alternatives that make reflection
easier to do or more informative

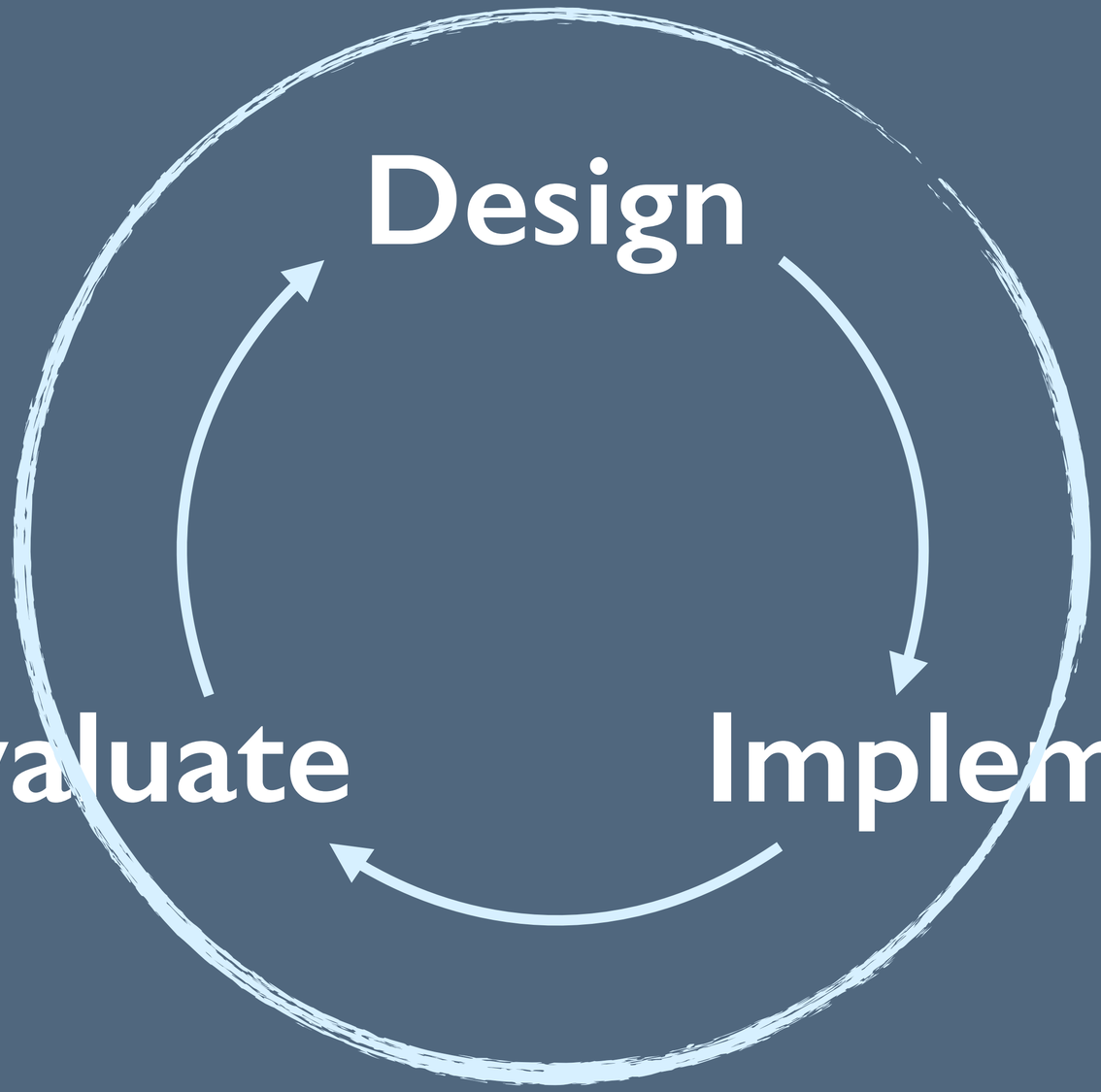


The tighter we can tune this loop…

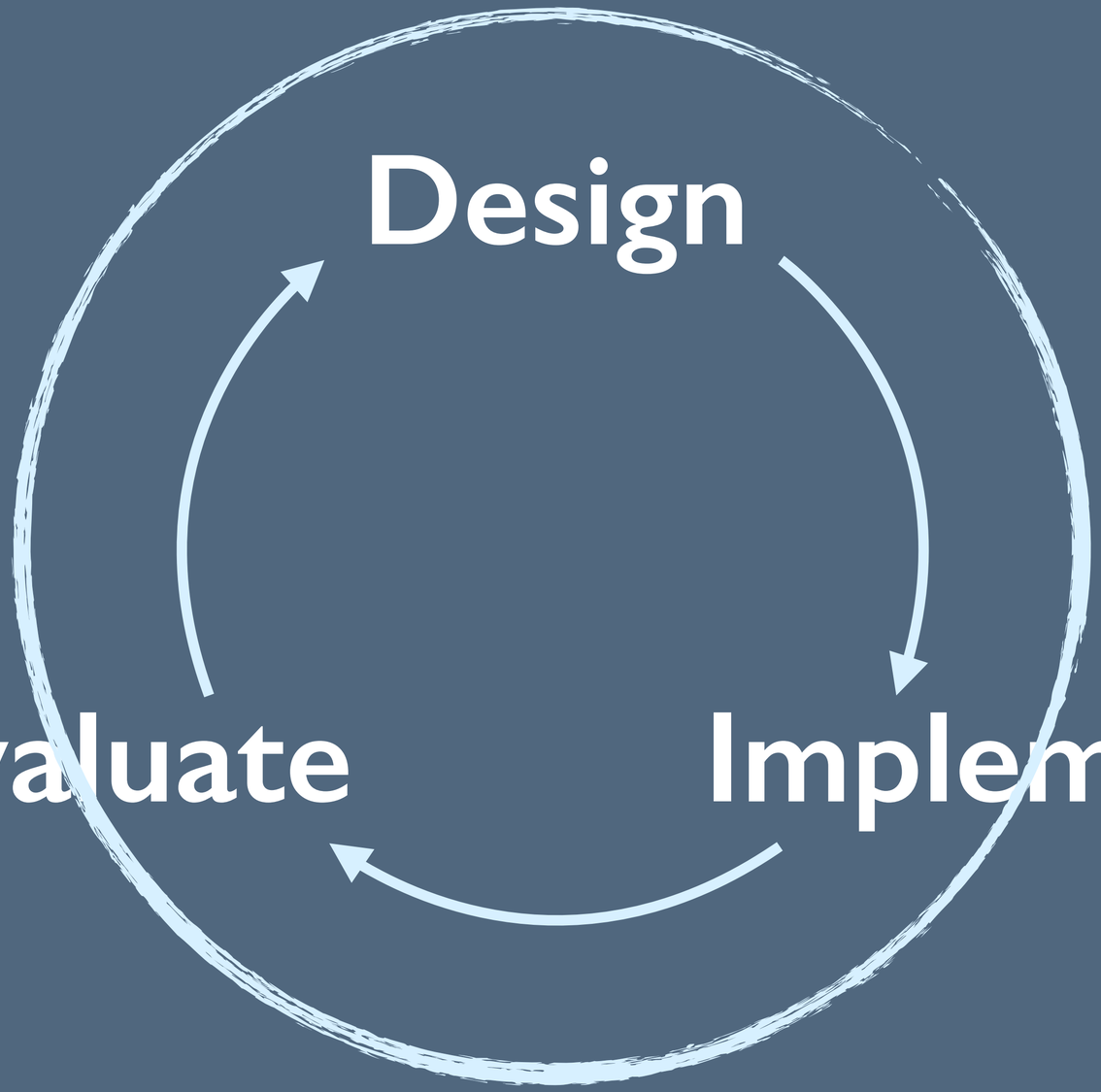



Design
Implement
Evaluate


Design
Implement
Evaluate


Design
Implement
Evaluate

…the more reflection we are doing, and the better our designs are.



Design tools improve reflection-in-action.

To create a design tool, look for a part of the reflection loop that feels loose—where reflection is slow or difficult—and tighten that part of the loop.

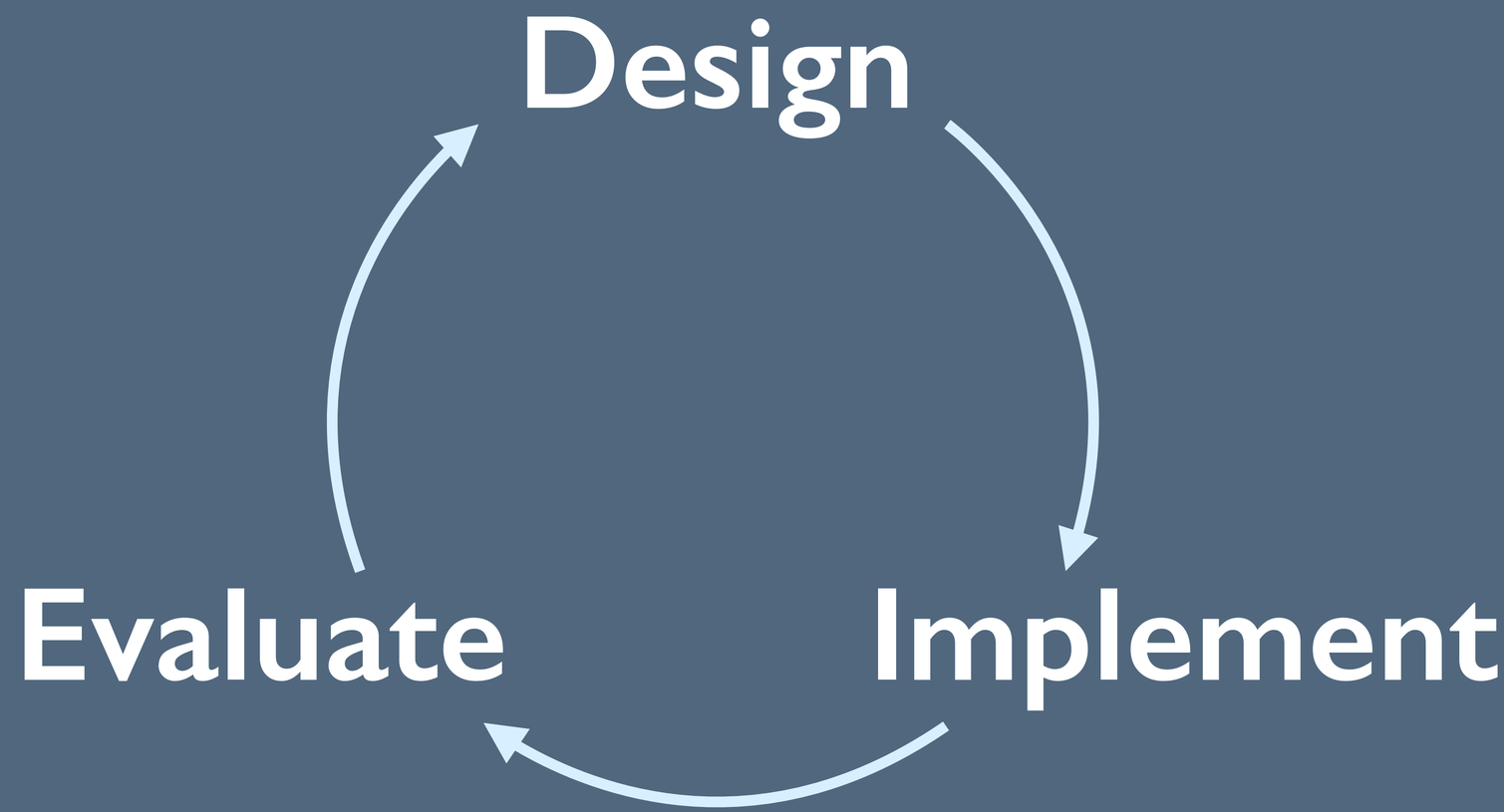
Design tools should...

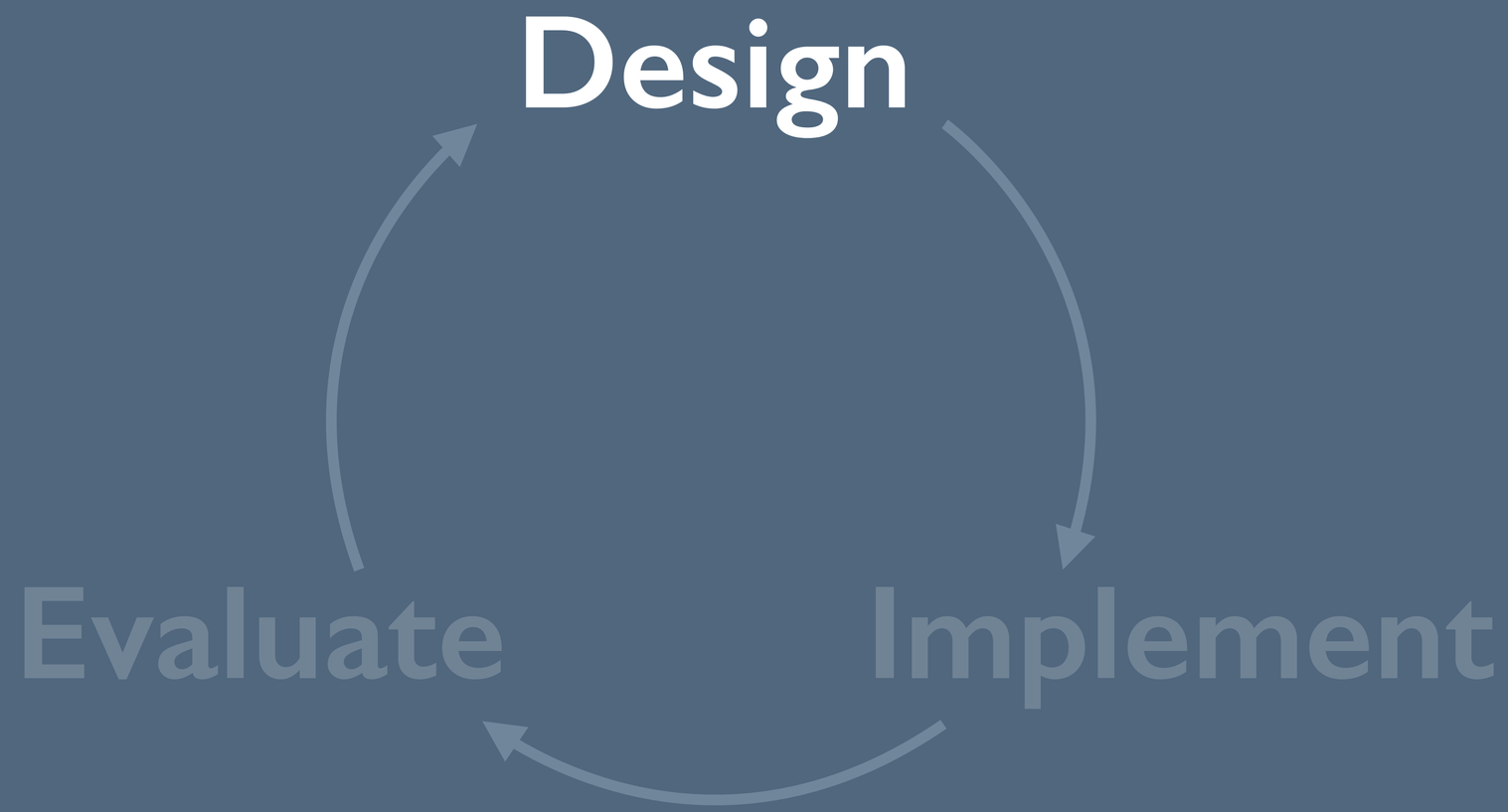
[Hartmann 2009]

Decrease UI construction time

Isolate designers from implementation details

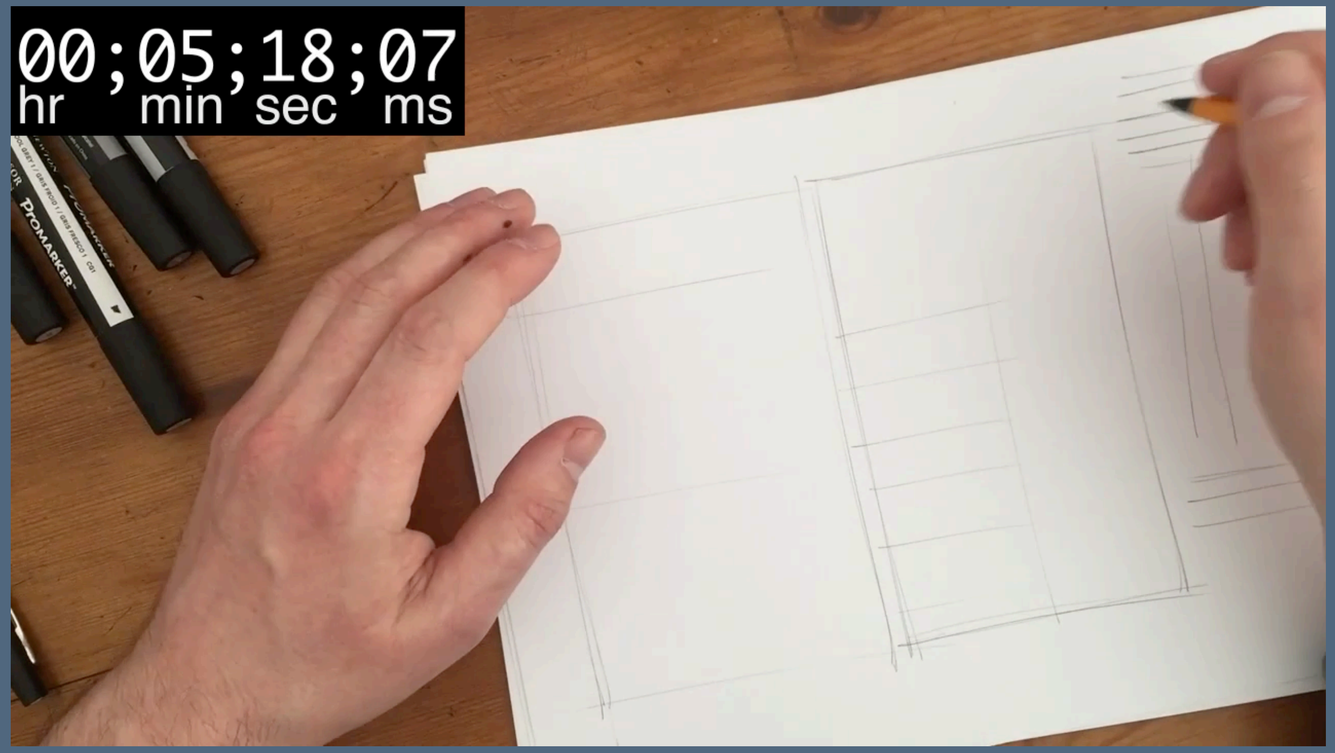
Enable designers to explore an interface technology previously reserved to engineers or other technology experts


Design
Implement
Evaluate


Design
Implement
Evaluate

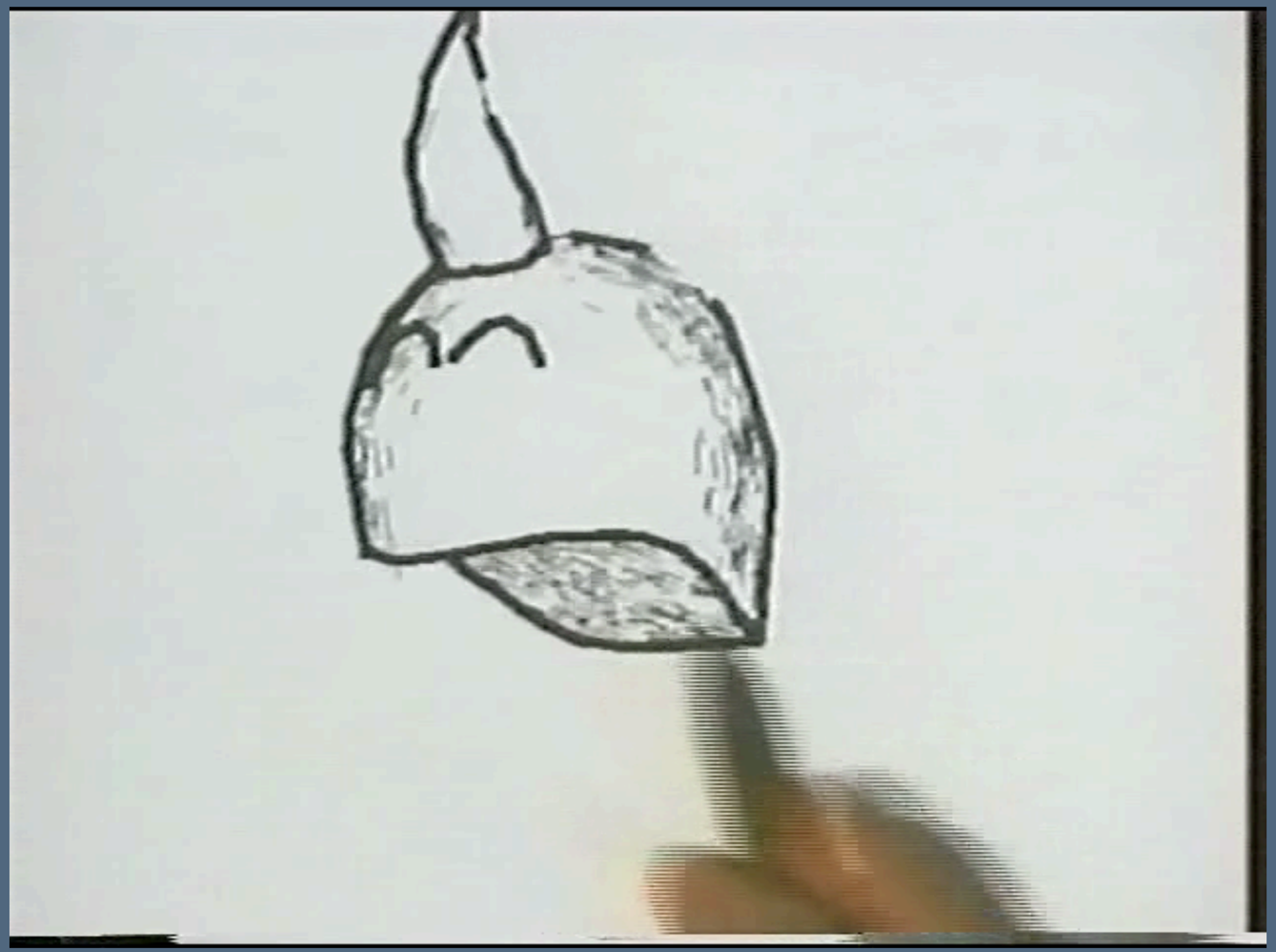
Goal of early-stage design tools: low-fidelity sketching

One major open loop in the design phase is the translation of an idea from the designer's head out into a sketch: the most rapid externalized representation possible



[Matt Corrall, edited by Bernstein]

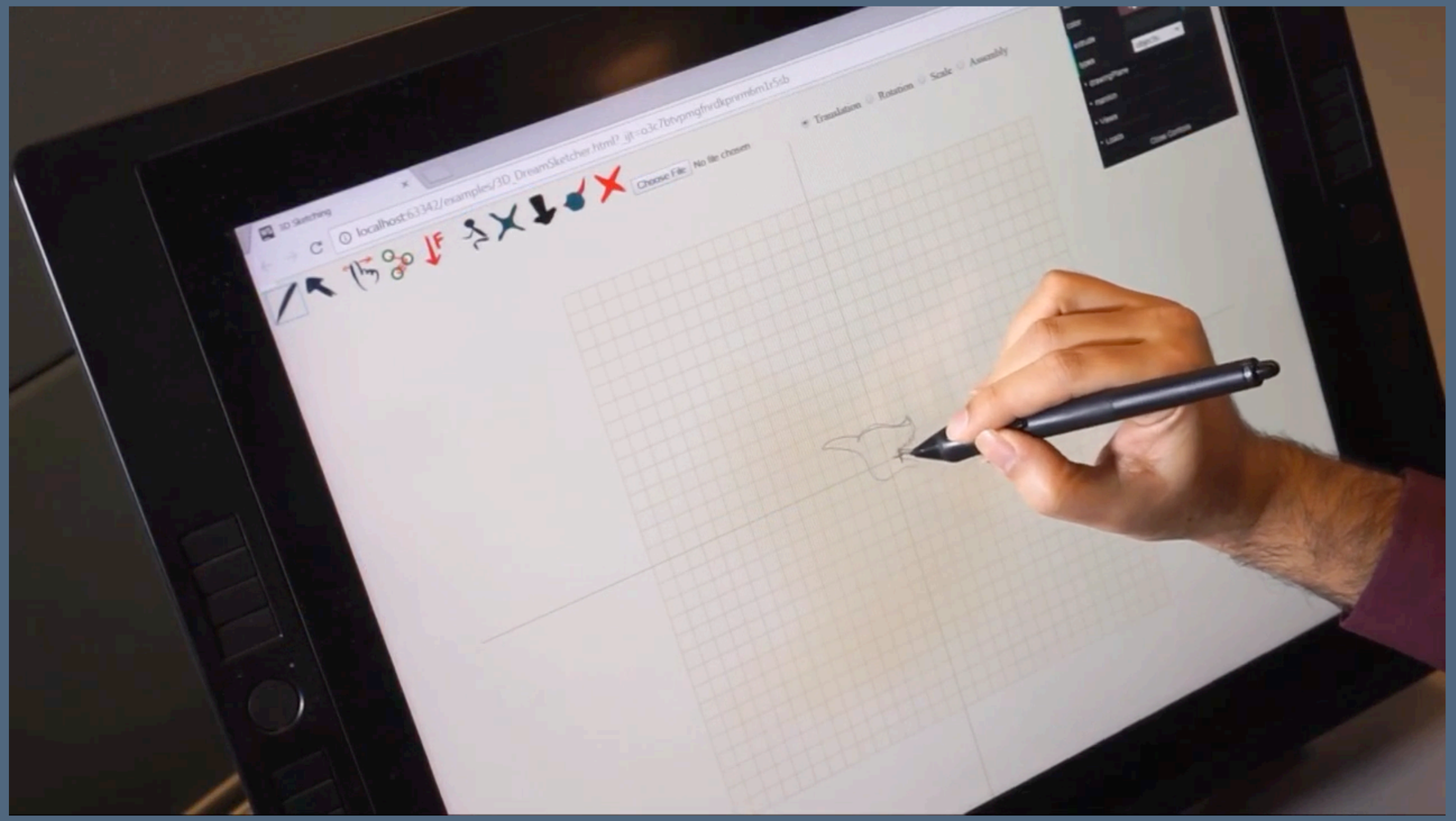
Enable rapid sketching



And keep the output sketchy + uncommitted.

Here, the designer uses the system to sketch a 3D shape to convey their idea [Igarashi, Matsuoka, and Tanaka 1999]

Computational sketches

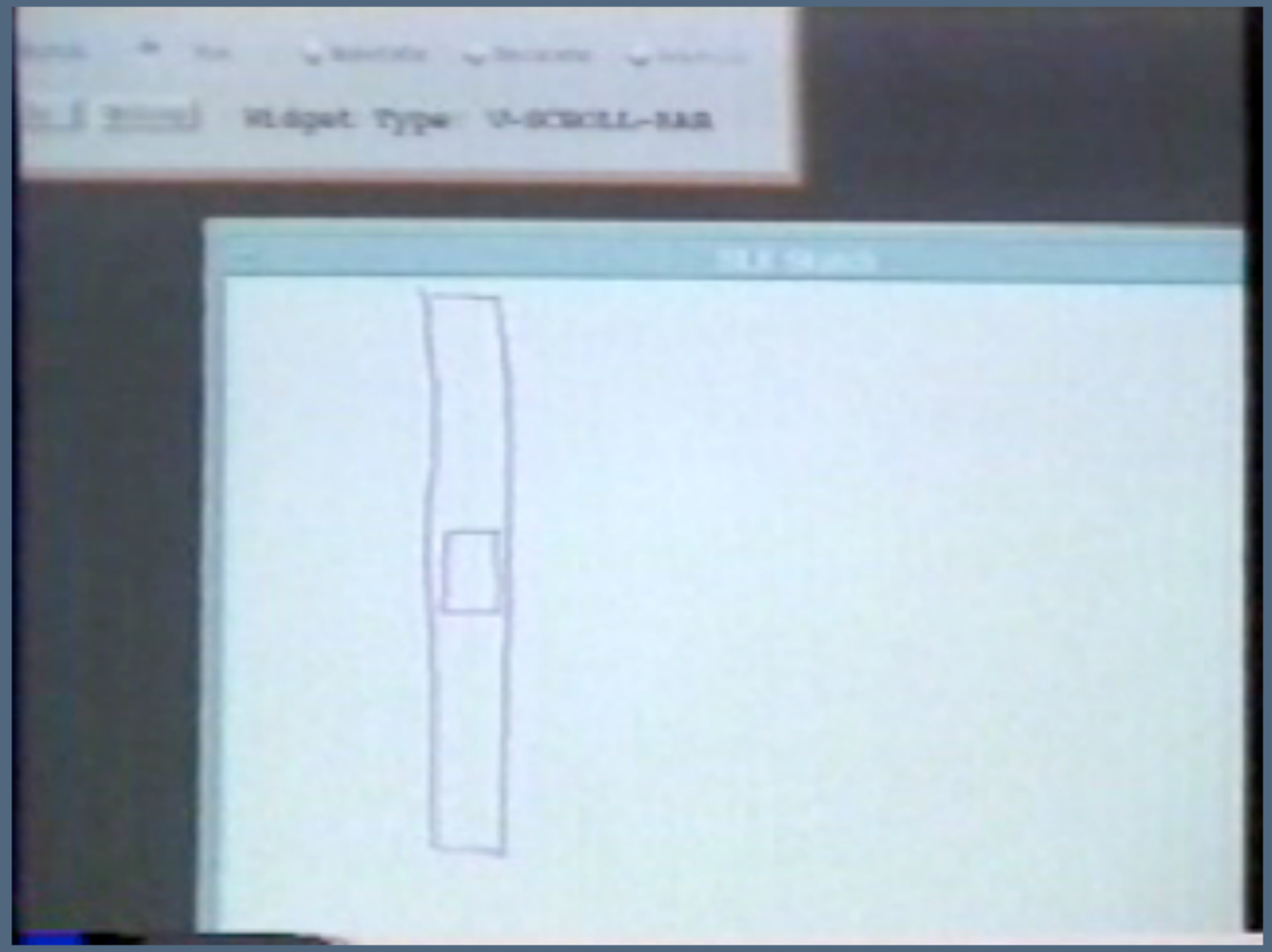


Imbue the sketch with computational properties.

The designer sketches while the system helps visualize potential physical constraints [Kazi 2017]

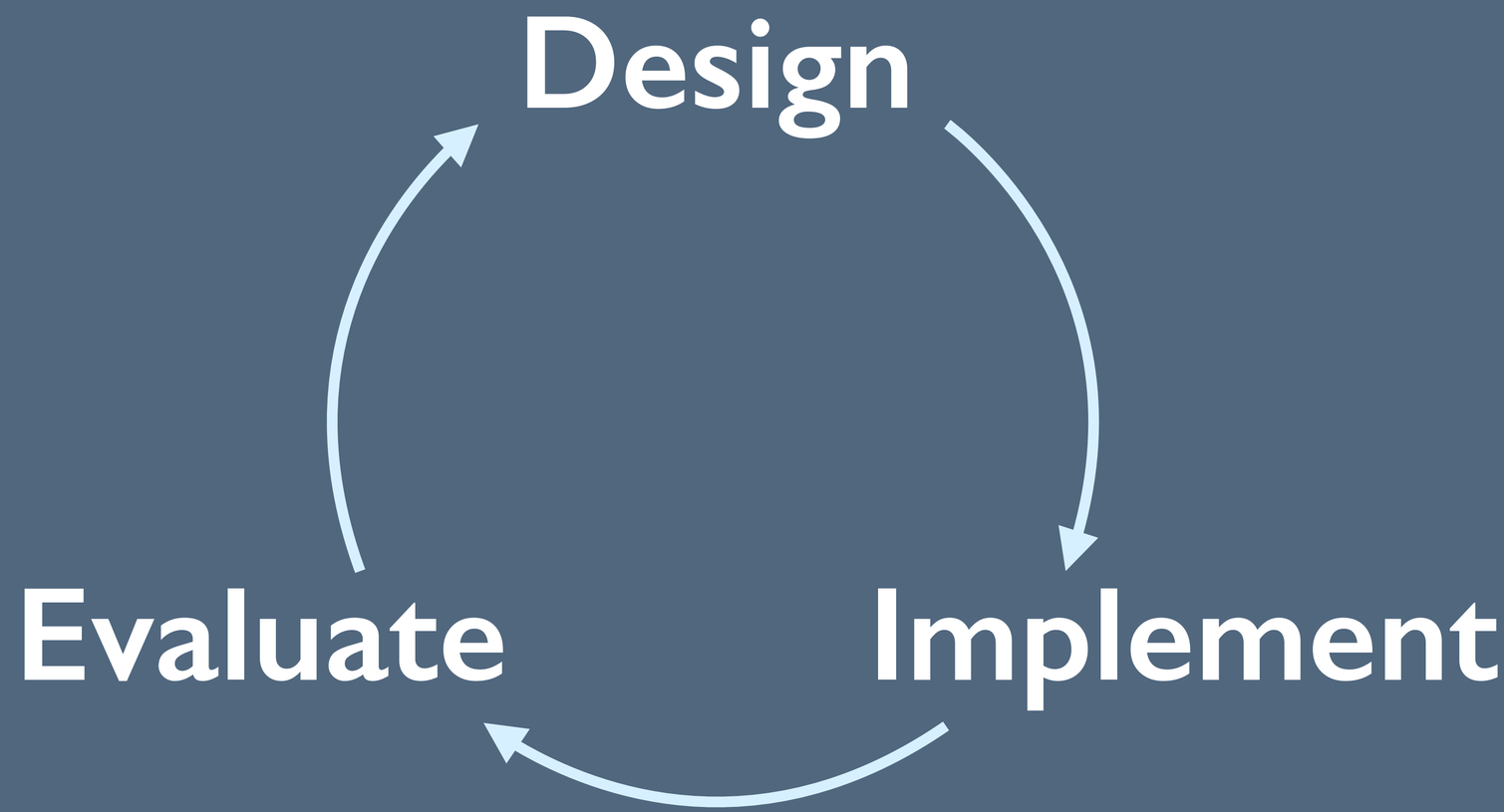
Low-fidelity prototypes

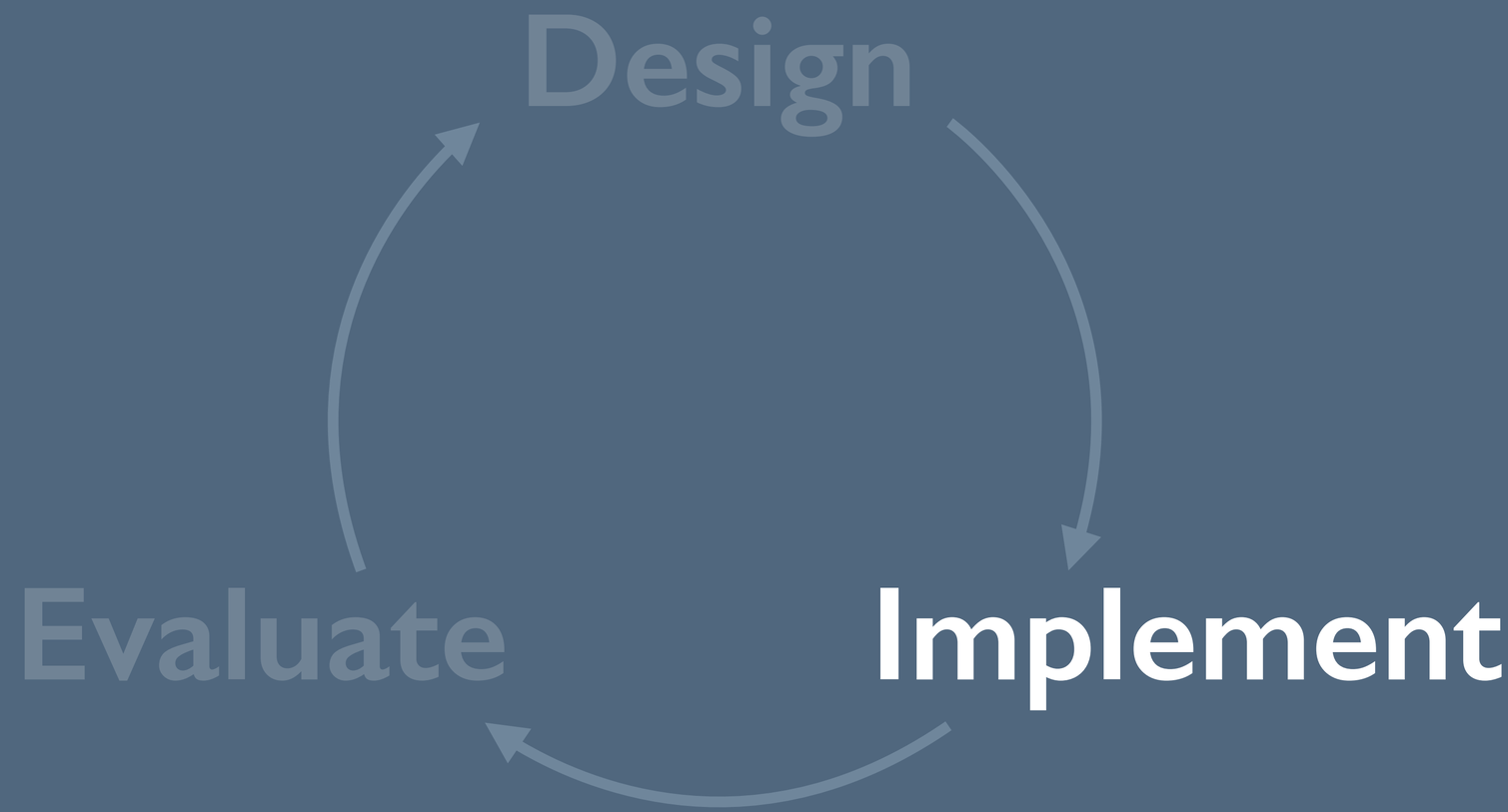
[Landay and Myers 1996]



Sketch recognition of UI components

Led to many projects on low fidelity prototyping of UIs


Design
Implement
Evaluate


Design
Implement
Evaluate

Goal of prototyping tools: decrease construction time

If we can realize our idea or sketch into a prototype faster, then we can get to a reflection stage faster

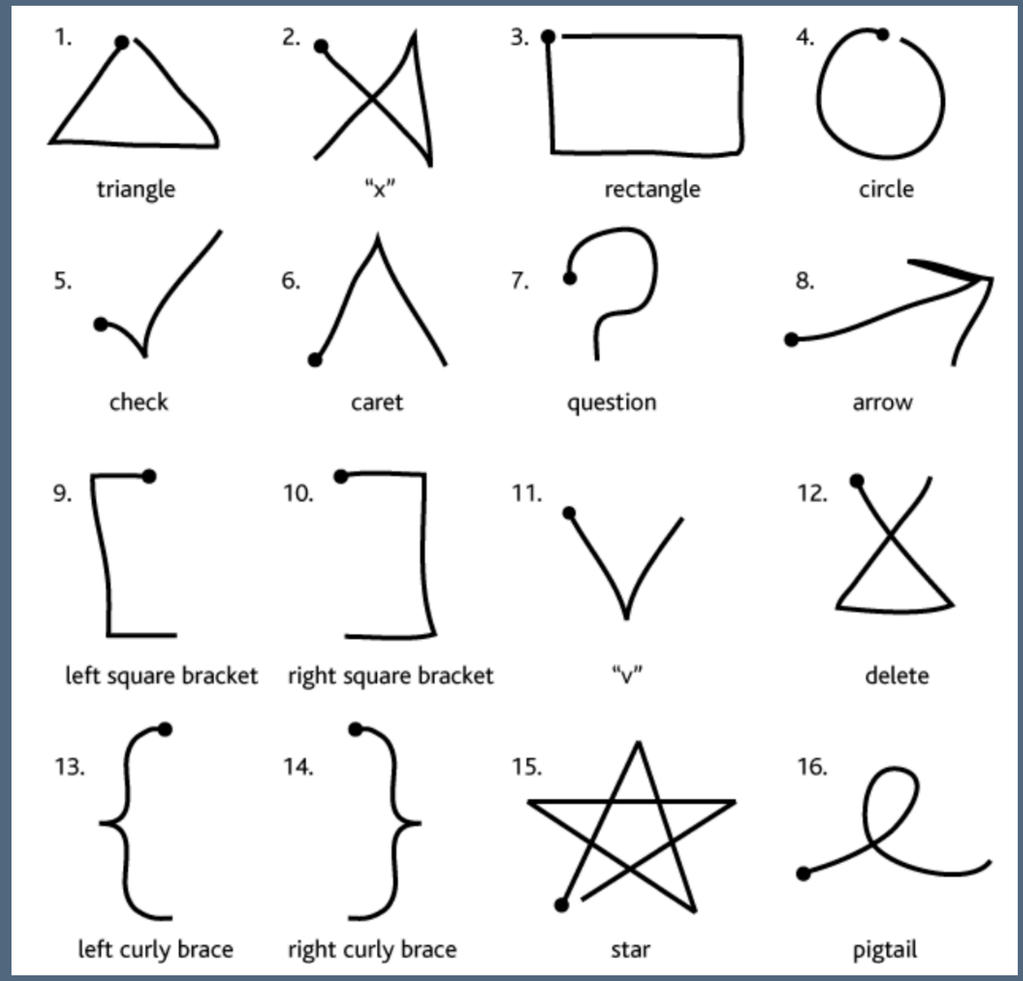
$1 gesture recognizer

[Wobbrock, Wilson, and Li 2007]

Training an end-to-end ML system for gesture recognition would take thousands of examples and a lot of time—infeasible for prototyping

The "$1 recognizer": quick 100 lines of code for 97% accuracy with only one example

Resample, rescale, rotate, and template match



Rapid, simple controls

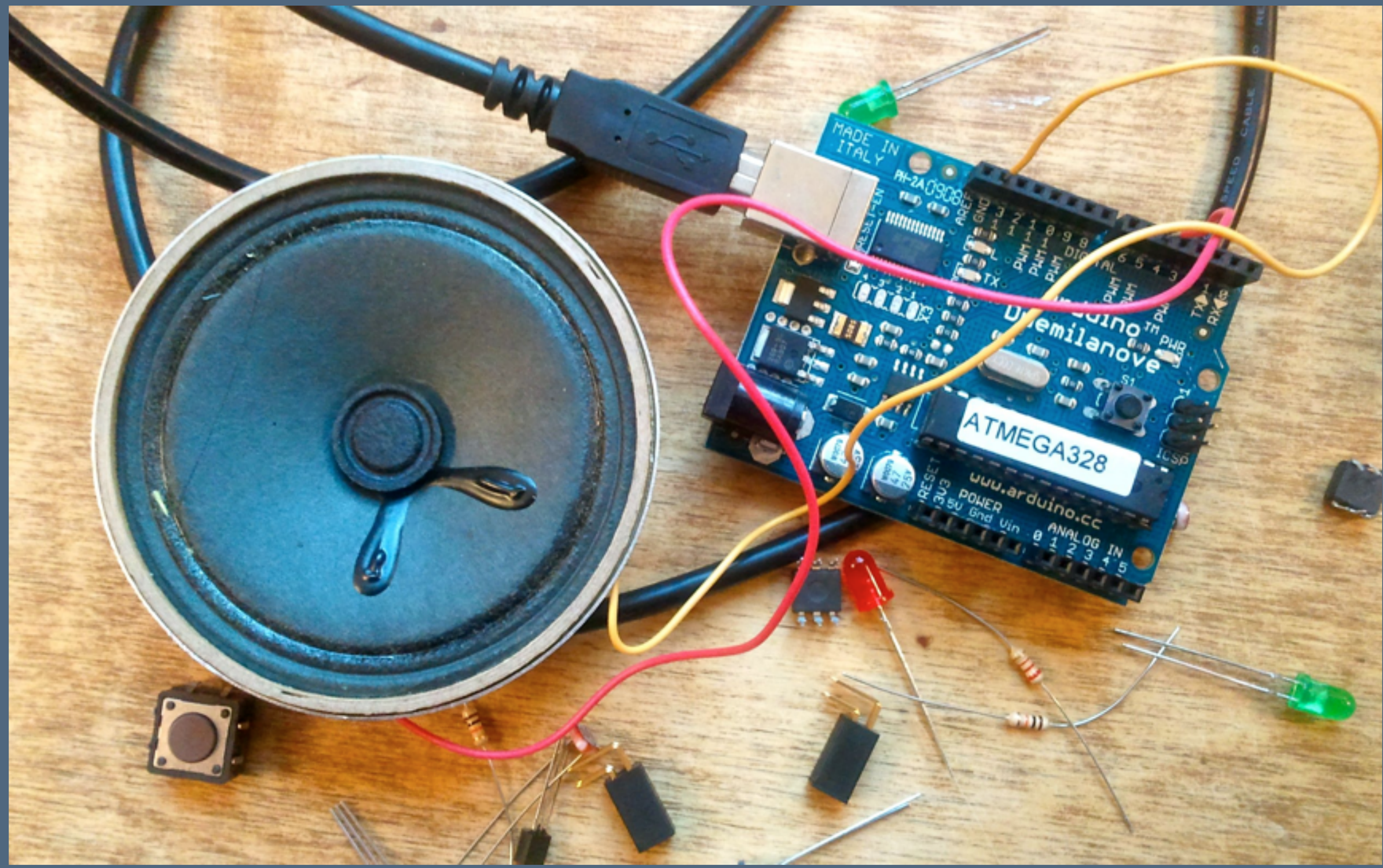
[Beginner's Mind Collective and Shaw 2012]

All you need is alligator clips

Can't do complex interaction with it, but lets you get off the ground quickly

Arduino

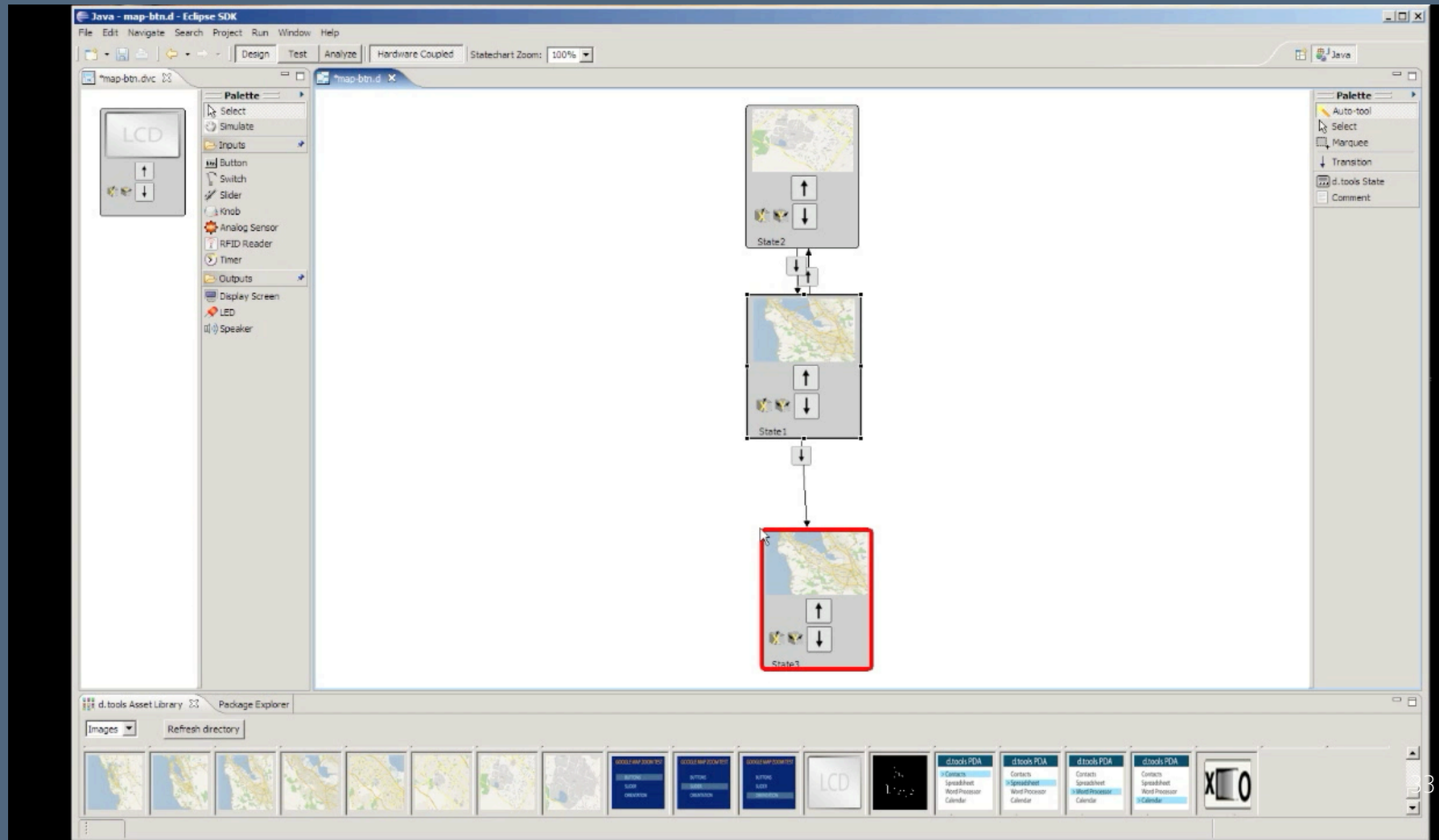
Maker board for artists, programmers and hobbyists



Prototyping physical computing

[Hartmann et al. 2006]

Plug-and-play hardware and visual statechart authoring



Replacing electronics with cameras [Savage et al. 2013]

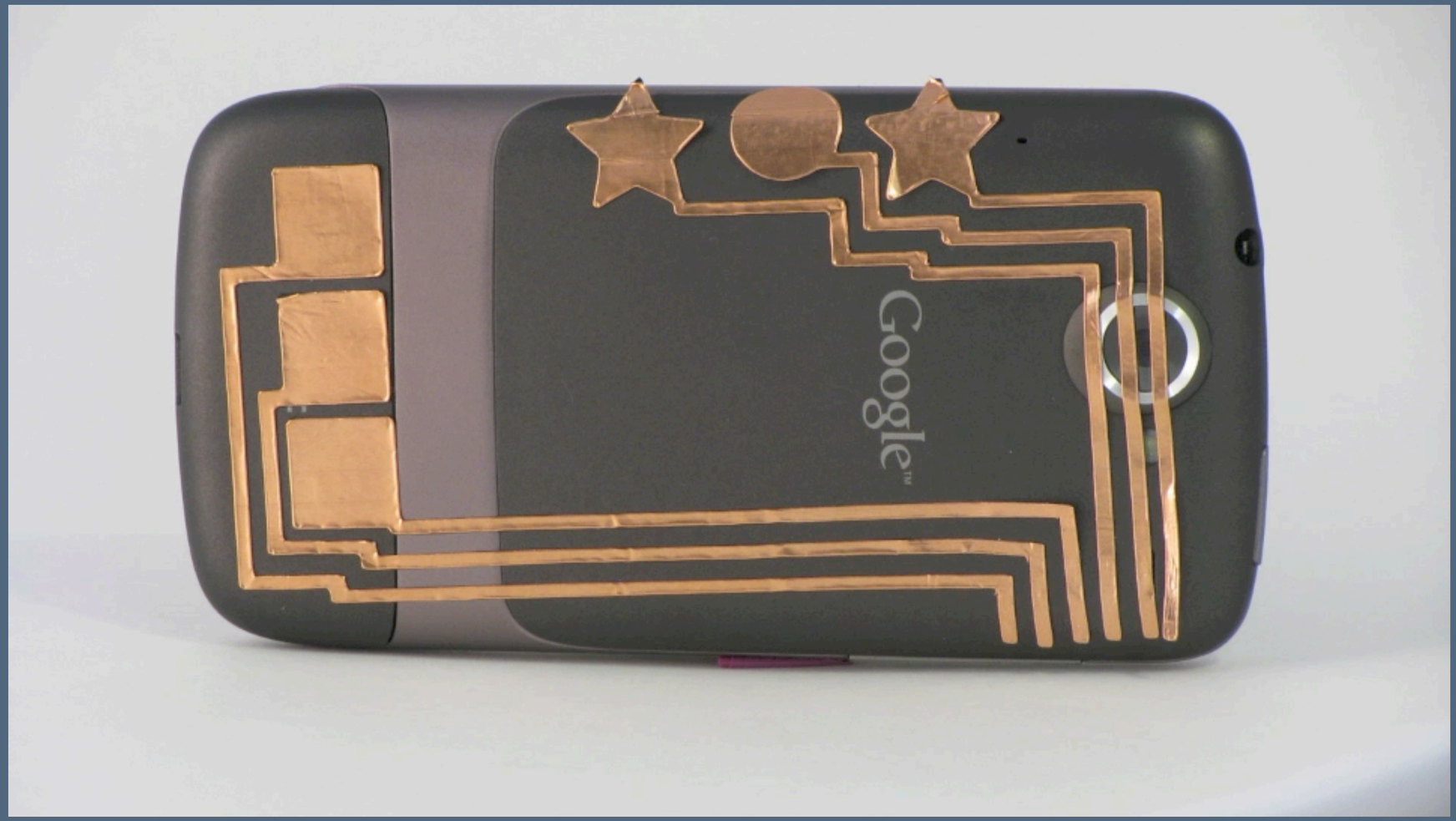
3D print your envisioned device, then screw a camera into the back of it and use computer vision instead of electronics

Prototyping touch-sensitive UIs

[Savage et al. 2012]

Make touch-sensitive physical devices in minutes

Create the UI layout, and software takes it from there



Goal of comparison tools: facilitate exploration

If we can generate many alternatives quickly, we can more rapidly explore a design space

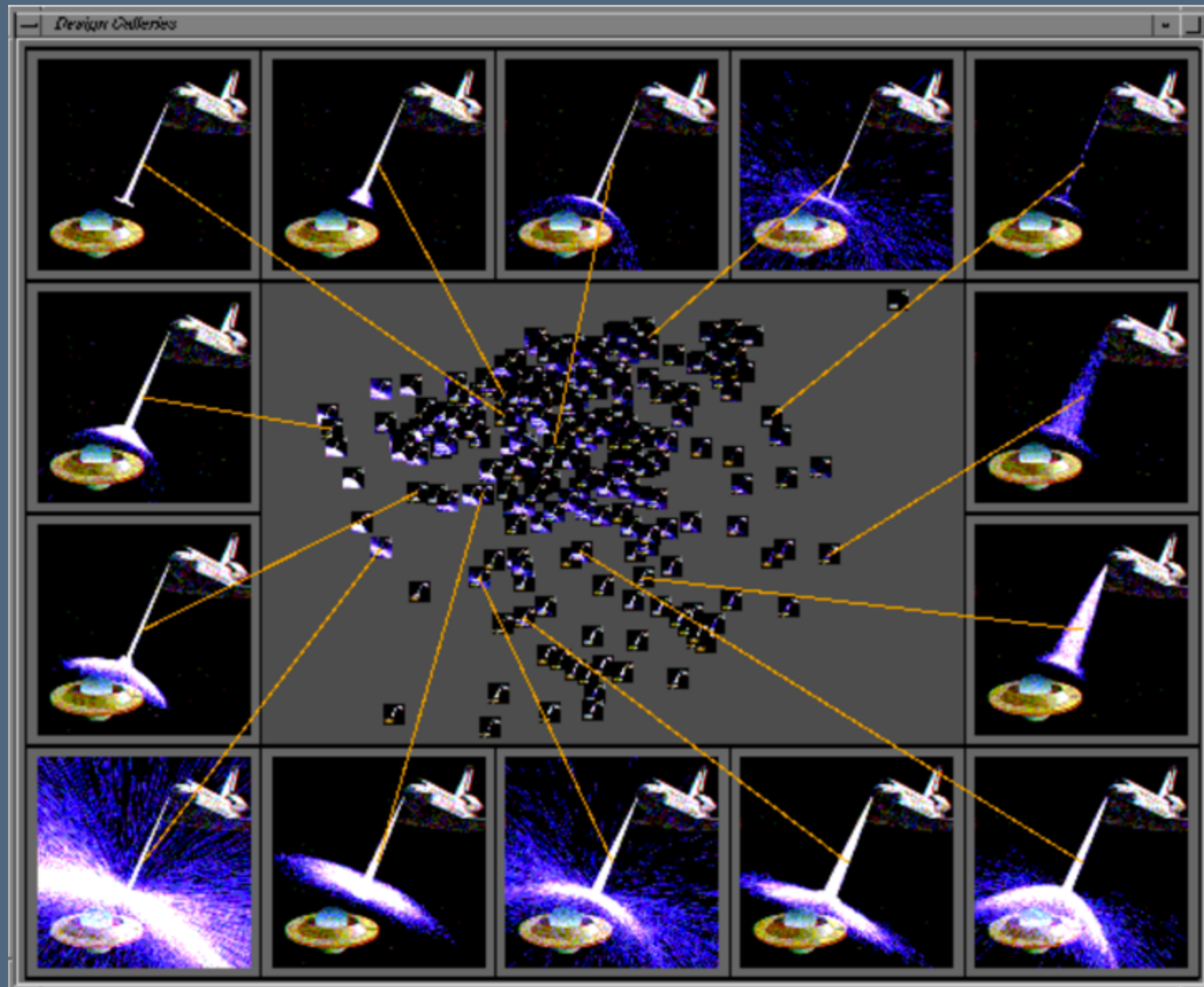
Design galleries

[Marks et al. 1997]

Automatically generate perceptually-varying alternatives within a design space

Helps the designer explore other feasible approaches

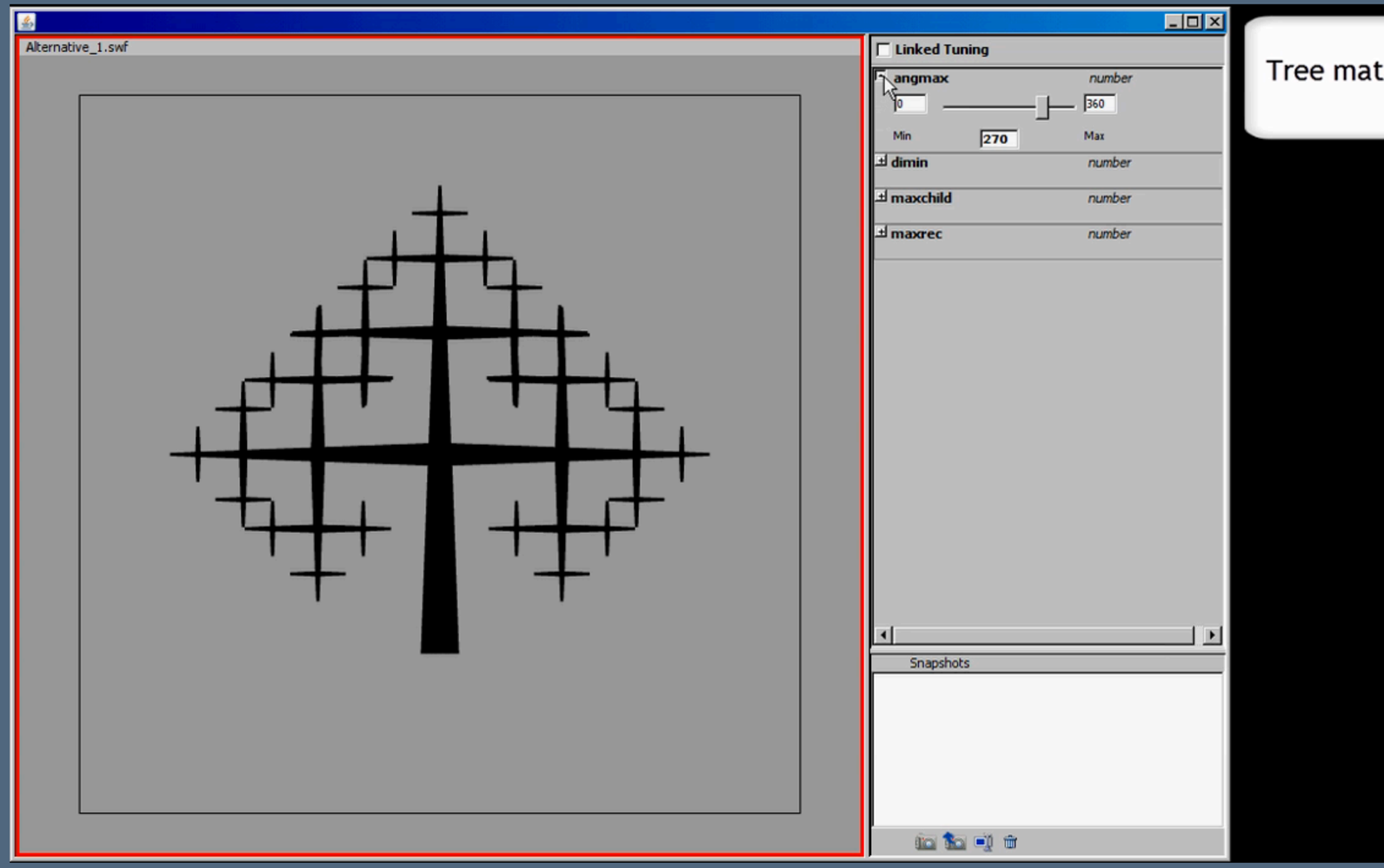
Now a widely-adopted technique inside of design tools



Explore alternatives

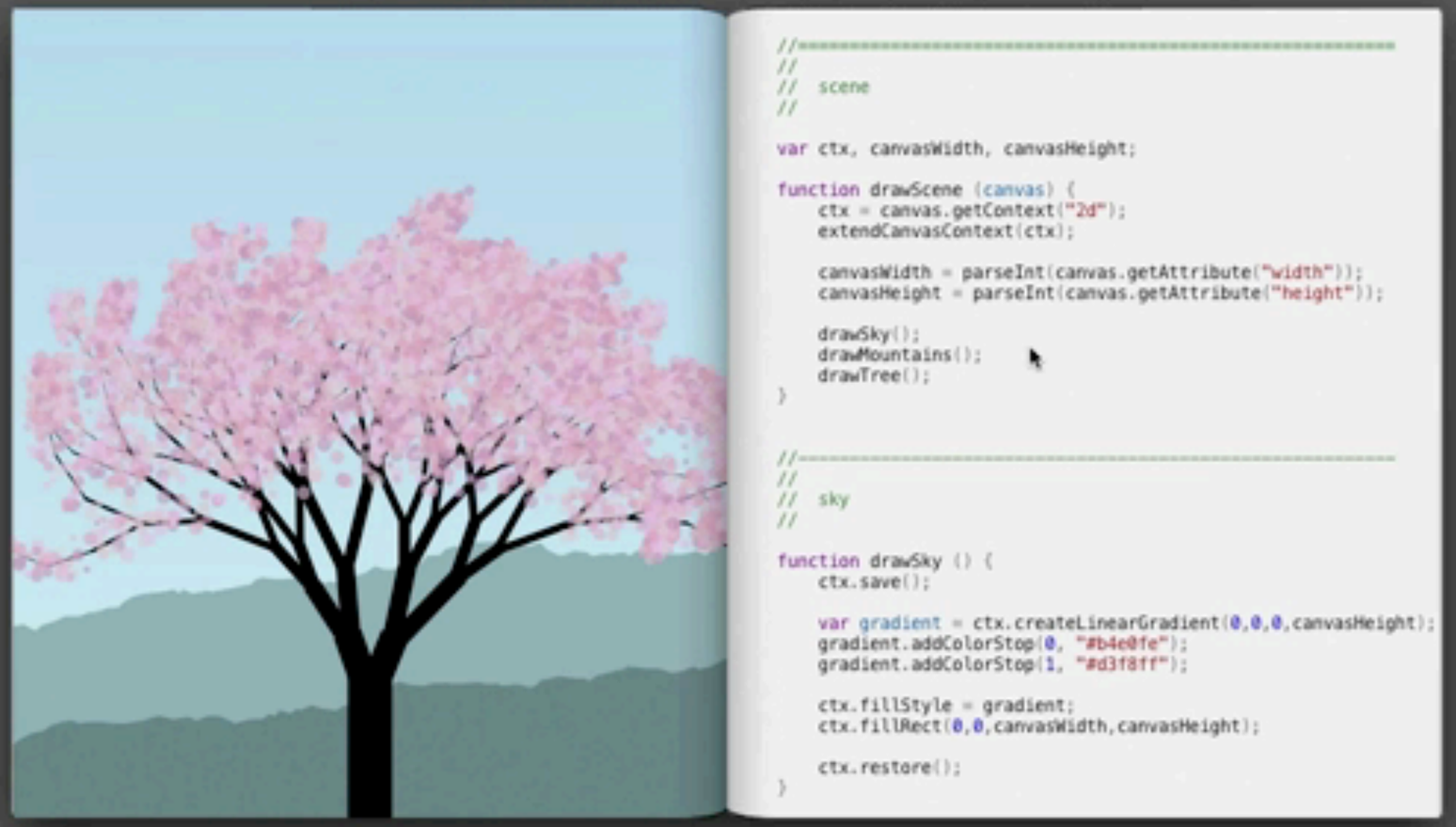
[Hartmann et al. 2009]

Tighten the loop by allowing exploration of design spaces and alternatives on a live version of the application



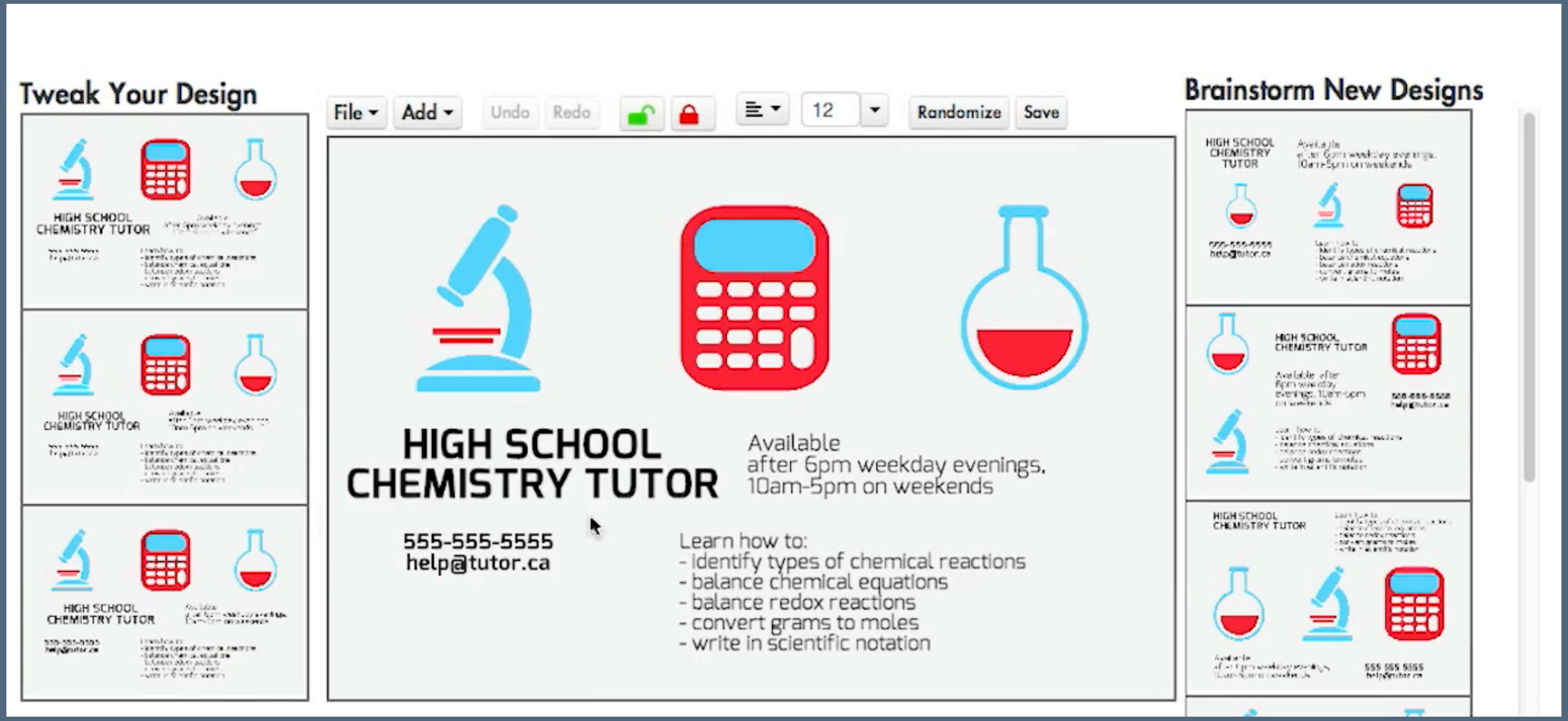
Explore alternatives

Inventing on Principle [Victor 2012]



What would a designer say?

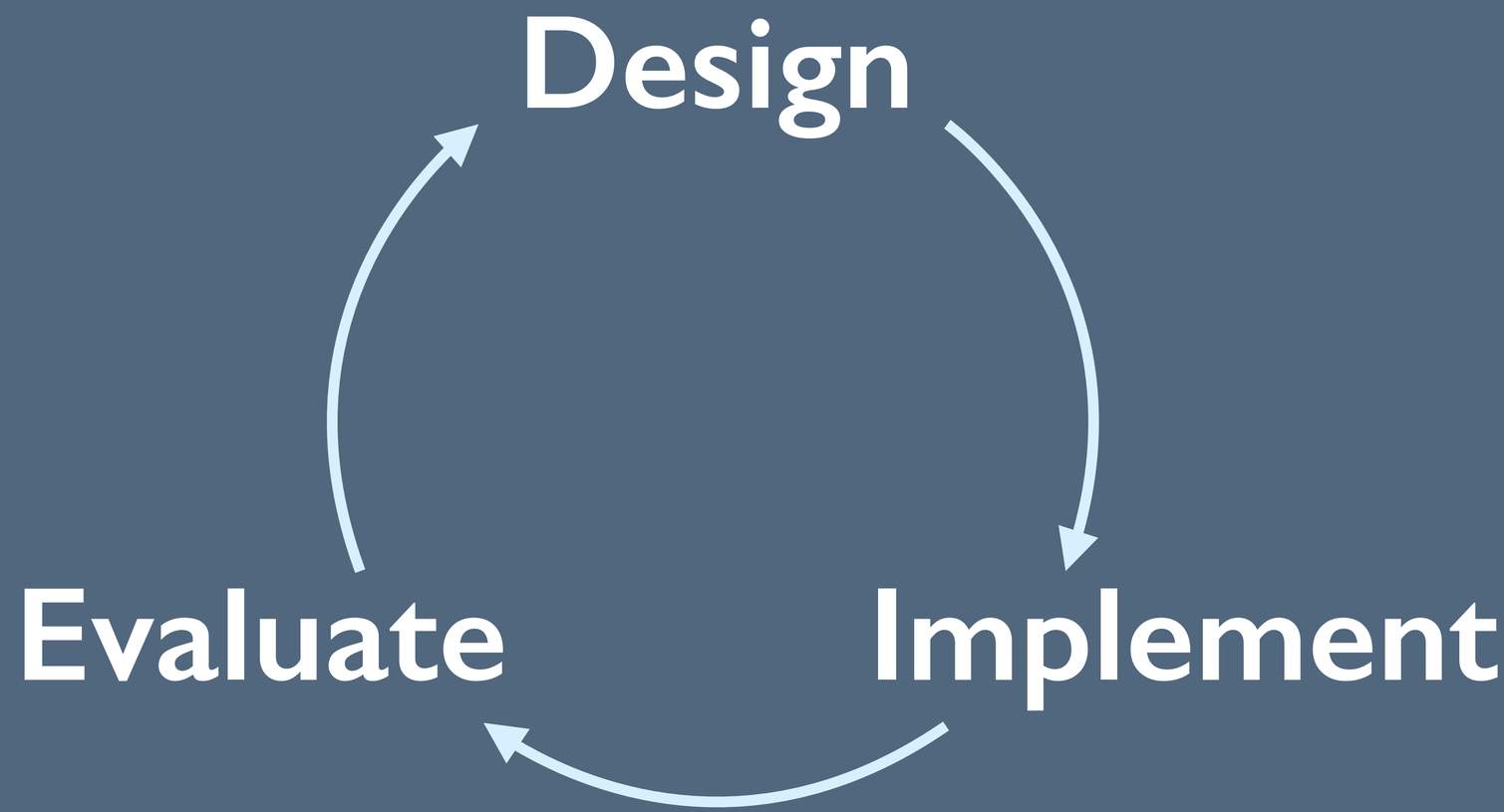
[O'Donovan, Agarwala, and Hertzmann CHI '15]

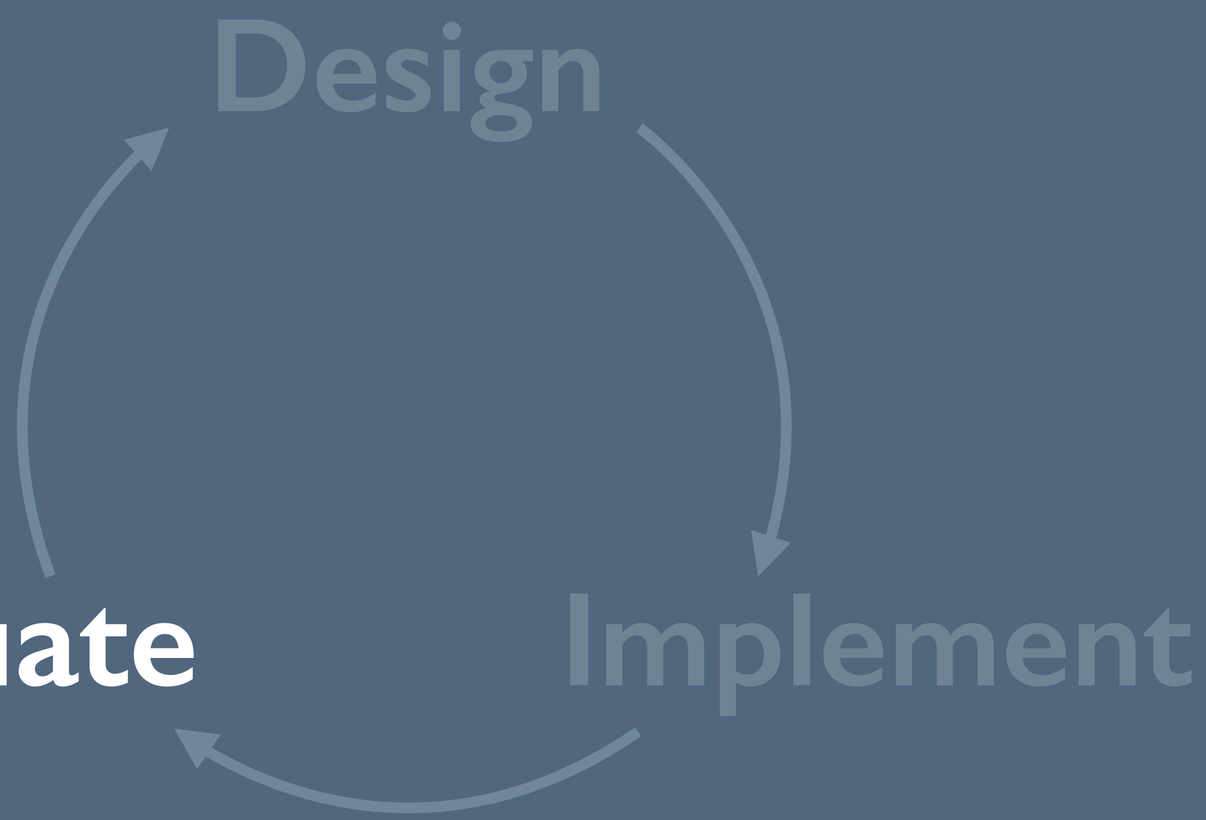


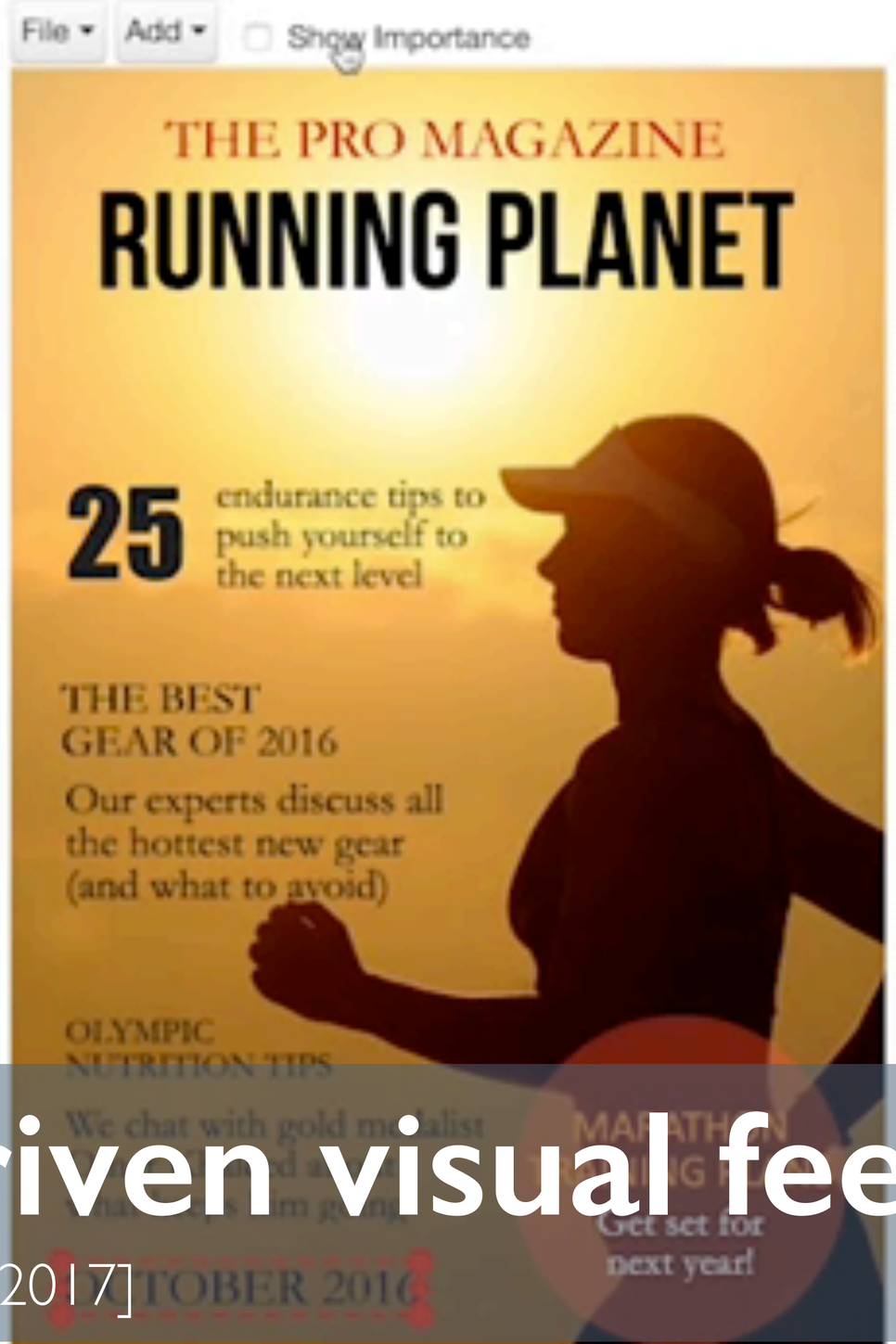
Suggesting alternatives

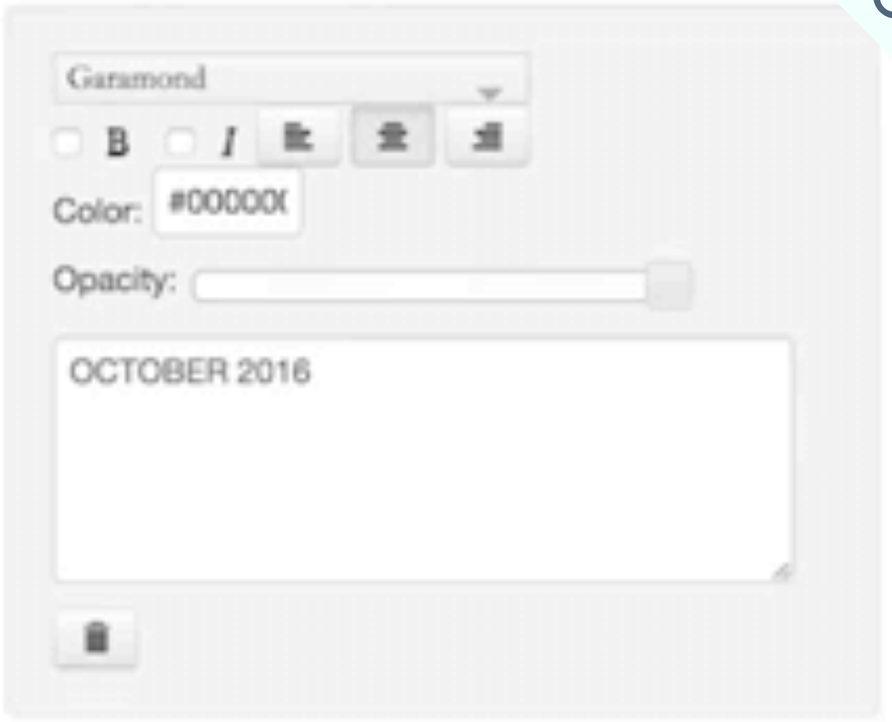
Explore parametrized design spaces by observing the designer's explorations thus far

[Koyama and Goto 2022]


Design
Implement
Evaluate


Design
Implement
Evaluate





YOU READ THIS

AI-driven visual feedback

[Bylinskii et al. 2017]

Summary

Schön's reflective practitioner: designers think in cycles; action then reflection

So, to make the designer better, enable more and better reflection

Design tools aid and accelerate reflection in action:

Early stage design: translation of an idea from the designer's head out into a sketch, the most rapid externalized representation possible

Implementation: if we can realize our idea or sketch into a prototype faster, then we can get to a reflection stage faster

Evaluation: provide better or more rapid feedback to support iteration

References

Beginner's Mind Collective, and David Shaw. "Makey Makey: improvising tangible and nature-based user interfaces." Proceedings of the sixth international conference on tangible, embedded and embodied interaction. 2012.

Bylinskii, Zoya, et al. "Learning visual importance for graphic designs and data visualizations." Proceedings of the 30th Annual ACM symposium on user interface software and technology. 2017.

Hartmann, Björn, et al. "Design as exploration: creating interface alternatives through parallel authoring and runtime tuning." Proceedings of the 21st annual ACM symposium on User interface software and technology. 2008.

Hartmann, Björn, et al. "Reflective physical prototyping through integrated design, test, and analysis." Proceedings of the 19th annual ACM symposium on User interface software and technology. 2006.

Hartmann, Björn. Gaining design insight through interaction prototyping tools. Stanford, CA: Stanford University, 2009.

Igarashi, Takeo, Satoshi Matsuoka, and Hidehiko Tanaka. "Teddy: a sketching interface for 3D freeform design." ACM SIGGRAPH 2006 Courses. 2006. 11-es.

Kazi, Rubaiat Habib, et al. "DreamSketch: Early Stage 3D Design Explorations with Sketching and Generative Design." UIST. Vol. 14. 2017.

Koyama, Yuki, and Masataka Goto. "BO as Assistant: Using Bayesian Optimization for Asynchronously Generating Design Suggestions." Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology. 2022.

References

Landay, James A., and Brad A. Myers. "Interactive sketching for the early stages of user interface design." Proceedings of the SIGCHI conference on Human factors in computing systems. 1995.

Marks, Joe, et al. "Design galleries: A general approach to setting parameters for computer graphics and animation." Proceedings of the 24th annual conference on Computer graphics and interactive techniques. 1997.

O'Donovan, Peter, Aseem Agarwala, and Aaron Hertzmann. "Designscape: Design with interactive layout suggestions." Proceedings of the 33rd annual ACM conference on human factors in computing systems. 2015.

Savage, Valkyrie, Colin Chang, and Björn Hartmann. "Sauron: embedded single-camera sensing of printed physical user interfaces." Proceedings of the 26th annual ACM symposium on User interface software and technology. 2013.

Savage, Valkyrie, Xiaohan Zhang, and Björn Hartmann. "Midas: fabricating custom capacitive touch sensors to prototype interactive objects." Proceedings of the 25th annual ACM symposium on User interface software and technology. 2012.

Schön, Donald A. The reflective practitioner: How professionals think in action. 1968.

Victor, Bret. Inventing on Principle. 2012. https://www.youtube.com/watch?v=PUv66718DII

Wobbrock, Jacob O., Andrew D. Wilson, and Yang Li. "Gestures without libraries, toolkits or training: a $1 recognizer for user interface prototypes." Proceedings of the 20th annual ACM symposium on User interface software and technology. 2007.