



Metatool

playful spaces,
machine muses

Metatool is a critical + technological course about creating **experimental design tools** using Rhino/Grasshopper as a metatool: a tool that enables the forging of other tools.

Read and discuss the philosophy/theory of pedagogy, prostheses, tools, and play.
Assemble your own experimental toolkit.
Create playful spaces to experiment with your tools.
Use machine learning to collaborate and to be inspired from your machines.

Instructor: Dan Taeyoung
Tuesdays 5pm - 7pm
300 Buell South
Columbia GSAPP
Fall 2017

Preamble

The architect encounters a site.

Armed with an arsenal of tools, she undergoes observation, research, and analysis. She understands space as a complex system, a field of possibilities in terms of space, event, and movement. More tools allow her to sketch, improvise, represent, modify, analyze, critique, and explore this field.

Over a long process of deliberation, introspection, and collaboration, she arrives at a decision that creates a new kind of site and a new context altogether, when set in motion.

It goes without saying: the architect's tools are her most prized possessions. They are akin to bodily prostheses: new augmentations that not only alter what can be done, but what can be represented and thus what can be conceptualized.

This critical architect might ask: Where does the tool come from? What does the tool want to do? What new tools can be created? How does her tools change how she thinks? Should not every process of design be one that reinvents new methods of thinking, and new tools for creation?

Thesis

As architects, we understand the social/political/cultural impact of the built environment on its inhabitants. Similarly, our creative and technical environment has an immense cognitive, political, and cultural impact on our process. Tools alter our abilities, cognition, and desires; not just what we can do, but what we want to do.

The core thesis of Metatool is: to design critically, intentionally, intelligently, and playfully, **architects must have the ability to design new critical / experimental / playful design tools.**

As a designer, how have your existing tools shaped your process? What kind of tools will you design for yourself? How will they change how you think? What will it feel like to play with them? To exchange them with others?

Course structure

Metatool is split 80/20 between the technological and critical.

Critical Metatool (20%):

Metatool is a course about discussing the nature of tools through a pedagogy/cognitive science/philosophy lens. Texts we will discuss include essays by Bruno Latour, Marshall McLuhan, and Martin Heidegger. We have workshops and discussions around our own tools, the hidden biases of tools, and the cultural/social implications of tools.

Tools, we will quickly realize together, are everywhere and complex, intricate and fascinating.

Technological Metatool (80%):

Metatool is a course about designing your own critical/experimental/playful design tools. Metatool treats Rhino/Grasshopper as a platform, or a **meta**tool: a tool that enables the creation of other tools. The course is grounded in a solid understanding of computational tools and workflows. Our main platform will be Rhino/Grasshopper and its robust ecosystem of plugins.

* * *

No prior experience with Grasshopper is required, but basic knowledge of Grasshopper is recommended. Students with prior programming knowledge (Python/C#/VB.net) or experience in Grasshopper are welcomed.

The course is structured along the lines of the 'flipped classroom': in which lectures, workshops, discussions, and peer critiques happen during class. Outside of class, video tutorials for Metatool will be made available, as well as the Skill Tree, an archive of general Grasshopper videos, developed in conjunction with the ADR curriculum.

Session A (9/5-10/20): Playful Spaces

In Session A, Grasshopper will be used to create tools that construct **playful spaces** for computational design. To use a playground metaphor, we will be creating the sandboxes, shovels, and sand -- environments in which designers can interact and play. How do you affect your architectural design process by shaping your environment? How do you balance intent with play, data with inspiration?

Using a series of custom scripts, techniques and workflows, Grasshopper will be used to create playful new interfaces that disappear and become part of the Rhino environment. Mouse and keyboards will be used to create new relationships with 3d modeling, data will be become incorporated into new forms of tools, and representation will be explored as a way to design the behavior of the user. Technology will not be used as form-generating machines, but both as intelligent prostheses and playful sandboxes of the designer, altering the design process as a result.

Session A is oriented around the tool and environment's agency or 'desire'; we will create one experimental tool: the tool will draw from a previous project, and will integrate in-depth technical knowledge with a larger design intent that it attempts to execute. Session A's focus is on balancing a deep understanding on how Grasshopper and computation works, with the conceptual question of how computation incorporates itself into your design.

The point is to design and create your own feedback loops, where new ideas can generate new design techniques, and new design techniques can thus generate new ideas.

Session B (10/23 - 12/1): Machine Muses

In Session B, we will experiment with data-driven processes such as machine learning or dimensionality reduction to create a **machine muse**, and to experiment with different ways of collaborating with computation. How can you use machine learning to explore new ideas? What do we learn when we treat computation as an imperfect collaborator, not a problem-solver? What will we create when we can collaborate with our computational tools? When we can seek them out as an oracle, or a muse?

Students will share a brief in order to formulate a series of experimental design tools that operate more like muses or collaborators. These tools will be data-rich, nonsensical, performance-oriented, deliberately erroneous, computationally intelligent -- design projects in their own right. The goal of Session B is to collectively amass a series of playful design tools in relation to a project, so that a group of designers are able to engage in a rigorous and playful design process through a series of tools, so well-crafted they manage to alter the designer herself.

Students are recommended to have taken Session A, regardless of their Grasshopper expertise. If you are interested in taking just Session B, please free free to email me.

Ethos / Code of intention

Every social environment, including a university or a class setting, has a culture that is often unconsciously set. As the course facilitator, one of my roles is to set the boundaries and tone of the class, and to consciously and actively create a space that is supportive, intentional, respectful, experimental and playful.

As an educator, I strongly believe in a warm culture of experimentation, play and risk-taking, and experience has taught me that only intentional spaces can really create radically open pedagogical cultures. True experimentation only happens in a supportive environment. In the class, we will prioritize learning over production, and experimentation over success, and will thus create conscious structures of support.

Together, Metatool is a space where we can be:

- respectful, positive, and thoughtful
- supportive and generous
- actively listening, and creating space for others to speak
- comfortable with silence
- grateful about valuable mistakes
- excited about risky ideas.

I am always available to talk and to provide help. I can best be reached at:
dan.taeyoung@columbia.edu

Grading

All courses that are part of Columbia GSAPP's curriculum have HP/P/LP grades. At their best, grades offer a mildly helpful metric to each student for how much change and progress she or he has achieved in a semester. At their worst, grades help generate a risk-averse system where exploration and curiosity are left unrewarded or punished.

The course places a strong focus on taking risks (relative to each student), in addition to project depth and class participation. A risky and difficult project or experiment that ultimately 'fails' and through which we learn is much more valuable than a perfectly-executed project that is very familiar to the student.

In the interest of transparency: grading is done by four factors: 1) class listening & participation 2) project execution 3) amount of learning growth 4) amount of risk taken. Each of those factors are equally weighted to each other. The factors are designed so that the way to get an 'ideal grade' is simply to be share and listen, be curious, and learn from working on a project that you find fascinating and interesting.

Overall schedule

SESSION A

Week 1: Introduction: Tools as prosthetics, augmentations.

Week 2: Hand-Eye Feedback Loops: simultaneous data analysis and design

Week 3: Representation As Manipulation: More outputs

Week 4: New interfaces and other inputs

Week 5: Soft Coercion: feedback, or how the tool designs the user

Week 6: Import/Export: sharing data with others over the web

Week 7: Desk crits / Pinup.

SESSION B

Week 8: Introduction: Computational Collaboration: loops & evolutionary solvers

Week 9: Slow Tools: delays and slowness as a technique

Week 10: Machine making: Neural networks and markov chains

Week 11: Machine thinking: dimensionality reduction and k-means clustering

Week 12: Going Faster: finding new techniques through Python and performance (1/2)

Week 13: Going Further: the larger landscape of Grasshopper and software

Week 14: Final Review