

permacomputing/ permacomputing

Permacomputing is both a concept and a community of practice oriented around issues of resilience and regenerativity in computer and network technology inspired by permaculture.

In a time where computing epitomizes industrial waste and exploitation, permacomputing encourages a more sustainable approach, maximizing hardware lifespans, minimizing energy use and focussing on the use of already available computational resources. Permacomputing asks the question whether it is possible to rethink computing in the same way as permaculture rethinks agriculture. Permaculture is the science and practice of creating semi-permanent ecosystems of nature. The resilience of any such ecosystem is equal to its diversity and interconnectedness. Permaculture design is a system of assembling conceptual, material and strategic components in a pattern which functions to benefit life in all its forms. It seeks to provide a sustainable and secure place for living things on this earth.

At first it may seem paradoxical to connect permaculture and computation. Indeed, an extractive technology that depends on a wasteful use of finite resources can hardly be permanent. Therefore, by making this connection, what we are truly asking is whether or not there can be a place for computer and network technology in a world where humans contribute to the well-being of the biosphere rather than destroy it? And if yes, how?

Permacomputing wants to imagine such a place and take steps towards it. It is therefore both utopian and practical. We want to find out how we can practice good relations with the Earth by learning from ecological systems to leverage and re-center existing technologies and practices. A radical reduction of wastefulness is a fundamental aspect of it: maximize the hardware lifespans, minimize the energy use. And this is not just about a set of technical problems to be fixed—the attitudes also need a radical turn. Understandability is aesthetics, virtual does not mean immaterial and doing things with less is not *a return to the past*. We want to investigate what a permacomputing way of life could be, and what sort of transformative computational culture and aesthetics it could bring forward.

The **principles** of permacomputing are:

- Hope for the Best, Prepare for the Worst
- Care for All Hardware — Especially the Chips
- Observe First
- Not Doing
- Expose the Seams
- Consider Carefully the Interaction Between Simplicity, Complexity and Scale
- Keep It Flexible
- Build on Solid Ground

- (Almost) Everything has a place
- Integrate Biological and Renewable Resources

Properties of permacomputing systems

The principles concretely manifest themselves in various forms so as to highlight the following properties:

- **accessible**: well documented and adaptable to an individual's needs.
- **compatible**: works on a variety of architectures.
- **efficient**: uses as little resources (power, memory, etc) as possible ([minimization](#)).
- **flexible**: modular, portable, adapts to various use-cases.
- **resilient**: repairable, [offline-first](#), low-maintenance, designed for disassembly, [planned for longevity](#), [maximized lifespan](#), descent-friendly or [designed for descent](#)

Some additional concerns are of indirect interest because they impose costs on the entire end-to-end process of software creation:

- it's [bootstrapped](#) from machine code without circular reasoning (bootstrappable builds)
- it's obvious what source code went into it (reproducible builds)
- it's easy to audit its source code, including all dependencies