

# STUDIO OF ART AND PROGRAMMING: REACHING OUT TO ART AND ARCHITECTURE FROM INSIDE ENGINEERING

**Etienne Delacroix**

Universidad de la Republica  
Facultad de Ingenieria  
Instituto de ingenieria electrica,  
Julio Herrera y Reissig 565,  
Montevideo 1130, Uruguay  
eti@mit.edu

## Abstract

*Combining aspects of engineering with traditions of studio art we investigate an interface between both worlds: using a substantial accumulation of electrodigital refuse, taken as a “raw expressive medium”, an elective course ( TAP: “taller de arte y programacion” – Studio of art and programming”) takes a large mixed group of students ( engineering, art, architecture, music, etc..) with very different levels of skills, for a sustained immersion into an exploration context. Eliminating in a large measure the problem of “costs” by using obsolete, discarded computer parts, students manipulate, observe, deconstruct, reconstruct functional hardware and use programming to produce an expressive documentation of the process. The objective is not to work on “products” but on the production of “symbolic value” by uncovering and staging the fundamentals of electro-digital-computational knowledge into a form of “theater of technology”.*

## 1. Introduction

This work is the outcome of a long process of experimentation in the context of a “traditional” artist studio but with a previous background of research in physics. It is only at the onset of my second post-doc that i came to the conclusion that my years of science could be a foundation for doing art.

Followed then 15 years of practicing drawing and painting in the most traditional context: an artist studio in Paris, completely cut-off from academic circles, consequence of successive transplants between Belgium, the United States and France. A legacy of physics in my work as a painter, was awareness of computers, of the gestures to assemble them, of the process of programing them, of the way they could model and control physical processes. Learning to paint went thus on par with studying emerging microcomputers. All along i focused on one question: What would happen to the practice of a traditional painter, if he was given “free” access to computers in all its aspects, in the same way that an artist must have unrestrained access to the “totality” his chosen media.

The answer came, over the years, as a series of drawings and paintings, woven with a corresponding series of software “sketches” which had evolved into singular, full fledged production tools. Using computers allowed me to organize works as sequences of gestures leading to paintings. Style had evolved into a generative process based on combining elementary shapes from a set i called “morphems”, with constant feedback between painting and writing code. From early nineties on, crude networking of machines and database technologies allowed me to make installations and performances where a painterly, narrative, interactive process could move onto the stage. I had been

working alone with no access to competent programers or intellectual resources from universities.

For these reasons, this work was not understood by art circles, nor received by technolgists, both sides being captivated by rapid progress of mass market digital graphic tools and the emerging of a high-end, high budget art-and technology paradigm.

Digital technology remained absent from the artist’s studio. Equipement and software prohibitively expensive and plagued by the difficulty of absorbing knowledge necessary to use this medium with the autonomy that is fundamental to studio artists. From technology’s side, gesture was becoming obsolete, traditions of studio art marginalized into handycraft and personal therapy. Large institutions were dominating public perception of where frontiers of art were heading.

## 2. Nomadic workshops

By end of the nineties, new techniques of fabrication and huge increase in processing power implied that large numbers of machines would rapidly become obsolete, containing, frozen in their parts, the knowledge needed to allow artists the possibility of re-enacting, with their own traditions, fundamentals of digital technology.

Rather than focusing art–tech to high-end projects, I was convinced that we should create low cost “interface spaces” between traditional “gestuality” of studio art and basic skills of electro-computer science engineering. Such interfaces could use the growing accumulation of electro-digital refuse as a “raw” expressive medium, mix it with skills and knowledge of engineering education and dedicate themselves to the “staging” of these fundamentals as a form of art. This woud release from within the

universities new, sustainable methodologies for technological outreach and foster multidiscipline investigative energies among young students

To seed the project, I imagined a performance/installation workshop travelling with compact kits of resources and proposing to art communities, schools, universities, an introduction to the fundamentals of the computer as a raw expressive media. In the process, mixed groups of students from engineering, art, social sciences etc... would assemble computers from obsolete off the shelf parts, invent their form factor, install operating systems, network machines together, connect them to simple devices and record the the questioning that it would raise. These recordings would provide the materials for documentary-like media produced by way of an introduction to programming code. An art "with programming", articulating core vocabularies to be shared by a wide variety of people confronted with computers as a "nexus" of encounter

These "nomadic workshops" should stay away from making or using "products" but rather focus upon production of symbolic value, explore esthetics of basic electro-digital-computational functionality, as Art, and explore patterns of "gestures" proper to this new medium with a concern for radical economy of means and autonomy. Each one would result in some sort of recording ( CD-rom ) of an installation/documentary on the fundamentals of electro-digital-computational technology and reflecting its perception by the group doing the workshop. The network of communities hosting workshops could then share an ongoing reflexion over the data produced by all. A form of Collective intelligence that the free software community had already actively begun to develop

Interestingly the virtual-technological euphoria of the end of the XX century, deafened ears of the academic world to this proposal: Technology had to be ultimate. Economy of means was irrelevant, products and markets were going to solve problems of the world. The credo of liberalization and open markets, however, was not matched by territorial traditions. Fierce battles over intellectual property left no room for such a "free", educational endeavour.

By 1998, I had lost my studio in Paris and resettled in Boston as a visiting scholar at MIT to confront these ideas with state-of-the-art research on media that was going on at the insitute and begun to run experiments to concretize this project: A one semester course dubbed "building a computer/making Art" at the Massachusetts college of art, workshops in various schools of New England, a continuous informal process with numerous students of all types. Nevertheless I kept encountering difficulties at overcoming resistences of established curricular and territorial interests.

Research in education was focused on marketing products dependent on expensive logistics and maintained an uneasy relationship with Art. Artists, were deemed unreliable, overly narcissic. Resources for computing in art schools were being absorbed by graphic design and focused on using proprietary software and production tools. Nothing was being done to uncover to Art students the underlying nature of electro digital technology, and the "culture of measurement" implied by this emerging world of "bits and atoms" with the complex relationships in which their

structures operate. A set of relationships nevertheless central to the ongoing mutation of our means of perception, mutation that is not only technological but also problematically cultural. Technology had come hard and fast, but its fundamentals were not transmitted at an imaginable level to the vast majority of people in the world.

### 3. Studio of art and programmig: a first design

By the year 2000 correspondence with a young uruguayan musician made me aware that the southern cone of latin america could allow another attempt. Good educational and cultural levels. Interest in the Arts. Substantial investments in information technology, yet deep economic crisis and scarcity of resources. Growing marginalisation of unemployed unscolarized youth.

A journey to Uruguay with a kit of resources to run workshops eventually put me in contact with the director of the Institute of electrical engineering ( IIE) of the Universidad de la Republica in Montevideo. By april 2001 we had ran a one month extension course as a joint project between IIE and the school of fine arts and outlined a full credit course to begin at the IIE in September 2001. This implied migrating the "nomadic workshop" from a freer, artistic process towards a "seditary" curriculum and root it inside of engineering. Instead of bringing technology to the artist's studio we were going to bring the art studio methodology to the curriculum of engineering.

Initial design of the course responded to necessities of engineering students in the Uruguayan context but also had to create conditions for their interaction with students from non-engineering disciplines, and as a perspective, explore possibilities of them designing together a low cost mobile workshop they could project outside of the university. Incremental production of a "documentary" of the learning process had to be central in the curriculum.

A weekly class for all participants would review current conditions of Art in the world. Showing a profusion of images of reference works, proposing critical texts on the impact of technology over culture and society, articulating for a very diverse audience a relecture of electro-digital-computacional technology as a visual "landscape" to explore.

Then, starting from a pattern of direct questioning, each "teorico" would expose, in a manner accessible to all, one technical topic: basic control structures in programing. The link between bits as ground level of digital media to programing languages and data objects. How to control time in a navigator using javascript. How to think architectures of microprocessors. What are the simplest electronic circuits? What is an oscillator? Difference between traditional programming and object oriented programming. How to use a static ram-chip.

Essential was the reference to "studio art". Throughout the six hours of weekly presence in the lab, the perspective of Art allowed displacement from the "problems, products, solutions" modality of competitive marketplaces, towards the freer modalities of "symbolic value" production: Use a navigator to make web pages not read from close up for "information content" but meant

to impact as dynamic elements of visual art installations. Cutting up motherboards to regenerate invented but functional electronics. Combine simple network technology with stepper motors to create sound spaces. Things without economic or cutting-edge value but with substantial visual, pedagogical impact. In the situation of Uruguay, it could also show that “something can be done”, in a context of scarce resources, to seed, among a larger number of younger students, a collaborative culture of research and implication, a multidisciplinary “fabric” of invention.

#### 4. Development of the curriculum: TAP1

Although the syllabus approved by the university outlined a series of items that the course was supposed to cover, over the two weeks used by 45 participants ( 15 non engineers ) to install computers, it became obvious that such a syllabus was completely inadapted to the mixture of students. The scant resources available on the student’s computers made it impossible to use classical programming tools. Only Javascript enabled navigators could provide “programming” environments available to everyone . Over the first weeks of the course, we collected a variety of javascript code that students would circulate among them, modify, revisit . Arrays, control-structures, timers, mouse-events handlers, simple command interpreters began to turn into many small, visually expressive projects involving images, text and sound. Advanced students began to program in Java. Small “server-like” applets were embedded into webpages that listen to internet events and soon we had an external client with rudimentary scripting capabilities sending dynamic web-pages across a wall of machines fitted with full screen browsers. Similar structures allowed to control stepper motors from old disk-drives banging steel rods on metal structures and plastic bottles. A rudimentary world of Bits and atoms, low-tech, but with evident expressive impact. Some students produced tutorials about free software systems and basics of java programming. At the end of TAP1, the syllabus presented for academic approval of the course had been reinterpreted into something sustainable, wide open to diversity.

The work had been individual, with a lot of interchanges between all. Final presentations led to animated discussions pointing to the same question: “why not more Art?”. All this had been done in an improvised space of 4x4 meters, in the basement of the institute: making “more Art” there, credible to the outside world’s esthetics, would be very difficult. TAP needed a better space.

#### 5. Making things: TAP2

At the onset of the TAP2 we had a large, empty platform hastily built atop of an obsolete high voltage machine and 150 applicants to the course. The challenge was to introduce “making things” on a much larger scale, without budget for materials, no assistants and practically no tools. The only material available would be circuit-board from motherboards desoldered with heat guns to generate electronic components and solder. This pcb could be cut with a guillotine cutter. We had a few protoboards to introduce minimal “classical” electronics methodology but the challenge was to use cut pcb to produce sustainable electronics projects.

Students assembled computers, and revisited javascript code inherited from TAP1. Fabrication began, with requirement that everyone make from pcb a led-connector to power from a PC-power supply a protoboard or any pcb structure they make. Making this connector was a way to introduce them to simple circuits and to a modularity in making more complex electronics. The goal was a system of small Lego-like zero-cost electronic modules to replace the usual protoboards. By semester’s end we had crude home-made protoboards fitted with oscillators or printer ports drivers flashing leds. Programming tutorials made by students had grown into a method for collective learning. Our presentation framework had evolved into a sophisticated system of java generated webpages. Incursions had been made into VRML and 3D graphics. About 90 students made group presentations and approved the course.

#### 6. Making art, a method for massivity: TAP3

In the fall 2003 we had 200 applicants. Massivity required one more element: how to connect individual achievement to loosely differentiated group work, among people that had extremely different schedules, and, more important, how to maintain relationships between teacher and participants. A natural step was to create a mosaic-like web page displaying individual pictures, with tooltips for their names, each picture linked to their working directory and personal web portal. This allowed interaction with individual progress of students. Definite “projects” were abandoned and replaced by a few “research poles”. Each pole had a list where interested students would add their e-mail.

Electronics pole. Information systems pole. Computational cinema pole. 3D pole. Programming languages pole. Students were encouraged to contact people in poles of their choice and find leadership there. People with skills and projects were encouraged to seek help from less autonomous students. Collective intelligence.

Earlier models of custom protoboards were difficult to make, but had allowed to design modules with a few components. Copies of one type of module, like oscillators, motor controllers, sound amplifiers, were being rebuilt by others, undergoing mutations that made their making easier and cleaner. Custom protoboards evolved into simpler structures. From there on, making microprocessor “development” systems, with memory, connectors for ports became feasible for the electronics “pole” as an incremental vocabulary. A long way from the contraptions of the beginning.

Previous exploration of 3D suggested to work on a virtual model of the lab, from the entire building down to electronics components. Architecture students used Autocad to model the building, computer science students interfaced a database to the Blender3D open source modeller using its python scripting capabilities and electro students created models for chips, resistors, condensers. A “virtual TAP” could now be explored in a web browser from the VRML description generated by Blender and we could use this to design proposals for customised, atelier-labs to be projected elsewhere.

The programming language pole focused on 2D graphics. Making, first with Java, and then Python and Game libraries,

programs for free-hand and scriptable drawing that could store designs as vector objects files compatible with the python readers we made for Blender, thus generating fancy rhythmic computational structures that could be fed in Blender.

The system pole had worked on integration of linux into the lab for low end, older systems. An installation of networked 386 linux machines with very little memory was designed to explore the esthetics of moving graphics across multiple screens.

By the end of the TAP3 we had produced at extremely low cost, elements for large scale, collaborative visual art installations combining a vocabulary of modularized, sculpture-like electronics with their virtual 3D counterparts.

## 7. Conclusions

Earlier years of engineering studies are spent mostly absorbing a steady flow of theoretical, abstract, pre-formatted concepts with little access to hands-on investigation. The creation of conventional research environments is, in most cases, prohibitively expensive.

By combining the expressivity of art and a rigorous demands of technological functionality, TAP opens interesting avenues for bringing younger students to the experience and "socialization" of research., fostering, at a sustainable cost, seeds for what Manuel Castells calls "fabrics of invention"

According to their comments on the course, engineering students, working with students of other, more visually oriented disciplines can benefit greatly from the mutual displacement of perspective offered by the traditions of studio art in the context of electro-digital-computational technology. Using "scripting" capabilities of navigators efficiently introduces "expressive" programming to non-programmers and brings them in working contact with tech-savvy students. Architecture and art students can make useful contributions to fabrication of electronics structures, while demystifying progressively fundamentals involved in the projects. The problematics of social impacts of technology can be shared by all.

Although crudeness of devices fabricated with such "makeshift" means may seem derisory in front of the sophistication of current electro-digital-computational technology, they essentially deal with the same fundamentals: bits, circuits, programing agility, modularity, ability to focus on relevant detail and psychomotricities essential to research. Substantial benefit to students come as they must learn, in a free context, how to integrate large quantity of informations, languages, physical skills and critical points of view.

Massivity, which lowers efficacy of traditional courses, tends to become an asset in the TAP context, provided that minimal communication can be maintained between teacher and students. This means: mailing lists, a mosaic web page of named individual pictures, the simple device of a sign-in book for their presence in the lab. Initial requirements of building computers, and deconstruct electronics refuse while generating documenting web pages by means of writing code, rapidly produces a sampling of skills and know-how of participants. The task of the teacher in this early phase is to identify abilities in the most creative participants and bring their emerging work to the attention of many. Accent is then on identifying students capable of taking natural leadership in the development of the "poles". This loose work-structure seem to favour diversification of the few thematics that emerge naturally into large number of differentiated individual learning curves that can be monitored from the mosaic of web-pages.

Even though many students appear, at first, disoriented by the complete freedom in the work, and, for some, by the simplicity of the tasks, which contrasts with the sophistication of their theoretical courses, exposure to emerging realisations by more dedicated participants eventually pulls everyone into a workflow. At two thirds of the course, it becomes clear that everyone is learning a lot.

Given Uruguay as its location, a useful development of this experiment could arise from interconnecting of similar courses-labs in various universities of the south of Latin America. Such a network could explore a pattern of short-term interchanges for younger students, significantly lacking in this region. The low-cost of the method coupled with uniformness of electro-digital refuse would allow students to make immediate contributions to the spaces they visit, even for short periods.

The need for sustainable re-connection of large social sectors to basic understanding of technology are immense while the costs of doing this are very high, and depend critically of the existence of a generation of trained and motivated "multipliers" . A mixture of engineering and art/architecture/comunication students working together in this environment could provide an efficient training ground for such "multipliers" .

In times of enormous increase in the complexity of our environments, we cannot afford the luxury of not including traditional Art in the diffusion of the fundamentals of technology to the social fabric at large. Curricular spaces of this type, operating from within engineering in collaboration with other parts of the university, would provide an efficient way of exploring this potential.