Gerry

My name is Gerry Gleason. I want to collaborate to create the future we want; one that can sustain humanity and our living planet.

I like to play with both things and ideas. Inspired by great designers and architects, I have a vision of an open future that we can only achieve as a collaboration of our best efforts past and future. My technical career gives me a deep knowledge and appreciation of complex systems, both how they work and how they fail. The change I experienced, from before the PC revolution through the rise of the Internet, is only accellerating and making career choices obsolete before they start.

This site has long been a place where I could play with my ideas in writing, and I am beginning to prepare some pieces for wider publication. I have no crystal ball, only the wisdom from a career of keeping up with technology. I have faith in my fellow citizens, especially the next generations who have only known this accellerated pace of change and must deal with crises just over the horizon. Let's fix this together!

Abbreviated Bio

- Born and raised in Chicago, Lane Tech H.S. 1978
- Builder and Hacker: Shops, Kit computers, VLSI design
- BSCS MIT 1983: Self-financed education working in PC Industry beginnings
- Long Career as a system engineer, developer and architect
- Ontological Design Course sparked renewed and broader inquiry into design and architecture
- Married twenty-five plus years with two grown children
- Works in design and leadership towards radically new currency systems
- Major code contributor to this platform. Wanted to add XML formats for currency systems, and recent coding experiments to connect to crypto currency data structures
- Organizing for change. We can design and achieve a sustainable and just future if we make it a collective mission. Whether we achieve it will depend on how we organize for the future. Everything I have learned up to this point says that we must shift our social architectures, and that we have the collective capacity to do exactly that. All we need is the collective will.
- Commons Based Peer Production: thinking and writing about financial infrastructure for the Commons

The links below are to other Decko *cards*, the unit of content of the system. They use features of the software to enable you to open their content in-line or as a page on the website. Other links are to File that can be downloaded, which are yet another Decko feature. Next you will find resources in this "Wagn" (now named Decko) instance.

I taught myself Ruby on Rails, and I have written some rather large change sets on this open source project, so I may be biased in claiming it is the best content management system available. Fundamentally Decko remains community editable as a Wiki, but with support to make some content more restricted or be more like a database application.

Unless you are logged in and have a special Role, you won't see any content in the Draft cards. In this way I use the features of this tool I helped create so I can write drafts with a small group of collaborators and later publish them by simply changing the Cardtype to Basic. You might also note the plus (+) at the beginning of references to these cards. This means they are related to this card, Gerry, which is both my User card and personal home page.

Recent Writings

Although this site is also an invitation to collaborate, it is first a place for me to write and think. Below is an evolving list of papers that are complete and edited enough to publish here. As noted above, the Draft type card is a good place to see works in progress if you are already a collaborator.

+writing

Currencies for Commons Based Peer Production		
The Commons of Retired Game Masters		
Supply Chain Commons		
Open Letter to Nathan Philips		

Books in Progress

Some of the Book here were initiated by others, and all are works in progress.

I am mostly self-taught and thrive on the availability of learning materials on the Internet. I want to share my knowledge of systems to inspire the next generation. If I have the opportunity to teach this in a hands on collaborative environment, this would be what I would offer to get started. The vision would be collaboratively building this out into on open learning platform connecting resources and organizing and recognizing knowledge and experience.

Bits and Things

In this book I'm connecting what I am learning about systems of all kinds. This is before I read Merrill's "Signs Grow: Semiosis and Life Processes" in which he confirms and goes beyond my intuitions about extending Peirce.

Signs of the Times

This one is most relevant to my recent writings on commons and new currencies. I have not revisited this project with my more recent thinking in mind.

Open Stewardship and the Commons

My partners working on Decko want me to focus on smaller projects to promote and teach what the platform can do, but I can't help wanting to tell a larger story to systems architects about this platform's potential to transform entire application spaces.

It's Cards all the Way Down

Below is more detail related to the short bio above, which can be found in each of these. The first note I wrote a number of years back when I was first working and thinking about new currencies and how to make a career transition of it. This was long before blockchain became a household word for some.

+Process Architect **Objectives --** To work as a process architect and designer to structure our work together as a living collaborati...

This is my technical resume. I've got the background but not the job history to be a great team lead and manager for an organization who wants more unconventional leaders.

+Software Engineer

Objectives - To find a permanent or consulting position in a dynamic organization where I can apply my multidisciplinary experience to design and implement network and open system projects, create development environments or deploy networked systems. I am particularly interested in positions that have team leadership, business development or management opportunities.

Specialties and Abilities - I have an exceptional capacity to understand systems design issues in depth, broad experience with open systems and technology integration, and the communications skills to address technical and nontechnical audiences. Although primarily a software and operations specialist, I have the extensive hardware and networking knowledge and the experience to call myself a true systems architect.

Education

Massachusetts Institute of Technology, Cambridge, MA.

1983, Bachelor of Science Electrical Engineering and Computer Science; including Compiler Design, Systems Design, Computer Architecture, Number Theory, Computational Theory, Economics, Linguistics, Computational Linguistics, VLSI Design, Digital Lab, Knowledge Based Systems.

Skills

Development Scalable Web Architectures, front end and back end. Design Patterns and Agile workflow design. MVC architecture (Ruby on Rails, Wiki and Wiki Federation with RESTful APIs) System Engineering Deployment architecture, methods and tools. Performance evaluations, modeling and architectural analysis. Operating Systems Linux, Solaris, HP, AIX, Mac OSX, Windows Languages Ruby, Go, C, C++, Java, Perl, Python, all shells, variety of assemblers Software Git, Subversion, ClearCase, ClearCase MultiSite, Apache, Jboss, Websphere, Tomcat, Java Web Server, Oracle, Sybase, Postgres, Mysql System/Network Architectures LAN/WAN: Switches, Routers, Hubs, NAT, Firewalls and Network management tools. Storage Solutions: RAID, NFS, SCSI, NAS and SAN. Leadership Led eight developer team. Leader and innovator cross-function teams to design strategies and processes. Led operations for critical systems. Proven ability to meet goals.

Experience

January 2018 - Present - Software Architect

Building a blockchain based alternative currency system. Learning Go and writing some exploritory code to implement signed messaged chains in Go (https://github.com/GerryG/authchain). The application for this and direct use of ipfs (Interplanetary Filesystem) protocols is to create an implementation platform for alternative currencies.

August 2013 - January 2018 - Senior Software Developer, Tribune Technology

Working on a team maintaining and enhancing a Content Management System for digital publishing. This system is used by producers and designers throughout Tribune's many affiliates to organize and publish content. In addition to RoR and Javascript development, I was lead developer working with operations to move many of our systems from in-house hosting to AWS systems including the development work to build, test and deploy with docker containers.

August – September 2012 Ruby on Rails Architect, Argonne National Labs

Created a new Ruby on Rails application from scratch. The Argonne team was creating a new RoR application, and wanted help getting the application created quickly using RoR model generation, and to be able to support custom UI components based on ExtJS (sencha.org) libraries and components created locally to handle things like user authentication and roles. In two months we completed a complex data architecture (over 15 tables and more associations), and I completed all the data models and prototyped the key UI interactions using Json (REST) APIs on the server side and ExtJS components and objects on the client side.

October 2010 – Present Lead Wagneer Grass Commons (sponsoring organization for Wagn http://wagn.org/)

Core developer on Wagn.org, a wiki with tagging written in Ruby on Rails. Wagn is a significant innovation in both the wiki space and as a platform for non-tabular data (objects) in a rich internet application environment. I was initially interested in a back end for experimenting with RIA applications with Adobe FLEX and similar platforms and I added XML rendering and a RESTful controller to serve XML data for the RIA app to access. With these XML extensions in place, I also started to develop a Metacurrency extension to the proof of concept phase when it became clear that Wagn would need some major refactoring to really integrate these features. I started then to do a major refactor of the view/rendering system to support the XML renderer as an option. This work became the 1.5.2 release of Wagn, the largest incremental change to date, and the rest of the work to support the Metacurrency extensions is soon to be released in a series of even larger group of improvements.

I went from a Ruby and Rails beginner to lead developer in about four years. With Wagn, we are bringing object oriented (OO) models to the end user platforms. Wagn, like Drupal and others, is a platform, only Wagn is completely OO to the core, so instead of adding largely independent plugins, you extend the Wagn model (cards) with an MVC api. We have just migrated to Rails 3, and will be releasing the full MVC api and platform over the coming months.

February 2010 – November 2011 Project Lead Chicago Mercantile Exchange (client), Object Wave, Chicago (primary contractor)

Objectwave maintains and enhances a system developed for CME by a third party. As the project lead, the work began with taking delivery of two trading systems from the third party. We started from bare hardware and OS and had to assemble all of the tools to build and deploy the system, a distributed system with about 30 open source and commercial software dependencies. The user interface components used Microsoft technologies with C#, .Net and ASP components, and I had a team of developers in Mexico who worked on these components. The Linux (RedHat) server components were C++, Python and Sybase stored procedures (SQL), and I was the sole developer for these components. The project has cleared all major milestones and is prepared for production roll out.

August 2006 – January 2009 Senior Applications Engineer Orbitz Worldwide, Chicago, IL

Orbitz is one of the top online travel bookers with a complex, high transaction volume Java based distributed web application. Primarily responsible for all major and minor deployments of an interconnected suit of application and data services on three java container platforms. We configured and deployed to environments for system testing and readiness and finally in production. Created tools to deploy and manage production deployments. Worked to bring new processes to our work methods to encourage innovation. Created Open Tools Space, with wiki pages and regular times designed to be filled by the participants.

April 2005 – March 2006 Senior Systems Engineer Peak6, Chicago, IL

Peak6 is a proprietary equity option trading firm in Chicago. Created a staging/QA environment from scratch and designed all release control processes and systems. Replaced developer managed manual releases with simple formal processes to verification of deploys and system readyness. First releasing a complex market specialist automation system involving 8-10 service processes running on 8 (staging) to 12 (production) hosts, managed all staging and production builds and releases of this system as well as several more major trading systems. Implemented and customized an open source issue tracking system (Trac) to track bugs and release requests, and a connected Wiki system for support and design documentation. Lead participant in a team make the internal Wiki into a key resource for the company.

July 2002 – Principal Technologist & Process Architect GeraldGleason Dot Com, Chicago, IL

Creating a consulting practice to serve emerging global wisdom driven networked enterprises with design services in technology and process arts. This emerging practice is based on the design philosophy of Christopher Alexander in architecture, widely adopted in the Pattern Language processes of object oriented software development, and now into organizational design and development. I am part of a network of pioneers who are creating and exploring pattern languages around group conversations and collaborations. I have an outline of the works I have participated in, and the path of my development on this line of work, which fully occupies me in through the gaps in my technological work.

November 2001 - July 2002 Senior Developer/Team Leader Geodesic Systems, Chicago, IL

Geodesic systems was a software technology firm with advanced C/C++ memory management technology. (Their Great Circle and High Availability products were purchased on continue in the Veritas product lines) Lead a team of 8 developers working on an advanced memory management product for C, C++ and Java language environments. We completed a final major release of the current product generation, then, planned, designed and implemented a new generation. This design took the product from being primarily a tool to improve the performance and reliability of C and C++ programs by improving performance and automatically fixing memory management problems to make it into a managed runtime environment for C, C++ and Java systems. I coordinated the design and planning of the entire project as well as implementing parts of the system. My largest piece was the communication sub-system that enabled management agents to communicate via shared memory queues with our runtime libraries to control and monitor running user processes. I also implemented an advanced build and test scheduling tool in perl to efficiently manage the running of nightly builds and smoke tests on all platforms (Windows, AIX, HPUX, and Linux, with multiple versions of several).

March 2000 - June 2001 Critical Systems Architect Perceptual Robotics, Inc., Chicago, IL

Responsible for design, implementation, management and monitoring of all production and pre-production systems. Sun/Solaris servers running JWS (Sun's JavaWebServer) and a Linux server running Oracle 8.1.5 were replaced with more Suns running Apache/Tomcat and Oracle 8.1.6. We evaluated storage options and settled on a NAS (Network Attached Storage) solution. Our evaluation and testing uncovered some latent problems with drivers or filesystems in a low cost Terabyte solution which was Linux based. A Network Appliance solution was then implemented in production. I designed and implemented a three-legged firewall using Linux and ipchains, and taught my junior colleague so that he could maintain and replicate it. Together we moved all or our systems from Evanston to our new offices in the West Loop including building out the network with Cisco routers and NetGear switches. My day to day responsibilities included managing the installation of new versions of our Java based web server software on test and production systems, building and maintaining systems administrations scripts such as rotating and gathering web server log files, installing Solaris software and support tools on new systems, configuring and building public domain tools and maintaining internal and external DNS and other network support systems.

September 1995 - March 2000 - Senior Consultant - Collective Technologies, Austin, TX, NetDox, Inc., Deerfield, IL, Board of Trade Clearing Corp., Chicago, IL

On assignment at Motorola's Schaumburg facility, I was the site manager for ClearCase/Multisite for about 10 teams of developers totalling over 800 Sun/Solaris workstations with ClearCase installed. Initially I shared this responsibility with a second engineer, and had primary responsibility for upgrading all workstations and servers to a Y2K compliant version of ClearCase.

On assignment at Warburg Dillan Reed in the Chicago Board of Trade building I performed as a short notice backup admin. I was able to quickly come up to speed in a complex environment and provide many hours as the admin on the help desk.

At NetDox, I immediately assumed primary responsibility for putting the new release in production, which was more successful that previous releases even having lost the knowledge of the first round implementors. We build a new test configuration including complex network and firewalls. An important part of the operations of this secure messaging service was the management of the cryptographic keys used to hide the text during transmission (PKI). I reviewed the documents and procedures related to key management and key recovery and updated the systems, scripts and documentation required to maintain a secure service.

At BoTCC, I made modifications to their margin calculator which is part of a C++ based object oriented system. Used Object Modelling Language diagrams to describe the changes needed to add foreign currency conversions to the margin calculator, and implemented these changes to the C++ code. My release package for this project was a good model of our new release process which included sections from design and code changes through tests of the business logic correctness and operations document changes.

At BoTCC, I started as a Sun/Solaris system administrator wuth responsibilities including NIS+, firewalls and TCP/IP networking. Selected and performed initial implementation of ClearCase for source code control. Wrote perl scripts to support ClearCase and release control process.

Jan. 1993 - Sept. 1995 Systems Database & Network Administration Consultant

As a consultant at Swiss Bank Corp. (same organization mentioned above as Warburg, Dillan, Reed while I was with Collective Technologies). I managed a group of test machines that included about six sybase servers and fifteen plus compute servers running locally developed three tier client server applications. This job involved a lot of DB work (Sybase) to maintain the systems and load production data for testing/auditing. I also set up my first web server and wrote many perl/CGI programs to support our admin work. This environment was SunOS and NextStep, with some conversions to Solaris by the end.

1988 - 1992 Network Systems Analyst & Systems Engineer

Zenith Data Systems, Buffalo Grove, IL & St Joseph, MI

UNIX specialist in a network marketing group to help sell Zenith PCs in LAN environments. Helped create a network demonstration lab with all current network technologies. Created client configurations to allow PCs to access multiple server types simultaneously. Configured and supported a similar network for trade shows.

Prepared technology for a bid for a large government contract, a leading edge multiprocessor platform and supported SCO commercial UNIX system as adapted for multiprocessing. Wrote UNIX device drivers for SCSI host adapters and simple I/O options, managed an outsource project for custom filesystems on optical drives. Integrated these custom drivers to create a custom Interactive UNIX OS release to meet bid specifications. My hardware and architecture background was also tapped for technologies planning efforts including a multidisciplinary team given the task of creating a business plan on a network technology strategy. Our recommendations for network marketing and product strategies.

1985 - 1988 Systems Engineering Consultant

At AT&T in Summit, NJ I joined a team that created an advanced print spooler subsystem that was release with AT&T's System V release 3.2. At AT&T in Naperville, IL, I ported the UNIX kernel to a new microprocessor (CRISP, later know as the Hobbit). Worked directly with the hardware engineer to make this kernel run on first silicon. Solved our bootstrapping problem by adapting the firmware for an intelligent serial card so that it could download directly into the host machine's memory.

Earlier Experience Available on request. References available on request.

These should have the same content as downloadable files:

+Resume

Download Gerry+Resume

+PDF Resume

Download Gerry+PDF Resume

This artist, who created caricatures of everyone who attended this conference, generously gave us permission under Creative Commons terms. Here is the one she did of me:

The conference was created using Open Space Technology, a process architecture I had recently experienced for the first time. This and another architecture proposed by someone who inspired me to begin to inquire into Collective Intelligence, Wisdom and Consciousness (CIWC) have caused me to think more deeply about issues we are faced with. How would you produce it, enhance it and embody it as Global Wisdom Driven Organizations (GWDOs)?

These ideas of architecture and design that I have been thinking and writing about have allowed me to see the importance of certain ideas in a broad range of fields. Christopher Alexander's concept of the Pattern Language in architecture has been applied in many fields, and I have even had some experience with communities trying to apply it in their field. It is a challenging task in any field to do justice to the process, and so we must be both critical and patient with the development of pattern languages.

I also discovered that architecture, as a field, is deeply engaged critical thought down to the bedrock of philosophy. Although I don't know how pattern languages relate to Peircaen Semiotics except through the universal processes of inquiry that drove Peirce's thoughts in philosophy and logic, I intuit that there is a deep connection between Peirce and Alexander's ideas.

Alexander grounds his thought in the language of builders before architecture, when most of the knowledge was passed on in the work of building and transmission of the physical practices of building. They are united in natural informational processes (signs, semiotics) that are the backdrop of all natural systems from cells through complex organisms to the social evolution of human symbolic practices. I also offer this biographical piece to frame my identity from a hacker's perspective, one of play and exploration that has just begun.

A Hacking Life

Up until this point of my life as a hacker, my potential employers have wanted no more than a summary of my technical feats and accomplishments. Although I have written about code, designs, plans and the coordination of implementations, there is no history of publication to cite as part of the currency of a thinking life. Therefore, I will tell my story about how my career has been grounded in experience of both practices of making and designing physical things and the arts of craftsmanship and the design, implementation and operations of digital systems.

My path is that of a hacker. Originally I started on this path because I would take things apart to learn about how they worked. I come from a working class background with parents with some education and some success in school. I was near the top of my class at one of the best high schools in Chicago. Because of a referral from my machine shop teacher, I landed a summer job in a machine shop. The next school year I learned Fortran and then about micro computer kits. My next summer job was programming in BASIC, which I taught myself from a book I took out of the library.

Susbsequently, I received a scholarship from MIT, and got my BS in EECS in three years out of five. I had to take a break to earn some money before getting my degree in '83. Along the way, I built a kit computer and started with a PC non-compatible company the same month the first IBM PCs came out in 1981. When I returned to MIT, I was in a small circle with the founders of the free software movement at the time when the GPL was created.

Looking back, I was never just a hacker; I also wanted to know how systems worked on all levels. Even earlier, I had a dream of becoming an architect. My mother was an artist, and I grew up near the studio home of Frank Lloyd Wright and the many homes he designed. I felt a need to inquire into everything much more deeply. I didn't want to just hack; I wanted to be a systems designer and architect. The title of architect in systems implementation doesn't necessarily reward the most far thinking approaches. The MIT environment was great for contextualizing what I had already learned as a hands on programmer, and started me on a path towards general systems thinking. Since I already had practical experience with systems, I came to see that I had a big advantage over most of my MIT classmates.

Although my degree concentration was on the CS side of EECS, along the way I designed a chip in a graduate VLSI design course. As a consultant at AT&T labs, I made UNIX run on a new chip design, and later at Zenith as well. I used my hacker interests to bridge my software roles into hardware, systems and network architectures. The more different kinds of design and architecture that I learned about, the more I could see the parallels and commons structures. All design is ultimately grounded in human needs, and architects have to study people and their social architectures just as much as we do engineering and science. I needed to learn more about philosophy and language. I knew from cognitive science that we must also ground this work in new systems thinking in biology that understands the organism as a network of relationships.

Systems, Signs and Autopoiesis

It wasn't until I connected to the work of C.S. Peirce in Semiotics and his theory of categories and signs that I could connect more deeply to what I had learned years previously In Fernando Flores' "Ontological Design Course." During that course, I was introduced to system biology in Maturana and Varella, and I realized only later how Peircean signs were a critical part of their theory of autopoeisis. I have begun to explore some of these ideas in writing, but I am no academic background and finding my audience is a bit trickier. I cite this story as background towards collaborative innovation in design and architecture in the service of humanity.

Analogies from systems of all kinds as well as the overhyped recent progress in AI suggests that evolution and the emergence of new systems levels as creative acts of evolution are common to and at the core of all systems change. Peircean semiotics and sign systems are an important and overlooked tool to unite our thinking about systems of all kinds. It is also clear that the work and context has advanced to the point that irresponsible applications could pose significant risks very soon, and we probably wouldn't know it publicly for some time if it did. There are similar risks in a number of fields, e.g., genetic engineering, and all of these would only be amplified with the advent of strong AI. Although I may not be able to write papers that advance a highly specialized field, I would claim this is exactly the kind of broad knowledge that is needed so critically to sort out the hype and realities of current risks as well as opportunities.

My practical experience in systems, as well as my life's path through organizations large and small, has shown me that much of the dysfunction we have produced in our world must be rooted out if we are to not only survive but thrive as human beings. It isn't *human nature* that constrains us, but our social systems as they have developed historically. We are not condemned to repeat our past mistakes if we can learn from them what we need to do in order to create new social systems. We must temper the enlightenment urge to overturn all of the traditions; progress can only be made within traditions. However, we must not, at the same time, lose the insights that flowed when the grip of tradition isn't mixed with power and authority not grounded in the traditions of knowledge themselves.

I offer this history and background as a point of embarkation on the rest of my life's work yet to come. Like this personal history, I have works in progress and some bits of completed thoughts to share. I hope this shows what I can write for the future, and what I might bring to important design conversations that must be collectively convened for the sake of all of us. I bring many more questions than answers, but I am confident about what I have to contribute to collective works.

An Offer To Serve the Critical Missions of Humanity

I have so much to offer to the benefit of many, and no adaquate access to the economic means to do it. Sure, I'm employable, but as with too many of my contemporaries, that way is neither sufficient for the family nor does it leave me free to produce the most possible value for many of us. Like many others like me, most of whom don't have many advantages that I have had, this is no way to live. It can be hard to even keep your head up enough to keep at it because it is drilled into us that our failure is our fault and our fault alone.

I don't accept this because my education and experience tell me that we, as an emergent global culture, face a number of immediate crises that will take all the collective knowledge and experience that we can muster. The crises are not created by the suffering people, but by the systems as they have evolved. They may have evolved to function reasonably in an historical context, but, at this moment, the system itself is deeply implicated in the dysfunction. The story told about it may once have been true enough, but now has become a cover for injustice and blaming the victim.

With my background, I can and will serve one or more missions in the future, but I can't do it alone. None of us can. We need everyone of us. Most likely a future me will look back at a path with many surprises. My education and experience tell me that collective success will depend much more on learning to form and act from the context of much more powerful collectives. Not just AI as in artificial, but more towards augmented intelligence. We need to reach beyond just intelligence and attempt to understand and create collective wisdom and consciousness. As scientists and engineers begin to work with collective intelligence, they are naturally asking the next questions about these next steps.

I'm well qualified to ask good questions about how to survive the singularity. Survival cannot be a limited mission, and division is failure. Can collective missions to save the living planet really work? What kind of collective vision is necessary for collective hope and to fight collective dispair? I do have hope, even confidence, that humanity has the collective power to choose our path, and therefore to choose living systems over dying and killing systems.

+print friendly

Download Gerry+print friendly