**Engineers do it First**

Scott Foerster,
Assistant Professor of Engineering,
Howard Community College

*Engineers do it First* by Scott Foerster, *Assistant Professor of Engineering, Howard Community College*

Abstract "Introduction to Engineering" courses introduce a university's unique programs, projects and tools. Ethics and privileges such as 24 hour by 7 access to tools, materials and rooms are negotiated. Community Colleges don't have the resources, graders, graduate assistants, and multiple professors teaching the course. This paper lists the problems of teaching "Introduction to Engineering" at a Community College. It argues that truly open ended projects can be taught to freshman. It proposes an external assessment process as well as internal grading system. It is inspired by DIY, maker/hacker spaces, wikibooks, and wikiversity. It starts from a new narrative of playing, doing things first, design and problem solving. "Introduction to Engineering" is where students learn "Engineers do it First."

Outline

Narrative Problem
Freshman Problem
Open Ended Project Problem
Course Management Problem
Internal Grading
External Assessment
What's Next?

*The Narrative Problem*

The current Engineering narratives are "Applied Science", "Solve Problems", "Make Things", "Design" and "Engineers help shape the Future." Do these narratives work? How many kids say they want to be an engineer? Who is they as in "They just created a new wire with carbon and no metal or copper." Only at community colleges do students declare an engineering major that can not pass an algebra placement test, want to move knobs up and down in a round stall b, already have some other undergraduate degree (like physics, or architecture), or are challenged in some way

Engineers live in a business to business world (B2B) where both clients and customers are other engineers. Doctors and lawyers interface with the public every day. The public has a strong narrative associated with most occupations. Engineering draws a blank.

Why aren't current narratives working? Look at the origin. The B2B world provides no ability to test narratives. Retaining freshman interested in engineering at four year schools is a different objective than explaining engineering to the public.

## K-12 "Trust Science" Narrative Dominance

The US military preferred to work with physicists rather than engineers during WWII. Engineers belonged to unions, not societies. They were either superTechs or tech managers. "Art and Practice" dominated the engineering profession. A deliberate decision to begin promoting Engineering Science was made during the creation of the National Science Foundation (NSF). This has been a success. Engineering undergrads now drop out to become physicists. Less prestigious colleges offer Physics instead of Engineering. Physics is cheaper, easier to staff and can simulate the efficiency/assessment demands of modern accountability.

The K-12 "trust science" narrative inspires education that discourages magic, establishes an understandable value set and promotes the "scientific process." This narrative does create a public that votes for a pure research infrastructure. Changing this would be dangerous. The gap between a needed "creating scientist" narrative and the K-12 "trusting science" narrative has now fallen on the shoulders of engineering.

## The Technology Narrative

Engineers face a real danger of becoming technologists. Engineering Technology programs fill a real need. They provide a career pathway between technology and engineering. They establish mutual respect. Unfortunately current engineering narratives blur the engineering/technician distinction.

To fight this, most colleges are either focused on engineering science or engineering technology. K-12 and community colleges have both. Most of the articulation problems between community colleges and four-year institutions exist because of technology narrative problems. Four-year institutions believe K-12 and community colleges don't implement the "create scientist" narrative, have to train technicians and thus cannot simultaneously support engineering science.

What is the technology narrative? The industrial arts, technician, technologist, certification, apprentice, master occupations can all be lumped into one narrative. "Do something for 15 years." Become an expert. Experience creates expertise. This is not an engineering or "creating scientist" narrative.

## Engineers do it first

The design requirements for an engineering narrative are: put "hands-on" in the right context, sandwich between "create scientist" and "technology" narratives, fit on a bumper sticker, and be typical guide for all engineering courses, and foster engineering projects.

The "Do It First" narrative emerged while sinking into the ugliness of "Design" and "Solve problems" narratives. It is the fruit of students suffering through process without relevance.

What is "it"?

- It extends and matures through the group play to ought "exploring."
- It values ignorance as much as experience and expertise.
- It creates inspiration that can't happen in education where the teacher says "Be like me."
- It sets the stage for an emphasis on repeatability and documentation.
- It matures into design and engineering problem solving versus technician problem solving.
- It can lead to the scientific narrative of theories, instruments and experiments

Asking the question "Who gets to do it first?" establishes the currency of engineering: RESPECT.

The full narrative is Play → Do It First → Design → Solve Problems.

*The Freshman Problem*

## Respect versus Expertise and Experience

Freshman engineers are crippled by the K-12 narrative: "experience creates expertise and ultimately respect." Today, first hand, direct experience is expensive. Expertise is now routinely gained from simulation. Ignorance can have as much value as expertise. The order has been reversed: demonstrate Respect, build some Expertise and then reward with direct Experience. The currency freshman engineers need to be taught is Respect.

## Push off versus Pull Up the Educational Mountain (expertise)

K-12 requires instructor expertise. But it cripples the self-motivated/self-learning that everyone is naturally born with. For example, a freshman team is tasked with building a toy robot boat. One person decides to design the hull. They feel their first task is to find a hull design expert or become an expert.

Many engineering programs combat this with "scaffolding" which merely reinforces education expertise rather than challenge it. Scaffolding homogenizes, puts content before creativity and provides no future skills other than exhaust yourself/do your best.

What is the alternative? Meet the world as it presents itself. There are toy boats. There are soda bottles. Pick one. Put it in the water. Attach the motors and radio controls. Create an on time, functional, attractive deliverable. Earn engineering revere respect. Modern adventures are in junk pile explorations. Let the project dictate the content.

Engineering education must become comfortable with pushing students off the mountain into unknown waters (for both student and teacher) and learning to swim on the ir own. We have to teach against the K-12 archetypes rather than leverage them. We have to never work harder and never know than our students. We have to resist doing the project ourselves. We must wear the client/project manager hat, not the student's engineering hat.

## My problem versus atoms (timidity)

How does an engineer gain the respect that will attract investment when nobody else knows? If the engineer calls the ir own personal lack of knowledge a "problem", there will be no investment. Engineering will not happen. The simple answer is to focus on service. "I am going to serve you by building a radio controlled model boat on time, that works and it will look gorgeous."

There are three paths an engineer walks. One is the path of exercising expertise. This is the public face of engineering and it unfortunately fits the K-12 education model s. But it is not the focus of engineering. The second path is exploration, learning about something that will improve the species. The third is learning what one doesn't know, something new to the engineer.

For an engineer, expertise is a fleeting, momentary consequence that disappears as soon as the next unknown is tackled and technicians take over. Expertise is not the goal. Service is the goal.

The typical question freshman engineers are asked ("What type of engineer do you want to be?") plays into the wrong archetype. Almost every engineer will describe their life by their role in a project (sales, support, design, testing, implementation, prototype, management). The non-engineering public expects expertise demonstrations. Do we want to graduate engineering students that will only look for jobs in the fields they have some educational defined expertise .. ie Statics = Bridge Building?

## Big versus little problems (scope)

Why introduce large project issues into freshman engineering courses? Most intro engineering texts describe big problems and then assign students a project such as "Here is a bunch of parts: build a model car." Students cannot connect the small problems they encounter to the big ones discussed.

K-12 does reflections that start with "I have a problem organizing", "I have a problem focusing", "I have a lack of expertise." None of these are engineering problems. Engineering problems are outside of our minds. They are stumbled upon. Some are small, some grow large and engulf everyone.

K-12 doesn't like problems. Problems exist only paired to solutions. The seduction of solutions, the impartiality of problems, the brainstorming of possible solutions, and the rationalization of testing gave the right answers. Most minimize the pain and frustration a problem triggers. Freshman avoid problems instead of finding engineering inspiration in them.

Engineering problems seeds are discovered by individuals. Most remain small and are solved immediately. Some grow. Some explode to many small problems. Documenting the process, collecting justifications, describing attempts and failure symptoms results in an engineering mind. Trying to fit anything into a big problem design mold confuses students.

## Feeling success

Many community college engineering students have experienced life narrowing failure. They have not experienced motivating, confidence building success.

Most introduction to engineering classes have a single project that all students do. Even large colleges with 7000 students, force them all into one project. Some officially label the whole experience a competition. Competition produces success for a handful and failure for most. This supposedly models competition in the real world. In reality it is discouraging to all but the best and brightest.

The alternative is many small different projects, three team members, smaller scopes, truer expectations, and smaller successes with enormous celebrations. The problem is not multiple choice questions on tests. The problem is that we only celebrate multiple choice question test success. Celebrations create Respect. Can we organize engineering departments to celebrate the design of a screw?

## Failure Respect

The hardest part of engineering is helping students find inspiration in frustration and problems. The second hardest thing is teaching students how to generate Respect in the face of failure. The secret is teaching students how to document failure.

The goal of Failure/Can't be Done documentation is to establish enough Respect that others don't attempt to repeat the failure. Typically this requires exhaustion of internet searches, time and inspiration. Any uncertainty or timidity will attract more engineering. All respect will be lost.

## Like and Timidity

Freshman are often motivated by social connections. Without carefully describing engineering team member relationships, most teams fall apart. The need to create social connections may create priorities that compete with their intellectual goals. Stories must be told of Respect lost when everyone agrees on everything, and stories must be told of Respect gained by working through differences.

Students will try to justify underbaked solutions by the K-12 efficiency archetype rather than acknowledge the like and be liked seduction. It is necessary to look for problems rather than gloss over them, to slow down and reword the problem different ways without solution, to sink into the frustration's despair and find inspiration there. Teams don't do this. Individuals do.

Most team activity is individual activity. The team aspect is comparing, fitting, testing and then negotiating how to split up again. Separating Me from I requires being accountable and transparent. It is one of the most important engineering ethical and respect issues. The maturing of freshman is has to begin with targeting the like and be liked timidity dance.

## Slackers

Engineering education grading curves makes those that can concentrate/learn from books and understand lectures those that learn verbally and can shrink complex topics to sound bytes. This is expressed by the saying "A students become instructors, B students end up working for C students." This is scary to community college students who will never transfer with a C.

Slacking issues appear immediately. At least half of all K-12 students learn from the class collective mind rather than from the teacher. Unique projects expose this paradoxical laziness. This has to be spun positively. Turn slackers into communicators. Talk about the other danger of "hard to work with." Identify and celebrate communicator success. Equate communication with documentation success. Both are needed on an engineering team.

This issue is not solved by Myers Briggs tests or even Johnson O'Connor lite rules. The conventional approach that creates slackers and the hard to work with has to be addressed early. Otherwise students will immediately begin labelling themselves and drop out of engineering.

*The Open Ended Project Problem*

The gray area between Open Ended and Canned Projects has been thoroughly explored and named. But there is still a lot of confusion. The definition used here is: students choose among a variety of new/old projects, projects are never finished (endless improvement), and instructors grade form and celebrate success.

## Inspiration versus Content

Education research has shown that students walk away from open ended problems with more inspiration and less content. It is a narrative problem that students don't value the inspiration until after graduation. Engineering is fun when content and inspiration emerge together. K-12, content outcome, and quality discussions are important. But they discourage inspiration. The comparisons fueled by competition inspire only a few. Engineering has to be about adding back the inspiration, not continuing K-12 expectations.

Engineering project grading typically slips into K-12 expectations. Experts lecture and assign homework on content that seems random to the students or supports the single project that everyone does. Neither accomplishes the inspiration objectives of an open ended project.

This is done because there is no model for grading inspiration. The next section proposes how to do this easily. Inspiration can be graded just like in the engineering workplace: through project management accountability, documentation and presentations.

## Project Management versus Content

The freshman engineering class needs to give up content. Instructors should not present themselves as experts. Freshman need to discover content themselves. They need to be forced to learn on their own. Instructors should point to options like a project manager.

The project management approach focuses on documentation, transparency and accountability. The documentation requirement creates a course management issue similar to engineering corporations. The goal is to hold individuals accountable and celebrate team success. Documentation should start off personal, and then move to team presentations and team documentation. Personal success can be withheld until the team's work is done.

## ]Materials and Facilities

Projects need space. Space is normally created by demonstrating need. Large projects that create bits of noise, dust, and debris can justify facilities.

Facilities can only be built slowly. External money and grants normally increase existing success. They don't finance start up programs. Facilities have to be built slowly over time from what students leave behind, from junk and discards of other departments.

Open ended projects don't fit the K-12 efficiency expectations of lesson plans, materials lists and ordering of materials during the summer for the entire year. Efficiency expectations kill most open ended projects enthusiasm and cause white elephant kit purchases. For example, open ended projects always involve searches for materials. Ordering materials requires a justification, a problem statement and confidence respected by the instructor. Ordering materials is another form of success that is missed if all materials are ordered during the summer by the instructor.

## Scaffolding versus DIY University

Project scaffolding is an excuse some engineering schools use to lecture students, force them to purchase text books, and meet expertise expectations. No development of rapid-self learning, skimming, need to know, just in time, or design build talents are possible. Scaffolding kills inspiration, discourages documentation and sets the example of starting from the beginning. Students don't have to find a starting point and then reverse engineer backwards as well as engineer forwards.

Scaffolding denies prior work documentation exists that has to be leveraged. Some colleges deliberately delete all documentation after each semester's work. This creates an unethical, underground documentation system. It forces competition complexity increases. This creates a crippling negative feedback loop. Scaffolding denies freshman the opportunity to wrestle with the start over, catch up, or repeat previous success decisions that all engineers wrestle with. Scaffolding denies freshman scope and scale experiences that moderate K-12 delusions of grandeur.

Maker Magazine, DIY University concepts and HackerSpace successes are stealing the open ended project archetype. The signature of an Unaffiliated Space member is "teach, learn, party." If engineering colleges don't embrace open ended projects, they will continue to grow in DIY spaces.

## Definition of Success

The big problem narrative forces an inflated definition of success that only a few can achieve. Engineering is about solving the small problems along a path of shifting scope, focus and subsystems. A suggested individual grading metric is "pushing the project forward". Celebrating these tiny successes weekly in class is absolutely necessary.

## Context

Projects don't have to be competitions. They can include: science demo, service, reverse engineering, art, improving, profiting, and inventing. The best project context is solving a problem defined by someone outside the Introduction to engineering class ... someone called a "client". This reduces the need for the instructor to wear the client, customer, project manager and engineer hats.

## Good Clients

A freshman engineering class works best when clients are outside the classroom. Ideally these are engineers in the community. Non-engineering clients are a lot of work. College support staff can help in two ways: educate non-engineering clients and become clients themselves.

*The Course Management Problem*

Managing open ended projects in a community college is difficult. Residential engineering colleges with successful open ended freshman projects grow a culture where juniors and seniors mentor freshman and sophomores. Creating this culture at a community college in isolation from the world has been impossible. Volunteers at Wikibooks and Wikiversity could change this.

## Engineering Notebooks

Engineering notebooks have not moved into the electronic world, and may never. The inspiration gained from hand writing compliments that gained by white board writing and word processing.

## In the Moment versus After the Fact

Students initially turn the engineering notebook into a torture device. They try to write perfect summaries or reports. This eliminates an inspiration gained by writing in the moment. The uncertainty fog, the minute detail, the tiny but significant decision rationalization disappears unless captured in the moment. A summary is a litany of lost opportunities and lost information. Writing in the moment creates more volume, better design, better problem solving, less play, .. better engineering. The reasons for writing in an engineering notebook have not changed. Writing in a notebook creates an extension of the brain.

## LMS

A Learning Management System (LMS) is a useful tool. The problem is that all LMS information is controlled by the college. Colleges kick students out of the LMS for not paying their bills. Colleges delete old course information, making DIY university portfolios more difficult. Newer open source LMS systems (Canvas) do eliminate the

college's storage volume and backup issues. But ultimately the student loses. Colleges will never give up their ownership, censorship, and reuse control of information.

## Wikiversity

Wikia space has changed the educational narrative. Content is polished through enthusiasm and loose consensus. Wikia space separates information from its organization. It depends heavily on search engines. Search engines require knowing what we don't know. To learn what we don't know, wiki organizes information using a category concept. This makes it possible to simultaneously document a variety of learning paths. The flatness of Wikipedia illustrates the focus on information. Wikibooks are merely extensions of the information with a lot of categorizing. The categorizational chaos of Wikimedia and Wikiversity is where the future of education can be seen.

## Wikia Promotes Reuse

Most education is trapped in a disclosure, copyright, fair use, create your own work narrative that is increasingly divorced from the modern world. Somebody somewhere has already done it and it can usually be found before a sentence is finished. Finding and combining that previous work emphasizes open ended projects, reuse and Respect.

Students need to work in a place where reuse is encouraged, where everything they create is put in the public domain. Some students want to improve the world, and yet don't want to worry about patents, copyrights, trademarks and fair use interpretations. Creativity doesn't require starting from scratch.

## Creating Value in Public

Students will create electronic documents from other's work and try to pass it off as their own because of LMS privacy. The minute students are asked to create something that can not be deleted, that will be around the rest of their lives, that will be searched by potential employers, that can change the world, that can establish respect, they start behaving professionally.

## Badges of Respect

Web sites reward desired behavior with points and badges of respect. This is no different than the Boy Scouts, fraternal organizations or the military. The success of badges has been proven by the Gates Foundation and can be seen in its full implementation at the Kahn Academy. Wikia sites have similar rewards. The engineering profession needs to extend Wikia rewards. Engineering awards started with FE, PE certification need to replace the increasingly awkward, random content test (despite how statistically significant it is).

## Wikia Space Introduction

Wikimedia includes Wikipedia, Wikibooks, and Wikiversity. The interwiki_map lists a variety of wikias, both for-profit and non-profit. Within wikia's there are spaces. The most obvious is article space. Every registered user gets a user space. There is no privacy associated with any space. Anyone can edit pages in any space including help, category, and file space. Both the content and the file storage type must be public domain in order to be uploaded.

### *The Informal Grading Problem*

Many different introduction to Engineering grading systems have been tried including problem/solution, all or nothing team, milestone submissions, time spent, and writing volume. All have problems fitting into any kind of engineering narrative. The "doing things first narrative" fits a portfolio context. Individual portfolios graded by instructors is promoted below. Team documentation assessed by outside experts is more appropriate for program/course assessment.

## Portfolios

Individual students have been building individual portfolios in the PLTW EDD class. The EDPPSR (Engineering Design Process Portfolio Scoring Rubric) is being used to encourage Engineering schools, the College Board and ABET to increase:

- admissions into other project-based programs;
- admissions into post-secondary studies;
- career pathway recognition; and
- advanced Placement or dual-credit into more rigorous academic courses.

The rubric doesn't measure teamwork documentation, commitment, transparency, contribution, integrity, or persistence. The rubric has tried to create an information control point by establishing a web site to upload portfolios. The portfolio is forced into a one size fits all mold so that an "efficient" assessment process involving outside experts (College Board) can be created. Its development has been dominated by technology and K-12 personnel, not STEM. It doesn't separate individual student grading (feedback on daily/weekly basis that forms the basis of gradual improvement) from program or curriculum assessment (goals are all individual, portfolio is individual). The rubric depends heavily upon subjectivity. EDPPSR hopes that there is a statistically significant, (but as yet undiscovered) "ideal portfolio" will be within the expert engineering portfolio evaluators. The positives of EDPPSR are that it focuses on documentation, raises assessment questions, and is general, not focusing on a particular technology.

## Team versus Individual

The words "teams and projects" lead students to think of sports teams, winning and loosing. They are surprised when there is an individual grading component in a "project" class. The first step in grading an "introduction to engineering class" has been to create a mechanism to separate "We" and "I."

Introduction to engineering students want to socialize. "We" will happen with no effort by the instructor. The focus has to be on the individual. Engineering problems need to be separated from personal, "life gets in the way" problems. Ignorance needs to be turned into a problem solving asset. Technician troubleshooting needs to be separated from engineering design. There are lots of "I" issues that start with individuals defining and solving small engineering problems themselves. Open ended projects are merely opportunities to explore small engineering problems.

## Atoms versus Documentation

Students learn through hands on experiences. Given enough stuff, students will begin playing. If only the final product is assessed, then a goal of moving students through playing, doing things first, design and problem solving is not addressed. Students can and will spend enormous amounts of time .. playing .. to get something to work. If this is all that is rewarded. The goal of the grading has to be encouraging students to stop and design; to stop and define engineering problems.

There are two ways to get students to stop. The first is to force students to write before, during after they do something of their choice. This improves handwriting, forces carrying engineering notebooks around and captures level of chaotic detail and inspiration that is missed in after the fact, summary writing. Currently this is being done through the GoingToDo, Doing and Analysis triplet. This evolved from the science triplet of Hypothesis, Procedure/Testing, Conclusion.

The challenge is to get students to Analyze. The Reflection of EDPPSR has too much in common with service learning reflection (personal life reflection) and not enough in common with testing, analysis, and conclusions. Analysis is described as answering questions such as: what went right/wrong, why something went wrong, what/where is the error, if had more time would have ....., a better tool would have been ....., if could do over again, would have ....., if could automate, would improve ....., this was expected, and this was not expected.

Three points are given per completed project triplet of "GoingToDo, Doing and Analysis". One point is given if there is no Analysis. Students generate 3 of these triplets per hour on average. This is called project work in the notebook grading system. It leads to design.

Six points are given per completed problem triplet of "problem, possible solutions and testing" that can get 6 points. This quickly raises the issue of "What is an engineering problem." "I can not pop all the balloons in the Pogo.com game Poppit!" is not an engineering problem. "Can all the balloons in each game possibly be popped?" is. Single solutions are not rewarded. Students must brainstorm possible solutions. Testing is determined by context and has to exist. Working through these issues matures student engineers.

## Time versus Accomplishment

Notebook writing gives a clear indication of time spent. Electronic documentation emphasizes accomplishment. Both are needed. Accomplishment starts off personal and passes from teammate to teammate like a hot potato, each adding something personal. To capture this process, four levels of electronic documentation are needed: weekly personal, weekly team, team summary, commons contribution.

Weekly personal documentation is graded with push points (accomplishment points) that range from 0 to 100. Weekly team documentation is graded 40-80 points, but only if a short presentation is made. Team Summary documentation is motivated by withholding project points from the notebook and push points from individual grading until the Team Summary is done. PLTW promotes the inventing engineer rather than the "do it first" engineer. A Commons Contribution fits into the sensible motive that forms the root of engineering ethics much more than misleading expectations of invention wealth. Help equate Open Ended Projects with "Open Source" engineering!

*The External Assessment Problem*

Engineers have supported educational projects through advice, materials, money and review. Project documentation review is all that is needed for external assessment. Once reviews have been accumulated, then some kind of metric can be created. Any other system is going to distort engineering. Engineers want to evaluate projects rather than evaluate individual student portfolios, teachers, courses, or engineering programs.

## Assess Uniqueness

Every project, every problem an engineer encounters is unique whether in school or on the job. Why are all star fonts are unique? Why does every McDonald's built require an engineer? Projects need to be unique. Let K-12 and tech programs argue the merits of hot dog launching projects.

## Assess Repeatability

The difference between Edison and Tesla was documentation. The audience of documentation is other engineers. Engineers read documentation with one question in mind, "Could I replicate this?"

## How

*Assess* by editing wikiversity pages. Change pages if the assessor can clarify, don't penalize. Celebrate communication that leads to clarity, not clarity itself. Don't try to create an assessment rubric. Read and react like an engineer. Read other assessments. Let them inform the current. Wikiversity's success is more about process than polishing stagnant information. Wikis has lowered the barrier to document storage, document archiving and document change control ... for the planet.

A critical mass of non-engineers is already in wikis. Freshman students named their team "MaxBearPig." A wikiversity editor deleted the team's project page within a day. Students solved the problem with a new name. No track record means no respect, no job. Wikis is already about respect.

## Gain Respect Yourself

Ultimately rewards of engineering projects will evolve an organization with a promotion process through levels of responsibility and influence. Start as a student. Continue as a working engineer. Help guide engineering assessment throughout the planet.

*What's Next?*

Help edit the book "General Engineering Introduction" in wikibooks. Assess general engineering projects in wikiversity.

## Fear Content

Content may no longer be the control point. Students can find answers before the question is finished. Calculator-only tests are cruel. When students can not find an answer to any question, a project is born. How does one prove there is no answer on the Internet? How does one prove the question is badly formed? These are the more important issues.

Respect now starts with creativity rather than content expertise. Projects are the starting point. Every engineering course should be evolving into an open ended project course. Imagine projects dictating content. Instructors advise. Students choose courses based upon the projects they tackle rather than arbitrary content. Projects can exist outside the educational institution boundaries of grades, semesters, and budgets. Instructors need to turn into invisible program managers. Projects need to dominate over content.

Pushing statics into high schools will not solve the content-volume problem. In summary, encyclopedia like courses can not compete with open ended projects.

## DIY, Hackerspaces

DIY and Hackerspaces are a modern church. Members pay dues and are always trying to dream up cool (open ended) projects that will attract new members. One hackerspace is hacking the Bible. They will replace traditional engineering practice if nothing changes.

## Feeding Birds

A hackerspace decided to celebrate the 60th anniversary of Richard Hamming's original paper. Six people sat facing each other over a narrow table with a power strip down the middle. Each had a netbook. They were reading the paper on line. They started off talking about how far they got in the paper, where they got stuck, what was confusing.

The older engineer tried to clear up some confusion. The hackerspace members glanced at him and then started chattering with each other across the table, typing madly at their computers. Then there was a collective silence and all of them looked at the older engineer. The older engineer talked again. The cycle repeated like if no one had anything to say.

The older engineer left feeling like he had just fed some birds. Some understood little, some a lot. They trusted the information they found on the net. They trusted that the collective mind in that room could figure anything out. They didn't feel obligated to understand everything. They had boundaries that the older engineer did not have. The older engineer felt obligated to absorb any information in the article or book. What set the boundary of the younger people?

The next generation's context is evolving on the web, not in textbooks or articles. What web sites are popular? What are their names? What has a wikipage? The boundaries of younger people seem to be set personally. Brains are being structured to hold searchable keywords (artifacts or archetypes like thingiverse or instructables), not content. Do we really want to answer the boundary question? ..yes it is "open ended projects".