New Product Development Methods:
a study of open design

by

Ariadne G. Smith

S.B. Mechanical Engineering
Massachusetts Institute of Technology, 2010

SUBMITTED TO THE DEPARTMENT OF ENGINEERING SYSTEMS DEVISION AND THE DEPARTMENT OF MECHANICAL ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREES OF

MASTER OF SCIENCE IN TECHNOLOGY AND POLICY AND
MASTER OF SCIENCE IN MECHANICAL ENGINEERING
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

SEPTEMBER 2012

Signature of Author: ______________________
Department of Engineering Systems Division
Department of Mechanical Engineering
August 10, 2012

Certified by: ______________________
David R. Wallace
Professor of Mechanical Engineering and Engineering Systems
Thesis Supervisor

Certified by: ______________________
Joel P. Clark
Professor of Materials Systems and Engineering Systems
Acting Director, Technology and Policy Program

Certified by: ______________________
David E. Hardt
Ralph E. and Eloise F. Cross Professor of Mechanical Engineering
Chairman, Committee on Graduate Students

New Product Development Methods:
a study of open design

by

Ariadne G. Smith

Submitted to the Departments of Engineering Systems Division and Mechanical Engineering on August 10, 2012 in Partial Fulfillment of the Requirements for the Degree of Master of Science in Technology and Policy and Master of Science in Mechanical Engineering

ABSTRACT

This thesis explores the application of open design to the process of developing physical products. Open design is a type of decentralized innovation that is derived from applying principles of open source software and crowdsourcing to product development. Crowdsourcing has gained popularity in the last decade, ranging from translation services, to marketing concepts, and new product funding. However, it is only in the past few years that open design has been considered as a method to create more innovative products in less time and for less money. While truly open design requires participants to collaborate and make contributions at each stage of the product development process, applications of open design to physical product development have been limited to accepting external contributions at only certain, less technical phases of design, such as planning, idea generation, or obtaining idea feedback. This thesis seeks to explore two questions related to creating a tool for open design in physical product development: what kind of tool can be developed to support crowdsourcing the full development of a physical product, and what types of design environments can benefit from this tool? Through a collaboration with GE Global Research for DARPA's Adaptive Vehicle Make (AVM) program, this thesis presents an early prototype of an online tool that allows for the open design of an entire product development process, in application to the development of a vehicle. Then, a framework is developed in order to identify the tool's applicability to other product development industries. Interviews with potential lead users in a number of different industrial sectors were conducted to better understand how this open design environment might be used and adapted for applications outside of a DARPA-driven vehicle design domain. Though the sensitive nature of projects in the defense and medical device industries prohibits this tool from used for its intended crowdsourcing purposes, there is promise for further development of the tool for uses in academic and education environments, and as an internal project management tool in other product development industries, such as aviation and consumer product design.

Thesis Supervisor: David R. Wallace
Title: Professor of Mechanical Engineering and Engineering Systems

Acknowledgements

It is hard to believe that six years at MIT have gone by so quickly. I will miss this place that is uniquely brimming with intellectual passion and curiosity. My experience would not have been what it was without the friendship and mentorship of those around me.

Thank you to my advisor, David Wallace. Your classes ignited my interest in product design and your tireless dedication to your students and selfless work ethic inspire me to be a better designer, engineer, and contributor to the world.

Qing and the rest of the team at GE; it was a pleasure to work with all of you on this project.

Thank you to my Cadlab family (Emily, Jeff, Lindy, Josh, My, Lauren, Taylor, and Ilan). I feel honored to have had the chance to work with such creative and brilliant people as you!

Thank you, Geoff, for wanting the best for me and helping me achieve it.

Thank you to my family, Mom, Dad, Noah, Ange, and Mario. You have given me the freedom and love to become my own person. From the first notes on my guitar and splashes in the pool to encouragement to study engineering, you have shaped my life for the best.

Table of Contents

List of Figures

List of Tables

Chapter 1

Motivation

The past ten years has seen immense changes in information, communications and prototyping technologies. We have smart phones that keep us connected to each other at all times. We have fast and light laptops that can store data and run heavy applications from the cloud. We have desktop 3D printers that can bring our prototypes and concepts to physical life. These technologies have revolutionized product development in the way that we communicate, design, and construct. Engineers no longer need to be in the same physical location to work as part of a team. We can easily share files and programs in the cloud. Rapid prototyping has facilitated low-cost exploration of new concepts. Product design and manufacturing have essentially been decoupled, as most engineering firms move operations oversees to lower development costs. In fact, in some cases, physical design activities have become so data-centric that physical aspects, such as prototyping and manufacturing, are merely execution steps at the end of a chain of digital manipulation (Shirky, 2007). Just as the way communication and construction has changed drastically over the past decade, the way we design and build products will, too.

In the 1980s, the integrated-circuit-manufacturing industry initiated a movement away from traditional manufacturing models and separated design from manufacturing as a stand-alone function, with designers outsourcing manufacturing to dedicated fabrication plants, or fabs (Belfiore 2012). Companies developing physical products in a variety of industries have decoupled the physical proximity of manufacturing and development, allowing remote work to become a standard component of their employment structure. As the US searches for new opportunities to maintain its competitive edge in manufacturing, product development, and innovation, alternative methods of product development should be considered as a potential solution. Decentralizing engineering product development outside of the traditional firm model is a new approach that has the potential to infuse new ideas and fresh perspectives into product development (Chesbrough 2007).

The most well-known cases of decentralized innovation come from open source design, as applied to software development (Raasch, Herstatt, Balka). The creation and maintenance of widely-used applications, such as Linux (Henkel 2006), Mozilla Firefox, and Apache HTTP Server, have proved that committed, yet otherwise completely unassociated, individuals have potential to create top-tier products. Decades after these initial demonstrations of open source software development, academic researchers are showing interest in investigating how to apply open source software principles to the development of physical products. Drawing from the successes of the open source software movement, open source design (also known as crowdsourcing or collective innovation) allows unaffiliated engineers and designers to be involved in the development of a company's product. Involving external users has been shown to yield more innovative ideas and expedite development time (Von Hippel 2005). Earlier research at MIT has focused on creating an open design space that fosters user-driven creation activities, allowing them to freely share contributions and build upon each others work (Sukkasi 2008). If implemented correctly, the open source design of physical products will come at a unique intersection of advances in digital fabrication techniques and engineering design and analysis software that will make decoupled, open source design possible.

The need to revitalize and rethink innovation is apparent now more than ever as the US continues to lose its edge in design and manufacturing. In early 2011, President Obama created the Advanced Manufacturing Partnership (AMP) in order to bring together government, industry, and universities to identify and invest in emerging technologies with potential to create "high-quality domestic manufacturing jobs and enhance the global competitiveness of the United States" (White House). More advanced manufacturing practices can not only help create more jobs domestically, but also reduce the current $500 billion trade deficit in manufacturing, *including* an $81 billion deficit in the manufacturing of advanced technology goods, such as consumer electronics, vehicles, and pharmaceuticals (Hockfield A23), which are all candidates for open design.

Aside from the creation of AMP, there have been several other government-led initiatives to rethink domestic manufacturing. DARPA, the Defense Advanced Research Projects Agency, tackled pharmaceutical manufacturing with the Accelerated Manufacturing of Pharmaceuticals Program by developing new way to product large amounts of high-quality vaccine-grade protein in less than 3 months in response to emerging and novel biological threats (Defense Sciences Office 2012). A new DARPA program, Adaptive Vehicle Make (AVM), aims to revolutionize automotive manufacturing by "compressing the development timelines for complex defense systems by at least five times" (Belfiore 2012). AVM aims to reinvent the way vehicles today are designed and built, and includes several tightly integrated projects to achieve this goal. META is a software package that aims to develop model-based design methods, specifically for the selection of components and testing of integration for vehicles. iFAB, also a software package, provides tools for choosing manufacturing methods for parts and also interfaces with factories to quickly adapt and scale to manufacture these parts. VehicleForge.mil is a collaboration space that hosts crowdsourced challenges to design large, complex, cyber-electro-mechanical systems, and utilizes interfaces with the software packages developed in META ("AVM Design Tools (META)") and iFAB ("AVM Manufacturing Foundary (iFAB)"). VehicleForge.mil is one of the first attempts at creating an online crowd sourcing platform for the entire physical product design process.

Thesis Overview

The concept of including external actors as contributors in the development of a product or service is not a novel idea. However, it is only in the past few years that companies have considered the merits of open design in physical product development and that researchers have thought of potential ways to provide environments to support this type of process. These efforts have focused on integrating the crowd in only *part* of the design process in physical products, such as voting on new concept ideas or submitting preliminary CAD (computer aided design) files. DARPA's effort to crowdsource the entire development of a vehicle through vehicleforge.mil is a novel and untested concept, but holds potential to change how firms approach physical product development. This thesis seeks to explore two questions related to creating a tool for open design in physical product development:

> *"What kind of tool can be developed to support crowdsourcing the full development of a physical product?"*
>
> *"What types of design environments can benefit from this tool?"*

Vehicleforge.mil provides a way to prototype a platform for open physical product design. Through work on the early-stage development of this tool, this thesis showcases how certain features would look and behave. The latter question, regarding design environments, is addressed through interviews with engineers in industries that are thought to potentially benefit from a crowdsourcing tool, such as that designed for vehicleforge.mil.

Chapter 2 provides background on the topic of crowdsourcing and open design, including a history of how crowdsourcing came to be developed and applied to various industries. The literature shows that crowdsourcing has been studied from many perspectives, from business innovation strategy to engineering, and draws primarily from case studies in the open source software movement.

Chapter 3 provides an argument for why crowdsourcing should be applied to physical product development, and what prior research has been conducted in this area.

Chapter 4 introduces the work done on vehicleforge.mil and reflects upon what additional features should be included to make the tool successful.

Chapter 5 presents a framework for evaluating the tool's application to industries outside of DARPA and the government. This chapter also includes the development of relevant interview questions, interviewee selection, and results from selected interviews.

Chapter 6 discusses what was learned through the work on vehicleforge.mil that can be applied to the next generation of an open physical product design tool.

Chapter 2

"Crowdsourcing is a distributed problem-solving and production process that involves outsourcing tasks to a network of people, also known as the crowd."

- Jeff Howe, "The Rise of Crowdsourcing"

The definition of crowdsourcing is simple - one *outsources* work (contracting it externally), where the external performer of the task is the crowd. This concept, though simple, has only recently become widespread. It has been implemented in a wide range of applications and industries, from gathering data to funding startups to designing logos. The research presented in this thesis is based upon open design, an idea that has evolved many the prior research concepts of collective invention, horizontal innovation, crowdsourcing, and open source development.

Background on decentralized innovation

Definition

The central concept of crowdsourcing is the idea that tasks can be decoupled from their owner and distributed across a wide array of "workers". This decoupling yields more favorable results for the owner in a variety of ways, whether that be in accelerated completion or more innovative and interesting results (Von Hippel 2005).

In the academic literature, papers writing on seemingly similar topics utilize a variety of terminology to describe this central concept of crowd-sourcing. Collective innovation, user innovation, commons-based peer production, open-source innovation all, like crowdsourcing, are theories of innovation that build upon the concept of decentralization and openness.

Comparison of models for decentralized innovation

It is helpful to first understand the source of open design and crowdsourcing, and how they fit into the the broader innovation landscape. The underlying principles behind these innovation strategies is that (a) product development is decentralized (spread among different participants and locations) and (b) information is freely revealed. What differs amongst them are their participants, stage of involvement in the product development process, and the task that they are accomplishing.

Collective Invention

One of the original explorations of decentralized innovation is by Robert Allen,— one of the first researchers to think about the benefits of intentionally revealing intellectual property (Allen 1983). At the time, Allen proposed that this method of free revealing could be most useful in corporations, and had the potential to "spread costs and risks of discovering a new superior design or technology on a number of competitors" (Schweisfurth 2011). Although he believed that collective invention would most manifest itself in corporate environments, such as pharmaceutical companies revealing a new biomarker, many modern examples of collective invention are in projects such as open software and university research (Meyer 2003). In the open source software movement (which is a recurring theme of inspiration in decentralized innovation), developers share code that contributes to a larger software project (Von Krogh et al 2006). A university-based example of collective invention is MIT's Biobricks project that provides a repository for organizations to contribute knowledge about reusable genetic and proteomic structures. The license on the site enables firms to pursue private commercial interests using knowledge they obtained from the repository (Powell 2009).

User Innovation Networks

Open source software has played a profound, if not founding, role in decentralized innovation research. New thought on collective invention, coupled with the open source software movement, inspired researchers, such as Eric Von Hippel, to examine the benefits of users freely revealing their inventions. Von Hippel is known for developing the concept of user innovation, where end-users, not manufacturers, are responsible for a significant amount of new innovations (Von Hippel 2003), which developed into the strategy of a user (or horizontal) innovation network. Inserting users into the product development process in roles of inventors and creators has advantages over manufacturer-centric innovation systems for a number of reasons. Participants in user innovation networks:

- create exactly what they want, thus yielding higher user satisfaction with products
- build on each other's new concepts
- feel more loyalty to the product because of their contributions to it
- lower product development costs for the firm (Von Hippel 2007)
    - less time spent on user testing
    - labor distributed across many different users

The sports industry is a rich source for examples of user innovation. Early participants in "rodeo kayaking" made their own customized kayaks for the new sport's requirements. These early "user-innovators" began building custom kayaks for others, which led manufacturers to begin to fabricate their own. In this case, initial users were the first designers and manufacturers of these new products, putting them at the same level as traditional kayak manufacturers. This type of development makes sense for a few reasons. First, this type of user-initiated product development allows for the transfer of "sticky" information, such as what is fun about the product, from users to manufacturers that wouldn't otherwise take place. In addition, users take on the role of manufacturers as they modify existing products themselves, lowering the cost of production (Shah 2000). In the case of extreme sport technology, user-initiated inventions have proved to be very useful: 40% of the innovations by users were solving an "urgent problem" for other users, almost 15% of these innovations were completely new products, and quarter of all user-innovations were produced for sale by manufacturers (Franke 2003).

Commons-based peer production

Commons-based peer production focuses on the "creative production of innovative digital content and cultural goods by a multitude of authors", which translates to accomplishing a large problem with the help of many decentralized participants who freely reveal their findings (Benkler 2002). This model is restricted to non-corporate individuals and capitalizes on the talent of people instead of an automated system. Many tasks that we assume are "crowdsourced", such as using Amazon's mechanical turk to proofread a book or developing free software, are arguably actually commons-based peer production.

Crowdsourcing

The concept of crowdsourcing has been around long before Von Hippel or the Internet; the Oxford English Dictionary, initiated in 1857, called for the community's help to index all the words in the English language and produce quotations for each word's usage (Winchester 1999). The actual term "crowdsourcing" was coined in 2008 (Howe 2006) in an article that promoted the idea that everyday people could collectively create content, solve problems, and even conduct research and development. Crowdsourcing has many advantages over traditional internal delegation of tasks. The crowd works faster and more cheaply than in-house development (Brabham, 2008), increases the quality of solutions, and delivers answers to question that are deemed otherwise "unanswerable" by company developers (Lakhani 2007). Crowdsourcing work in firms has even been shown to reduce the risk of product development and increase market success (Ogawa and Piller 2006).

Crowdsourcing is different from other decentralized innovation models in that it is usually used by a company and delegates very specific tasks or steps in the product development process to actors external to the firm.

Open Source Innovation

The idea behind open source innovation is applying principles from the open source software movement to the design of other, non-software goods. Open source innovation is:

> *"...characterized by free revealing of information on a new design with the intention of collaborative development of a single design or a limited number of related designs for market/non-market exploitation."*
> - Raasch, Herstatt, and Balka 2009

The contributions can be generated by commercial or private contributors, and are intended to be part of a larger design. Though crowdsourcing allows actors to participate in one stage of product design, such as information gathering or editing, open source design is intended to integrate users into the entire product development process. Crowdsourcing can be applied to physical products, services, content, as well as software. The terms "open source design" or "open design" refer specifically to applying open innovation to physical products (Vallance et al. 2001). The output of an open source design project should be a product that is ready for production, or at least in a refined prototype stage.

Comparison

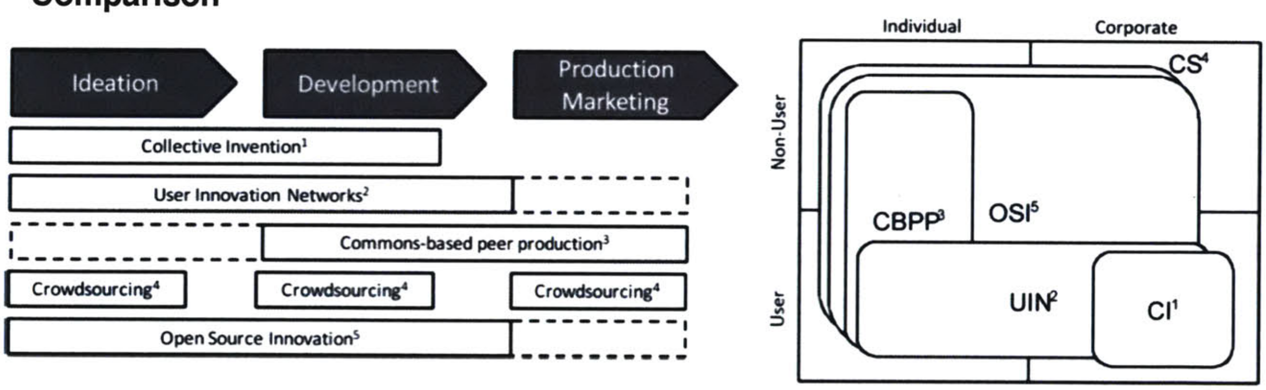

Fig. 1: Comparison of decentralized innovation models with respect to phases of the product development process and participants involved (Schweisfurth et al, 2011)

These models of decentralized innovation differ based on who is involved and what tasks are completed by participants. In collective invention, the participants are companies who share knowledge, not actual designs, with the goal of advancing progress in their industry. User innovation networks, on the other hand, involve end users changing a product for their own benefit, with manufacturers then integrating these changes back into the product. Commons-based peer production is non-corporate individuals and users coming together to complete tasks and designs. Crowdsourcing involves interaction between a company with a task and external individuals completing the task. The outcome is not a fully designed product, but a pool of data, an idea, or solution concept that a company later integrates into a larger design. Open innovation is fairly generic with respect to actors, but aims to create entire designs by using participant contributions at every stage of the product development process.

Summary

The various models of decentralized innovation have been applied in many different industries for diverse applications. Though the core principles behind decentralized innovation, free revealing and development by external users, have been applied for many years, it is only recently that the general public has become aware of these strategies. At the moment, the most well-known and discussed type of decentralized innovation is crowdsourcing. Though this thesis focuses on applying *open design* to physical product development, crowdsourcing has been by far the most widespread application of decentralized innovation. It is thus useful to follow the recent examples of crowdsourcing in order to better understand how physical product development fits into current decentralized innovation landscape.

Examples: non-physical products

Since Jeff Howe first coined the term in 2006, crowdsourcing has been used in thousands of projects ranging from translation apps to product funding to logo design. Since crowdsourcing revolves around outsourcing a task to the crowd, different examples can be categorized by the task they aim to accomplish.

Crowd labor

One of the most common applications of crowd sourcing is accomplishing a task, normally insurmountable by a small group, by leveraging the collective power of the crowd. Crowd labor is greatly facilitated by the mass connectivity that the Internet provides. Amazon's Mechanical Turk is an example of how to accomplish small tasks on a large scale through community involvement. Tasks that lend themselves well to crowd labor are those that do not require a high level of skill and can be accomplished in a relatively short amount of time. Publishing a paragraph and asking for translation in exchange for a small monetary reward (a few cents) Mechanical Turk has made it possible to translate entire novels and iPhone applications. An even more efficient use of crowdsourcing is when the crowd labor task accomplishes two relevant tasks simultaneously. The Captcha and Re-captcha programs provide website security by administrating tests that only humans can pass. These tests include typing words from historical newspapers and texts. Over 40 million words are transcribed each day through these programs (Von Ahn 2011). Other uses of crowd labor include providing information on traffic conditions and navigation, Waze (Waze Mobile 2012), curating quiz questions and answers, Smarterer (Smarterer, Inc., 2012), and providing peer reviews for scientific papers, IMPROVER systems biology verification (PMI and IBM Research, 2012).

Crowd funding

A popular application of crowd sourcing is funding projects or startups through small investments spread out over the community, instead of raising large sums from investors. Most crowdfunding platforms, like Kickstarter (Kickstarter, Inc., 2012), collect money through "pre-orders" of products that are promised to be manufactured and delivered if the funding goal is reached. Similar sites such as Indiegogo (Indiegogo, Inc., 2012) and Rockethub (Rockethub, Inc., 2012) also exist, but Kickstarter has been, by far, the most successful. In its third year, the company raised $119 million for its participating projects, and took home $6 million in commission (Jackson 2012). Projects raising the most money on crowdfunding sites were film & video and music. Design projects, such as the TikTok iPod Nano watch (raising over $1M on Kickstarter) have also yielded successful results (Dillow 2011). The Pebble, a watch that displays information from an iPhone, raised over $10 million as of May 18th, 2012 ("Pebble: E-Paper Watch for iPhone and Android"). This type of funding is scarce for a series A for a promising consumer web startup, let alone a product design project funded by the crowd. Most projects on crowd funding sites have found that traditional methods of financing do not apply to their product, either because the market is too small or too risky. Crowd funding allows products to both exemplify market traction and try a first run of manufacturing with early adopters before opening sales to the greater public (Fabricant 2012). Startups raising funds through sites like Kickstarter also retain all ownership of their company instead of trading it for money from venture capitalists (Wortham 2012). Besides launching consumer products, crowd funding has been tried in a venture capital firm that invests small sums in startups based on the crowd's input, (VenCorps) (CNET 2008), and has been applied to funding undergraduate scientific research programs (the Open Source Science Project) (Linton 2011) .

Crowd ideating

Crowd ideating leverages the crowd to come up with creative ideas. Crowdsourcing in this application has the potential to increase the quality of solutions and deliver answers to questions that a smaller group would not be able to figure out. The creativity and problem-solving skills of the "crowd" have been elicited for coming up with creative ideas in marketing and branding, international development, and consumer products. Examples of crowdsourced early stage idea generation include IBM's innovation jam (IBM InnovationJam 2008), in which thousands of employees from external companies provided ideas on how to make IBM the "enterprise of the future" and OpenIDEO (IDEO, Inc., 2012), where users submit solutions to social development challenges. In marketing and branding, sites such as 99 designs (99Designs, Inc., 2012), Idea Bounty (IdeaBounty, Inc., 2012), and GeniusRocket (GeniusRocket, Inc., 2012) seek crowdsourced ideas for logo, web, or marketing campaign design. There have even been a few cases of crowdsourced consumer graphic design, such as Pepsi's "design our pepsi can" (Pepsi Design Our Can 2008) where consumers submitted new can designs, Threadless' (SkinnyCorp, LLC, 2012) crowdsourced design of t-shirts, and Quirky (Quirky, Inc., 2012), where users suggest ideas for new, every day consumer products with the community curating these ideas, and Quirky product designers and engineers finalizing the design and manufacturing final products.

DIY/Digital Fabrication

The "do-it-yourself" digital fabrication movement embodies many of same qualities as the open source software movement. Digital fabrication is a process that creates physical prototypes through 3D modeling software and CNC (computer numerically controlled) machining. Examples of digital fabrication include laser cutters, CNC mills, and 3D printers. Though there are digital fabrication and rapid prototyping machines that are produced by established manufacturers, a significant amount of development has been accomplished through open source contributions. The motivation that everyone should own a digital fabrication/rapid prototyping machine on their desktop that is easy and cost-effective to make themselves has kept most of the machine development in an open design environment. RepRap (RepRap 2012) (short for "replicating rapid prototyper") is an open design project with the goal of making a 3D printer that can print its own components (Hoeken 2008). All information on how to build a RepRap machine is freely available online, and the community is encouraged to contribute to the movement by revealing new designs and improvements to the machines. Makerbot (Makerbot Industries, LLC, 2012) has several generations of a low cost, desktop 3D printer, and though it profits on the sales of its machines, the company encourages its users to share designs made on MakerBot printers on a site called Thingiverse (Makerbot Industries, LLC, 2012). On this site, users can upload STL files that other users can freely download and make on their machines.

Along with the development of rapid prototyping machines comes other tools for DIY digital fabrication. Ponoko (Ponoko, Inc., 2012) is a site where users can buy and sell their plans for fabrication, and upload their own files for the company to manufacture. Alibre (Alibre, Inc., 2012) and Autodesk (Autodesk, Inc., 2012) have both developed either free or low-cost 3D modeling software for users to design parts for digital fabrication. Freely available design software, such as these, makes the open design of physical products more feasible, as users gain greater access to the tools necessary for design.

Fig. 2: Companies or programs engaging in crowdsourcing

Why is open design useful?

Industry Success in Open Design

Despite the hesitation of most firms to adopt open design, there have been successful instances of companies using decentralized innovation. InnoCentive, a platform that allows organizations to solve important problems through posted challenges, showcases some of these successes (InnoCentive, Inc., 2012). Large corporations and government organizations, such as pharmaceutical giant Eli Lilly, participate by hosting challenges to solve their organization's problems . Participating organizations pay $10,000 to $100,000 per successful solution. Colgate-Palmolive's challenge of finding a way to inject fluoride powder into a toothpaste tube without dispersing it into the air awarded a prize of $25,000. This particular challenge was solved by a "tinkerer" from Ontario and was discovered much faster than an internal R&D solution, and for a fraction of the cost (Howe 2006). These impressive results are due to the fact that crowdsourcing platforms, like Innocentive, attract a great diversity of intellectual backgrounds (Lakhani 2007).

In network theory, the most efficient networks are those that link to the broadest range of information, knowledge, and experience, and the same is true of tools like Innocentive (Howe 2006). Proctor and Gamble, a multinational consumer goods corporation, has also found great results from decentralized innovation. As R&D costs skyrocketed, the company decided to change the way they approached research from an organizational perspective. A push to increase the percentage of external innovation to 35% yielded an increase of 60% in R&D productivity.

Risks of Open Design

The merit of crowdsourcing tasks has been proven through a multitude of projects utilizing unaffiliated individuals to collectively accomplish great things. It is a relatively easy for a company to try crowdsourcing a single task or section of their product development process. Risks are fairly low, as a company still has control over its final product, and potential benefits are high as tasks can be completed faster and more cheaply with crowdsourcing. Convincing firms to utilize open design, however, is more complicated, especially when it involves opening the design of the entire physical product to external actors. Though open design is participant-agnostic, it has most often been used for non-physical product development by non-corporate users, as is the case in open source software development. There are a number of reasons why companies might be hesitant to use open design in their physical product development. First, there is uncertainty as to whether or not principles of open innovation (which has been most used in software development) can be applied to physical products. The nature of software development with its low capital costs makes it much more economically feasible to share and develop. For example, contributors wishing to add code to an open Github (GitHub, Inc., 2012) repository need no manufacturing or prototyping equipment—other than their computer and a wifi connection. The files they exchange are easy to work with; changes are noticeable and code can be reflected to document its design. On the other hand, a hardware product development process typically involves prototypes, manufacturing, and physical stores to distribute the products. While much of the development work in open design can be accomplished virtually, the ultimate purpose is the design and production of a physical artifact (Raasch and Herstatt 2011). In the world of physical products, in contrast to the digital realm, development, production, and distribution do not coincide and can moreover be very costly (Von Hippel 2007); designs can be technologically complex, necessitating a diverse set of highly specialized skills and possibly costly equipment; and institutions allocating intellectual property rights differ in large measure, relying on patenting rather than copyrights and lacking a pool of open design licenses to choose from (Hope, 2003).

Second, though much has been said regarding the benefits of integrating users into product development, as is the case with open design, companies can still be reluctant to do so. As the idea of user-innovators is still fairly new in practice, there is a lack of understanding of how to best leverage user activity and reap its benefits (Raasch 2011). There are real risks associated with companies using open design for their product development.

When the user-innovators are lead users of a product and alter it because of their own end needs. However, when less closely affiliated users are brought in for creative input (such as is the case in open design projects), their needs and skills might not be as aligned with those of the organization or end users. This can lead to ideas that are not useful, or even detrimental to the organization, such as creating unsafe or unpopular products. There is also the risk that engaged users will not feel comfortable fully sharing their contributions and yield will be low. Furthermore, it is difficult for companies to create lasting ways of engaging with users beyond the user contribution, making continuity and integration difficult. From a purely organizational standpoint, it is time and labor-intensive for a company to organize and execute an open design program. The benefits of external user input might not outweigh these costs of designing and implementing such a tool. Any tool designed for companies using open design should seek to minimize these risks of organizational challenges and user engagement.

Chapter 3

Why open design?

Crowdsourcing and open source development have proven to be successful methods of accomplishing large tasks and receiving creative inputs on problems. Physical product development, however, is rarely the subject of crowdsourcing or open source development. A variety of translation, navigation, and idea generation tasks have been addressed with crowdsourcing, and many substantial software projects have been made possible through open source development. But it is only within the last five years that researchers have turned their attention to the open design of physical products. There is research exploring the application of open source principles to the realm of physical product design, but few do so based on empirical evidence, as there are still limited examples to draw from. Despite this lack of evidence, there is still encouragement for firms to move to embrace the consumer-innovator. Research by Von Hippel, Ogawa, and de Jong has shown that in the US, UK, and Japan, millions of citizens innovate to create and modify consumer products to better fir their needs. They encourage businesses to think about how to reorganize product development systems to efficiently accept and build upon prototypes developed *by* users (Von Hippel et al 2011).

There have been a number of key technological developments over the past few years that now makes the development of a tool for open physical product development a possibility.

The physical world is becoming increasingly digitized...

Hardware is becoming more like software, thus priming it to take on open source design challenges previously only possible in software projects (Von Hippel 2005). Physical activities are becoming data-centric to the point that physical aspects are merely executional steps at the end of a chain of digital events (Shirky 2007). Companies designing physical products in the United States have virtually no in-house manufacturing or prototyping, and complete all "physical execution" at factories over seas. A majority of design in larger product development companies (CAD, multiphysics simulations, plans for manufacturing) is done on front of a computer and, besides in-person team meetings, can be accomplished remotely, so long as users have access to the software required.

Companies are making their hardware increasingly open to users...

In 2007, Nokia began to ship phones with user-accessible APIs. Arduino (Arduino 2012), the circuit board company, has built a following amongst DIY gadget-builders by putting everything out in the open (schematics, design files, software) and encouraging users to modify and improve the product. As more companies follow this trend of opening up their products for tweaking, a larger population of users will be inspired to innovate for themselves (Rowetel 2008). Free-revealing in hardware does carry some risks. If every design artifact public knowledge, there is very little preventing competitors from making a similar product at a lower cost. As hardware becomes more open source, manufacturers will also have to switch their perception of users in a traditional role of a consumer, to that of an innovator.

Design and prototyping is becoming more tangible for users...

Not only has outsourcing the manufacturing of products oversees made production costs plummet, but the increasing availability of open source modeling software and collaboration tools has greatly decreased the cost of designing products as well. Freely available CAD (computer-aided design) programs, such as Google Sketch-Up (Trimble, Inc., 2012), make it possible for anyone to design products for manufacturing. Personal digital fabrication machines, such as the aforementioned 3D printers and laser cutters, then let users actually fabricate their products. Desktop manufacturing like this has spawned a new community of amateur makers who are interested in building upon each other's ideas (Anderson 2012).

There is also greater access to more advanced prototyping and manufacturing tools than Makerbots and desktop laser cutters, which are intended more for small, hobby-scale projects. Proto Labs (Proto Labs, Inc., 2012) offers small runs (less than 10 parts) of CNC machined and injection molded parts in only 1-3 days. Open hacker spaces, where members pay a monthly fee to use a fully stocked machine shop and fabrication facility, are popping up across the country. TechShop (TechShop, Inc., 2012), for example, offers use of laser cutters, 3D printers, vinyl cutters, CNC lathes, and many more tools, for around $100 per month. Places like this make design and fabrication much more accessible for the every day user-innovator especially when compared to the cost of purchasing a machine for hundreds of thousands of dollars.

It is because of these technologies that users and companies are more apt to adopt open design into their product development. The feature that separates open design from the other models of decentralized innovation is that its output is a fully designed product, instead of just part of product or service. This requires the open design tool to collect and consolidate contributions from users at each stage in the product development process.

Product Development Process

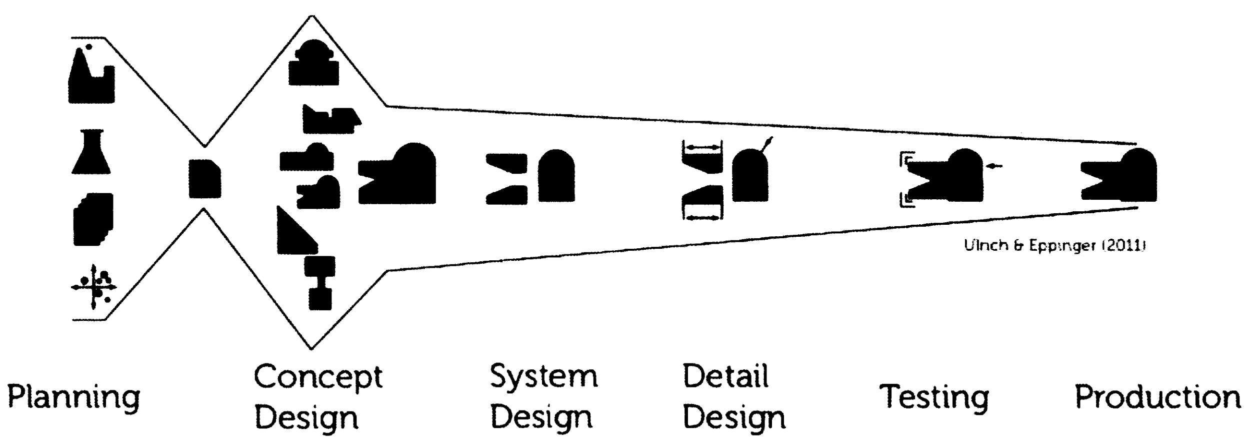

Fig. 3: A typical product development process (Ulrich and Eppinger, 2011)

What does it actually mean for an open design project to output a completed product? The participants should make contributions that follow the product development process, which encompasses the steps of designing a physical product from initial idea generation to final detailed concept design. There are six general phases of development, as shown in Figure 3. The entire process can be thought of as a funnel that starts with many concepts and is subsequently narrowed and refined to reach the final prototype (Ulrich and Eppinger 2011).

Planning

During the planning phase, the team discusses the mission of the project, identifies a target market, business goals, key assumptions, and constraints.

Concept development

In this phase, more detailed user needs are developed and documented. The team generates as many ideas as possible for concepts that fits these user needs. One or more of these concepts are selected for future development. This selection can be accomplished through sketch models (rough physical models out of low-cost, easy to fabricate materials) to answer broad questions on feasibility and market traction.

System-level design

The selected concepts are divided into sub-systems and components that can more easily be developed during the detail design phase. Breaking products into components helps create division of labor. Initial plans for manufacturing, such as the general materials and processes used, are also defined in this stage.

Detail Design

The team works on the final geometry, materials, and tolerances of the final components. This is when teams finalize 3D models representing these design decisions. Software simulations may be used during this phase to ensure that selected materials and geometries meet the desired engineering specifications (such as force or thermal requirements). Manufacturing plans are developed further developed through CAM (computer-aided manufacturing) software.

Testing and refinement

In this phase, the team builds more complex detailed prototypes that seek to answer questions regarding the product's technical feasibility and user needs. Earlier stage (alpha) prototypes test whether the product will work as designed, and how it meets user needs and engineering specifications. Later stage (beta) prototypes use production-grade parts and test performance and reliability.

Production ramp-up

In this phase, products are manufactured using the machines and tooling paths specified earlier in the design process.

Open design of physical products

One of the goals of this thesis is to begin the development of an open design tool for physical product development. In this case, the tool is not so much a crowdsourcing mechanism in the sense of outsourcing tasks as it is an open environment for collaboration and innovation. Kristen Balka's concept of "open source innovation" is a more appropriate label for this type of tool:

> *"...characterized by free revealing of information on a new design with the intention of collaborative development of a single design or a limited number of related designs for market/non-market exploitation."*
> - Raasch, Herstatt, and Balka 2009

This definition highlights a number of important qualities of an open source design project:

1. all innovations and ideas should be freely revealed
2. contributions should be shared for the purpose of further development of the product
3. collaboration between actors should be encouraged and expected

In addition, the tool developed in this thesis is intended for the open design of an entire physical product, and has additional requirements:

4. the majority of product development should be accomplished by actors
5. contributions should span all phases of the product development process
6. output should be a physical product ready for manufactured

Examples of crowdsourcing and open design

Though there are many examples of physical products that have been created in open design environments, according to Balka's definition. The additional stipulations 4-6 for open design in physical product development disqualifies many many of these current open design projects from being considered such by this definition. These projects usually include the free revealing of information, shared contributions, and collaboration, but require contributions in only certain phases of the design process, or do not output products ready to be manufactured.

Table 1 shows a number of recent open design projects that focus on physical products. What makes them different from the tool in this thesis is that users do not necessarily participate in all stage of the design process and that the artifact of the open design project (output) is not necessarily a finished product.

Table 1: open design projects focusing on physical products

organization	project	actions by users	artifacts
Google Lunar X Prize (X Prize Foundation 2012)	$30 million prize available to 1st privately funded team to safely land a robot on the surface of the moon, have that robot travel 500m over the lunar surface, and send video, images, and data back to Earth.	Privately funded companies develop technology in parallel to meet challenge specifications.	lunar robot
Local Motors (Local Motors, Inc. 2012)	open-source community for car designers and fabricators	Users create projects and upload photos, sketches, renderings, and CAD models of vehicle concepts	CAD models, sketches, renderings
Ecomagination (General Electric Corp., 2012)	$200 million in prizes for ideas on how to build the next-generation power grid. Challenges include solutions for renewable energy, grid efficiency, and eco homes and buildings.	Teams post ideas on potential solutions to challenges. Selection is made with general public voting and GE judging. Winning teams are awarded with the chance to pursue a commercial relationship with GE.	Description of idea
Grabcad (GrabCAD, Inc. 2012)	Forum for the exchange of CAD models and labor. Users can upload files to share and solicit their skills for hire.	Users upload CAD models to share with the community and demonstrate their skills.	CAD models
Quirky	users create innovative consumer products	users suggest ideas for products and the community votes on which ideas should be pursued. Quirky designers and engineers refine successful ideas and prepare them for sale.	consumer product sold through Quirky & distributors.

Most of these projects involve freely revealed contributions by actors, shared contributions, and majority of work completion by actors. Having collaboration between actors that spans all phases of the product development process and yields a finished product is scarce. Though this sample is small, it is representative of the types of open physical product design projects that exist today.

organization	Google X Prize	Local Motors	Ecomagination	Grabcad	Quirky
innovations freely revealed	✓	✓	✓	✓	✓
contributions shared for further development		✓	✓	✓	✓
collaboration between actors			✓		
majority of work done by actors	✓	✓	✓	✓	
contributions span all phases of product development process	✓				
output is finished product	✓				✓

Table 2: compatibility of selected open design projects to this thesis' definition of open design

There are a number of recent papers exploring the realm of open design in product development. Kristen Balka, in her work Open Source Product Development, presents a multitude examples of what she defines as open source design (Balka 2011). Over the course of two years, she solicited entries for open source projects through her website, open-innovation-projects.org, and identified 104 of these as meeting the criteria of an open source design project. In her survey of these projects, Balka focuses on examining the project characteristics, structures, and successes in order to map a landscape of open design projects and to compare these projects to those in open source software development. These projects all include some element of free revealing towards the development of a public good, however Balka's focus on the development process, such as what artifacts the actors contribute and how this corresponds to the different stages of the product development process, is narrow. She classifies each project in a stage of development, from 1-5, with 1 being planning and virtual development, and step 5 having completed development. What is missing is how each team arrived at each stage of development. A crowdsourced product could be thought of by random contributors, but modeled, built, and tested by a sponsoring organization, as is the case with Quirky, but would not fit the definition in this thesis of open physical product development.

Of the six hardware open design projects that Balka profiles in-depth (Oscar, RepRap, Free Beer, OSGV, and Neuros OSD), none fit the requirement of having actors participate in every stage of the design process. Oscar solicited ideas for a cheap and simple car. RepRap provides the design of its 3D printer in the hopes of volunteers furthering its development. Free Beer revised open recipes based on community feedback. OSGV (open source green vehicle) had the community complete a preliminary design for a fuel efficient SUV, licensing the development to an external startup company. Neuros OSD is an open-source, Linux-based media center that integrated user feedback into new design iterations. Each of these examples includes a part of product development that is crowdsourced and made open to the public. However, a governing organization is usually responsible for integrating feedback, testing feasibility of concepts, and actually producing the product.

Chapter 4

There have been several examples, as shown in chapter 3, of product development that has been aided by opening at least one phase of development. One of the central questions in this thesis is investigating what kind of tool can be developed to support open design of the entire product development process.

This chapter shows the early stage design of an online tool for the open design of physical product development, in which users participate and collaborate in each stage of the product development process.

Case study: DARPA AVM

The early stage design of an open design tool was developed through a joint research effort with GE Global Research and MIT for a DARPA-initiated program, called Advanced Vehicle Make (AVM). AVM seeks to revolutionize automotive manufacturing and by "compressing the development timelines for complex defense systems by at least five times" (Paul Eremko, program manager) and democratizing the design process (Lohr 2012).

DARPA's background with open design

The AVM project is not DARPA's first foray into crowdsourcing and open design for larger physical products. LocalMotors, an open-source community of car designers and fabricators, hosted the XC2V (Experimental Crowd-derived Combat-support Vehicle) Design Challenge from February to March of 2010. The design brief was to create a vehicle that could transport people and/or cargo quickly and efficiently in a potentially hostile environment. The base chassis (shown in Figure 4) for the vehicle was provided by LocalMotors, as well as a few broad requirements for the vehicle's performance (e.g. front and rear suspension travel and rear wheel drive).

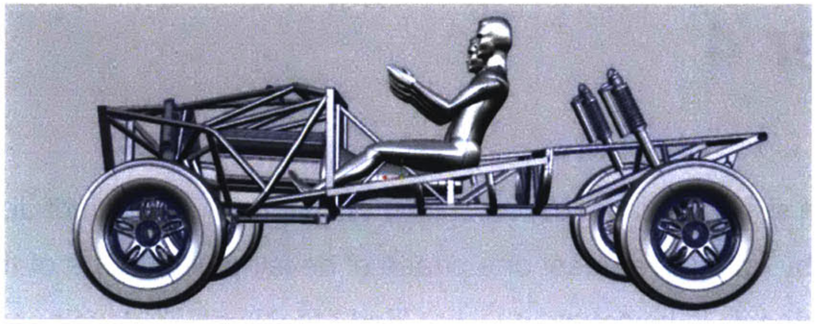

Fig. 4: chassis provided for DARPA's XC2V challenge (LocalMotors.com)

The final deliverables requested are images of what the vehicle might look like. The winning submission is shown in Figure 5. There is no engineering analysis or development provided for this design. The concept is well thought out and might provide good inspiration for future development, but only encompasses the planning and concept design stages of product development.

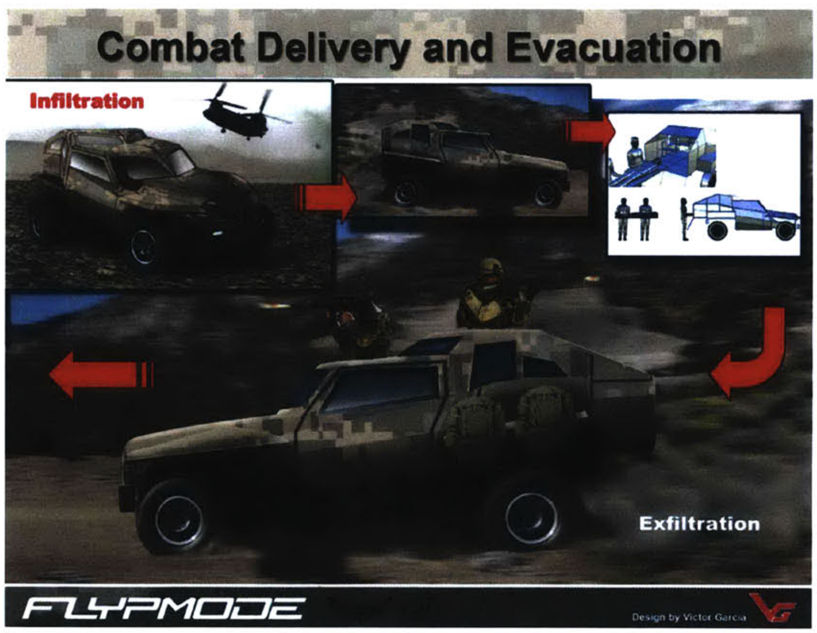

Fig. 5: winning submission of DARPA's XC2V challenge (LocalMotors.com)

Another example of a DARPA-led open design environment is the UAV forge challenge. The site, uavforge.net, provides a virtual environment for teams to form and collaborate with the goal of developing small unmanned air vehicles (UAV forge 2012). Teams submit videos in order to demonstrate completion of milestones, with down selection of concepts accomplished by crowd voting. The winning team will work with a selected external manufacturer to fabricate a first run of 15 UAVs and jointly iterate on the design.

Unlike the XC2V challenge, UAV forge asks for demonstrations of working prototypes via video submission. DARPA also defines much more specific product specifications than in XC2V, such as a requirement to identify persons or activities of interest up to 100 feet away. Through these milestone videos, UAVforge forces teams to go through planning, concept design, and a bit of system design in the product development process.

Based on interviews with an MIT team participating in the final stages of the competition, it is clear that the workflow is lacking in a few areas. The idea of demonstrating engineering accomplishments through videos is a good idea in theory, but in practice, teams can easily fake successes and make it difficult to determine what parts actually comprise their design. In addition, the collaborative features of the site do not work as intended. Forums on UAV forge are meant to act as a central space for teams to exchange ideas. The competitive nature of the project, however, caused many teams to not to reveal ideas for risk of other teams taking them. This destroys one of the core tenets of open design: collaborative development of a design. While it is true that collaboration does occur *within* participating teams, the goal of such sites like UAVforge is to induce sharing between large groups of participants.

Adaptive Vehicle Make Program

Most projects within AVM are software tools that are intended to digitize the design process, from concept generation, to running engineering simulations, to prototyping and coordinating manufacturing. There are three primary elements of the program: META, iFAB, and FANG.

META

The goal of META is to improve systems engineering, integration, and testing processes for defense systems. The tools currently being developed help users with part integration in complex systems, such as vehicles.

Additionally, a program called C2M2L (component, context, and manufacturing model library) seeks to develop domain-specific software models that are aid in design, feasibility verification, and manufacturing of parts.

iFAB

iFAB is where the crowdsourced vehicles are actually built. These factories are intended to have modular machining equipment that can adapt quickly based on new vehicle designs and help compress fabrication times. In addition to making factories programmable and flexible, there is also interest in distributing the manufacturing between different locations to accommodate a wider range of systems. The software components of iFAB that are currently being developed are intended to transfer knowledge traditionally held at a factory floor by someone with extensive manufacturing experience to the vehicle designers. For example, one of the current programs takes an input of a CAD file and outputs estimates of cost and time to manufacture, broken down by each feature of the part. The program also provides recommendations on how to optimize time and cost by modifying the part's geometry or materials.

vehicleforge.mil

Vehicleforge.mil (AVM Collaboration Capability 2012) is the open source design tool used to host the design challenges in FANG. The products developed will be complex, cyber-electro-mechanical systems created by unaffiliated designers and engineers. The program aims to draw parallels from the open source software movement to the development of defense systems, largely by democratizing design tools that are currently prohibitively expensive to be used in crowdsourced applications with META and iFAB.

FANG

FANG (FANG 2012), which stands for Fast Adaptable Next-Generation, is a series of design challenges that will crowdsource a prototype military ground vehicle from initial concept development to final manufacturing. FANG utilizes META, iFAB, and vehicleforge.mil in these design challenges presented over the course of one year. The first two contests will include collaborating on a design for an amphibious vehicle for the Marines, with a $1 million prize initially given for mobility and drive-train subsystem design, and another $1 million prize given six months later for the design of the chassis and other subsystems. A $2 million prize in 2014 will be awarded for the best design of an entire vehicle. Participants ("unaffiliated designers and engineers") in these challenges may be individual, small teams, businesses, and major defense contractors.

Overview of design tool

The goal of the vehicleforge.mil project is to design a tool where users can collaboratively develop highly technical, physical products from planning to production. As many tools attempting to curate open source design communities for physical products have fallen short in the past, there are a few crucial elements that must exist in order for vehicleforge to function as intended.

Open design projects discussed in chapter 3 mostly encompass activities in the planning stage and beginning of the concept development stages of product development process.

Concept development, system-level design, detail design, and testing/refinement involve rough back-of-the-envelope calculations, physical prototypes, and software modeling in order to select concepts based on feasibility and further their development. In order to meet the requirements of allowing users to participate in each stage of the product development process, vehicleforge.mil should have the following features:

seamless sharing of artifacts

Most open design tools fall short due to their lack of coherent file sharing. UAV forge teams, for example, post no design files, making it difficult to collaborate seamlessly and build upon others designs. GrabCAD users post CAD files, but are not organized into product components or projects.

collaboration/communication between participants

It is necessary to have not only good communication between teams, but also communication across the site between different projects and participants.

tools for verifying concept feasibility

Since vehicleforge.mil users are expected to work on products from planning to production, there needs to be a way to credibly verify product feasibility. This involves making engineering tools that are currently reserved for companies who can afford their costly licenses available to the every day user so that he/she can produce more complex and credible designs.

The last feature on verifying concept feasibility echoes several papers in open innovation that stress the importance of companies including a "toolkit" in open design projects so that user-innovators are better able to provide credible products (Baldwin 2006, Von Hippel 2005, Hope 2005, Benkler 2002)

In addition to the features above, DARPA has defined a set of requirements that the vehicleforge.mil site must fulfill (see Appendix A). These include requirements on:

- using version control to manage file exchange on the site
- hosting some form of system-level representation of complex cyber-electro-mechanical systems

- hosting public and private projects through which users organize and publish their contributions
- collaboration tools (e.g. wiki, message board, mailing lists) that facilitate collaboration

Encompassing the entire product development process

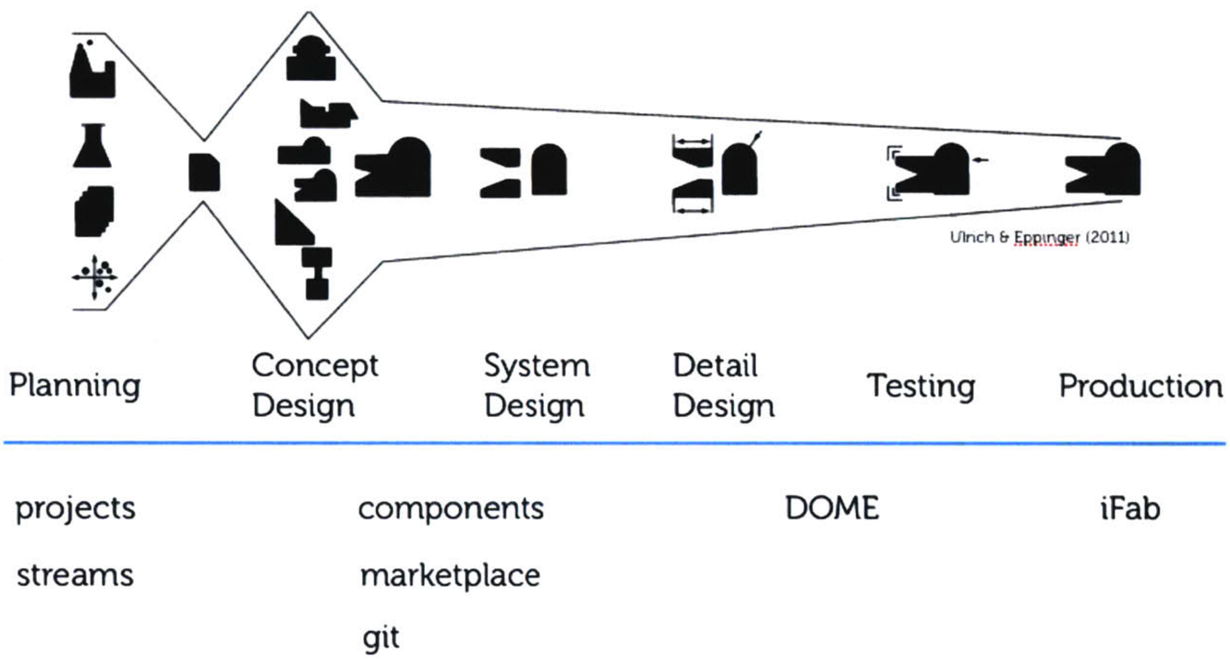

Fig. 6: features of vehicle forge and the phases of the product development process

Vehicleforge needs to provide for the seamless sharing of artifacts, collaboration/ communication between participants, and tools for verifying concept feasibility. The core features of the site (see Figure 6) address these requirements at each stage of the design process.

<u>Planning and concept design</u>

Projects on the site are analogous to those in a traditional engineering setting. Team communication and project management takes place here.

Streams are a way of communicating across the entire vehicleforge site. Users can subscribe to streams to receive updates on users, projects, or specific parts.

Concept/System/Detail Design

Version Control

A version control system, git, is used to manage all file activity on vehicleforge. Git, like most other version control systems, maintains versions of files at progressive stages of development. Every change to a file is stored and the file can be reverted to a previous version at any time. A repository is where all files and their versions are stored. Users who access a repository can "check out" a working copy of a file, which is a copy of the latest file to which users can make changes. After making desired changes, users can then "check in", or commit, the changes back to the repository, which creates a new version of the file that includes information about what changes were made and who made them. In most uses of version control, each user on a team has a full copy of the repository on his/her local machine. When making changes, the user commits changes to the local repository (on his/her personal machine), and then "pushes" changes back to the shared team's repository. A user can also "pull" changes from another repository to update files to the most current version. Branches are a way for users to work on making significant changes (like adding new features) while leaving the original files untouched and deployable (ENTP 2012). One can branch part of a project, and when changes are complete, merge the branch into the master copy (see Figure 7).

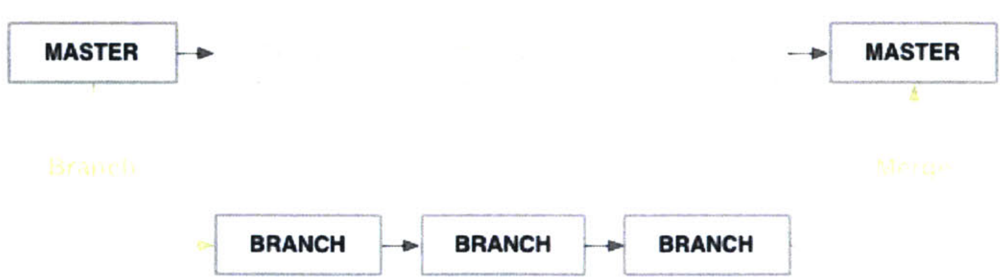

Fig. 7: Branching and merging in Git

Github, a popular web-based hosting service for software development projects, uses a concept called "forking" to make coding more social. When a user is interested in gaining access to files from another team/project's repository, he/she can "fork" the project, which creates a copy of the code in his/her own account. From this local copy, the user can make changes as he/she wants, and can submit a "pull request" to the original repository to see if the owner wishes to incorporate these changes into the original files.

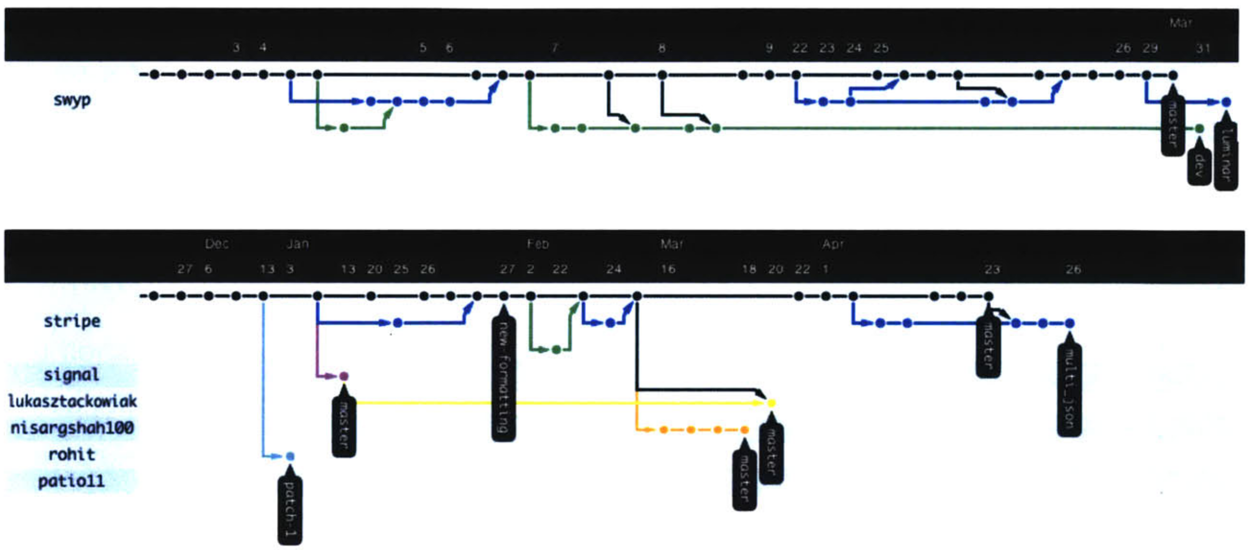

Fig. 8: Example activity on Github (github.com)

Figure 8 shows examples of activity on github, including forking branches, and pushing and pulling code.

Vehicleforge uses git as its version control system. Having a way to manage files is a critical component of an open design tool. With repositories sitting on users' local machines, workflow with vehicleforge is very similar to that of any other project. Updates from other users are pulled automatically, and changes can easily be pushed back, without having to do any uploading/downloading with the vehicleforge site. Working in version-controlled repositories also ensures that all files are accessible to all users at all times.

Components

A component is a way of organizing all artifacts of product development. It contains all the information required to follow a part's development, from initial sketches to stress testing to CAD files. This allows users to more easily transfer knowledge by sharing components instead of individual files.

Marketplace

The marketplace is where users share components and services. Users can browse through listings, fork components for their own use, and push back changes to the original component's owner.

DOME Services

The requirement of providing tools for concept feasibility verification is accomplished by giving users access to powerful engineering models and simulations. DOME, which stands for Distributed Object-Based Modeling Environment , is a tool designed by Prof. Wallace and his graduate students that allows computational services of models from anywhere to be shared, utilized, and combined to make new services, all online (Borland et al 2000) (Senin et al 2003). The services can be anything from a Matlab simulation to an Excel file to a Solidworks (Dessault Systems Solidworks Corp., 2012) finite element analysis, and are all executed remotely. Plug-ins can be written for virtually any software program, and services can be combined to create integrated simulations. Services are a fundamental part of democratizing the physical product development process. A large barrier to making physical product development analogous to open source software is the cost of engineering modeling programs. As the cost of digital fabrication falls, so will the cost of the the design software that goes along with it. Take the example of COMSOL (COMSOL, Inc., 2012), a powerful finite element analysis, solver, and simulation package, is reasonably priced at upwards of $7,000. This price point makes it inaccessible for the every day participant, but remains a critical step in the design process. Without simulations to verify how a digital design file will act in the physical world, withstanding temperatures and loading, it is very difficult to build a functional product.

DOME services act as toolkits for innovation, which have been recommended as a way to create truly open design development of physical products. They not only lower the barrier to participation and get users to participate (Raasch 2011), but also lower development times (Von Hippel 2005), as was the case in the semiconductor industry where integrating toolkits cut development time by 2/3.

The development of the vehicleforge site shown in this thesis was completed in collaboration with other colleagues at MIT and GE Global Research. My contributions involved front-end development, with sketches, mock-ups, storyboards, and web design. Vehicleforge is built using Bootstrap, a tool from Twitter for quickly developing websites and web applications. The code is hosted on github (http://twitter.github.com/bootstrap/) and includes HTML, CSS, and Javascript modules that are easy to integrate and modify for one's application.

The structure of the site is centered around its core functionality of developing products. There are projects, where teams form and development takes place, a marketplace, where users can share and explore components and services from other users, and a community page that hosts site-wide discussions, job postings, and other relevant information. The "splash page" is shown below in Figure 9.

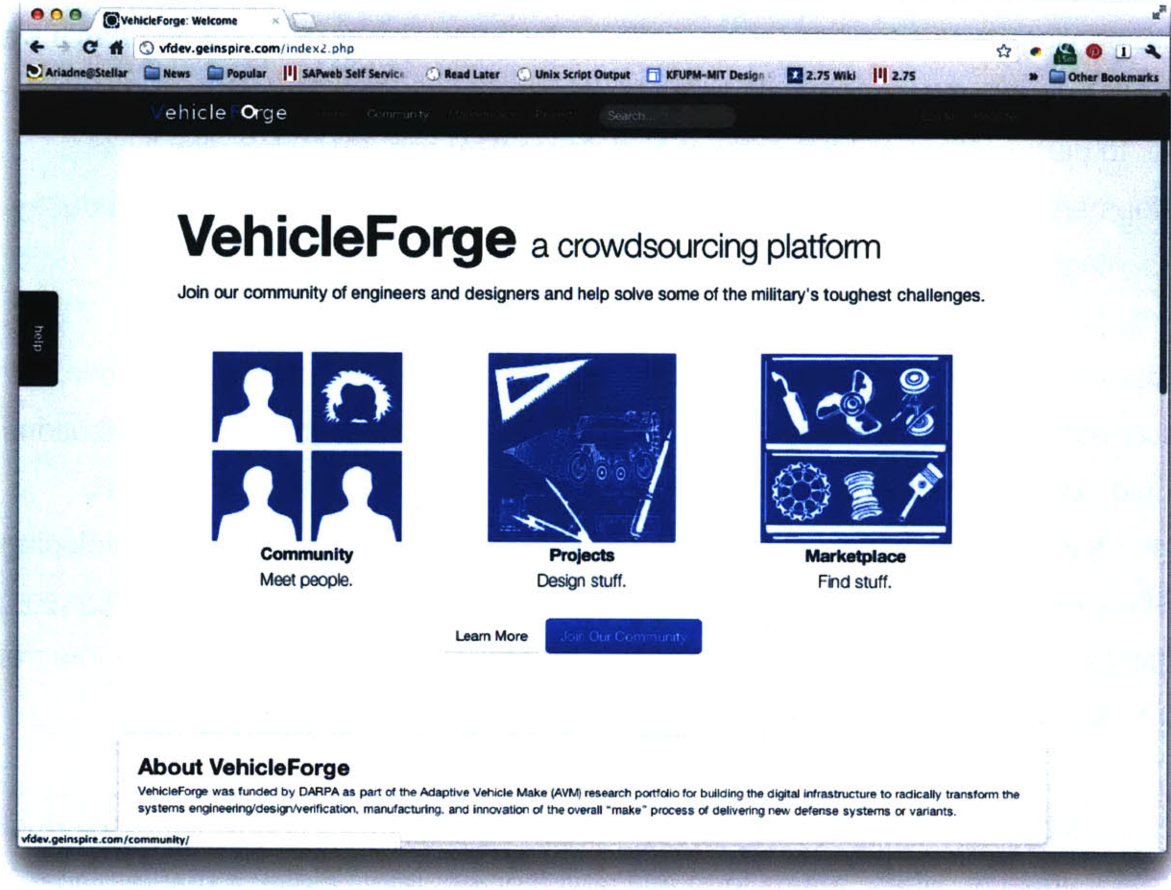

Fig. 9: Vehicleforge landing page

Registered users have pages where they can view the projects they're involved in and people that they are following, and get access to quick links to create projects. The dashboard and streams are inspired by Google Plus and Facebook style updates. Users can use streams to follow updates from users, projects, and components and can also be used for communication between users, as is seen in Figure 10, where Jason has sent a message to Tom. This facilitates communication between users and makes product development and project management more transparent.

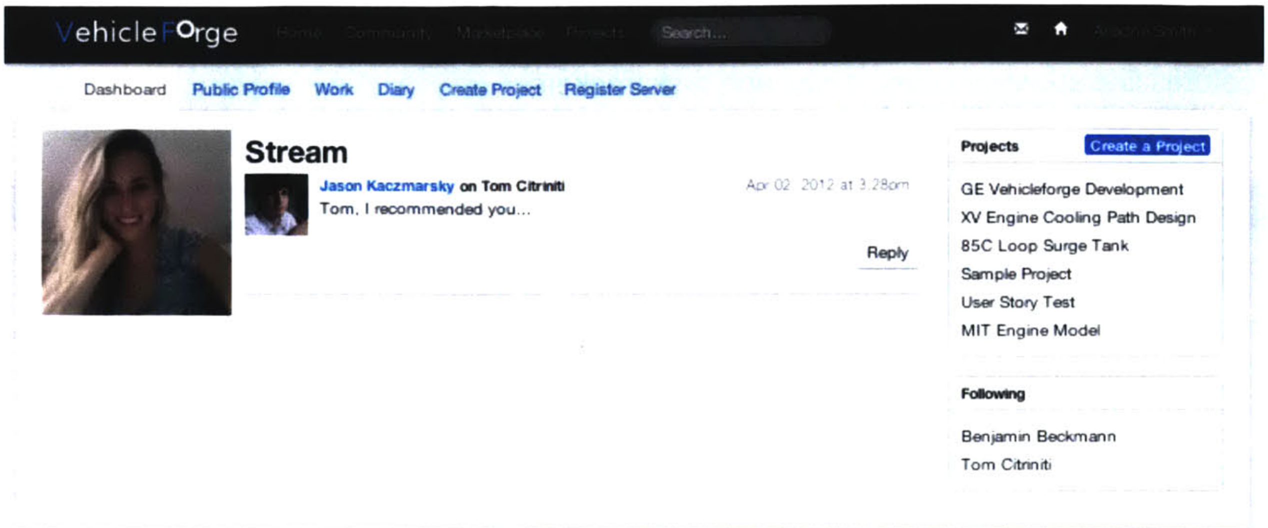

Fig. 10: Vehicleforge dashboard

Projects are where the product development work is organized. Each project shows team members, activity, components, services, and other project management tools, such as wikis and artifacts (similar to an issue tracker).

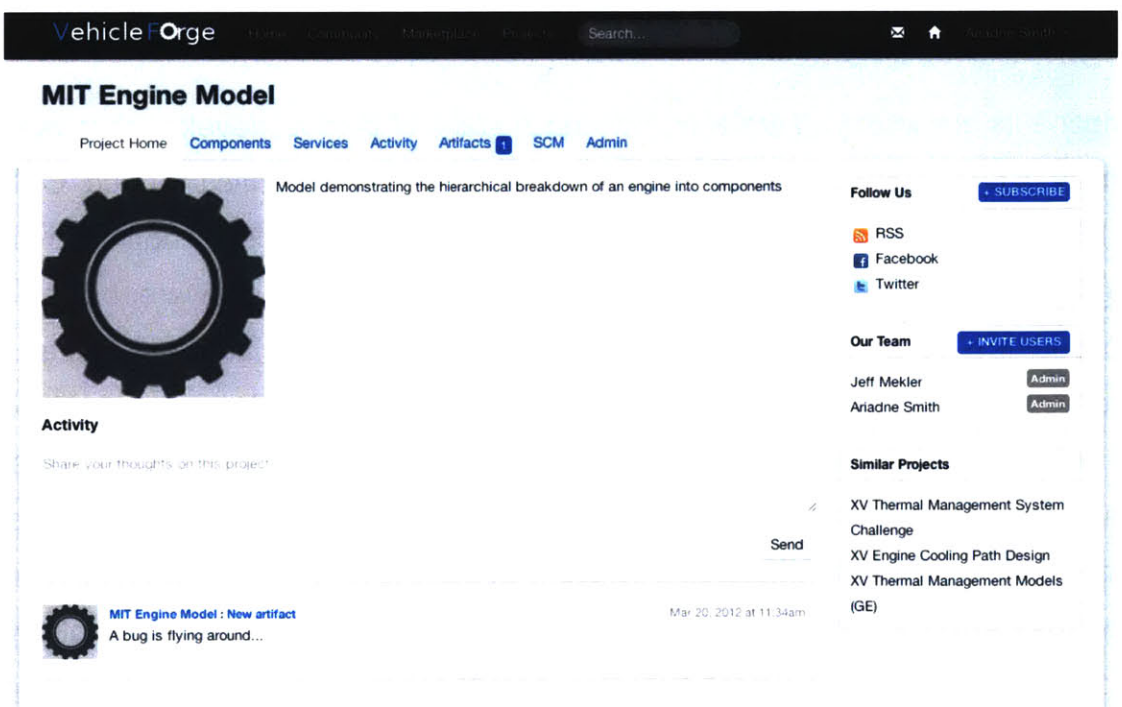

Fig. 11: Vehicleforge project view

Built-in project management tools are customizable. Users can add and remove features like surveys, news, wikis, and task trackers.

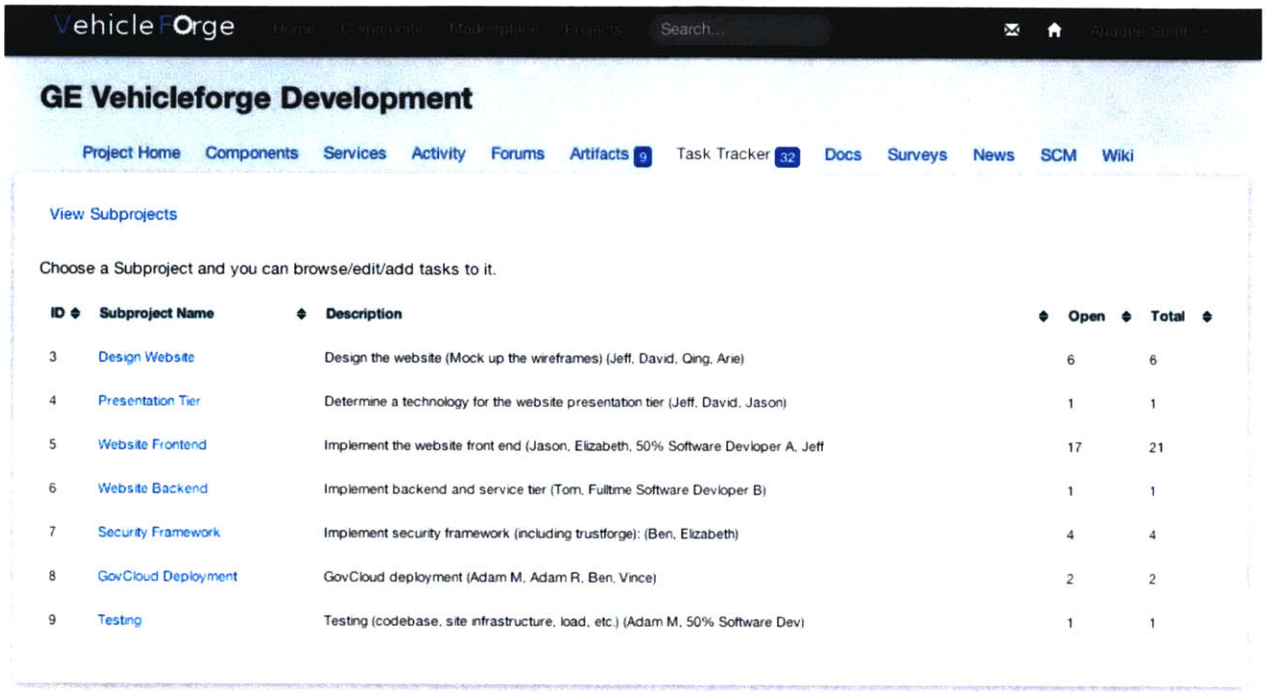

Fig. 12: Vehicleforge project task tracker

Components are where all artifacts, from each stage of product development, live. For example, a crankshaft pulley might include a sketch that a user has scanned, CAD files, or a bill of materials. Each component is its own repository and is version controlled using Git. This approach is a clearly moving hardware closer to software, as has been described as a trend in the literature (Thompson 2008). Arriving at the idea of a component was not simple. To software engineers, allowing each component its own repository makes sense. However, mechanical engineers think of components as physical parts that make up machines and other systems, such as springs or pulleys, whose dependencies are physical. Hierarchy of parts is most often represented in CAD assemblies where it is easy to see how parts are related. Components essentially act as containers for systems, where sub-components are the sub-systems. Including file types other than just CAD in components helps transfer information between team members. Mechanical engineering teams working on projects in classes at MIT use

Dropbox (Dropbox, Inc., 2012) to organize their files. This facilitates sharing files, but still makes organization difficult. A Solidworks assembly contains many parts, but no information on what work was done before a part was solid-modeled. A subcomponent that is a crankshaft pulley, for example, might contain a CAD file for the actual part, as well as a document with justifications for dimensions and pictures of a foam prototype. Grouping all of these files together helps communicate the design vision, as well as any prior work done, to any user looking at or working with the component.

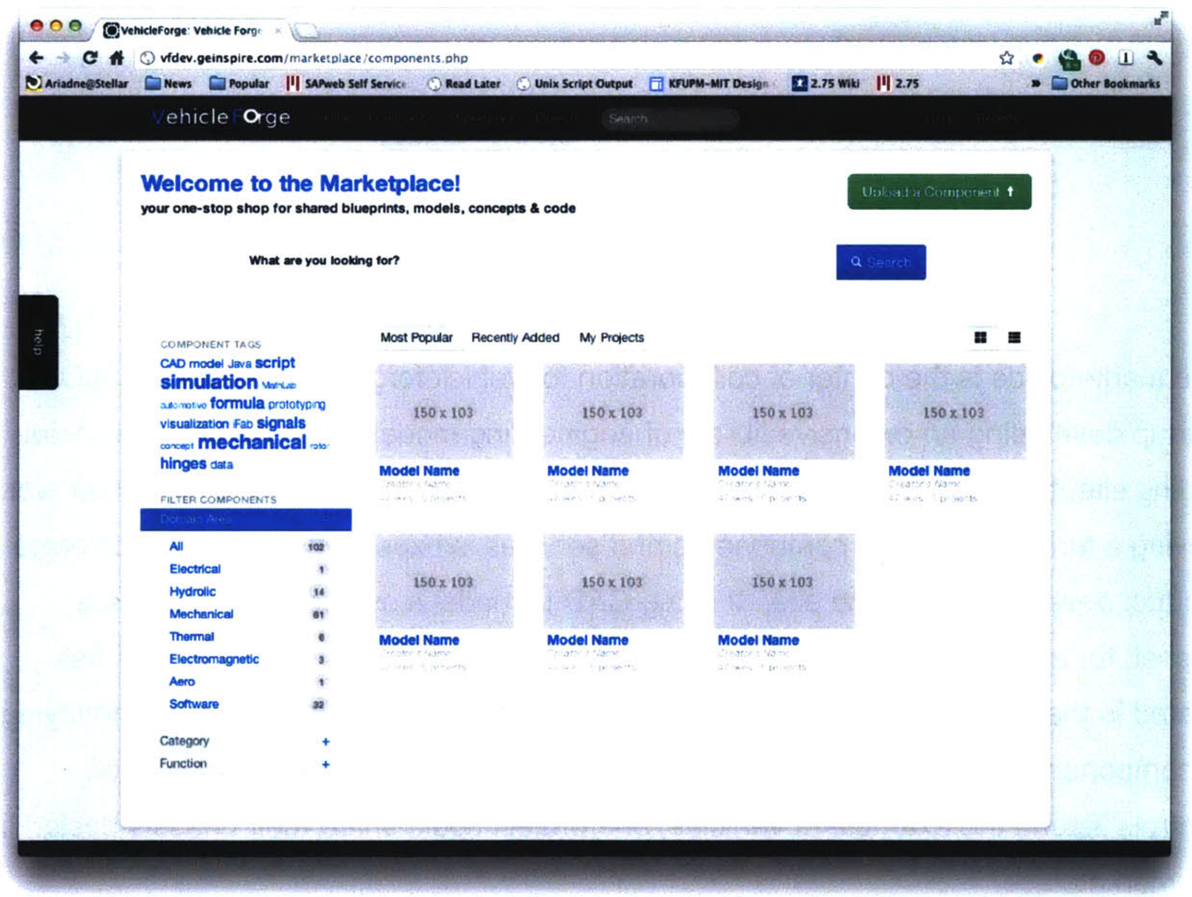

Fig. 13: Vehicleforge component view

Services are the DOME models described earlier. They take inputs (either metadata on a component or the component itself) and return useful information for future product iterations. A basic service that has been developed is one to calculate area. However, a more complicated service might be to run computational fluid dynamics (CFD) analysis of a shell for a human-powered vehicle to see where its aerodynamics might be improved.

Fig. 14: Area interface DOME service

The marketplace is the center of collaboration for vehicleforge. Inspired by GrabCAD, a startup developing an extensive library of engineering models, and Github, the social coding site, the marketplace is a place where users are encouraged to share their work. Having a thriving library of components and services is expected to rapidly accelerate product development on the site. If using CAD to model a system that includes a ratchet, for example, it is much easier to start with a ratchet that someone else has shared in the marketplace than to model one from scratch. The sharing and modifying of components and services in the marketplace will be managed through version control. Taking the example of vehicles, one project could share their chassis design in the marketplace.

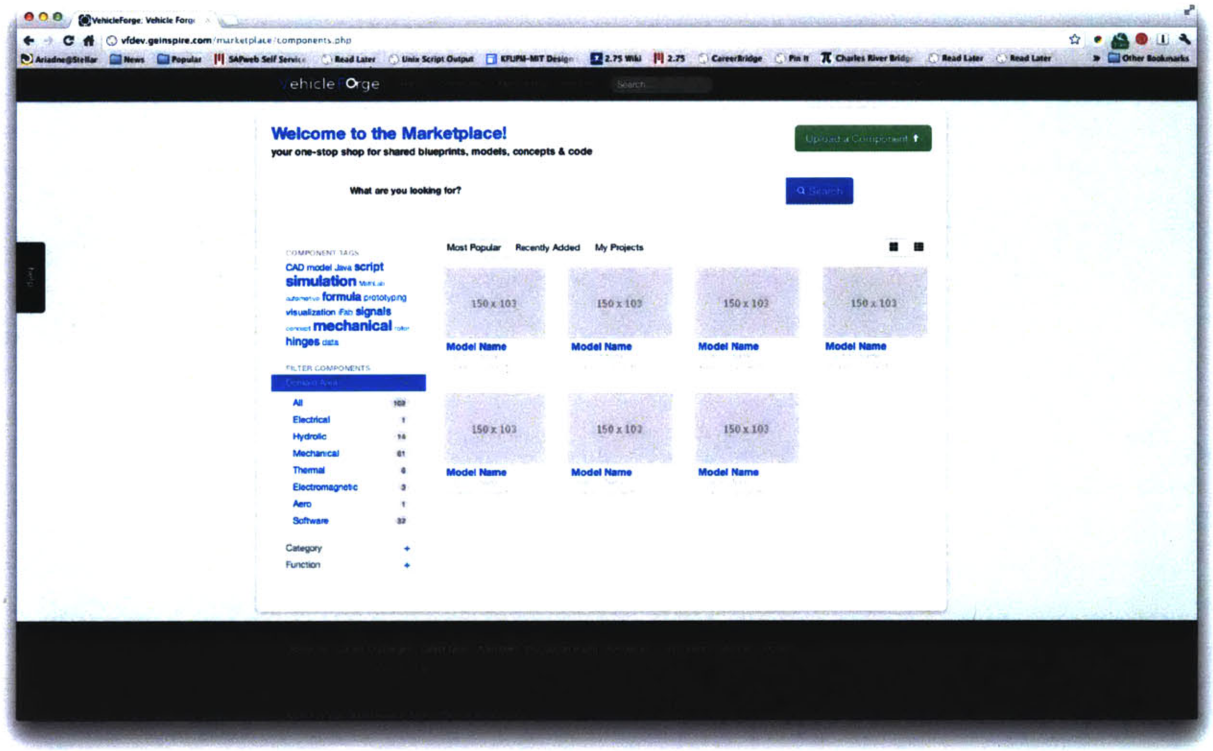

Fig. 15: Vehicleforge marketplace

Chapter 5

The initial development of vehicleforge shows how using version control, creating components as containers for development files, providing a marketplace for sharing models, and using DOME for engineering feasibility verification provides an environment in which physical products can be designed and tested by the "crowd". The second research question in this thesis investigates what types of design environments can benefit from this tool.

DARPA has shown strong interest in applying this tool to the development of military vehicles. In this thesis, the author is also interested in learning how organizations outside the military could benefit from a tool like vehicleforge, and what specific features are appealing to different organizations. Since this tool addresses a relatively new need of crowdsourcing physical product development, it was useful to create a framework that can help evaluate the application of this tool to other industries.

Open Design Framework

In her doctoral thesis, Open Source Product Development, Kerstin Balka (Balka 2011) presents a conceptual framework for studying open source innovation (Figure 16). In this model, actors, that include participants, investors, and manufacturers, collaborate to produce an innovative outcome within a development process.

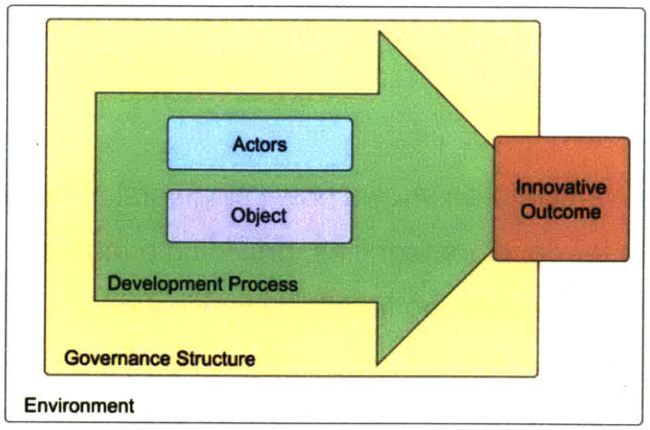

Fig. 16: The open source innovation framework (Balka 2011)

The term 'actors' refers to who is participating in the design process and describes their background and motivations for participating. 'Object' describes what the project is, with respect to technical complexity, cost, and uncertainty. 'Development Process' refers to who leads the project, what the phases are, and how efficiently development progresses. 'Governance Structure' is how the project deals with intellectual property and incentives. 'Environment' includes tools for communication and development. 'Innovative Outcome' is what is finally produced; it is measured by its degree of 'innovativeness', intended audience, and market success.

Balka analyzes 104 tangible good open design projects in order to determine what characteristics of the above components lead to successful open design projects. . Keep in mind that these projects were selected to fit her definition of open source design, which does not stipulate that users make contributions at each stage of the product development process.

In its early design stage, the tool was thought to be used by organizations with two main interests; crowdsourcing the development of a product or improving internal project management. Starting with Balka's conclusions on what characteristics make for successful open design projects, the following framework was developed to help determine what other industries and organizations, besides DARPA, might benefit from a vehicleforge-like tool.

Object; what is designed

- engineered, physical product of low to medium complexity
- artifacts of product development must be shared online
- potential for coordination with physical phases of product development (i.e. prototyping or manufacturing)

Actors; community that is designing

- must be attracted from a wide audience
- diverse (but highly specialized) backgrounds

Incentives:

- can offer actors
    - IP recognition
    - monetary rewards
    - social recognition
- can offer organizations
    - more efficiency in internal product development
    - improved communication between groups

Process & Outcome

- development can be separated into manageable pieces
    - clear milestones & checkpoints
- contributions can be turned into physical prototypes, plans for manufacture, and/or real products

Applications in Industry

The motives for DARPA to use an open design tool is clear: drastically compress military vehicle development times and receive innovative external input through the process. Why should other organizations, outside of the military, be interested as well? As discussed in Chapter 2, open design offers firms new and original perspectives and increases user engagement, effective marketing, and more pleasing solutions to consumers. In addition to providing all this through open design, the vehicleforge tool also facilitates project management by:

- organizing all development files in one place
- helping users and teams visualize the product development process through projects and milestones
- increasing and streamlining communication both within and between teams
- allowing for integration with agile development techniques (task tracking, etc)

Balka's survey of the 104 open design firms spanned across the following industries:

1. consumer electronics
2. IT hardware
3. machinery
4. automotive
5. consumer products
6. energy and utilities
7. pharmaceutical and healthcare

Based on the revised characteristics in the open design framework, the following industries were selected for further investigation as to their potential use of the vehicleforge tool:

1. healthcare
2. education
3. consumer products
4. aviation
5. defense contractor

Interviews

Process

Interviews with individuals from companies in the five industries described in the previous section were conducted in order to learn about the applicability of an open design tool for physical product development outside of the military. The goal of the interviews was to find preliminary evidence for future market adoption of the tool, and to serve as a stepping stone for future development. Vehicleforge is still in very early stages, and while these interviews are by no means comprehensive, they aim to guide research in useful directions.

Interviews were conducted over a period of two months in the spring of 2012, in a mixture of in-person, phone, and email settings. The interview questions and background material is included in the Appendix B.

Results

Education:

Based on interviews with three MIT students working on various engineering projects, I was able to paint a picture of how a tool like vehicleforge would be useful in the university context.

The features that students were most excited about were for use in class projects. In a non-competitive setting, they are eager to share their progress with each other and saw benefits from learning from their peers. At MIT, when students are stuck on a problem, they usually ask other classmates. Having a site where students can easily ask questions of their peers and view previous work could make this exchange even easier.

Using the tool as a way of documenting work was another desirable feature. Students could use this feature to explain how their products evolved. One student suggested including functionality that could help students generate a "portfolio view" of their projects within the site.

Defense:

Due to the sensitive nature of most defense projects, crowdsourcing full product development processes was not as applicable to the defense contractors interviewed as would be potential use of a project management tool. A small amount of community involvement could be present in gathering information on new technologies and ideas, but would be limited to the very early concept generation stage. Companies in this industry expressed interest in using DOME services as a tool during design reviews in order to make them more objective, as well as in testing multiple candidates for a particular component. A tool that incorporated DOME services, like vehicleforge, could also help move organizations to describe models in a more objective way in terms of data simulations instead of words.

Aviation:

Product development in the aviation industry is a good candidate for vehicleforge's internal product management potential due to the long development time and multidisciplinary engineering involvement. Again, like defense contractors, involving external users deeply in the design process is difficult for security reasons. There is interest (as with most companies) for a better project management tool for internal development. In one company interviewed, DOME services are already being used to facilitate multidisciplinary studies, design for variation, and optimization. The services link models from different engineers' computers together and allows them to see real time changes in model inputs and outputs.

Medical Device:

The medical device industry is one that undergoes some of the most rigorous safety regulation, and thus relies on heavy reliability testing. This is a characteristic that the author had overlooked when preparing industries in which to conduct interviews. Because so much testing must be physical and can't necessarily be replaced with a DOME service, open design of medical devices becomes difficult when users do not (as is the case with most) have access to the relevant test equipment to verify safety and feasibility. One potential use for crowdsourcing is gathering data on the timescale and cost of engineering projects to help companies better predict how long and expensive a project will be, based on empirical data. I spoke with a CEO of a small medical device company who said that poor project scope estimates are a chronic problem that could be mitigated with a slightly different application of crowdsourcing.

Consumer Products:

There has already been the expression of considerable interest in consumer product companies to integrate users into their product development process. Speaking with a prominent furniture design firm revealed a few other needs for a tool like vehicleforge. Though there was interest expressed in involving external users to optimize user studies and customize products, this particular firm found a need for better internal coordination, especially with designers working in different countries for different markets. An interesting idea that emerged was using the tool to allow customers to make their own tradeoffs between a product's cost, quality, and time to delivery. For example, if a customer wanted office furniture for a very urgent deadline, she could decide what features to forgo to have it delivered on time. Tighter integration with current iFab software that predicts manufacturing cost and time could help with this need.

Tangible Future Directions

There are a number of interesting directions that vehicleforge can take for future development. Crowdsourcing the development of an entire complex physical product is a large endeavor. For the industries of defense, aviation, and medical device, security and safety currently prohibit external crowdsourcing of their products. It would be more interesting to use vehicleforge in these cases as an internal tool to test how it aids inter-team collaboration and if companies can come to rely on DOME to accelerate product verification and testing.

The idea of adapting vehicleforge to work as an educational tool can have many merits. MIT has many product design classes that could take advantage of a tool to document work and aid collaboration, and it can also be used to test DOME simulations as a method of prototyping.

Interest from firms producing consumer products is significant, however testing first within a smaller, more controlled environment, such as a class at MIT, might be a good way to resolve any fundamental design issues and deploy new features before working with a larger commercial company.

Chapter 6

Conclusions

In the past few years, crowdsourcing has become somewhat of a phenomenon. It has given us data and ideas, solutions to our problems, and funding for our startups. Crowdsourcing relies on the principle of decentralized innovation, where ideas are conceived and developed outside the traditional bounds of a firm. Much of the recent movements were inspired by open source software, in which the design and continual improvement of software applications is accomplished by the community. There has been recent interest in applying principles of open source software development to physical product development. Coined as 'open design', it has been applied to select parts of the product development process in several industries. Like its derivate movements of open source software development and user-innovator networks, open design has the potential to compress development times and increase the quality of products.

This thesis has examined recent applications of open design to physical product development and revealed that they tend to span only the early stages of the product development process.

The question of what kind of tool can be developed to support crowdsourcing the full development of a physical product was addressed with the early development of such a tool for DARPA's AVM project. Encompassing every stage of product development is difficult due to need for physical prototyping, manufacturing, and distribution. The key elements of the tool designed in this thesis address these needs in open design projects. Streams and projects facilitate communication and collaboration across the site. Components and version control integration allow for seamless knowledge transfer and file sharing. The Marketplace gives users a place to exchange and build open other's work. DOME services allow for the necessary verification and testing of engineered parts. As this is a first iteration, there is much that can be built upon the tool. Methods for including incentives, milestones, community activity, and physical prototyping are all needed to make vehicleforge a successful and useful product.

Interviews with lead users in education, medical device, aviation, defense, and consumer product design industries addressed the question of what types of design environments can benefit from this tool. There is interest in using vehicleforge as an internal tool for project management, with DOME services as a means of facilitating inter-team development and making design reviews more objective. The sensitive nature of projects in the medical device, defense, and aviation industries prohibits the use of external user involvement. However, education and academic environments could greatly benefit from the features that the vehicleforge tool currently has to offer. A good method for testing additional features would be to develop a version of vehicleforge for use in an educational setting, such as MIT product development courses. This would allow the tool to be refined before being offered to other interested organizations.

Future Work

The version of VehicleForge presented in this thesis is very early-stage and meant to demonstrate concepts that will be integrated into its final implementation. The core features are ones that make the open design of physical products a possibility: Using version control for files allows users to easily share and collaborate on artifacts of the design process (Anderson 2012). Projects, streams, and task trackers allow users to have better communication and project management while working. Components organize all relevant information about parts in one place, facilitating sharing development both in projects and also in the marketplace. DOME services allow for feasibility verification of concepts, and also give users access to software and simulations required throughout the design process.

VehicleForge requires much more development before it is a usable open design and crowdsourcing tool. There are a few areas that have not yet been addressed in this early stage of development, and would need to be integrated into any open design tool.

Incentives:

Currently, little thought has been put into how users will be incentivized to share their contributions on the site. Providing incentives in private-collective innovation environments, such open design, is crucial to for knowledge-sharing to occur (Gächter et al 2010). The prizes that DARPA has presented are substantial; a $1 million prize for designing a drivetrain, $1 million prize for designing a chassis, and $2 million prize for designing the entire vehicle. These prizes will motivate users to join VehicleForge and submit designs in hopes of winning. Fostering a vibrant collaborative community between competing teams is more difficult. Interviews with the MIT team participating in the UAVforge challenge revealed that teams were not interested in collaborating via forums and sharing components when competing with each other. The goal with Vehicle Forge is to have an active marketplace where users are continually sharing components and building upon designs of others. There are a couple of ways in which these can be accomplished. The first is to ensure that the site has some non-competitive projects taking place at any time. LocalMotors hosts several competitions running at once, but also encourages users to start their own projects. The same is true of GrabCAD. The result is a large library of sketches and CAD models that community members comment on. Another technique is to create a concrete incentive for users to share and comment on parts. Economic incentives matter greatly in initial knowledge sharing decisions (Gacht, Krogh, Haelfliger). Users who shared components could work towards small monetary rewards, either in real currency or one that is only used on the site, and can be traded for access to special components or services. Users also respond well to social recognition as an incentive to participate (Oreg and Nov, 2008). GrabCAD utilizes a system of badges to recognize users for various accomplishments. These acknowledgements users stand out during team selection or job recruiting, which relates to the forthcoming community section of the site.

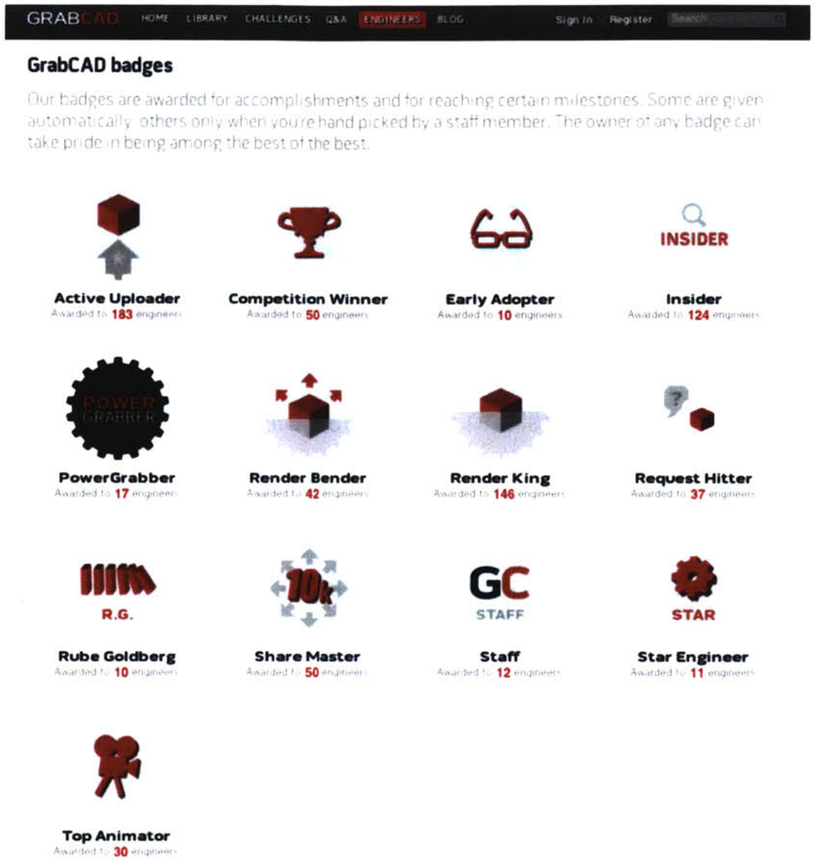

Fig. 17: GrabCAD incentive badges (grabcad.com)

<u>Community:</u>

The top navigation bar on vehicleforge shows a community tab that has yet to be developed. The community section of the site is where users go to find team members, post jobs, or join general discussions. Users will have a profile in which they can list relevant skills, experience, and interests, in addition to site statistics, such as number of contributions or shares. This should facilitate forming a team of diverse skills, as is necessary with a cross-disciplinary project like a vehicle. Users can further be incentivized to share files and make contributions to projects so that they appear as more attractive candidates to employers.

<u>Milestones:</u>

The current features on vehicleforge (wikis, streams, task trackers) provide for project management within projects. However, there needs to be ways for competition sponsors (in the case of vehicleforge, DARPA) to manage the development that goes on within these projects. Existing crowdsourcing sites use milestones or checkpoints to move teams along. UAVForge has a series of video submissions that are used to prove completion of various milestones, such as proof of flight. OpenIDEO divides work into different sections that alternate between users presenting and refining concepts and community-members voting up the best ideas. Vehicleforge needs a set of milestones in order to guide development. DOME services should be used as checkpoints for teams throughout the process. This can make feasibility verification and concept down-selection much more objective than video or community voting. For example, as a checkpoint, each team designing a chassis for a cargo-carrying could be required to show a stress analysis under different loading (cargo) conditions. Successful teams could easily be chosen based on whether or not they passed a checkpoint. Further selection can be provided by community judging, provided that teams have met the engineering requirements put forth.

Physical Prototyping:

Physical prototyping is an integral part of the product development process and should not be excluded from the vehicle forge design process. Besides UAVForge, other crowdsourcing efforts in physical product development include only design that can be completed and shared on a computer. Users on Vehicle Forge should be encouraged to make physical sketch models to test their concepts. However, prototyping requires materials and space, both of which cost money. There are a couple of ways to make this happen. After passing certain milestones, vehicleforge teams could be allotted a certain amount of money to buy materials, and could be required to present prototypes for future milestones. Vehicleforge could also partner with local fab labs or machine shops (like Tech Shop) to provide space and resources for users participating in the site's competitions.

References

99Designs, Inc. . (2012). Retrieved August 1, 2012, from http://99designs.com/

AVM Collaboration Capability (vehicleforge). (2012). Retrieved from http://www.darpa.mil/Our_Work/TTO/Programs/AVM/AVM_Collaboration_Capability_(Vehicleforge_mil).aspx

AVM Design Tools (META). (2012). Retrieved August 1, 2012, from http://www.darpa.mil/Our_Work/TTO/Programs/AVM/AVM_Design_Tools_(META).aspx

AVM Manufacturing Foundary (iFAB). (2012). Retrieved August 1, 2012, from http://www.darpa.mil/Our_Work/TTO/Programs/AVM/AVM_Manufacturing_Foundary_(iFAB).aspx

Adaptive Vehicle Make (AVM). (2012). Retrieved August 1, 2012, from http://www.darpa.mil/Our_Work/TTO/Programs/Adaptive_Vehicle_Make__(AVM).aspx

Alibre, Inc. (2012). Retrieved August 1, 2012, from http://www.alibre.com/

Allen, R. C. (1983). Collective Invention, (February).

Anderson, C. (2012). Wanted: Version Control for Stuff. Wired Magazine. Retrieved July 5, 2012, from http://www.wired.com/design/2012/05/we-need-version-control-for-real-stuff/

Arduino. (2012). Retrieved August 1, 2012, from http://www.arduino.cc/

Autodesk, Inc. (2012). Retrieved August 1, 2012, from http://autodesk.com/

Baldwin, C. Y., Clark, K. B., David, P., Bessen, J., Henkel, J., Aoki, M., Ikeda, N., et al. (2005). The Architecture of Participation : Does Code Architecture Mitigate Free Riding in the Open Source Development Model ?

Balka, K. (2011). Open Source Product Development, 3-8. Wiesbaden: Gabler. doi: 10.1007/978-3-8349-6949-1

Balka, K., Raasch, C., & Herstatt, C. (2009). Open Source beyond software : An empirical investigation of the open design phenomenon, (April).

Belfiore, M. (2012). Adaptive Vehicle Make: DARPA's Plan to Revolutionize Auto Manufacturing. Popular Mechanics. Retrieved March 6, 2012, from http://www.popularmechanics.com/technology/military/news/adaptive-vehicle-make-darpas-plan-to-revolutionize-auto-manufacturing-6646618?click=pm_latest

Benkler, Y. (n.d.). Coase's Penguin, or, Linux and "The Nature of the Firm."

Borland, N., & Wallace, D. (2000). Environmentally Conscious Product Design A Collaborative Internet-based Modeling Approach. Journal of Industrial Ecology, 3(2), 33-46.

Brabham, D. C. (2008). Crowdsourcing as a Model for Problem Solving: An Introduction and Cases. Convergence: The International Journal of Research into New Media Technologies, 14(1), 75-90. doi:10.1177/1354856507084420

CNET. (2008). Pretend you're a venture capitalist, with vencorps. Retrieved August 1, 2012, from http://news.cnet.com/8301-17939_109-9959387-2.html

COMSOL, Inc. (2012). Retrieved August 1, 2012, from http://www.comsol.com/

Capello, R., & Faggian, A. (2005). Collective Learning and Relational Capital in Local Innovation Processes. Regional Studies, 39(1), 75-87. doi: 10.1080/0034340052000320851

Chesbrough, H. W., & Appleyard, M. M. (2007). Open Innovation and Strategy, 50(1), 57-77.

Defense Sciences Office. (2012). H1N1 Acceleration (Blue Angel). DARPA. Retrieved April 10, 2012, from http://www.darpa.mil/Our_Work/DSO/Programs/H1N1_Acceleration_(BLUE_ANGEL).aspx

Dessault Systems Solidworks Corporation. (2012). Retrieved August 1, 2012, from http://www.solidworks.com/

Dillow, C. (2011). A Record-Breaking Kickstarter Success Turns an iPod Nano into a Wristwatch. Popular Science. Retrieved April 22, 2012, from http://www.popsci.com/gadgets/article/2011-05/record-breaking-kickstarter-success-turns-nano-wristwatch

Dropbox, Inc. (2012). Retrieved August 1, 2012, from https://www.dropbox.com/

FANG. (2012). Retrieved August 1, 2012, from http://www.darpa.mil/Our_Work/TTO/Programs/AVM/AVM_Design_Competitions_(FANG).aspx

Fabricant, R. (2012). Kickstarter Rescues Startups that VCs Won't Touch, But Here's What's Missing. Co.Design. Retrieved from http://www.fastcodesign.com/1669698/kickstarter-rescues-startups-that-vcs-wont-touch-but-heres-whats-missing

Franke, N., & Shah, S. (2003). How communities support innovative activities: an exploration of assistance and sharing among end-users. Research Policy, 32(1), 157-178. doi:10.1016/S0048-7333(02)00006-9

General Electric Corp. (2012). Retrieved August 1, 2012, from http://www.ecomagination.com/

GeniusRocket, Inc. (2012). Retrieved August 1, 2012, from http://www.geniusrocket.com/

GitHub, Inc. (2012). Retrieved August 1, 2012, from https://github.com/

GrabCAD Inc. (2012). Retrieved August 1, 2012, from http://grabcad.com/

Gächter, S., von Krogh, G., & Haefliger, S. (2010). Initiating private-collective innovation: The fragility of knowledge sharing. Research Policy, 39(7), 893-906. doi:10.1016/j.respol.2010.04.010

Harhoff, D., Henkel, J., & von Hippel, E. (2003). Profiting from voluntary information spillovers: how users benefit by freely revealing their innovations. Research Policy, 32(10), 1753-1769. doi:10.1016/S0048-7333(03)00061-1

Henkel, J. (2006). Selective Revealing in Open Innovation Processes : The Case of Embedded Linux, (March).

Herstatt, C. (2011). Free revealing in open innovation : a comparison of different models and their benefits for companies Tim Schweisfurth *, Christina Raasch, 13(2).

Hoeken, Z. (2008). Objects as Software: The Coming Revolution. 25th Chaos Communication Congress . Retrieved May 1, 2012, from http://events.ccc.de/congress/2008/Fahrplan/events/2781.en.html

Hope, J. E. (2004). Open Source Biotechnology, (December).

Howe, J. (2006). The Rise of Crowdsourcing. Wired Magazine. Retrieved April 10, 2012, from http://www.wired.com/wired/archive/14.06/crowds.html

IBM InnovationJam 2008. (2008). Retrieved August 1, 2012, from http://www.ibm.com/ibm/jam/

IMPROVER systems biology verification. (2012). Retrieved August 1, 2012, from http://www.sbvimprover.com/

IdeaBounty, Inc. (2012). Retrieved August 1, 2012, from http://www.ideabounty.com/

Indiegogo, Inc. (2012). Retrieved August 1, 2012, from http://www.indiegogo.com/

InnoCentive, Inc. (2012). Retrieved August 1, 2012, from http://www.innocentive.com/

Jackson, B. (2012). In its third year, Kickstarter successfully raises over $119 million, taking home $6 million in commission. . The Next Web. Retrieved April 22, 2012, from http://thenextweb.com/insider/2012/04/22/in-its-3rd-year-kickstarter-successfully-raises-over-119-million-taking-home-6-million-in-commission/

Kickstarter, Inc. (2012). Retrieved August 1, 2012, from http://www.kickstarter.com/

Lakhani, K. R., Lohse, P. A., Panetta, J. A., & Jeppesen, L. B. (2007). The Value of Openness in Scientific Problem Solving.

Linton, M. (2011). Startup Monday: Crowdsourcing Venture Capital. Life in a Web 3.0 World. Retrieved April 22, 2012, from http://morganlinton.com/startup-monday-crowdsourcing-venture-capital/

Local Motors, Inc. (2012). Retrieved August 1, 2012, from http://local-motors.com/

Lohr, S. (2012). BITS; Seeking Ideas For U.S. Defense. The New York Times. New York, NY. Retrieved April 9, 2012, from http://query.nytimes.com/gst/fullpage.html?res=9D07E4D8153EF93AA35757C0A9649D8B63&ref=stevelohr

Luis von Ahn: one word at a time. (2011).Digital Studio for Public Humanities (University of Iowa). Retrieved April 4, 2012, from http://dsph.uiowa.edu/tag/crowdsourcing/

Makerbot Industries, LLC. (2012). Retrieved August 1, 2012, from http://makerbot.com/

Meyer, P. B. U. s. B. of L. S. (2003). Episodes of Collective Invention.

Ogawa, S., & Piller, F. T. (2006). Reducing the Risks of New Product Development. MIT Sloan Management Review. Retrieved March 3, 2012, from http://sloanreview.mit.edu/the-magazine/2006-winter/47214/reducing-the-risks-of-new-product-development/

Open Hardware 3 Years On. (2008).Rowetel: open telephony and open hardware. Retrieved April 6, 2012, from http://www.rowetel.com/blog/?p=75

OpenIDEO. (2012). Retrieved August 1, 2012, from http://www.openideo.com/

Pebble: E-Paper Watch for iPhone and Android. (2012). Retrieved August 1, 2012, from http://www.kickstarter.com/projects/597507018/pebble-e-paper-watch-for-iphone-and-android

Pepsi Design Our Can. (2008). Retrieved August 1, 2012, from http://www.designourpepsican.com/

Ponoko, Inc. (2012). Retrieved August 1, 2012, from http://www.ponoko.com/

Powell, W. W., & Giannella, E. (2009). Collective Invention and Inventor Networks, (October).

Proto Labs, Inc. (2012). Retrieved August 1, 2012, from http://www.protolabs.com/

Quirky, Inc. (2012). Retrieved August 1, 2012, from http://www.quirky.com/

Raasch, C. (2011). The sticks and carrots of integrating users into product development. International Journal of Technology Management, 56(1), 21. doi:10.1504/IJTM. 2011.042460

Raasch, C., & Herstatt, C. (2011). How companies capture value from open design. International Journal of Information and Decision Sciences, 3(1), 39. doi:10.1504/ IJIDS.2011.038840

Raasch, C., Herstatt, C., & Balka, K. (2009). On the open design of tangible goods On the open design of tangible goods.

RepRap - RepRap Wiki. (2012). Retrieved August 1, 2012, from http://reprap.com/wiki/ Main_Page

Rockethub, Inc. (2012). Retrieved August 1, 2012, from http://www.rockethub.com/

Schweisfurth, T., Raasch, C., & Herstatt, C. (2011). Free revealing in open innovation: a comparison of different models and their benefits for companies. International Journal of Product Development, 13(2), 95 - 118.

Senin, N., Wallace, D. R., & Borland, N. (2003). Distributed Object-Based Modeling in Design Simulation Marketplace. Journal of Mechanical Design, 125(1), 2. doi: 10.1115/1.1540993

Shah, S. (2000). Sources and Patterns of Innovation in a Consumer Products Field : Innovations in Sporting Equipment Sonali Shah * Massachusetts Institute of Technology Sloan Working Paper # 4105 March 2000 Initial Draft January 2000 Revised November 2000, 1-27.

Shirky, C. (2007). Generalizing Peer Production into the Physical World. Yahoo Groups. Retrieved March 30, 2012, from http://finance.groups.yahoo.com/group/ decentralization/message/6967

Smarterer, Inc. . (2012). Retrieved August 1, 2012, from http://smarterer.com/

Snow, C. C., Fjeldstad, Ø. D., Lettl, C., & Miles, R. E. (2011). Organizing Continuous Product Development and Commercialization: The Collaborative Community of Firms Model. Journal of Product Innovation Management, 28(1), 3-16. doi: 10.1111/j.1540-5885.2010.00777.x

Sukkasi, S. (MIT). (2008). Commons-Oriented Information Syntheses: A Model for User-Driven Design and Creation Activities.

TechShop, Inc. (2012). Retrieved August 1, 2012, from http://www.techshop.ws/

The Open Source Science Project. (2008). Retrieved from http:// www.theopensourcescienceproject.com/

Thingiverse. (2012). Makerbot Industries, LLC. Retrieved from

Threadless. (2012). SkinnyCorp, LLC. Retrieved August 1, 2012, from http://www.threadless.com/

Trimble, Inc. (2012). Retrieved August 1, 2012, from http://sketchup.google.com/

UAVforge.net. (2012). Retrieved August 1, 2012, from http://www.uavforge.net/uavhtml/

Ulrich, K. T., & Eppinger, S. D. (2012). Product Design and Development (Fifth.). New York, NY: McGraw-Hill .

Von Hippel, Eric. (2005). Democratizing Innovation. Cambridge, MA: MIT Press.

Von Hippel, Eric, & Krogh, G. V. (2003). Open Source Software and the " Private-Collective " Innovation Model : Issues for Organization Science, 14(2).

Von Hippel, Eric, Ogawa, S., & P.J. de Jeong, J. (2011). The Age of the Consumer-Innovator. MIT Sloan Management Review. Cambridge, MA. Retrieved April 2, 2012, from http://sloanreview.mit.edu/the-magazine/2011-fall/53105/the-age-of-the-consumer-innovator/#ref16

Waze Mobile. (2012). Retrieved August 1, 2012, from http://www.waze.com/

Wortham, J. (2012). Start-Ups Look to the Crowd. The New York Times, p. B1. New York, NY. Retrieved from http://www.nytimes.com/2012/04/30/technology/kickstarter-sets-off-financing-rush-for-a-watch-not-yet-made.html?pagewanted=1&#59&_r=1;pagewanted=all

X Prize Foundation. (2012). Retrieved August 1, 2012, from http://www.googlelunarxprize.org/

entp. (2012). Version Control: Git for Designers. Retrieved March 12, 2012, from http://hoth.entp.com/output/git_for_designers.html

von Hippel, E. (2007). Horizontal innovation networks--by and for users. Industrial and Corporate Change, 16(2), 293-315. doi:10.1093/icc/dtm005

von Krogh, G., & von Hippel, E. (2006). The Promise of Research on Open Source Software. Management Science, 52(7), 975-983. doi:10.1287/mnsc.1060.0560

Appendices

Appendix A: DARPA BAA

https://www.fbo.gov/spg/ODA/DARPA/CMO/DARPA-BAA-10-88/listing.html

Program Scope and Structure

The *vehicleforge.mil* program will consist of two technical areas. Technical Area One covers the development, launch and maintenance of a collaborative design environment for large, complex cyber-electro-mechanical systems. It consists of a 12-month initial phase, focusing on the development of the open source environment and website, followed by a 36-month follow-on phase, where the principal tasks include enabling operations, executing maintenance, and providing technical support for the development environment created during the base period. Technical Area Two covers the generation of innovative credentialing and verification schemes to more accurately determine the trusted status of participants in an open source development environment. It consists solely of one 12-month base period.

Offerors may submit to one or both technical areas. Offerors that choose to submit to both technical areas may include one as an option to the other. In that case, however, offerors must describe the totality of their proposal within the page limits specified in the BAA. Instead, so as to afford adequate space to develop the technical concepts in some depth, independent proposals to the two technical areas are encouraged. Cost proposals for Technical Area One **must** include a 12-month base period and three successive 12- month option periods. Cost proposals for Technical Area Two should include only a 12- month base period.

Technical Area One: Collaborative Design Environment

The principal objective of the *vehicleforge.mil* effort is to generate an open source development collaboration environment and website for the creation of large, complex, cyber-electro-mechanical systems by numerous unaffiliated designers. The following capabilities are desired:

1. Hosting of large, complex, cyber-electro-mechanical system representation models in one or more metalanguages which are being developed by META and META-II performers. For proposal development purposes, offerors should plan and demonstrate compatibility with at least SysML (Systems Modeling Language), Modelica, and AADL (Architecture Analysis & Design Language).

2. Hosting of multiple projects, including: public projects for which access is not limited; semi-public projects for which access is based on verification of U.S. person status for export control purposes; and private projects for which access is limited to a fixed set of individuals or entities.

3. A version control system, enabling check-out and check-in, reversion, bug/issue tracking, and other features that enable effective collaborative development and project maintenance.

4. Public and private version branching.

5. Collection of metrics and statistics on access patterns, user activity, and design (re)use.

6. Collaboration tools including a wiki, message board, mailing lists, and other features that enable effective collaborative development.

7. User registration and "U.S. person" status verification for export control compliance.

8. Information assurance controls sufficient to protect export controlled and contractor proprietary information in accordance with Department of Defense, Department of State, Department of Commerce, and any other statutory or regulatory requirements or best practices.9

9. Support for information assurance testing, certification, and accreditation activities.

10. Public hosting and dissemination of electronic materials and announcements associated with a series of Adaptive Make Challenges.

11. Documentation appropriate in scope and quality to enable effective development and maintenance of the *vehicleforge.mil* code base by third parties unaffiliated with the proposer.

[9] See, e.g., ASD(NII), Directive-Type Memorandum (DTM) 08-027, "Security of Unclassified DoD Information on Non-DoD Information Systems."

1. Domain name registration, hosting, and upkeep. DARPA will supply the necessary government approval for compliance with DoD Instruction 8410.01 for

registration of a .mil domain. In case this proves infeasible, an alternative top-level domain will be utilized.

2. All servers, Internet connectivity, software, and services needed to develop, deploy, and operate the above-enumerated capabilities.

Proposers may choose to develop the necessary capability from a fresh code base, or develop the capability as a derivative of an existing product. In the latter case, proposers should be especially mindful of the intellectual property restrictions that may be imposed by the licenses associated with the existing code base.

The following maintenance, support and operations activities are desired:

1. Maintenance of the *vehicleforge.mil* code base, to include bug fixes, addition and modification of minor features, and other routine code maintenance.

2. On-demand technical support to users. The proposed level, means, medium, and capacity of technical support should be described.

3. User registration, including U.S. person status verification for access to export-controlled projects and information.

4. Implementation of credentialing and verification schemes relevant to open source development.

5. Collection of metrics and statistics on access patterns, user activity, and design (re)use.

6. Support for information assurance testing, certification, and accreditation activities.

7. Domain name hosting and upkeep. DARPA will supply the necessary government approval for compliance with DoD Instruction 8410.01.

8. All servers, Internet connectivity, software, and services needed to develop, deploy, and operate the above-enumerated capabilities. The proposer should plan for steady traffic loads consistent with several thousand concurrent users. Assumptions about traffic loads should be explicitly laid out in the proposal.

9. Public hosting and dissemination of electronic materials and announcements associated with a series of Adaptive Make Challenges. For planning purposes, the proposer should assume that a 200 Mb dataset will need to be hosted for public downloading associated with each challenge. Burst traffic loads of up to 100,000 downloads over 24 hours should be accommodated over brief periods.

Technical Area Two: Credentialing and Verification Schemes

DARPA is also interested in novel credentialing and verification schemes for ensuring trusted contributions to open source development projects such as those hosted by *vehicleforge.mil.* Our society employs a variety of mechanisms for ascertaining the level of trust placed in an individual; examples include personal knowledge of an individual, their biographical credentials, citizenship and immigration status, security clearances, etc. All of these mechanisms are only proxies for trust—and some are more effective and burdensome than others. This technical area seeks the development of novel approaches to credentialing contributors to crowd-sourced designs that are less limiting to global, grass-roots participation than current approaches, such as requiring all participants to be U.S. persons or hold a particular security clearance, while providing some degree of assurance against malicious contributions and/or re-dissemination of all or elements of the design. Verification approaches —wherein the benevolent nature of a contribution can be ascertained—may serve as a useful complement to credentialing and are also of interest.

The principal goal of this technical area is to develop a credentialing and/or verification scheme for *vehicleforge.mil* contributors that:

1. Provides some level of assurance against malicious contributions to crowd-sourced designs; and

2. Provides some level of assurance against re-dissemination of all or some elements of the design beyond the credentialed community.

While satisfaction of both objectives is ideal, approaches that address only one or the other are acceptable. Note that only revolutionary approaches that constitute a significant advancement over the current state of the art are of interest.

The following attributes constitute the key discriminators and measures of merit of any proposed scheme:

1. Resulting barrier to participation

   1. Level a. b. c.

   2. Level a. b. c.

of assurance against malicious contributions Probability of occurrence
Magnitude of malicious contribution Probability of detection

of assurance against re-dissemination Probability of occurrence
Scope of potential re-dissemination Probability of detection

4. Ease of implementation and operation

Offerors should describe their proposed approach in the context of these attributes. An appropriate series of experiments, proofs of principle, prototypes, and/or pilots should likewise be proposed such that a level of maturity appropriate for subsequent insertion into an operational crowd-sourcing infrastructure such as vehicleforge.mil can be attained over the course of the 12-month development effort.

Potential offerors are advised that awardees under this BAA may be, subject to further conflict of interest assessment, precluded from participating as competitors or team members in the planned Adaptive Make Challenges staged under the FANG program, or may be precluded from eligibility for pecuniary prize awards. As presently planned, these Adaptive Make Challenges will include a series of prize-based design challenges supporting the design of a next-generation infantry fighting vehicle for the U.S. Army.

Appendix B: Interview Questions

MIT Cadlab Research Interview
Ariadne Smith
May 2012

My research involves creating an online tool for crowdsourcing military vehicle development. The project, called VehicleForge, is part of DARPA's AVM (Advanced Vehicle Make) program, which aims to revolutionize the automotive design and manufacturing process by shortening development time and opening the design process to the community. The motivation behind VehicleForge is derived largely from successes of the open source software movement, and recent crowdsourcing projects, such as Kickstarter.

Users of VehicleForge will participate in every stage of the product development process, from initial concept generation to sketch models and feasibility analysis to design for manufacture. One of the key features of the site is the incorporation of DOME (Distributed Object-Based Modeling Environment), which allows users to gain access to complex simulation and modeling software that is run remotely on a server. This is step towards distributed product development, where users do not have to be in the same location or organization to contribute to the engineering development of a product.

VehicleForge also includes many project management tools for teams participating in projects. All files are managed through a version control system (git) so that every file is up-to-date and never overwritten. Projects within VehicleForge have wikis, task and bug trackers, and Google+ style "streams" to communicate seamlessly across the site. There is also a marketplace, where users are encouraged to share files and DOME simulations to build off of each other's work.

Many software projects have benefitted from open source, but it is only in the past few years that hardware manufacturers have begun to think about the benefits of integrating open design into their product development process. As part of my thesis, I am interested in seeing what design environments, outside of the military, might be interested in such a tool as VehicleForge.mil

1. Name: Doug
2. Role at organization:
3. Years at organization:
4. Please briefly describe the product(s) you currently work on and its development process: (ie how long it takes, different groups/expertise involved, etc)
5. Are users currently involved in the design of your company's products?
   a. If so, how?
   b. If not, is there interest in involving users (or generally, any engineers/designers external to the organization) in the product development process? What kind of user participation would be the most useful? (this can mean involvement at any stage of development, in any capacity)
6. What current features of the tool would be most useful in your company's product development process? Why?
   a. internal use as a project management tool
   b. internal use of DOME as a way to connect different engineering groups and stages of development
   c. external use to involve users/unaffiliated designers/engineers in product development
7. What are some other uses of this tool you can envision being useful for your organization?
8. Are there any additional features you would like to see in this tool? If so, what are they?