The Language of Creativity:
How Tech Workers Use Language During Innovation

Bruce Rinehart
School of Information, UC Berkeley
May 10, 2006

Introduction

This document is an ethnography of the creative aspects of the language used by network application developers during project development. Using methods and theory from cognitive linguistics for analysis, especially using the theories of conceptual metaphor and conceptual blending, this paper focuses on the ways that programmers and designers use and create language while innovating in the work setting. Initially, an informal interview process was conducted where tech workers were polled about their work practices and the way they used words, diagrams, and gestures during the development process to describe the items they were creating and the design decisions they made. Then a series of six interviews were videotaped for use both as examples in the analysis here and for inclusion in a sixty-minute video illustrating this topic.

The subjects of this ethnography were all involved in some form of network application development. For the most part, those who participated work on websites, although several are employed working on wireless and embedded applications. Tech workers in this sample (software engineers, user interface designers, and user interaction designers) were intentionally selected based on involvement in Internet application development. By looking at tech workers with different talents the objective was to better understand the language used in development from several different perspectives.

The process of innovation is an important element in technology development, yet there is no clear definition of how an organization or work group uses language as an integrated part of the overall creative act. This paper claims that tech developers are wildly creative when negotiating simple tasks. Indeed, when confronted with complex and intractable design problems, the imperative for creativity requires negotiation, both within the group and between groups at a level well beyond that of regular problem solving. Often such negotiation requires the use of conceptual blending in order to extend the conceptual framework of the problem set, which allows for understanding across domains of expertise. In short, just as in everyday life, when one needs to be understood, sometimes it is necessary to recast an idea in terms that others might better understand. Technical development is no different, indeed, such demands for recasting are greater the more heterogeneous the type of technology under development happens to be.

In order to make something new, programmers must name the thing they are working on to be able to talk about it during the development process. Often there are no words for the items under development because they are new. This is when the language of creativity emerges. While much of the language needed in technical development is domain specific (programmers use language from the domain of computer science, user interface designers use language from HCI), situations that are technologically heterogeneous require groups and individuals with different domain knowledge to work together in order to create an application that uses several different technologies.

An example of this is website development where server software may be integrated into an application that uses a database for storage, engages users in new ways in terms of interface, and allows interaction among users. In this example, there may be compiled code, one or more types of markup languages, interface controls that change based on

context, and user collaboration. In other words, many of today's web pages are derived from Java code using a database, pulling style information from CSS, including data from other 'includes' files, with AJAX interface elements, themselves a mixture of code types. The complexity of the various code bases used in website development requires a team effort since specialization occurs around certain domains of technology, each integrated into the web. In functional terms, there may be a team working on Java, a team doing interaction design, a team designing the interface, and a team deploying the project as a product for a company.

With the different types of expertise such as those above, there are boundaries between the groups which are often mediated by a few individuals who act as translators. As each group must work together with the other groups to create an integrated application, there is a communication demand that is not easily met. Perhaps in circumstances where the project under development is fairly conventional, developers can use conventional language for the purposes of communication. An example of this is 'shrinkwrap' software development that has reached a level of convention over the past couple of decades where the process was fairly well defined and the activities and language of interaction were defined by convention. This paper is not focused on conventional technical development, not because it lacks creativity, indeed most language is seen by cognitive linguists as being substantially creative. Rather, this paper deals with innovative technical development mainly because it is there, by virtue of the lack of convention (a definition of innovation) that emergent language and social interaction can most easily be seen.

This paper begins with a discussion of metaphor theory and conceptual blending that frames the investigation. Next, various types of creativity will be discussed in terms of how they are situated in the conceptual activity of the developers. Third, general findings are discussed followed by specific findings of note.

The video of this project will show segments from the interviews and should be considered a companion piece, complementing the information here by rendering an accessible artifact that hopefully will describe to the non-technical person some of the inner workings of technological innovation.

The Setting

Tech Developers

When software engineers work together on a project, they report that they create words and phrases to describe elements of their work activity and work product. This is not surprising, as most people will use nicknames for items in their workplace. Such a behavior is so commonplace that it is barely noticed at all. In technology it becomes more pronounced when the project under development is new. In this case, "new" means that there is no precedent set for the item under development, indeed, which then means sometimes there is no conventional process by which the project should be undertaken!

Language

The entire body of linguistics, cognitive science, and cognitive linguistics is important to this discussion, however given both the nominal training of the investigator and the scope of this document, only partial description of the linguistic data will be provided, and then only the items which are relevant to the discussion will be covered. Specifically, the key assumption with regard to language here is that linguists generally think of the act of speaking to be a creative act. The theories of Lakoff et al about metaphor and of Fauconnier and Turner about conceptual blending will be the primary tools of analysis. The decision to choose such a narrow aperture from a broad field is part a function of the intellectual curiosity of the investigator. Having taken classes from George Lakoff, I have learned of an emerging field of cognitive science, cognitive linguistics, which appears to account for much of the behavior I have been exposed to in the workplace over the last decade.

My Background

I have participated in network application development, mostly websites, for the past twelve years. My roles have included user interface design, database administration, scripting and light coding, as well as managing groups of developers and startups. Over this time I have witnessed some of the findings in this paper, however until I read *Metaphors We Live By* by George Lakoff I did not have a good framework for understanding what was going on. This study is the result of following my curiosity in an attempt to understand just how people get creative. Also, I would consider myself to be a creative person, a blues musician (harmonica), a creative technologist who participated in the emergence of the world wide web since it became commercial, and a photographer and videographer of minor accomplishment.

Creativity

There has been much coverage of creativity and innovation in the popular press as well as academic press. The assumptions about creativity necessary for this paper require not a broader definition of creativity, but instead, an allowance that creativity occurs more frequently than is generally assumed. When most people think of creativity they think of the lone artist or ‘mad scientist’ devising their creations alone. Certainly the stereotype of great writers is one of a lone creative mind crafting master works ‘out of nothing.’ While such creativity should not be excluded, it does represent an idealized version of a fairly common, and no less extraordinary, activity. Cognitive linguists contend that the very act of mundane conversation involves creative acts that are extraordinary. Those same theorists claim that most of our most important thinking is done unconsciously.

Cognitive Linguistics

I am not a linguist, nor have I studied cognitive science extensively. Nevertheless, having read on the subject and taken classes from George Lakoff I have been impressed by the way theories about metaphor and conceptual blending fit with the world as I experience it. More specifically, the fit with my observations of technical creativity has led me to pursue the subject of cognitive linguistics. Two areas of cognitive linguistics are of particular interest to me, conceptual metaphor and conceptual blending, because of the way they allow for analysis of the creative act (the work of the technologist). The former describes how meaning is derived from an embodied mind, that most of our daily

language is full of conceptual metaphor, which, although it is unconscious, is the basis for the way we speak and understand what others are saying. The latter describes how, through a process of recombination, we use concepts to understand other concepts that are not naturally related. This section will briefly outline the workings of these two theories in order to provide a theoretical basis to analyze various examples of language used by subjects reporting on their work activities.

Lakoff

Writing *Metaphors We Live By* in 1980, George Lakoff described how much of our common language is based on metaphor. Instead of finding meaning as an abstraction 'out in the world,' Lakoff claims that any meaning must be a function of processes in the brain and body of the person. While the implications of such a theory call into question the nature of much of the claims of western philosophy, that meaning is out in the world, Lakoff says such philosophy is an *a priori* endeavor that simply does not fit with the way language is actually learned and used in the world. The assertion and defense of this implication is beyond the scope of this paper, however it is important to acknowledge, both because it hints at a contested theory, and because it qualifies the theory as new and, possibly, as a paradigm shift in the sense of Thomas Kuhn [*The Structure of Scientific Revolutions*, 1962].

Conceptual metaphor usage is almost entirely unconscious. When we say, "prices rose," we mean that prices increased. We understand an increase as a movement upward based on the fundamental conceptual metaphor 'MORE IS Up'. In fact, prices never actually rise, we just talk about them that way. We conceptualize increase as movement upward due to the experience of seeing increase during our development as children. This is the experientialist nature of conceptual metaphor where we learn at such an early age that an increase is conceptualized as movement upward that we assume the meaning is in the words, not our brains. Lakoff offers a good deal of evidence supporting this claim and shows that learning conceptual metaphor is a function of embodied experience.

The fundamentals of conceptual metaphor involve an unconscious mental function that maps a semantic frame from a source domain to a target domain. In the case of MORE IS UP, 'up' is mapped from the domain of direction to the domain of quantity. Entailments of this mapping can be seen in statements such as "the temperature is rising," "she is an upstanding citizen," and "Garcia's guitar playing was the height of guitar mastery."

For the purposes of this paper conceptual metaphor is mentioned here because it provides an accessible framework for understanding the way we construct meaning in the world. In order to understand creativity, we must first understand, or make some assumptions about, how we negotiate meaning. Metaphor theory provides the substrate for understanding what we are doing when we are innovating. Our conceptual metaphoric framework is unconscious and unchangeable. We now turn to conceptual blending to understand how creativity emerges from our thought processes either in a group or individually.

Conceptual Blending

Gilles Fauconnier and Mark Turner collect much of the work on conceptual blending in their book, *The Way We Think* published in 2003. In this work they present a model of how concepts are blended. Similar to metaphor theory, an image schema is used to graphically depict the mental function of blending. Also similar to metaphor theory, Fauconnier and Turner claim that most thought is unconscious, and that blending, while an unconscious function of the brain, yields very conscious results.

An simple example of conceptual blending is the notion of 'trashcan basketball.' You and a friend could be in a room with a trashcan and decide to play trashcan basketball. Immediately, you both would understand the basic framework of what to do, to throw a piece of trash into the trashcan to score. This simple blend hardly appears profound in any sense, partly because of the common usage of blending in our everyday experience. However, upon closer inspection blending allows for a great deal of creativity. So how does it work?

In the trashcan basketball example we have two inputs, the current setting and that of basketball. These two conceptual spaces are blended together allowing us to understand the goal of the activity, to predict how to behave, and to prepare for innovation within the frame. At first glance it appears to be mere analogy. The trashcan is analogous to a basket on a basketball court. Yet, were analogy the only affect of this example we would not find room for creativity. Instead, by blending these spaces we end up with a blended conceptual space where certain activities are allowed which do not exist in either of the input frames. An example of this would be if we were to devise a way to 'dribble.' Of course, a balled up piece of paper cannot be bounced regularly enough to allow for dribbling like a basketball. However, we could devise a way to 'dribble' that involves bouncing the 'ball' up into the air, tapping to keep it aloft. Such an activity is not part of either input space, it only exists in the blended space. Moreover, the activity is only one of any number of possible activities which might be 'seen' as 'dribbling.'

By using these two basic cognitive linguistic concepts, a simple understanding can be obtained of how creativity and innovation takes place. Also, by describing how the creativity occurs, we can see various functions that creativity takes. More than merely allowing for a mental space in which improvisation can take place, it appears that the language of creativity can be used to situate ourselves in a social setting, understand social boundaries, and manipulate the conceptual frames of our work activities. Using examples from interviews with technical developers, the following section will address various ways that tech workers use the language of creativity to conduct their work.

Hand Waving Disclaimer

This is the Language of Creativity. There are many languages of creativity. There certainly are plenty of perspectives on the subject. I have chosen to portray a few notions about creativity by showing how people talk about how they make network digital media. We all are wildly creative, each in our own way.

Neuroscience and Conceptual Spaces

Although it is outside the scope of this paper, it is rather important to acknowledge that convincing evidence in neuroscience support frames, metaphor, and conceptual blending at the neural level.

Findings

This section describes general concepts and specific findings that were derived from the six hour-long interviews and twenty-one short informal interviews. The first part of this section covers general principles that were common to the experiences the participants reported. The second part deals with specific language used by the participants in describing their work with particular interest in the conceptual makeup of that language with respect to the artifact under development.

General Concepts

The general concepts discussed here are broken into four sections: background of the participants, organizational behavior of Internet application companies, common occurrences of metaphor and conceptual blending, and larger frames involved in technology.

Participants

The main participants of this study, also featured in the video, were selected from the social and professional network of the author with special interest in interviewing designers and engineers. Most were not formally trained in computer science or software engineering, however all are currently employed in technical positions at companies developing network applications. Participants were chosen who have a range in skills from designer to software engineer in hopes that a spread of talent area might best portray the demands of communication not only within the project group, but also between groups. The division between groups will be discussed in the next section.

As mentioned before, websites and network applications in general consist of a combination of different digital technologies that require individual expertise. For example, a moderately complex website requires user interface design expertise, software engineering expertise, database administration, and organizational sophistication capable of managing and delivering this complex media.

Interviewees:

Derek Powazek	Designer/Founder	Technorati/Fray.com
Beau Smith	Interface Engineer	Six Apart
Mary Hodder	Founder/SIMS grad	Dabble.com
Bill Gray	Senior Engineer	Glu.com
Mark Johnson	Senior Software Engineer	Internet Archive, Scribe
Pam Davis	Technical Producer	C|net

Technical Organization

All participants work at Internet related companies, although they are not all working on websites. Wireless games and book scanning running over IP networks are non-web companies, Glu.com and the Internet Archive Open Source Books project, respectively. Because network applications require heterogeneous skill sets of developers, the organizations that produce websites and other network media are divided into various groups and divisions that carry out skill specific tasks. A typical organization will have software engineering, user interface design, and web production (HTML, CSS, XML), although the barriers between these groups vary by degree. Because of the different groups and the varied domains of expertise, different understandings of how to carry out development initiatives is common.

All participants reported communication difficulty when attempting to solve problems across skill groups. Participants said that they routinely need to recast problems when negotiating with another group. A typical example is when a designer needs the technology to support a user-centered issue, such as creating a readable URL. On the one hand, engineers supporting the infrastructure of a website regard a URL as a programmatic issue rather than an interface issue. The designer, however, realizes that the URL is not only seen by a user, but relied upon for navigation and understanding of the site as a whole. One of the designers in this study reported this story in absolute terms, saying, “The user needs it, so you just have to deal.” In such a case, a conceptual blend is happening, where the designer relies upon the engineer’s understanding of the product as a whole, specifically that the user needs are paramount to the organization’s efforts. On one hand, there is an input space that consists of the organization’s purpose, to serve the needs of its users. In order to make his argument, the designer invokes the needs of the user with a third input space of the contested frame of the URL. By linking the necessity of a readable URL with the space containing the user, and then mapping that to the purpose of the organization, the designer creates a rich blended space in which the engineer can realize the value of making the change.

Most participants reported similar exchanges between design and engineering, although the nature of the conflict appears more intense upon inspection here. Since conceptual blending is mostly unconscious, these conflicts are routinely resolved without appreciable friction. Indeed, especially amusing to the participants were stories of when a person from the ‘business side’ would question or misunderstand various limits on technology. Although there were no good examples of this from the interviews, all participants referred to such incidents as being common. Such commonality may suggest that there is a cultural facet at play. It appeared that the larger the organization, the more apt it was to encounter conflict. This may suggest a partial reason for the successful innovation at smaller firms.

Metaphor and Conceptual Blending at Work

Metaphor and conceptual blending can be seen in three types of interaction as reported by participants. First, as in the previous section, problem solving is a prominent setting where metaphor and blending occur. Second, the artifact under development in a project usually is subject to a metaphorical mapping or conceptual blend that render it as a

material object capable of manipulation. Third, a translation role is playeds by metaphor and blending, where project goals and context, which themselves are often changing, need broad understanding for project success. These three types of interaction represent categories of how participants reported communication involving metaphor and blending.

While the example of the readable URL in the previous section should suffice in explaining the how metaphor and blending are used in conflict resolution, another, more humorous, example is offered here. While working on Xoom.com one participant reported that an engineer was arguing that 100 plus items returned from a database call should be presented on a web page in three columns, alphabetized horizontally. For example:

Alice	Ben	Frank
Hanna	Innez	Jack
Linda	Mary	Steve

The participant reported that an interface designer had been unable to sway the engineer from his conviction. The problem of finding alphabetized information when presented horizontally would have caused the user great difficulty, yet the engineer couldn't see why that was the case. The designer had tried to explain that the properties of browsing and eye scan were such that the table should be arranged vertically:

Alice	Hanna	Linda
Ben	Innez	Mary
Frank	Jack	Steve

Drawn into the argument, the participant, also an engineer, was unable to make a convincing argument and the engineer held his ground. Then, off handedly, the participant mentioned that the phonebook presented information this way, to which the engineer immediately conceded the point. In this case, the phonebook represented a metaphor for conventional alphabetized information presentation. Once it was blended with the issue at hand, the case was uncontestable (or maybe the engineer just gave up). Problems can be solved by mapping a metaphor to the problem space.

One of the most widely reported metaphor structures was that which rendered the digital artifact as a material object. Certainly this general concept is reflected in the notion of object-oriented programming where entailments and inferences are rooted in how an organization can best organize a code base for development. In this study, several participants used words like 'widget' and 'chunk' to describe elements of the code base they worked on. An interesting facet of this is the way these words were used and reused in different work settings over time to refer to different artifacts altogether. It appears that this blended space had become a prototypical structure for the participants.

The last type of metaphor and blending reported by participants involved the role of the translator. Most of the participants either described themselves as translators between various groups as part of their role in the organization or described a situation in which

they needed to translate concepts from one technical domain to the other. All acknowledged the difficulty and the necessity of such a function in technical development. Likewise, all described the translation process as one that fits into the conceptual blending paradigm.

One could speculate about other types of metaphor and conceptual blending usage, that there are more than three types. Indeed, as linguistic realities one could merely acknowledge the pervasive role that metaphor and blending play in most communication. The respondants offered these observations without specific prompting with regard to metaphor or blending. Rather, they saw it as a conventional part of communication in the technical setting. One might speculate that such a consciousness of communication is heightened in the technical environment given the conceptual demands of the abstract activity of digital media development.

Buzzwords and Frames

"Web 2.0" is the buzzword of the day. Replete with a metaphorical mapping from software, (that web 2.0 is an upgrade from web 1.0) this term, and those like it, constitute the popular framing of contemporary Internet technology. Interestingly, but not surprising, most of the respondents did not like the term as it was considered too vague, and therefore not very useful. However, all the respondents were involved in new development in Internet applications, what might otherwise be considered Web 2.0 technology. One criticism of the term was that it served mostly business needs. Several participants shrugged it off as 'marketing' or 'marketing speak.' In one sense, Web 2.0 represents a larger frame, one that might be considered a radial category that serves the social function of indicating the newness or freshness of the technology. Those that are involved in creating the technology do not need the popular term since the blended space that it represents has little semantic input for them. In other words, Web 2.0 does not have specific properties that enable emergent order, therefore as a blended concept it may indicate new technology, yet have little conceptual usage given its broad, non-specific usage. In the next section specific terms and phrases provided by participants are examined in terms of their usage and usefulness to the creative process.

Specific Findings

"Tuna Cans"

One participant noted that he had heard the phrase 'tuna cans' used to refer to databases by a business consultant. This term derives from the short, cylindrical shapes on whiteboard diagrams that often represent databases. A programmer remarked upon hearing this that when he thought of databases he thought of boxes as that was the shape of the symbol used in representing tables. Another participant recalled in the 1980s having a stencil produced by IBM that had a cylinder, which at the time was used to represent a hard disk. Indeed, at the time, hard disks were so large, that the profile of the platters was visible in the shape of the device. Here the metaphorical structure and conceptual blends are too numerous to describe completely. Yet the notion that different symbols are used to refer to the same thing, or more generally, that symbol usage varies so greatly is evidence of a dynamic conceptual space emerging from the technical realm.

Whiteboard Diagrams and Social Time

Several participants mentioned the use of whiteboard diagrams during meetings. The metaphorical nature of such diagrams is obvious in that they are a metaphor for the artifact under development. One participant talked about how whiteboard diagrams were conceptually nested, where greater levels of detail were contained in the shapes at various levels of granularity. These diagrams make up a large chunk of the metaphorical structure of the artifact. What's more, two participants noted that diagrams were used to indicate decisions made in previous meetings. This observation suggests that the usage of whiteboard diagrams in this case was more than acting as a blended metaphor for the artifact under development. Rather, the diagram was used to indicate a temporal notion of the social process of development itself. Here we can see a blended space which facilitates a social understanding of the history of development.

"Chunks," "Widgets," and "Headers and Footers"

"Chunks" and "Widgets" were mentioned before as metaphors that allowed the digital artifacts to take on physical qualities of objects, which then allowed for better communication. The aspect of reuse, or that they had become prototypical metaphors that were used in different settings to refer to entirely different entities is the subject of this discussion. One participant mentioned "headers and footers' in the same sense, where a web page might have a 'header and a footer', but that a component of a web page might have a 'header and a footer.' This observation by several of the participants might be seen as evidence supporting a theory of language creation not only specific to a single situation, but a generalized language useful in a type of conceptual framing of the digital artifact. This is perhaps too speculative, yet noteworthy.

"Extreme Usability"

Extreme Programming is a new method of software development where two programmers work together in real time. A flavor of peer programming, this method is touted to produce better code by allowing the corrective and collaborative affect of two people looking at the same task. One of the participants coined the phrase 'extreme usability' to indicate a similar collaboration between an engineer and a usability expert. In her description, she noted that a typical discussion would involve an exchange where the usability expert would describe a typical user need or expectation and the engineer would respond with solution possibilities. While some of the motivation of the interaction was admittedly to make sure certain interface issues were addressed, the method by which solutions were agreed upon involved a complex blend of the user interactions with certain features in the code under discussion. The participant reported that this method provided great advantages to interaction innovation because more solutions could be explored. The central phenomenon here is that the blended space that included inputs from the usability expert, inputs from the programmer, and inputs from the user space allowed access to the artifact under development. Without any one of these elements, the solution space would not have as rich a set of properties.

"Pipelines," "Derivers," and "Overload"

Perhaps the most interesting notions were reported by an engineer who described two common words at his organization, 'pipeline' and 'deriver.' Certainly metaphors, these

two words also had the prototypical properties as mentioned in the discussion above regarding ‘chunks,’ ‘widgets,’ and ‘headers and footers.’ However, the engineer referred to them as ‘overloaded’ terms. ‘Overloaded’ is a term from Java which refers to two functions in the same method which have the same name but can have different types of arguments. Here, the engineer is blending Java with his spoken language showing a reversal in the direction of the metaphorical mapping. This is common in technology where phrases are lifted from a technology to describe non-technical things. Common phrases like this might include, “I don’t have the *bandwidth* to deal with this,” or “just Google my name to find me.” Here we can see the creation of language has come full circle, having first been invented to enable innovation, it is then repurposed to indicate the scalar nature of prototypical terms as they apply to different levels of granularity in the development and understanding of digital artifacts.

These are only the highlights of interviews with a relatively small group of engineers, designers, and producers involved in the daily work innovation. The language of creativity occurs in all walks of life, however in technology it is more pronounced due to the high degree of creativity required in the process of innovation.

Notes on the Video

A video of various clips of interviews with the primary participants in this study is available as a companion to this paper. The hope is that the video will allow the participants to tell their own story. Of course, these are edited clips, therefore any claim to a true portrayal of the participants must be constrained by the understanding that the author has a point of view which is inescapable. But such a video does provide an opportunity for the reader to create conceptual blends for themselves, allowing an emergent order not unlike that of the language of creativity.

Conclusion

The language of creativity is the emergent language that programmers and designers create during technological innovation. This ethnography is a collection of various linguistic traits that are analyzed based on new theories in cognitive linguistics. The intent of this paper has been to outline areas for further research. Also, some of the theoretical density has been thinned in order to make the material more widely accessible. For example, each of the items in the ‘Specific Findings’ section above would require more than ten pages each in order to completely explain the full conceptual structure.