Toward an understanding of user innovation in communication software.

Wendy Elizabeth Mackay
26 November 1989

outline

1. Introduction
    a. Why study innovation?

    b. Why study it in the context of electronic communication?

2. Definitions of innovation:
    a. Invention vs. innovation

    b. Kinds of innovation I'm interested in here.

3. Examples

4. Lens

5. Case Studies

6. Experiments

7. Conclusions

Introduction

Why study innovation?

Interested in how individuals in an organization change their patterns of behavior and produce new techniques for accomplishing work, becoming more productive. Innovation is defined as successful application of new ideas to solve problems -- small and large. Small

Why study innovation in the context of electronic communication?

Fluidity represented by software in organizations (JSB). Argue that because organizations are so dynamic, people within them must constantly their behavior patterns in large and small ways. They also modify their rules for handling their communications with others -- this is affected by reorganizations, changes in task, promotions, relocation of offices, level of interest in different areas, etc. Because it is so dynamic, the need (or potential) is there (?) for constant innovation.

However, people do not always innovate -- why not? what are the barriers to that inovation? Suspect that major changes result in entrenched ideas and clinging to the "old way of doing things", whereas smaller changes (what are they?) are less threatening and have the potential for new ideas. Also, the strategy of placing applications in contact with theoretically based development efforts forces people to come into contact with unexamined assumptions and to look at things from a new perspective (OZ example with Mike Powers and Connie Parkes), Medley--> Sun experiences at PARC.

What are the circumstances under which people innovate and what are those in which they don't. (Should imply that one can artifically increase the probability of innovations if one understands that... keep in mind that one is measuring probabilities and the techniques required to do that.)

Need to have things in which we can study the artifacts -- and see the results of the innovations. (Examples -- lens allows us to see the rules that people write, which indicates a strategy for managing electronic communication. But, it doesn't show major innovations -- have to get things like the use of rules post reading messages through interviews -- because the predefinded system is not set up to record those kinds of changes.)

Other examples of studying the artifacts -- where do we have transcripts and records that show changes in patterns of behavior?

Definitions of innovation:

Invention vs innovation

First, explain the definition of invention and how it is different from innovation. (Innovation includes the successful application of a new idea.) Distinguish between process and product innovation -- am interested in both kinds here. Software innovations are particularly susceptible to process innovations -- or at least, the use of software changes the patterns of work, the ways in which tasks are approached, and the kinds of tasks attempted (more than anecdotal support?), which means that the introduction of the technology may produce process innovations.

Kinds of innovation contrasted

Within the system vs. outside impact

Some software is designed to be "flexible" or user modifiable (emacs, speadsheets, hypercard). Innovations may fall within the bounds expected, indicated by the system or they may go outside those bounds. Examples: using a spreadsheet to handle columns of numbers -- not as a budget but for something else that requires this, is an expected kind of innovation. It's a situation in which an application exists that fits the model for the software and is the kind of thing the software designers hoped for. A different kind of example is the use of a spreadsheet for a circuit diagram -- the squares are all used as either 1 or 0 and the pattern of the squares matches the grid of a circuit -- can calculate the shortest path (or whatever they did...find the source of this example). This is in some sense a much more restricted use of the calculation power of the spreadsheet and a different view of the area of the spreadsheet -- that of a 2 dimensional space rather than rows and columns. (At some point, it's like Riemann sums and calculus: taking advantage of the fact that one can break up a continuous area into discrete segments and calulate the area.)

Repair of an error vs. a workaround that creates something new.

Explore the difference between innovation and error-repair -- one may be just the positive side of the other. Two sides of the same coin. In one case something is wrong that needs to be fixed. In another, a need is perceived and achieved through an innovation -- without there necessarily being a prior perception that something was wrong.

Look at repair theory here and why it's relevant -- the idea that people have principled errors and the idea that something isn't perceived to be a problem if there are clear workarounds. Xerox copier examples.

Is there an equivalent "innovation theory" here? The idea being that people have principled models for how they accomplish things and how the technology they use works. If they come up with deeper insights about those models, or make certain aspects of them explicit, then apply something in a new way, repartition the task at hand resulting in an improvement, or whatever, then it's just a new principle being applied. Also, perhaps people don't perceive something as an innovation if they perceive a need and apply something else -- one person's innovation may be another person's regular work. (Is organizational learning simply the sharing of "obvious" innovations from one person to another -- the creation of a group model of the interaction of the technology and the behavior the the people in the group?)

Within dimensions designed by user vs. not

Seems to be confused with i. above. What's the difference here? Lens rules -- vs changing the ways in which rules are applied. The former is a within-dimension innovation and the latter is external. In i, the distinction is based on the result or impact of the innovation. This one is a distinction based on the intensions of the designer -- or at least the things that the design makes easy. (Colab examples -- what are the things the designer designed to make easy -- the abstractions that were explicit -- and the things that are made easy by accident, unanticipated by the designer. Then look at the things that are made hard by the design -- perhaps explicitly by the designer as a conscious tradeoff for some other important feature or design, and the things that are unintentionally harder, as in the colab where things taken for granted in working on a white board (everyone being aware of the same referrent) are suddenly difficult -- the latter is sometimes hard to detect.)

Not that things are not "absolutely" hard or easy. "Easy" is certainly affected by physiological things (e.g. your fingers have more control than your neck, so a pointing device that you move with your hand is "easier" than one controlled by head movements), but it is also very much affected by previous knowledge. Knowledge of one text editor provides interference effects when trying to learn a new one.

Dynamic model

Software task analysis -- design model is too static. Software changes users, who change software. DEsigners react, over time, to user changes and change the software. Sometimes through minor changes (e.g. those that do not affect the overall functioning of the system or the system model) and other times through changes in underlying structure (in which the changes are not independent).

Choice of model / abstraction affects the kinds of innovation

Examples

Tie back to definitions in 2.

Perhaps these are placed together.

Spreadsheet -- used for circuit diagrams. Xerox copiers -- kinds of repairs people use. (Some are unknown innovations-- may take someone else to make it successful by appropriate application.

CHI kiosk example -- people pinned the output of the system to the message board -- good way to quickly identify who the message is for. Copied very quickly.

Lens data

Need support for these intuitions, more than anecdotes and examples Also, need to make more concrete conclusions -- not questions, but statements that can be verified and tested.

1. User perception is not equal to reality (user’s don’t really know what their "real behavior is")
2. Rules are based on perceptions, not reality.
3. Presenting users with data about their own behavior changes it.
4. Innovations occur (errors too).

Hypothesis: frustration/chaos/innovation -- may predict times when innovation is more likely, but not what kind of innovation. (Unless you already knew a great deal about the task, the clashes in models, etc.)

Effect of application coming into contact with a new technology -- system model clash vs user model clashes.

LEarning to learn vs learning to not learn.

Case studies

Look indepth at a few examples of innovation. Can we get a feel for what causes/correlates/precedes innovations? What encourages, discourages?

John Tang and IDE and Notecards, Evelyn and Muse and French

Micro experiments

Ho Test: Providing feedback to users about their own behavior will change work process and increase p[innovation].

Set up a situation with a combinations of a within and between subjects design in a field study. Look for natural environmental changes (trips, reorgs, earthquakes) and also induced changes (feedback on own behavior, interviewer's presentation of a new idea, ideas from others).

Need operational definitions of baseline behavior and innovative behavior.

Need to figure out how to provide feedback.

Comparisons: email, discuss, file handling, on-line docs, on-line library, bug report handling, programming,

Kinds of changes: task partitioning, order of work, priority, number of items handled.

Learning that inhibits innovation -- functional fixedness. Ref to inhibition.

Or -- can we find innovation in the repair theory data?

(JOke -- child never talks until he's 5 years old. Then suddently speaks a complete sentence, asking for something. Parents surprised, say "You can talk" -- explanation -- sure. Didn't talk before because nothing was wrong.)

```
OK  OK  OK  error  OK  OK  OK  OK  error  error      OK  OK
         |          |               |       |         |
         --repair----               |repair-|         |
                                            --repair --
```

Alternatively:

```
              --innovate--        --innovate--
              |          |        |          |
OK   OK  OK  mismatch   OK   OK  ???        OK
```

Stu Card's point --

What happens when faced with an entirely new system -- can result in a clash of user models that is more productive because people can do something new or in a different way or have an insight into a proplbem. BUT, users often move into a core "survival" mode, in which users only try to do a small subset of things, and every time they try to do something that was easy with the old system, they are thwarted. Unaware of the new user model (which is rarely taught and is generally not accessible to people who know the new system, because they are unaware of their own assumptions about the system and how it frames their thinking), they do not take advantage of the system and do a very limited subset of their former work. Over time, as they become used to the new system, they graduallyadopt the assumptions of the new system, and are able to slowly branch out. BUT if they continue to switch among systems, there is a serious interaction effect that prevents "automaticity" (cite Stu's reference here) -- continues to prevent learning.

The tool influences the kinds of thing you try to do... Stu's example of a problem in which he wanted to create a set of columns of different widths in a text editor -- and couldn't. Ended up not doing the project.

Talked to Ramana -- doesn't use LEns any more, uses rmail, which has some of the same features.

Stu -- earthquake didn't change his office organization. Came back and was immediately too busy to reorganize -- also, hasn't done it since he moved his office. "Anything with a time period of more than 2 hours to learn to do just won't get done" -- referring to setting up a new easier email system. If the set up time (transition cost) is too high, then it's a problem. Anti- innovation, in some sense.

What is the right combination of factors that results in an innovation? A change, but not too dramatic a change. A need that overrides the short-term costs of doing something new.

Distinguish between incremental innovation and radical. Internal changes seem like the former, external seem like the latter.

Changes in how people view the task they do can have major effects beyond the specific technolgy -- may change how they work, and how they use other technology.

How can we predict when innovation will occur?

Predictors of when, not necessarily what. Also, may only predict an increase in probability.

1. Changes in the environment:
2.
    a. getting a new project
    b. social -- ideas from friends
    c. ideas from interview
    d. access to new model (e.g. can apply to rules)
    e. observing own behavior
3. Conflict between user and system metaphor

4. Change in process by user (causes packaged behaviors to change. Lucy example -- no longer have to do things that slow processing because new technique partitions task differently.)
5. Can you introduce systematic changes and watch effects?
6. Control or use the Hawthorne effect?

Table of Contents