Designing for adoption

extract of draft for new edition of HCI book
Alan Dix, www.hcibook.com/alan

In 1999 a number of companies announced smell interfaces [[**xref chapter**]], devices that used a number of tiny inkjet like nozzles to spray scents based on computer control. By varying which scents were sprayed different overall odours could be produced. Because this was the year of the dot.com explosion the business plans of these companies centred around the internet. The idea was that web pages would include tags that would describe required scents and then if you had one of the scent producing devices you would experience the relevant odour when you visited the page.

The companies did not thrive. This was partly because the financial markets lost their appetite for Hi-Tech in 200 with the dot.com crash, but this is not the only reason. These products with this deployment plan could *never* have succeeded. Imagine you were the first person to buy such a device, why would you do so? Only if there were lots of websites already with smell mark-up. Imagine you are a web developer, would you add mark-up? Only if people already had the devices. There was *no trajectory* from no use at all to widespread use.

A compartmentalised view of design sees the designers and technologists creating a product and then simply handing it over to sales and marketing who try to get people to use it, or in a less commercial environment, you make some software and simply put it on a web site for people to download.

In chapter [[**chapref**]] we discussed personal value. Each time we make the decision whether to use or not use a product (buy it, or simply pick it up), there is a weighing up of value: is the product worth using *now*. The complexity of the smell devices is that the decision is affected by other people's previous decisions. The value of the smell device to me depends on whether web site developers have already adopted the technology. The economists call these *network effects* [[**ref**]]. If you use a particular word processor, then it is more valuable for your work colleagues to use the same word processor so they can share files with you. The economists are interested in this as it changes the dynamics of free markets and creates effective monopolies when levels of usage are high (as is the case with Microsoft Office). Here we are thinking about the opposite end, how to get started at all.

[[ref to book website for network effects /e4/online/network-effects/]]

Single products and critical mass

In Chapter [[**ref**]] we discussed the idea of critical mass in groupware and CSCW systems. As with the smell device, the early adopters of a communication product, like the telephone, get little or no value but still have to expend effort learning how to use the product and cost buying it. As more people use the product the value for each increases until there comes a point where the value exceeds the cost. If the number of users is below this critical mass then they are likely not to

adopt the product at all, or to stop using it, only once the number exceeds the critical mass can it 'take off' and become self sustaining.

[[** note should add Grudin ref [159] to that chapter **]]

[[[**repeat figure 13.1 from page 455**]]

While Grudin's original analysis of critical mass [[**ref**]] was focused on explicit collaborative technologies, it is equally true of any product where there are network effects (like the word processor).

Once we understand this problem, we can begin to consider design solutions that address it.

First we may try to increase the *zero-point value*, the value to the very first user of the system when no one else is using it. For example, we may add a whole set of features that are not part of the critical use of the product, but give it initial value. For example, web photo albums are useful just to put up your own images, but when lots of people use a site such as Flickr it also enables community interactions.

We might try to release an initial deployment that exceeds the critical mass. This could be a company-wide roll-out or incorporating it into a release of an existing mass product or operating system. This is of course only possible if you already have sufficient market power. However, the size of group can sometimes be reduced if we can identify a *clique*, a small group of users who have a larger level of co-value. For example, early domestic telephones were owned by a rich elite, who, of course, talked mainly to each other. This clique does not need to be the rich it may be those who simply like new technology, an interest group, a small department in a company, or people in a small geographic area.

Within a clique the cost-benefit curve changes, you effectively make the curve steeper for small numbers of users and so reduce the critical mass. In order for the growth to continue however, the identified group must have 'soft edges', people who are at the edge of the group to whom usage can gradually extend. In the case of the telephone those a little less rich!

In the case of the small devices they could have been targeted at a vertical market, such as travel agents, although they are relatively hard-edged group so would served to maintain the product's viability, but not create widespread use.

[[* diagram of effects on cost/benefit graphs **]]

Linked products and lattice of value

In the case of the smell devices we effectively have two products (or aspects of a single product) that are linked: the device itself and the web page mark-up. Instead of a single homogeneous target group who we are trying to get to use a product, we instead have a heterogeneous target groups, in this case the web site publishers and people surfing the internet. The situation is similar to the simple critical mass

problem, but here the web developers need a critical number of surfers with the device and the surfers need a critical number of sites with the mark-up.

This is a common situation for web-related products (although not limited to them, think of cars and roads) where we often have two products, product A for web developers (e.g. the smell mark-up) and product B for web users (e.g. the smell device). We can depict the situation using the *lattice of value* (figure [[**LV**]]). The initial situation (a) is where no-one has our web products at all. We would like both web developers and web users to widely use our products (d). In order to get there one of two things must happen:

Adoption route 1.

transition (i): web developers choose to use product A, leading to situation (b)

transition (iii): web users choose to use product B because of the web pages already using product A

Adoption route 2.

transition (ii): web users choose to use product B, leading to situation (c)

transition (iv): web developers choose to use product A because of the web users already using product B

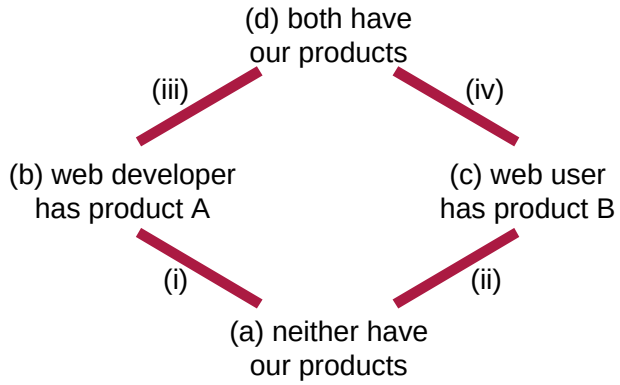

Figure [[**LV**]]. The Lattice of Value for web products

Both routes involve a zero-point value either (i) for the developer who must start to use product A with no current users of product B, or (ii) for the user who must use product B without sites using product A. Without at least one of these zero-point values the product cannot succeed. This was precisely the case for the smell interfaces.

It is interesting to see how a success, Adobe PDF, overcame this. It has two connected products the Acrobat development suite (product A) and the Acrobat reader (product B). Adobe only charge for the development environment, the reader (product B) is free, but of course there is s till a 'cost' in terms of downloading and installing the software. There is no reason why a user would choose to do this initially, but web developers could include free download links on their sites. This meant that even if *none* of their visitors had acrobat reader when they first visited the site, it was still valuable for them to use the development suite and put the PDF

files online (transition (i)) – that is there was zero-point value for them. Once this happened users would download the reader (transition (iii)) to read the PDF and hence the product spread.
Transition(ii) is a choice by the web user so has to be of value to the user. However if this is also of value to the web developer then the developer effectively wants users to have product B … they become a promoter of your products! If in addition (iv) is of value to the user (they want sites to use product A), this produces a market pressure on the site owners. The two together make a strong *feedback loop* promoting the products.

To see the importance of this let's return to PDF. Today web technology is a lot more stable than when PDF was first released. So if it were being released today an alternative zero point strategy for (i) would have been to have a PDF to HTML converter so you could use the development tools but produce pure HTML version, or perhaps a Java applet viewer. However, then there would have been no advantage to the developer or user in the user having Acrobat Reader slowing the uptake of the technology. Not all 'fixes' are equally good.

[[** link web case study – using LoV to guide the design of a site map tool **]]

In the case of the smell device we might imagine addressing transition (i) by having a piece of code running on the user's machine that scans the web pages and compares with a large dictionary of terms with associated smells automatically creating appropriate smells for pages. Note this is an example of a general strategy of 'good enough then even better' design – the smell device is good enough on pages without mark-up to justify using it (transition (ii)), but even better when pages have mark-up, so that smell device users would preferentially visit marled-up sites and thus create market pressure.

Note that the design of the second-edition HCI Book search described in Chapter [[**ref**]] can also be analysed in terms of the lattice of value. Here there is a single user/reader, but two 'products' the book itself and the web site. The search was designed to have zero-point value (useful without the book – transition (i)), but to have added value of you also own the book (transition (iii)). Also it was designed that if you already had the book (situation (c)), the web search was of additional value (transition (iv)).

Understanding connections and market ecology

The lattice of value assumes you already have the broad products in place and is about understanding how to modify them to increase their likelihood of success. A precursor to this is to understand the interacting groups of people related to your design area and to see how to use that knowledge to create products that can bridge that chasm from no use to use.

While for any area there are interactions between people (friends, family, student–teacher, developer–user), for internet products there is a particularly close connection with design as people may email one another about things that work or don't work (word of mouth recommendations) and where web or desktop products can link to email or other communication technologies to make this recommendation

easier. Furthermore, we may deliberately design complementary products using the lattice of value of we understand who the groups are.

When mapping out the different groups of users and their connections there are two particularly important things to look for:

Fan–out – Where one (comparatively small) group affects a larger group, for example teachers and students. Targeting the smaller influential group is an obvious route to the larger group. This is used in traditional marketing, for example, the textbook publishers give complimentary copies to teachers in the hope they will use them in courses and recommend them to students! This can also lead to opportunities for complementary (N.B. different spelling!) products; for example, this book has associated learning materials that teachers can use in courses. These can be used on their own, but are of course most advantageous when students have the book.

Feedback – In the case of web developers and users we noted that if both routes through the lattice of value give value then web developers will encourage more uptake of web users and vice versa. That is we have a feedback loop. Feedback loops may include several groups (and may be indirect group X influences group Y influences group X influences group X). feedback can also be within a group as would be the case of early telephone owners who encouraged friends to get one. While fan-out merely multiplies your initial effort, feedback leads to exponential growth – the *viral growth* beloved of marketers.

Of course if the technology being introduced is itself a communicative one, for example, in the early days of instant messenger, then it may create new connections, relationships and feedback loops!

Where communication channels already exist, we might simply make it easy to use them in connection with our new technology product, for example, “mail to a friend” links on web sites. Alternatively we may design this deeper into our products, deliberately creating community value, things that increase in value as friends or colleagues also use it. That is we may deliberately turn standalone products in community or CSCW products. While we initially started looking at critical mass as a problem for which increasing zero point value was a solution strategy, here we turn this on its head. Given a product that already ahs a zero point value, we add community features to increase that value with other users and hence promote growth. The community features (wish lists, bookshelves etc.) added to eCommerce sites are an example of this.

As well as the interacting groups of users, the users also use existing devices, technologies and products. We can often leverage these to increase zero-point value. Imagine you are designing a new WiFi communicator. It is a small phone-sized device that allows groups of friends to video-chat whenever they are close to a WiFi hotspot – live video chat virtually for free. By now you should already be asking “what about the first user?” – in order to make it valuable to that user we would simply add bridges to exiting web-based video chat and mobile-phone based video services. You get the convenience to be able to chat cheaply on the move (zero-point value), but it will be cheaper still for you and your ‘ordinary’ phone using friends if they also get the new device (community value).

Design guidelines for adoption

From the above we can extract a few guidelines for managing this difficult adoption path from no use to widespread use.

increase zero point value – Make sure the first user of the product can get value from it even if no-one else uses it.

identify cliques – Find small groups who can be early adopters getting co-value from the product, but who are 'soft-edged' enough to be a seed for growth.

understand relationships – Understand the different groups of users surrounding your design area, not just the obvious direct users, and how they influence one another. Identify where there may be opportunities for complementary products.

create community value – Deliberately design features that give extra value for a user of your product if other people (from the same group or a different group) also use the product (or a complementary product). The initial users will then be natural advocates for your products.

leverage existing products – Enhance zero-point value by linking to existing products, importing and exporting common data, etc. Where possible exploit the 'good enough then even better' principle so that there is added community value when other people use your new product too!

Note that these guidelines do not create good designs for you, or even suggest which particular features to add or change, but they do guide as to where it is worth investing your limited design energy.

Reading

Andy Cockburn's thesis