USID2009

Design principles to accommodate expert & novice behavior in enterprise applications

Raman Khanna
Design For Use,
Delhi, India.
rkhanna@designforuse.net

Abstract

The extent of user expertise in using software applications has long been a subject of discussion for User Interface Design professionals. This concern is more so in the case of enterprise applications where varied kinds of users have to be accommodated while designing the interface. There are people who are experts within the specific enterprise, adept with its procedures and practices and then there people who are starting afresh with the organization. Finding the right balance between the expert and novice behavior thus becomes critical in the design process.

There has been ample research on both personas but it has stopped short of providing ready to use guidelines for the UI practitioners. Especially, in the enterprise domain. This paper aims to learn from this research, assess the behavioral differences, evolve guidelines with illustrations and propose a further direction in research.

1. Introduction

Existing research and available literature has documented differences between the expert and novice behavior in the context of consumer software applications. There are observations and set of rules based on cognitive models by using audio and video recordings. [7] Others have put forth high level guidelines for expert and novice users for ease of achieving their goals. Some visual perception principles have also been derived on how the extent of granularity on the screen affects the novice and expert perceptive behavior [5]. Posts and comments on blogs by HCI practitioners have been trying to explore this area for some time now.

There is also literature on learned behavior and transition from a novice to an expert in terms of motivation to learn and openness to experience. An Expert-Novice Ratio method has also been developed to assess the impact of such factors on software products [6]. Effects of both intuition and experience on performance have also been documented in great detail.

All research primarily concludes that in order to design a good human computer interaction the designer has to choose the type of interface and interaction style to fit with the class of aimed users. And that this understanding becomes more important when a number of behaviors have to be accommodated in a single interface.

It would also be essential to note that this paper builds upon this existing research, combines it with the examples from the enterprise domain to distil some useful guidelines that can be used in design practice. In no way does the paper try to evolve its own analysis on behavioral patterns or contend the research done so far.

2. Enterprise Applications

Enterprise software, also known as enterprise application software (EAS), is software intended to solve an enterprise level problem in areas like supply chain, human resource, accounts management etc. As business enterprises have similar departments and systems in common, enterprise software is often available as a suite of programs that have attached enterprise development tools to customize the programs to the specific enterprise [2]. Each enterprise or organization has a certain way of functioning and a defined set of processes. A software application is aimed generally at improving its productivity and efficiency. Thus, expertise at

Page Number

these processes and knowledge of the domain is very important for the workforce of any organization, as the applications are usually an augmentation of these factors.

Any organization at a given time will have people generally with different levels of such expertise:

- High level of domain expertise but low level of software proficiency.
- Low level of domain expertise but high level of software proficiency.
- High level of domain and software expertise.
- Low level of domain and software expertise.

Software expertise here pertains only to the use of the specific enterprise application in use by the organization.
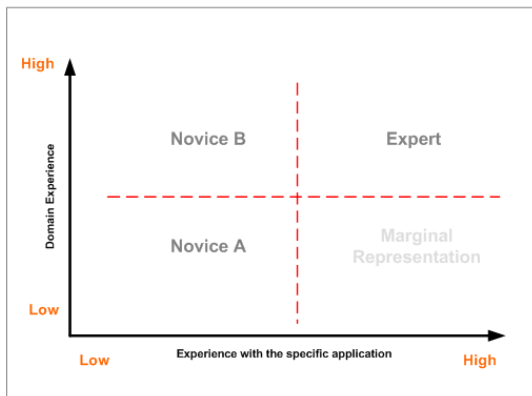

Figure 1: Comparison of Domain Knowledge in Expert and Novice Users of a system.

As we progress, we will limit ourselves to two generalizations of the above mentioned persona categories:

- Experts: Workers with high software and domain proficiency
- Novices: Workers with low software proficiency but with varying amount of domain expertise. (Figure 1, Novice Type A & B)

3. Expert Novice Behavior

Research demonstrates that there are significant differences in both the behaviors in terms of the possession of domain-specific knowledge that comes from experience and expertise in the field. It also shows the difference evidence for each category [7]. They also differ by a level of

experience in their own domain, and possession of more episodic knowledge [7].

3.1. Expert Behavior

In terms of use of an enterprise application, experts do things differently as they know more. They are able to take large amounts of information and see them as connected units or chunks. They are able to characterize new problems and sometimes possess a refined set of high level concepts about the specific enterprise domain.

When using an interface, expert users can quickly form goals and sequences of actions to achieve the goal. They would like the number of interactions to be reduced and thus the action execution is accelerated. They require that the pace of interaction is quick. In a word, they want a highly efficient interface [8].

3.2. Novice Behavior

Novices progress by trial and error and mostly possess general domain specific knowledge. While using an interface they require easy learnability. They would like the workflows to be guessable. Novice uses work on assumptions, learning, use of instruction manuals and their pattern perception is smaller then experts. There is also a significant difference between the memory recall in both the personas.

Besides this novice users have the motivation to acquire the skills and knowledge about the system to get to a level of better expertise. There is also a segregation in types of beginners when it comes to such an application.

First, there are personas with low knowledge of enterprise domain, they are complete beginners (Type A, Figure 1). Then there are people who have a high or longstanding domain experience but are completely naïve when it comes to working with a specific software application (Type B, Figure 1). Thus, one has to account for both these types while designing.

3.3. Accommodating for both in a single interface

Creating usable applications is not easy and most enterprise applications need to be designed to accommodate both novices and expert users. Beginners or novices generally gain skill over time and become expert users.

Enterprise applications are complex and have modules and features. At any given time users make extensive use of a small subset. So in a way, there is a novice in every expert user of such a system.
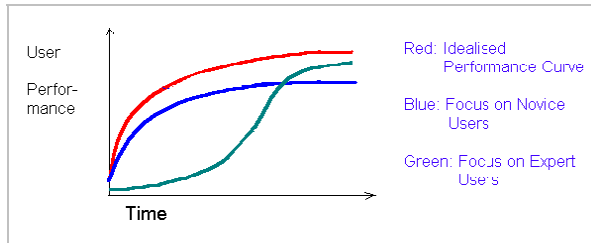

Figure 2: User Performance Curves against the dimension of time [8].

"Ease of learning refers to the novice user's experience on the initial part of the performance curve. Highly learnable systems have a steep incline for the first part of the learning curve and allow users to reach a reasonable level of usage proficiency within a short time. Our goal is to design the interface so that the performance curve rides the best parts of the blue curve and green curve and reach the idealized curve. To achieve the goal, we need to combine the efforts to support the novice users with the effort to support expert users in one interface."[8]

4. Guidelines

Keeping the above research in mind some guidelines can be evolved for accommodating both behaviors in an enterprise application to serve as tools for UI practitioners.

4.1. Apply appropriate personas at each phase of the design process

User profiles are a representation of the actual users of the application and their behavioral characteristics that can be used as a checklist to validate the design recommendations at various stages. A good design methodology is always persona driven. It is always a good practice to create distinct personas for beginners and experts before starting out.

In the design phase it is recommended to initially focus only on the novice/beginner behavior. Consequently, the preliminary usability test ought to have a larger representation of novice users as compared to that of experts. For example in the case of an enterprise application, one could go with 6

novices with low domain knowledge, 6 with high domain knowledge, and only 3 expert users (who are proficient both with the application and the domain itself).

Having done this, it makes sense to start accommodating for experts in the detailed design. Add some accelerators, shortcuts, custom controls (discussed later). Followed by a summative usability testing, this time with an almost equal representation of both the user segments.
In addition to this, it is also a good notion to have some rough idea or an expert-novice notion while doing a heuristic evaluation exercise during the analysis phase.
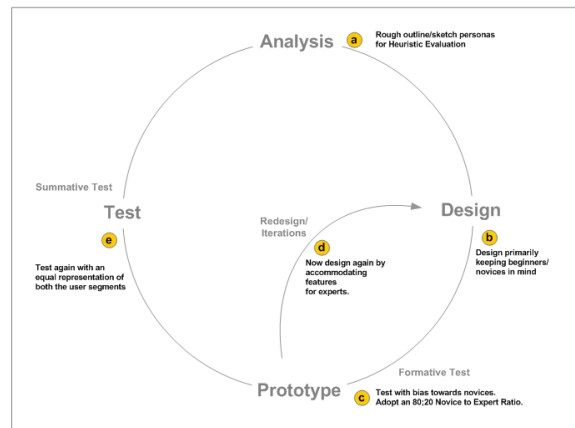

Figure 3: Recommended flowchart of persona integration and test samples during the design process.

4.2. Use affordances to leverage prior experience

Knowledge or existing experience with other applications can be used differently for novices and experts.

4.2.1. For Novices

Leverage the experience with other previously used consumer applications and carry it forward to the new enterprise application or module. For example, use the same shortcuts or tab interactions in the MS Office or similar suite for similar tasks like copy/cut/paste or show/hide in the enterprise application.

Another idea is to guide a person with high domain knowledge but lesser experience with a new application. A good example of a guided transition from command line to a GUI is

illustrated below (figure 4). This interface combines a GUI with a traditional command line option which can be collapsed. The user has an option of typing out the entire search query which gets updated in the GUI fields above. This helps transition   from one version/level to the other by using the prior expertise of the user.
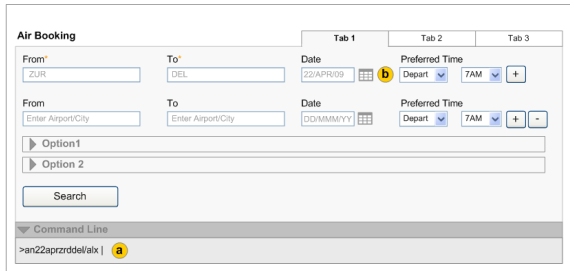

Figure 4: Combination of a command line with a Graphical User Interface.

4.2.2. For Experts

For experts, use the knowledge of task and interface concepts across the suite of applications. This is also known as developing reusable Patterns of interaction (similar to the Yahoo pattern library, Welie.com, etc.)

Standardization of certain features like a) search options ,b) error conditions and c) login widgets makes the application suite easy to comprehend. (figure 5)
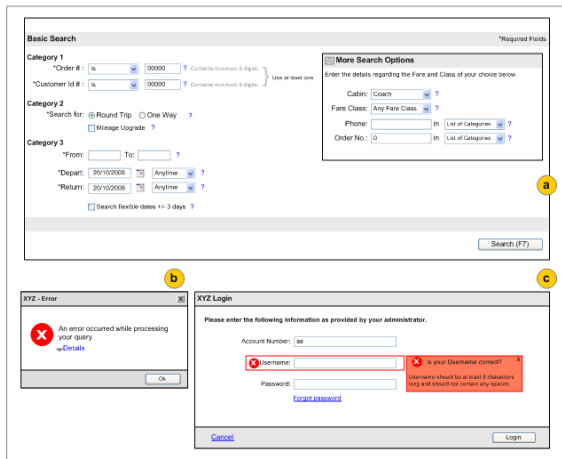

Figure 5: Pattern libraries aim to standardize interactions across multiple modules or varied applications in a single suite.

4.3. Provide alternate paths to expert and novice users

4.3.1. Novice Users

Novice users progress by trial and error methods and thus have higher chances of running into road blocks when given a task to perform on an application. Especially, if the workflow is complex. If the common tasks or "happy flows" are already defined in a certain way in the application interface it shall act as guides to the users to go about the task in a fluid manner.

The help and other supports could be built around these "happy flows". Expert users will have the choice to skip any of the steps that in their experience are not required in a specific situation or the interface can give the expert users a choice to execute tasks in one go, without using a broken down option.

4.3.2. Expert Users

We all tend to evolve our own strategies to quicken our pace while doing a task repeatedly, we form our own models and go along them. Experts are better at this process than the novice users and thus an interface trying to accommodate both the behaviors should provide for tools and workflows that make this process easy.

Again, this increase in efficiency can be achieved in two ways. First, by using shortcuts like hotkeys, extended facility to use a command line interface etc.

And secondly, by using the domain experiences of the expert users. These could be in the form of decision support tools or suggestions that give alternate routes and flexible options in the task flow based on the situations that the expert users have been facing over a large span of time.
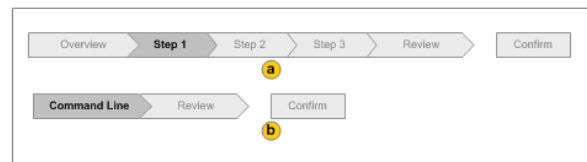

Figure 6: Command line input as an alternate option for the same set of steps/wizard (refer Figure 7).
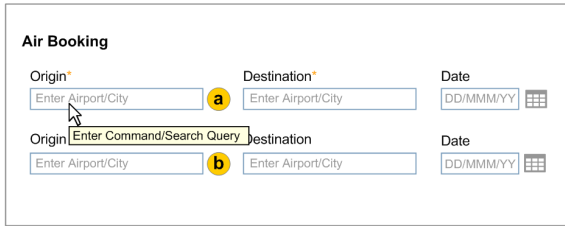
Figure 7: The command line option can also be combined in the way shown above:
a) The first field is "hybrid" and can either be used to enter an airport/city code or an entire command line search string for the complete query in one go.
b) The second field is "normal". It is used only to key in an airport/city code.

4.4. Allow for tiered customizations to match increasing expertise

A good enterprise application allows for increased expertise. Everyone starts as a novice, grows to an intermediary and later becomes an expert. And this tendency is around specific areas or modules or flows of an application. A good application design allows users to quickly gain a degree of competence and start being productive and efficient. This means that the same interface and set of tools needs to be used differently at different behavior levels. This can be achieved by intervening at the following levels.

4.4.1. At Application Level

At an application level, a user can be given the rights to decide more than the color scheme, skins or password. The size of fonts, the state of shortcuts or the display of error conditions can all be set as preferences at this level.
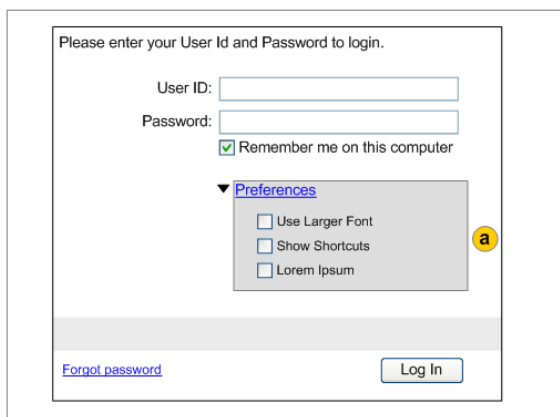

Figure 8: The illustration shows how preferences (a) can be set at the time of login into the application.

4.4.2. At Task Level

Then after, the user can be given some options at the task level as each individual is accustomed to looking at information in his own peculiar manner. Options to set the display of search results (all/with filters) or pagination in searches and matrix view have all been provided in certain supply chain applications. In supply chain and HR related applications such switches like (option between matrix or a tabular view) have been found to be of great advantage.

4.4.3. At Screen Level

Further, screen layouts should be customizable to allow prioritization of the content.
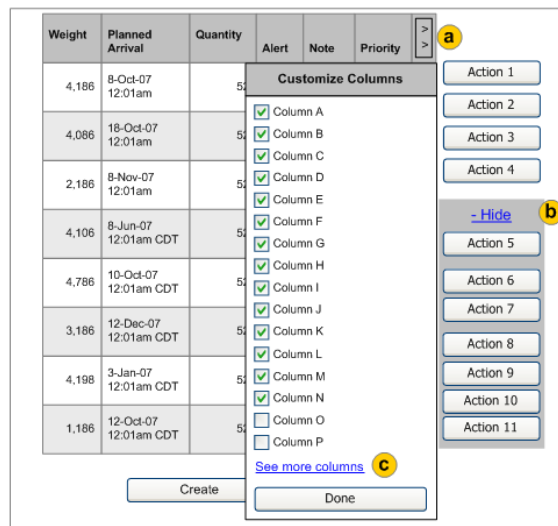

Figure 9: Example of progressive disclosure being used in a supply chain related enterprise application.

a) A button reveals a customize columns option where a user can select/deselect the visibility of the columns

b) Hide/Show mechanism for to prioritize frequently used actions.

c) Less frequently used columns and column sequence is revealed at the next level.

4.5. Provide cues to encourage novices to become experts

Sustained motivation to explore the options in an application is a viable concern for increasing expertise over time. One has to be able to complete common tasks easily and then have some alternatives to explore.

A good enterprise application allows this by providing scents or cues along the interactions and action objects in the form of suggested shortcuts or mouse over messaging as illustrated.
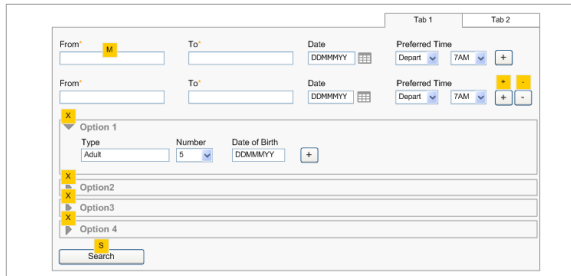

Figure 10: The illustration shows how the shortcuts are displayed on pressing a specific function key. These are overlays and all shortcuts are meant to be used with the alt key (alt+shortcut).
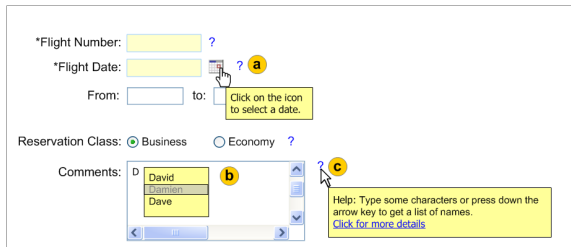

Figure 11: Several examples of extending cues to a novice user:
a) mouse over instructions on icons
b) auto-complete suggestions
c) mouse over help instructions wherever needed.

Along with these cues, recommendations from expert experiences and practices can be conveyed to the novice users during certain tasks. This design recommendation has been tried for a desktop travel application, where the system suggests successful searches made by other agents each time an agent runs a search for air, car or hotel on the application, prompting them to explore more and also helping them take a more informed decision.

5. Looking Ahead

The above guidelines and examples help in the process of designing an interface for enterprise applications. Analogies from domains like gaming which lay a lot of stress on creating levels of expertise in a singular interface may also serve as a good reference and have so far been overlooked.

Some research has also gone into cognitive aspects and differences between an expert and novice users and a few behavioral guidelines have also surfaced. Yet refinement and distillation of the abstract research into tangible guidelines and more comprehendible patterns and templates is needed.

6. Summation

To sum up the research and the experience (of the user interface design) in enterprise domain, one can conclude that most of the users are intermediaries and would need some expert or novice support to go about doing their tasks in an easier and more efficient fashion.

One needs to design keeping both the behaviors in mind. First account for novices and then build upon it to account for experts with some amount of customization for both the personas. Allow for stepped processes for novices and shortcuts/jumps for experts. Provide ample suggestions and clues to get the intermediaries to explore the application further, and finally sustain the motivation of all the various user segments for prolonged usage of the suite.

7. Acknowledgements

This paper would not have been complete without the support of the Principal Designers at Design For Use, especially Mr Nishant Jain who helped in reviewing the document.

Thanks to all the clients of Design for Use for extending their support to help us achieve our design goals and in turn making it easy to deliver better applications for them.

8. References

[1] Jain, N., Richard, D., Designing a Better Ballot: Achieving full usability within constraints.

[2] Enterprise Software.
http://en.wikipedia.org/wiki/Enterprise_software

[3] Reed, D., Creating Usable Applications.
http://www.simpletalk.com.

[4] Majlinda, F., Loskoska, S., Fetaji, B., and Mirlinda, E. Investigating Human Computer Interaction Issues in Designing Efficient Virtual Learning Environments. *In Proc. BCI 2007, Sofia, Bulgaria,* Technology Enhanced Learning, 318-322.

[5] Rauterberg, M. From Expert to Novice Behavior. *From Experience to Innovation*, IEA 97, 521-523.

[6] NEM: Novice Expert ratio Method. NOVAS Inc., Shizuoka University.

[7] Popovic, V., Expert and Novice User Differences and Implications for Product Design and Usability. 1-4.

[8] Jing, W., Accommodating Both Experts and Novices in One Interface
http://www.otal.umd.edu/UUGuide/jingwu/