

MMI & the user guide

The demands to end user communication

A definition of the interface between a machine and man must comprise not only the exact zone where man meets and operates the machine, but also include the overall “communication” taking place - along with the documentation (in whatever form it comes), describing the users proper operation of the machine, and expressing warnings of risks and hazards.



Not all gadgets are commonly thought of as “machines”, but in this text the word machine is used to describe mans counterpart in any MMI situation and the term must be understood in its broadest context: Paper clips are thus “machines” to hold sheets of paper together, shovels are “machines” for digging and space shuttles are “machines” for traveling in space.

Simple machines generally present a simple MMI - there is little “communication” with a paperclip or a shovel - and they are generally understood intuitively by the user. The ability to use such gadgets or machines springs from our basic understanding of the world we live in, and is often acquired through childhood play or explained to us by experienced users - learning to ride a bicycle is a good example. The MMI we encounter in these situations consists of simple controls and the “user guide” of cognitive experience or verbal instructions. As gadgets increase in complexity, more thorough instructions becomes necessary and by the same token, the users intuitive understanding of the MMI fades and he relies more and more on the documentation provided:

- On an ordinary shovel for digging, the **MMI** is represented by the handle - *and any verbal instruction or cognitive experience in the noble art of in digging that the user may have acquired.*
- In an automobile the **MMI** consists of the steering wheel, the pedals, the stick (if any) and the various switches and meters - *and the education given in a driving school - along with the instruction manual for the car and any warning or instruction expressed on the dashboard or under the hood.*
- On a computer (or a cell phone, or a VCR for that matter), the **MMI** is represented by the keys and knobs along with the readings on the display - possibly also by audio signals (sounds

and/or speech) and maybe even tactile signals (vibrations etc.)
- *and let us not forget the comprehensive (and often rather incomprehensible) 450 page manual supplied with the device or the software.*

- Supervising the operations in a nuclear power plant probably requires comprehensive documentation, manuals, flowcharts, instruction billboards etc. and thorough training, maybe years of education. The accrual interface with the controls, pushing buttons and reading meters become less a part of the **MMI** than the knowledge. Most likely you could train a monkey to push the buttons - it just wouldn't have a clue of what it was doing. *The actual interfacing takes place in the operator's brain, not at his fingertips.*

It follows, that the concept of MMI includes interfacing on multiple levels: physical, intellectual, emotional (more than you should think) and cognitive, and cannot be viewed solely as a physical act. That in turn constitutes the user documentation as an integral part of the MMI.

This recognition and the demands it raises to MMI and documentation can be expressed in a number of statements or "principles" that may serve as guidelines for the both the MMI developer and the technical writer.

1. The principle of SAFETY.
"See you in court, a.....!"

Safety comes first for many reasons, but as in so many other aspects of human life, selfishness is also here a powerful motivator. It is actually very simple: Unsafe products (including safety problems created by MMI's and documentation) tend to aggravate the user. If you are lucky the consequences of his aggravation will confine to a "no-buy" of your products the next time he needs to upgrade or renew, and poor recommendations to friends and relatives. If he is also hurt physically, financially or even just emotionally by your product, you can be pretty sure to hear from his lawyer. Loss of money, loss of time, loss of credibility, loss of business opportunity - all in all a scenario you don't want to be a part of.

In spite of what the sales department may wish, the safety aspect must be sufficiently underlined to attract the user's attention. Warnings must be unambiguous and prominently placed both in connection with the product and in the user guide, risks must be clearly revealed and first-aid measures described.

2. The principle of product and documentation UNIFICATION.
"Products and User Guides are Siamese twins".

You cannot separate a product from its user guide. Legally, technically and intellectually the user guide is a part of the product. Thus, a quality user guide not only *describes* the MMI, it is also a tool in *forming* the MMI. Therefore products and user guides must be created

in conjunction, mutually shaping each other during the creation process into a unified whole.

The understanding of this is important and somewhat a breach with tradition. In the good old days of today, many user guides are written from specifications laid down months or years before the work with the guide was initiated. In many cases, the product has reached full-grown maturity and is on the verge of being marketed before someone is ordered to write the documentation. The tech writer in this situation assumes the role of the historian, trying to understand and explain why Caesar defied the warnings and went to the Senate on idus Marti, rather than being able to advise him to stay at home.

3. **The principle of USER ORIENTED DESIGN.** ***“Now listen up, Mr. engineer...”***

MMI's should emerge from user demands. This is so trivial a fact that it hardly seems worth mentioning, but nonetheless the opposite happens again and again. Today's MMI's are largely designed by engineers alone, the offset often being the technical capabilities inherent in the hardware applied rather than what would be logical and desirable from a users point of view. This *“Inside/out”*- thinking, where the hardware end up forming the framework the MMI must be created within, often leads to the introduction of technologies and implementation of features **no user has ever requested**, simply because the possibility to create them exists.

It must be stressed however, that the engineering community does not alone carry the responsibility for this situation. Many factors contribute to the deadlock we are in, most prominently a lack of qualified and realistic input from sales- and marketing sources. Sales people tend to *“want it all”*, simply to be ahead of the competitors in every aspect and thus present R&D with a sort of: *“- gimme everything, so that I can pick the bits and pieces that I want”*- attitude. R&D in turn responds by narrowing this immodest wish list by focussing on the things they find important - no big surprise to anyone.

The obvious solution - since none of us are blessed with divine powers to unmask the secret expectations of every user - is to form a panel of engineers, sales people and technical communicators with the responsibility to clearly specify the MMI, thus utilizing the synergy between different skills and approaches from each group and letting each group feel the ownership to the result. The MMI specification must be detailed and i.e. comprise templates that developers can apply to their individual creations to create a homogenous MMI and must serve as a *“recipe”* (you wouldn't bake a cake without one, would you) for the joint efforts.

4. The principle of PREDICTABILITY & TRANSPARANCY.

“Hey, what the heck...”

Machines should do what the user expects. A lot of the gadgets we use today have their roots back to the dawn of industrialization more than a hundred years ago. During that period some ways of doing things have become so widespread acknowledged that they today are thought of as “standards” although never officially appointed as such. **Example:** Most vehicles with manual transmission have the 1st gear situated in the upper left “arm” of an imagined “H”. Placing the reverse in this position in a new model car will likely give most drivers an unpleasant surprise. **Another example:** In most countries light switches goes down to turn the light on and up to turn it off. Nobody has specified that, yet we expect it to be so and become surprised, when we experience otherwise as e.g. in Japan.

Transparency means that the operations of a machine or system are obvious. **Example:** - that you can immediately identify the coin slot in a vending machine and locate the reject money compartment. When left to ourselves with no instructions, we rely on the transparency of the gadget in front of us. **Another example:** Remember the first time you had to change a wheel on your car? Probably no one ever told you how to do it, and probably you weren’t able to find the instruction manual, but the operation was sufficiently transparent for you to succeed anyway. Transparency is thus important because it helps the user intuitively to understand what is going on and to do the right thing.

5. The principle of USER SUPERIORITY.

“I’m the boss, not you, you worthless piece of junk!”

Machines are made to serve man - not the other way around (although I admit, that with some software applications, you sometimes seriously doubt that said philosophy has been the offset for the design). In simple terms that mean that the user must be in charge of all operations and the MMI must **present the user to choices** rather than issue commands. Choices and their consequences must be clear and unambiguous and it must always be possible to abort actions initiated. Defaults must be the “safe” choice, i.e.: “- do you want to delete this database? Yes, No”. (No should be the default option).

Some computer software, i.e. the Microsoft Office package, features automated “help” to the user in spelling and grammar. Much as the user may need and appreciate this help, it should be a **choice** the user takes by activating various “radio buttons”, and not a default as the case is today. I may be a grumpy old man, but in my view, Microsoft’s attitude tends towards a “Big Brother” ideology, telling me what to do and when to do it. Although it is probably meant as helpful tool, it compromises the principle of user superiority, especially because the function is set as a default.

6. The principle of MINIMUM COMPLEXITY.

“Gimme the short version, for crying out loud!”

If a system MMI comprises multiple levels with increasing complexity, default must be the lowest level. The majority of users will

never have the need for the more advanced features of the system, and the fraction of users that does, will most likely be capable of performing the operations needed to climb to higher levels all by themselves. Comprehensive MMI's with an array of rarely used features packed into them waste valuable time and are confusing and aggravating for most everyday users. If for technical reasons a system requires a high degree of complexity, everyday users can be "protected" by passwords leaving only the necessary sections of the MMI open to them.

7. The principle of IMMEDIATE RESPONSE.
"Work! God dammit, work!"

The principle of immediate response must be observed. That is, that when the user issues a command to the system, this must by immediate action respond to the command. If the command itself cannot be carried out right away, the system must respond by informing the user that work is in progress, i.e. by displaying a working symbol such as an hourglass or similar. Preferably the system should also provide information on exactly what it is doing and how long the operation is expected to take.

8. The principle of CLARITY IN COMMUNICATION.
"Do you speak the language of your readers?"

User guides are often written by the same engineers who designed and developed the gadget. No doubt these people have vast insight in the product, they know every operation the product is capable of performing and every limitation to the use. And that is exactly the problem with that set-up. Knowing too much often result in superficial instructions and a "people know that"-attitude. The same applies to text messages i.e. in cell phone displays. "Use concatenated?" is a real engineer's term, but do you understand it? (It means "Send SMS up to 460 characters")

User guides should be written by communicators who are sufficiently ignorant to ask a lot of questions to the engineers, and sufficiently "technical" to understand the answers and communicate these to the user.

9. The principle of VERIFICATION.
"MMI's and User Guides must pass the test!"

There is really only one boss, only one judge to approve your creations and that is the user himself. If the user finds your work poor - it's poor! Therefore - while the engineers run the final tests on the product, you - the writer - must test your creation on layman users. You can elaborate a questionnaire to provide for a structured feedback, or you can conduct interviews with the test subjects to reveal the inevitable weak points. Both methods has their advantages and the important thing is that you actually get the necessary feedback And do also listen to the engineers - maybe there is a point or two, you haven't fully understood or maybe alterations has been implemented after design freeze. It happens.