# Three Switches and a Valve

About the Complexity of simple User Interfaces

Stefan Sachs

2010

**Why I wrote this Text**

About a year ago, I bought a new espresso machine from a manufacturer with good reputation. The results were delicious and my cappuccino consumption soared, so everybody should be happy. Unfortunately, I noticed, that frequently, when I am diverted, while preparing my cappuccino, I make mistakes. Now one may attribute this to first signs of dementia, probably caused by excessive caffeine abuse. Nevertheless, even after one year exposure to caffeine overdoses, the same does not happen to me, when I use the predecessor of this machine, which has been retired to our summer cottage, so this obvious explanation cannot be confirmed. Therefor I started an analysis of the new machines user interface (of course with the intention, to save my reputation as barista)

**Technical View**

To make a good cappuccino, we need hot water, about ninety centigrades and a pressure of about ten bar to brew the coffee. Further we need pressured steam to whip the milk foam. The machine has everything required:

- A heat exchanger, which can be set to 90 or 110 centigrades
- A water pump
- A steam nozzle

These components are controlled by:

- A main switch, which switches the machine completely off, located at the left side of the machine
- A pump control switch for the water pump at top left of the front panel.
- A water/steam switch to control the temperature of the heat exchanger at bottom left of the front panel
- A manually operated steam valve, at top right of the front panel

The temperature of the heat exchanger is indicated by two control lights:

- The first, located in the pump switch, lights when the brewing temperature has been reached.
- The second, located in the water/steam switch, lights when the heat exchanger is hot enough, to produce steam.

From an engineers point of view, these controls give the user straightforward access to any function of the machine.

**The user-oriented View: Seven Steps to a Cappuccino**

When we look at the steps required to prepare a cappuccino, we can distinguish seven states. State changes are triggered by one or more user interactions or by state changes of the machine, which are indicated to the user. We omit in this description the additional efforts of coffee-making which are not directly related to machine operation (like grinding coffee, placing the cup, filling the sieve, cleaning the sieve…)

| State | Actions or Events required to proceed to next State |
|---|---|
| Machine off | main switch on |
| Machine heating to brew | brew light on |
| Machine ready to brew | pump switch on |
| Machine fills cup | pump switch off, steam switch on |
| Machine heating for steam | steam light on |
| Machine ready to whip milk | open valve |
| Machine whipping milk | close valve, steam switch off, main switch off |
| Machine off | |

**Analysis of the Process**

We can see, that the three switches and the throttle can be operated independently, so the user can control sixteen different states of the UI.

The heating system has three temperature levels: Cold, 90°C for coffee and 110° for steam indicated by the control lights. So we have a total of 48 possible states of the system.

Out of these, only the seven different states mentioned above make sense in normal operation, one more state is useful to bleed the water supply.

So we have a chance less than one out of six, that an arbitrary state of the system is useful for our purpose.

If we look at the steps of the cappuccino process, we can for each of those steps determine the required interactions and determine the interactions leading to an undesired system state. It does not make sense, to calculate blindly the probability to finish the process successfully by simply comparing feasible to correct actions, since we expect a user to act on purpose on the machine. Yet it makes sense, to asses the affinity of each of those faulty steps to the intended outcome, because this is somehow related to the probability to deviate form the correct sequence. So it is rather unlikely, that a user will try to start the process by opening the steam throttle, because the throttle is tightly connected to the final process of stirring the milk. On the other hand, it happens frequently, that the user initially presses the pump switch instead of the main switch, because this switch is tightly related to the intended action and to the indicator LED attached to this step.

When the user has to perform multiple actions to proceed to the next step, the risk to omit one of these is rather high too.

Further it is reasonable, to look at the outcomes of a deviation. These range from negligible (to press the wrong switch a the beginning of the process doesn't do any harm and can be corrected easily) to a ruined cappuccino (like starting the pump before the temperature is reached), to an unusable machine by releasing the overheating protection (happens sometimes, if the machine is left on due to an incomplete final shutdown for an extended time period, and the required action to fix this problem is not described in the manual; it may even destroy the heater).

So evaluating even such a simple user interface is not trivial and requires a decent amount of (hopefully educated) guesswork.

Even when we do only a superficial evaluation of the process, we can determine quite a few hotspots, where chances of harmful operator errors cumulate:

The first step is really error-prone since there is no visual feedback, that the machine started heating and so the chance, that the machine remains switched on, when the user is distracted, is rather high. The outcome of this state is not harmful; heating up to 90 centigrades will not affect the machine. If the user has already loaded the sieve with coffee, this coffee will soon be ruined.

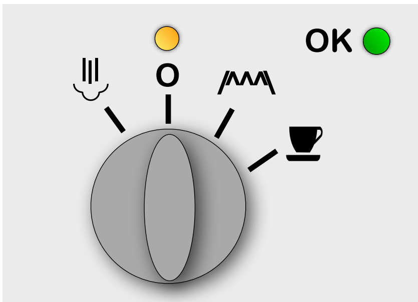
The next steps are not complicated: The user is waiting for the LED indicator, switching the pump on and off is closely related to the filling of the coup, therefore the user will nott be distracted and the tasks are straightforward.

Problematic are the last steps. Here we have a high cognitive workload; the proper use of the pannarello needs some attention (you have to stop, before the milk gets too hot), Usually the pannarello is removed after this step, cleaned under running water and put back to its position. To finalize the process correctly, two more user actions are required: switching off the steam switch and switching off the main switch. The steam indicator LED may be off at this point, so there is not necessarily a hint, that the machine needs further attention. All this contributes to a high probability, to leave the machine switched on for steam production, which can after an extended period of time trigger the overheat protection.

Another weak point is the indication of the correct temperature, when trying to brew another coffee immediately after using the pannarello. The system has to cool down, and the user has to wait for the brewing indicator to go off and on again to be sure, that the correct temperature has been reached. That requires continuous attention; alternatively he may use the pump to cool the heater down.

**How to improve**

A considerable amount of problems result form the design of the electric switches, which can be operated independently. While this may save some effort to the designer of the device, it offers abundant possibilities of user errors. Reducing the permutations by combining switches and enforcing sequential operations is definitely helpful.




If we look at the design of the other (cheaper) machine mentioned above, besides the valve, there is only one switch left. It has four positions, each corresponding to a meaningful combination of the three switches of the first machine. Further, the sequence is partially enforced by arranging the states, so that the proper sequence is followed by advancing the switch one step at a time.

The arrangement of the operating modes makes sense too: The off position is in the center, turning clockwise will advance us through the coffee making steps, turning counterclockwise through the milk foam process. These two processes are mutually exclusive (even some time to cool down is required, to produce coffee after steam).

So the different directions reflect the different operating modes.

The function of indicator lights is more consistent too: There is one operation indicator, which lights, when the machine is switched on. The second indicator lights, when the necessary water temperature has been reached (regardless of the operation mode). What is still missing is an indicator for too high water temperature, the user has to wait for the temperature indicator to go off and on again)

**Lessons to learn from this example**

First of all, what is straightforward from the internal design point of view, is not necessarily straightforward, if we take the process oriented approach. So a clean hardware design (and, if you open the machine, it is definitely beautifully designed hardware, superb quality and perfectly assembled) still can end up in an ugly user experience.

Second, even when taking the hardware designers viewpoint, the reliability of the machine is vastly improved, when inconsistent states are eliminated by design. As a first step, collecting the elements of the UI at one place would improve the perception of the actual state of the machine by the user. Further, by collecting the feasible combinations of states and classifying them, the designer can easily determine, which of them are useless, if not harmful and subsequently think about measures to lock them out (there has been a long and famous tradition of mechanical locks preventing misuse).

Third, further improvements are possible, when the design is changed to comply with standard way of use of the device. There exists no general approach for this task because each device has its own priorities, but in this case the limited number of uses cases without variations strongly supports a design, where the identified steps are reflected by some kind of progression in the user interface, as it is realized by advancing a turning knob (as realized in the second machine) or by using any other kind of sequential operation (The same considerations lead to the development of sequential gearboxes).

These improvements do not significantly contribute to higher cost of the system and they will be a large improvement of the user experience.

Any further steps would be more complicated: To integrate the control of the steam valve we would need an electromechanical device instead of the simple mechanical valve. The same holds for measures to prevent operations with incorrect temperatures; here either a possibility to block the operation of the switch or the transition to an electronically guided user interface would imply major design modifications. It can be doubted, if the cost of these modifications would pay off in terms of consumer satisfaction considering the device as a manually operated system.