A structured approach to the development of a set-top box user interface

Martin Springer

martin@camorra.org

Abstract

The paper reports our experience of developing a user interface for a set-top box. The paper describes the activities and processes used for communicating with other parties involved in the development. One of the resulting tools is a structured scheme for the coherent representation of a user interface design based on scenario templates.

1 Introduction

A common interpretation of the term convergence is that different technologies are unified into one technology that covers a whole field of applications. The digitalization of media has triggered the convergence of networks, services and communication devices [1]. Examples for convergence already occuring are Unified Messaging Services [2] offered by Telcos or handheld devices like the Visor [3].

Current communication devices enable their users to access a variety of services which were bound to different devices and networks in the past. Consequently, the development of user interfaces for these devices has become more complicated.

Convergence Integrated Media GmbH is developing an operating system for digital television set-top boxes. The software is called LinuxTV and is based on the open source software Linux.

The concept/ui developer group (two concept developers and two interface designers) has developed the user interface of the LinuxTV set-top box. One of the main difficulties was to realize an effective communication between the designers and the other teams involved in the development process.

2 Quality Management

The user interface is the software layer that provides control of the set-top box. It is the final link between the product and the customer. Therefore, the quality of the user interface accounts for the customer's satisfaction.

In order to continuously improve the quality level of the user interface, we decided to follow the principles of the ISO 9000 norm family [4]. The norm attaches great importance on the ”process approach”, which postulates identifying and managing development work and activities as processes which are continuously controlled and improved.

Prior to beginning work, we therefore identified the underlying processes leading from concept to prototype. The development of a user interface represents an almost linear situation, where the output of one process often forms the input of the next process (see Fig. 1).

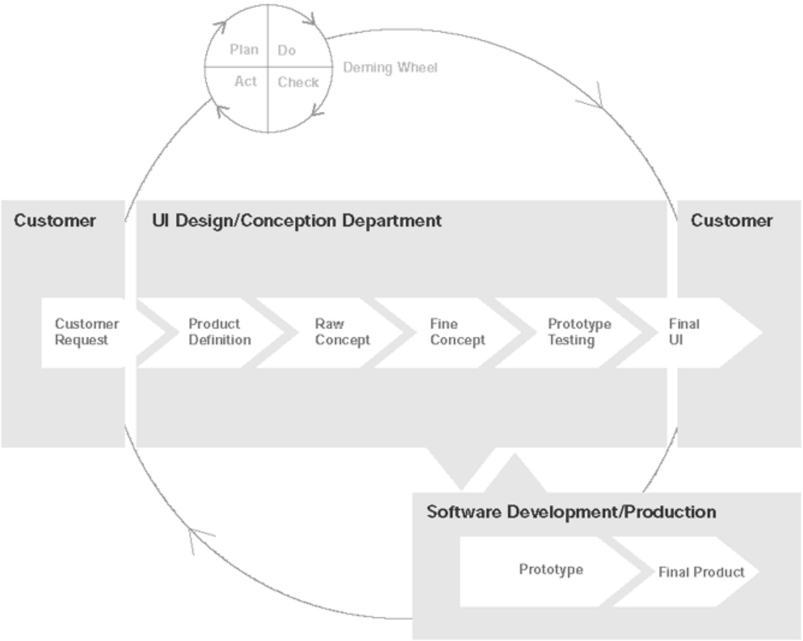


Figure 1: User interface development workflow

We were inspired by the works of W. Edwards Deming [5], [6], one of the fathers of total quality control (TQC). By applying principles like the Deming Wheel (Plan, Do, Check, Act) to our development process, we were able to detect weaknesses in concept and design and could take the appropriate corrective actions with respect to the quality. An exhaustive documentation effort of our work, as well as frequent and systematic design reviews helped us to achieve objective evidence of the work done and verify compliance to customer requirements in every phase of the development process.

3 Product Definition

Our first task was defining the product. By its nature, a set-top box running the Linux operating system can have all the functionality known of a computer. Since the majority of the TV users are computer illiterate, the presence of many features does not necessarily improve the product. Hence the core question was which applications were useful to the users of our set-top box.

As an input, we were provided with a possible hardware specification. A list of possible software features was derived of similar products already on the market (e.g. Tivo

[7]). This list acted as an ersatz for the acquisition of customer requirements from a real customer.

The user interface of the box would consist of two components:

- a remote control
- an on-screen display on the TV screen

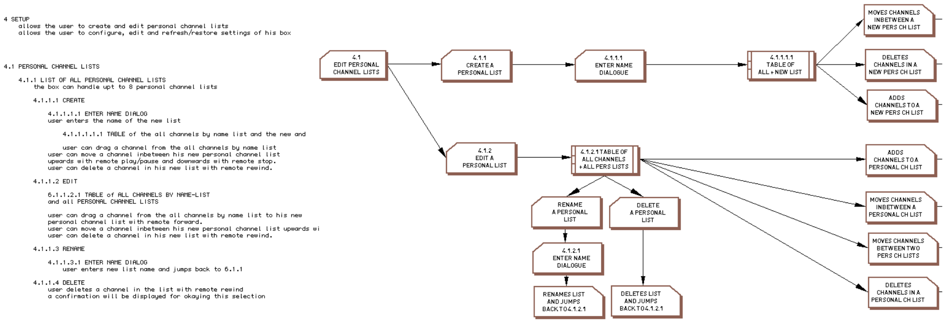
The user interface would allow users to:

- watch digital TV and listen to digital radio
- switch channels
- browse an electronic programme guide
- record programmes and schedule recording jobs
- replay recordings
- configure the box

Subsequently, other features were proposed to us and the features initially proposed slightly changed.

4 Raw Concept

Starting with the product definition, we produced a first concept paper. Initially, this raw concept was a text file describing the functionalities of the box and a graphical structure tree which structured the functionality in groups (see Fig. 2).



```
4 SETUP
    allows the user to create and edit personal channel lists
    allows the user to configure, edit and refresh/restore settings of his box


4.1 PERSONAL CHANNEL LISTS

    4.1.1 LIST OF ALL PERSONAL CHANNEL LISTS
        the box can handle upt to 8 personal channel lists

        4.1.1.1 CREATE

            4.1.1.1.1 ENTER NAME DIALOG
            user enters the name of the new list

                4.1.1.1.1.1 TABLE of the all channels by name list and the new and

                user can drag a channel from the all channels by name list
            user can move a channel inbetween his new personal channel list
            upwards with remote play/pause and downwards with remote stop.
            user can delete a channel in his new list with remote rewind.

        4.1.1.2 EDIT

            6.1.1.2.1 TABLE of ALL CHANNELS BY NAME-LIST
            and all PERSONAL CHANNEL LISTS

            user can drag a channel from the all channels by name list to his new
            personal channel list with remote forward.
            user can move a channel inbetween his new personal channel list upwards wi
            user can delete a channel in his new list with remote rewind.

        4.1.1.3 RENAME

            4.1.1.3.1 ENTER NAME DIALOG
                user enters new list name and jumps back to 6.1.1

        4.1.1.4 DELETE
            user deletes a channel in the list with remote rewind
            a confirmation will be displayed for okaying this selection
```

4.1 EDIT PERSONAL CHANNEL LISTS
4.1.1 CREATE A PERSONAL LIST
4.1.1.1 ENTER NAME DIALOGUE
4.1.1.1.1 TABLE OF ALL + NEW LIST
MOVES CHANNELS INBETWEEN A NEW PERS CH LIST
DELETES CHANNELS IN A NEW PERS CH LIST
ADDS CHANNELS TO A NEW PERS CH LIST
4.1.2 EDIT A PERSONAL LIST
4.1.2.1 TABLE OF ALL CHANNELS + ALL PERS LISTS
ADDS CHANNELS TO A PERSONAL CH LIST
MOVES CHANNELS INBETWEEN A PERSONAL CH LIST
MOVES CHANNELS BETWEEN TWO PERS CH LISTS
DELETES CHANNELS IN A PERSONAL CH LIST
RENAME A PERSONAL LIST
DELETE A PERSONAL LIST
4.1.2.1 ENTER NAME DIALOGUE
RENAMES LIST AND JUMPS BACK TO 4.1.2.1
DELETES LIST AND JUMPS BACK TO 4.1.2.1

Figure 2: Raw concept

To structure the raw concept with more detail, chapters and subchapters were introduced. After some iterations of structuring, a unique number could be assigned to every possible scenario [8] we imagined.

Some initial sketches of the on-screen navigation showed how the structure could be represented on a TV screen. The screens and the structure tree were presented to the Convergence management, who decided that the interface would satisfy the customer requirements.

5 Fine Concept

In order to better understand the interaction between the user and the set-top box, we modelled Unified Modelling Language (UML) activity diagrams [9] which visualized the different situations and the possible decisions a user could make with the remote control.

By means of the UML diagrams, we could identify certain patterns in the users' situations throughout the whole concept as well as certain scenarios we had not previously identified.

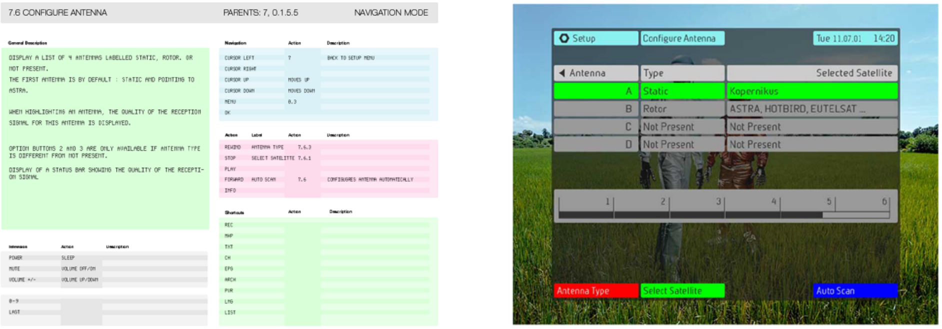
The UML activity diagrams helped us to restructure and simplify the raw concept. However, it became clear that a coherent representation of the user interface in the form of activity diagrams would be much too complex.

To solve this problem, we designed a template for the description of scenarios. A scenario template contains the following elements:

- number of the scenario (from raw concept)
- title of the scenario
- list of parents (which lead to this scenario)
- text description of the scenario (what the user can do, what the box will do, what appears on the screen)
- table of remote control buttons (name of button, action, description of the action)
- screen design

Every scenario (see Fig. 3) describes how the set-top box behaves if the user presses a certain remote control button. Remote control actions which lead to other scenarios are described by their respective numbers. Thus the scenarios are linked with each other and a software developer can determine how the box is supposed to work as a whole by browsing the set of scenarios.

By means of the scenario template we described the complete interaction of a user with the set-top box. The complete set of scenarios represented the fine concept of the user interface.



7.6 CONFIGURE ANTENNA PARENTS: 7, 0.1.5.5 NAVIGATION MODE

General Description

DISPLAY A LIST OF 4 ANTENNAS LABELLED STATIC, ROTOR. OR NOT PRESENT.
THE FIRST ANTENNA IS BY DEFAULT : STATIC AND POINTING TO ASTRA.

WHEN HIGHLIGHTING AN ANTENNA, THE QUALITY OF THE RECEPTION SIGNAL FOR THIS ANTENNA IS DISPLAYED.

OPTION BUTTONS 2 AND 3 ARE ONLY AVAILABLE IF ANTENNA TYPE IS DIFFERENT FROM NOT PRESENT.

DISPLAY OF A STATUS BAR SHOWING THE QUALITY OF THE RECEPTION SIGNAL

[illegible]	Action	Description
POWER	SLEEP	
MUTE	VOLUME OFF/ON	
VOLUME +/-	VOLUME UP/DOWN	
0-9		
LAST		

Navigation	Action	Description
CURSOR LEFT	7	BACK TO SETUP MENU
CURSOR RIGHT		
CURSOR UP	MOVES UP	
CURSOR DOWN	MOVES DOWN	
MENU	0.3	
OK		

Action	Label	Action	Description
REWIND	ANTENNA TYPE	7.6.3	
STOP	SELECT SATELITTE	7.6.1	
PLAY			
FORWARD	AUTO SCAN	7.6	CONFIGUGRES ANTENNA AUTOMATICALLY
INFO			

Shortcuts	Action	Description
REC		
MAP		
TXT		
CH		
EPG		
ARCH		
PVR		
LNG		
LIST		

Antenna	Type	Selected Satellite
A	Static	Kopernikus
B	Rotor	ASTRA, HOTBIRD, EUTELSAT ...
C	Not Present	Not Present
D	Not Present	Not Present

Figure 3: Scenario template

6 Prototype

Using the fine concept as guidance, the software developers implemented a prototype of the user interface on a set-top box. The prototype already allowed remote control navigation via on-screen menus, channel selection and tuning, the display of programme schedules and recording and replaying programmes.

During the production of the prototype, we systematically checked if the implementation followed the fine concept. This required a close collaboration between the software developers and our team, who compared the developer's TV screen with the scenario templates.

The LinuxTV prototype was presented at the Internationale Funkausstellung trade fair in Berlin. During the show we had the opportunity to have visitors test the user interface. We monitored the behaviour of the users and recorded the many conflicts and inconsistencies which were revealed.

7 User Interface Concept

Having evaluated the feedback from the trade fair we rereviewed the fine concept, recognizing problem areas.

We grouped different scenarios (e.g. applications, navigation, messages) and introduced rules for design elements associated with scenarios (e.g. navigation icons, message icons). We identified rules for the usage of the remote control (e.g. pressing a shortcut button twice reverts an action). Subsequently we produced a new version of the fine concept based on improved scenario templates.

Once the user interface concept was complete and consistent, it became possible to separate the function of the design elements from their actual graphical representation. Consequently we broke down the screen designs into specific components or modules.

Each screen of the user interface consists of several scenes. Scenes may consist of elements which have a certain state, a color, possibly a fontsize. Scenes are independent of the text content of their elements and their location on the screen. The dimensioning is described separately in a layout concept.

The resulting user interface concept contains:

- a diagram of the remote control
- the complete set of the scenarios
- an appendix explaining some concepts in text (e.g. 2-d navigation, parental control)
- an appendix listing the names and the meaning of the icons
- the set of scenes and elements together with color and fontsize
- a layout concept describing the dimensioning

8 Testing

The user interface concept was added to the CVS [10] repository of the software developers as an Acrobat pdf file. A set-top box, updated daily with the current snapshot of the software from CVS was installed in our office.

We split the testing process into 2 phases, the compliance testing and the user testing.

By means of a bug tracking system (Bugzilla [11]), we noted the differences between the current CVS snapshot and the user interface concept. For this puropse we compared the user interface concept with the implementation using a bound book of the printed scenarios. The bug tracking system enabled the software developers to efficiently debug the software and make it compliant to the concept.

Once the software implementation had reached a sufficient level of compliance, we could proceed with the next phase, the user testing. For the testing we assembled a representative panel of ”final users”.

We prepared a test protocol which described realistic tasks to be performed by the users. The users were asked to think aloud during the test session so that we could monitor their reactions. Finally, they had to fill a result form where they could express their feelings about the user interface highlights and shortcomings.

The user testing allowed us to spot the last defects of the user interface, and to measure its usability performance. The testing also generated valuable feedback for the marketing and sales department, since it pointed out the keys features of our interface, as reported by the users.

After the testing, we provided a final updated version of the user interface concept document to the software developers.

9 Conclusion

For an effective communication between concept developers, interface designers and software developers, a common language is required. The syntax of this language must be understood by the different parties involved in the development process.

A user interface for a set-top box is too complicated to visualize only by means of structure trees or UML activity diagrams. Furthermore UML is not easily understood by designers.

Linking scenario templates, which consist of design elements and textual descriptions of the interaction between input and output device make possible a complete and concise description of a user interface.

The scenario templates are the key for substantially simplifying the task of communicating a user interface design concept to software developers.

10 Appendix: Designing the Interface

10.1 Limitations

Set-top boxes are devices that merge or converge elements and characteristics of computers with those of regular TV sets. They are - more or less - computers lacking computers' input devices, which is a significant difference for the design of a user interface (UI).

The viewer (which is our word for user) cannot navigate on-screen-graphics with a mouse or trackball. In contrast to the many different possibilities of the set-top box his interaction is limited to the basic layout of his remote control.

The viewer might be computer illiterate, sobasic usage concepts known from the computer cannot be presumed as "learned" by the viewer. Even if a set-top box is way more complex than a TV it has to be as easy to operate as one.

Furthermore, the quality of the TV screen, being the output device for the box, is quite poor. This narrows down the design of a UI even more with regard to the choice of colors, typography and so on.

As digital TV enables the viewer to receive more than 1000 TV and radio channels, he is confronted with a vast amount of channel numbers and -names, programmes and programme information which the UI should manage and display in a reasonable way. To reduce the viewer's short-term memory load, he should only be confronted with certain chunks of information at a time.

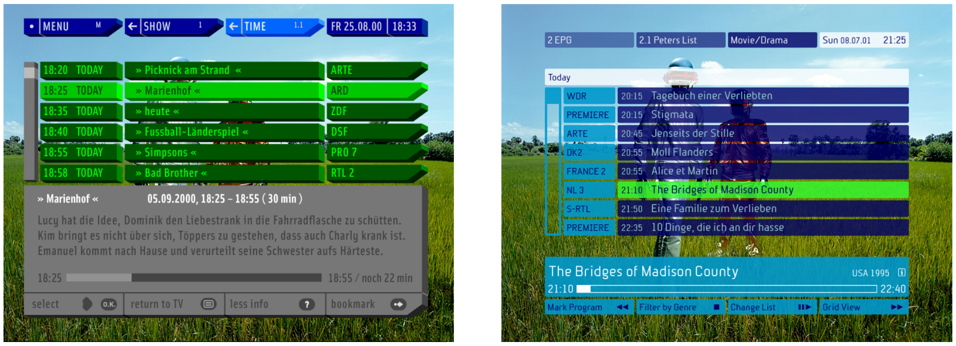
10.2 Concept

Before we started the actual design process, we defined the general goals for the UI of the set-top box. We wanted a UI which is highly consistent, easy to use and fast to learn, performing very well and provoking a minimum of usage errors. Any action unintended by the viewer should be easily reversible.

In our opinion, watching TV would still be the most important task for a set-top box. All other functions of the box - browsing through program information, recording TV on to the harddisk, or even accessing the internet - would be of second order relevance.

10.3 Design

Since it was technically possible to display the UI directly on the moving TV picture - some older systems still use a static screen as a backdrop for their user interface - and to blend the UI into the moving image, we deciced to do both and thus let the design be elegant and quiet: no three dimensional golden frames, no animated cartoon characters, no gimmicks.



18:20 TODAY	» Picknick am Strand «	ARTE
18:25 TODAY	» Marienhof «	ARD
18:35 TODAY	» heute «	ZDF
18:40 TODAY	» Fussball-Länderspiel «	DSF
18:55 TODAY	» Simpsons «	PRO 7
18:58 TODAY	» Bad Brother «	RTL 2

» Marienhof « 05.09.2000, 18:25 – 18:55 (30 min)

Lucy hat die Idee, Dominik den Liebestrank in die Fahrradflasche zu schütten. Kim bringt es nicht über sich, Töppers zu gestehen, dass auch Charly krank ist. Emanuel kommt nach Hause und verurteilt seine Schwester aufs Härteste.

18:25 18:55 / noch 22 min

Today

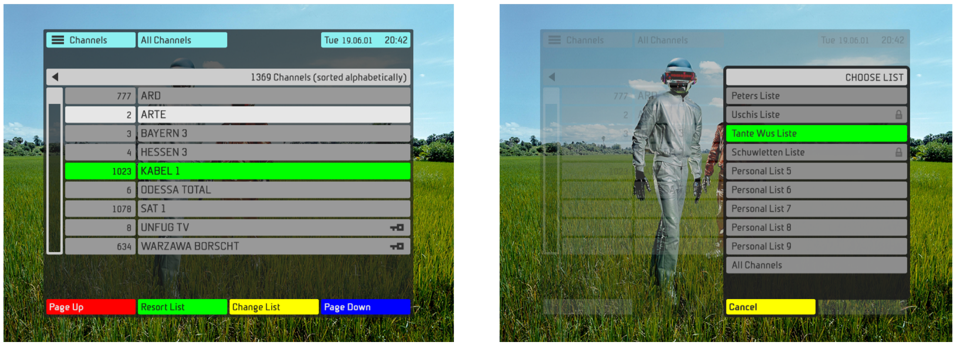
WDR	20:15	Tagebuch einer Verliebten
PREMIERE	20:15	Stigmata
ARTE	20:45	Jenseits der Stille
DK2	20:55	Moll Flanders
FRANCE 2	20:55	Alice et Martin
NL 3	21:10	The Bridges of Madison County
S-RTL	21:50	Eine Familie zum Verlieben
PREMIERE	22:35	10 Dinge, die ich an dir hasse

The Bridges of Madison County USA 1995

21:10 22:40

Figure 4: Different versions of the design

Though we decided to use transparency for the UI, we made it almost opaque, so that the viewer would not get too confused by the two layers merged together, but would still be able to follow the programme being played beneath the interface (see Fig. 5).



1369 Channels (sorted alphabetically)

777	ARD
2	ARTE
3	BAYERN 3
4	HESSEN 3
1023	KABEL 1
6	ODESSA TOTAL
1078	SAT 1
8	UNFUG TV
634	WARZAWA BORSCHT

CHOOSE LIST

- Peters Liste
- Uschis Liste
- Tante Wus Liste
- Schuwletten Liste
- Personal List 5
- Personal List 6
- Personal List 7
- Personal List 8
- Personal List 9
- All Channels

Figure 5: Different levels of transparency

If an additional overlay appears on the UI (e.g. a list of choices must be shown), instead of displaying a new dialogue the UI under this overlay becomes almost completely transparent. Here we used transparency to show the viewer the context within a task and thus make the interface easier to navigate.

It was clear to us that the amount of text messages and dialogs alone would draw so much of the viewer's attention - with the TV picture moving in the background - that typography would be the most important design element for the interface. Drawing the viewer's attention through color was the second important task for the design.

Thus RGB green was chosen as the highlight color for selected options and items. It provides both a good contrast against the real colors of the running picture and is widely known by most of the viewers from the volume-control-displays in ancient TV sets.

Throughout the process of developing the interface design, it's basic elements evolved from a set of archaic buttons either pressed in or out of the TV background (see Fig. 4) into a set of fields aligned to a fixed grid. The color set changed from RGB-green for highlighted items and typography together with dark and light blue for all other elements to a new and bigger set to improve readability and navigability: green as highlight color, light blue for the header of the interface (history + date & time element) light and mid-grey for other elements (e.g list rows) and primary colors for each of the four action buttons.

We also detected that it is straining to see portions of the runnning picture between the different screen elements. We closed the gaps by introducing a translucent backdrop. Thus we learned to minimize the number of edges any interface shares with the TV background.

References

[1] Martin Springer, *On Convergence*, December 1999.
(http://camorra.org/on_convergence.pdf)

[2] Unified Messaging Services, see http://unifiedmessaging.com

[3] Visor, see http://www.visor.com

[4] The ISO 9001 norm specifies requirements for a quality management system where an organization needs to demonstrate its ability to provide products that fulfill customer and applicable regulatory requirements and aims to enhance customer satisfaction.
(http://www.iso.org)

[5] MIT Website about the work of W. Edwards Deming.
(http://www-caes.mit.edu/products/deming/index.html)

[6] Deming's 14 Points.
(http://deming.eng.clemson.edu/pub/den/deming_info.htm)

[7] TiVo, see http://www.tivo.com

[8] In Prakash, C. Rolland and B. Pernici, editors, *Information System Development Process*, IFIP Transactions A-30, Elsevier Science Publishers, September 1993.
(http://www.isi.edu/soar/johnson/wg81.ps)

[9] Unified Modelling Language (UML) activity diagrams, see e.g. http://www.aw.com/cseng/titles/0-201-89542-0/techniques/Activity.htm

[10] Concurrent Versions System, see http://www.cvshome.org/

[11] Bugzilla Website.
(http://bugzilla.mozilla.org)