

# Programming of a Visualization for a Robot Teach Pendant

Sebastian Galen<sup>1</sup>, Dirk Liedtke<sup>2</sup>, and Daniel Schilberg<sup>1</sup>(✉)

<sup>1</sup> Bochum University of Applied Sciences, Bochum, Germany  
daniel.schilberg@hs-bochum.de

<sup>2</sup> Miele AG, Munich, Germany

**Abstract.** The intention of this paper is to develop and implement a user interface for the teach pendant of the new generation of robot controllers by Stäubli. The user interface is realized on the basis of an already existing visualization in the production of Miele in Bielefeld. In order to guarantee a structured development, the method of software-reengineering is used. In the first two steps it is necessary to analyze the tasks and functions of the existing software and to model the current state and the actual technical implementations. The requirements for the redesign of the software have to be compiled next. With the help of this requirements it is possible to develop a new user interface which includes the future layout and utilization [1, pp. 385–387]. The finished software helps to introduce the new technology into familiar operational procedures without any training of the employees.

**Keywords:** Robotics · User interface

## 1 Introduction

In the age of globalization and irresistible technical progress it is indispensable to constantly improve the production and automation processes to be able to place new, innovative and competitive products in the market at a reasonable price. In order to automate the production processes, the application of robots plays a more and more important role. The control of a robot can also be used to control other devices within the robot cells via industry bus systems. Industrial computers with touch panels are often used to guarantee the service and the supervision of devices and the robot. These computers allow, e.g. the settings, the visualization of sensors or the starting of processes. In order to steadily improve safety and handling of the robots, controls have been developed whose teach pendants are able to take over the real programming tasks of the robots as well as over the just explained service and supervision of the external devices. As a consequence of individually provided visualizations indicated on this extended teach pendants, industrial computers with touch panel are no longer necessary.

At this, the always returning job of introducing the new processes and functions to the employees as simple as possible plays an important role. This will only be possible if, in spite of new devices and controls, the usual work flow can be retained and changes can be reduced to a necessary minimum. If the operating surfaces of new machines are

adaptable, optics and functionalities should be as close to the visualization known by the employee as possible in order to minimize the training effort and to maximize the acceptance. Hence, the operating surface of the new teach pendant should orientate itself very strongly towards those of the industrial computer with touch panel.

This paper deals with the development and conversion of the visualization of the teach pendant of the new robot control by means of a practical example in the production of Miele in Bielefeld where robots of the company Stäubli are used.

### 1.1 Objective of the Paper

In this paper, the basics of the necessary computer languages for the surface programming HTML and JavaScript are explained first. Afterwards, the requirements for the development of the visualization possibilities of the operating surfaces of touch panels are developed and adapted for the special use of Miele. The aim of this conversion is to program an adaptably applicable and very intuitively served visualization which relieves the later user who has to adapt himself from an old to a new operating surface. Besides, the enlargement and the adaptation of the operating surface should be very simple to move by additional functions for the programmer.

The visualization is realized with HTML. JavaScript is used for the control of the HTML visualization and VAL3 for the linking with the robot computer language.

### 1.2 Approach

In order to guarantee a structured development, the method of software reengineering is used. An already existing software is adapted to a new system in the form of a so-called “1:1\* separation”. This means that the structure of the software is redesigned, but it has still the same functionalities. The software reengineering method includes two

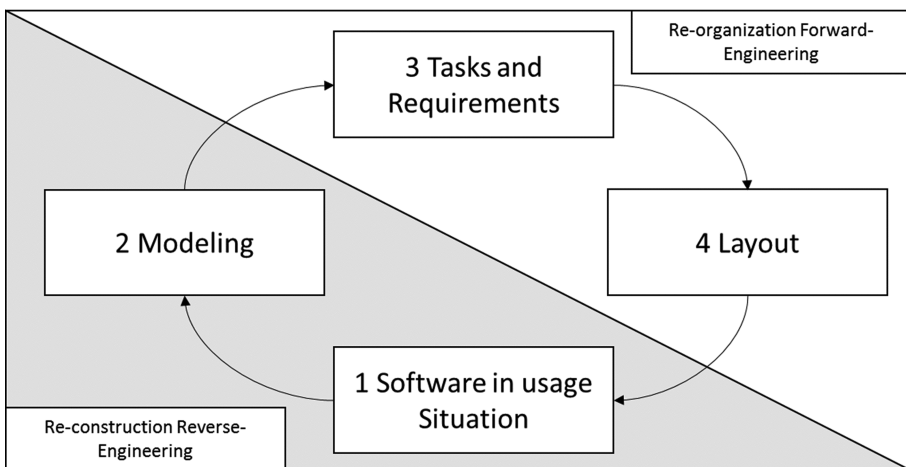


Fig. 1. Re-engineering method [1, pp. 385–387]

phases, the reverse engineering and the forward engineering, which are again subdivided into two steps.

During the first step (Fig. 1) the tasks and functions of the already existing software product are analyzed in order to model the current status and the present technical conversion is modeled (step 2). In the next step the requirements for the redesign of the software are listed regarding the point of view of the later user and the programmer. Finally, a new software concept can be developed by means of these requirements (step 4) [1, pp. 385–387].

## 2 Essentials

### 2.1 HTML

HTML means “Hypertext Markup Language” and is a standard markup language to create documents for web pages. The HTML code is interpreted by the browser and forms the basic construction of a website. The source code is built up with so-called “Tags” which are responsible for the representation and arrangement of the elements. These are embedded in so-called “Markups” [2]. In the following the basic structure of a HTML web page is shown:

```
<HTML>
<HEAD>
<TITLE>title of the website </TITLE>
</HEAD>
<BODY>
main part of the website
</BODY>
</HTML>
```

### 2.2 JavaScript

JavaScript is a computer language which is used and translated in all current web browsers. In contrast to the high-level languages, like C or Java, JavaScript is evaluated line by line and does not have to be compiled. Web pages written with HTML are very static, without any functions or events. JavaScript is used to make these websites dynamic. With JavaScript it is possible to change and control the appearance and the contents of websites, the web browser and document contents. A so-called “event handling” can be used to detect the operator’s actions, e.g. mouse click on a button. Different functions and events can be effected according to the detected action [3, 4].

### 2.3 VAL3

There is a huge number of computer languages for robots which differ within the manufacturers. The probably mostly used languages are VAL3 of the company Stäubli and KRL of the company KUKA. Both are very similar to the high-level languages, like C,

BASIC and Java. Loops and statements, main programs and subroutines as well as the declaration of variables form the basis of the program. In addition, special extensions have been implemented for programming of movements of the robot and input and output instructions.

### 3 Requirement Analysis

#### 3.1 Analysis of the Current State

As already described in the introduction, the robot control is able to control all external devices connected via bus systems, e.g. inputs and outputs or conveyor belts. In order to monitor and request these devices clearly, an additional industrial computer with an operating surface is required. The industrial computer is connected with the robot control and is able to communicate with all devices in the system.

The operating surface looks the same on every industrial computer, there are just little differences when it comes to the complexity of the robot cell. In spite of the separately running visualization on the industrial computer, the robot still has a teach pendant with which his movements can be programmed and controlled. As the robot control is connected to the local network, the programmer has access to the operating surface on the industrial computer (Fig. 2).

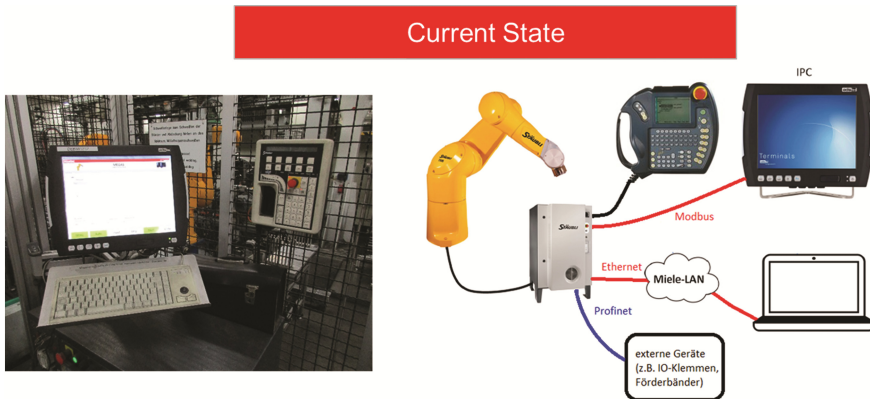
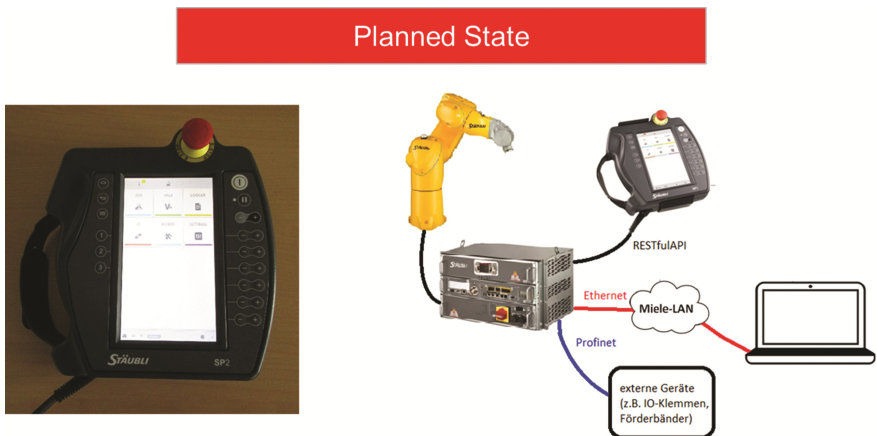


Fig. 2. Current state

One of the disadvantages of this set-up is the high costs for acquisition and installation because the integration of an industrial computer with touch panel is necessary in every robot cell. The maintenance of the software is very complex because the implementation of enlargements and new functions requires a solid knowledge of the computer language C# and the robot computer language VAL3. A change of visualization requires an immediate configuration and implementation of the programming environment of the Stäubli Robotics Suite. Combining the control and the functions of the robot and of the periphery just in the teach pendant would not have these disadvantages.

The new robot control of the company Stäubli allows exactly this (Fig. 3). It has a teach pendant on which not just the robot movements can be programmed and controlled, but also self-programmed visualizations can be shown. The programmer has no longer problems with the integration in the Stäubli Robotics Suite and the development of enlargements and functions because the visualization can be developed directly in the Robotics Suite. Therefore, an example program which contains the JavaScripts with the necessary functions is provided by Stäubli. Nevertheless, the programmer is free to create the surface and the development of other own functions. The surface is realized with the programming language HTML which is controlled with the functions of the JavaScripts and connected with the robot programming language VAL3.



**Fig. 3.** Planned state

### 3.2 Requirements for the Visualization

With the help of the new set-up described in the current state analysis arise different requirements for the new surface to be developed. This should also guarantee that the work routines remain unchanged for the operator.

1. Look and Feel:
  - Optics very similar to the old operating surface:
    - Menu structure
    - Arrangement and size of the operating elements
  - Same function of the buttons
  - Feedback after activity of an element
2. Modular site:
  - Inputs and outputs are indicated
  - Construction always similar, functions and labels different
3. VAL3 standards
  - Later changes of the program maybe just necessary in VAL3
  - Keep programming standard in VAL3

- Maintain program structure in VAL3

#### 4. XML databank

- A XML databank has to be used and integrated in the new visualization. In this databank the names of the single menu dots and sensors as well as the inscriptions of the possible switch states are built. By the use of this databank, updates or extensions of the software must not be customized in the source code, but only the accompanying entry in the XML file.

## 4 Realization

In order to describe the realization of the user interface with the help of the explained requirements in a structured way, the three programming languages HTML, JavaScript and VAL3 will be discussed individually.

### 4.1 HTML

The basic structure, the so-called layout, must be created first. In order to fill this layout with contents, the other HTML pages must be written and uploaded into the frames. Using the frameset layout, it is possible to divide the main window into various independently working windows. Hereby, updates of the layout and of the contents can be made separately. The individual frames can communicate with each other in order to change the contents of a frame by the click on a button of another frame [5].

Important requirements for the visualization are the arrangement and the size of the operating elements. The old touch panel is bigger than the new teach pendant. That is the reason why it must be paid attention to the fact that the elements do not become too small and are still comfortable for the operator when it comes to the handling. However, this problem only appears if several columns with operating elements are needed. For the solution of this problem a button has to be added which leads to a page on which other columns with operating elements can be displayed.

Any functions of the operating elements in the menu bar and on the main page have to stay the same. This allows an easier familiarization for the user with the new software. Every single function must be analyzed in detail with the help of the experts' know-how of the old system. In addition, after a button is pressed by the operator, the background color of it has to be changed as soon as the desired function is executed. This feedback must be initiated from the robot control. That is why the background color should not be overwritten by an already firmly defined color in HTML or in the Stylesheet (Fig. 4).

There are also several different menu dots which open a page on which different inputs and outputs can be indicated. The indicated modular page is always the same, merely the inscriptions and of course the functions of the buttons change. The inscription is performed by the import of a XML databank. In order to differ between the functions of the buttons depending on the selected menu dot, the VAL3 program has to call the suitable subroutine. In this subroutine the inputs and outputs can be selected and a feedback can be sent to the teach pendant.

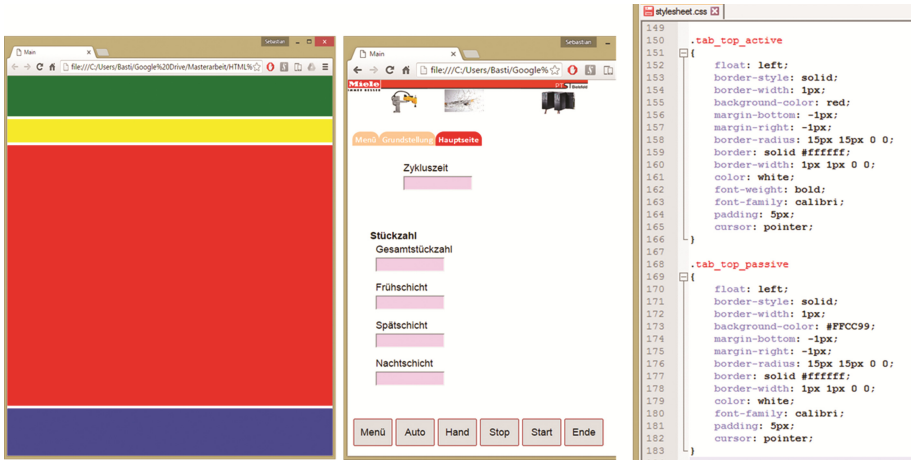


Fig. 4. Style and menu setup

### 4.2 JavaScript

In order to make the described HTML pages dynamic, different JavaScript functions have to be written. These functions are necessary e.g. to open pages, to control the navigation bar, to import the XML databank or to establish the link between the robot control and the visualization. Every HTML document has its own JavaScript which is read by loading the actual page. The functions of the scripts are now available to the document and can be activated. The JavaScript can access and change the operating elements of the HTML document with the help of an assigned ID or a certain tag.

The import of the XML databank and automatic marking of the operating elements are important requirements for the visualization. Therefore, a function was created which reads the XML file by means of a XMLHttpRequest and marks the operating elements (Fig. 5).

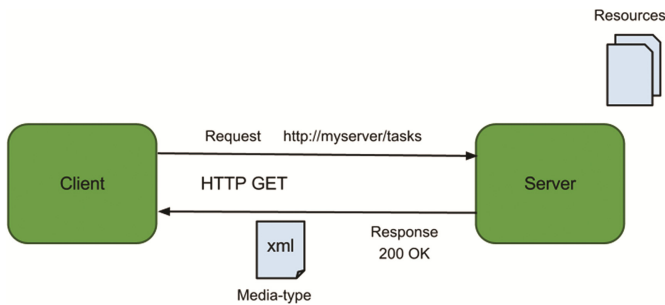


Fig. 5. Communication setup

In order to allow the communication between the operating elements on the HTML surface and the robot control, the company Stäubli has provided the already mentioned

JavaScript libraries. One of the functions, for example, is able to recognize the user's inputs and to transfer it to a variable in VAL3.

### 4.3 VAL3

In order to guarantee a trouble-free integration of the new visualization, the program has to be customized in VAL3 with the standards of the old software. Thus, existing robot systems can be put into operation with the new robot control and the new operating surface very fast and without major updates in its programming (Fig. 6).

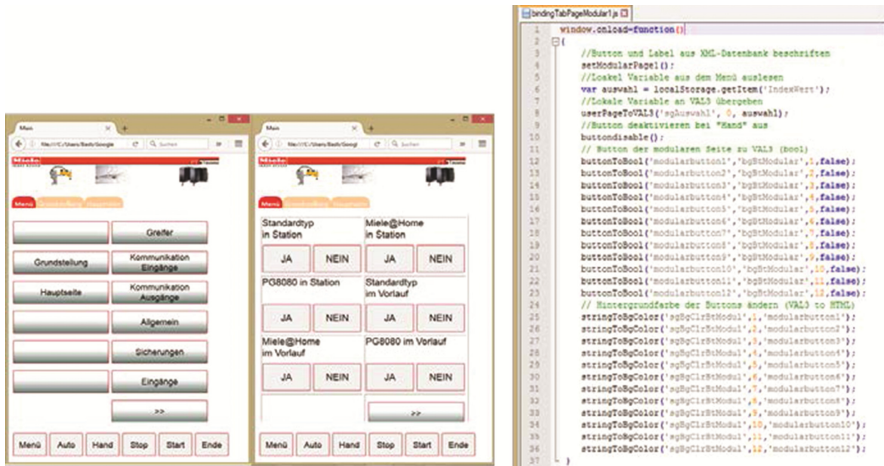


Fig. 6. Menu structure detachment

First the function “start()“ is executed automatically after the start of the application. The HTML surface is loaded on the teach pendant with the command “User-Page(“main”)“. Then the task for the control of the user interface is called. The initialization processes take place in the task to unset all variables, e.g. the background colors of the buttons. The execution of this initialization program is necessary only once before the cycle begins in the while loop. The calls of the subroutines which are responsible for the control of the HTML visualization take place successively in the cycle. All these programs are processed permanently, until the whole application is quit by the user.

## 5 Conclusion

The introduction of a new robot control to the production processes of Miele is supposed to replace the industrial computer in the future. The control and monitoring of the external devices is now possible via the new teach pendant. Therefore, it is necessary to develop a new user interface which guarantees the service of the connected devices with all functions used before. The main purpose of the present work is to make the conversion from the old to the new operating software for the user as easy as possible.



Therefore, the software reengineering process was applied. In the course of that process the old software product was analyzed and the requirements for the new software were defined. With the help of that a fully operational user interface corresponding to the requirements with the explained programming languages has been created. On the basis of this work and the new developments of Stäubli, the efficiency in the automation areas of the production of Miele can be increased in the future (Fig. 7).

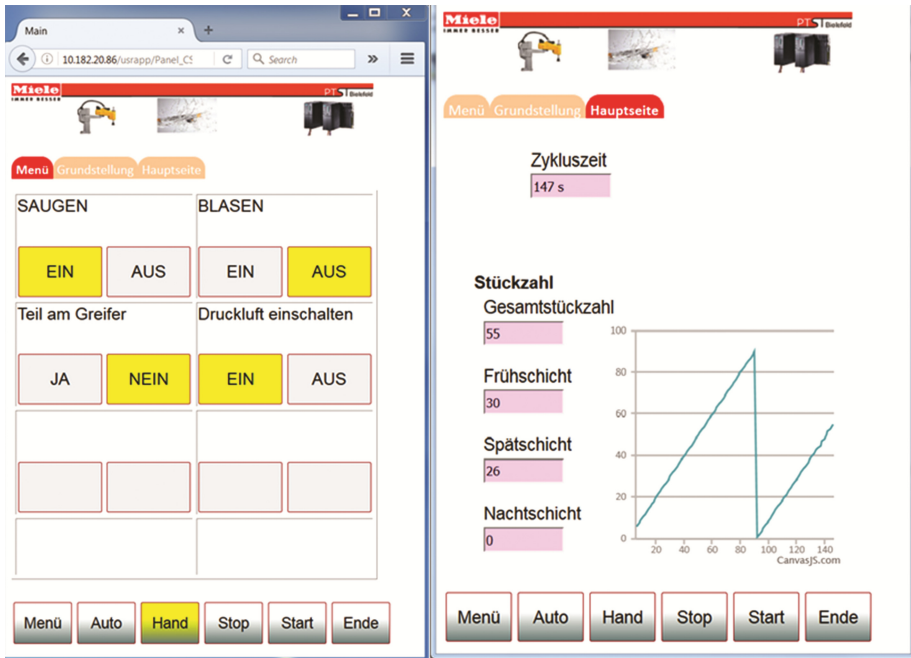


Fig. 7. Menu 1:1\* detachment

## References

1. Henning, K., Gramatke, A., Schilberg, D.: Informatik im Maschinenbau, Aachen (2008)
2. IT Wissen, HTML (2016). <http://www.itwissen.info/definition/lexikon/hypertext-markup-language-HTML.html>. Accessed 24 May 2016
3. Feuerstack, T.: JavaScript in 2 Tagen, Hagen (2004)
4. IT Wissen, JavaScript (2016). <http://www.itwissen.info/definition/lexikon/JavaScript-Java-Script.html>. Accessed 25 May 2016
5. Daum, M.: LOS!Web (2004). [http://los.webdaum.de/inhalt/inhalt\\_schule/seite\\_teilen.html#tab](http://los.webdaum.de/inhalt/inhalt_schule/seite_teilen.html#tab). Accessed 11 July 2016