AutoCAMS 2.0
Manual

Dietrich Manzey, Marcus Bleil, J. Elin Bahner-Heyne, Anne Klostermann, Linda Onnasch, Juliane Reichenbach & Stefan Röttger

Technische Universität Berlin
Fachgebiet Arbeits-, Ingenieur- und Organisationspsychologie (FGAIO)

September 2008

Index

1. Purpose and Overview

The present report provides a description of the micro-world AutoCAMS 2.0. It has been written as a manual accompanying the software package which represents a kind of toolbox for scripting and running experiments with this micro-world. The main purpose of this manual is to familiarize new users of AutoCAMS 2.0 with its main features and to provide a basic introduction how to use it as a research tool.

The manual is structured in six sections: Section 2 provides a brief review of the basic task and the history of its development including references to the research which has been conducted with this tool or one of its predecessors, so far. Section 3 represents the main part of the manual. It provides an in-depth description of the micro-world including its different sub-systems, its user interface, and the different functions which have been implemented in order to vary the conditions under which the task has to be performed. Details of data storage and of scripting experiments with AutoCAMS 2.0 are described in sections 4 and 5. Section 6 provides a brief description of the software and its implementation. Finally, section 7 provides a description of ManuCAMS which comes together with the AutoCAMS 2.0 toolbox.

Users who use the AutoCAMS 2.0 toolbox for the first time and are not already familiar with one of its predecessors, i.e. CAMS (Hockey, Wastell and Sauer, 1998) or AutoCAMS (Lorenz, DiNocera, Röttger and Parasuraman, 2002), can use the manual as a basic introduction and guide for an active exploration of the micro-world and its capabilities. Users who already know CAMS or AutoCAMS may find it useful to identify the changes which have been incorporated in the new version in addition to the complete revision of the user interface design.

2. Background

AutoCAMS 2.0 represents a “micro-world” (Brehmer, Leplat & Rasmussen, 1991) which has been developed to simulate a complex work environment. The task model chosen represents a supervisory process control system (Sheridan, 1997). The most classical example of such systems involves human operators supervising and controlling autonomous processes in chemical or power generating plants. In addition, similar supervisory control systems meanwhile are common also in vehicle control (aircraft, spacecraft, ships), manufacturing (robots, automated production lines), or medicine. In such systems the operator’s role is to monitor the progress of an autonomously running process in order to make sure that everything is going as planned and to identify any indication of a deviation from normal progress. In case of such indication the operator has to identify the underlying cause of the malfunction and to take any action to manage the fault, i.e. to take over manual control, and to intervene in the system in order to recover its normal functioning. In addition to this main task often other (secondary) tasks need to be accomplished, e.g. responding to special requests from the management, or performing maintenance tasks.

As a typical “micro-world”, AutoCAMS 2.0 does not claim to simulate the demands of real-life supervisory control systems on a high level of fidelity. Instead it presents a laboratory task which is, on the one hand, complex enough to allow for studying human performance in supervisory process control, and, on the other hand, easy enough to be performed by any subject with limited amount of training.

AutoCAMS 2.0 is largely based on the earlier developments of Hockey, Wastell and Sauer (1998) and Lorenz, DiNocera, Röttger and Parasuraman (2002). The credit for the basic idea and development of the first version of the task goes to Hockey et al. (1998). As part of a research project focussing on the effects of environmental stressors on complex work performance, Hockey et al. developed what they referred to as “Cabin Air Management System (CAMS)”. This CAMS still represents the core of AutoCAMS 2.0. CAMS is a generic model of a typical life support system of a closed and self-contained environment like a space capsule or submarine. The operator’s task is to supervise the system and to intervene whenever s/he detects a malfunction which might be caused by failures of any of the technical components of the system (see section 3 Task Description for an in-depth description of CAMS and this basic task). Originally CAMS has mainly been used to investigate predictions derived from the Compensatory Control Model (Hockey, 1997) about human performance consequences of different stressors, e.g. sleep deprivation (Hockey et al., 1998), occasional night work (e.g. Sauer, Wastell, Hockey & Earle, 2003), or confinement and isolation (Sauer, Hockey & Wastell, 1999). Yet, recently CAMS also has been used as a tool for studying effects of operator characteristics and different training approaches to establish and develop complex performance skills (e.g. Burkolter, Kluge, Schüler, Sauer & Ritzmann, 2008; Hockey, Sauer & Wastell, 2007). A bibliography about the research conducted with CAMS (as of September 2008) is provided in the Appendix.

An important step of the further development of the task was performed by Lorenz et al. (2002). In order to study performance consequences of the use of automated decision aids, Lorenz et al. added an Automated Fault Identification and Repair Agent (AFIRA) to CAMS which supports the supervisory control task of the human operator to a varying degree, operationally defined through different function allocations ("levels of automation", Lorenz et al., 2002). The resulting overall system was called "AutoCAMS" and has repeatedly been used for investigations of human-automation interaction since then (Bahner, Hüper & Manzey, 2008; Lorenz et al. 2002, 2003; Röttger, Bali & Manzey, 2008).

AutoCAMS 2.0 represents an advanced version of these earlier systems, covering the basic functions of CAMS and AutoCAMS under one umbrella. Furthermore, a special version of the system has made available where a defined number of processes must be controlled manually by the operator (see section 7 ManuCAMS). Compared to the original versions of this micro-world many changes have been implemented, including a completely new interface and some changes of task features which are described in some detail in section 3 of this manual. However, the most important change is a completely rewritten source code. Whereas the former versions of CAMS and AutoCAMS were designed using MS Visual Basic, the current version is written in Java SE 6. This makes it possible to run the system on any computer platform, i.e. independent of any specific operating system. In addition, the program code has been completely modularized, so it is much easier now than in earlier versions to adapt the system's characteristics to the specific requirements of a given study.

3. Task Description

AutoCAMS simulates an autonomously running life support system of a spacecraft (or other self-contained environment) consisting of five subsystems that are critical to maintain atmospheric conditions in the space cabin. The primary task of the operator involves supervisory control of the subsystems including diagnosis and management of possible system faults. In addition, the operators have to perform two secondary tasks: (1) a “connection check” where the operator has to confirm the proper connection with the space cabin whenever s/he is prompted to do so, and (2) a “recording task” where s/he has to repeatedly check the current CO_2 level of the cabin and to record this date in a data-bank.

In the default version of the system the primary supervisory control task solely is supported by a general warning function. In case of a fault, a general master alarm occurs which is indicated by a visual signal, i.e. a green light which turns red as soon as a fault is detected in any subsystem. The operator’s task is then (1) to identify which of the different sub-systems has been affected, (2) to take over manual control over this system in order to stabilize the overall system’s state, and (3) to search for the specific cause of the fault. In order to perform these tasks, the operator has access to different sources of information about the system state (e.g. history graphs for all system parameters, recordings of tank levels, or flow rates at certain valves of the system), as well as access to a control menu which allows for active intervention in the different sub-systems. As soon as a system fault has been identified, the operator has to activate an appropriate repair order to recover the system from the fault. The repair of the system usually takes 60 seconds. During this time, the operator has to control the system manually. After successful repair of the fault (indicated by a change of the red warning light to green), the operator needs to give the control back to the system.

In the assisted mode, the supervisory control task of the operator is supported by different versions of an automated diagnostic aid (*Automated Fault Identification and Recovery Agent*, AFIRA). Dependent on the specific version of AFIRA selected the aid provides an automatically generated diagnosis of the cause, as well as further automation support for fault management (see below section 3.3).

In the following important components of the system are described in more detail, including the five sub-systems (3.1), the interface (3.2), the system faults which might occur, the available automated aids (3.4), and the secondary tasks which have to be performed (section 3.5).

3.1. AutoCAMS 2.0 Subsystems

AutoCAMS 2.0 consists of the five subsystems oxygen, pressure, carbon dioxide, temperature, and humidity. For each subsystem, a target range, as well as a normal range are defined. The “normal range” represents the range within which parameter may vary without seriously affecting the health and comfort of the crew. The “target range” is a more strictly defined range which defines the target limits for the autonomous control system, i.e. the limits within which parameters are kept as long as the control system operates in a fault-free state. However, due to the internal dynamics of the autonomous process control, parameters may go beyond the target range for short times even in states of correct operation.

3.1.1. Oxygen Subsystem

The oxygen sub-system controls the level of oxygen in the ambient air of the cabin and ensures that there is a steady and safe oxygen level in the breathing air for the crew. It consists of an oxygen tank and a pipeline controlled by valves. The following parameters are defined to represent the two ranges:

Normal range: 19.0 – 20.5 %

Target range: 19.6 – 20.0 %

Whenever the oxygen value drops below 19.6%, the main oxygen valve automatically opens and oxygen flows into the cabin. When the oxygen value rises above 20%, the valve is closed automatically.

3.1.2. Pressure Subsystem

Beside a sufficient level of oxygen, a constant atmospheric pressure is needed in the cabin. Cabin pressure is regulated by the control of the different gas levels (i.e. oxygen, nitrogen, CO_2) in the ambient air. Similar to the oxygen system, the pressure system also consists of a tank and a pipeline controlled by valves which are involved in this regulation. The following parameters are defined to represent the ranges of normal and target pressure:

Normal range: 970 – 1040 mbar

Target range: 990 – 1025 mbar

Whenever the pressure drops below 990 mbar the nitrogen valve automatically opens. When the pressure rises agian, the nitrogen valve is closed. Pressure rises above 1025 mbar cause an hyper-pressure valve opens and a vent off of cabin air until pressure is in a save range again.

3.1.3. Carbon Dioxide Subsystem

In a closed environment the carbon dioxide which is exhaled by the crew members living in this environment represent waste and needs to be removed from the cabin air. In case of AutoCAMS this is processed by special scrubbers. The following parameters are defined to represent the two ranges:

Normal range: 0.1 – 0.8 %

Target range: 0,2 – 0,6 %

The scrubber is turned on when the carbon dioxide concentration exceeds 0,6%. The scrubber is turned off when the concentration is below 0,2%.

3.1.4. Temperature Subsystem

The temperature in the cabin is regulated by a heater and a cooler in order to provide comfortable conditions for the crew members. The following parameters are defined to represent the two ranges:

Normal range: 18,5 – 23,0 °C

Target range: 19,5 – 22,0 °C

Whenever the temperature falls below the target range, the heater is activated. In case of temperature rise beyond the upper limit of target range the cooler is activated.

3.1.5. Humidity Subsystem

Humidity within the cabin is regulated by a dehumidifier in order to provide comfortable environmental conditions for crew members. The following parameters are defined to represent the two ranges:

Normal range: 36,5 – 44,0 %

Target range: 38,0 – 42,0 %

3.2. AutoCAMS interface

The AutoCAMS interface consists of the following elements that are described in detail below (see Figure 1): a) history graph; b) flow chart; c) control panel; d) master alarm; e) repair menu; f) AFIRA message display; g) time display; h) secondary task: connection check; i) secondary task: CO_2 check.

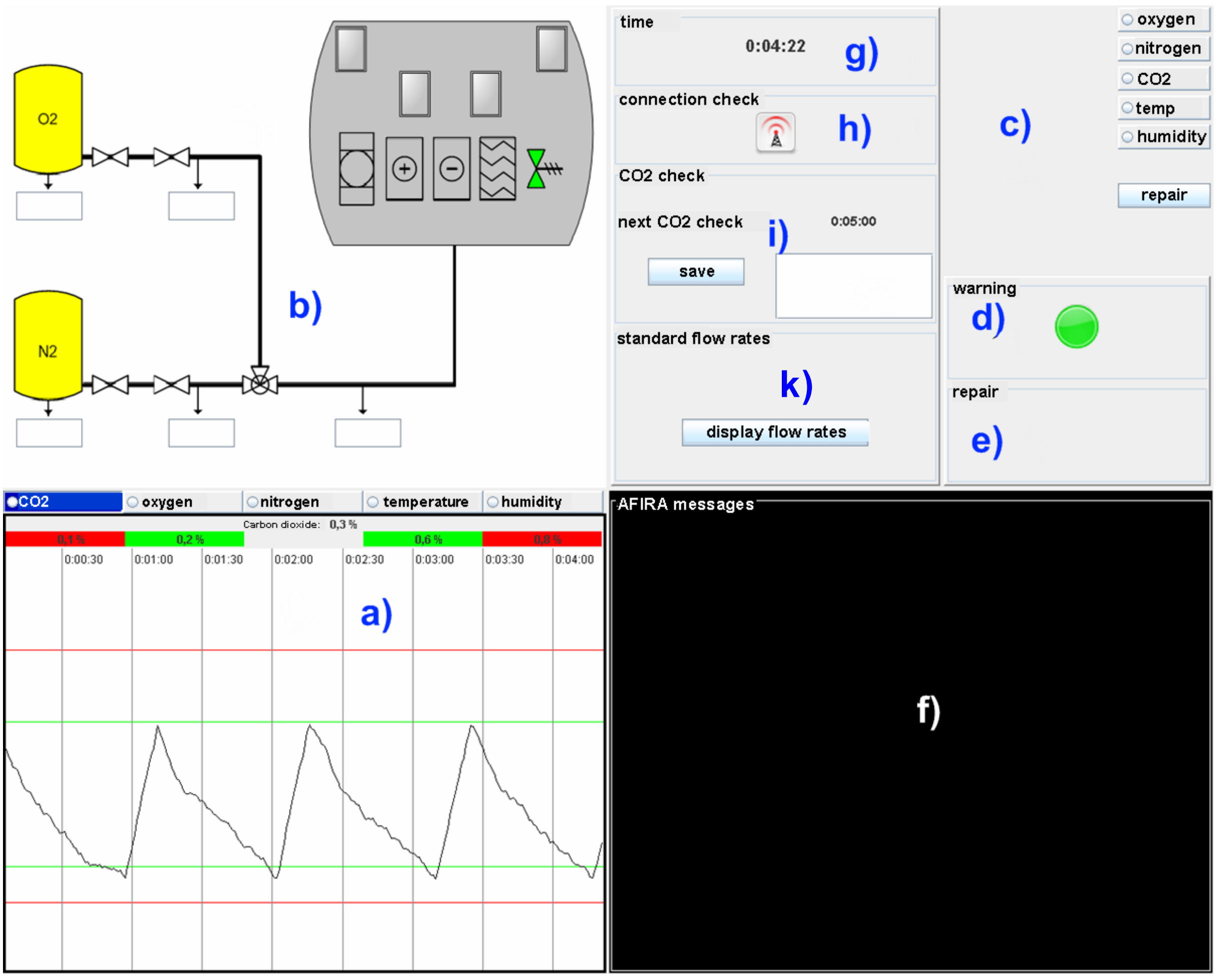

Figure 1 AutoCAMS 2.0 user interface

3.2.1. History graph (a)

The history graph in the lower left hand quadrant shows trend information for the past 240 seconds for each of the parameters oxygen, pressure, carbon dioxide, temperature, and humidity. It is organized as a tab strip window with each tab representing access to a specific sub-system. Within the window for each sub-system the target range and the normal range are marked with green and red boundaries, respectively. Values exceeding the red boundaries and thus being outside the normal range are considered to be hazardous for the crew and the technical equipment in the cabin. Values passing the green boundaries (target range) may be taken as indicators of a system fault, at least if the deviation persists. The selected parameter and its current value (in digital format) is displayed in the upper part of the history graph.

The carbon dioxide information which repeatedly has to be checked and recorded (see section 3.5) is displayed by default. The other parameters can be accessed via mouse click on the appropriate tab. They are displayed for ten seconds. After ten seconds, the default parameter (CO_2) appears again.

3.2.2. Flow chart (b)

The flow chart in the upper left hand quadrant displays a schematic overview of tanks, pumps and valves for the two subsystems oxygen and nitrogen (pressure), as well as an operation mode display for the carbon dioxide scrubber, heater, cooler, dehumidifier and pressure relief valve (from left to right below the windows of the cabin, see Figure 2). Information about tank levels and gas flow are shown for ten seconds when accessed by a mouse click on the appropriate fields. Figure 2 displays all elements of the flow chart.

3.2.3. Control panel (c)

The control panels are located in the upper right hand quadrant of the interface. The user can select any of the different control panels for single sub-systems by mouse-clicking on the appropriate field. Then specific panels are displayed which allow for manual control of the selected sub-system. In the default version of AutoCAMS, the operator can solely control the oxygen and pressure sub-system manually by setting the flow rates of a parameter to be „standard“, „medium“, or „high“ under both manual and automatic control. Yet, control of the other sub-systems can selectively be activated in the program code.

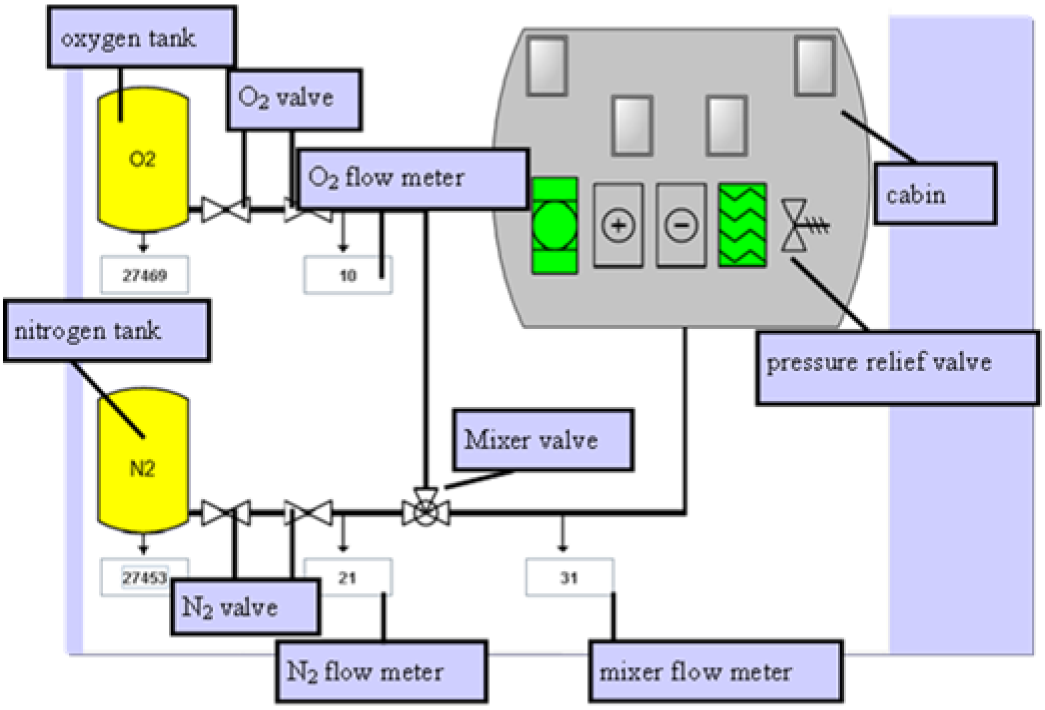

Figure 2 AutoCAMS 2.0 flow chart

3.2.4. Master Alarm (d)

In the upper right hand quadrant, there is a warning light representing the Master Alarm. Whereas the light is green during fault-free operation, it turns immediately red whenever a fault is detected in any of the different sub-systems. Subsequent to the fault repair it turns green again.

3.2.5. Repair menu (e)

Repair orders can be sent via the repair menu where the appropriate repair order has to be selected. It then takes 60 seconds to repair the system fault. During this time, the repair menu is inactive. If a wrong repair order was sent, the alarm stays red beyond the usual 60 seconds needed for repair, and a new repair order can be sent.

3.2.6. AFIRA message display (f)

The AFIRA message display is located in the lower right hand quadrant of the interface. Dependent on the activated version, AFIRA provides messages about the elapsed time since the occurrence of the error, an automatically generated fault diagnosis, and recommendations for fault management (see section 3.4. for a more detailed description).

3.2.7. Time Display (g)

The time period since the start of a block is displayed in the upper right hand quadrant.

3.2.8. Connection Check (h)

This field is required for the “connection check task” which the operator has to perform as one of two secondary tasks. A command button with the “connection check” symbol is displayed to prompt the operator’s action (see section 3.5.1 for details).

3.2.9. CO_2 Check (i)

This panel including a text box and a “save” button is needed for the second secondary task the operator has to perform, i.e. the CO_2 check and recording task (see section 3.5.2 for details).

3.2.10. Standard flow rates (k)

Pressing the button "standard flow rates" displays a table showing the flow rates that should be observed if all valves are free of blockages and leaks. The values are given for both oxygen and nitrogen and for each level of the flow rate settings (standard, low, high). As these values depend on the cabin pressure, they will change over time.

3.3. System Faults

There are nine possible system faults that may occur during simulation. These faults are characterized by specific patterns of symptoms depending on the subsystem wherein a malfunction occurs and what part of it fails. The malfunctions correspond to four different types of faults which either affect the oxygen subsystem, the pressure subsystem, or the mixer valve. These four fault types are:

Leaks in a valve: A considerable proportion of the resource (nitrogen, oxygen) escapes from the valve, i.e. less is supplied to the cabin than is leaving the tanks, resulting in a reduced supply to the cabin and a waste of the resource.

Blockage in a valve: The valve cannot be opened completely and remains partially blocked resulting in a reduced tank outflow and also a reduced supply to the cabin.

“Stuck-open” valve: The valve cannot be closed, so more of the resource than necessary is supplied to the cabin which also results in a waste of the resource.

Defective sensor: The sensor for the resource in the cabin does not work properly resulting in no automatic control response. This could be either a failure to turn on the supply when the lower threshold has not been recognized or a failure to shut off the supply when the upper threshold has not been recognized.

The following table (Table 1) provides a detailed overview of the different changes that occur in the overall system in case of each of the system faults. Knowledge about these “symptoms” allows to identify the cause of a given fault and, thus, to select an appropriate repair order.

Table 1 System faults and their symptoms

	Oxygen	Nitrogen	Mixer
Valve leak	**Leak of oxygen valve** O_2 decrease in the cabin Tank outflow reading > flow meter reading Flow meter reading can be smaller than default flow setting	**Leak of nitrogen valve** Pressure decrease in the cabin Tank outflow reading > flow meter reading Flow meter reading can be smaller than default flow setting.	--
Valve blockage	**Blockage of oxygen valve** O_2 decrease in the cabin Tank outflow reading = flow meter reading Flow meter reading < default flow setting	**Blockage of nitrogen valve** Pressure decrease in the cabin Tank outflow reading = flow meter reading Flow meter reading < default flow setting	**Blockage of mixer valve** Pressure decrease in the cabin O_2 decrease in the cabin All flow meter readings (mixer, N_2, O_2) indicate a reduced flow
Valve stuck open	**Oxygen valve stuck open** O_2 increase in the cabin Flow meter reading indicates permanent O_2	**Nitrogen valve stuck open** Pressure increases up to the upper end of	--

	flow Pressure raises up to the upper end of target range (venting function prevents an increase above target range)	target range (venting function prevents an increase above target range) Flow meter reading indicates permanent N_2 flow O_2 graph looks irregular with long climbs and short falls	
Defective sensor	**Defective oxygen sensor** Depending on the occurrence of the failure: A) O_2 increase OR B) O_2 decrease in the cabin A) O_2 increase: Rise into red zone, permanent inflow of O_2 B) O_2 decrease: drop below target state, flow meter reads zero	**Defective nitrogen sensor** Depending on the occurrence of the failure: A) pressure decrease OR B) pressure increase in the cabin A) Pressure increase: Pressure rises into red zone, permanent inflow of N_2 B) Pressure decrease: drop below target state, flow meter reads zero	--

3.4. Automatic Fault Identification and Recovery Agent (AFIRA)

AutoCAMS 2.0 can be operated in different modes, dependent on whether the fault identification and management task has to be performed by the operator alone (as described above) or is supported by an automated decision aid (AFIRA). The message display of AFIRA is shown in the lower right part of the interface. Three different variants of decision aids are available which differ with respect to the level of automation, i.e. in how far the human operator is kept actively involved in the overall process of fault management. Based on the model of types and levels of automation (Parasuraman, Sheridan & Wickens, 2000), the three aids are referred to as “Information Analysis Support”, “Action Selection Support”, and “Action Selection and Implementation Support”.

3.4.1. Information Analysis (IA) Support

In case of IA support, the decision aid provides a specific diagnosis for a given fault. Yet, any planning and implementation of necessary actions to manage the system and to initiate an appropriate repair action is left to the operator. The diagnosis is displayed on the AFIRA screen together with the timer showing how much time (seconds) has been elapsed since the identification of the fault. Both the diagnosis and the timer immediately appear after the Master Alarm (see Figure 1 and Figure 3).

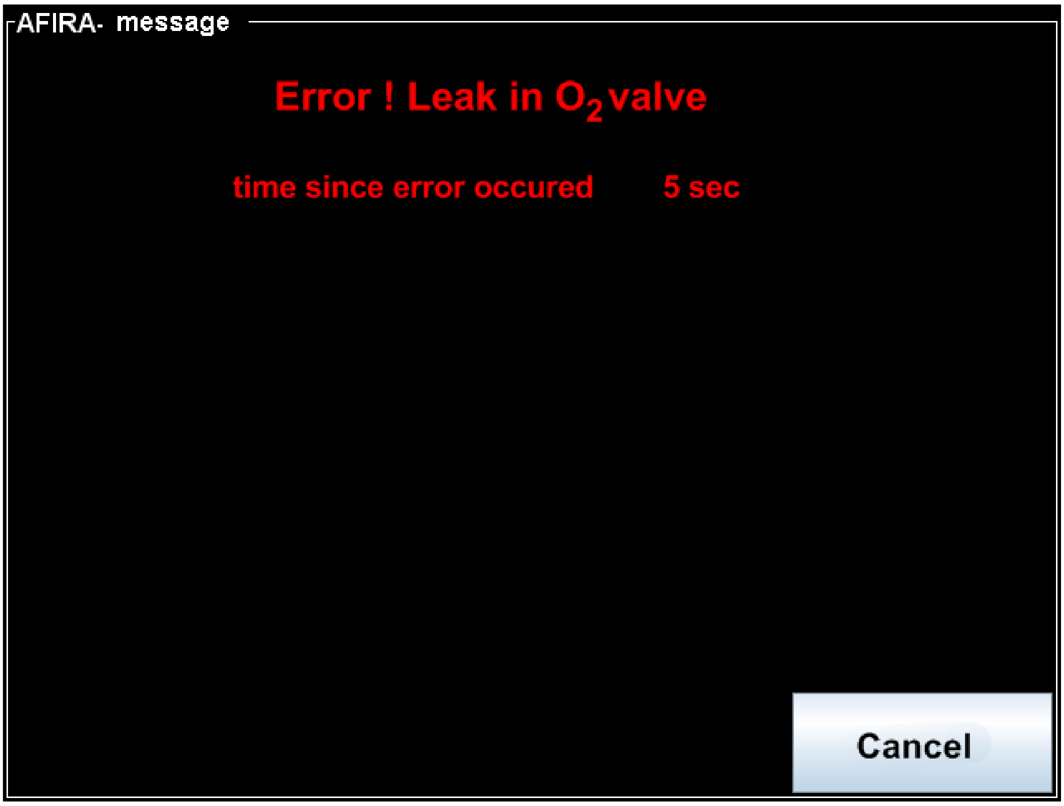

Figure 3 Information Analysis (IA) Support

3.4.2. Action Selection (AS) Support

In case of AS support, the decision aid does not only provide an automatically generated diagnosis of the given fault but also a list of appropriate actions recommended for a smooth management of the fault. This list of actions has to be implemented manually by the operator in the given sequence. Both the diagnosis and the list of recommended actions is displayed on the AFIRA screen immediately after the occurrence of a Master Alarm. An example of the screen is shown in Figure 4.

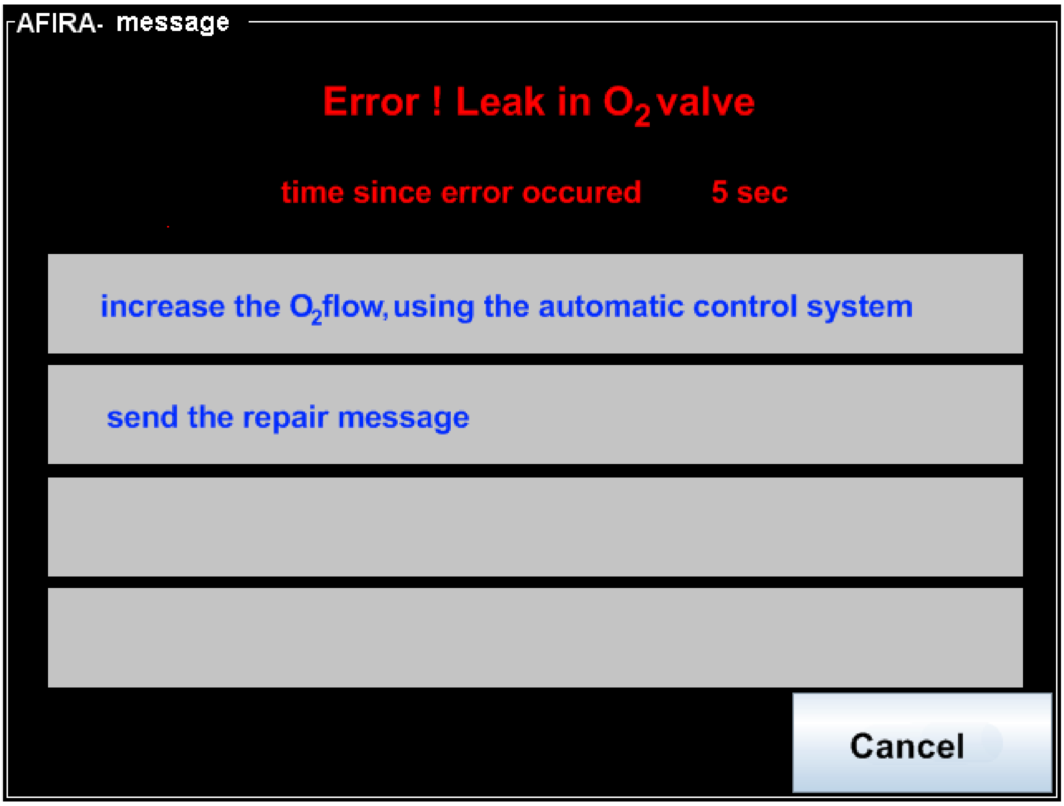

Figure 4 Action Selection (AS) Support

3.4.3. Action Selection and Implementation (AI) Support

AI support represents the highest level of automated support available. In this case, the decision aid provides support for information analysis (by providing a diagnosis for a given fault), action selection (by recommending steps of actions for fault management), as well as action implementation. The latter is implemented by AFIRA autonomously implementing all necessary actions if confirmed by the participant (click on “okay” button). During implementation of the different steps AFIRA provides feedback to the operator about which action currently is in progress and which actions have already been completed (see figure Figure 5 for an illustration of the screen).

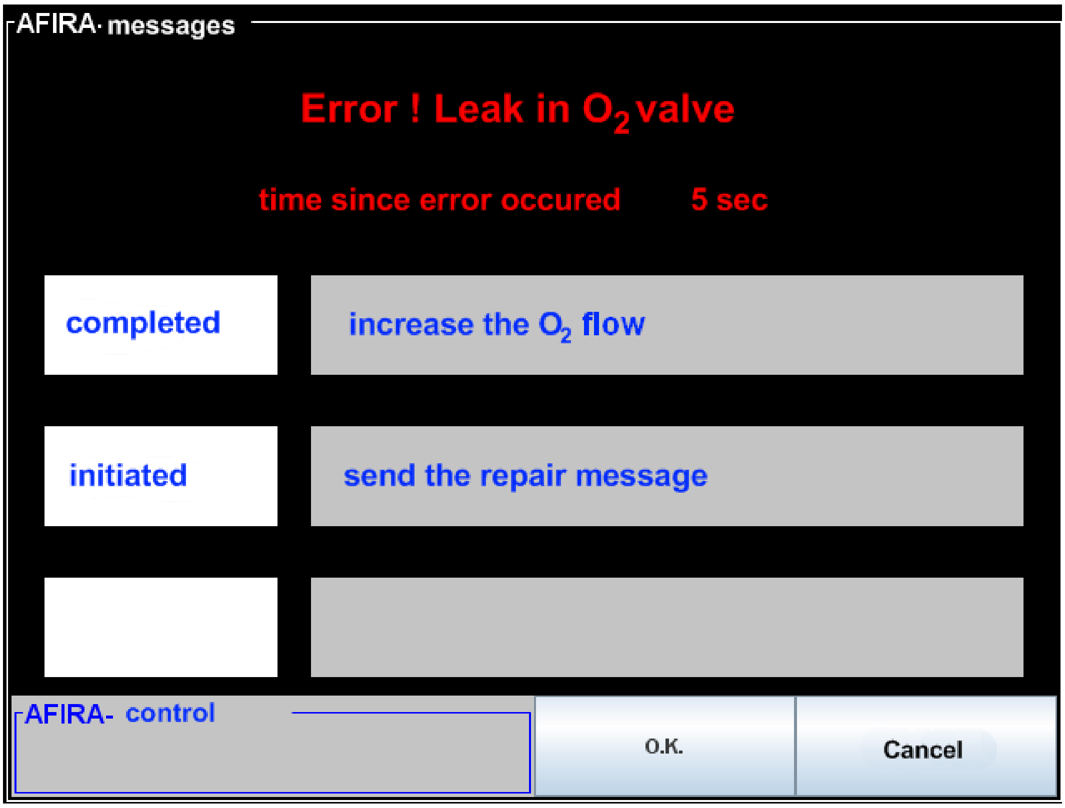

Figure 5 Selection and Implementation (AI) Support

3.5. Secondary Tasks

3.5.1. Probe Reaction Time ("connection check")

Besides the primary task, two secondary tasks have to be accomplished. The first one includes a simple reaction-time task which requires operators to click as fast as possible on a randomly appearing "connection check" button (see Figure 1: h)) which disappears again after a click has been recorded. This task is introduced to the participants as a necessary confirmation of a seamless communication between the ground control and the spacecraft. The command button with the connection check icon appears unpredictably with a mean inter-stimulus-interval one minute.

3.5.2. Prospective Memory ("CO_2 check and recording")

The second secondary task consists of a prospective memory task that requires the operator to check and record the level of CO_2 in the cabin at fixed intervals (every 60 s) without specific prompting. In order to fulfill the task, the operator needs to monitor the system clock (see Figure 1: g)) and to sample the CO_2 level at the appropriate time from the history graph. Then the sampled value must be typed into the corresponding text box (see Figure 1: i)) and saved (mouse-click on "save" button).

4. Data Storage

During runtime, all data regarding the state of the system are written into a log file with a frequency of 1 Hz. In addition, the operation of all buttons and switches is registered with a timely accuracy of about 20 to 150 milliseconds.

The log files for a given run can be found in the subdirectory "`logs`", the name of the log file consists of the IP address of the computer plus a running number and the extension `.txt`.

The first line of the log file contains information about the script (see section 5) that has been used for the block that generated this log file. A log file contains 14 columns, which are separated by semicolons. The first two columns contain time information. The other columns contain data representing the state of the different subsystems, as well as a time-related log of events. Data are always written into each of the 14 columns, even if the entry of only one column changed (e.g. an action performed by the operator). Table 2 gives an overview of the columns' contents including examples. Note that the decimal separator always is a colon, e.g. 30 = 30,0.

Note: A complete list of abbreviations used for coding events in columns 10 and 11 is available upon request.

Log files can be imported into a database in order to make any further analysis of the raw data more flexible and easy.

Table 2 Overview of log file contents

Column #	Description	Sample content
1	Time since start [seconds]	`1`
2	Time code of the Operating System [milliseconds]	`1220633126890`
3	Oxygen concentration in the cabin air [percent]	`19,868600`
4	Cabin pressure [bar]	`1,010427`
5	Temperature [degrees Celcius]	`21,040000`
6	Carbon dioxide concentration in the cabin air [percent]	`0,529409`
7	Humidity	`40,050000`
8	Tank level oxygen tank / Amount of oxygen supply left	`30000,000000`
9	Tank level nitrogen tank / Amount of nitrogen supply left	`30000,000000`
10	Event: which part of the system is involved	`ox_second`
11	Event: what happened	`Open`
12	Error phase (green: no system error, red: system error present)	`GREEN` `RED`
13	Running number	`0`
14	Event: who caused the log file entry, cams system or the operator	`CAMS_SYSTEM` `OPERATOR`

5. Scripting Experiments

The scripting of experiments with AutoCAMS 2.0 is supported by a graphical user interface that allows for a definition of different experimental runs. This interface represents a tab strip window with two different tabs (see Figure 6).

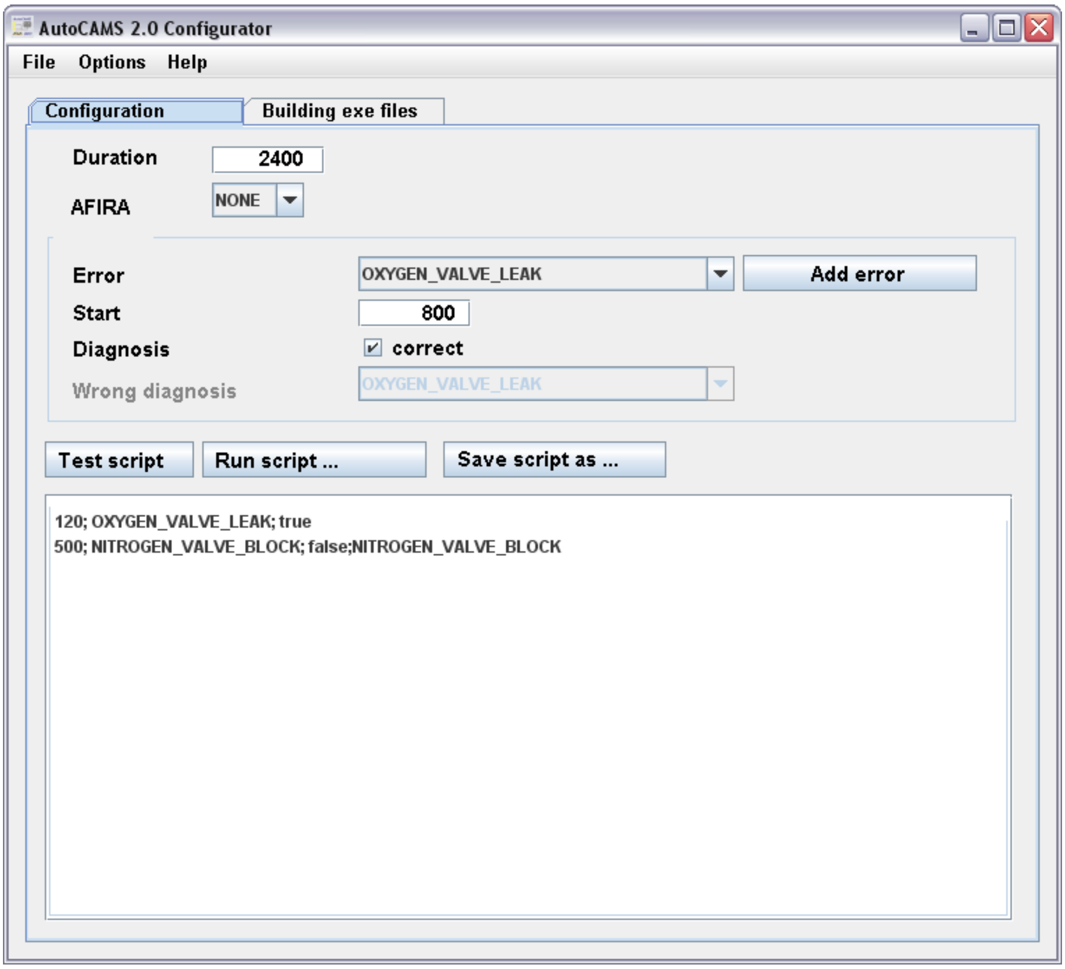

Figure 6 AutoCAMS 2.0 Configurator

5.1. Configuration Tab

For each experimental run ("block") the following basic characteristics can be set:

1. the duration of the block [s] and
2. the level of automated assistance (no, IA support, AS Support, or AI Support) provided within this block,
3. the sequence of system faults that shall occur, defined by type of fault, and time of occurrence ("start time"),

In case of AFIRA support it also can be defined for each fault whether the diagnosis provided by AFIRA shall be correct or wrong.

Scripts generated under the configuration tab can be tested immediately ("test script" button), and then saved ("save script as…" button). In addition, already existing scripts can be loaded and run ("run script…" button).

In order to run an experiment, AutoCAMS 2.0 could be run with one script as a command line parameter. For example if `AutoCAMS 2.0.exe` is stored in `D:\Programs` and the script is called `oxygen_errors.xml` (stored in `D:\Programs\scripts`) execute following command in Windows Command Prompt:

```
D:\Programs\AutoCAMS 2.0.exe scripts\oxygen_errors.xml
```

5.2. Building exe files

In order to run the specified experiment, the script need to get incorporated in an executable file of AutoCAMS 2.0. The window supports a file open dialog. The user can select script files for which specific an exe-file shall be generated. A message box provides feedback on the proper generation of exe-files.

6. Software and Implementation

The control system implemented by AutoCAMS 2.0 was extracted from a former version of the CAMS software which was developed with Microsoft Visual Basic Version 6.0. The control system for the cabin parameters is a simple on-off control.

The programming language for AutoCAMS 2.0 is Java SE Version 6. Java is used because a Java application can be executed on many platforms. Also Java supports most of object-oriented programming paradigm (OOP). It was expected that the OOP will help to get a flexible and extendable new CAMS software. The decision for developing a new version was encouraged by the termination of support for Visual Basic 6.0.

Programming started with the integrated development environment (IDE) Eclipse. Eclipse is used because it could be extended with plug-ins. E.g. it is possible to work on a database server out of Eclipse. Since Netbeans version 6.1, an IDE provided by Sun Microsystems, it is used as second IDE. Netbeans has advantages in building user interface and Model-Driven Software Development.

As version control system Subversion (SVN) is used. It is running on LINUX server. To access the server the svn protocol is tunneled trough SSH.

Design patterns were used to develop the AutoCAMS 2.0. For example the abstract class `ACamsSystem` extends `Observable`. Any other class could observe an `ACamsSystem`, which stands for control system and the cabin (according to observer pattern). The user interfaces from AutoCAMS 2.0 and ManuCAMS interact both with an object of `ACamsSystem` (according to Model View Presenter pattern).

The AutoCAMS 2.0 application can be controlled by script files. The file format is an XML-format. The behaviour of the subjects and the AutoCAMS 2.0 control system is written to comma separated value files.

Installation of AutoCAMS 2.0 is done by extracting a zip-file. One has to mind some conventions for folder structure. You must not delete the folders `buttons`, `config` and `logs`. The folder `logs` contains the log files. The folders `config` and `buttons` contain images for the user interface and configuration conventions.

7. ManuCAMS

The AutoCAMS 2.0 toolbox includes a specific component for the investigation of manual process control tasks – ManuCAMS. In the following sections, the ManuCAMS display and possible operator tasks, as well as the logging of data and scripting of experiments will be described.

7.1. The ManuCAMS display

The ManuCAMS display (see Figure 7) contains a permanently visible main display. This main display provides information about the process states of the system parameters in the two quadrants on the left hand side and a control panel for system interactions in the lower right quadrant.

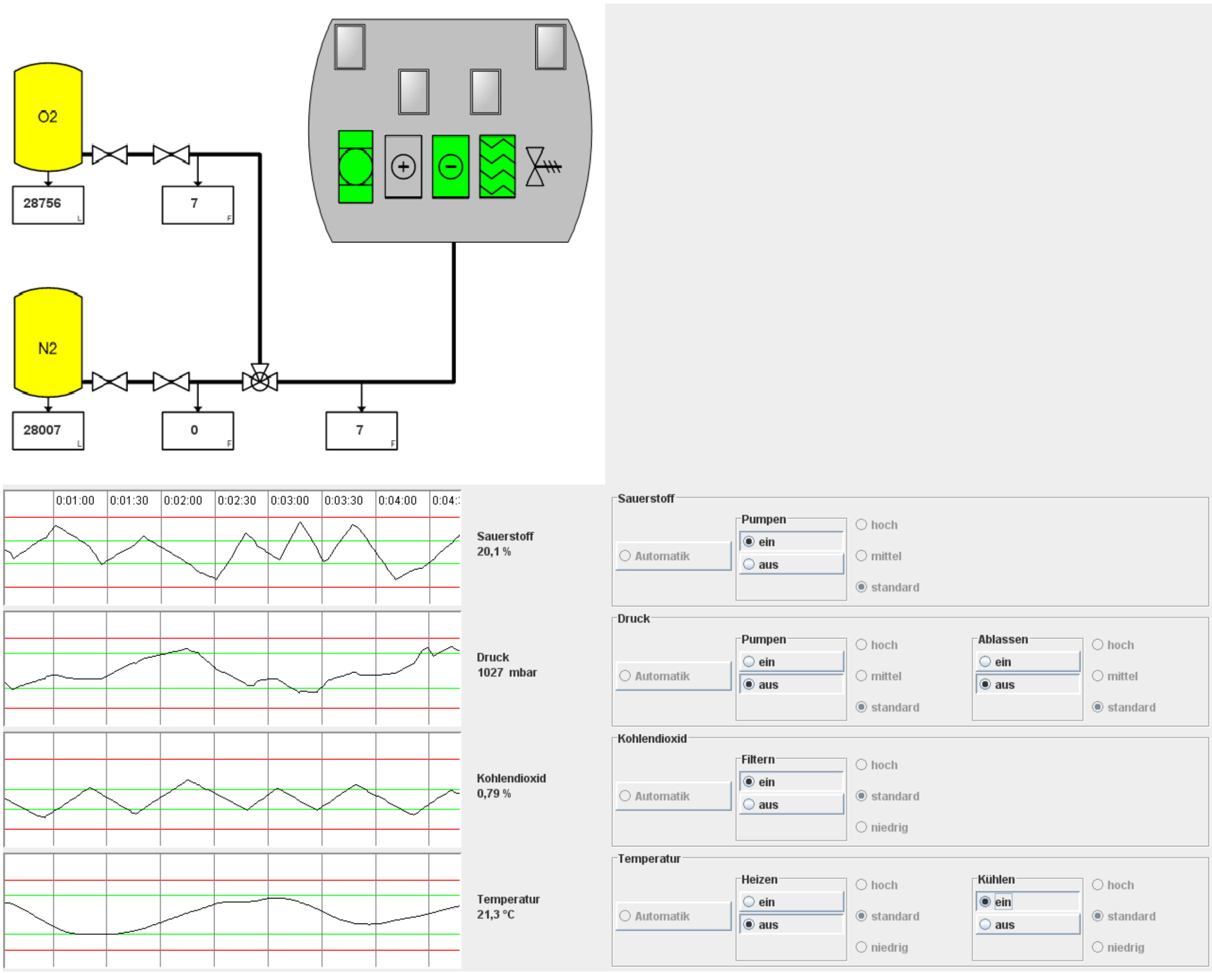

Figure 7 ManuCAMS display

The flow chart in the upper left quadrant displays a schematic overview of tanks, pumps and valves for the two subsystems oxygen and pressure as well as an operation mode display for the carbon dioxide filter, the heater, cooler, pressure valve and the dehumidifier.

The process trends in the lower left quadrant are permanently displayed for the parameters oxygen, pressure, carbon dioxide and temperature. Information about the progress of the concentrations of the four parameters can be accessed. For each parameter, oscillations within and outside of both the target and the normal operation range are displayed for a period of 240 seconds.

The control panels are located in the lower right quadrant. The user can manually control the four parameters oxygen, pressure, carbon dioxide and temperature by pointing-and-clicking at the corresponding buttons.

7.2. The manual process control tasks

Subjects are presented with the scenario that a permanent breakdown of the automatic control occurred. Accordingly, the operator's task is to manually control the affected parameters for a predefined period of time. Possible variations in the tasks concern the selection of the damaged parameters, ranging from one to four that have to be manually controlled. Further variations concern the period fixed for the manual control task.

7.3. Log File Data

During runtime, data regarding the state of the system is written into a log file with a frequency of 1 Hz. In addition, the operation of all buttons and switches is registered with a timely accuracy of about 20 to 150 milliseconds.

The log files can be found in the subdirectory `logs`, the name of the log file consists of a code and the extension `.txt`. The code can be defined in the script file before running ManuCAMS (see next section for details on the settings in the script file). If no name was set, the log file name will simply contain four digits (e.g. `0000.txt`).

A log file contains 16 columns, which are separated by semicolons. The first line of the log file contains the headers of the columns. Figure 8 shows the first couple of lines of a sample log file. Data are always written into each of the 16 columns, even if the entry of only one column changed (e.g. an action performed by the operator). Table 3 gives an overview of the columns' contents. Note that the decimal separator is a colon, e.g. ½ = 0,5.

```
fileName;time;timeEvent;event;eventNo;automaticSwitch;automaticSwitchNo;radioButtonSwitch;radioButtonSwitchNo;rel_oxygen;pressure;
ManuCAMS0005.txt;01.09.2008 12:33:38;1220265218651;timer;36;no_change;100;no_change_radio_buttons;102;19,900498;1,010025;21,000000
ManuCAMS0005.txt;01.09.2008 12:33:39;1220265219166;timer;36;no_change;100;no_change_radio_buttons;102;19,875380;1,010427;21,040000
ManuCAMS0005.txt;01.09.2008 12:33:39;1220265219666;timer;36;no_change;100;no_change_radio_buttons;102;19,845588;1,010829;21,080000
ManuCAMS0005.txt;01.09.2008 12:33:40;1220265220010;ox_man_on;1;no_change;100;no_change_radio_buttons;102;19,820828;1,011231;21,120
ManuCAMS0005.txt;01.09.2008 12:33:40;1220265220166;timer;36;no_change;100;no_change_radio_buttons;102;19,820828;1,011231;21,120000
ManuCAMS0005.txt;01.09.2008 12:33:40;1220265220666;timer;36;no_change;100;no_change_radio_buttons;102;19,863122;1,012411;21,148954
ManuCAMS0005.txt;01.09.2008 12:33:40;1220265220870;ox_man_off;2;no_change;100;no_change_radio_buttons;102;19,905086;1,013582;21,17
ManuCAMS0005.txt;01.09.2008 12:33:41;1220265221166;timer;36;no_change;100;no_change_radio_buttons;102;19,905086;1,013582;21,178056
ManuCAMS0005.txt;01.09.2008 12:33:41;1220265221307;ox_middle;4;no_change;100;no_change_radio_buttons;102;19,880853;1,013985;21,218
ManuCAMS0005.txt;01.09.2008 12:33:41;1220265221666;timer;36;no_change;100;no_change_radio_buttons;102;19,880853;1,013985;21,218056
ManuCAMS0005.txt;01.09.2008 12:33:42;1220265222166;timer;36;no_change;100;no_change_radio_buttons;102;19,852708;1,014387;21,258056
ManuCAMS0005.txt;01.09.2008 12:33:42;1220265222463;pressure_man_pump_on;7;no_change;100;no_change_radio_buttons;102;19,824242;1,01
```

Figure 8 ManuCAMS log file

Table 3 Overview of log file contents

Column #	Description	Sample content
1	Name of log file	`ManuCAMS0005.txt`
2	Time stamp	`01.09.2008 12:33:38`
3	Time code of the Operating System (milliseconds)	`1220265218651`
4	Event:: Operation of a switch or just logging system state	`ox_man_on` `timer`
5	Numerical Code of event given in column 4	`36`
6	Change of availability of automated control	`automaticTurnedOff`
7	Numerical code of event given in column 6	`101`
8	Change of availability of radio buttons	`no_change_radio_buttons`
9	Numerical code of event given in column 8	`102`
10	Oxygen in cabin air (%)	`19,900498`
11	Cabin pressure (bar)	`1,010025`
12	Temperature (°C)	`21,080000`
13	Carbon dioxide in cabin air (%)	`0,552422`
14	Humidity of cabin air (arbitrary unit)	`40,100000`
15	Amount of oxygen supply left (arbitrary unit)	`29982,444675`
16	Amount of nitrogen supply left (arbitrary unit)	`30000,000000`

7.4. Scripting Experiments

There are only few settings to make before running ManuCAMS. These settings must be defined in a xml-file, which has to be given as the first argument value when starting ManuCAMS from the command line. For example

```
C:\ManuCAMS> ManuCAMS.jar scripts\ManuCAMS.xml
```

will start ManuCAMS with the settings defined in `ManuCAMS.xml`, which is located in the directory `C:\ManuCAMS\scripts`. Figure 9 shows the content of a sample ManuCAMS script.

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <!--
3      Document    : ManuCAMS.xml
4      Created on : 01.09.2008
5      Author      : Stefan Röttger
6      Description: Sample ManuCAMS script
7  -->
8  <script>
9      <!-- nur den Wert von name aendern -->
10     <vpalias name="ManuCAMS" />
11     <duration seconds="120" />
12     <speed factor="0.5" />
13     <task start="20">
14         <interface>
15             <autobuttons ox="false" pressure="false" co2="false" temp="true"/>
16         </interface>
17     </task>
18     <radio_off start="20">
19         <radiobuttons ox="false" pressure="true" co2="true" temp="true" />
20     </radio_off>
21 </script>
```

Figure 9 ManuCams script file

Lines 3 to 6 of the script file can be used for purposes of documentation.

Settings relevant for the execution of ManuCAMS start in line 10. There, a code can be defined which will become the base of the log file name and will occur at the beginning of each line of the log file. A series of four zeros will be attached to the code. For example, the entry in line 10 will produce the log file `ManuCAMS0000.txt`. If a log file with this name already exists, data will be written into `ManuCAMS0001.txt` instead. If there is a file with this name as well, the number attached to the code will be increased until a unique file name is found. Leaving out the code in line 10 will result in file names consisting only of four digits and the extension `.txt`.

In line 11, the duration of the trial must be given in seconds. Note that this value must be multiplied with the speed factor in line 12 to obtain the actual duration of the trial. In the example above, ManuCAMS will terminate 60 seconds after it started.

Line 12 contains the speed factor of the simulation. The standard setting would be 1, meaning that each second, the value of the parameters change. As the displays on the interface are updated every second as well, a change in a parameter can be seen immediately. The value of 0.5 in the example above means that parameters need only half of the standard time to change, i.e. half a second. Thus, the speed of the parameters' dynamics has doubled. The changes of the parameters are greater and more abrupt. Setting this value to 2 would make the dynamics of the parameters slower – they would need twice the time of the standard setting for the same amount of change. The refresh rate of the display is unaffected by this setting, and remains at 1 Hz.

In experiments on manual process control, experimenters may want to disable the automated control at a predefined time after the start of the trial. In line 13, the number of seconds after the start of the trial after which automated control shall be disabled can be given. Please note that the actual time of this event calculates from the product of the values in lines 12 and 13. In the example in , the human operator must take over control 10 seconds after the start of ManuCAMS.

In line 15, it can be specified which of the four parameters lose automatic control. In the sample script file, automated control remains active for the temperature system only.

As for the automated control, the availability of the radio buttons for adjustments of the four systems of the life support system may be subject to experimental manipulations. Line 16 contains the time after which some or all of the radio buttons shall be disabled and line 17 the specification for each of the four systems. Again, the actual time of this event calculates from the figure in line 16 times the speed factor from line 12. In essence, the settings in the example will disable the radio buttons of the oxygen system 10 seconds after the start of the trial.

8. References

Brehmer, B. & Dörner, D. (1993). Experiments with computer-simulated micro-worlds: escaping both the narrow straits of the laboratory and the deep blue sea of the field study. *Computers in Human Behavior, 9,* 171-184.

Burkolter, D., Kluge, A., Schüler, K., Sauer, J. & Ritzmann, S. (2008). The predictive qualities of operator characteristics for process control performance: The influence of variables relating to cognition and personality. *Ergonomics*, in press.

Hockey, G. R. J., Sauer, J. & Wastell, D. G. (2007). Adaptability of Training in Simulated Process Control: Knowledge- Versus Rule-Based Guidance Under Task Changes and Environmental Stress. *Human Factors, 49*(1), 158-174.

Hockey, G.R.J. (1997). Compensatory control in the regulation of human performance under stress and high workload. *Biological Psychology, 45*, 73-93.

Hockey, G.R.J., Wastell, D.G. & Sauer, J. (1998). Effects of sleep deprivation and user interface on complex performance: A multilevel analysis of compensatory control. *Human Factors, 40*, 233-253.

Lorenz, B., Di Nocera, F., Röttger, S., & Parasuraman, R. (2002). Automated fault-management in a simulated spaceflight micro-world. *Aviation, Space, and Environmental Medicine, 73*, 886-897.

Parasuraman, R.J., Sheridan, T.B. & Wickens, C.D. (2000). A model of types and levels of human interaction with automation. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, 30*, 286-297.

Sauer, J., Hockey, G.R.J. & Wastell, D. (1999). Maintenance of complex performance during a 135-days spaceflight simulation. *Aviation, Space, and Environmental Medicine, 70*, 236-244.

Sauer, J., Wastell, D. G., Hockey, G. R. J. & Earle, F. (2003). Performance in a complex multiple-task environment during a laboratory-based simulation of occasional night work. *Human Factors, 45*(4), 657-669.

Sheridan, T.B. (1997). Supervisory control. In G. Salvendy (ed.), Handbook of human factors and ergonomics 2nd ed. (pp. 1295-1327). New York: Wiley.Appendix: Bibliography of Resarch with CAMS/AutoCAMS (1998-2008)

Appendix: Bibliography of Resarch with CAMS/AutoCAMS (1998-2008)

i. Research with CAMS (all versions)

Burkolter, D. , Kluge, A., Schüler, K., Sauer, J. & Ritzmann, S. (2007). Cognitive Requirement Analysis to derive Training Models for Controlling Complex Systems: In B. Hockey & D. de Waard (Eds.), Human Factors Issues in Complex System Performance (475-485). Shaker.

Burkolter, D., Kluge, A., Schüler, K., Sauer, J. & Ritzmann, S. (2008). The predictive qualities of operator characteristics for process control performance: The influence of variables relating to cognition and personality. *Ergonomics*, in press.

El-Samahy, E., Mahfouf, M., & Linkens, D.A. (2006). A closed-loop hybrid physiological model relating to subjects under physical stress. Artificial Intelligence in Medicine, 38, 257-274.

Hockey G.R.J., Wastell D. and Sauer J. (1998). Effects of sleep deprivation and user interface on complex performance: a multilevel analysis of compensatory control. *Human Factors*, *40*, 233-253.

Hockey, G.R.J. & Sauer, J. (1996). Cognitive fatigue and complex decision-making under prolonged isolation and confinement. In SL Bonting (ed.): Advances in Space Biology and Medicine, Vol. 5. Greenwich: JAI Press, pp.309-330

Hockey, G.R.J., Nickel, P., Roberts, A.C., Mahfouf, M. & Linkens, D.A. (2006). Implementing adaptive automation using on-line detection of high risk operator functional state. In ISSA Chemistry and Machine and System Safety Sections (eds.), Design Process and Human Factors Integration: Optimising Company Performance (1/004:1-13). Paris: INRS.

Hockey, G.R.J., Sauer, J.& Wastell, D.G. (2007). Adaptability of training in simulated process control: Comparison of knowledge- and rule-based guidance under task changes and environmental stress. Human Factors, 49, 158-174

Hockey, G.R.J., Wastell, D.G. & Sauer, J. (1998). Effects of sleep deprivation and user interface on complex performance: a multilevel analysis of compensatory control. Human Factors, 40, 233-53.

Kluge, A. & Burkolter, D. (2008). Trainings Principles for Process Control Tasks and Innovative Approaches to Training. Joint International Symposium of ISSNP2008/ CSEPC2008/ ISOFIC2008. Volume 2 (pp. 99-106). Harbin/China: Harbin University Press.

Mahfouf, M., El-Samahy, E., & Linkens, D.A. (2004). A new model for exploring the interactions between the cardiovascular and thermoregulatory systems under physical stress. 10th Mediterranean Conference on Medical and Biological Engineering (MEDICON 2004), Ischia, Naples.

Mahfouf, M., El-Samahy, E., & Linkens, D.A. (2005). A generic closed-loop model for the cardiovascular system, thermoregulation and brain activity under physical stress conditions. 16th IFAC World Congress, Prague.

Mahfouf, M., Zhang, J-H., Linkens, D.A., Nassef, A., Nickel, P., Hockey, G.R.J., & Roberts, A.C. (2007), Adaptive fuzzy approaches to modelling operator functional states in a human-machine process control system, IEEE International Conference on Fuzzy Systems, London, pp. 234-239.

Nassef, A., Ting, C-H., Mahfouf, M., Linkens, D.A., Nickel, P., Hockey, G.R.J. & Roberts, A.C. (2008). A new framework for real-time adaptive fuzzy monitoring and control for humans under psychophysiological stress, Biosignals 2008, International Conference on Bio-inspired Systems and Signal Processing, Funchal, Madeira (Portugal), pp. 320-325.

Nickel, P. & Hockey, G.R.J., (2005). On-line assessment of psychophysiological markers of operator functional state under dynamic task load. Psychophysiology, 42, S25.

Nickel, P., Hockey, G.R.J., Roberts, A.C., & Roberts, M.H. (2006). Markers of high risk operator functional state in adaptive control of process automation. Proceedings of the IEA 2006 congress, Maastricht, The Netherlands, pp. 0244:1-6.

Nickel, P., Roberts, A.C., & Hockey, G.R.J. (2006). Assessment of high risk operator functional state markers in dynamic systems - preliminary results and implications. In D. De Waard, K.A. Brookhuis & A. Toffetti (eds), Developments in Human Factors in Transportation, Design, and Evaluation. Maastricht: Shaker Publishing, pp. 271-284.

Nickel, P., Roberts, A.C., Roberts, M.H., & Hockey, G.R.J. (2007). Development of a cyclic loading method for the study of patterns of breakdown in complex performance under high load. In D. de Waard, G.R.J. Hockey, P. Nickel, & K. Brookhuis (eds.), Human Factors Issues in Complex System Performance. Maastricht: Shaker Publishing, pp. 325-338.

Nickel, P., Roberts, A.C., Roberts, M.H., & Hockey, G.R.J. (2008). Inter-individual differences in executive control activity during simulated process control: Comparison of performance and physiological patterns. In D. de Waard, F.O. Flemisch, B. Lorenz, H. Oberheid, and K.A. Brookhuis (eds.), Human Factors for Assistance and Automation. Maastricht: Shaker Publishing, pp. 313-323.

Rivolier, J., Bachelard, C., Cazes, G., Gaud, R., Le-Scanff, C., Rosnet, E., Novikov, M., Gushin, V., Efimov, V., Eskov K., Vinokhodova A., Hockey, G.R.J. & Sauer J. (1998). A methodological study in a space flight simulation context. European Review of Applied Psychology, 48, 211-212.

Roberts, M.H., Roberts, A.C., Nickel, P. & Hockey, G.R.J. (2007). Training operators in Micro-World simulations. In D. de Waard, G.R.J. Hockey, P. Nickel, & K.A. Brookhuis (eds.), Human Factors Issues in Complex System Performance. Maastricht: Shaker Publishing, pp. 491-496.

Sauer J. (2000). Prospective memory: a secondary task with promise. Applied Ergonomics, 31, 131-137.

Sauer J. (2003). CAMS as a tool for human factors research in spaceflight. Acta Astronautica, 54, 127-132.

Sauer J., Felsing T., Franke, H. and Rüttinger B. (2006). Cognitive diversity and team performance in a complex multiple task environment. Ergonomics, 49, 934-954.

Sauer J., Hockey G.R.J. and Wastell D. (1999). Maintenance of complex performance during a 135-day spaceflight simulation. Aviation, Space, and Environmental Medicine, 70, 236-244.

Sauer J., Hockey G.R.J. and Wastell D. (1999). Performance evaluation in analogue space environments: adaptation during an 8-month Antarctic wintering-over expedition. Aviation, Space, and Environmental Medicine, 70, 230-235.

Sauer J., Hockey G.R.J. and Wastell D.G. (2000). Effects of training on short- and long-term skill retention in a complex multiple-task environment. Ergonomics, 43, 2043-2064.

Sauer J., Wastell D. and Hockey G.R.J. (1999). Multiple-task performance on a computer-simulated life support system during a space mission simulation. Acta Astronautica, 44, 43-52.

Sauer J., Wastell D. and Hockey G.R.J. (2000). A conceptual framework for designing micro-worlds for complex work domains: a case study of the Cabin Air Management System. Computers in Human Behavior, 16, 45-58.

Sauer J., Wastell D., Hockey G.R.J. and Earle F. (2003). Performance in a complex multiple-task environment during a laboratory-based simulation of occasional night work. Human Factors, 45, 657-669.

Sauer, J. (2000). The use of micro-worlds for human factors research in extended spaceflight. Acta Astronautica, 46, 37-45.

Sauer, J., Burkolter, D., Kluge, A., Ritzmann, S. and Schüler, K. (2008). The effects of heuristic rule training on operator performance in a simulated process control environment. Ergonomics, 51, 953-967.

Sauer, J., Wastell, D.G. & Hockey, G.R.J. (1997). Skill maintenance in extended spaceflight: A human factors analysis of space and analogue work environments. Acta Astronautica, 39, 579-587.

Ting, C-H., Mahfouf, M., Linkens, D.A., Nassef, A,. Nickel, P., Roberts, A.C., Roberts, M.H., & Hockey, G.R.J. (2008), Real-time adaptive automation for performance enhancement of operators in a human-machine system. 16th Mediterranean Conference on Control and Automation, Ajaccio, Corsica (France), pp. 552-557.

Ting, C-H., Mahfouf, M., Linkens, D.A., Nassef, A., Nickel, P., Hockey, G.R.J., & Roberts, A.C. (2007), Towards on-line supervision and control of operational functional state (OFS) for subjects under mental stress. IEEE/NIH Life Science Systems and Applications Workshop (LISSA 2007). Bethesda, MD (USA), pp. 77-80.

Zhang, J., Nassef, A., Mahfouf, M., Linkens, D.A., Hockey, G.R.J., Nickel, P., & Roberts, A.C. (2006). Modelling and spectral analysis of HRV of operators subjected to physical and mental workloads. In IFAC (Ed.). Proceedings of the 6th International Federation of Automatic Control Symposium on Modelling and Control in Biomedical Systems (including Biological Systems). Oxford: Elsevier, pp. 189-194. [ISBN 0-08-044530-6]

Zhang, J.; Mahfouf, M.; Linkens, D.A.; Nickel, P.; Hockey, G.R.J. & Roberts, A.C. (2007). Adaptive Fuzzy Model of Operator Functional State in Human-Machine System: A Preliminary Study. In

J.W. Gardner (ed.), Biomedical Engineering – BioMED 2007 (Proceedings of the 5th International Conference on Biomedical Engineering, February 14-16, 2007, Innsbruck, Austria). Calgary: ACTA Press, 555-017: 118-123. [ISBN 978-0-88986-648-5]

ii. Research with AutoCAMS

Bahner, J.E., Elepfandt, M. & Manzey, D (2008). Misuse of diagnostic aids in process control: The effects of automation misses on complacency and automation bias. *Proceedings of the 52nd Meeting of the Human Factors and Ergonomics Society, New York, September 2008*. Santa Monica: HFES.

Bahner, J.E., Hüper, A.-D. & Manzey, D. (2006). Complacency in automated fault management: How to keep operators alert towards possible failures of automated aids. In R.N. Pikaar, E.A.P. Koningsveld & P.J.M. Settels (Eds.), *Proceedings of the 16th World Congress on Ergonomics*. Amsterdam: Elsevier.

Bahner, J.E., Hüper, A.-D. & Manzey, D. (2008). Misuse of automated decision aids: Complacency, automation bias and the impact of training experiences. *International Journal of Human-Computer Studies, 66*, 688-699.

Elepfandt, M.F., Bahner, J.E., Manzey, D. (2007). Automation Bias und Complacency: Der Einfluss von Systemausfällen im Training auf die Überwachung einer Automation. In M. Rötting & G. Wozny (Hrsg.), *Prospektive Gestaltung von Mensch-Technik-Interaktion. 7. Berliner Werkstatt Mensch-Maschine-Systeme* (S. 323-326). Düsseldorf: VDI Verlag.

Lorenz, B. (2002). Automatisches Fehlermanagement in einer komplexen Prozesskontrollaufgabe: Implikationen für die Aufrechterhaltung des Situationsbewusstseins. In: M. Grandt & K.-P. Gärtner (Hrsg.), *Situation Awareness in der Fahrzeug- und Prozessführung. DGLR-Bericht 2002-04* (S. 11-28). Bonn: Deutsche Gesellschaft für Luft- und Raumfahrt.

Lorenz, B., Di Nocera, F., Röttger, S., & Parasuraman, R. (2002). Automated fault-management in a simulated spaceflight micro-world. *Aviation, Space, and Environmental Medicine, 73*, 886-897.

Lorenz, B., DiNocera, F. & Parasuraman, R. (2002). Display integration enhances information sampling ad decision-making in automated fault management in a simulated spaceflight micro-world. *Proceedings of the 46th Annual Meeting of the Human Factors and Ergonomics Soeciety, Balitmore, September 2002*. Santa Monica. HFES.

Lorenz, B., DiNocera, F. & Parasuraman, R. (2002). Varying types and levels of automation in the support of dynamic fault management: An analysis of performance costs and benefits. In: D. de Waard, K.A. Brookhuis, C.M. Weickert & A. Toffetti (eds.), *Human Factors in transportation, communication, health, and the workplace*. Maastricht: Shaker.

Manzey, D., Bahner, E.J. & Hüper, A.-D. (2006). Misuse of automated aids in process control: Complacency, automation bias, and possible training interventions. *Proceedings of the 50th Annual Meeting of the Human Factors and Ergonomics Society, San Francisco, 16-20 October, 2006*. Santa Monica: HFES.

Röttger, S., Bali, K. & Manzey, D. (2008). Impact of automated decision aids on performance, operator behaviour and workload in a simulated supervisory control task. *Ergonomics*, in press.

Röttger, S., Bali, K., & Manzey, D. (2007). Do cognitive assistance systems reduce operator workload? In D. d. Waard, G. R. J. Hockey, P. Nickel & K. A. Brookhuis (Eds.), *Human Factors Issues in Complex System Performance* (pp. 351-360). Maastricht: Shaker Publishing.

iii. Research with AutoCAMS 2.0

Manzey, D., Reichenbach, J. & Onnasch, L. (2008). Performance-consequences of automated aids in supervisory control: The impact of function allocation. Proceedings of the 52nd Meeting of the Human Factors and Ergonomics Society, New York, September 2008. Santa Monica: HFES.

iv. Research with ManuCAMS

Röttger, S., Klostermann, A. & Manzey, D. (2007). An information theory-based approach to measure orderliness of control behavior. In M. Rötting & G. Wozny (Hrsg.), Prospektive Gestaltung von Mensch-Technik-Interaktion. 7. Berliner Werkstatt Mensch-Maschine-Systeme (S. 131-136). Düsseldorf: VDI Verlag.