	HoliDes **Holi**stic Human Factors **Des**ign of Adaptive Cooperative Human-Machine Systems	

D7.5 - Modelled and Model-based Analysis of the Aeronautics AdCoS and HF-RTP Requirements Definition Update (Feedback)

Project Number:	332933
Classification:	Public
Work Package(s):	WP7
Milestone:	M3
Document Version:	V1.0
Issue Date:	30.06.2015
Document Timescale:	Project Start Date: October 1, 2013
Start of the Document:	Month 19
Final version due:	Month 21
Deliverable Overview:	**Main document:** Modelled and Model-based Analysis of the Aeronautics AdCoS and HF-RTP Requirements Definition Update – Public **Annex I:** Feedback towards MTTs – Confidential **Annex II:** Common Annex to D6.5, D7.5, D8.5, D9.5 - Public
Compiled by:	Frank Rister - TRS
Authors:	Frank Rister - TRS Jan-Patrick Osterloh - OFF Zdenek Moravek - HON
Reviewers:	Fabio Tango - CRF Elisa Landini - REL
Technical Approval:	Jens Gärtner, Airbus Group Innovations
Issue Authorisation:	Sebastian Feuerstack, OFF

DISTRIBUTION LIST		
Copy type[1]	Company and Location	Recipient
T	HoliDes Consortium	all HoliDes Partners
D	Artemis JU	Patrick Vandenberghe

[1] Copy types: E=Email, C=Controlled copy (paper), D=electronic copy on Disk or other medium, T=Team site (AjaXplorer)

RECORD OF REVISION		
Date (DD.MM.JJ)	**Status Description**	**Author**
29.04.2015	Initial Version	FR
29.05.2015	Input HON	ZM
30.05.2015	Input OFF	JPO
04.06.2015	Input TRS	FR
08.06.2015	Final Check for Review	JPO
23.06.2015	Integration of Review Feedback	FR, ZM, JPO

Table of Contents

List of figures

1 Glossary

AC	-	Aircraft
ATO	-	Approved Training Organisation
CO	-	Confidential
DivA	-	Diversion Assistant
EATT	-	Enhanced Aircraft Transition Training
FCU	-	Flight Control Unit (A320 Autopilot HMI)
FFS	-	Full Flight Simulator
FSTD	-	Flight Simulation Training Device
HF-RTP	-	Human Factors Reference Technology Platform
HMI	-	Human Machine Interface
ILS	-	Instrument Landing System
LFT	-	Lufthansa Flight Training
MCP	-	Mode Control Panel (B737 Autopilot HMI)
MTT	-	Methods, Techniques and Tools
SOP	-	Standard Operating Procedure(s)
UML	-	Unified Modelling Language

2 Introduction

This report describes the results of using the HF-RTP and methodology with new techniques and tools from WP2, WP4, and WP5 to model the AdCoS and to perform model-based analysis. Feedback will be described for WP1-5. The deliverable is furthermore supplemented with a common annex, providing a cross-domain introduction and conclusion to AdCoS modelling (Annex II).

The PU Part will contain the general conclusions about the use of the models and requirements. The CO Part will contain all necessary details and information related to how the AdCoS and HF-RTP will be implemented in the project.

2.1 Objective of the document

This document focuses on the MTTs employed during the modelling of the different AdCoS and AdCoS modules and the models and analysis results obtained during the development process.

In contrast to previous deliverables, mainly focussing on **what** MTTs have been or are planned to be used, this deliverable shall report on how these MTTs were used.

3 AdCoS modelling

3.1 DivA AdCoS

3.1.1 Description of the AdCoS

Diversion assistant (DivA) is an application that will assist the pilot in the integration and evaluation of various types of information in situations when original destination and alternates become unavailable due to airport closure and/or weather restrictions. In such situations, pilots need to consider various sources of information – for aircraft performance, overall weather situation, parameters for airports in reach etc. Based on the updated information, pilots are supposed to evaluate options for available airports and negotiate approach and landing for the selected one.

DivA will support pilots in the task by monitoring the relevant information and assuring its currency and it will perform necessary calculations following aerodynamics and task-related models to create scores for available diversion options. DivA displays the results for pilots to make decision.

DivA will take into account cockpit conditions, workload, or fatigue of the pilot, i.e. the calculations and HMI will be adapted to actual situation. The state of the pilot will be determined from physiological properties measured on the pilots during the operations. The physiological data will be processed by appropriate models, see below, to assess the respective state. DivA will adapt to the fluctuating properties of the environment as well as to the state of the pilot. Therefore the system should be able to detect and classify the situation and to apply relevant actions.

3.1.1.1 Operational definition

The intended function of the diversion assistant is to provide the user with all necessary information in order to choose an appropriate alternate airport. The following areas of information will be provided

- Aircraft performance, specifically fuel calculations, range predictions and optimal flight profile.
- Weather, both en-route and at airports being considered. Actual situation as well as extrapolations will be provided.
- Functional limits en-route and at airports will be considered. This includes status of airways and airspaces and actual limits in available
- User preferences for both, the pilot based on his experience and predefined preferences and the airline considering contracts and support for each airport.

Some degree of automation in the data pre-arranging process will be involved, but the final decision is expected to be made by the crew. The automation will address

- Flight performance and route optimization
- Landing performance for given runway and its conditions
- Combination of heterogeneous aspects into a composite score

3.1.1.2 The environment of the AdCoS

DivA is a cockpit class 2 application hosted by EFB device. As a consequence, it has limited access to aircraft dynamic and performance data and it must rely on its own models. It has also limited options for interaction with the pilots due to regulations.

Controlled entity

DivA controls currency of relevant information and arranges its update through aircraft information system. It also controls relevance of airports close to current aircraft performance with respect to the diversion.

Operator of the AdCoS

There can be several different personas to operate the system:

- Kuba is a dispatcher for small airline. For sake of economic reasons he is not supposed to provide real time support for pilots in the air, but he may manage supporting applications such as the diversion assistant.
- Ivan is a business jet pilot flying customers to diverse destinations. In flying he relies mostly on his own judgment, but often he needs to respect requirements of his customers. Therefore his goal is often to be on time at required destination regardless of price.
- Dominik is a pilot of small charter airline. He flies to exotic destinations, often to smaller airports. The destinations may change during the season depending on contract the airline gets. Dominik flies with little support of his airline and the price of flight is an important factor in his decisions.
- Karel is a pilot of larger airline and while in the air he enjoys full support of his airline. Price of flight is important, but so is the comfort of passengers who expect a certain level of service from the airline.

External environment

The external environment consists of cockpit, physical environment of flight, geographical position of diversion trigger and the weather situation. The external environment influences the system at different levels:

- Usefulness – the shorter the time pilots have to make a decision the more they will appreciate support.
- Reliability of information may depend on aspects such as fluctuating dynamical parameters, unstable weather, out-dated information. As result the accuracy of outcome from the system may be affected.
- Usability – the system is a tablet-like application using touch modality for operations. Aspects such as vibrations in turbulences may render the system difficult to operate.

3.1.1.3 Modelling techniques employed

Hierarchical task modelling - modelling the sequence of activities and pieces of information that the pilots use in re-planning the destination. The technique also allows for estimating how long each step in the activity usually takes and how difficult the pilot finds the step. With all this information, the scope of diversion assistant was defined in order to address the most relevant and still feasible aspects of the task.

UML modelling first described what in high level the system will do when it is invoked for assisting the pilots. As the DivA re-uses several components for specific subtasks (such as calculation of flight profile), UML provides means for modelling interaction and data interfaces among the components.

A number of physical and mathematical models were used for specific calculations, however these models are parts of re-used components and are thus external to DivA.

With respect to the information about pilot's mental state, namely his fatigue, two approaches are used. In the first step, psychological models of the fatigue are used and extended towards a definition of a number of measurable quantities – physiological signals. Based on this model selected methods will be used to measure fatigue markers. Later a classifier model will be built to combine the markers into a composite fatigue score using supervised learning methods. The two models are built in cooperation with WP2 and WP3.

3.1.2 The models

3.1.2.1 Task analysis

The diversion is a complex task that covers all types of pilot's behaviours – aviation, navigation and communication. Despite there is a general sequence of steps in diversion and hence a general task model, there are variants of the model:

- Flight phase in which diversion is initiated, influences priorities of steps and thus their extent and order.
- Airlines provide variable level of support for their pilots from as little as 'friend on phone' to a complete re-planning of the route.

As a result, some detailed models can leave out large blocks of the general model and replace them with simple activities.
A high level model is depicted in Figure 1. Obviously, the most demanding part of the task is to get updated information and create accurate mental model of the situation. The amount of information and its temporal validity makes it very difficult and errors such as oversimplification, incorrect judgment or missed information can easily take place as can be illustrate on one of numerous examples:

> *On 10th May 2010, after two unsuccessful approaches at Alicante airport due to adverse weather, a crew of Ryanair Aircraft B 737-800 flying from Stansted airport diverted to Valencia. On the way there, they stated urgency (PAN-PAN) due to the fact they were below the final fuel reserve and then during the approach they even stated emergency (MAY DAY). Luckily, they landed at the runway uneventfully. The follow-up investigation by Spanish accident investigation agency revealed that the incident partially happened due to "…inadequate decision making … in their choice of an alternate airport." (CIAIAC Report, 2010)*

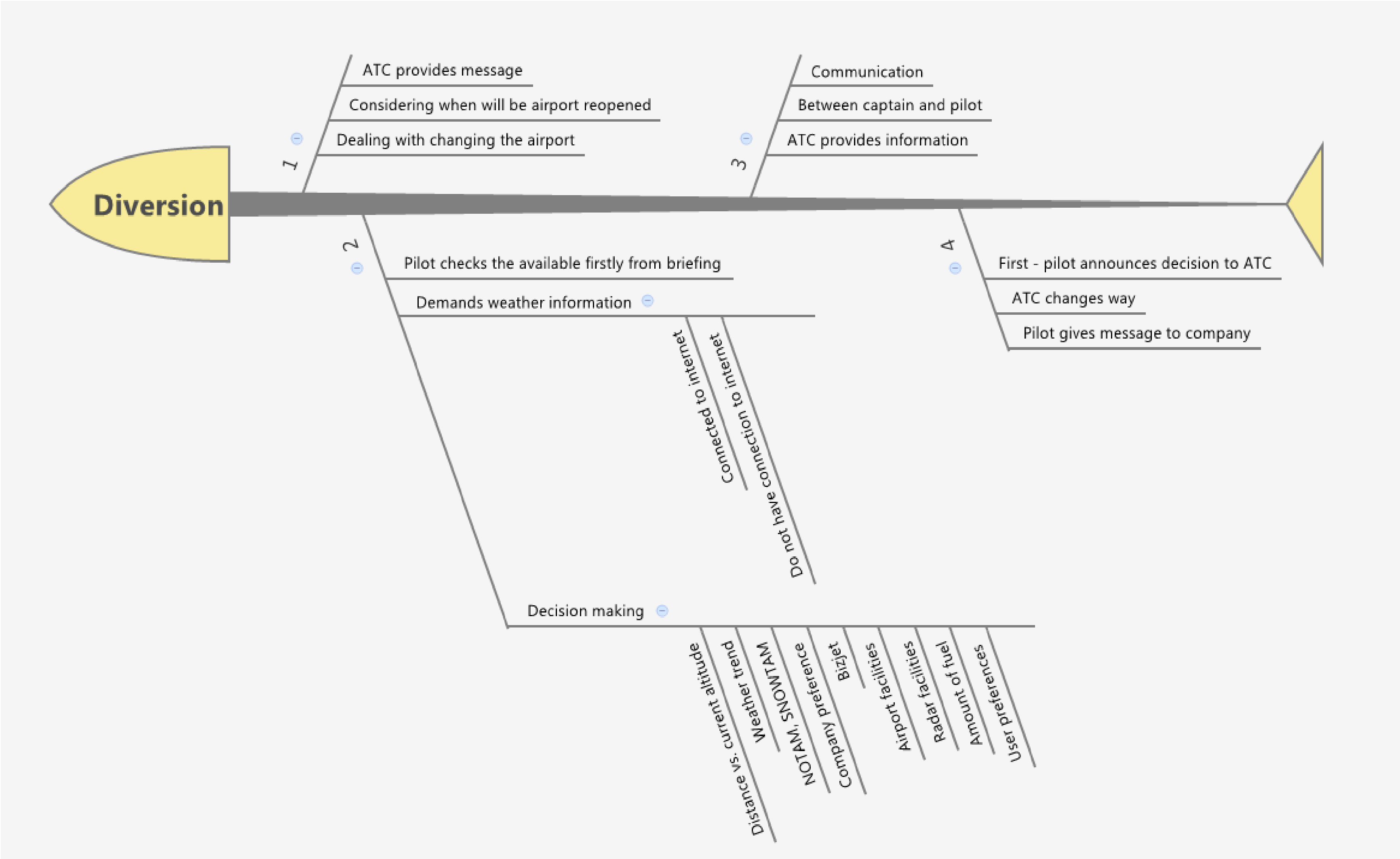

Figure 1: High level task model of the aircraft diversion.

3.1.2.2 UML component architecture

Based on the task analysis, the diversion assistant can support the crew in automating the following four activities:

- Update and interpretation of various information sources – weather, flight environment, airport status, pilot state.
- Background calculations for aircraft performance and for development of weather situation.
- Global calculation of airport score – integration of information from the information sources and from the calculators and its processing via a situation model.
- Creating a reliable and up-to-date situation model with clear display of diversion options and their score.

With use of UML, relevant workflows and dataflows are created. The flows show, which information appears at each step of the diversion procedure and who is producer or consumer of that information. Finally, classification and grouping of data and services allows for definition of standalone components with defined mutual interactions. Component model created for diversion assistant is show in Figure 2.

The model clearly separates components that handle background data and provides basic services for interpretation of data – such as its relevance to the flight, near future evolution of the data, availability or accuracy. On top of them, a data centre component creates unified access to the data.

Other components provide simulation models, such as aircraft performance or a user model and offer various types of calculations using those models. These components form calculation core of the diversion assistant.

Finally, there are components designed to interact with the user – either to provide interface for tuning the behaviour of the application or displaying the results of diversion evaluation.

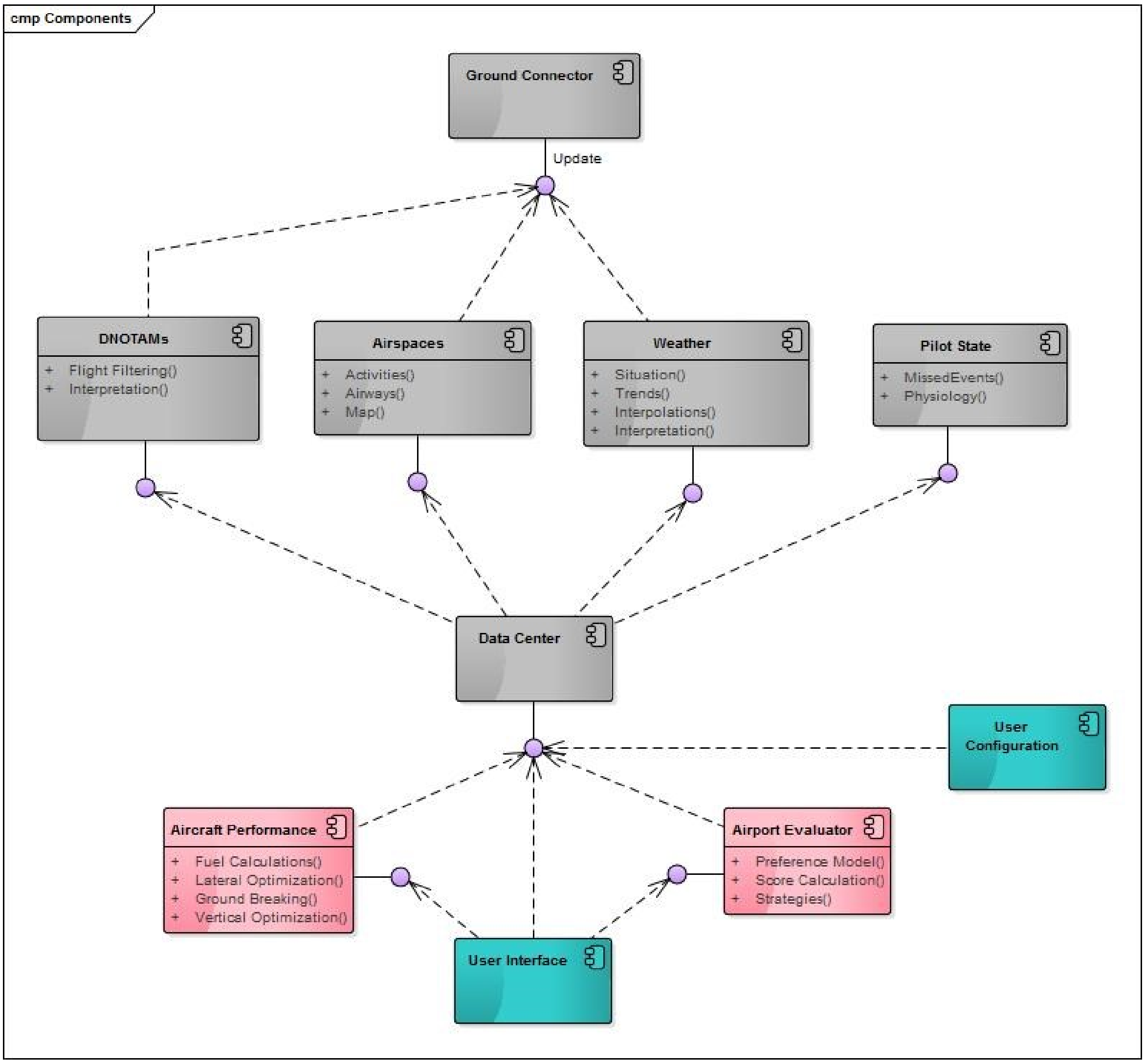

Figure 2: UML component model for diversion assistant.

3.1.2.3 Pilot fatigue model

The performance of human operator is influenced by his engagement in the activity, which heavily depends on operator's cognitive state. It has been demonstrated that fatigue can cause break-down of the performance and therefore early detection can prevent from critical situations such as attention tunnelling, information missing or misinterpretation.

However, the fatigue is a complex psychological phenomenon and its effect on performance has strong subjective component. There is also a strong temporal component as the fatigue develops in time due to varying level of activity and subjective ability of adaptation influences the speed of fatigue onset.

There is a number of models explaining the origin, character and consequences of the fatigue. The hypothesis of compensatory control will be used as the starting point for fatigue modelling in WP7.

The compensatory control hypothesis can be explained by a two level model where the lower level corresponds to the execution of familiar tasks guided by directed attention whereas the upper level refers to the intervention of executive control. The executive control is conceived as a limited resource for controlled processing especially for activities such as planning, problem solving, and task scheduling. The prolonged (over)use of this mechanism is the cause for the phenomena of mental fatigue.

To model the situation, UML state diagram is used. The variables in the model are the fatigue and the most prominent factors that affect the fatigue – the ability of adaptation as negative feedback decreasing effects of fatigue, the stress as positive feedback, and the arousal as indifferent factor. The states represent cognitive activities that are performed in task execution and which may affect the model variables. Transitions between states are described by transition functions that take into account effects of time, external environment etc. The model is shown in Figure 3, the red components form the original compensatory control model.

The transitions functions describe how situation influences the variables of the model. They take into account time on task and history of previous tasks to address long term development of fatigue. The form of the transition functions will be specified by further modelling and experimental work.

The function of the model is to assess level of fatigue in real time based on directly measurable information. Inputs from both the task analysis and physiological markers will drive the model in time and will determine how often the supervisory controller is used and thus how the fatigue builds up.

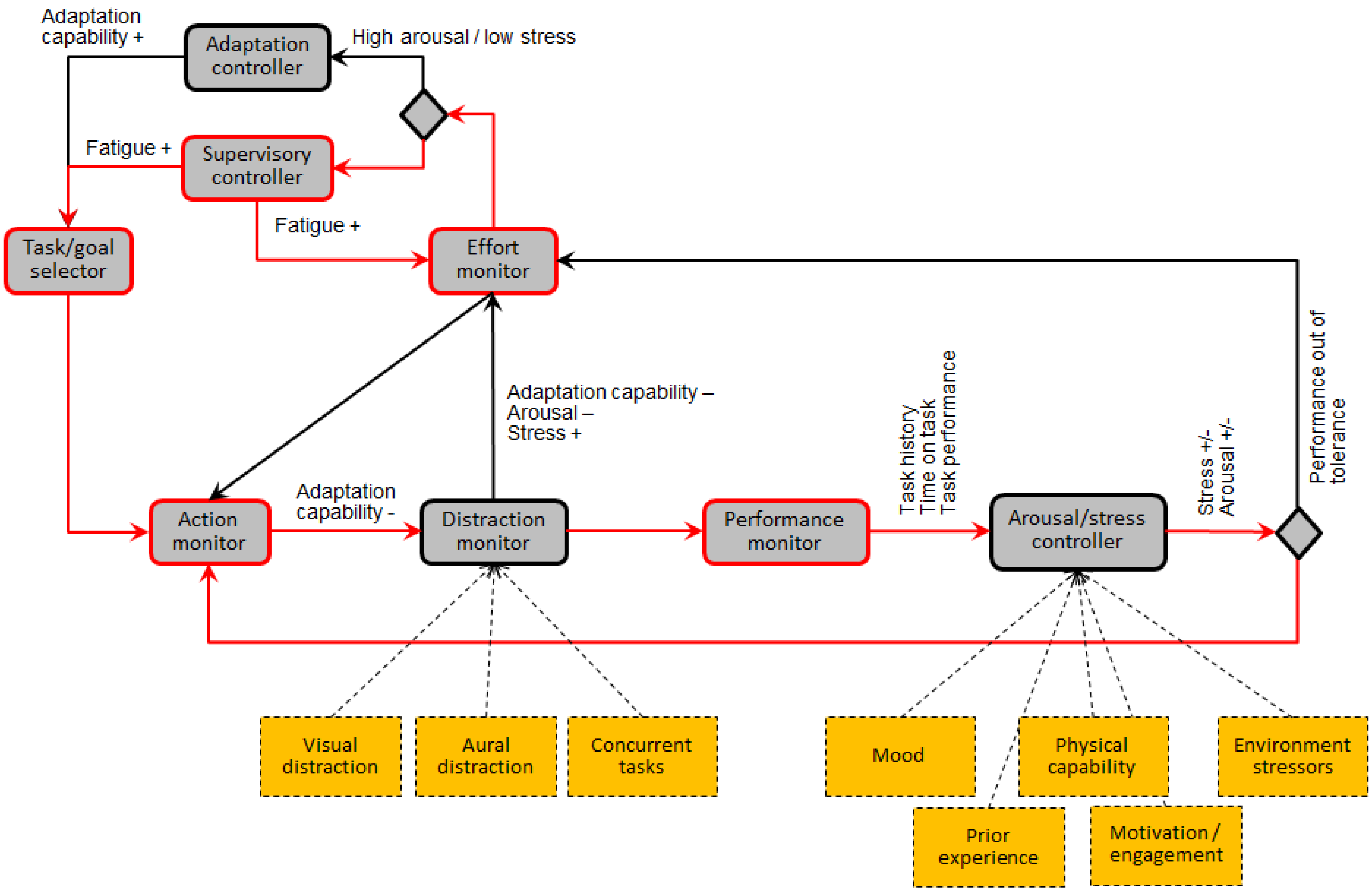

Figure 3: State diagram of compensatory control model extended for temporal properties. In yellow, measurable properties that influence the behaviour of the model.

3.2 EATT (Training AdCoS)

3.2.1 Description of the AdCoS

To date, pilot transition training (training to fly a different aircraft type, after having been experienced on a former aircraft), provides the same training contents and effort as for new, inexperienced pilots. This "one-size-fits-all-style" of training can neither be regarded efficient nor contributing to safety. Objective of the EATT (Enhanced Adaptive Transition Training) is to overcome this problem, by providing an adaptive, model-based training tool.

3.2.1.1 Operational definition

EATT is an adaptive, model-based transition training tool that accounts for the trainees' previous experience. Based on cognitive task models, learning theory-based models and system comparisons, EATT provides training that emphasizes the differences between two aircraft types (e.g. B737 and A320), identifies areas of higher training requirements (e.g. when procedures significantly differ or only "mask" to be identical), areas of less effort and also allows for an enhanced learning curve by redistributing training tasks to an optimum level.

EATT's goal is to enhance safety, make training success-rates more predictive, simulator training planning and training syllabus generation more intuitive and less prone to critical and dangerous time-planning or neglecting of training demands.

In addition to that, EATT will be also usable for trainers to manage syllabi for non-transition trainings.

3.2.1.2 The environment of the AdCoS

The following sections describe the environment EATT will interact with.

Operator of the AdCoS

EATT will be operated by certificated flight instructors, who are usually active pilots licensed to train pilots on a certain aircraft type. Training usually consists of a theoretical part and a practical part in full flight simulators (FFS). Each part is concluded with a test. Trainings are ordered for multiple pilots by an airline. Before the training starts, an instructor has to specify a training syllabus in accordance to regulations as well as customer needs (e.g. number of FFS sessions and with that budget, usable simulators, and deadlines). A syllabus describes for each planned session, especially the FFS session, what the content of the session will be (e.g. session 1: cockpit preparation, engine start; session 2: cockpit preparation, engine start, taxi and take-off; …), including instructions on how to setup the simulator.
Currently the syllabi are specified with word. EATT can be used by the instructor to specify and modify the syllabus, and gets support on

consistency, feasibility, and learning methods by EATT. Input for EATT is the aircrafts Standard Operation Procedures (SOPs) as well as the training model (see Deliverables D2.4 and D2.5 for task models (SOPs) and Training Model). Output is a training syllabus.

3.2.1.3 Modelling techniques employed

Main modelling technique that has been used for EATT is task modelling, because very detailed models of the standard operation procedures (SOPs) of the A320 and B737 are needed. The models are compared in order to find differences between these two aircraft types, so that the transition training from B737 to A320 can be optimized according to the differences (and also similarities). For the modelling of the SOPs, the WP2 tool MagicPED has been used, which allows modelling hierarchical tasks as well as modelling rule-based on a lower level.
In addition to the tasks and rules, the systems of both aircrafts have been modelled with UML class diagram, also in MagicPED.

3.2.1.4 Input to the modelling process from other work packages

Figure 4 shows the flow between the models and tools from WP2 and WP7. The models (Systems Aircraft A/B and Procedures Aircraft A/B) are specified in WP7 with the tool MagicPED as Aircraft A/B MODEL. These models are used by the Model Comparator to compare the similarities and differences in the models. This information, plus the training models (Learning Models) from WP2 are then used by the Training manager to generate the new syllabus. The first version of the syllabus, which is mainly handmade, but according to the model output, is now in the first evaluation phase. Feedback from this evaluation will then go directly into development of the syllabus generator.

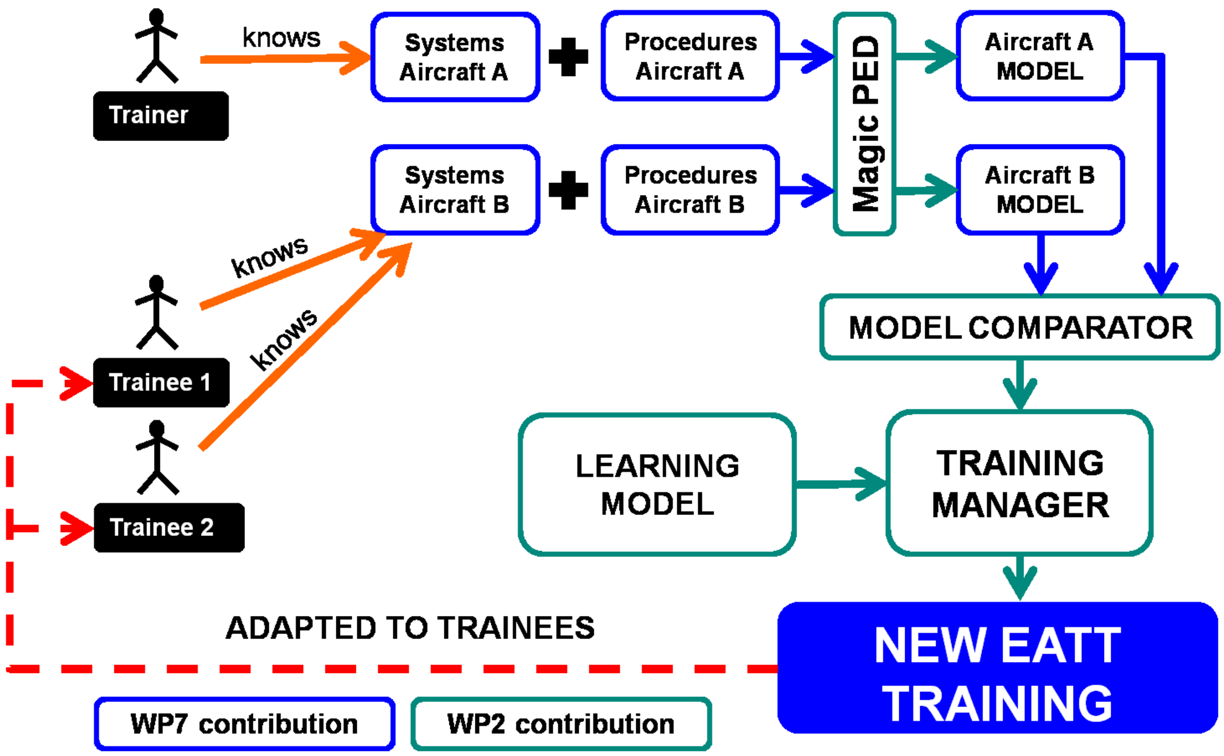

Figure 4: Flow of modelling MTTs related to EATT

3.2.2 The model

3.2.2.1 Aircraft Systems

All system components of both aircraft types (B737NG and A320) that interface with the human agents (pilots) have been identified and collected in tables for preparation of this modelling step, and then modelled as UML classes in MagicPED. The following table shows an overview on the number of modelled systems (UML Classes) and their respective attributes:

Aircraft	Classes (i.e. Systems)	Attributes
A320	89	1850
B737	75	1065

In the following text, the FCU (Flight Control Unit) of the A320, and the MCP (Mode Control Panel) of the B737 will be used to show examples for the system modelling in UML classes. Both instruments are used to control the autopilot of the aircraft. The following figures show the classes for the FCU and the MCP, and below pictures of the FCU and MCP are given.

Figure 5: UML class for A320 FCU

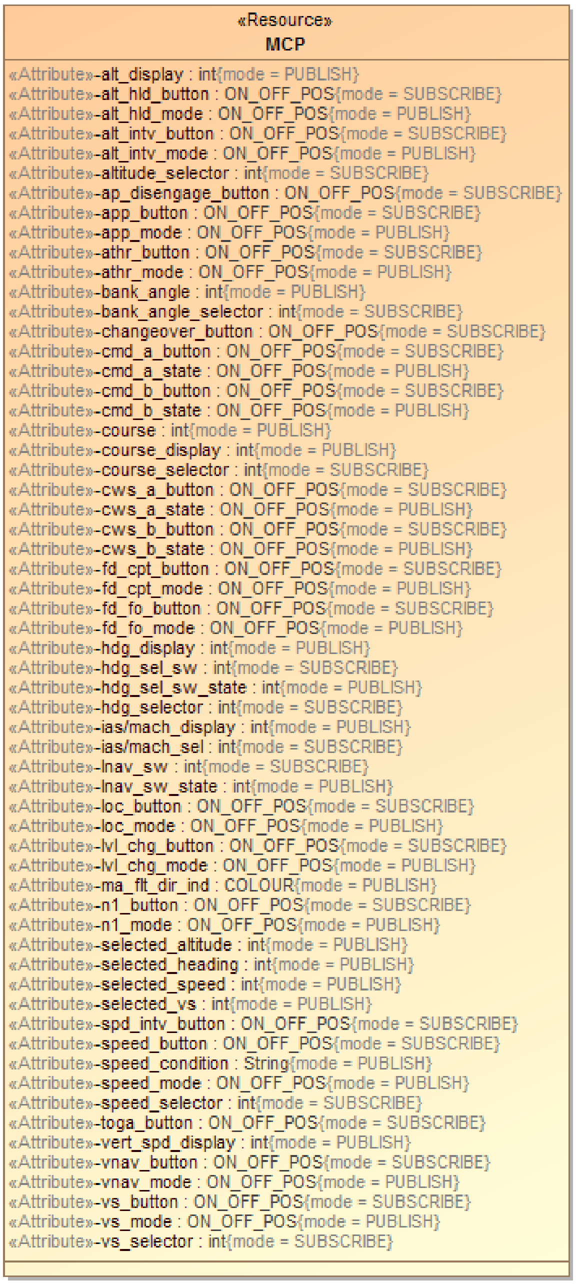

Figure 6: UML class for B737 MCP

Figure 7: FCU A320

Figure 8: MCP of B737NG

As one can easily see, the MCP of the Boeing is more complex than the Airbus one, which corresponds to the number of attributes needed for both classes.
The models of the systems are used in the specification of the SOPs model, i.e. on rule level to model conditions on system states and motor actions.

3.2.2.2 Standard (Aircraft) Operating Procedures (SOPs)

Typically, SOPs are split into several sections for any aircraft type, e.g. normal procedures, supplementary procedures and non-normal/emergency procedures. As the primary purpose of the Training AdCoS will be simulator training, only those procedures that are required for the conduction of type-rating training and checking have been selected: normal procedures (all), non-normal & emergency procedures (all systems/components, but only possible malfunctions/emergencies that are addressed in simulator training). As supplementary procedures mostly relate to special operational aspects, they are not part of type-rating/simulator training but will be part of the line-training section of a type-rating course.

In MagicPED, up to eight levels (instantiations of procedures) have been modelled for both aircraft types. As outlined before, it was of importance to have the same architecture to be kept identical as long as possible for comparator reason. Currently, the models have been successfully kept identical for at least the upper 4-6 levels. Figure 9 shows an excerpt of the containment tree. The green shaded area highlights the task levels that are identical and relate to all affected aircraft types (here: A320 & B737). The amber-shaded area highlights those tasks that became type-specific, e.g. that certain tasks are existent o one aircraft type, but not on the other or that these tasks had to be named differently or that these tasks include steps that differ by number (e.g. on A320 there is one button more to press than on B737 for one specific task (Figure 9).

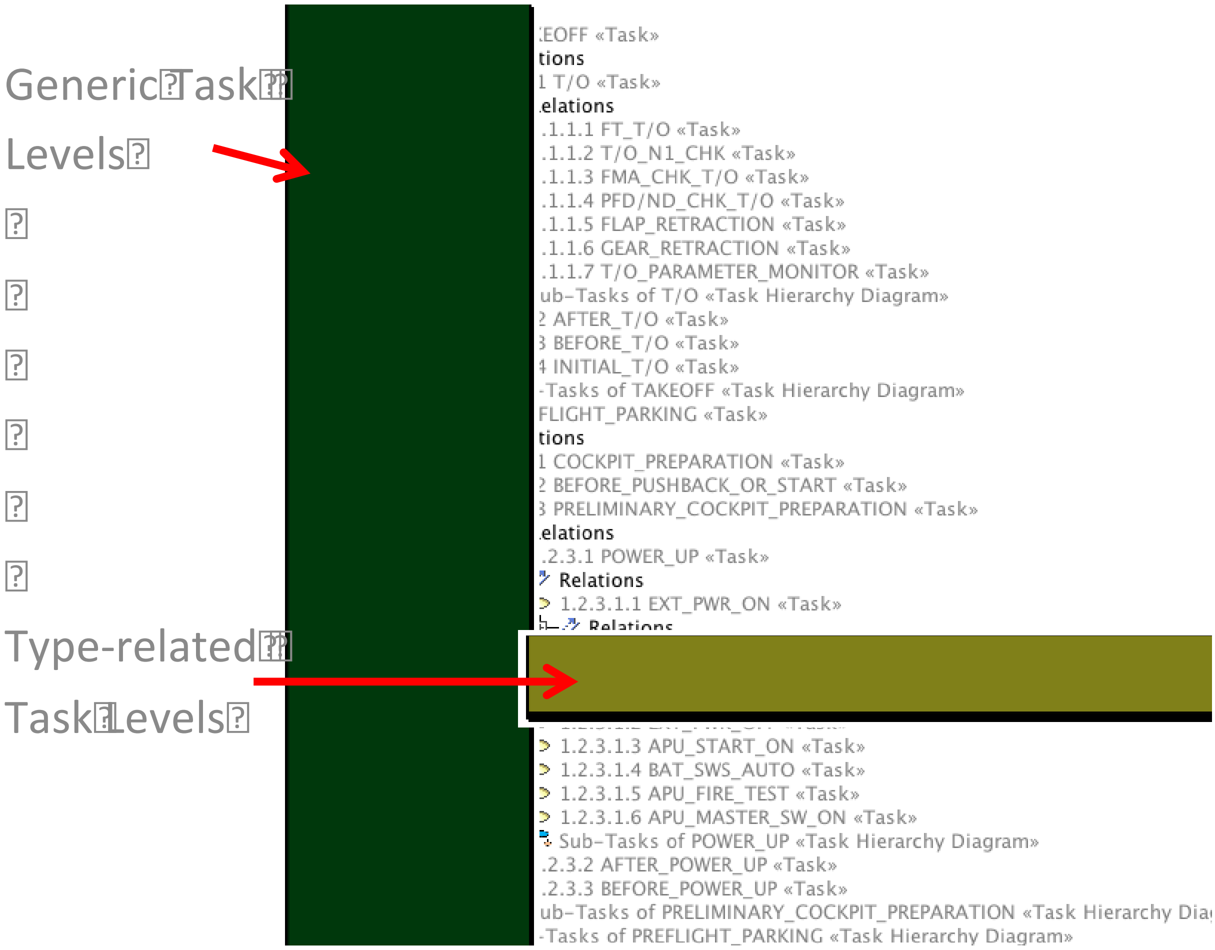

Figure 9: Excerpt of MD Containment Tree with Task Levels (Green= generic, Yellow = type-related)

Figure 10 shows the task diagram in MagicPED for the first level of tasks. This is mainly the normal SOPs (NSOP) for the phases of the flight, plus the flying skills that are performed during the flight (Aviate, Navigate, Manage, Communicate), and the Non-Normal Procedures, as explained earlier.

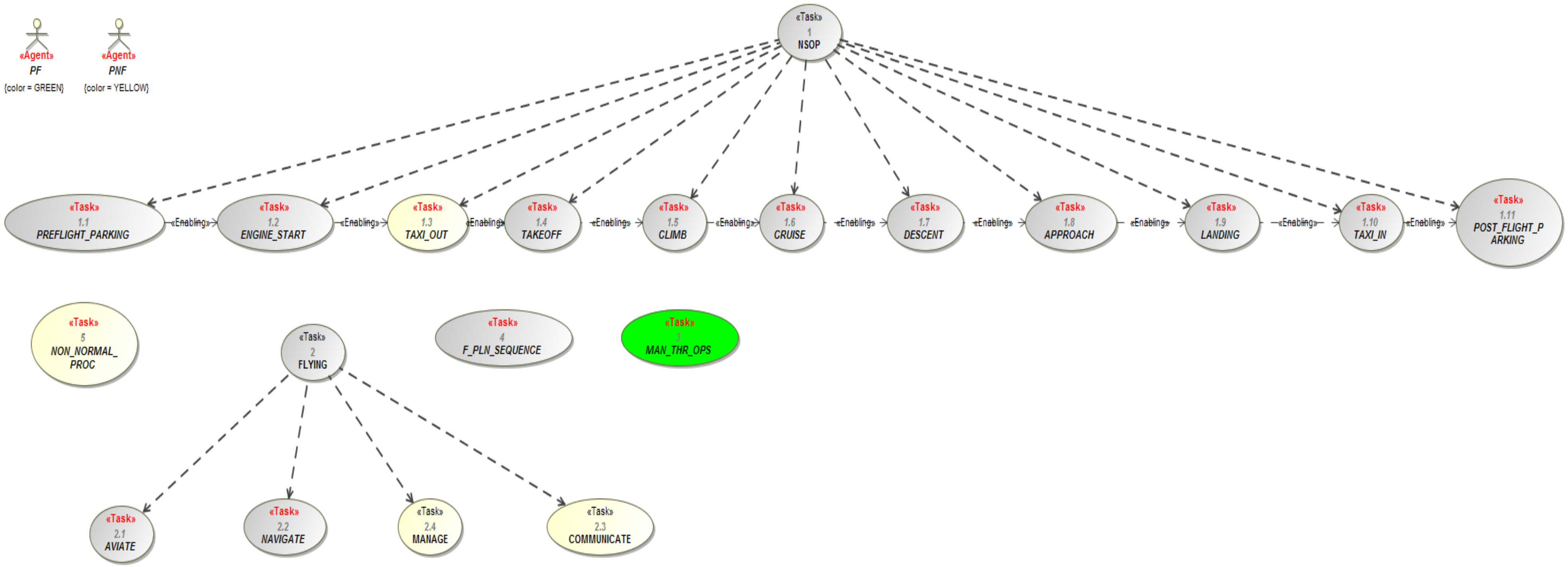

Figure 10: First Level Task Diagram

By defining rules, a very detailed model of the SOPs can be created, down to key-stroke level, which would not be possible only on task level. This is important as the task itself might be identical, but only the keystroke level reveals differences, such as switch/button modalities, resulting in differing system behaviour (even if the task is the same).

Figure 11 shows an example for rules from the B737 model. With these two rules, it is expressed that when the goal ILS_APP is active, you have to check if the Approach Mode[2] on the MCP (mcp.app_mode) is activated (rule 1493), and if not (rule 1495), to press the button in order to activate the Approach Mode.

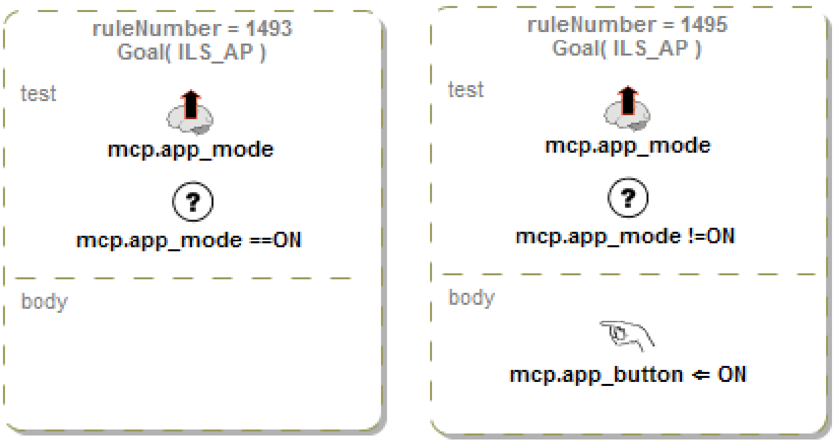

Figure 11: Example Rules "Activate APPROACH MODE" on B737

The following table shows how many tasks and rules have been modelled for both aircraft types:

Aircraft	Tasks	Rules
A320	1722	4088
B737	2795	5540

[2] The approach mode arms the capturing of the localiser and glide-slope of the instrument landing system (ILS), i.e. if the localiser or the glide-slope signal is received and within the required angles, the autopilot automatically starts to follow this signal.

4 Conclusions

4.1 Diversion assistant

Three different aspects of modelling have been discussed for DivA AdCoS. The first one covered the initial phase of system development (following the V-model nomenclature, the phase of 'requirement engineering'). The task modelling allows for deep insight in the working environment and tasks the real user has to complete. This phase has been finished for DivA AdCoS and the task model provided the basis for selection and definition of function requirements.

The following two phases of the V-model, conceptualization and design, have been modelled using the UML language. So far, the system was described in a high level diagrams that will be presented to pilots for first feedback collection. After the feedback is available, low level designs of the software solution will be created.

Third modelling activity aimed at conceptualizing the fatigue phenomenon. In its first step the current state-of-art has been mapped. In the next step, the proposed model will be addressed by number of small size experiments to understand interaction of its factors. The model will be extended with machine learning techniques in order to provide first offline fatigue assessment and ultimately online fatigue assessment.

4.2 EATT

The first stage of the modelling was to model the system variables, system properties and interface aspects (output/input variables from/to the user). The second stage was then to model the tasks according SOPs for normal and later for non-normal procedures for two different aircraft types of different manufacturers (here: Boeing B737NG and Airbus A320 Family). This means that all pilot activities/interaction for normal and non-normal situation that are described by the manufacturer have been covered, including permanent tasks such as monitoring the system and flying the aircraft.

These stages are completed and serve now as the basis for model comparison, workload comparison, syllabus generation and all other aspects of EATT. Currently, the syllabus generation is the work being focused on whilst a first round of experiments is just finished on June 8 – a complete type rating of B737NG pilots on A320.

This syllabus generation compares the two aircraft models, and thus compares the existing pilot (procedural & system) knowledge from the "old" model with the required knowledge for flying the "new" aircraft type.
Based on learning algorithms an optimised training syllabus for pilots that know about the "old" aircraft type will be generated.

The feedback of this training will serve the development of the syllabus generation and adaptation algorithms before the second round of experiments will start with further training courses of the same type but with a more automated training program design by the help of the EATT syllabus generator.
This second round of experiments will deliver data by the start of year three. Feedback on these trainings will allow for a redefinition/refinement of algorithms, UI layout and go in parallel with tool updates to allow for effort reduction in the modelling of the training tasks (e.g. SOPs).