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

D6.5-7.5-8.5-9.5 – Common Annex for Modelled and Model-Based Design of the Application AdCoS

Project Number:	332933
Classification:	Public
Work Package(s):	WP6, WP7, WP8, WP9
Milestone:	M3
Document Version:	V1
Issue Date:	30.06.2015
Document Timescale:	Project Start Date: October 1, 2013
Start of the Document:	Month 20
Final version due:	Month 21
Deliverable Overview:	**Main document:** D6.5-7.5-8.5-9.5 – Common Annex for Modelled and Model-Based Design of the Application AdCoS (public)
Compiled by:	Morten Larsen, AnyWi
Authors:	Morten Larsen, AnyWi Mark Eilers, OFFIS Robert Sharples, Airbus DS
Reviewers:	Jan-Patrick Osterloh, OFFIS
Technical Approval:	Jens Gärtner, Airbus Group Innovations
Issue Authorisation:	Sebastian Feuerstack, OFFIS

DISTRIBUTION LIST		
Copy type[1]	Company and Location	Recipient
T	HoliDes Consortium	all HoliDes Partners

[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
13.05.2015	Initial structure	Morten Larsen, AWI
25.05.2015	Added section "Definition and notion of an AdCoS"	Mark Eilers, OFF
01.06.2015	Added section on commonalities between AdCoS.	Morten Larsen, AWI
02.06.2015	Inserted section on common models used in tools authored by Robert Sharples	Morten Larsen, AWI
26.06.2015	Updated section on commonalities	Morten Larsen, AWI
29.06.2015	Restructuring and updated contributions for all sections.	Mark Eilers, OFF
30.06.2015	Review	Jan-Patrick Osterloh, OFF
06.07.2015	Revision	Morten Larsen, AWI Mark Eilers, OFF
08.07.2015	Integration of comments	Morten Larsen, AWI

Table of Contents

1 Introduction 5
2 Definition of the notion of an AdCoS 5
3 Cross-domain AdCoS commonalities 7
3.1 Workflow support and management 7
3.2 Prepare controlled entity for safe execution of main activity 8
3.3 Assistance for safe execution of main activity 8
4 Cross-domain model commonalities 9
4.1 Model overview 9
4.2 Model dissemination 10
4.3 Model purposes and classification 12
5 Conclusions 15
6 References..16

1 Introduction

The primary objective of HoliDes is the development of *methods*, *techniques*, and (software) *tools* (MTTs) that enable the development and qualification of *Adaptive Cooperative Human-Machine Systems* (AdCoS). *Methods* are rules, procedures, or simply steps to be followed in order to solve specific problems or to design a system (e.g., design patterns) whose applications are structured in terms of concepts, notations, and/or processes in terms of methodologies, *techniques* are instantiations of methods (e.g., algorithms, protocols), and *tools* are concrete implementations of such techniques in terms of software and/or hardware. The general purpose of the MTTs provided by the HF-RTP is to support engineers in their development and analysis tasks. Especially in the human factors domain, a lot of techniques are not strictly formalized and not yet supported by specific tools. Results are often stored in descriptive paper form or excel sheets. One of the major goals of the HF-RTP developed in HoliDes is to close the gap between the engineering disciplines which are to a large extent supported by tools and computer aided techniques, and the human factors discipline.

Throughout the application domains for Health (WP6), Aeronautics (WP7), Control Rooms (WP8), and Automotive (WP9), the MTTs provided by the HF-RTP were used for the modelling of different domain-specific AdCoS, using a multitude of different modelling techniques. This document serves as a common cross-domain introduction and conclusion to the deliverables D6.5 [4], D7.5 [5], D8.5 [6], and D9.5 [7]. The document is structured as follows: Section 2 act as a reminder of the definition and notion of AdCoS. Section 3 will give an overview of the commonalities and differences across AdCoS from different domains that can be identified and used to improve the modelling process. Section 4 will give an overview of the commonalities and differences of the models used for AdCoS development thus far. Finally, Section 5 provides a cross-domain conclusion concerning the modelling and model-based analysis of AdCoS.

2 Definition of the notion of an AdCoS

A cooperative system is defined as a "combination of technology, people and organisations that facilitates the communication and co-ordination necessary for a group to work together effectively in pursuit of a shared goal, and to achieve gain for all its members" [8]. Based on the results of the project D3CoS, we can adapt this definition to Cooperative Human-Machine Systems (CoS), as depicted in Figure 1: A CoS ① is defined as a set of human ②[a] and machine ②[b] agents that interact and communicate with each other to achieve some common, superordinate goal, thus creating a cooperative network, in respect to the operation of a set of controlled entities ⑧. In order to achieve this goal, tasks ④ are allocated ⑥ to the agents, who achieve them by using ⑦ a set of (potentially) shared resources ⑤. The cooperative system and the controlled entities are immersed in an external environment ⑨. In HoliDes, when speaking of a CoS, we refer to the set of agents and their interaction, the set of tasks and their allocation, and the set of resources and their allocation.

An AdCoS extends a CoS by the introduction of adaptation, i.e., an AdCoS is defined as a CoS, that is able to adapt to changes of its internal and external context, where adaptation is specified in terms of *what* parts of the AdCoS are adapted, *why* the system is adapted, *how* the adaptation is performed, and *who* performs the adaptation. The internal context denotes the configuration of a CoS at any given time in terms of the current state of the agents, the currently defined tasks and available resources and their current allocation, while the external configuration denotes the state of the controlled entity and the environment. A more detailed overview about AdCoS and the what, why, how, and who of adaptation can be found in D3.5 [2].

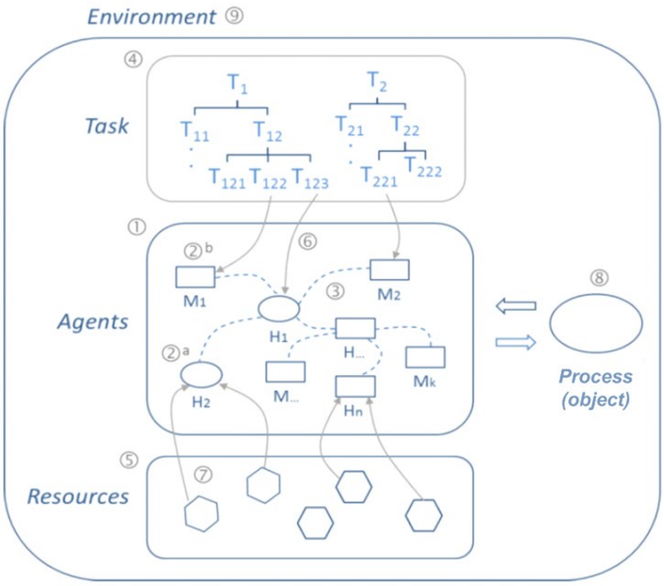

Figure 1: Abstract graphical model of a Cooperative Human-Machine System.

We'd like to emphasize that AdCoS development does not require the explicit specification of all aspects of an AdCoS. Commonly, the actual AdCoS development within the different domains will focus on a subset of entities of the overall AdCoS, e.g., the development of (a subset of) machine agents.. Accordingly, AdCoS modelling does not necessarily imply the development of a single unified model of the AdCoS but may refer to the use of models during the development process or the modelling of aspects of the overall AdCoS. Throughout this document and the application domains, the term AdCoS is used interchangeably to refer to both the AdCoS as a whole or the subset of an AdCoS to be developed.

3 Cross-domain AdCoS commonalities

There are common elements in the modelling of AdCoS across domains, and across roles and purpose of different AdCoS within the same domain. This includes types of AdCoS that fulfil a similar purpose for the operator or maybe common AdCoS components in systems that otherwise do not seem to have many similarities. Examples of AdCoS that perform similar tasks for the operators are discussed in the following.

3.1 Workflow support and management

AdCoS that have the main purpose of helping the users organise the workflow of a single person or a group make up a distinct category in HoliDes.

The problem at the heart of these use cases is that a complex workflow needs to be followed by the staff, while it must at the same time be ensured that the staff members have all the relevant information. The challenges are diverse, and go from the tasks varying over time, as seen in the control room use cases, to systems that have to work with operators of very different levels of training and experience.

The AdCoS that share these characteristics are:

- Operator task schedule and guidance (WP6)
- Guided patient positioning (WP6)
- Control Room AdCoS (energy and border control) (WP8)

To support the devlopment of the AdCoS in these diverse use cases, the same aspect has been targetted for modelling, and that is task analysis, specifically a workflow analysis. The purpose in each case is to improve the insight into the problem and to identify areas where the AdCoS can provide support to the operator.

3.2 Prepare controlled entity for safe execution of main activity

Another category of AdCoS that is found in more than one application domain is the AdCoS to help the operator(s) in setting up the equipment for a main activity.

Examples of this from the healthcare (WP6) and aeronautical (WP7) domains are:

- Guided patient positioning (WP6)
- 3D acquisition (WP6)
- Diversion Airport (WP7)

The two examples from the healthcare domain both have in common that the aim of the AdCoS is to prepare the patient and the equipment (MRI scanner or interventional X-ray equipment) in such a way that the main activity can be performed safely and with the best possible outcome.

In the case of the MRI scanners, the desired outcome is an image that can help create the most accurate diagnosis, while in the case of the iXR equipment, the purpose is to acquire an accurate 3D image of the patient in order to be able to perform the subsequent intervention with the highest level of accuracy and safety.

The example from the aeronautical domain, the Diversion airport AdCoS, shows an AdCoS that has the intended function to provide the user with all necessary information in order to choose an appropriate alternate airport.

The commonality of the AdCoS of this class is also reflected by the fact that the chosen modelling technique across both domains and all three AdCoS is a task analysis (which may lead to more formal model if needed).

3.3 Assistance for safe execution of main activity

Although by now limited to the automotive domain (WP9), different AdCoS in WP9 assist the human operator to safely execute their main activity. Examples for this are the AdCoS:

- Adapted Assistance
- Adapted Automation

Both examples have in common that the aim of the AdCoS is the assistance of a single human agent by a set of machine agents to enhance safety and reliability and/or reducing workload of the driving activity.

4 Cross-domain model commonalities

4.1 Model overview

Based on Deliverable D2.4 [1], this section will provide a brief description of the types of models to be produced within the application domains. This section discuss and define common models, but not go into detailed discussions about the tools used to produce them. In the following, we will provide a brief overview of the different model types developed in HoliDes.

Task Models – Task models are, as the name implies, attempts at describing tasks, subdivisions of a workflow, in such a way that they can be treated formally and provide a useful level of predictability for their intended purpose. As the intended purpose of task models in HoliDes varies across the application domains, a clarification of the matter is in order.

Resource Models - In general, a resource is a source or supply from which benefit is produced, but there are several other definitions, depending on the field a resource is defined in, e.g. in biology a resource are substances or object required by a biological organism for normal maintenance, growth or reproduction. It can also be natural resources (anything from the environment), human resources (skills, energies, talents, knowledge, etc.), or computer resources (memory capacity, network capacity or speed, CPU availability, etc.). Due to this wide range for definitions, there is also a wide range of modelling languages and models available for “resources”, mostly in the form of one or more mathematical formulas, e.g. for network planning. To our knowledge, there is no modelling language, which allows to cover all, or a subset of these various definitions/models.

Cooperative Models - Two forms of cooperation cohabit in HoliDes. One relies on the notion of assistance (*assistance paradigm*), when a single human agent is assisted by one or more machine agents, typically to enhance safety and reliability and/or reducing workload. The other involves multiple human and machine agents cooperating in the realization of common super-ordinate tasks, using shared resources (*human-machine system paradigm*). In the *assistance paradigm*, the focus is on a single user, cooperatively assisted by one or more machine agents. In the *human-machine system paradigm*, the focus is on the system of multiple human and machine agents cooperating on the super-ordinate tasks.

Human Operator Models - As described in the description of work, in HoliDes we differentiate between two types of human operator models. First the cognitive models and second the human behaviour models. Human behaviour models are used for the development of AdCoS, while the cognitive models are more focused on the evaluation of an AdCoS in a later development state.

Training Models - The first version of the training model has been specifically derived for the WP7 Training AdCoS, where the model is evaluated. In future versions, this model will be generalized to other domains.

HMI Interaction Models – For more than 30 years, dedicated languages and methods have been designed and used to deal with the development of critical systems (transportation, health, nuclear, military systems). These languages and methods are used for the development of safe, functionally correct systems. For example VHDL is hugely used for the development of hardware circuit, or SCADE is used for control and command systems.

4.2 Model dissemination

With reference to Section 4.1, with such a wide range of model types, techniques, and tools, the definition of a common model will have to be at a high level of abstraction. The following section details the use of model types at this high level, starting with model types have been used on all work packages, then 3, 2, 1 and none respectively.

Common model analysis should be performed in conjunction with the upcoming deliverables D6.7, D7.7, D8.7, and D9.7 "Implementation of the Application AdCoS and HF-RTP Requirements Definition Update (Feedback)", as these will provide a detailed view of the actual models produced for each demonstrator.

Task Models - Task modelling is by far the most used modelling technique within HoliDes and has been used by all four application Work Packages

- WP6 - The task analysis is the basis for elaborating a shared understanding between the AdCoS and the MTTs
- WP7 - the task models are heavily used for modelling the procedures for process modelling for training purposes
- WP8 - has used task modelling for process, Sequence and State Transitions
- WP9 - Preliminary task modelling and task analysis have been carried for sequences and process modelling.

The common models within the four Work Packages are Use Cases models, Sequence Models, State Transition Models and Process Models, principally using SysML or UML. However task modelling has been used for a wide range of topics, so common model analysis should be carried out after the completion of the demonstrations in November and the four demonstration Work Packages DX.7.

HMI Interaction Models – All four WPs are using the same tool for human efficiency modelling (HEE), therefore the models produced will be common across the HoliDes project. This should lead to common results and metrics collection across WP domains. WP6 and WP7 are using the same method of petri net modelling; therefore the models will be common.

Human Operator Models – are broken down in Cognitive models which are being used by WP9 and WP8 (IRN), and indirectly by WP6 as the model underlying the Human Efficiency Evaluator, and Human behaviour models which are only being used by WP9. Therefore the cognitive models may not have common models at a high level as they are produced by different tools, however an OSLC interface may help to share data. As the Human behaviour models are currently only being used by WP 9, at present it cannot be regarded as common.

Training Models – These are currently being used only by WP7 and therefore it cannot be regarded as common at this stage of the project.

Cooperation Models – These are currently being used only by WP9 and therefore it cannot be regarded as common at this stage of the project.

Resource Models – An initial version of these are currently being used by WP7 and therefore it cannot be regarded as common at this stage of the project.

	Health	Aeronautical	Control room	Automotive
Task Models	X	X	X	X
HMI Interaction Models	X	X	X	X
Human Operator Models			X	X
Training Models		X		
Cooperation Models				X
Resource Models		X		

Table 1 Model types used in HoliDes in the various domains

4.3 Model purposes and classification

Throughout the application domains, AdCoS modelling is performed on different levels of abstraction and with different purposes. In the following, we will attempt to provide an initial overview or classification of models, based on the potential purposes, which can act as a guide to better understand and distinguish the different modelling activities within the application domains. As a starting point, we denote that HoliDes distinguishes two categories of (software) *tools* [2], namely:

- Tools that are meant to be integrated into the toolchains of the AdCoS owners to improve their development process and deal with lifecycle data, such as requirements, test cases, analysis results and behavioural models.
- Tools that are meant to be embedded into the AdCoS to provide or improve its functionalities.

Models are inextricably linked with the tools developed in HoliDes, in that tools are used as *editors* to develop models or that tools serve as a wrapper for models to make them *executable*. Accordingly, we can provide a very similar classification, in that by now we can roughly distinguish

- models used for knowledge acquisition and/or structuring of this knowledge,

- executable models of the AdCoS, or aspects thereof, to be used to provide system functionality or enable model-in-the-loop simulations.

Although by now, different model types seem to centre on distinct purposes, we emphasize that there is no definite mapping from the different common models addressed by HoliDes to these use-cases.

Throughout **all** application WPs, models have been developed to structure relevant information and elaborate a common understanding of the functionality and purpose of the AdCoS and the development task. In general, models based on the Unified Modeling Language (UML) have been used to structure use-cases, develop high-level AdCoS and system models, and/or model the information flow within AdCoS. Concerning the human factors models explicitly addressed by HoliDes, primarily task models have been used in all application domains to understand the task allocation and requirements of human operators.

With higher level of detail (i.e., low level of abstraction), AdCoS models or aspects thereof can evolve into executable models or specifications, which allows their use in model-in-the-loop simulations to enable tests and evaluations in early design phases. As described in D2.4 [1], HoliDes promotes a *model-based design* (MBD) approach for AdCoS development by defining new and extending existing modelling languages that allow designers to model and analyse isolated aspects of the AdCoS and/or the AdCoS as a whole, including system behaviour, human behaviour, and adaptation. In general, MBD is a method for addressing problems associated with designing complex systems and is based on syntactically and semantically (e.g. mathematically) defined abstractions of the system, the environment, and the interactions between them [9]. As such, the development process is based upon a system (or in the case of HoliDes, an AdCoS) model, that is constantly evolved from requirement specification, through design, implementation, and testing. When following the MBD approach thoroughly, the AdCoS should be modelled as an executable specification that is continually refined and elaborated throughout the development process [9]. Simulation of the model allows for easier testing and thus improvement of the product quality, while gaining shorter development times at the same time. This is strengthened by the use of automatic code generation, and support for model-based analysis, i.e. verification and validation.

The HF-RTP provides several MTTs that can be connected for the design and development of an AdCoS as executable specifications in order to perform model-in-the-loop simulations and enable tests and evaluations in early design phases. For these executable models, we can in general distinguish between models of the AdCoS application itself, i.e., models that provide the functionality to the system under development (type 1), and models that that simulate aspects of the AdCoS not covered by this AdCoS application (type 2), e.g., models of the environment, users, or resources. In contrast to "abstract" models, like e.g., task-models for which fully functional editors are already available, executable models need severely more tailoring and/or modelling effort. As such, these models are mainly developed and used by the respective MTT owners themselves, which then provide the tailored models and/or analysis results to the AdCoS owners.

By now, executable models are primarily used within the health (WP6), aeronautics (WP7) and automotive (WP9) domain (c.f., Table 2). Concerning the executable models of type 1, WP7 and WP9 are developing human operator models to enable context-specific adaptation during runtime by the means of context assessment, e.g., the MTTs CONFORM, Driver Intention Recognition, or the Pilot Pattern Classifier. Concerning the executable models of type 2, WP6 uses the MTT Human Efficiency Evaluator in conjunction with cognitive models for performing workload analysis based on the available task models, while WP9 prepares the use of cognitive models, e.g., models developed for the cognitive architecture CASCaS, and COSMODRIVE, for the simulation of human operators interacting with the systems under development, and various kind of environmental and/or sensor models, provided e.g., by RTMaps and Pro-SIVIC.

	Health	Aeronautical	Control room	Automotive
Knowledge structuring models	X	X	X	X
Executable Models, type 1		X		X
Executable Models, type 2	X			X

Table 2: Model classes used in HoliDes in the various domains.

5 Conclusions

In the course of the process of defining and following the development workflows for the many AdCoS being explored in HoliDes, two main uses of models have emerged:

- Knowledge acquisition models that help acquire and/or structure process knowledge
- Executable models of the AdCoS, or aspects thereof, that will be used to provide system functionality or enable model-in-the-loop simulations

The knowledge acquisition type of model is the most widely used, with extensive use in the health, aeronautical and control room domains, mostly in the design phase. The most common human factors aspect modelled with this type of model is the operator work, whether in the form of more formalised task descriptions or less formalised workflow diagrams. The focus also varies from detailed UI analyses (down to the position of buttons on the interface of medical equipment) to an overall workflow analysis of a group of operators working in a control room and in the field. In all of these cases, a major, common outcome of the modelling is a much improved insight into the working situation of the operator(s) and the role of the AdCoS, which means that many issues can be caught in the design phase, rather than having to be discovered in the prototype testing phase of the development cycle.

The automotive domain primarily focuses on the development, implementation and integration of MTTs and associated models that intend to provide functionality to the AdCoS itself, i.e. as components of an AdCoS or AdCoS modules, e.g., in terms of adaptive HMIs, assistance functionalities, or as a means for context assessment. While not the main model type used in the aeronautical domain, the DivA AdCoS also uses this type of model. These MTTs are not used by the AdCoS owners in the traditional sense; instead the different AdCoS components are developed simultaneously by the respective MTT owners and provided to the AdCoS owners in terms of functional AdCoS modules. Consequentially, the primary focus in WP9 has been laid on the development and establishment of a common unified framework enabling a fast and robust integration of the different AdCoS modules into the overall AdCoS. Though not completed yet, the use of this common framework already allowed to easily adapt the interfaces of and connect the different AdCoS components and will serve as a basis for future model-based analysis of the integrated AdCoS addressed.

6 References

[1] HoliDes consortium (2015). D2.4 – Modelling Techniques and Tools V1.0. HoliDes deliverable.
[2] HoliDes consortium (2015). D2.5-D3.5-D4.5-D5.4 Annex HoliDes Integration Plan. HoliDes deliverable.
[3] HoliDes consortium (2015). D3.5 – Techniques and Tools for Adaptation Vs1.5 incl. Handbooks and Requirements Analysis Update. HoliDes deliverable.
[4] HoliDes consortium (2015). D6.5 – Modelled and Model-Based Analysis of the Health AdCoS and HF-RTP Requirements Definition Update (Feedback). HoliDes deliverable.
[5] HoliDes consortium (2015). D7.5 – Modelled and Model-Based Analysis of the Aeronautics AdCoS and HF-RTP Requirements Definition Update (Feedback). HoliDes deliverable.
[6] HoliDes consortium (2015). D8.5 – Modelled and Model-Based Analysis of the Control Room AdCoS and HF-RTP Requirements Definition Update (Feedback). HoliDes deliverable.
[7] HoliDes consortium (2015). D9.5 – Modelled and Model-Based Analysis of the Automotive AdCoS and HF-RTP Requirements Definition Update (Feedback). HoliDes deliverable.
[8] Ramage, M. (1999): The Learning Way: Evaluating Co-operative Systems. PhD-thesis, Department of Computing, Lancaster University.
[9] Vijayagopal, R., Shidore, N., Halbach, S., Michaels, L., and Rousseau, A. P. (2010): Automated model based design process to evaluate advanced component technologies. SAE Technical Paper, No. 2010-01-0936.