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

D7.2 – Tailored HF-RTP and Methodology Vs0.5 for the Aeronautics Domain

Project Number:	332933
Classification:	Public
Work Package(s):	WP7
Milestone:	M2
Document Version:	V1.0
Issue Date:	12.09.2014
Document Timescale:	Project Start Date: October 1, 2013
Start of the Document:	Month 10
Final version due:	Month 10
Deliverable Overview:	**Main document:** D7.2 Tailored HF-RTP and Methodology Vs0.5 for the Aeronautics Domain – Public
Compiled by:	Zdenek Moravek, HON
Authors:	Zdenek Moravek, HON Frank Rister – TRS Jan-Patrick Osterloh – OFF
Reviewers:	Ian Giblett – EADS Fabio tango – CRF
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

[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
01.08.2014	Document outline	Zdenek Moravek
20.08.2014	Input for DivA	Zdenek Moravek
22.08.2014	Input for EATT	Frank Rister Jan-Patrick Osterloh
29.08.2014	Version for project review	Zdenek Moravek José Daniel Gomez De Segura Ugalde Bohuslav Krena
02.09.2014	Internal review comments implemented	Fabio Tango Ian Giblett Zdenek Moravek

Table of Contents

1 Introduction

1.1 Reference technology platform (RTP)

A development environment is a set of workflows, tools and practices that are applied for all phases of a product development. Nowadays, the environment is typically heterogeneous with a significant overhead work spent on moving from one step to another or transferring data from one tool to another. Recent activities have addressed the problem creating interconnected tool chains providing a uniform approach to batteries of tasks however a global uniform approach is missing.

Such a uniform approach requires a set of tools that can cover the whole development process and that can exchange or share data via a defined and respected standard. Together with tools the approach should define methods and processes that on one hand form a framework, within which the tools are used and on the other they describe generically what should be done without explicitly defining tools to be used. Such system is further called a reference technology platform (RTP).

1.2 RTP tailoring process

RTP is based on complementarily and interoperability of its compounds. There are many ways how interoperability can be achieved, the most efficient one relies on well-defined communication and data formats. Each compound can communicate and understand the data, still its role and configuration remains undefined until the compound is used in a specific context. The process of selection, configuration and linking of RTP compounds is called tailoring and it produces an RTP instance applicable for a specific use.

2 Tailoring steps for aeronautics domain

2.1 Project scope and objectives

Rationale: According to the definition of the tailoring process, RTP instance is created for the needs of one particular project. As the RTP instance may address just part of the development process, the scope must be defined first. As a result, the workflow for the project can be defined.

2.2 Evaluation of a workflow and identification of weak points

Rationale: the usual workflow contains an instance of RTP per se, however up-to-date RTP may provide elements that improve the usual workflow by new tools or interconnectivity. The inspection of usual workflow identifies weak points that could potentially be improved.

2.3 Selection of tools for the workflow

Rationale: identify what the RTP provides and make the best selection for the project workflow/dataflow. Rationale for the selection and current state of the art (baseline) for the tool should be given.

2.4 Connection of RTP instance in the workflow

Rationale: define and create interfaces for non-RTP tools (perhaps extend RTP for those tools) that are part of the workflow. In this step, these item should be identified

- data to exchange
- data providers/consumers
- interfaces to connect the tools

2.5 Instantiation of dataflow

Rationale: define how RTP and non-RTP tools are to be connected and how data should be exchanged via IOS or similar standard so that tools can work in chains.

3 Aeronautics tailoring examples

3.1 Airport Diversion Assistant AdCoS – DivA

3.1.1 Scope and objectives

Diversion assistant (DivA) is an application that will assist the pilot in the integration and evaluation of various types of information in emergency situations (due to aircraft failure, adverse weather, etc.). In such a situation, the performance of the aircraft can be affected so that the aircraft can no longer reach its final destination or the aircraft is not able to land at the final destination (weather, aircraft landing/braking performance, fuel constraints). The system will monitor various aircraft systems displaced across the cockpit and the system will perform complex background calculation to evaluate diversion options that the pilot has. The system will take into account cockpit conditions, workload, or state of the pilot (i.e. the calculations and HMI will be adapted to current situation). The system will then report the results to the pilot so that the pilot can decide what diversion strategy will be negotiated with ATC and applied.

As mentioned, 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. When testing the system, applied methodology needs to cover all possible layouts or HMI variants due to adaptation so that the behaviour of the system is assessed against all situations that may arise in real conditions.

3.1.2 Workflow weak points

The standard development workflow is extended in aeronautics domain for specific activities to assure the regulations and constraints are respected. The main focus in these activities is safety and robustness of the system being developed.

When analysing the workflow of the HoliDes use-case of the diversion assistant the adaptation was the key aspect not properly covered in the usual development workflow. As a consequence the following two steps were identified as weak points

- Validation experiments contain activities nowadays done manually. These activities are often error prone with significant effect on schedule

or cost. Specifically, definition of experiment scenarios, time synchronization of measuring devices and data handling should be addressed.
- Parallel processing in software modules is error prone and the classical approach is painful and time-consuming.

The Figure 1 shows the process in detail, activities identified as weak points are highlighted in yellow.

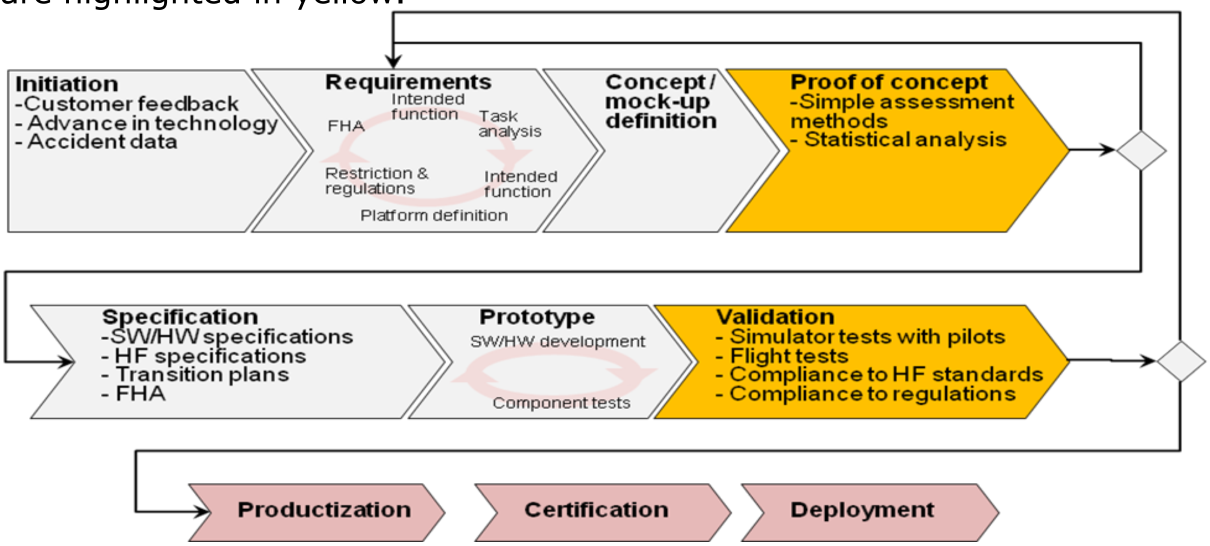

Figure 1: Specific activities in the aeronautics development process. In grey – iterative development, in red – waterfall development and in yellow – activities related to the HoliDes RTP.

3.1.3 Selection of tools

To address the weak points identified in the previous section, the current version of the HoliDes RTP was inspected and a satisfactory match was found. The selected tools are listed in the following subsections.

3.1.3.1 RTMaps

A typical experiment combines a number of devices to record various types of information – flight simulator, system specific logging, video recording, sensor recordings, etc. The weak point identified in previous section showed a large amount of error prone activities – experimenter has to:

- Set-up the experiment – connect necessary devices, define how data will be recorded and stored.
- Start the experiment – assure that all devices start at the same time, or share information about their relative times. This may be difficult as the devices may be distributed on several computers.
- Verify data recording – usually an inspection of recorded data to assure all compounds work correctly
- Pre-process data – before doing an analysis, the recorded data may require adjustments – filtering, trimming, transformations, etc.
- Data organization – keeping record on where the data was stored and adding annotations (test type, demographics data, etc.)

Currently the HoliDes RTP provides an integrated platform that addresses all these questions: RTMaps. It is expected that application of RTMaps will improve the usual workflow as compared in Figure 2.

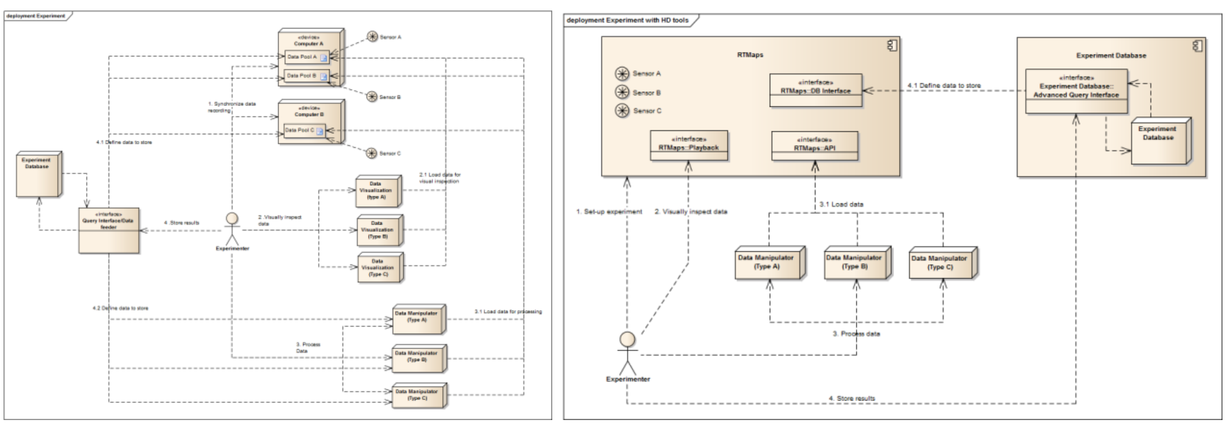

Figure 2: Activities connected to the set-up and execution of an experiment as currently done.

3.1.3.2 Great SPN

A definition of experiment scenario is a process that assures a specific event happening in a realistic environment with a very high likelihood. During the process it is necessary to use domain knowledge (a pilot) and a flight simulator to set up the environment and to iteratively tune system parameters to increase likelihood of tested event to take place in not fully deterministic environment.

For example, when an anti-collision system is tested, relevant scenario needs to bring two or more aircraft in collision – i.e. into a limited mutual distance and time by varying their positions, speed, headings, etc.

The process may become cumbersome for classic systems. For the case of an adaptive system, the process is even more demanding because of an increased number of system states. However, the design process can be modelled by a state automat, for which the HoliDes RTP provides a modelling and simulation tool: GreatSPN.

By means of the tool, the back-tracing of system parameters, which is the critical activity of scenario design, can be automated. The original and the new process are depicted in Figure 3.

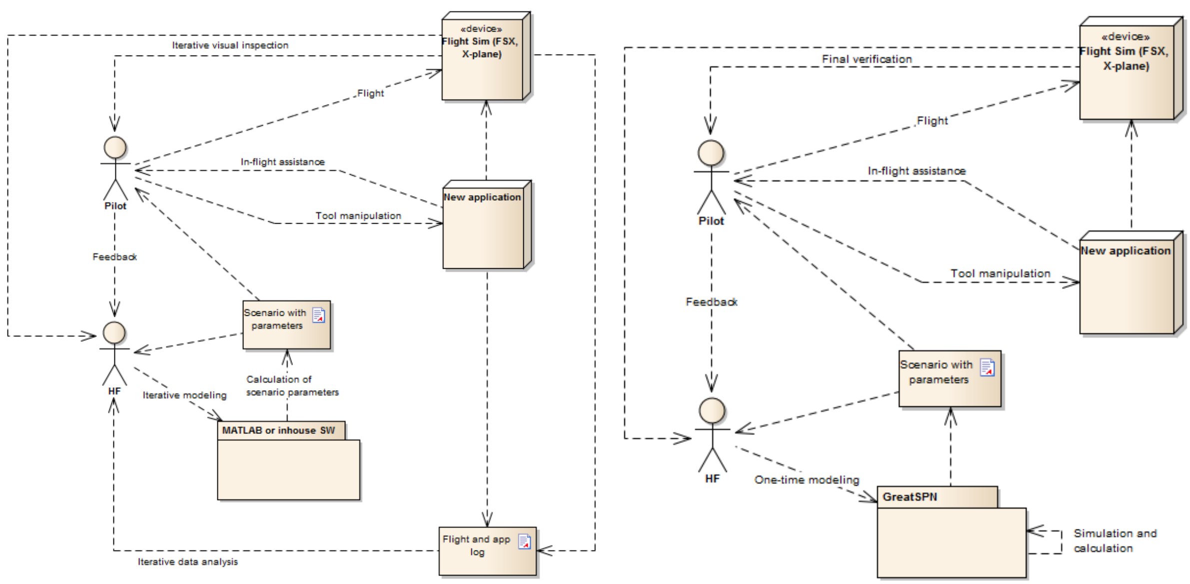

Figure 3: Traditional design of the experimental scenario. It relies on trial-error approach with a significant deal of manual work (on left) and integration with RTP tools (on right).

3.1.3.3 Experiment Database

Experimental data appears in the workflow at several occasions

- when recorded
- when processed
- when analysed
- when reported

Data exists in several instances – as raw data, digitally processed or annotated – and each dataset may be linked to other datasets.

A solution of the problem is to create a relational database to keep the data. As the form of experimental data is rather simple and does not change much in various experiments, the HoliDes RTP shall provide flexible DB solution so that the design and implementation can be reused among various experiments.

3.1.3.4 Pilot monitoring and video surveillance

Modern and future systems on human-machine interface will exploit information about the operator state. In order to determine various psychophysiological quantities such as workload, focus, fatigue, etc. These quantities can be used for system adaptation to prevent deterioration of performance even before it really starts.

Inference of these quantities is based on either direct methods (measurement of various bio-signals) or indirect methods (task observation, task models).

The black box connecting the input signal or a combination of signals and the output psychophysiological quantity provides generic algorithms for pattern recognition and state classification.

In aeronautics AdCoS, the following quantities should be obtained from pilot monitoring and used for system adaptation: workload, perception of visual information and attentional tunnelling. In current HoliDes RT the following tools were selected to address the objectives

- classifier of heterogeneous signals and
- video camera surveillance.

In contrast to current practice the tools are independent on used recording devices and methods of data acquisition

3.1.3.5 Tests on parallel processing in AdCoS

Errors due to thread racing are easy to produce but difficult to eliminate. These errors are characterized by low rate of occurrence, irregular occurrence, they appear without distinct cause and debugging tools often produces misleading information.

As most of the sophisticated systems support parallel processing, the problems of thread race are encountered regularly. Instead of usual practice of painful code review and analysis, HoliDes RTP offers autonomous tools that are able to increase likelihood of error occurrence to support debugging and resolution of the error.

3.1.4 Connection of RTP instance

3.1.4.1 RTMaps

RTMaps is a modular development platform that facilitates and mediates data flows between various devices or software systems. RTMaps can support any client device or system that wants to participate via standard protocols (CAN bus, DDS, NMEA…) or proprietary protocols or drivers, using corresponding RTMaps adapters.

RTMaps provide a proprietary development environment (a C++ API) to support building new adapters. Additionally, providers of RTMaps have already created adapters for many commonly used devices. To some extent RTMaps can be considered as an example of RTP.

3.1.4.2 Great SPN

GreatSPN is an autonomous tool for building and analysing Petri net models. GreatSPN can load a model obtained from another tool, yet standard for data exchange may not be defined. However, in DivA AdCoS, the tool is not intended to do so – the modelling should start with GreatSPN. Sharing results, on the other hand, can be valuable.

As the intended use for DivA AdCoS is to design experimental scenarios, GreatSPN may rely on external algorithms to calculate some parameters of the Petri net in both ways – statically before GreatSPN simulation and in runtime. To support this, GreatSPN needs to have an interface for providing Petri net data and for connecting external tool to accomplish adequate calculations.

3.1.4.3 Experiment Database

The Experiment Database will be specifically set up for the HoliDes project. It will contain a standard RDBMS, such as MySQL, to store the data created during the experiment phase of the process. The contents of the records to be kept will include:

- Unique id
- Date / time
- User
- Additional attributes: experiment type, subject id, access level required
- Data format
- Binary data or pointer to file with actual data

This DB will be accessed by both RTMaps tool and the user. The communication with RTMaps will be performed by an adapter integrated in an advanced query module (AQM), which will also render a UI for the user.

The Experiment Database should be:

- DBMS independent, which will be achieved using SQL to communicate with the AQM
- HF-RTP compliant
- Extensible to new data formats

The above expressed technical requirements will guarantee the future compatibility of the tool with other domains and AdCoS.

3.1.4.4 Pilot monitoring and video surveillance

Pilot monitoring

This tool will infer the pilot status using sensor data as inputs. It is based on grouping different multi-dimensional input patterns under a certain distance measure. Let each dimension of the input data correspond to a certain physical quantity captured by sensors deployed on the cabin or the pilot itself. The system will pre-process the successively recorded data and find patterns therein that might correspond to different status of the monitored human. The advantage is that these algorithms are able to infer such patterns in a non-supervised fashion, i.e. without any a priori knowledge of the actual status of the pilot. This is accomplished by novel clustering techniques whose number of clusters is optimized by means of avant-garde meta-heuristic solvers.

The data received from sensors should be in a highly structured way, such as a comma-separated-values (CSV) file. The description and meaning of the fields of each registry must be agreed a priori. In future versions, an XML file can easily be used as well.

The output will be stored in the Experiment Database and used primarily by RTMaps, but other tool access is not precluded. The monitoring tool will directly use SQL to write data. For future versions, other formats can be utilised, provided that they are supported by Python, which is the internal programming language.

Video surveillance

The tool interprets video streams of an operator in his working environment in order to derive operator's head orientation. There are the following prerequisites

- video stream is understood by the tool.
- the quality of the stream must be sufficient to allow reliable interpretation. The quality is affected by recording device, its parameters must be selected with respect to the properties of the working environment and intended use. In particular sensitivity/contrast, dynamic range, field size, image stabilization must be considered. If the tool determines insufficient quality, respective error code is output.
- coordinate system must be defined in the working environment so that proper spatial alignment and definition of orientation is possible. The camera must be placed so that features used for determination of orientation are always visible. If they are not, respective error code is output. At the same time cues to link head orientation to predefined coordinate system must be defined and always available.
- video stream needs to be linked to the global system time so that the resulting head orientation has correct timestamp.

The tool can benefit from video data processing (filtering and signal refinement, signal conversion and compression, quality detection).

3.1.4.5 Tests on parallel processing in AdCoS

Adaptable NAtive-code CONcurrency-focused Dynamic Analysis (ANaConDA) is a tool suitable for revealing concurrency bugs in C and C++ programs using noise injection technique. Natural thread scheduling is disturbed during the program execution, for instance, by insertion, removal or modification of delays in hope to examine less common thread interleavings where concurrency bugs are typically hidden.

Injecting a noise into an application is implemented by instrumentation of the application’s binary code. As an input, ANaConDA tool needs to know which parts of the application to instrument because the noise injection slows down the execution significantly. Different heuristics for selecting type of noise and places where to be injected as well as probability of injecting the noise at a selected place and its strength can be chosen.

The ANaConDA tool provides reports about found concurrency bugs including information useful for the debugging purposes.

The ANaConDA tool can be combined with SearchBestie infrastructure that is designed to provide environment for experimenting with applying search techniques in the field of program testing. Finding suitable settings of the noise injection for particular application is not easy. Some of the settings are, actually, even able to hide a concurrency bug[2].

3.1.5 Instantiation of dataflow

In this initial version of RTP tailoring the intended instance of RTP (iRTP) is described in Figure 4.

[2] B. Krena, Z. Letko, R. Tzoref, S. Ur, and T. Vojnar. Healing Data Races On-The-Fly. In Proc. of 5th International Workshop on Parallel and Distributed Systems: Testing and Debugging – PADTAD'07, London, UK, 2007. ACM Press.

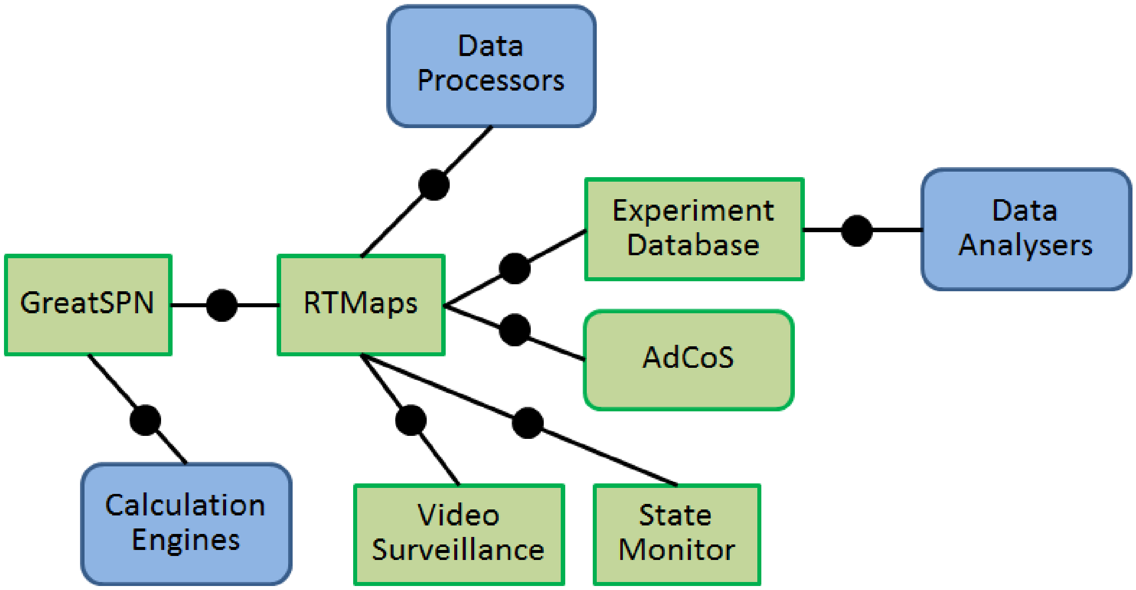

Figure 4: HoliDes RTP tools (green boxes) connected to non-RTP tools (rounded rectangles).

The iRTP addresses numerous activities during preparation and execution of an experiment for AdCoS and during data processing. The whole process starts with GreatSPN for design of experiment scenario. GreatSPN may offer interface for calculation engines developed in-house that determine weights for transitions in the net. The calculations require domain knowledge. GreatSPN determines values of the scenario parameters and these parameters will be transferred to RTMaps and eventually to the database.

In the next step, the experimenter works with RTMaps and devices selected for the experiment. Scenario parameters are loaded via RTMaps and synchronization achieved. RTMaps manages the data recording and registration in the database. RTMaps works as a mediator to the database for tools like video surveillance or pilot state monitor. RTMaps can also use data processors selected by the experimenter either from RTMaps pool, from 3rd party or built in-house.

In the last step of iRTP application, data is supposed to be analysed with non-RTP tools. An interface to database is needed and RTMaps may be used again for their playback capability.

3.2 Enhanced Adaptive Transition Training AdCoS - EATT

3.2.1 Scope and objectives

To date, the generation of training syllabi is solely based on regulatory requirements and the skill of the respective training organisation to write such a syllabus. The result is a one-size-fits-all training solution, as little to no credit is given for neither the aircraft type, nor the trainees' individual needs. Syllabi in the industry today are often copy-pasted and induce an inefficient and ineffective training effort that bears considerable costs and time without quality improvement. This affects especially the transition training, where pilots already having a type rating retrain to another aircraft type.

Therefore, the Adaptive Transition Training AdCoS (ATT-AdCoS) objective is to provide a training syllabus (e.g. Airbus A320) adapted to the skills and knowledge of pilots, who already have a type rating for another aircraft type (e.g. Boeing 737). Unlike pilots who not yet have a type rating, pilots who already have a type rating and typically also many flight-hours, are experienced in flying, navigating, communicating, crew coordination, and management. As many of these skills can be re-used on other aircraft types, it is not necessary to train them again, e.g. communication with ATC, planning of the flight, navigation planning, different types of approaches are the same for B737 as for A320, and also all other aircraft. Thus, transition training has to focus on the aircraft specifics, such as system differences, handling qualities and procedures.

The ATT-AdCoS will adapt the existing training syllabi for A320 to transition training from a B737 to A320. This allows shortening the transition training significantly, such that a) the trainee is earlier available for the employee, and b) the training organisation can offer more variable training (better usage rate of simulators) to more attractive conditions for the customers. In addition, the customer satisfaction will rise due to the increase in predictability of the training success.

In order to adapt the training, a model-based process will be used, allowing to detect procedural differences between the aircraft types, and to automatically create a new training syllabus based on latest knowledge on learning strategies of humans.

This model-based approach will also allow, in future versions of the ATT-AdCoS, to dynamically adapt the training online, based on the individual progress and training success of a trainee.

3.2.2 Workflow weak points

The standard development workflow is extended by specific activities to assure that the trainee's background, the customer needs and the modularity is properly addressed combined with automated consideration of regulations and constraints. The main focus in these activities is quality assurance, the enhancement of safety and efficiency and effectiveness of the training.

When analysing the workflow for HoliDes use-case of EATT the adaptation to the above mentioned points was the key aspect not properly covered in the usual development workflow. As a consequence the following two steps were identified as weak points

- Validation experiments contain activities nowadays done manually. These activities are often error prone with significant effect on schedule or cost. Specifically – definition of experiment scenarios, time synchronization of measuring devices and data handling should be addressed.
- Syllabus Generation as developed in HoliDes does not exist to date, nor does a model-based approach exist that complies with regulations but also adapts to the individual trainee's background and training needs.

Figure 5 shows the process in detail, activities identified as weak are highlighted in orange, new activities are highlighted in yellow:

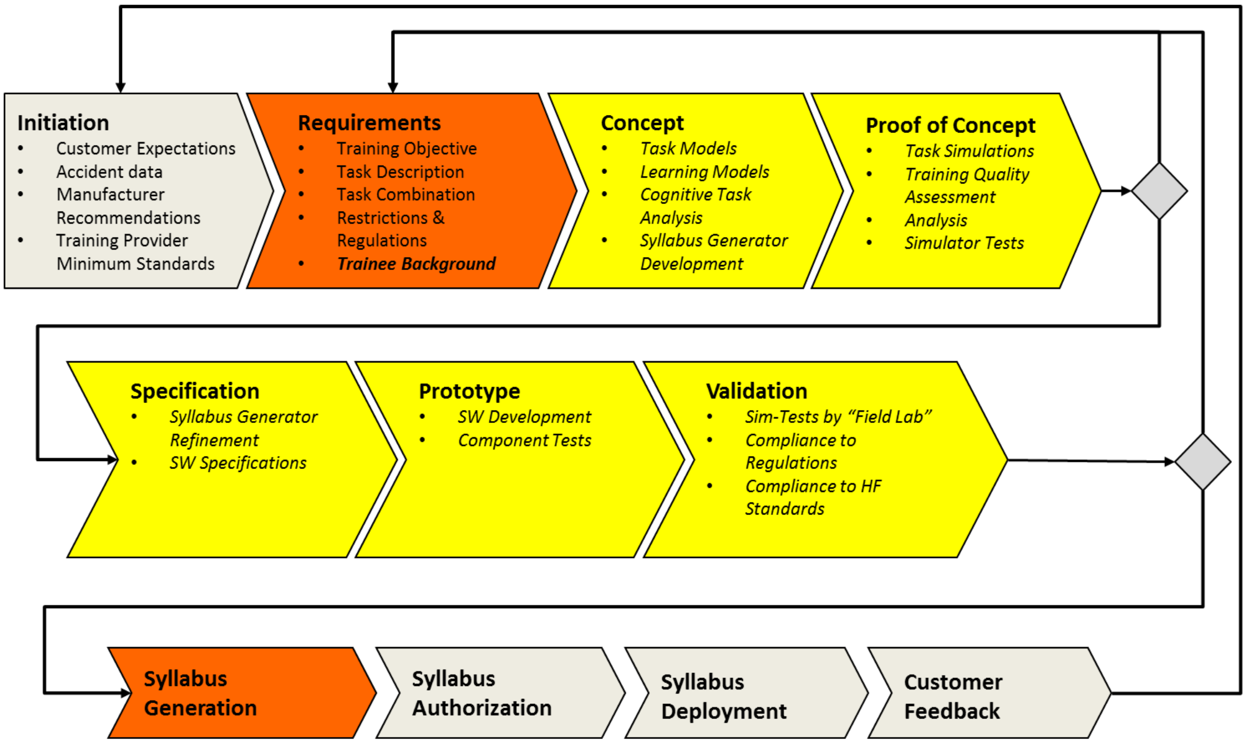

Figure 5: Specific activities in the EATT development process.

The orange activities show weak points and the yellow activities are new activities proposed by HoliDes. The orange and yellow activities are related to the HoliDes RTP, the grey activities will not be touched by HoliDes.

3.2.3 Selection of tools

Basis for the development of the adapted syllabi are models of the tasks the crew performs on a flight. Both B737 as source model and A320 as target model will be modelled using the UML COTS tool MagicDraw from NoMagic Inc. Therefore, a UML profile defining the elements of the task model will be defined. These models will be compared to each other, in order to identify the tasks that are either common or different. The differences will be categorizes and ranked. The categories and ranks are then used by the – to be developed - Syllabi Generator to compute the new syllabus. Thus the selected Tools for EATT are:

- Requirement Tool (DOORS), optional in first cycle
- MagicDraw (as provider for the Task Models)
- Syllabi Generator

3.2.4 Connection of RTP instance

Figure 6 shows the currently planned RTP instantiation. The regulations are defined in a Requirement Tool, like Doors or Jira. The Tasks will be developed in MagicDraw extended by a TaskModel UML Profile defined in HoliDes. As task models can be used in many different development steps, it is planned to provide the models via RTP to all tools that need them.

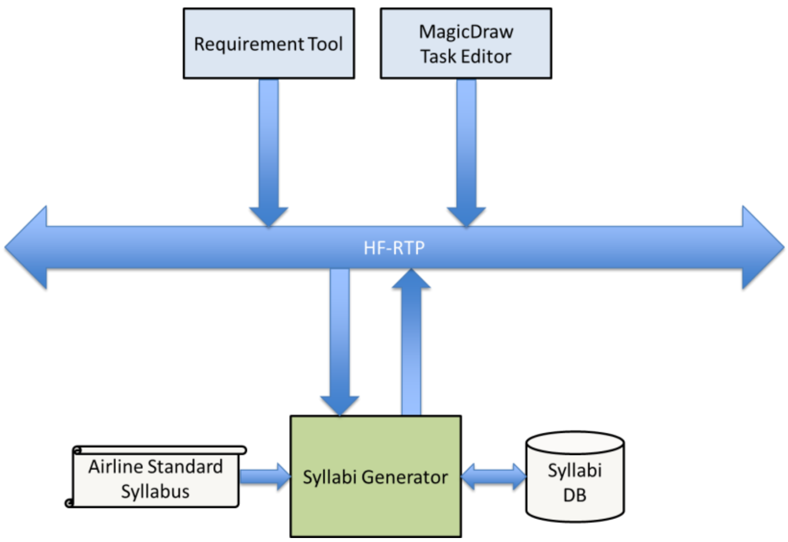

Figure 6: EATT Architecture with needed modules

The Syllabi Generator will be developed within HoliDes.

3.2.5 Instantiation of dataflow

The following Figure 7 shows the data that has to be exchanged between the tools.

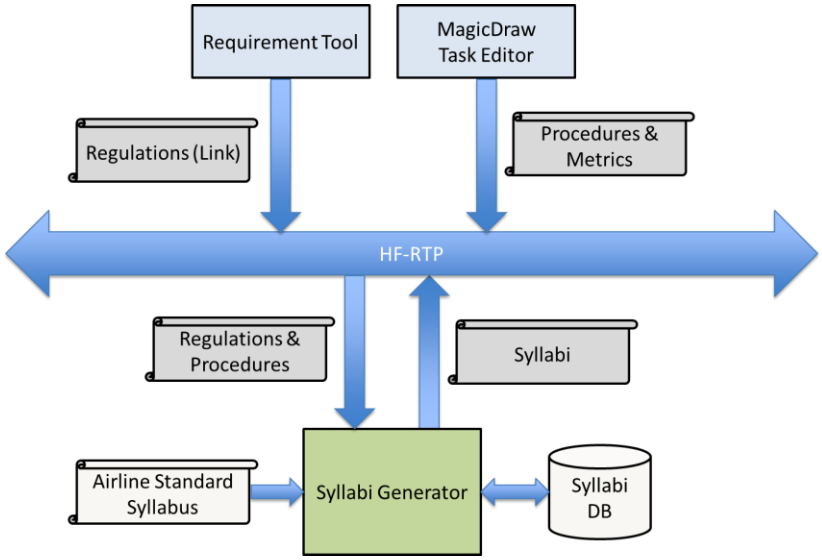

Figure 7: EATT Architecture with Dataflow