APPLICABILITY of SecureChange TECHNOLOGIES to the SCENARIOS

A. Armenteros (TID), B. Chetali (GTO), Q-H. Nguyen (GTO), E. Chiarani (UTN), M. Felici (DBL), V. Meduri (DBL), A. Tedeschi (DBL), F. Paci (UNITN), Michela Angeli (UNITN)

Document information

Document Number	D1.2
Document Title	Report on the applicability of SecureChange Technologies to the scenarios
Version	1.4
Status	Draft
Work Package	WP 1
Deliverable Type	Report
Contractual Date of Delivery	30/01/2011
Actual Date of Delivery	30/01/2011
Responsible Unit	GTO
Contributors	DBL,GTO, TID, UNITN
Keyword List	
Dissemination level	PU

Document change record

Version	Date	Status	Author (Unit)	Description
0.1	16/04/2010	Draft	B.Chetali (GTO)	Initial document structure
0.6	26/04/2010	Draft	M. Felici, V. Meduri, A. Tedeschi (DBL)	Executive Summary; Section 2 on the Applicability argumentation; Section 3 for the ATM case study
0.7	03/05/2010	Draft	A.Armenteros (TID)	HOMES case study
0.8	04/05/2010	Draft	B.Chetali (GTO)	Update
0.9	03/10/2010	Draft	B.Chetali, A.Tedeschi, A. Armenteros	Tentative for an Harmonized version Following Budapest GA
0.10		Draft		
0.11	11/10/2010	Draft	B.Chetali (GTO)	Integration and harmonization
0.12	10/11/2010	Draft	A.Armenteros (TID)	HOMES Feasibility studies and criteria added
0.13	11/11/2010	Draft	B.Chetali (GTO)	§5.3.2 and Chapter 5 update
0.14	10/11/2010	Draft	A. Tedeschi, V. Meduri, M. Felici	• Feasibility Criteria for the ATM Case Study • Feasibility Studies for the WPs working on the ATM Case Study
0.15	16/11/2010	Draft	A. Tedeschi, V. Meduri, M. Felici	New chapter on the Scientific and Validation Criteria
0.16	19/11/2010		B.Chetali	Re organization of the

				document and update of POPS part
0.17	22/11/2010		B.Chetali	Added a section to each case study
0.21	02/12/2010		B.Chetali	Update of POPS part
0.22	07/12/2010		A. Tedeschi, M. Felici, V. Meduri	Update of ATM part, Review of Section 1 (Introduction) and Section 2
0.23	09/12/2010		Q-H. Nguyen (GTO)	Update POPS
0.24	13/12/2010		A.Armenteros (TID)	Update HOMES
1.0	16/12/2010		B.Chetali	Preparing the document for the quality check
1.1	27/12/2010		A. Tedeschi, M. Felici, V. Meduri	Updated Criteria for WP2
1.2	30/12/2010		Michela Angeli (UNITN)	First quality check completed. Minor remarks.
1.3	10/01/2010		Q-H. Nguyen, B.Chetali	Update following the first quality check Updates required by Bjornar on HOMES
1.4	19/01/2011		B.Chetali, A.Tedeschi	Update following GA Innsbruck

Executive summary

This deliverable describes the feasibility studies that have been carried out on each case study of the SecureChange project. The feasibility studies assess the applicability of the artefacts developed in the SecureChange technical WPs to solve the change-related security problems arising within industry domains. The feasibility studies identify criteria to evaluate the applicability (that is, the industrial relevance) of the means (model, tools and methodologies) proposed by technical WPs.

To this aim, industry partners responsible for each case study have selected the main requirements illustrating the change that occurs in the corresponding contexts [D1.1], and have identified the relevant security properties that will be used to focus the scope of the analysis and the applicability for the SecureChange artefacts.

The document identifies the selected requirements for each case study that are to be addressed by means of models, tools or methodologies developed within the project. For each case study and its security requirements, the means and the associated feasibility criteria are defined.

The objectives of the document are:

- Identifying the WPs that will provide artefacts to solve the security problems
- Describing the feasibility studies that have been carried out between the WP1 partners, the users of the artefacts and the technical WPs, the providers of the technologies.
- Introducing the criteria that will be used for the evaluation of the artefacts during the third year

The case studies in WP1 are ATM (led by DBL), HOMES (led by TID) and POPS (led by GTO). The case studies are drawn from different industry domains. They highlight different change requirements and security properties, respectively. Therefore, the solutions provided to solve their change-related security problems are quite different. Hence, they are evaluated according to a wide variety of criteria. For example, the *consistency* of the risk analysis modeling artifact for the ATM case study is critical while the performance criteria is more suitable for the embedded running code for POPS or the *availability* criteria for the services for HOMES. However, the common evaluation criteria of the three case studies will be the suitability in an industrial context.

For the ATM case study, the main change requirements concern *process level change* and *organizational level change*, for which WP2, WP3, WP4 and WP5 collaborate to preserve *information access, protection and provision* security properties.

For that; WP2 is providing a meta-model to describe the ATM system and related processes with respect to Security issues arising after AMAN tool introduction, WP3 is providing model, languages and tools for foster a simpler and more effective collection of requirements that takes into account evolutionary aspects, WP4 is providing models for better analyzing and reasoning about Security Properties and WP5 is supporting a complete Risk Assessment for the Organizational Changes occurred in ATM.

For the HOMES case study, the main change requirements are the *update of a security module* and *bundle lifecycle operations*, for which WP2, WP5, WP6 and WP7

collaborate to preserve the *security expandability, secure extensibility, policy enforcement and resilience to trust changes* security properties.

More precisely, WP2 is delivering an implementation of a Security-as-a-Service Engine to make possible application of additional security functionalities along with a tool implementing a methodology to assess the impact of changes into the system; WP6 is providing a tool to verify the actual code of security modules to detect vulnerabilities and a methodology to check the exchange of data between OSGi services; WP7 is designing a test model and test suites to check the impact of changes in advance and finally WP5 is working in collaboration with WP7 in risk modeling of the system with feedback from the testing work.

For the POPS case study, the main change requirements are the *update of the embedded security software* and the *specification evolution*, for which WP3, WP4, WP6 and WP7 collaborate to preserve *information protection* and *deny of service* properties.

More precisely, WP6 and WP4 provide tools and associated modeling and verification techniques to check that the "new" application to be loaded on the card verifies the critical security properties *from its development to its installation on the card*. For that, an off-card tool checks the application with respect to *denial of service* properties, and then the WP4 verify the *secure communication* between the terminal and the card during the loading. After the loading and before the installation, on-board verification techniques will be provided by the WP6 to check that the loaded application respects the *information flow policy* of the card. The WP7 and WP3 will provide test suites and traceability techniques to check that a new implementation w.r.t an evolution of the specification of the underlying platform respects the information access control properties.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

This document presents how the technologies developed within the project are applied to specific problems (in terms of changes requirements and security properties) identified for each case study. The relevant case studies included in Secure Change are vehicles for demonstrating the advantages and benefits coming from the research activities in this project. The case studies are led by industrial partners and provide adequate scenarios where changes and evolution related issues arise. Secure Change's technologies deal with such issues.

The three case studies lie on different industry domains. The aim is to show the Secure Change approach is relevant in different industries. The case studies are representative of relevant but not exclusive application domains of Secure Change output. They are named and identified within the project according to their industry domains:

- Air Traffic Management case study (ATM)
- Home Network case study (HOMES)
- Smart Card case study (POPS)

Each case study comes with his specificities of changes and impacts. Therefore the analysis of the applicability of developed technologies will follow different feasibility criteria. **Figure 1** shows how the WP1's deliverables highlight an industry-driven validation of the SecureChange artefacts.

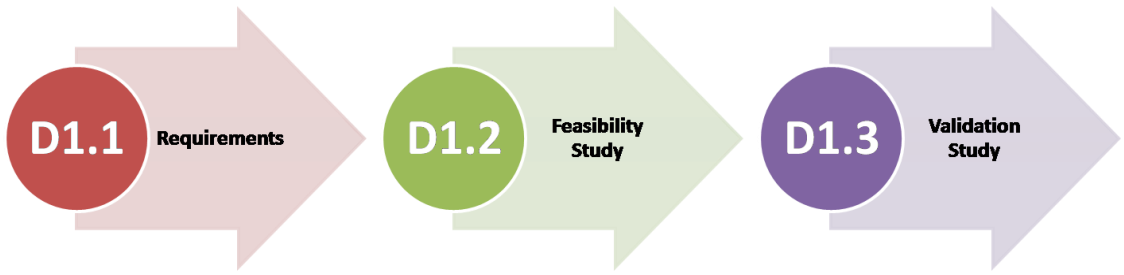

Figure 1: WP1 deliverables highlighting an industry-driven validation of SecureChange artefacts

Deliverable D1.1 identified relevant change requirements and security properties drawn from the three different industry domains [1]. Deliverable D1.1 intentionally identified a wide range of changes requirements and security properties in order to support the discussion of relevant evolutionary concepts underlying the SecureChange project. On the contrary, this report, Deliverable D1.2, intends to narrow the scope of the problem according to industry feasibility criteria. The point is to ease the adoption of SecureChange artefacts by aligning them with current industry practices and expectations. Deliverable D1.3, finally, will assess SecureChange artefacts according

to a wider range of evaluation criteria comprising feasibility and validation criteria. Deliverable D1.2, therefore, has a pivotal role in linking the industry problems (Deliverable D1.1 [1]) to the validation of the SecureChange artefacts (Deliverable D1.3) by narrowing the scope according to industry feasibility criteria.

2 Applicability argumentation

2.1 Introduction

This report focuses on a small set of concrete problems, derived by the more general Case Study depicted in D1.1 [1]. These problems were agreed by all the participants in the WP1. For each Case Study, selected changes requirements and security properties have been identified in order to show how the project and mainly the technical WPs address them. These problems could be general statements as well as concrete scenarios that will be used to illustrate change requirements and their impacts in term of security properties. Nevertheless, to provide a homogeneous description for each Case Study, a common feasibility argumentation has been developed with the aim of providing a general format, valid for all the Case Studies, but also enough descriptive to work as a basis for any assessment process. Each Case Study follows a similar reasoning and feasibility argumentation.

2.2 Narrowing the Scope

This section describes the overall processes for narrowing the scope of the feasibility study. The description highlights how the narrow scope is the result of exemplifying relationships between high-level requirements identified by the case studies and low-level SecureChange artefacts. **Figure 2** depicts such narrowing process consisting of three steps: identifying relevant properties (feasibility criteria), identify specific requirements (drawn from the case studies) and capturing feasibility arguments.

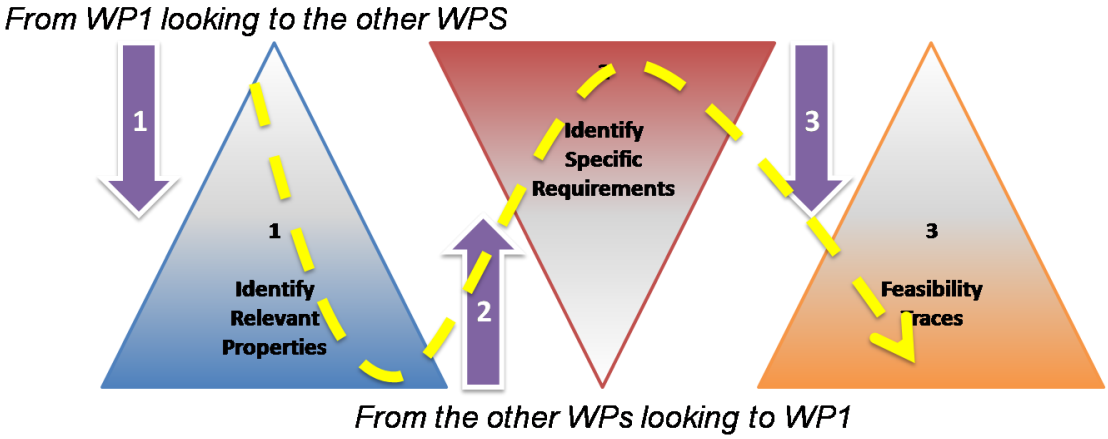

Figure 2: A three step process for identifying relevant properties, requirements and assessing the feasibility of WP artefacts

2.3 Structured feasibility argumentation

This section describes a structure for the feasibility argumentation of the case studies.

Changes REQUIREMENTS: The requirements of changes characterizing the case studies.

Security PROPERTIES: the security properties that must be satisfied w.r.t the change requirement that occur.

Security MEANS: the SecureChange means (i.e., Tools, Models, and Methodologies) that could be developed and used to guarantee the identified security properties, hence, providing evidence of supporting the change requirements. Each case study will identify the means that are the first candidates to handle the corresponding security properties. Each case study will highlight the MEANS applicable to its specificities and to the security properties identified. A table will summarize the relevant means organized per types (i.e., Tools, Models, Methodologies).

FEASIBILITY criteria: those criteria describe the scope of the evaluation of the security means to be applied to resolve the change problems of each case study. For example, if for a security mean, we need to formalize the complete system, this security mean will be considered not feasible to tackle the given security problem.

FEASIBILITY Studies: feasibility assessments that will be carried out, in order to inform the subsequent phases of the project. The Feasibility Studies, will report the experience of how the SecureChange outcomes address the selected change requirements and security properties while complying with industry feasibility criteria. In particular, we will report our preliminary experiences about the application and the assessment of the SecureChange solutions within the three Case Studies. This allows us to compare different aspects with respect to the implementation and adoption of technical WPs artefacts into technical systems tailored for specific application domains. Each partner will describe the different feasibility assessment activities conducted and the relative results.

The main activities, performed by different partners with different characteristics, maturity levels and scopes, are: first requirements collection with domain experts and end users, solution identification and instantiation, evaluation by the Case Study of the solution requirements, interviews and questionnaires, modeling activities with end users and experts, preliminary feasibility assessment by means of application of the solution to particular change requirement provided by industrial partners in each Case study. These activities allow us to highlight how the SecureChange project supports Security and evolutionary aspects within industry domains. The applicability evaluation activities rely on qualitative as well as quantitative approaches. The empirical results stress the feasibility of the SecureChange results from a practitioner and operational viewpoint and their applicability in real industrial contexts.

Figure 3 shows the basic elements involved in feasibility argumentations of the SecureChange artefacts. Note that the above elements intend to clarify the terminology in order to provide us a systematic way of arguing about the feasibility of SecureChange means in order to address security concerns related to the changes requirements drawn from the different case studies.

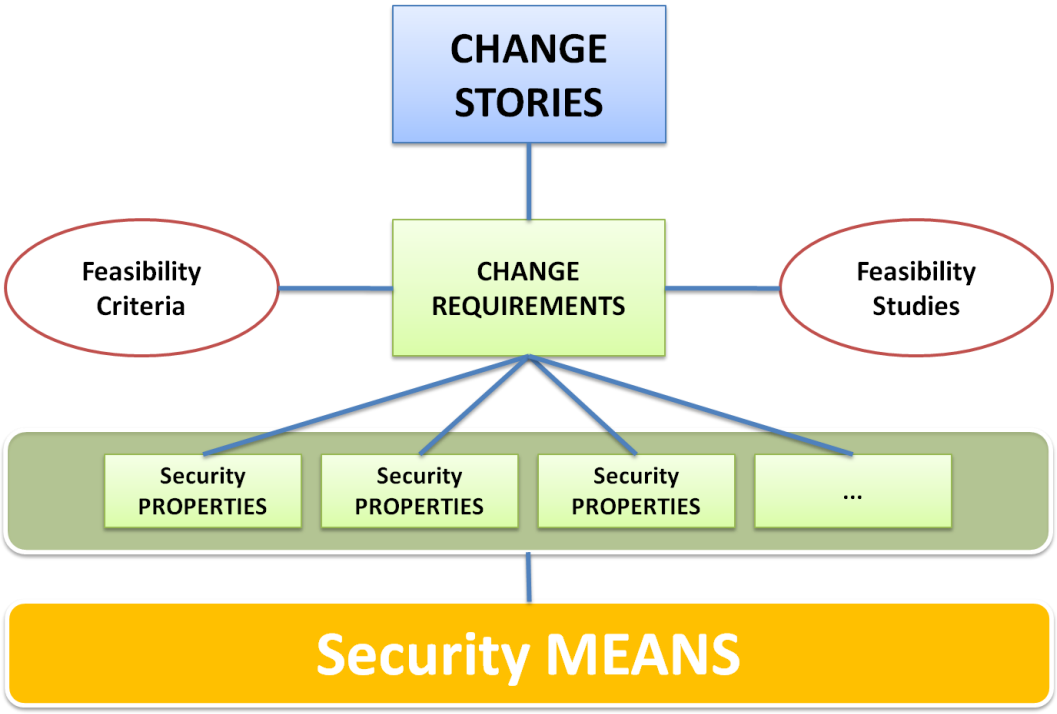

Figure 3: Basic elements for feasibility argumentations of SecureChange artefacts

EVALUATION criteria: This section identifies criteria that will be used to evaluate the specific security means when it is applied to the considered case study. We have identified industry relevant criteria. The criteria will come from an agreement and technical discussion with the partners. The results of the evaluation of the artefact with respect to those criteria will be described in the deliverable D1.3.

3 ATM Case Study

The ATM case study is concerned with **changes in the operational processes of managing air traffic in Terminal Areas**. Arrival management is a very complex process, involving different actors. Airport actors are private organizations and public authorities with different roles, responsibilities and needs. The subsequent introduction of new tools, i.e., the Queue Managers, and the introduction of a new ATM network for the sharing and management of information affects the ATM system as a whole at a **process** and **organizational** level. The next section describes the selected requirements drawn from deliverable D1.1 [1].

3.1 Change Requirements

3.1.1 Process Level Change

ATM procedures need to be updated in order to accommodate the introduction of the AMAN (**A**rrival **MAN**ager). The AMAN is an aircraft arrival sequencing tool helping to manage and better organize the air traffic flow in the approach phase. It is directly linked to the airport organization and the turnaround process, because arrival sequencing/metering is important for airline operational control and airport operations (e.g., ground handlers, catering services, airlines, security and health authorities, etc.) in order to organize the ground services efficiently.

The introduction of the AMAN requires new operational procedures and functions (as described in the deliverable D1.1 [1]). Such new procedures and functions are supported by a new information management system for the whole ATM, an IP based data transport network that will replace the current point to point communication systems with a ground/ground data sharing network which connects all the principal actors involved in the Airports Management and the Area Control Centers.

Goal: The resulting ATM system (with the AMAN and the communication network introduction) needs to comply with suitable security properties, which prevent from corruption, accidental or intentional loss of data and guarantee the integrity and confidentiality of the aircraft sensible data against malicious attacks or intrusions.

3.1.2 Organizational Level Change

The introduction of the AMAN affects Controller Working Positions (CWPs) as well as the Area Control Center (ACC) environment as a whole. The main foreseen changes (as described in the deliverable D1.1 [1]) in the ACC from an operational and organizational point of view are the automation of tasks (i.e. the usage of the AMAN for the computation of the Arrival Sequence) that in advance were carried out by Air Traffic Controllers (ATCOs), a major involvement of the ATCOs of the upstream Sectors in the management of the inbound traffic.

These changes will also require the redefinition of the Coordinator of the Arrival Sequence Role, who will be responsible for monitoring and modifying the sequences generated by the AMAN, and providing information and updates to the Sectors.

Goal: The AMAN's interfaces that provide access to different roles and authorizations need to make information available only to authorized personnel or trusted systems.

3.2 Security Properties

The following security properties need to be guaranteed at the process and organizational level and will be the focus of the technical WPs.

Information Access. Authorized actors (or systems) must have access to confidential information regarding queue management in the terminal area. Access to information needs to comply with specific role-based access control rules drawn from the operational requirements.

Information Protection. Unauthorized actors (or systems) are not allowed to access confidential queue management information.

Information Provision. The provisioning of information regarding queue management sensitive data by specific actors (or systems) must be guaranteed 24 hours a day, 7 days a week, taking into account the kind of data shared, their confidentiality level and the different actors involved.

3.3 Security Means

The SecureChange means (i.e., Tools, Models and Methodologies) guarantee the identified security properties, hence, providing evidence of supporting the top security requirements. The Security Means that will be used for the architectural modeling, the requirement collection and the risk analysis of the proposed Change Stories of the ATM Case Study will be the ones provided by WP2, WP3, WP4 and WP5, namely:

WP2

- A **meta model** for the integrated view of (service oriented) systems from several points of view and at several layers of abstraction, integrating security related information (e.g. requirements, risks, security controls) capturing the security of a system at any point of time
- A **tool** supporting the management and evolution of system artifact configurations at the level of the meta-model concepts

WP3

- A **conceptual model** for the characterization of evolving requirements,
- A **methodology** for the identification, monitoring and transformation of requirements,
- A **proof-of-concept tool** implementation that will show the potential of the approach

WP4

- Thales Security Engineering Tools
- UMLseCh Language

WP5

- **modeling languages** with the expressiveness to capture forecasts of future change,
- specialized security **risk assessment methods** for models with forecasts,
- **tool**-supported ways of representing links and dependencies between models with forecasts and assessment results.

3.4 Feasibility Criteria

This section identifies the feasibility criteria for the ATM case study. For the identified changes requirement, each WP working on the ATM case study provides examples stressing the *feasibility* with respect to the identified security properties, and the *applicability* of the security means, that is, the SecureChange artefacts delivered, in the ATM domain. In order to support best-practices in industry, we are particularly concerned with three main feasibility criteria: 1) to support structured approaches to changes, 2) to capture security properties affected by changes, and 3) to provide mechanisms dealing with subsequent changes.

1. **Supporting structured approaches to changes.** ATM stakeholders and industries are particularly interested in structured approaches to changes, because this would provide further support to traceability practices in industry. Identifying changes in a structured manner allows us to relate specific changes to their rationale. Therefore, we would like that all security means, that is, the SecureChange artefacts, provide us with structured models that capture changes. This would enable us to capture subsequent changes into different system accounts.

2. **Capturing security properties affected by changes.** ATM stakeholders and industries would like to be able to model and assess security properties that are affected by changes. This is to support model-driven developments and deployments with respect to security and dependability. This would enable the predictability of deploying technology innovation (changes) in industry. Moreover, this would also allow the assessment of changes with respect to work practices. This would support sensitivity analyses of security properties affected by changes.

3. **Providing mechanisms dealing with subsequent changes.** Industry needs to accommodate different changes due to various key performance drivers. These changes, in order to be feasible, need often to be prioritized. Moreover, it is often difficult to maintain an historical account of such changes. Therefore, we would like to be able to maintain a rationale of such changes. That is, we would like that SecureChange artefacts enable us with mechanisms for maintaining relationships between subsequent changes. This would allow us to

link subsequent changes and maintain change rationale too. This would comply with industry practices dealing with changes [2].

3.5 Feasibility Studies

The feasibility criteria allow us to identify specific problems for each of the WP working on the ATM case study, dealing with specific changes requirements and security properties. Table 1 summarizes on what changes requirements and security properties each WP is working on.

	Change REQ 1 Process Level Change	Change REQ 2 Organizational Level Change
Sec. Prop 1. **Information Access**	**WP3**	**WP3**
Sec. Prop. 2 **Information Protection**	--	**WP2, WP4, WP5**
Sec. Prop. 3 **Information Provision**	--	**WP2, WP4, WP5**

Table 1: ATM Requirements & Security properties

The remainder of this section shows the feasibility exercises we have conducted for each WP working on the ATM changes requirements and their security properties. The studies allowed us to acquire an account how relevant SecureChange artefacts address our feasibility criteria.

3.5.1 Process Level Change

3.5.1.1 Feasibility for WP3

WP3 will focus on process level change requirement and the information access and information protection properties. The scenario fragment we are going to consider is transmission of FDD data to the AMAN via the new communication network. We want to focus on how to enforce access control policies on FDD transmission and how to ensure confidentiality of FDD. In terms of security means, we are going to apply the SeCMER methodology for requirement change management to the ATM case study. We will produce SeCMER models before and after changes of introducing the Arrival Manager tool and the communication network; the argumentation analysis for the security goal of protecting FDD from malicious attack; and we will show how to use

evolution rules for monitoring and adaptation to the triggering and reactive changes to the SeCMER models.

Feasibility Arguments

This section describes examples drawn from ongoing WP3 work. We have collaborated with WP3 partners in order to gather feasibility arguments for WP3 with respects to our feasibility criteria (that is, *supporting structured approaches to changes*, *capturing security properties affected by changes*, and *providing mechanisms dealing with subsequent changes*). The feasibility trial involved focused modeling exercises of the ATM Changes Requirements and their relevant Security Properties. For the first criteria on the structured approach to requirements, we requested partners to provide models that captured changes in a structured manner. This involved the comparison of requirements models *before* and *after* the changes. Note that models have been reviewed and iterated in order to capture domain expertise with respect to the changes requirements. **Figure 4**, for example, shows a SecMER requirement model (partially inspired by the Problem Frames approach to requirements engineering [3]) capturing the relevant domains before the changes. The model captures the given domains and their interfaces (represented by the connections between the domains) as they currently are in ATM domains.

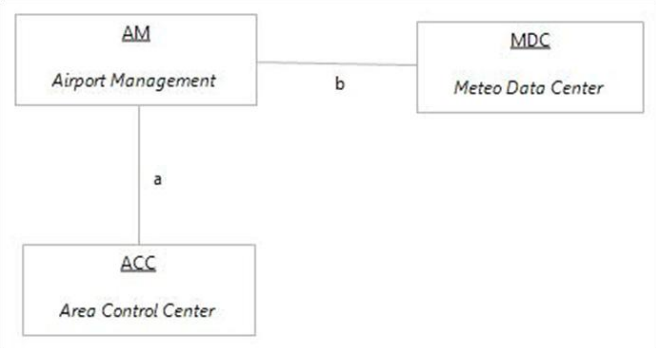

Figure 4: A SecMER requirement model capturing the relevant domains before the changes

Figure 5 shows how a SecMER requirement model stressing how the introduction of the SWIM Network, an IP based data transport network, relates to other domains necessary for its integration within the current ATM settings.

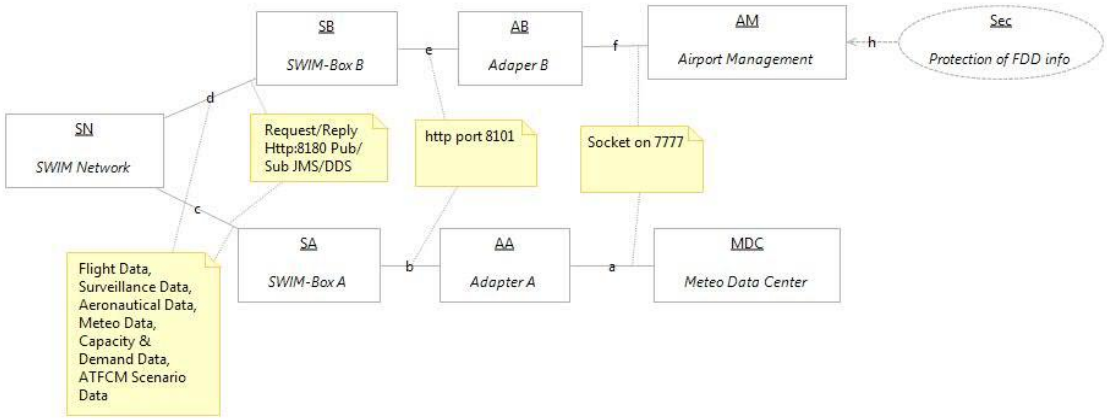

Figure 5 : A SecMER requirement model capturing how the SWIM Network relates to other domains

The structured models allow us to analyze the impacts of changes requirements. This allows us to discuss and communicate relevant changes requirements with domain experts. The preliminary analysis of changes by the SecMER requirement modeling allowed us to clarify changes requirements. **Figure 6** shows a fragment of the SeCMER model for the introduction of the AMAN and the SWIM network. Note that the modeling of change requirements by subsequent arguments (different structured arguments correspond to different rounds) takes into account the evolutionary aspects of changing requirements in a structured manner.

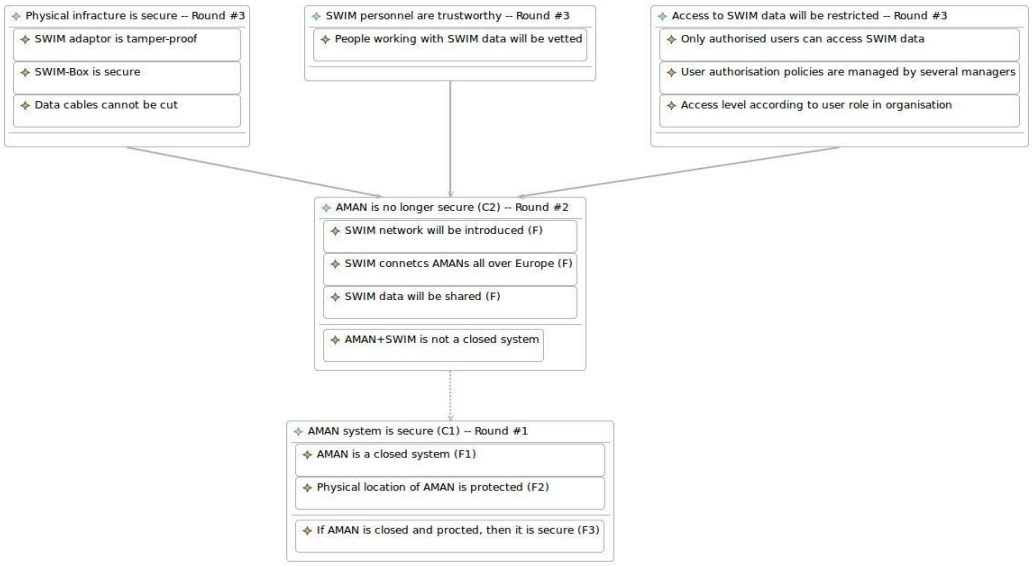

Figure 6: A fragment of a SeCMER argument model

Similar models are then used for the analysis of the Security Properties that concern the ATM changes requirements. **Figure 7** shows a Problem Frame model for the security property (**Security Property 2 – Information Protection**): *Protection of FDD (Flight Data Domain) info*. This supports the discussion how relevant security properties are affected by changes requirements. It stresses the necessity to change

required security properties in order to accommodate changes while maintain the same security level. The introduction of the AMAN and the SWIM Network requires additional security measures. Relevant emerging security properties could be, for instance: *'Queue Management Information shall not be accessible by meteo data centres'*, or *'Queue Management Information shall not be accessible by anyone other than those working with AMAN'*. The structured SeCMER's argumentation supports the verification of such emerging properties.

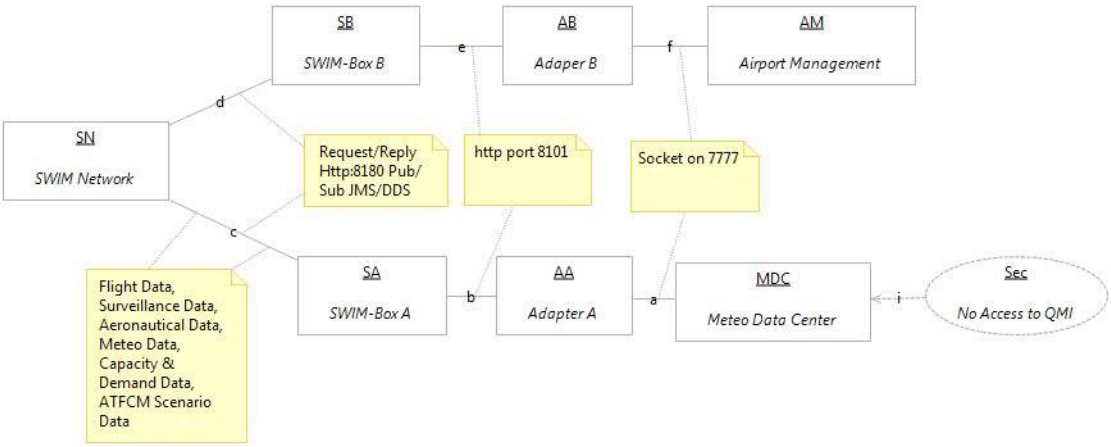

Figure 7: A SecMER requirement model with a relevant additional Security Property with respect to Changes

Finally, the SeCMER's evolution rules support the monitoring and deployment of changes requirements by identifying relevant actions and constraints.

Feasibility for WP3-WP2 collaboration

The integration between WP2 and WP3 will be based on the artefacts and process, and will be exemplified by the process level change requirement and its impact on the information access and protection properties of the ATM system. A specific scenario of this change is the introduction of SWIM technology to the AMAN systems, where the information protection property will have to be enforced by a combination of security means including role-based access control. For the purpose of integration with WP5 and WP4 we will also consider the organizational level change but not at same level of detail of the process level change.

3.5.2 Organizational Level Change

3.5.2.1 Feasibility for WP2

WP2 instantiates the Integrated Process on the ATM case study and will address the second change requirement: Organization Level Change. The security properties that are addressed are information protection and information provision. The technical solutions applied to the ATM case study are the instantiation of an Integrated Model (a specific System Model, a specific Risk Model and their Mapping Model). In addition we

will outline the use of state machines to capture and propagate changes throughout the Integrated Model.

Feasibility Arguments

WP2 is working on a management process that takes into account an architectural view of changes. We have collaborated with WP2 and WP5 in order to assess how structured account of changes enables us to better understand the impact of changes and any risk associated with them. WP2 has applied the Living Security Engineering Process (LSEP) on the ATM case study in order to outline the concept of a change driven security engineering process. The application of LSEP involved three main aspects: Target description before change (provides a description of the simplified system model before the change), Target description after change (provides a description of the simplified system model after the change) and Change handling (provides a step-by-step walkthrough on how change is handled in the Living Security Engineering Process). The target of the analysis is an Area Control Center and the activities of the Air Traffic Controllers in the arrival management process. The models were created during system modeling and risk modeling workshops with ATM experts. Structured (architectural) models are useful as a basis for a step-by-step walkthrough of the LSEP. These models are a subset of the larger set of models created during the application of the risk assessment methodology of WP5. The before and after models allowed a detailed analysis of changes requirements (at the architectural level) and a step-by-step account of the chances with respect to relevant security properties. The combination of an architectural view of the system together with a procedural view of the changes enables risk analysis too. For instance, after the addition and update of security objectives and their refinement in security requirements the activity of risk analysis starts. The risk analysis identified new risks (e.g., Critical A/C position data leaks to unauthorized third parties, Eavesdropping ADS-B communication, and Spoofing ADS-B data) in the context of the newly added model elements and related security requirements. **Figure 8** shows a structured (architectural) model capturing the new models introduced by the changes and their risk evaluated with respect to the security properties.

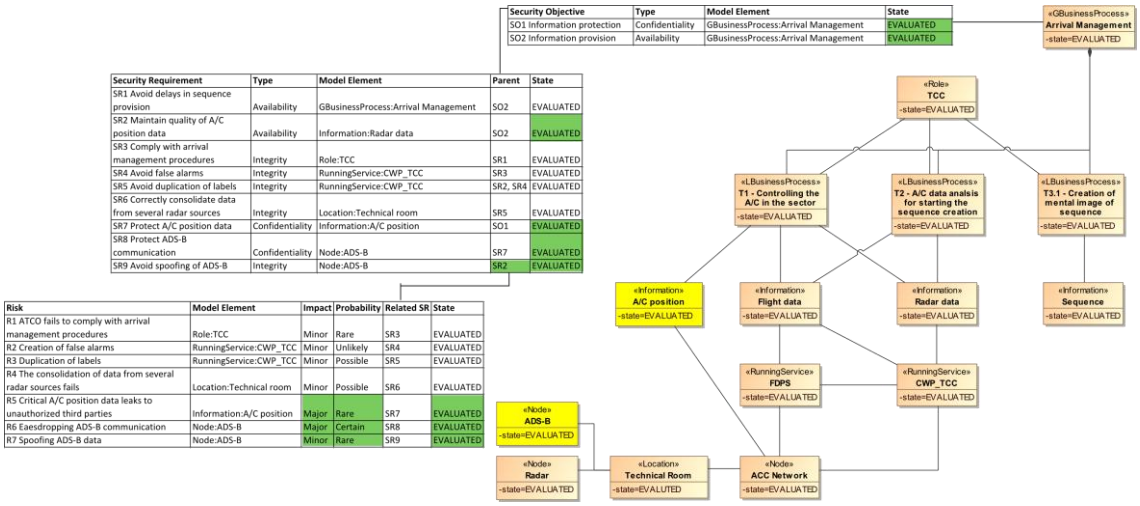

Figure 8: All model and security elements evaluated after changes

3.5.2.2 Feasibility for WP4

Focus on the Organizational Level change and the Information Protection and Information Provision security properties. We provide a demonstrator of the Thales Security Engineering tools including a Design modeling Demonstration of an application of the ATM case study to UMLseCh. This will focus on the Information Protection security property.

Feasibility Arguments

This section describes how the UMLseCh language and modeling tools can contribute to the design of an Arrival Manager secure interface for the different Air Traffic Controllers (ATCOs) roles in an ACC. The Arrival MANager (AMAN) is an aircraft arrival sequencing tool helping to manage and better organize the air traffic flow in the approach phase. Arrival Management is a very complex process, involving different actors. A high level description of the Arrival Management process involves:

- Setting Goals (e.g. maximum usage of runway capacity, minimizing noise or fuel consumption).
- Creating a plan to meet the goals.
- Monitoring the conformance to the plan.
- Adjusting/updating the plan if necessary.

Before AMAN introduction, the sequence creation and adjustment was carried out by the Sector Team, in particular by the Tactical Controller with the Planner Controller support. The main AMAN functionalities are:

1. The creation of an arrival sequence using ‘ad hoc’ criteria.
2. The management and modification of the proposed sequence.
3. The provision of data to the HMI to allow controllers to implement the proposed sequence.
4. The support of runway allocation at airports with multiple runway configurations.
5. The generation of advisories on: (a) Time to lose or gain, (b) Speed, (c) Top-of-descent, (d) Track extension, holding.

The computation of the sequence is carried out no more by the ATCOs, but by the AMAN tool itself. Moreover, a new role in the ACC has been introduced: the Sequence Manager (SEQ MAN), who will monitor and modify the sequences generated by the AMAN and will provide information and updates to the Sectors’ Teams. After AMAN introduction, ATCOs have different privileges according to their roles. For example, the Sequence Manager can modify the sequence of arrivals provided by the AMAN, while the Tactical (TCC) and Planner (PLC) can only view it. Thus, the AMAN tool needs different functionalities and subsequent access rules for different ATCOs roles. ATM Engineers customizing the AMAN for an ACC have the problem to design suitable Role-Based Access Control and Users Interfaces for the AMAN tool. UMLseCh modeling can help us with the evaluation of alternative solutions and designing the most secure system.

Figure 9 shows an UMLseCh diagram in which the SEQ MAN and AMAN are introduced, but the previous activities of the TCC are not deleted, thus resulting in an <<rbac>> violating evolution. This configuration poses a problem of overlapping and possibly conflicting actions for different roles. A security violation occurs.

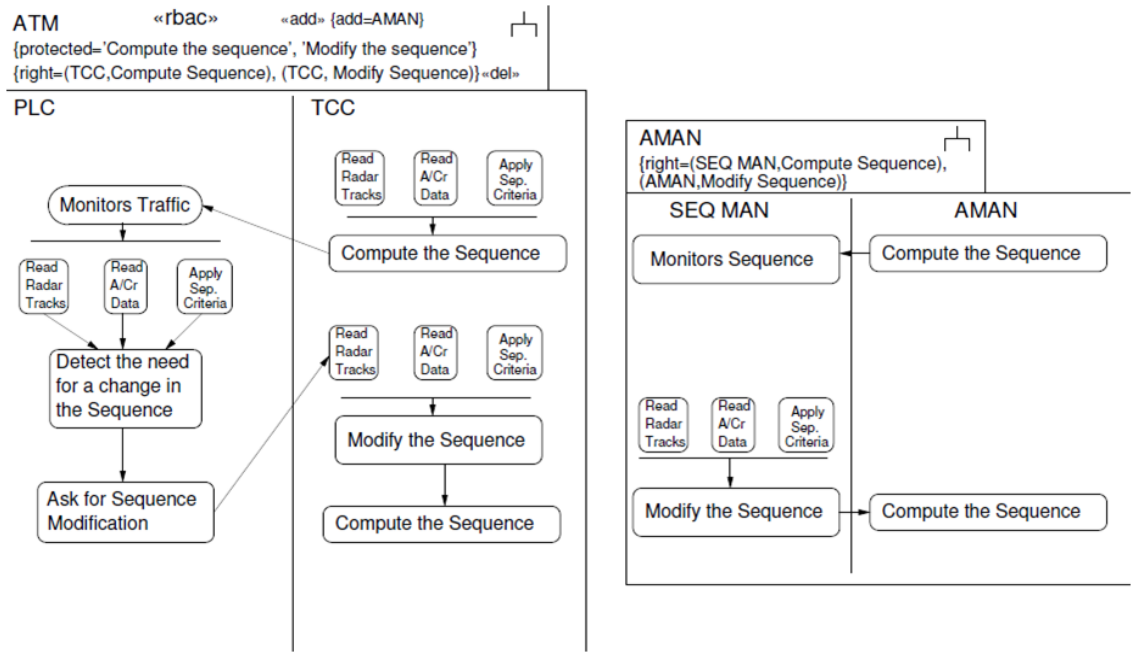

Figure 9: An UMLseCh diagram stressing a security violation

The UMLseCh language allows us to model and analyze alternative solutions .**Figure 10**, for instance, shows another solution. This second diagram captures that the TCC activities are deleted, and that TCC new activities are added, addressing thus the problem with the initial solution. This allows us to identify a solution that addresses the problem with the security violation. The problem could be solved by creating a graphical interface for the AMAN installed in the TCC and PLC that does not allow sequence modification.

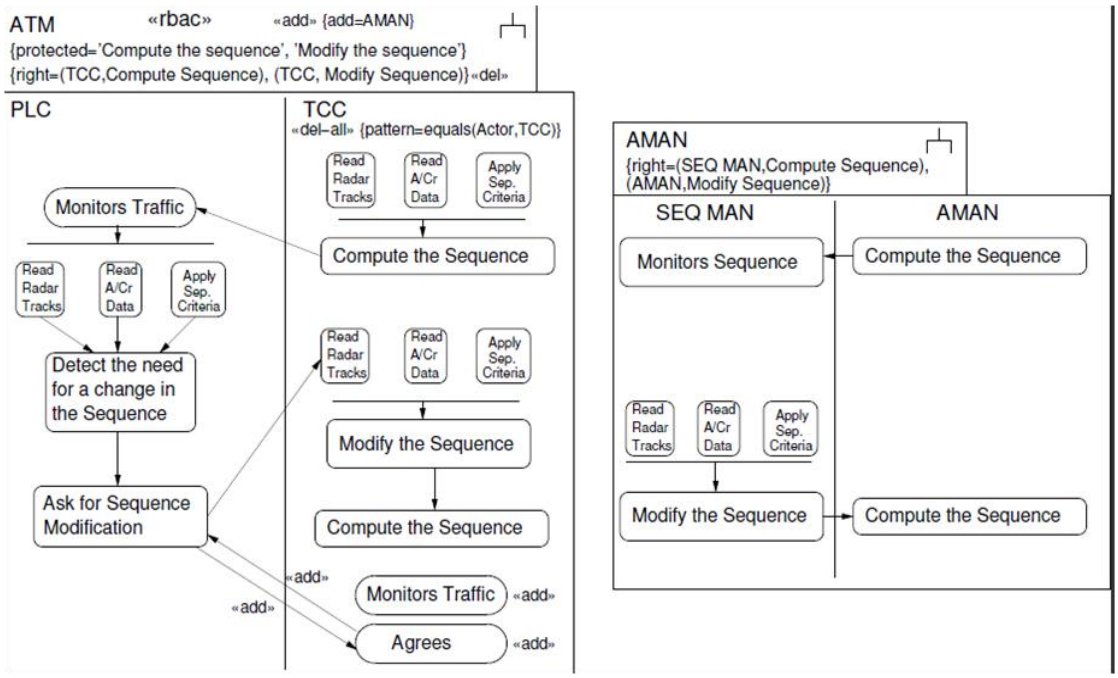

Figure 10: Another UMLseCh diagram

3.5.2.3 Feasibility for WP5

The change requirement that WP5 addresses is the Organization Level Change. The security properties that are mainly addressed are information protection and information provision. The technical solutions we use in the ATM case study are the modeling language for documenting and reasoning about changing risks, and the assessment method for conducting and documenting the risk analyses of changing and evolving systems.

Feasibility Arguments

This section describes examples drawn from ongoing WP5 work. We have collaborated with WP3 partners in order to gather feasibility arguments for WP5 with respects to our feasibility criteria (that is, *supporting structured approaches to changes*, *capturing security properties affected by changes*, and *providing mechanisms dealing with subsequent changes*). The feasibility trial involved a focused risk analysis of the ATM Changes Requirements and their relevant Security Properties. The risk analysis was conducted by means of design models capturing the main entities characterizing an ATM domain. In order to take into account how changes requirements affect the ATM contexts and their organizations, the WP5 partners produced structured (UML) models capturing the ATM settings *before* and *after* the changes. These models were reviewed and revised by ATM experts who are currently involved in various activities concerning the SESAR project. The models were used a starting point for the risk analysis in order to have a common understanding of the changes requirements among the people (i.e., ATM experts, WP1 and WP5 partners) involved in the risk analysis exercise. **Figure 11**, for instance, shows a conceptual model of an ACC after changes. Similar models have been drawn for other aspects charactering ATM settings and practices (e.g., models capturing different roles and procedures). These models supported discussion and communication between ATM experts and Risk Analysis

modelers. Moreover, they have been used to focus and organize the risk analysis on both *before* and *after* changes.

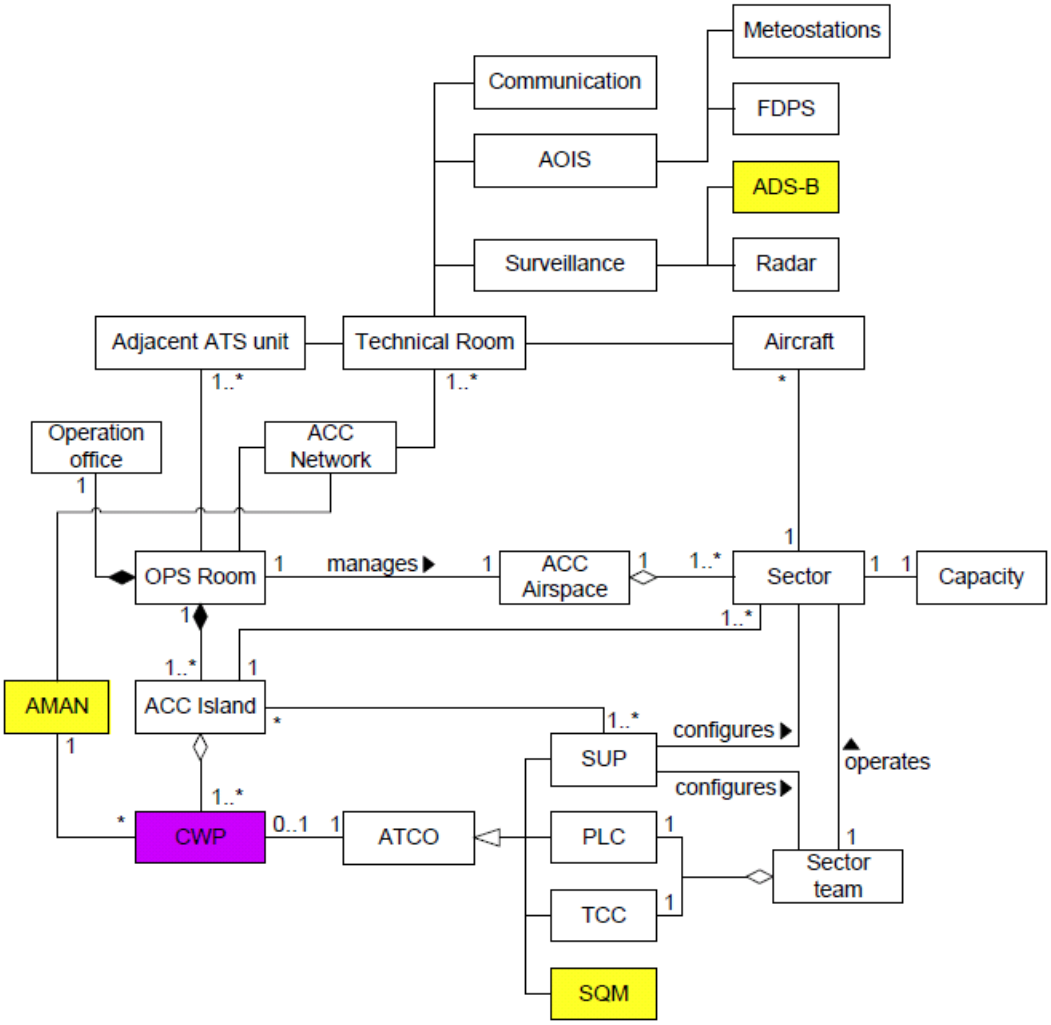

Figure 11: Conceptual overview of ACC after changes

The risk analysis trial was conducted during a dedicated workshop in Rome hosted at the Deep Blue's premises. The risk analysis was carried out over two days. The first day of the workshop was dedicated for the risk analysis of the *before* case. This phase supported the identification of specific hazards, and how ATM practices mitigate them. This allowed the communication of domain-specific knowledge about current ATM practices. The second day of the workshop was dedicated for the risk analysis of the *after* case. That is, the risk analysis of the changes requirements and how they potentially affect security properties. The remainder of this section discusses some of the risk analysis models obtained during the workshop as supporting arguments for our feasibility criteria. The first activity involved a high-level risk analysis of the AMAN introduction. The structured models were used in order to support a walkthrough analysis of the changes requirements and to identify potential hazardous situations.

			
Who/what caused it?	**What is the scenario or incident? What is harmed?**	**What makes it possible?**	**Target element**
System Failure	Loss of the AMAN leads to loss of provisioning of information to ATCO		AMAN
Attacker	Attacker broadcasts false ADS-B signals, which lead to the provisioning of false arrival management data.	Use of ADS-B; dependence on broadcasting	ADS-B
Software fail	Provisioning of unstable or incorrect sequence by the AMAN leading to ATCO reverting to manual sequencing	Immature software	AMAN

Table 2: Examples of hazardous situations

The subsequent risk analysis phases involved risk identification, risk estimation and risk evaluation. **Figure 12** shows sample risk analysis models for the after case. The model supports a structured risk analysis of changes requirements and their impact on critical security properties. Among the risk analysis outcomes were models assessing emergent risk due to the changes requirements and their impact on critical security properties. These models supported a systematic way of analyzing the risk of changes and their impact on security aspects. **Figure 12** shows one of such models. Note that the model captures different hazards and relate them to the target of analysis as wheel as to other relevant hazards. The resulting network of causalities is used in order to assess the risk of changes and related them to specific security properties (e.g., Availability as **Information Provision**).The same network of causalities is then used to assess the risk in terms of frequency of events and their severities. This is useful to revise risks with respect to emergent hazards, which are related to changes requirements. The final phases involved the identification and discussion of suitable mitigations for the analyzed hazards.

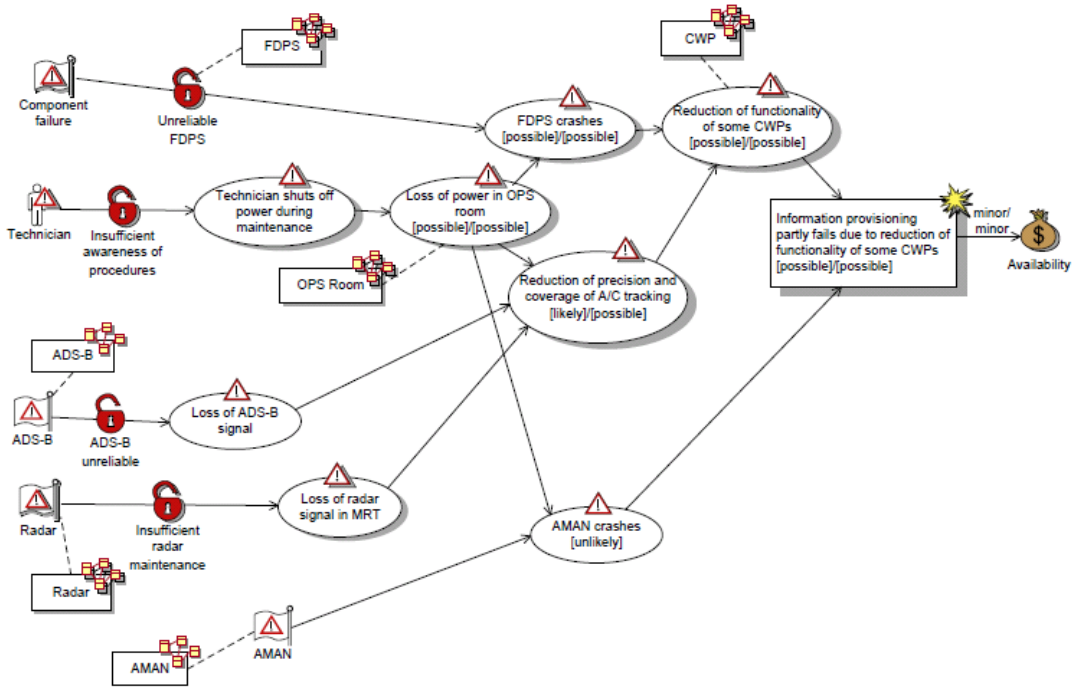

Figure 12: A sample risk model for reduction of functionality

ATM experts were involved in the risk analysis. They reviewed the models describing the change requirements and actively participated in the risk analysis trial. In order to account for model effectiveness as a means to investigate risk analysis with respect to change requirements, we collected relevant information about the experts' profiles and perceptions. At the beginning of the risk analysis trial, ATM experts as well as other project partners filled in a *Safety Culture Questionnaire*. The questionnaire has been developed and tailored by Deep Blue taking into account relevant information drawn from the ATM domain [4][5]. It covered ten different areas (e.g., Regulation and Standards, Safety Assessment, Safety Occurrence Report, etc.) by fifty three questions contributing to Safety Culture. The questionnaires aimed at profiling expert knowledge rather than assessing expertise. The reason we wanted to profile expert knowledge with respect to Safety Culture is because Risk Management and Change Management are often critical practices for an organizational culture of safety. Therefore, we collected Safety Culture profiles in order to understand further the relationship between safety and risk with respect to changes requirements and relevant security properties. **Figure 13** shows a Safety Culture Profile for one of the ATM experts taking parts in the risk analysis trials. The profile it is useful to analyses is perception of different safety aspects covered by the questionnaire.

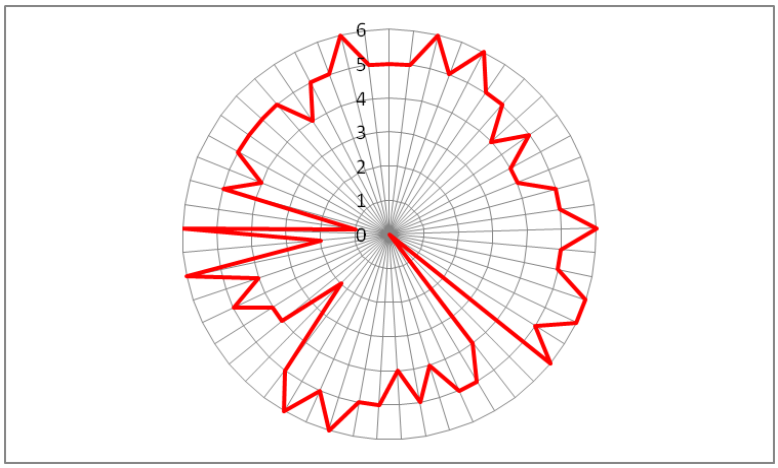

Figure 13: Safety Culture Profile

After each one of two risk analysis sessions (i.e., the before risk analysis on the current ATM practices and the after risk analysis taking into account changes requirements), we collected other information by an *Evolutionary Risk Questionnaire*. The questionnaire has been developed and tailored by Deep Blue in order to account of perceived hazards, hence risk perception, as captured by risk analysis models concerning current and future change requirements. The questionnaire consists of twelve different points drawn from relevant work in the ATM domain [6]. The questionnaire is concerned with Area of Changes (AoC) as a means to discuss relevant Changes Requirements and Hazards pertinent to current and future ATM. **Figure 14** shows some of the questionnaire statements.

		Strongly Disagree						Strongly Agree
1.1	This AoC increases the likelihood of well-understood current hazards that will exist in the future	0	1	2	3	4	5	6
1.2	This AoC creates new hazards synergistically with other AoC's or with the Future that would not have come into being without the presence of the AoC	0	1	2	3	4	5	6
1.3	This AoC increases the subjective likelihood of Future hazards to an unacceptable level	0	1	2	3	4	5	6
1.4	This AoC creates increased potential for human error, procedural non-compliance or equipment failure	0	1	2	3	4	5	6
1.5	This AoC decreases the resilience of the projected safety system	0	1	2	3	4	5	6

Figure 14: Sample questionnaire statements

Figure 15 shows the questionnaires' outcomes (for the same Safety Culture profile). It is interesting to notice how risk perceptions with respect to current situation and future

ones change. The dedicated risk analysis sessions helped to capture this shift in perception with respect to changes requirements. Moreover, the specific points highlighted by the questionnaires identify aspects for further investigation in order to refine and gain confidence on the risk analysis concerning future changes requirements.

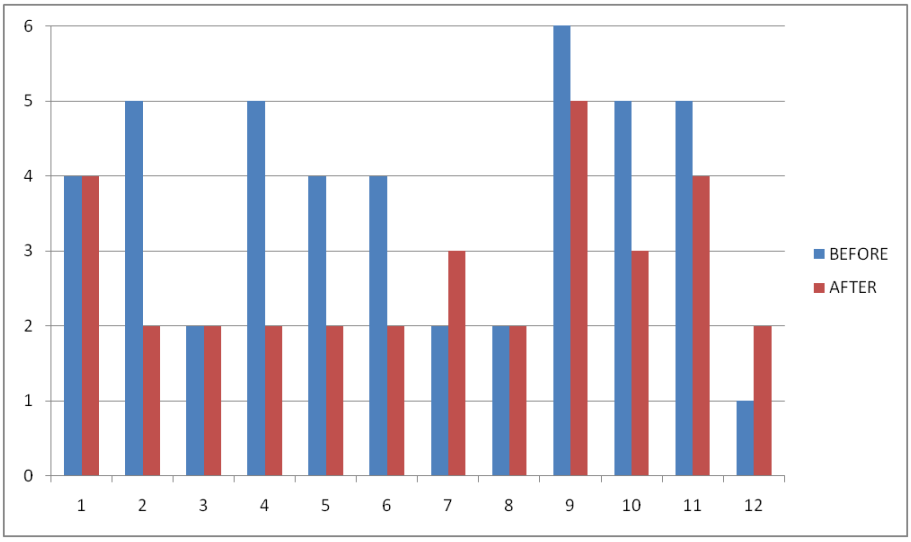

Figure 15: Evolutionary risk perception

Feasibility for WP2-WP5 collaboration

In both WP2 and in WP5 the second change requirement Organization Level Change is addressed. Both WPs are focusing on the security properties information protection and information provision. In WP2-WP5-Integration it will be outlined how the technical solutions of WP5 fit into the overall Integrated SecureChange Process. We will outline what artefacts are needed and provided as a result of the WP5 activity. In addition we will show how the WP5 methodology fits in the overall Integrated SecureChange Process methodology.

The risk assessment methodology will therefore serve as an example of how a specific methodology or solution can be integrated in the overall Integrated SecureChange Process. We want to show that a methodology can come with its own internal process of dealing with a change on the level of its own model. The Integrated SecureChange Process must be able to provide a flexible way of combining different methodologies.

Therefore the Integrated SecureChange Process should be abstract enough to not constrain the particular risk assessment methodology in any way. On the other hand we need to be concrete enough to provide traceability in the Integrated Model (via Mapping Models) in order to apply the principles of change driven security engineering.

Using the ATM case study as an example we outline how the specific approach of the risk assessment methodology with its own artefacts and activities is integrated in Integrated SecureChange Process.

Feasibility for WP3-WP4 collaboration

The feasibility study preliminary assessed how UMLseCh can be used to help with verifying that requirements are actually met by a system and that they are complete with respect to high-level security objectives. The feasibility study addressed the organization level change and the security properties of information protection and information provision.

Feasibility for WP3-WP5 collaboration

The integration between WP3 and WP5 will be at the level of artefacts and process. The major outcome will be an integrated change management process that combines requirements analysis and risk analysis steps. The integration will be based on the organizational level change and the information protection property.

3.6 Evaluation criteria

This section describes the different criteria that will be used to evaluate the specific security means when it is applied to the ATM case study. Those criteria come from an agreement and technical discussion with the partners.

During the third year of the project, the high level Evaluation criteria defined below may be broken down into more detailed and measurable Validation criteria for each foreseen Validation exercise. This process of decomposition has to be repeated several times resulting in a hierarchical structure of more and more detailed criteria instantiated in the various Validation activities. The decomposition of objectives ends with the identification of basic indicators and evidences to be measured and/or collected during Validation exercises. Note that indicators and evidences provided can be quite diverse. For instance, some indicators can be measurable in a quantitative way. Whereas, other evidences might highlight compliance with standards or development processes. Finally, other evidences can be just qualitative evaluations, obtained with the support of domain experts and practitioners. The results of the evaluation of the artifact with respect to those criteria will be described in the deliverable D1.3. For each criterion, some improvement direction will also be provided as the evaluation is with respect to a industrial context that could be too restrictive.

We will not provide additional criteria for the inter-WP collaboration and integration. Such criteria will be directly derived from the ones defined for each single WP by juxtaposition or intersection, depending on the foreseen linking activities.

3.6.1 For means provided by the WP2

The external criteria are evaluation criteria for the WP2 methodologies of change-driven security engineering process and change patterns. We provide criteria for applicability and human effort. The degree of fulfillment is given by categorizing the level of achievement of applicability and effort.

Change-Driven Security Engineering Process

- Applicability: The change-driven security engineering process can be applied to the ATM case study. We operate with the following increasing levels of fulfillment:
  - The change-driven process case study can be conducted by the researchers developing the methodology
  - The report documenting the results of the case study can be understood by the relevant stakeholders
  - The major principles of the change-driven process can potentially be established by a software provider
  - The principles of the change-driven process can be fully applied by a software provider
- Human effort: The second evaluation criterion is that the change-driven software engineering process can produce the desired results with less effort than by using alternative, traditional methods. We operate with the following increasing levels of achievement:
  - The steps of the security engineering process are doable, no matter the level of required human effort
  - Handling a change request with the change-driven security engineering process is doable with the same level of human effort as traditional methods and/or manual approaches
  - A change request can be handled with significantly less human effort than by using traditional methods and/or manual approaches

Tool-Support by MoVE Framework

- Applicability: The framework is applied to the Change-Driven Security Process. We operate with the following increasing levels of fulfillment:
  - An implementation of the framework is available and demonstrated with academic examples.
  - An implementation of the framework is available and is applicable to the ATM case study.
  - The implementation of the framework can be adopted by relevant stakeholders and applied to their tool landscape
  - The framework and its interfaces are adopted by software providers and further developed.
- Human effort: The second evaluation criterion is that a process supported by the MoVE Framework can produce the desired results with less effort than by using alternative, traditional methods. We operate with the following increasing levels of achievement:
  - The installation of the framework enables the implementation of a change driven security process.
  - The installation of the framework reduces the communication and synchronization overhead, reducing human effort.

3.6.2 For means provided by the WP3

SeCMER Modeling Language

- **Applicability:** The first evaluation criterion is that the SeCMER modeling language can be applied on the ATM case study for modeling and reasoning on evolving requirements.
- Both functional and security requirements characterizing the introduction of the AMAN must be modeled using SeCMER concepts
- Evolution of requirements associated with the introduction of the AMAN must be modeled using SeCMER concepts
- The requirement models related to the introduction of the AMAN must be analyzable by using reasoning techniques
- The requirement modeling must be computer aided
- **Human effort:** The second evaluation criterion is that the modeling of changing requirements in the ATM case study can be conducted with less effort than by using state of the art requirements modeling languages or techniques.
    - The modeling of changing requirements using SeCMER methodology is doable
    - The modeling of changing requirements using SeCMER methodology saves effort

SeCMER Methodology

- **Applicability:** The first evaluation criterion is that the SeCMER methodology can be applied on the ATM case study for modeling and reasoning on evolving requirements. We can identify several sub criteria for the applicability to the ATM case study
- The SeCMER methodology should consists of well defined, precise and easy to apply steps
    - Each step can be understood/applied by the researcher
    - Each step can be understood/applied by the stakeholder
    - Each step can be understood/applied by the stakeholder, at least partially
    - Each step can be understood/applied by the stakeholder, in complete independence
- Explicit linkage of produced artefacts with SeCMER methodology steps
- The methodology can be applied to the case study
    - Can be done by the researcher
    - Results can be understood by the stakeholder

    - Can be done by the stakeholder, at least partially
    - Can be done by the stakeholder, in complete independence

- **Human effort:** The second evaluation criterion is that the SeCMER methodology can be applied to the ATM case study with less effort than other existing requirement engineering approached.
- SeCMER methodology steps can be executed no matter the level of required human effort
- SeCMER methodology steps can be executed with the same level of human effort as traditional methods and/or manual approaches
- SeCMER methodology steps can be executed with (significantly) less human effort than by using traditional methods and/or manual approaches

SeCMER CASE Tool

- Technical Usability
    - Look and Feel
    - Learnability/Memorability
- User Acceptability
- Human Effort
- Presentation of Information
- Domain Applicability
    - The SeCMER CASE Tool can be used to model and analyse the case study
        - Can be done by the researcher
        - Results can be understood by the stakeholder
        - Can be done by the stakeholder, at least partially
        - Can be done by the stakeholder, in complete independence
    - Additional knowledge or research is required to run the SeCMER CASE Tool
    - The SeCMER CASE Tool cannot be used in the existing ATM software engineering processes
    - The SeCMER CASE Tool can be used only with revising the existing processes
    - The SeCMER CASE Tool can be used without major revision of the processes

    - A tool for requirement evolution management is already used
    - This SeCMER CASE Tool contributes to a better support for ATM evolution requirement management

- Impact of a change can be assessed by the SeCMER CASE Tool
- The SeCMER CASE Tool can present the analysis of the change in a usable format for end-users

3.6.3 For means provided by the WP4

- Effective Usage:
    - Overall well-defined system engineering process with clear steps and links
    - Compliancy with already existing tools, standard and/or work-practices in the ATM domain
    - Computer aided support for system modeling
- Usability and Applicability:
    - The research technique can be applied on the ATM case study
    - Results can be understood by the ATM domain expert
    - Can be done by the ATM domain expert, at least partially
- Required human effort
    - Equivalent to manual approach
    - Saves effort (in terms of time, workload and needed expertise)
    - Enhance the system models (providing further details and clearer modeling)

3.6.4 For means provided by the WP5

The main case study in WP5 is ATM, and it is therefore this case study that provides the most thorough basis for the evaluation.

3.6.4.1 Effective Usage

The criteria of effective usage of the artifacts require that the artifacts can be applied in the ATM case study. We provide evaluation criteria for applicability and for the required human effort. The degree of fulfillment is given by categorizing the level of achievement of applicability and effort.

3.6.4.2 Risk assessment methodology

Applicability: The first evaluation criterion is that the risk assessment methodology and its techniques can be applied on the ATM risk assessment. We operate with the following increasing levels of fulfillment:

- The ATM risk assessment can be conducted by the researchers developing the methodology
- The report documenting the results of the case study can be understood by the relevant ATM stakeholders such as the external risk assessment participants
- The ATM risk assessment can be conducted only partially by a risk analyst trained in traditional risk assessment methods
- The ATM risk assessment can be fully conducted by a risk analyst in complete independence

Human effort: The second evaluation criterion is that the risk assessment methodology and its techniques can produce the desired results with less effort than by using alternative, traditional methods. We operate with the following increasing levels of achievement:

- Conducting the ATM risk assessment is doable, no matter the level of required human effort
- Conducting the ATM risk assessment is doable with the same level of human effort as traditional methods and/or manual approaches
- The ATM risk assessment can be conducted with (significantly) less human effort than by using traditional methods and/or manual approaches

3.6.4.3 Risk modeling language

Applicability: The first evaluation criterion is that the risk modeling language can be applied on the ATM case study for modeling and assessing changing risks. We operate with the following increasing levels of fulfillment:

- The consistent and syntactically correct modeling, as well as the semantically correct interpretation, of the ATM risk models can be conducted by the researchers developing the risk modeling language
- The ATM risk models can be understood by the relevant stakeholder both during the risk identification and assessment, and as part of the documentation of the results
- The consistent and syntactically correct modeling, as well as the semantically correct interpretation, of the ATM risk models can be conducted only partially by a risk analyst trained in traditional risk modeling
- The consistent and syntactically correct modeling, as well as the semantically correct interpretation, of the ATM risk models can be conducted by a risk analyst in complete independence

Human effort: The second evaluation criterion is that the modeling of changing risks in the ATM case study can be conducted with less effort that by using traditional risk modeling languages or techniques. We operate with the following increasing levels of achievement:

- Conducting the modeling of changing risks is doable, no matter the level of required human effort
- Conducting the modeling of changing risks is doable with the same level of human effort as using traditional risk modeling languages or techniques
- The modeling of changing risks can be conducted with (significantly) less human effort than by using traditional risk modeling languages or techniques

4 HOMES Case Study

HOMES is focused on digital home networks where some sensible changes take place from the point of view of the security. We consider some changes, from the large set of changes that anyone may identify in this context, very related to configuration and deployment. Our target is the home gateway as a critical point in the home network architecture.

4.1 Change Requirements

4.1.1 Core Security Module Update

Home Gateway has some security modules implementing, for instance, NAC functional components like the PEP. NAC technology [7] and its functional elements are properly described in the deliverable D1.1. During the lifecycle of the whole system some component updates shall be required for various reasons (better performance, bug fixes, etc.). Updating one of these core security modules in the home gateway is a critical operation and a relevant change. Any attack or failure in this process may result extremely harmful.
There shall be many reasons driving to an update. Just as an example, a possible update on the core security modules could be the extension of information for the security assessment (more information in deliverable D1.1). In these cases, the home gateway needs to be updated so that the new security status information is understood and assessed correctly.

Goal: Show that the security properties detailed below are still preserved after an update of a security module

4.1.2 Bundle Lifecycle operations

A Home Gateway is also a service platform for the home. Customers can install new home services, upgrade or delete existing ones. These changes are expected but not scheduled (they mostly depend on customer will). Those services may come from third parties and therefore some control over this software may be required. Also, the addition of new services may cause lateral effects what shall be avoided.

Along with these services we consider the special case of security services that are not controlled by the Customer but the Operator. In this case the security services are managed following a kind of schedule driven by trust relationships.

In the business model of HOMES case study, the main stakeholders are the Customer, the Operator and the Third Party Service Providers. The trust relationship among these parties is taken in count in terms of security: those relationships shall determine the level of security required to the third parties to deploy their services in the gateway. This trust relationship may evolve over time. We can assume that, by default, once the Operator and the SP sign a commercial agreement, it trusts the SP and its services.

This trust is translated into a basic level of control over the SP and its services, i.e. Operator does not impose strict constraints to the services. Nevertheless, this trust might **degrade** with the pass of time. Operator shall degrade the trust on a certain SP because of several reasons (reports on bad quality of the offered services, critical bugs into the services or even malware, non delivery of services, etc)

The trust degradation shall drive to the imposition of severe constraints to that SP in the form of strict security requirements that mitigates the mentioned threats:

1. **delivery of certified bundles only**: due to the new trust relationship between operator and third party service provider, the operator requests that only certified bundles of this operator may be deployed on the home network
2. **deployment of a new security service**: due to the new trust relationship between operator, third party service provider, and customer, the operator requests that a non repudiation protocol may be run between the parties to prevent denial of having subscribed, received or delivered a service

Goal: Bundles have to be managed (update, addition, removal) in compliance with the trust relationships and assuring system consistency, i.e. the security properties need to be preserved despite these changes.

4.2 Security Properties

The following properties will be the focus of the technical WPs.

Secure extensibility. The home gateway can be extended at run time with additional general software, coming from third parties in many cases. Such extensions should be verified to be secure in the sense that they do not introduce unauthorized information leaks or the possibility of denial of service

Policy enforcement. The Policy Decision Point (PDP) is located in the security domain of the operator. The Policy Enforcement Point (PEP) is a core security module installed on the home gateway. The PEP always enforces policy decisions forwarded by the PDP so that only allowed actions can be carried out.

Resilience to trust changes. The system shall be able to accommodate a change in the trust relationships (among service provider, customers, 3rd parties) with a minimal impact on the software architecture

Security expandability. System security can be enhanced by taking advantage of the home gateway extension ability (mentioned in the Secure Extensibility property) through the deployment of new security services to meet new security needs (e.g., deployment of a non-repudiation service bundle required to low-trusted services). The infrastructure shall be able to efficiently enforce such new requirements without causing a regression of the security functionality of the system

4.3 Security Means

The Security Means that will be used for the proposed Change Stories of the HOMES Case Study will be the ones provided by WP2, WP5, WP6 and WP7, namely:

WP2

- A complete SeAAS (Security as a Service) deployment for the HOMES case study supporting the addition of new security functionalities.
- A tool to assess the impact of trust changes into the system

WP6

- A tool to validate some core security modules (programs written in C language)
- A methodology to verify the "software contract" of OSGi bundles

WP5

- Creation of risk model for HOMES
- Identification of treatments to address risks

WP7

- Creation of a test model for HOMES
- Creation of test suites to study:
  - The impact in policies and enforcement changes
  - The impact of the inclusion of a new security service (check for lateral effects)

4.4 Feasibility Criteria

Here we describe the feasibility criteria applied into the HOMES case study. In our specific case we have a slightly different approach in studying the feasibility of the Secure Change technologies than the ATM case study. In our context, we want to find out whether specific solutions (tools, methodologies, etc.) are really applicable to our scenario. In other words, we need to know if current output from technical WP is in fact applicable on the current environment provided by us as case study owners. Thus, we are mostly talking about bringing the technical results to real application in our scenario. Depending of the kind of result delivered by technical WP, we may need to determine:

Technical feasibility of usage in HOMES: decide if current release of the results from technical WPs is really applicable or identify critical technical issues preventing to do so.

Theoretical feasibility of methodologies in HOMES: for non-software results, we have to determine if current methodologies fits well into the case study or identify theoretical gaps not allowing us to use them

4.5 Feasibility Studies

	Change REQ 1 Core Security Module update	Change REQ 2 Bundle Lifecycle operations

Sec. Prop 1 **Secure Extensibility**	**WP6**	**WP6**
Sec. Prop 2 **Policy Enforcement**	**WP7**	**WP5, WP7**
Sec. Prop 3 **Resilience to trust changes**		**WP2**
Sec. Prop 4 **Security expandability**		**WP2,WP5,WP7**

Table 3: HOMES Requirements & security properties

This Section shows the feasibility exercises we have conducted for each WP working on the HOME changes requirements and their security properties.

4.5.1 Core Security Module Update

4.5.1.1 Feasibility for WP6

WP6 is focused on the “Secure extensibility” property and the main targets are the core security modules written in C. This WP is delivering a tool to analyze some critical security software. This kind of assessment fits very well with the mentioned security property.

In this study the aim is to focus in the PEP module, which is a key component of the NAC system deployed in HOMES. To check the feasibility, the criteria is to determine whether the current tool is able to analyze some HOMES core security model code to at least detect one type of error.

TID has shared the proprietary source code of the PEP module to let WP6 (KUL) asses the possibility of using the verification tool on it. The goal is to check potential infinite loops, core faults, etc. The assessment is currently on its way.

4.5.1.2 Feasibility for WP7

WP7 works on “Policy enforcement” property. The goal is to have test model and, upon some eventual change on the system or requirements, identify the affected tests and derive tests suites. In this case the targets are policy-related and enforcement-related tests.

Here the feasibility check is softer; we just need to come up with some applicable test models. There are not software artefacts.

The criterion is to be able to create working tests for HOMES to study the impact of changes/evolution covering the previously indicated security properties.

Currently, WP7 has achieved that goal and the feasibility can be considered checked.

4.5.2 Bundle Lifecycle Operation

4.5.2.1 Feasibility for WP2

WP2 is working in two security properties: “Security expandability” and “Resilience to trust changes”.

With regards to the **“Security expandability”** property, WP2 wants to apply the Security as a Service (SeaaS) paradigm in the HOMES prototype to allow easy security functionality extensions to the overall system. Basically, the challenge is taking the actual HOMES prototype (mostly the Home Gateway) and improving it by integrating a SeaaS engine. This engine shall allow the addition of new security functionalities with least impact (harmful side effects) possible. WP2 plans to deploy a non repudiation service specifically, to enhance a deployed News Feed service.

The criterion in this case is clear: the feasibility is determined by the possibility to apply the SeeaS paradigm into the HOMES prototype. In this case we are talking about an actual implementation with software artefacts and integration of components. Thus the feasibility depends on the actual ability to integrate the SeeaS architecture within the current prototype. The integration implies several tasks mostly related to software development and integration. Right now, there is a work plan set by TID and UIB to deploy all the required elements.

Tasks to reach the goal are distributed between TID and UIB. Technological choices have been made and approved by both partners. Technical issues that may arise are linked to the extremely small footprint of the current HOMES hardware. Nevertheless, this issue is more of a practical concern rather than a conceptual one. The risk is mitigated by the fact that the platform is basically developed to run on an emulator with identical technical properties than the current hardware but adjustable in terms of memory. The current deadline is end of December.

The second security property being covered is **“Resilience to trust changes”** and the related activities consist in instantiating a change scenario (with impact at architectural level) by means of the Change Pattern methodology and supported by a prototype. The prototype helps foreseeing the potential impact of change and automates the evolution.

Currently the prototype has been developed and a catalogue of Change Patterns (with regard to evolution of trust relationships) has been implemented in the prototype.

Also, a change story in the context of the HOMES case study has been identified. The change story refers to the degradation of a trust relationship between the between network operator and a 3rd party service provider.

The feasibility is determined by supporting the change story by means of the prototype and the companion methodology based on Change patterns.

4.5.2.2 Feasibility for WP5

The Operator (depicted as User in the change story) orders the risk analysis team to update the existing risk analysis and find reasons and causes for the increasing number of complaints. The risk analysis team applies a risk assessment from a maintenance perspective and provides an updated risk picture, including newly identified threat scenarios and proposed treatments.

The treatment is considered as an actual change to the system and therefore analyzed from a before-after perspective by the risk analysis team. The resulting risk models depict the risk before the application of the treatment and the risk after the application of the treatment. In addition the risk to change is analyzed.

The results of the risk analysis are threat scenarios that serve as input to the test engineers in order to derive test cases. In addition the risk to change diagram provides the basis for the test engineers to derive regression tests, which are run after the application of the system change.

The criteria in this case are whether the risk assessment methodology developed in WP5 is suitable to depict risks at the right level of abstraction for the HOMES case study.

4.5.2.3 Feasibility for WP6

. The feasibility criteria are to map that WP6 technology into bundle manifest what are part of the bundle.

Feasibility study results are not available by the deliverable deadline and nothing could be concluded since the actual code of the OSGi bundles was not available to conduct the study. It is expected to have the code during the next weeks, and therefore proceed with the study and get the conclusions before the next review.

WP6 (on-device) is interested in the Secure Extensibility property. The main concern is control of interactions between Java OSGi bundles with respect to permitted/forbidden information flow paths, especially between bundles provided by different stakeholders.

WP6 will deliver techniques that verify at loading time absence of illicit information flow paths between bundles. The techniques should be incremental. Thus, when a change in the system occurs, verification should be optimized and advantages over complete system re-verification should be demonstrated.

The feasibility criterion for WP6 (on-device) is applicability of (some of) proposed loading-time verification techniques to the OSGi bundles. Additional metadata required by the verification process should be placed into bundle manifests. Verification algorithms should capture illicit interactions of bundles (for chosen definition of interaction) and be sound and fast.

As on the moment of writing the current deliverable D1.2 and D6.3 and D6.4 of WP6 (on-device) the code of sample OSGi bundles was not ready yet and the detailed specification of the bundles interactions was not yet presented, the details of applicability study will be presented later.

4.5.2.4 Feasibility for WP7

WP7 works in two security properties related to Bundle lifecycle operation: “Policy enforcement” and “Security expandability”. The situation is pretty similar to the previous change requirement. The basis is the same, but having different targets; service policies for the former and deployment of new security services for the latter.

Feasibility for WP5-WP7 collaboration

The interaction activity is related to “Security expandability” property. It is about creating mapping models to relate artefacts from WP5 and WP7 to each other. Some specific artefacts are to be considered from each WP to the other. WP5 shall provide an attack model and risk values upon a new security feature addition. Then WP7 can generate new tests and prioritize existing tests according to risk values. The test results are used as feedback for the risk model and the risk values are updated accordingly.

The criterion for the WP5-WP7 integration based on HOMES targets towards the actual usability of risk model artefacts for the work of the test engineers. We have identified several potential integration points between these two solutions. Using an actual application on the HOMES case study it is shown how the test engineers can derive regression tests from a risk to change model. In addition we outline how generic risk models can be used by the test engineers to develop new tests. Treatments provide another potential input which is used to derive functional security tests. Vice versa the results of the tests can be fed back to risk models to update risk values accordingly by confirming the risk reduction of a specific treatment. In addition specific tests could confirm whether vulnerabilities or a specific attack vector are still exploitable.

- Concrete WP5 artefacts used as input for WP7 artefacts are:
  - Attack-/threat model
  - Risk values
  - Treatments
- Concrete WP7 artefacts fed back to WP5 artefacts are:
  - Test results of attacks directly affecting risk value
  - Test results of treatments directly affecting risk value

4.6 Evaluation Criteria

This section describes the different criteria that will be used to evaluate the specific security means when it is applied to the HOMES case study. Those criteria come from an agreement and technical discussion with the partners.

During the third year of the project, the high level Evaluation criteria defined below may be broken down into more detailed and measurable Validation criteria for each foreseen Validation exercise. This process of decomposition has to be repeated

several times resulting in a hierarchical structure of more and more detailed criteria instantiated in the various Validation activities. The decomposition of objectives ends with the identification of basic indicators and evidences to be measured and/or collected during Validation exercises. Note that indicators and evidences provided can be quite diverse. For instance, some indicators can be measurable in a quantitative way. Whereas, other evidences might highlight compliance with standards or development processes. Finally, other evidences can be just qualitative evaluations, obtained with the support of domain experts and practitioners. The results of the evaluation of the artifact with respect to those criteria will be described in the deliverable D1.3. For each criterion, some improvement direction will also be provided as the evaluation is with respect to a industrial context that could be too restrictive.

We will not provide additional criteria for the inter-WP collaboration and integration. Such criteria will be directly derived from the ones defined for each single WP by juxtaposition or intersection, depending on the foreseen linking activities.

4.6.1 For means provided by the WP2

Within WP2 the HOMES case study focuses on the application of the **Security-as-a-Service** Architecture concept and **the change patterns** methodology and tools. In the sequel we give internal and external evaluation criteria with several levels of fulfillment.

4.6.1.1 Security-as-a-Service Architecture

4.6.1.1.1 Effective usage

Applicability

The SeAAS approach is applicable to the HOMES case study. We identified the following levels of completion in increasing order:

- Involved researchers are able to apply the SeAAS approach to the HOMES case study.
- The infrastructure can be configured by relevant stakeholders.
- The infrastructure and the model driven configuration approach can be adapted and extended by relevant stakeholders.
- The SeAAS approach can be realized by a software provider.

Human effort

The SeAAS approach supports efficient configuration of target platforms by relying on a model driven code generation process:

- infrastructure can be (re-)configured through a model based approach with less effort than by working with declarative security at the code artefact level.
- the infrastructure and the model driven code generation process can easily be extended to cope with new security requirements with less effort than by working with declarative security at the code artefact level.

4.6.1.2 Change-Patterns Methodology and Tool Support

4.6.1.2.1 Effective usage

Applicability

- The Change Pattern methodology can be applied by a researcher on the case study
    - Yes/no

Human effort

- The required human effort to perform the Change Pattern methodology makes it usable in practice
    - Yes/no
    - Appreciation of effort

4.6.1.2.1.1 Specific Industrial criteria

- The results of the Change Pattern methodology are of value to the industrial stakeholder
    - Yes/no
    - Appreciation of value

4.6.2 For means provided by the WP5

In principle, external criteria corresponding to the ATM external criteria of WP5 apply also to the HOMES case study. However, HOMES serves only as a minor case study in WP5 for providing further example cases and demonstrating integration scenarios with WP7. The HOMES case study nevertheless provides some basis for evaluation.

In the following we have omitted the human effort criteria for the methodology, as a fully fledged and proper risk assessment was not conducted for HOMES, which is required for this aspect to be evaluated.

4.6.2.1 Risk assessment methodology

4.6.2.1.1 Effective Usage

Applicability: The first evaluation criterion is that the risk assessment methodology and its techniques can be applied on the HOMES risk assessment. We operate with the following increasing levels of fulfillment:

- The HOMES risk assessment can be conducted by the researchers developing the methodology
- The report documenting the results of the case study can be understood by the relevant HOMES stakeholders

- The HOMES risk assessment can be conducted only partially by a risk analyst trained in traditional risk assessment methods
- The HOMES risk assessment can be fully conducted by a risk analyst in complete independence

4.6.2.1.2 Specific Industrial criteria

For this methodology we only consider as the main criterion a seamless integration in the industrial process. To achieve this, the risk assessment shall be performed easily by stakeholder staff (in this case TID). This relates with the effective usage. From an industrial point of view, the optimal result should be to have a methodology easily applicable by non-experts with the support of tools. The minimum required result is to have a methodology easy to understand and to learn by the stakeholder.

4.6.2.2 Risk modeling language

4.6.2.2.1 Effective Usage

Applicability: The first evaluation criterion is that the risk modeling language can be applied on the HOMES case study for modeling and assessing changing risks. We operate with the following increasing levels of fulfillment:

- The consistent and syntactically correct modeling, as well as the semantically correct interpretation, of the HOMES risk models can be conducted by the researchers developing the risk modeling language
- The HOMES risk models can be understood by the relevant stakeholder both during the risk identification and assessment, and as part of the documentation of the results
- The consistent and syntactically correct modeling, as well as the semantically correct interpretation, of the HOMES risk models can be conducted only partially by a risk analyst trained in traditional risk modeling
- The consistent and syntactically correct modeling, as well as the semantically correct interpretation, of the HOMES risk models can be conducted by a risk analyst in complete independence

Human effort: The second evaluation criterion is that the modeling of changing risks in the HOMES case study can be conducted with less effort that by using traditional risk modeling languages or techniques. We operate with the following increasing levels of achievement:

- Conducting the modeling of changing risks is doable, no matter the level of required human effort
- Conducting the modeling of changing risks is doable with the same level of human effort as using traditional risk modeling languages or techniques
- The modeling of changing risks can be conducted with (significantly) less human effort than by using traditional risk modeling languages or techniques

4.6.2.2.2 Specific Industrial criteria

No specific industrial criteria are considered in this case, beyond the effective usage.

4.6.3 For means provided by the WP6

4.6.3.1 Wp6 Development-time verification

4.6.3.1.1 Effective usage

In this case we need to evaluate the algorithm and tool provided by WP6 in terms of industrial applicability.

The effective usage of the tool is partially covered by the feasibility study, but here we go one step beyond, assessing the true application of the algorithm and tool. In this sense we need to evaluate the following:

1. The tool can be used on the case study, applying it on actual HOMES PEP source code.
    - Can be done by the researcher
    - Results can be understood by the stakeholder
    - Can be done by the stakeholder, at least partially
    - Can be done by the stakeholder, in complete independence
2. Required human effort
    - Doable
    - Equivalent to manual approach
    - Saves effort

4.6.3.1.2 Specific industrial criteria

Now we focus in the application trade-off. Here we need to determine if the solution is worth for the stakeholder (TID in HOMES). The criteria here shall be:

- Flexibility: the tool can work with different security modules (current and future) without major changes (no re-compilation, only re-configuration)
- Effectiveness: The tool throws a good enough number of true positives rate, i.e. it is over a reasonable threshold.
- Usability; the tool is easy to use for non-experts (people with few or no knowledge at all about the algorithm)
- Performance: a minimum performance rate is required to make the tool worth to use. This requirement is not really high in HOMES but at least some minimum threshold shall be met.

4.6.3.2 WP6 on-device verification

4.6.3.2.1 Effective usage

Here we may consider the same criteria than in the previous case.

4.6.3.2.2 Specific industrial criteria

Here we need to evaluate the methodology to detect non-permitted data exchange (according to bundle specification). In this case we have a different context leading to different criteria:

- Performance: in this case we have higher requirements since we are dealing with a runtime situation.
- Effectiveness: The tool throws a good enough number of true positives rate, i.e. it is over a reasonable threshold.
- Level of automation: we need fully automatic procedure for an optimal result, or at least with very small human interaction to achieve the minimal positive result.

4.6.4 For means provided by the WP7

4.6.4.1 Effective usage

In this case we need to evaluate the test model and test suites provided by WP7.

Applicability

The test model and test suites can be used on the case study, applying it on actual HOMES prototype.

- The test model can be done by the researcher, including all the relevant entities
- The model can be understood by the stakeholder
- The model can be done by the stakeholder, at least partially
- The model can be done by the stakeholder, in complete independence

The same applies to test suites

Human effort

- Doable
- Equivalent to manual approach
- Saves effort

4.6.4.2 Specific Industrial criteria

Completeness: test model shall be able to include all the relevant entities in the system (from the point of view of security)

Usability: test suites shall be easy to run by stakeholder (TID).

5 POPS Case Study

The POPS case study is based on an USIM card used for mobile payment. In particular, POPS deals with the embedded software on this card, which is made of the OS, the Java Card platform [8][9][10] and the Global Platform component [11]. To be used for mobile payment, this embedded software includes at least two applications: an USIM application for pure mobile service and a payment application.

The main hypothesis is that the USIM card has been certified with respect to **Common Criteria** security certification[13]. This means that the embedded software on this device ensures a set of properties related to (at least) **confidentiality**, i**ntegrity** and **availability** of its assets (but also non-repudiation, authentication, etc). The properties related to the robustness of the security mechanism (e.g. robustness) are out of the scope of this project.

But this embedded system will evolve during its life cycle since it has been built as an “open card” to allow the loading of new applications in the field (after it issuing to the card holder). Strictly speaking, the **Common Criteria** impose that any change requires to a re-certification of the card, or a justification that the change has no **security impact**. Several working groups are investigating means to alleviate the process while achieving this CC requirement. Moreover, the evaluation process is expensive in term of cost and delay, specifically for this kind of complex product.

In this context, the SecureChange project may provide means to speed up the re-certification of the card or built strong justification on the security impact of the change. The means may be any kind of artifact that can be used for the evaluation or for the justification: model, proof, test suites, tracing elements, etc.

The objective is then to demonstrate this USIM card, after two realistic scenarios of changes described in Section 5.1, still ensures several security properties described in Section 5.2.

5.1 Change Requirements

5.1.1 Specification evolution

The UICC card provides a component called the card manager implemented according to Global Platform specifications v2.1[11]. This card component has been extensively verified and tested. The Global Platform specifications have been enhanced and extended and v2.2 has been issued. The card manager software component has been updated and extended against this new version. For simplicity reason, we restrict the 2.2 scope to the UICC configuration [12].

The goal is prove/demonstrate/test that the security properties are still preserved. For that we will concentrate on specific properties detailed in §5.2.

5.1.2 Software update

The certified UICC card is deployed in the field. The mobile operator, owner of the card, has a new partner, a bank. The bank installs its representative which is a new security *domain (a special kind of Java Card application, whose code is resident on the card)* on the UICC (card) using an OTA (Over-The-Air) mechanism. This bank will have the delegated management privilege from the Mobile Network Operator to manage its applications in a **confidential** way. In particular, the bank will use its security domain to load an e-purse on the card[1].

The goal is to prove/demonstrate/test that the new application preserves (does not break) the consistency of the existing and implemented security policies. Again the specific properties are detailed in §5.2.

5.2 Security properties

Deny of service: Any application on the card does not generate a DoS. This means that some robustness properties must hold for the applets, such as absence of runtime exception, absence of infinite loop. Also the memory consumption must be bounded (in order to avoid memory overflow, and memory access especially update operations) due to the durability of the EEPROM and the Flash. For example, the call-stack should be bounded and the loops that update the persistent memory should be handled with care.

Information protection by Access control: Any command received by the card must respect the card and applet lifecycle. Its means that any command received in a state s leads to a state s' and the resulting transition from s to s' is correct w.r.t. the specifications.

Information protection by Flow control: The applications on the card must be "isolated" (application segregation) i.e. no illegal access to the data from one application to another. In order to enforce isolation, several security policies are described and assumed to be implemented on the card, like the Java Card firewall (access control implemented by the virtual machine) or the security domains of GP (key isolation relying on the underlying Java Card firewall and the GP API). Therefore, some properties must be verified, when an applet is added on the card, like the consistency of the security domain hierarchy, the non-violation of the information flow policy implemented on the card, etc.

Secure communication: A secure channel protocol (SCP) provides a secure communication between an on-card application and the off-card entity during a working session. It means that the protocol must ensure the authenticity, the integrity and the confidentiality of the transmitted data.

[1] Adding an an application "into" a security domain means that the bank system and this application comunicate using the same cryptographic key that is handled by the security domain.

5.3 Security means

The Security means that will be used for the proposed change requirements of the POPS Case Study will be provided by WP3, WP4, WP6 and WP7 as shown in the following table:

	Change REQ 1 **Specification Evolution**	**Change REQ 2** **Software update**
Sec. Prop 1. **Denial of service**		**WP6**
Sec. Prop. 2 **Information flow control**		**WP6**
Sec. Prop. 3 **Information access control**	**WP7,WP3**	
Sec. Prop. 4 **Secure communication**	**WP4**	--

Table 4: POPS Requirements & Security properties

WP6: Verification

- Development-time verification of a Java Card applet
  - A tool, VeriFast, that uses the source code of an applet and the properties are expressed using annotations. The two properties considered are the absence of runtime exceptions and infinite loops
- On-device verification:
  - The first approach consists in information policy checker
  - The second approach is security-by-contract

WP7: Model-based Testing

- A model-based tool for test generation (LTD) and a testing methodology
- Properties related to the card lifecycle, and to the hierarchies of Security domains.
  - Applet and card life cycle:
    - Whenever the card is in the TERMINATED state, it should not be possible to revert to another state.
    - It should not be possible for an application that doesn't have the Card Terminate privilege to switch the card lifecycle state to TERMINATED

- The consistency of the Security domains hierarchy with respect to the privileges: Properties related to the Authorized Management privilege of SDs: ensuring that for any possible execution of the card, it could never happen that two (or more) SD with authorized management lie on some branch on the hierarchy)
- Properties related to the secure channel capabilities of the SD: ensuring that whenever a SD is moved across the hierarchy, the relevant authentications and accesses to secure channels are cleaned accordingly.

WP3: Requirement

- The artifacts that will be produced through the link with testing. Indeed, a property can be represented as a requirement in the testing models. The requirements used for testing will be formalized in a model of requirements provided by the WP3.

WP4: Modeling

- UMLseCh will be used for modeling the Global Platform secure channel protocols in order to formally verify the properties such as the confidentiality of transmitted data.
- UMLseCh will be used for modeling the Applet and Card life cycle in order to verify the evolution (related to this point) between two versions (2.1.1 and 2.2) of the Global Platform specification

5.4 Feasibility criteria

In the context of the embedded software on smart card, the feasibility criteria for the tools and methodologies that are to be provided by the project are related to two main areas: usability and scalability. Moreover, the software verification for embedded software on a hardware device may be done **off-card** in the development phase of the software or **on-board** during the installation phase on the device or **at runtime** during the execution of the application.

- **Usability** in the industrial development and validation processes of the software: any methodology and tools to be used in the life cycle of the software developed must respect the constraint of this life cycle. If a language is proposed to be used by the developer for its software, the knowledge of this language is a feasibility criterion. For example, if the language is not included into the list Java Card, C or assembly, the solution is not “feasible”.
    - If a tool is proposed to generate tests suites to be executed on the software, the generation time must fit in the validation life cycle. For example, if it takes 3 days to generate the test suites for a Java Card application, this tool is not usable and then does not fit the feasibility criterion.
- **Scalability**: any tool to be developed as part of the software embedded on the card must respect the size (footprint) and performance constraints

- If a tool to run as part of the embedded software takes 5 seconds to check an access on a command, it does not fit the feasibility criterion.

- **Relevance**: this criterion consists in identifying properties that are useful from an industrial point of view. For example, the correctness of the "whole system" to be tested, although necessary, is considered less relevant than the consistency of the security domain hierarchy.

Therefore, for each methods developed in the project, the evaluation criteria will be identified with respect to those feasibility criteria.

5.5 Feasibility studies

This section describes the feasibility exercises we have conducted for each WP working on the POPS changes requirements and their security properties.

5.5.1 Software update

5.5.1.1 Feasibility for WP6

WP6 deals with verification of security properties. The software update change requirement is about adding an application to the software embedded on the card. The challenge is then to demonstrate that this new application verifies a set of properties: some of them are related to the behavior of the application, the others are related to the card itself. The security means provides both off-card and on-card verification.

The feasibility studies conducted with the partners concentrated on the language, the properties and the tools. This enables to identify the scope of the evaluation and some directions to facilitate the evaluation.

We discussed the **Verifyfast** tool to be used for verification during the development of the application by checking directly its Java Card source code. Instead of formalizing the behavior of the application in a formal language, the use of the source code fully meets the **Usability** criteria. Indeed, source-code verification can be integrated into the application development process, and then facilitates the applicability in an industrial context.

For the on-device verification, two methods has been discussed with the partners:

- The first approach consists in checking that an applet, after being loaded on the card (byte-code format) and before its installation (linking) respects a given information flow control policy.

- The second approach is a security-by-contract approach: The methodology is based on "contract". Each Java Card applet comes with a contract that describes which services it needs from the other applets and which services it proposes to the other applets. The methodology is based on two "tools": a claim checker and a policy checker.

The security-by-contract approach is particularly challenging as several attempts have been done without success with respect to the scalability criteria.

Relating to the **properties**, we discussed with the partners which properties of Section 5.2 that the verification language and the tool are able to tackle.

- For example, with respect to the DoS property, the bounded memory consumption is a property that has been considered out of the scope. The two properties to be considered are the absence of runtime exceptions and infinite loops.

For the development-time verification we discussed the formalization in order to check the capability of a security engineer to express them in its code. The result of the feasibility study leads to a set of properties to be expressed as annotations to be inserted in the code.

For the on-device verification, the approach based on the verification of an embedded policy targets the following properties:

- No illegal access to a service: to avoid collusion between applications when using the services provided by other applications.
- Non-interference to avoid illegal information flows between applications
- Global control of interactions: no illegal sequence (of method calls)

The security-by-contract approach will ensure a given security policy by two tools:

- The contract claim checker checks that each applet (byte-codes) respects its contract (e.g., if the A1 contract claims that A1 calls m2 from A2, the checker parses the byte-codes to verify that this call exits and that there is no other call from A1). This could be done off-card, before the loading for example.
- The contract policy checker verifies at the loading or installation of the applet that the contract respects the policy of the card. Otherwise, the new applet is rejected.

Working with the partners, we also discussed the criteria of size, performance, and usability. For development-time verification, we discussed the time spent during the verification and which feedbacks are given to the user by the tool w.r.t the proof of a property. For on-device verification, the big challenge is the size and performance of the "tool" to be embedded as part of the software.

5.5.2 Specification evolution

5.5.2.1 Feasibility for WP7

This WP deals with software testing and more precisely model-based testing. The feasibility studies concentrated on

- For the specification evolution change requirement, we discussed the scope of the specification to deal with. It is clear that the *scalability* criterion is used in such a way that the complete specification could not be considered but the subset must be large enough to allow a relevant evaluation.
- The *relevance* criteria have been applied to discuss the properties to be tested. Since we are not dealing on functional testing, the studies focused on specific properties that come from the validation expertise of the use case provider. For

example, with respect to the security domains hierarchy, we discussed the importance of the secure channel capabilities of a security domain used for secure communication and we focus on testing the property that ensures that whenever a SD is moved across the hierarchy, the relevant authentications and accesses to secure channels are cleaned accordingly.

- The scalability criteria have been used for the properties to be tested. One of the properties is about the change in the finite state machine that describes the authorized transitions between lifecycle states of the card. The generic property is that for any possible execution of the card, the sequence of successive visited states should be accepted by the finite state machine described in the specification. We discussed the restriction of the property for some relevant values of the card state and some relevant transitions. An example of property is that when the card is into the “mute” state, no action could revert it to an operational state.
- With respect to the security domain hierarchy, a large number of properties could be tested. Following several discussions with the partners, we decide to focus on the properties related to the Authorized Management privilege of SDs: ensuring that for any possible execution of the card, it could never happen that two (or more) SD with authorized management lie on some branch on the hierarchy.
- The scalability criteria for the tests generation tool we discussed is the capability of the tool to provide an incremental generation of tests, and their automatic classification and adaptation (obsolete tests, no impacted tests, adapted tests, additional tests).
- The usability criteria concern the traceability from the test objectives to the generated tests. This traceability is one of the main features needed by the Common Criteria evaluation.

5.5.2.2 Feasibility for WP3

The feasibility of WP3 is closely related to that of WP7 because WP3 is indirectly evaluated using the WP7 results. We expect that WP3 provides the same level of Usability and Scalability as those WP7 descried above.

5.5.2.3 Feasibility for WP4

The feasibility discussion to investigate the application of the model-based formal verification started with the need for verification of communication protocol but taking into account the source code. The choice was between verifying the protocol using a model of its behavior but without any link to the code implementing it or verifying directly the implementation. After some discussion using the GP Secure Channel protocols as the target, first it appears that the second approach is not feasible due to the non availability of the implementation. Second the UMLseCh methodology developed is more suitable for the first approach and could provide interesting result with respect to the specification evolution requirement.

Hence, this work package uses the UMLseCh language for modeling the behavior of the secure channel protocols. In contrast to the applet verification in WP6, the

specification modeling and verification is expected to be done by a Formal Method expert (rather than by a developer).

In terms of scalability, we expect that UMLseCh provides all necessary elements to formalize a security protocol and its properties.

Usability is less demanding because the final user has already a formal method background. However, a UML tool integrating UMLseCh is expected for usage purpose.

5.5.3 Integration

The feasibility of collaboration between two WPs is the conjunction of those of each WP (with the focus on the collaborating scope).

5.5.3.1 Feasibility for WP3-WP7 collaboration

These work packages will collaborate around properties, modeled as requirements and then used for testing: for example, let P a security property (e.g. users must be authenticated), P will be refined in a given number of requirements that will be explicit in the testing model and then a test suite will be generated for each requirements that will represent the test of the given properties P. The WP3 will exhibit a methodology explaining why the requirements are sufficient to test P.

With respect to a change in the specification (or a new attack not yet handled by the list of requirements), one must be able to check if the set of requirements is still sufficient, and to add the necessary requirements and generate only the necessary tests.

5.5.3.2 Feasibility for WP4-WP7 collaboration

These work packages about modeling and testing will collaborate through the application of the model verification techniques of security properties to the testing models. The integration will contribute to the specification evolution for the Card and Applet life cycle.

The feasibility discussion outlines the following idea: the UMLseCh model is used to verify properties. Whenever a change is done, a new UMLseCh model M' is generated. The WP4 methodology will generate an XML file describing the delta between the two models. This delta will be used to generate the testing model M' (for details, please refer to D4.2).

5.5.3.3 Feasibility for WP6-WP7 collaboration

The idea here is that when the WP6 proves a runtime property P on a Java Card applet, it relies on the Global Platform secure loading and installation hypothesis. This hypothesis is actually checked in WP7 by testing. On the other hand, for WP7, some properties cannot be validated with a model-based testing approach but they can be verified by the WP6 approach.

For example, WP6 aims at ensuring that the communications between the applets through the Java Card shareable interfaces respect the security policy of these applets.

Consider the case where two applets are associated (directly or indirectly) to the same security domain, with which they may communicate through the use of the GP API. These communications are not verified by WP6. Instead, WP6 uses the underlying assumption that these communications are correct. This assumption is validated by testing in WP7.

On the other hand, WP7 assumes that the confidentiality of the security domain keys is protected through the Java Card shareable interfaces. This property is indeed verified by WP6.

5.5.3.4 Feasibility for WP4-WP6 collaboration

The idea of this cooperation is defined on “transitive control flow” and “non-interference” security properties. These properties are specified and verified on the model UMLseCh (by WP4). They are also specified and verified on the implementation by WP6. The two verification processes will produce two outputs that will be compared. The analysis of the detected differences provides hints on the consistency between the model, the implementation and the properties (more details are found in the deliverable D4.2).

5.6 Evaluation Criteria

This section describes the different criteria that will be used to evaluate the specific security means when it is applied to the POPS case study. Those criteria come from an agreement and technical discussion with the partners.

During the third year of the project, the high level Evaluation criteria defined below may be broken down into more detailed and measurable Validation criteria for each foreseen Validation exercise. This process of decomposition has to be repeated several times resulting in a hierarchical structure of more and more detailed criteria instantiated in the various Validation activities. The decomposition of objectives ends with the identification of basic indicators and evidences to be measured and/or collected during Validation exercises. Note that indicators and evidences provided can be quite diverse. For instance, some indicators can be measurable in a quantitative way. Whereas, other evidences might highlight compliance with standards or development processes. Finally, other evidences can be just qualitative evaluations, obtained with the support of domain experts and practitioners. The results of the evaluation of the artifact with respect to those criteria will be described in the deliverable D1.3. For each criterion, some improvement direction will also be provided as the evaluation is with respect to a industrial context that could be too restrictive.

We will not provide additional criteria for the inter-WP collaboration and integration. Such criteria will be directly derived from the ones defined for each single WP by juxtaposition or intersection, depending on the foreseen linking activities.

5.6.1 For means provided by WP3

The main criteria will be to identify to which extent the requirements modeling and traceability to the security objective could be applied to the case study. Potentially,

some criteria listed in Section 5.6.4.1 about the industrial usability may be used for the evaluation.

5.6.2 For means provided by WP4

- Usability of the UMLseCh methodology (stereotype) by a validation engineer
- Scalability
- Ability to express security protocol elements and properties

5.6.3 For means provided by WP6

5.6.3.1 Development-time Verification

A tool to be used for Java Card verification, during the design and development phase, will be evaluated against the following criteria:

1. Scalability & performance
    a. Code size: it is about the size of the application that could be tackled by the tool, and generally the Java Card applications obey to a set of constraints.
    b. Modularity of the verification
    c. Speed of the verification process: e.g. if a proof obligation takes 3 days, the usability of the tool could be impacted with respect to the development life cycle of an application.
    d. Change tolerance: impact of a code change on the annotations or the model describing the application
2. Modeling: expressiveness of the language to model the properties
    a. Pollution of the code if annotations are used: the amount of annotations must not exceed a certain percentage of the code
3. Proof capability
    a. Degree of Automation
    b. User interaction: e.g. does the tools provides hints to complete the proof/to debug the annotation, etc?
4. User-friendly interface
    a. Requirement on the level of user expertise
5. Integration in the industrial development process: e.g. what are the missing features to integrate the tool in the application development process ?

5.6.3.2 On-device verification

Two feasibility approaches will be used, that depend on the evaluation strategy. First approach is to evaluate the tool by embedding it on a specific platform. The second approach is an abstract one (theoretical) where the evaluation will be performed by extrapolation. The evaluation will then consist in checking and discussing a feasibility report.

In case of concrete implementation, the tool (piece of software that will run on the card) will be evaluated against the following feasibility criteria:

1. For the tool to be embedded:

a. Footprint on the platform: this criteria is critical for a tool supposed to be part of an embedded software and several tools have been developed but never been embedded concretely.
b. RAM consumption: critical for performance issue, as memory is a constrained resource on a device like a smart card.
c. Time overhead

2. For the applet: how the policy is integrated to the code
    a. Methods and tools to define the security policy
        i. How the policy is expressed?
        ii. How the policy is attached to the application (to its byte-codes)?
        iii. Overhead on the code size to store when the applet is on board

If an implementation on a specific platform is not possible (due to time and resource constraint), then these criteria cannot be measured precisely. In this case, the criteria will be estimated by extrapolating the results obtained on a PC implementation of the methods and tools. The academic partner will provide the necessary software to re-produce these results in GTO and also the necessary justification to extrapolate these results to get the criteria described above.

If even a PC implementation is not available, then only the theoretical complexity of the approach can be evaluated. In this case, the POPS case study provider needs the description and the analysis of complexity (in terms of time but also persistent and RAM space) of all algorithms used for managing the application's security policies and checking them.

5.6.4 For means provided by WP7

The criteria below deals with changes introduced by Global Platform specification evolution (cf §5.1.1, §5.2). The test model verifies the impact of the specification evolution on two subparts: *card life cycle* and *card contents management.*

For model-based testing, the following criteria will be used for the evaluation of the approach.

5.6.4.1 Models

- Is it a correct abstraction of the specification?
- Has it a sufficient level of details to express the properties to be tested
- What is the language to express the properties, is it independent from the model?
- Level of expertise to develop a model

5.6.4.2 Tool

- Test quality: what kind of coverage, code or requirements?
- Negative testing

- Regression testing
- Bounds testing

5.6.4.3 Efficiency

- Generation time of the test cases for the model
- Number of test cases generated
- Test quality vs. changes: do we have to regenerate all the tests ? do we have to execute all the tests ?
- Execution time
- Polynomial vs. percentage of changes in the specification
- Traceability and documentation
- Use friendly aspect of the tool
- Level of expertise to manipulate the tool: is it for a validation expert or for model-based testing expert?
- Integration with industrial test environment: this could include several evaluation on the criteria above

6 Glossary

6.1 ATM Case Study

Acronyms	Definition
ACARS	Aircraft Communications Addressing and Reporting System
ACC	Area Control Center
ADD	ADD Aircraft Derived Data
ADS-B	Automatic Dependent Surveillance Broadcast
ADS-C	Automatic Dependent Surveillance Contract
AMAN	Arrival MANager
ANS	Air Navigation Services
ANSP	Air Navigation Services Provider
ATC	Air Traffic Control
ATCO	Air Traffic COntroller
ATM	Air Traffic Management
BT	Business Trajectory
CNS	Communication Navigation Surveillance
CORA	COnflict Resolution Assistant
CWP	Controller Working Position
DMAN	Departure MANager
ETA	Estimated Time of Arrival
EUROCONTROL	The European Organization for the Safety of Air Navigation
FAA	Federal Aviation Administration
FDP	Flight Data Processing
FMS	Flight Management System
HMI	Human Machine Interface
ICAO	International Civil Aviation Organization
MSAW	Minimum Safe Altitude Warnings
MSP	Multi Sector Planner
MTCD	Medium -Term Conflict Detection
RTA	Required Time of Arrival
SESAR	Single European Sky ATM Research

SMAN	Surface MANager
SSR	Secondary Surveillance Radar
STCA	Short-Term Conflict Alert
SWIM	System Wide Information Management
TMA	TerMinal Area

6.2 HOMES Case Study

Acronyms	**Definition**
DHCP	Dynamic Host Client Protocol
FTTP	Fiber To The Premises
NAC	Network Access Control
NAT	Network Address Translation
OSGi	Open Service Gateway Initiative.
PDP	Policy Decision Point
PEP	Policy Enforcement Point
PLC	Power Line Communication
PPPOE	Point-to-Point Protocol over Ethernet
QOS	Quality of Service
VPN	Virtual Private Network
WIMAX	Worldwide Interoperability for Microwave Access

6.3 POPS Case Study

Acronyms	**Definition**
AID	Application identifier
APDU	Application Protocol Data Unit
SCP	Secure Channel Protocol
EMV	Europay MasterCard Visa
ISD	Issuer Security Domain
SIM	Subscriber Identity Module
USIM	Universal Subscriber Identity Module

References

[1] D1.1. Description of the Scenarios and their requirements, Version 1.4, SecureChange, 2010.

[2] ESARR 4. EUROCONTROL Safety Regulatory Requirement – ESARR 4 Risk Assessments and Mitigation in ATM, Version 1.0, EUROCONTROL, 2001.

[3] Jackson, M.. Problem Frames: Analyzing and structuring software development problems, Addison-Wesley, ACM Press, 2001.

[4] Gordon, R., Kirwan, B., Mearns, K., Kennedy, R., and Jensen, C.L. A Safety Culture Questionnaire for European Air Traffic Management, 2007.

[5] EUROCONTROL. Understanding Safety Culture in Air Traffic Management, Safety, Security and Human Factors Business Division (DAP/SSH), 2006.

[6] FAST. The FAST Approach to Discovering Aviation Futures and Associated Hazards, Methodology Handbook, 2008.

[7] [MS-SOH]: Statement of Health for Network Access Protection (NAP) Protocol Specification. http://download.microsoft.com/download/9/5/E/95EF66AF-9026-4BB0-A41D-A4F81802D92C/%5BMS-SOH%5D.pdf

[8] Runtime Environment Specification Java Card™ Platform, version 2.2.2 Sun Microsystems, Inc., 2006.

[9] Virtual Machine Specification Java Card™ Platform, version 2.2.2 Sun Microsystems, Inc., 2006.

[10] Application Programming Interface Java Card™ Platform, version 2.2.2 Sun Microsystems, Inc., 2006.

[11] Global Platform Specification 2.2, 2006

[12] Global Platform UICC Configuration Version 1.0, 2008.

[13] Common Criteria for Imformation Technology Security Evaluation, Version 3.1, Revision 3, July 2009; (CCMB-2009-07-001, 002 and 003). See also ISO 15408 (re-published as an ISO Standard)