SEVENTH FRAMEWORK PROGRAMME

RASEN

Compositional Risk Assessment and Security Testing of Networked Systems

Deliverable D5.4.3

A Toolbox for Security Risk Assessment and Security Testing

Project title:	RASEN
Project number:	316853
Call identifier:	FP7-ICT-2011-8
Objective:	ICT-8-1.4 Trustworthy ICT
Funding scheme:	STREP – Small or medium scale focused research project

Work package:	WP5
Deliverable number:	D5.4.3
Nature of deliverable:	Report
Dissemination level:	PU
Internal version number:	1.0
Contractual delivery date:	2015-09-30
Actual delivery date:	2015-09-30
Responsible partner:	FOKUS

Contributors

Editor	Jürgen Großmann (FOKUS)
Contributors	Fredrik Seehusen (SINTEF), Jürgen Großmann (FOKUS), Michael Berger (FOKUS), Frank Werner (SAG).
Quality assurors	Arthur Molnar (Info World), Frank Werner (Software AG)

Version history

Version	Date	Description
0.1	15-07-10	TOC
0.2	15-07-28	Integration scenarios and initial updates
0.3	15-09-06	Updates w.r.t integration scenarios
1.0	15-09-22	Final version after review

Abstract

The RASEN risk assessment and security testing toolbox provides integration support for the RASEN approach to risk-based security testing and test-based security risk assessment. This deliverable contains updates of the RASEN Data Integration Model and RASEN Data Exchange Format. It presents the Integration Scenarios that have been achieved in using the RASEN Data Exchange Format.

Keywords

RASEN, Security Risk Assessment, Security Testing, Data Integration Models

Executive Summary

The RASEN risk assessment and security testing toolbox provides integration support for the RASEN approach to risk-based security testing and test-based security risk assessment. The overall integration approach provides a lightweight data exchange format on basis of XML. The format specifies the artifacts and elements that enable integration of tools and allow for data exchange. These artifacts and elements are modeled by means of UML in the RASEN Data Integration Model. The model is taken to serve as the basis for the interface definitions at tools and as the source for XML schema generation.

This deliverable contains final updates of the RASEN Data Integration Model and final updates of the RASEN Data Exchange Format. Finally, the deliverable describes the use of the RASEN Data Exchange Format to realize the main integration scenarios originally specified in RASEN Deliverable D5.2.1.

Table of Contents

1 Introduction

The RASEN techniques and methodologies aim explicitly for the integration of data from security risk assessment and security testing. The RASEN Deliverables D5.2.1 [4], D5.4.1 [5] and D5.4.2 [6] have provided the conceptual and technical basis for a Data Exchange Format that allows for a systematic data exchange between the RASEN tools. In general, the overall tool integration approach in the RASEN approach has been defined as follows:

1. Create deployable generic security testing and security risk assessment models. These models will be designed in UML.
2. Create a deployable generic risk assessment and testing model (RASEN Data Integration Model) on basis of the models defined beforehand.
3. Define conceptual tools and their interfaces
4. Instantiate a common XML-based exchange format on basis of the RASEN Data Integration Model.
5. **Develop export/import adapter for the relevant instantiations of the conceptual tools (concrete tools) with respect to the XML format from 4 and the interfaces from 3.**
6. **Identify additional integration use cases and update the models if necessary.**

While this document describes updates for steps 1 to 4, it specifically adds the results from step 5 and 6 (typeset in bold face). Both steps complete the definition of the RASEN Exchange Format and its implementation by the RASEN tools.

In summary, this deliverable provides updates to the specifications that have been already provided by D5.2.1, D5.4.1 and D5.4.2 so that the conceptual model and the interfaces become aligned with results and changes that have been introduced by the other technical work packages. It documents the status of the implementation of the export/import adapters by the RASEN tools. Finally, it introduces the notion of integration scenarios to show how the RASEN Export Format and the respective tool adapters are used for the systematic data interchange between the RASEN risk assessment tools and the RASEN testing tools.

Section 2 presents updates with respect to the classification of the conceptual RASEN tools. Section 3 provides an update of the data integration model, which is used as the conceptual basis for the RASEN Data Exchange Format, and Section 4 provides an overview on the adapters that have been realized for the RASEN tools. Finally, Section 5 specifies the integration scenarios that are used to realize the RASEN toolbox. Section 6 concludes this report. The final XML schema specification of the RASEN Data Exchange Format is given in Appendices A-G.

2 Update of the Conceptual RASEN Tools

The following table introduces five different conceptual tools as targeted by the RASEN project. This table is an update of a similar table in Deliverable D5.2.1. Table 1 shows the RASEN conceptual tools and their implementation by the actual RASEN tools.

RASEN conceptual tool	RASEN tool
Security Risk Assessment Tool (SRAT) for supporting compositional risk assessment	**CORAS** risk assessment tool (developed by SINTEF) **RACOMAT** risk assessment and testing tool (developed by Fraunhofer) **ARIS Business Architect** providing models for risk assessment (developed by Software AG)
Security Test Pattern Database (PDB) for managing test patterns	**RACOMAT** risk assessment and testing tool (developed by Fraunhofer)
Security Test Derivation Tool (STDT) for supporting the derivation and prioritization of test cases based on the risk assessment	**CertifyIt for Security Testing** (developed by Smartesting) **RACOMAT** risk assessment and testing tool (developed by Fraunhofer)
Security Testing Tool (STET) for adapting the test item and executing test cases	**CertifyIt for Security Testing** (developed by Smartesting) **RACOMAT** risk assessment and testing tool (developed by Fraunhofer)
Security Test Result Aggregation Tool (STRAT) for supporting the aggregation of security test results into a format that allows us to verify the risk picture and to update the risk picture based on the results	**Fraunhofer Testing Dashboard** (developed by Fraunhofer) **RACOMAT** risk assessment and testing tool (developed by Fraunhofer) **CORAS risk monitor prototype** (developed by SINTEF), which defines rules for updating the risk assessment at run-time **ARIS Business Architect** providing the risk aggregation and risk rating functionality (developed by Software AG)

Table 1 – Conceptual RASEN tools

Figure 1 shows the formal relationship between the conceptual RASEN tools and the actual RASEN tools. The conceptual RASEN tools are represented by UML interfaces and the actual RASEN tools are specified by UML classes. The integration interfaces for data exchange are defined on the level of the RASEN conceptual tools, thus by means of UML interfaces that represent these tools. On basis of that, we can provide a flexible mapping between the conceptual RASEN tools and the actual RASEN tools by considering that each concrete RASEN tool that takes the role of a conceptual RASEN tool has to formally implement the interface that represents the conceptual RASEN tool.

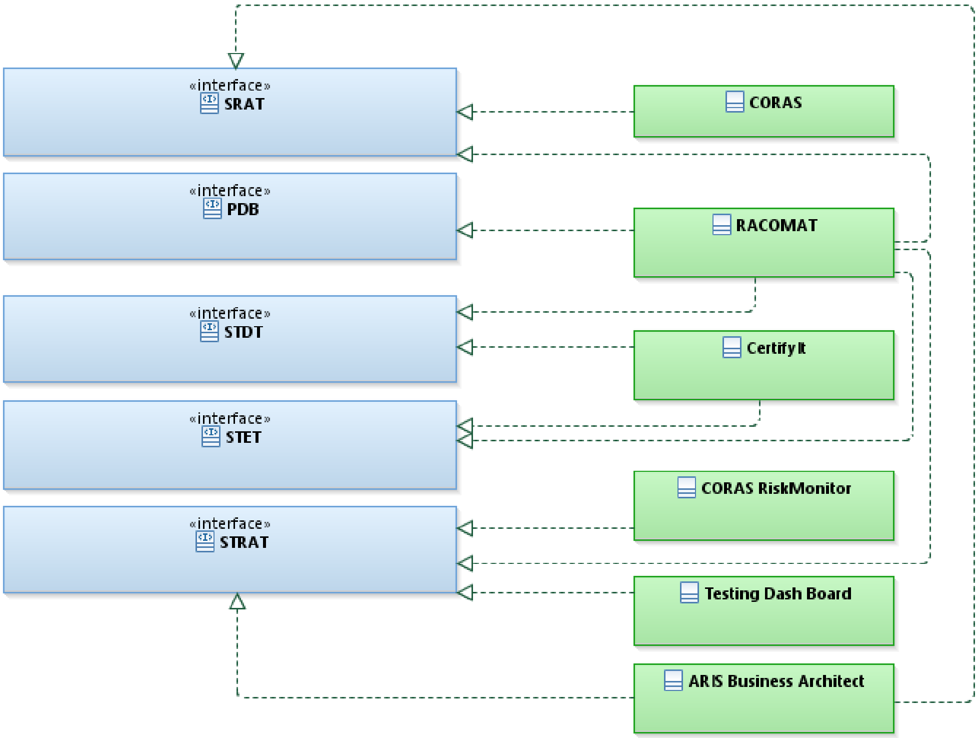

Figure 1 – UML diagram modeling the relationship between conceptual and concrete tools

3 Update of the RASEN Data Integration Models

This section describes the basic concept of the RASEN Aggregated Test Report Format. The format is defined by a Metrics and Measurement Schema as well as by the Aggregated Test Report Schema. Both XML schemata can be found in Appendix E and F of this document.

3.1 Metrics and Measurements

Metrics and measurements are generally described in [7] and a detailed description of the metric processing in RASEN can be found in [8].

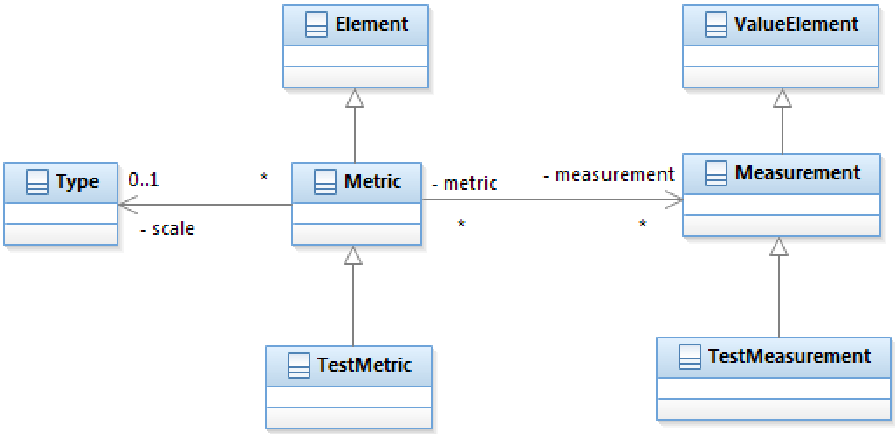

Figure 2 – Security Test Metric and Measurement Model diagram

A security test measurement (cf. Figure 2) describes a quantitative value of test results for a System under Test (SUT). It consists of a type and a value and can be named and described, like defined for element in the foundation package. A test metric assesses such measurements: it describes a qualitative value for one or more related measurements. Test metrics can be used in different applications, for example the RASEN security dashboard [8] and for the aggregated test report. The security test metric and measurement model schema is given in Appendix E and is necessary for the aggregated test report.

3.2 Aggregated Test Report

The aggregated test report (cf. Figure 3) aggregates test results to a reporting with qualitative analysis data. Each aggregated test report is related to one test item. Such an item represents one element of the analysed system. That could be a hardware component, a software part or a system related object like an analysed security risk attack path. This aggregated report can be used for security test reporting, further analysis, and as input for new risk assessment analysis. The model schema for the RASEN aggregated test report model format is given in Appendix F.

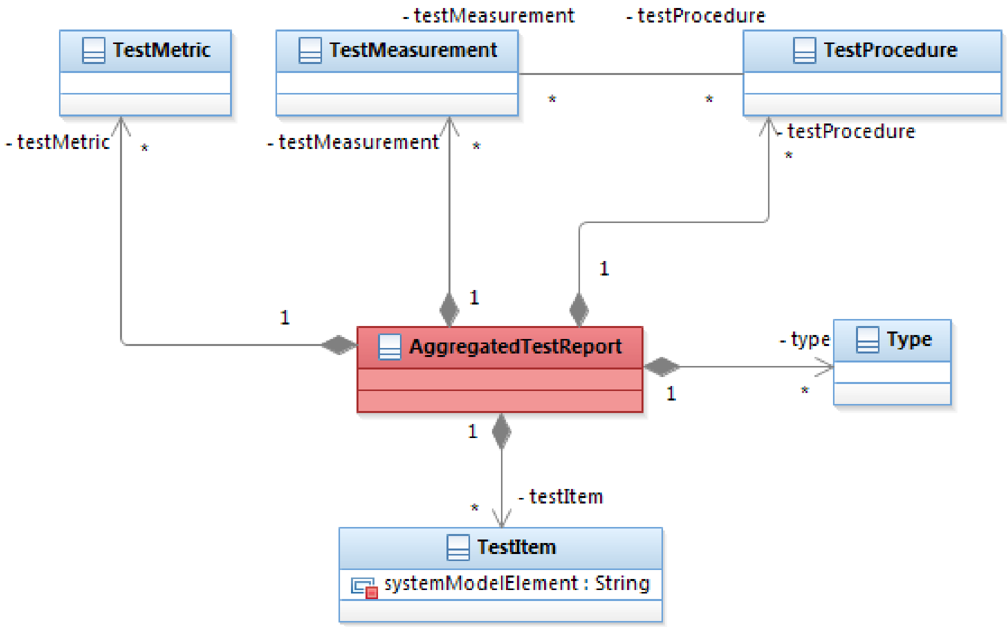

Figure 3 - Aggregated Test Report Model diagram

4 Realization of Export/Import Adapters at RASEN Tools

RASEN tools realize export/import adapters according to their role in the overall risk assessment and testing process. Table 2 documents the implementation of export/import adapters for the individual RASEN tools.

RASEN tool	Export adapter	Import adapter
CORAS risk assessment tool (developed by SINTEF)	RASEN Risk Model Format	RASEN Aggregated Test Report Format
RACOMAT risk assessment and testing tool (developed by Fraunhofer)	RASEN Risk Model Format RASEN Test Report Format	RASEN Risk Model Format ARIS Vulnerability Components
ARIS Business Architect providing models for risk assessment (developed by Software AG)	ARIS Vulnerability Components	RASEN Test Report Format for data integration with RACOMAT
CertifyIt for Security Testing (developed by Smartesting)	RASEN Test Report Format	RASEN Risk Model Format
Fraunhofer Testing Dashboard (developed by Fraunhofer)	RASEN Aggregated Test Report Format	RASEN Risk Model Format RASEN Test Report Format

Table 2 – RASEN tools and realization of export/import adapters for the RASEN Data Exchange Format

5 The RASEN Data Exchange Format and its Application

This section documents the use of the RASEN Data Exchange Format and its respective tools adapters to realize the Integration Use Cases of the RASEN project. The Integration Use Cases have been originally defined in RASEN Deliverable D5.2.1 with the aim to document the main requirements for tool integration. To show how these uses case have been realized, we introduce the notion of Integration Scenarios that are specified by means of the following template:

Name	***<The name of the integration scenarios>***
Tools	*<The RASEN tools that are involved in the integration scenario>*
Coverage	*<References to the integration uses case (e.g. IUC_01) that are covered by the integration scenario. The integration use cases are defined in Deliverable D5.2.1>*
Actors	*<The actors that are referred to in the scenario (supposed to be the user roles that act in the scenarios). We use the following two actors* • *Risk analyst (RA): The person responsible for doing the security risk assessment.* • *Security tester (ST): The person responsible for doing the security testing.* *>*
Precondition	*<The precondition that need to be enabled to properly start the integration scenario>*
Postcondition	*<The postcondition that is valid when the scenario has been accomplished successfully>*
Scenario	*<The actual scenario that describes the individual actions taken by the actors and tools>*
Data exchanged	*<The RASEN Data Exchange Format that is used during the integration scenario>*

Table 3 – Integration scenario template

An Integration Scenario is able to serve multiple Integration Use Cases. The coverage of Integration Use Cases by Integration Scenarios is denoted in the 'Coverage' field of the scenario table. With respect to the RASEN method we distinguish test derivation integration scenarios (see Table 4 - **Table 6** in Section 5.1) and test result aggregation integration scenarios (see Table 7 - Table 9 in Section 5.2). Since the ARIS GRC has not been used during the project the RASEN method for legal and compliance assessment has been completely realized as an extension of the CORAS tool. Thus, tool integration was not necessary to realize the Integration Use Cases IUC 7 – IUC9 from Deliverable D5.2.1.

5.1 Test Derivation Integration Scenarios

The following scenarios describe the integration of tools during the risk-based derivation of security tests.

Name	**Risk-based Security Testing**
Tools	**CORAS (SRAT)** and **Smartesting CertifyIt (STDT)**

Coverage	**IUC_01, IUC_03**
Actors	Risk Analyst (RA), Security Tester (ST)
Precondition	A risk assessment model with likelihood estimates exists in **CORAS**
Postcondition	A prioritized list of test-scenarios exists in **Smartesting CertifyIt**
Scenario	1. **RA** loads the *risk assessment model* in the **CORAS** tool 2. **RA** annotates *the risk assessment model* in the **CORAS** tool with annotations (e.g. testability and uncertainty) that are used to guide the test-scenario prioritization. 3. **RA** starts the test scenario prioritization in the **CORAS** tool 4. **CORAS** automatically identifies potential test scenarios in the annotated risk assessment model and generates *a prioritized list of test procedures*. 5. **RA** selects the test scenarios that should be tested and exports the *risk assessment model* to the **RASEN Risk Model Format** 6. **ST** imports the *risk assessment model* in **RASEN Risk Model Format** into **Smartesting CertifyIt**. 7. **ST** assigns test purposes to the vulnerability in the *risk assessment model* 8. **Smartesting CertifyIt** generates a certain amount of security test cases with respect to the risk values in the *risk assessment model.*
Data exchanged	Data in **RASEN Risk Model Format**

Table 4 – Integration scenario: Preparation for security testing based on risk assessment results

Name	**Test identification on basis of Attack Pattern + Test Pattern combination**
Tools	**CORAS (SRAT)** and **RACOMAT (STDT)**
Coverage	**IUC_02**
Actors	Risk Analyst (RA), Security Tester (ST),
Precondition	Risk assessment is done
Postcondition	Security test cases that are specific to the risks addressed in the risk assessment have been generated.
Scenario	1. **RA** exports the *high level risk assessment model* **from CORAS** to the **RASEN Risk Model Format** 2. **ST** loads the *risk assessment model* in **RASEN Risk Model Format** into the **RACOMAT** tool. 3. **ST refines the** *risk assessment model with respect to testing needs (assigns threat interfaces, refines threat scenarios)* 4. **ST** identifies test pattern in the **PDB** with respect to the refined *risk assessment model*. 5. **RACOMAT** calculates risk-based priority values for the threat interfaces and

	associated threat scenarios. 6. **RACOMAT** generates a certain amount of security test cases with respect to the vulnerabilities, threat scenarios, test pattern and the related risk values
Data exchanged	Data in **RASEN Risk Model Format**

Table 5 – Integration scenario: Test identification on basis of Attack Pattern + Test Pattern combination

Name	**Risk assessment and automated Security Testing combination**
Tools	**ARIS Business Architect (SRAT)** and **RACOMAT (STDT)**
Coverage	**IUC_01, IUC_02, IUC_06**
Actors	Risk Analyst/Software Architect (RA), Security Tester (ST)
Precondition	A risk assessment model has been devised in **ARIS Business Architect** containing components along with their CWEs and the component hierarchy.
Postcondition	A list of present CWEs and possibly aggregated component risks is provided by RACOMAT and imported to ARIS Business Architect
Scenario	1. **RA** uses the *risk assessment model* in the **ARIS** to model a software 2. **RA** model contains product-specific/generic components where a vignette and possible CWEs are included. 3. **RA** triggers in **ARIS** the export interface and a software product hierarchy is generated and exported to **RACOMAT** 4. **ST** derives (automatically) a risk tree, vulnerabilities (CWSS) and in the subsequent step generates test cases in **RACOMAT** which are tested against the system under test (residing as a black-box in a VM). 5. **ST** selects the test scenarios that should be tested and exports the found vulnerabilities and aggregated risks in the **RASEN Risk Model Format** 6. **RA** imports the *risk assessment model* in **RASEN Risk Model Format** into **ARIS Business Architect**. 7. **RA** has a complete picture about the present risk in components and the top product.
Data exchanged	Data import and Export in ARIS/RACOMAT JSON format

Table 6 – Integration scenario: Test identification on basis of Attack Pattern + Test Pattern combination

5.2 Test Result Aggregation Integration Scenarios

The following scenarios describe the integration of tools during the process of test result aggregation to serve test-based risk assessment.

Name	**Calculating Vulnerability Coverage**
Tools	**Smartesting Certifyit** or **RACOMAT (STET), CORAS (SRAT)** and **Fraunhofer Test Metric Dashboard**
Coverage	**IUC_04**
Actors	Security Tester (ST), Risk Analyst (RA)
Precondition	The results of a test run exist in form of a *TestLog* in **Smartesting CertifyIt** or **RACOMAT**
Postcondition	Vulnerabilities in the **risk model** provide information about the number of failed and passed related security test cases.
Scenario	1. **RA** exports *risk assessment model* **from CORAS** to the **RASEN Risk Model Format** 2. **ST** exports the *TestLog* from **Smartesting CertifyIt** or **RACOMAT** to the **RASEN Test Report Format** 3. **ST** imports the *risk assessment model and the TestLog,* both in **RASEN FORMAT,** into the **Fraunhofer Test Metric Dashboard.** 4. **Fraunhofer Test Metric Dashboard** displays the vulnerabilities with an overview on related security test cases and their results
Data exchanged	Data in **RASEN Risk Model Format** and **RASEN Test Report Format**

Table 7 – Integration scenario: Calculating Vulnerability Coverage

Name	**Updating vulnerability information based on test results**
Tools	**Smartesting CertifyIt** or **RACOMAT (STET), CORAS (SRAT)** and **Fraunhofer Test Metric Dashboard**
Coverage	**IUC_05**
Actors	Risk Analyst (RA), Security Tester (ST),
Precondition	The results of a test run exist in form of a *TestLog* in **Smartesting CertifyIt** or **RACOMAT**
Postcondition	Measurements related to *Vulnerabilities* (likelihood of existence, exploitability) are updated in the *risk assessment model* based on the test results.
Scenario	1. ST exports the *TestLog* from **Smartesting CertifyIt** or **RACOMAT** to the **RASEN Test Report Format.** 2. ST imports the *TestLog* in **RASEN Test Report Format** into the **Fraunhofer Test Metric Dashboard.** 3. ST applies a selected set of testing metrics to aggregate the results in the *TestLog.* **4.** ST exports an *Aggregated Test Report* **in RASEN Aggregated Test Report Format** from the **Fraunhofer Test Metric Dashboard**.

	5. **RA** imports the *Aggregated Test Report* **in RASEN Aggregated Test Report Format** into **CORAS** and updates *the risk model model* with updated vulnerability measures. 6. **SRAT** updates the risk picture based on the new vulnerability measurements
Data exchanged	Data in **RASEN Test Report Format** and **RASEN aggregated Test Report Format**

Table 8 – Integration scenario: Updating vulnerability information based on test results

Name	**Getting information about new vulnerabilities based on test results**
Tools	**Smartesting CertifyIt** or **RACOMAT (STET), CORAS (SRAT)** and **Fraunhofer Test Metric Dashboard**
Coverage	**IUC_06**
Actors	Risk Analyst (RA), Security Tester (ST),
Precondition	The results of a test run exist in form of a *TestLog* in **Smartesting CertifyIt** or **RACOMAT**
Postcondition	The new *vulnerabilities* identified in **Smartesting CertifyIt** or **RACOMAT** exist in the *risk assessment model* of the **SRAT**
Scenario	1. **ST** exports the *TestLog* from **Smartesting CertifyIt** or **RACOMAT** to the **RASEN Test Report Format.** 2. **ST** imports the *TestLog* in **RASEN Test Report Format** into the **Fraunhofer Test Metric Dashboard.** 3. The **RA** manually places the new vulnerabilities at the right location in the tool specific risk assessment model.
Data exchanged	Data in **RASEN Test Report Format** and **RASEN aggregated Test Report Format**

Table 9 – Integration scenario: Getting information about new vulnerabilities based on test results

6 Summary and Outlook

This deliverable is the last part of a series of 3 deliverables that specify the RASEN Data Exchange Format on basis of the RASEN Data Integration Model. The Data Integration Model has been derived from the RASEN Conceptual Model that was introduced in RASEN Deliverable D5.3.1. The RASEN Data Exchange Format addresses the integration between risk assessment and testing and depicts the technical aspects that are necessary for feasible data exchange between tools. In combination with the tools that have been developed in other technical work packages, the RASEN Data Exchange Format constitutes the RASEN Toolbox for risk-based security testing and test-based risk assessment.

This deliverables provides final updates of the RASEN Data Exchange Format, namely the RASEN Aggregated Test Report Format and the ARIS Import/Export Format. It documents the implementation status of the tool adapters at the RASEN tools and documents the use of the RASEN Exchange Format in Integration Scenarios that serve the major RASEN Integration Use Cases. The complete schema of the RASEN Data Exchange Format has been documented in the appendices of this document.

References

[1] RASEN Deliverable D5.3.1, Methodologies for legal, compositional, and continuous risk assessment and security testing v.1, 2013

[2] RASEN Deliverable D5.3.2, Methodologies for legal, compositional, and continuous risk assessment and security testing v.2, 2014

[3] RASEN Deliverable D5.3.3, Methodologies for legal, compositional, and continuous risk assessment and security testing v.3, 2015

[4] RASEN Deliverable D5.2.1, Risk assessment and security testing toolbox requirements and design, 2013

[5] RASEN Deliverable D5.4.1, A toolbox for risk assessment and security testing v.1, 2013

[6] RASEN Deliverable D5.4.2, A toolbox for risk assessment and security testing v.2, 2014

[7] Wayne Jansen: Directions in Security Metrics Research, NISTIR 7564, Computer Security Division, Information Technology Laboratory, National Institute of Standards and Technology, Gaithersburg 2009

[8] RASEN deliverable 4.2.3, Techniques for Compositional Risk-Based Security Testing v.3, 2015

Appendix A: Risk Model Schema

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns:RiskModel="http://RiskModel"
xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore"
xmlns:foundation="http://foundation" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
ecore:nsPrefix="RiskModel"
ecore:package="de.fraunhofer.fokus.rasen.dataexchange.RiskModel"
targetNamespace="http://RiskModel">
  <xsd:import namespace="http://www.eclipse.org/emf/2002/Ecore"
schemaLocation="platform:/plugin/org.eclipse.emf.ecore/model/Ecore.xsd"/>
  <xsd:import namespace="http://foundation" schemaLocation="foundation.xsd"/>
  <xsd:simpleType ecore:name="COMMON_RISK_PARAMETER_NAMES"
name="COMMON_RISK_PARAMETER_NAMES">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="Likelihood"/>
      <xsd:enumeration value="Consequence"/>
      <xsd:enumeration value="RiskValue"/>
      <xsd:enumeration value="RiskNodeType"/>
      <xsd:enumeration value="RiskRelationType"/>
      <xsd:enumeration value="Vulnerability"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType ecore:name="COMMON_RISKNODE_TYPE_VALUES"
name="COMMON_RISKNODE_TYPE_VALUES">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="Threat"/>
      <xsd:enumeration value="UnwantedIncident"/>
      <xsd:enumeration value="ThreatScenario"/>
      <xsd:enumeration value="Asset"/>
      <xsd:enumeration value="Event"/>
      <xsd:enumeration value="EventSource"/>
      <xsd:enumeration value="Risk"/>
      <xsd:enumeration value="Vulnerability"/>
      <xsd:enumeration value="Treatment"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:complexType ecore:implements="foundation:ParameterValue" name="RiskElement">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:attribute ecore:name="testPatterns"
ecore:reference="RiskModel:TestPattern" name="testPattern">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
        <xsd:attribute ecore:name="testProcedures"
ecore:reference="RiskModel:TestProcedure" name="testProcedure">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="RiskFunction">
    <xsd:attribute ecore:reference="foundation:Type" name="consequenceType"
type="xsd:anyURI"/>
```

```
    <xsd:attribute ecore:reference="foundation:Type" name="likelihoodType"
type="xsd:anyURI"/>
    <xsd:attribute name="name" type="ecore:EString"/>
    <xsd:attribute ecore:reference="foundation:Type" name="riskType"
type="xsd:anyURI"/>
  </xsd:complexType>
  <xsd:complexType name="RiskManagementModel">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:sequence>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="riskModels" type="RiskModel:RiskModel"/>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="testPatterns" type="RiskModel:TestPattern"/>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="testProcedures" type="RiskModel:TestProcedure"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="RiskModel">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:sequence>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="riskFunctions" type="RiskModel:RiskFunction"/>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="types" type="foundation:Type"/>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="elements" type="RiskModel:RiskElement"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="RiskNode">
    <xsd:complexContent>
      <xsd:extension base="RiskModel:RiskElement"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="RiskRelation">
    <xsd:complexContent>
      <xsd:extension base="RiskModel:RiskElement">
        <xsd:attribute ecore:reference="RiskModel:RiskNode" name="source"
type="xsd:anyURI"/>
        <xsd:attribute ecore:reference="RiskModel:RiskNode" name="target"
type="xsd:anyURI"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestPattern">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestProcedure">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:attribute ecore:reference="foundation:ValueElement" name="priorityValue"
type="xsd:anyURI"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:schema>
```

Appendix B: Test Pattern Schema

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore"
xmlns:foundation="http://foundation" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
ecore:documentRoot="TestPatternDokument" ecore:nsPrefix="Testpattern"
ecore:package="de.fraunhofer.fokus.rasen.dataexchange.Testpattern">
  <xsd:import namespace="http://foundation" schemaLocation="foundation.xsd"/>
  <xsd:element name="CAPEC_ID" type="xsd:string"/>
  <xsd:element name="CWE_ID" type="xsd:string"/>
  <xsd:element name="Description_Of_Test_Coverage_Items"
type="Description_Of_Test_Coverage_Items_._type"/>
  <xsd:element name="Discussion" type="xsd:string"/>
  <xsd:element name="Effectiveness" type="Level"/>
  <xsd:element name="Effectiveness_Description" type="xsd:string"/>
  <xsd:element name="Effort" type="Level"/>
  <xsd:element name="Effort_Description" type="xsd:string"/>
  <xsd:element name="Generalization_Of" type="xsd:string"/>
  <xsd:element name="Manual_Solution" type="xsd:string"/>
  <xsd:element name="Other" type="xsd:string"/>
  <xsd:element name="Parameter" type="Parameter_._type"/>
  <xsd:element name="Process" type="Process_._type"/>
  <xsd:element name="Reference" type="Reference_._type"/>
  <xsd:element name="References" type="References_._type"/>
  <xsd:element name="Strategy" type="Strategy_._type"/>
  <xsd:element name="Test_Coverage_Item" type="xsd:string"/>
  <xsd:element name="Test_Data" type="xsd:string"/>
  <xsd:element name="Test_Observation_Strategies"
type="Test_Observation_Strategies_._type"/>
  <xsd:element name="Test_Pattern" type="Test_Pattern_._type"/>
  <xsd:element name="Test_Pattern_Catalog" type="Test_Pattern_Catalog_._type"/>
  <xsd:element name="Test_Stimulation_Strategies"
type="Test_Stimulation_Strategies_._type"/>
  <xsd:element name="Test_Technique" type="xsd:string"/>
  <xsd:element name="Test_Tool" type="xsd:string"/>
  <xsd:element name="Weakness_Description" type="xsd:string"/>
  <xsd:complexType ecore:name="DescriptionOfTestCoverageItems"
name="Description_Of_Test_Coverage_Items_._type">
    <xsd:sequence>
      <xsd:element ecore:name="testCoverageItem" maxOccurs="unbounded"
ref="Test_Coverage_Item"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:simpleType name="Language">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="c"/>
      <xsd:enumeration value="c_sharp"/>
      <xsd:enumeration value="java"/>
      <xsd:enumeration value="perl"/>
      <xsd:enumeration value="xml"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="Level">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="veryLow"/>
      <xsd:enumeration value="low"/>
      <xsd:enumeration value="lowToMedium"/>
      <xsd:enumeration value="medium"/>
      <xsd:enumeration value="mediumToHigh"/>
      <xsd:enumeration value="high"/>
      <xsd:enumeration value="veryHigh"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:complexType ecore:name="TestStrategyParameter" name="Parameter_._type">
    <xsd:attribute ecore:name="description" name="Description" type="xsd:string"/>
    <xsd:attribute ecore:name="direction" name="Direction" type="ParameterDirection"
use="required"/>
    <xsd:attribute ecore:name="name" name="Name" type="xsd:string" use="required"/>
    <xsd:attribute ecore:name="type" name="Type" type="xsd:string"/>
  </xsd:complexType>
  <xsd:simpleType name="ParameterDirection">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="in"/>
      <xsd:enumeration value="inout"/>
      <xsd:enumeration value="out"/>
      <xsd:enumeration value="return"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:complexType ecore:name="TestStrategyImplementation" name="Process_._type">
    <xsd:sequence>
      <xsd:element ecore:name="data" name="Data" type="xsd:string"/>
    </xsd:sequence>
    <xsd:attribute ecore:name="language" name="Language" type="Language"
use="required"/>
  </xsd:complexType>
  <xsd:complexType ecore:name="TestPatternReference" name="Reference_._type">
    <xsd:attribute ecore:name="iD" name="ID" type="xsd:string"/>
    <xsd:attribute ecore:name="referenceLink" name="ReferenceLink" type="xsd:anyURI"/>
    <xsd:attribute ecore:name="title" name="Title" type="xsd:string"/>
    <xsd:attribute ecore:name="type" name="Type" type="RefType" use="required"/>
  </xsd:complexType>
  <xsd:complexType ecore:name="TestPatternReferences" name="References_._type">
    <xsd:sequence>
      <xsd:element ecore:name="reference" maxOccurs="unbounded" ref="Reference"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:simpleType ecore:name="ReferenceType" name="RefType">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="CAPEC"/>
      <xsd:enumeration value="CWE"/>
      <xsd:enumeration value="other"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:complexType ecore:name="TestStrategy" name="Strategy_._type">
    <xsd:sequence>
      <xsd:element ecore:name="parameter" maxOccurs="unbounded" minOccurs="0"
ref="Parameter"/>
      <xsd:element ecore:name="implementation" maxOccurs="unbounded" minOccurs="0"
ref="Process"/>
    </xsd:sequence>
    <xsd:attribute ecore:name="description" name="Description" type="xsd:string"/>
    <xsd:attribute ecore:name="name" name="Name" type="xsd:string"/>
  </xsd:complexType>
  <xsd:complexType ecore:name="TestObservationStrategies"
name="Test_Observation_Strategies_._type">
    <xsd:sequence>
      <xsd:element ecore:name="strategy" maxOccurs="unbounded" minOccurs="0"
ref="Strategy"/>
```

```
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType ecore:name="TestStimulationStrategies"
name="Test_Stimulation_Strategies_._type">
    <xsd:sequence>
      <xsd:element ecore:name="strategy" maxOccurs="unbounded" minOccurs="0"
ref="Strategy"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType ecore:name="TestPattern" name="Test_Pattern_._type">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:sequence>
          <xsd:element ecore:name="effort" minOccurs="0" ref="Effort"/>
          <xsd:element ecore:name="effectiveness" minOccurs="0" ref="Effectiveness"/>
          <xsd:element ecore:name="testTechnique" minOccurs="0" ref="Test_Technique"/>
          <xsd:element ecore:name="weaknessDescription" minOccurs="0"
ref="Weakness_Description"/>
          <xsd:element ecore:name="manualSolution" minOccurs="0"
ref="Manual_Solution"/>
          <xsd:element ecore:name="effortDescription" minOccurs="0"
ref="Effort_Description"/>
          <xsd:element ecore:name="effectivenessDescription" minOccurs="0"
ref="Effectiveness_Description"/>
          <xsd:element ecore:name="discussion" minOccurs="0" ref="Discussion"/>
          <xsd:element ecore:name="descriptionOfTestCoverageItems" minOccurs="0"
ref="Description_Of_Test_Coverage_Items"/>
          <xsd:element ecore:name="testData" maxOccurs="unbounded" minOccurs="0"
ref="Test_Data"/>
          <xsd:element ecore:name="testTool" maxOccurs="unbounded" minOccurs="0"
ref="Test_Tool"/>
          <xsd:element ecore:name="references" minOccurs="0" ref="References"/>
          <xsd:element ecore:name="testStimulationStrategies" minOccurs="0"
ref="Test_Stimulation_Strategies"/>
          <xsd:element ecore:name="testObservationStrategies" minOccurs="0"
ref="Test_Observation_Strategies"/>
        </xsd:sequence>
        <xsd:attribute ecore:name="generalizationOf" ecore:opposite="specificationOf"
ecore:reference="Testpattern:Test_Pattern_._type" name="Generalization_Of">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
        <xsd:attribute ecore:name="specificationOf" ecore:opposite="generalizationOf"
ecore:reference="Testpattern:Test_Pattern_._type" name="Specification_Of"
type="xsd:anyURI"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType ecore:name="TestPatternCatalog" name="Test_Pattern_Catalog_._type">
    <xsd:sequence>
      <xsd:element ecore:name="testPattern" maxOccurs="unbounded" minOccurs="0"
ref="Test_Pattern"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Appendix C: Test Report Schema

```xml
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema
xmlns:TestReport="platform:/resource/de.fraunhofer.fokus.rasen.testreport/model/TestRe
port.xsd" xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore"
xmlns:foundation="http://foundation" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
ecore:nsPrefix="TestReport"
ecore:package="de.fraunhofer.fokus.rasen.dataexchange.TestReport"
targetNamespace="platform:/resource/de.fraunhofer.fokus.rasen.testreport/model/TestRep
ort.xsd">
  <xsd:import namespace="http://www.eclipse.org/emf/2002/Ecore"
schemaLocation="platform:/plugin/org.eclipse.emf.ecore/model/Ecore.xsd"/>
  <xsd:import namespace="http://foundation" schemaLocation="foundation.xsd"/>
  <xsd:complexType name="RiskElement">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestCase">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:attribute ecore:opposite="testCase"
ecore:reference="TestReport:TestResult" name="testResult">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestIncident">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:attribute name="cause" type="ecore:EString"/>
        <xsd:attribute ecore:opposite="incident"
ecore:reference="TestReport:TestResult" name="testResult" type="xsd:anyURI"
use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestIncidentReport">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:sequence>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="incident" type="TestReport:TestIncident"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestItem">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:attribute name="systemModelElement" type="ecore:EString"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestLog">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:sequence>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="testResult" type="TestReport:TestResult"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestPattern">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestProcedure">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:sequence>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="log" type="TestReport:TestLog"/>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="testCase" type="TestReport:TestCase"/>
          <xsd:element ecore:resolveProxies="true" minOccurs="0" name="priorityValue"
type="foundation:ValueElement"/>
        </xsd:sequence>
        <xsd:attribute ecore:reference="TestReport:RiskElement" name="riskElement">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
        <xsd:attribute ecore:reference="TestReport:TestPattern" name="testPattern">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestReport">
    <xsd:sequence>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="incidentReport" type="TestReport:TestIncidentReport"/>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="procedure" type="TestReport:TestProcedure"/>
      <xsd:element ecore:resolveProxies="true" name="item"
type="TestReport:TestItem"/>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="testPatterns" type="TestReport:TestPattern"/>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="riskElements" type="TestReport:RiskElement"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="TestResult">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:attribute ecore:opposite="testResult"
ecore:reference="TestReport:TestIncident" name="incident">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
        <xsd:attribute ecore:opposite="testResult"
ecore:reference="TestReport:TestCase" name="testCase" type="xsd:anyURI"
use="required"/>
        <xsd:attribute ecore:unsettable="false" name="verdict"
type="TestReport:TestVerdict"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:simpleType name="TestVerdict">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="pass"/>
      <xsd:enumeration value="fail"/>
      <xsd:enumeration value="none"/>
      <xsd:enumeration value="incomplete"/>
      <xsd:enumeration value="error"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

Appendix D: Metric and Measurement Schema

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns:Metrics="http://Metrics"
xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore"
xmlns:foundation="http://foundation" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
ecore:nsPrefix="Metrics" ecore:package="de.fraunhofer.fokus.rasen.metrics.Metrics"
targetNamespace="http://Metrics">
  <xsd:import namespace="http://www.eclipse.org/emf/2002/Ecore"
schemaLocation="platform:/plugin/org.eclipse.emf.ecore/model/Ecore.xsd"/>
  <xsd:import namespace="http://foundation" schemaLocation="foundation.xsd"/>
  <xsd:element ecore:ignore="true" name="Metric" type="Metrics:Metric"/>
  <xsd:element ecore:ignore="true" name="Measurement" type="Metrics:Measurement"/>
  <xsd:complexType name="Metric">
    <xsd:complexContent>
      <xsd:extension base="foundation:Element">
        <xsd:attribute ecore:reference="foundation:Type" name="scale"
type="xsd:anyURI"/>
        <xsd:attribute ecore:reference="Metrics:Measurement" name="measurements">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
        <xsd:attribute ecore:derived="true" name="statement" type="ecore:EString"
use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="Measurement">
    <xsd:complexContent>
      <xsd:extension base="foundation:ValueElement"/>
    </xsd:complexContent>
  </xsd:complexType>
</xsd:schema>
```

Appendix E: Aggregated Test Report Schema

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns:Metrics="http://Metrics" xmlns:TestMetrics="http://TestMetrics"
xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore"
xmlns:foundation="http://foundation" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
ecore:nsPrefix="TestMetrics"
ecore:package="de.fraunhofer.fokus.rasen.testMetrics.TestMetrics"
targetNamespace="http://TestMetrics">
  <xsd:import namespace="http://www.eclipse.org/emf/2002/Ecore"
schemaLocation="platform:/plugin/org.eclipse.emf.ecore/model/Ecore.xsd"/>
  <xsd:import namespace="http://foundation" schemaLocation="foundation.xsd"/>
  <xsd:import namespace="http://Metrics" schemaLocation="Metrics.xsd"/>
  <xsd:element ecore:ignore="true" name="TestMetric" type="TestMetrics:TestMetric"/>
  <xsd:element ecore:ignore="true" name="TestMeasurement"
type="TestMetrics:TestMeasurement"/>
  <xsd:element ecore:ignore="true" name="AggregatedTestReport"
type="TestMetrics:AggregatedTestReport"/>
  <xsd:element ecore:ignore="true" name="TestProcedure"
type="TestMetrics:TestProcedure"/>
  <xsd:element ecore:ignore="true" name="TestItem" type="TestMetrics:TestItem"/>
  <xsd:complexType name="TestMetric">
    <xsd:complexContent>
      <xsd:extension base="Metrics:Metric"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="TestMeasurement">
    <xsd:complexContent>
      <xsd:extension base="Metrics:Measurement">
        <xsd:attribute ecore:opposite="testMeasurements"
ecore:reference="TestMetrics:TestProcedure" name="testProcedures">
          <xsd:simpleType>
            <xsd:list itemType="xsd:anyURI"/>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="AggregatedTestReport">
    <xsd:sequence>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="testMeasurements" type="TestMetrics:TestMeasurement"/>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="testMetrics" type="TestMetrics:TestMetric"/>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="testProcedures" type="TestMetrics:TestProcedure"/>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="types" type="foundation:Type"/>
      <xsd:element ecore:resolveProxies="true" name="item"
type="TestMetrics:TestItem"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="TestProcedure">
    <xsd:attribute name="identifier" type="ecore:EString" use="required"/>
    <xsd:attribute ecore:opposite="testProcedures"
ecore:reference="TestMetrics:TestMeasurement" name="testMeasurements">
      <xsd:simpleType>
        <xsd:list itemType="xsd:anyURI"/>
      </xsd:simpleType>
    </xsd:attribute>
  </xsd:complexType>
  <xsd:complexType name="TestItem">
    <xsd:attribute name="systemModelElement" type="ecore:EString" use="required"/>
  </xsd:complexType>
</xsd:schema>
```

Appendix F: Foundation Schema

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore"
xmlns:foundation="http://foundation" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
ecore:nsPrefix="foundation"
ecore:package="de.fraunhofer.fokus.rasen.dataexchange.foundation"
targetNamespace="http://foundation">
  <xsd:import namespace="http://www.eclipse.org/emf/2002/Ecore"
schemaLocation="platform:/plugin/org.eclipse.emf.ecore/model/Ecore.xsd"/>
  <xsd:element ecore:ignore="true" name="Element" type="foundation:Element"/>
  <xsd:element ecore:ignore="true" name="Parameter" type="foundation:Parameter"/>
  <xsd:element ecore:ignore="true" name="ParameterValue"
type="foundation:ParameterValue"/>
  <xsd:element ecore:ignore="true" name="CustomType" type="foundation:CustomType"/>
  <xsd:element ecore:ignore="true" name="Type" type="foundation:Type"/>
  <xsd:element ecore:ignore="true" name="ValueElement"
type="foundation:ValueElement"/>
  <xsd:complexType name="Element">
    <xsd:sequence>
      <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="parameters" type="foundation:Parameter"/>
    </xsd:sequence>
    <xsd:attribute name="identifier" type="ecore:EString" use="required"/>
    <xsd:attribute name="name" type="ecore:EString"/>
    <xsd:attribute name="description" type="ecore:EString"/>
    <xsd:attribute ecore:reference="foundation:Type" name="type" type="xsd:anyURI"/>
  </xsd:complexType>
  <xsd:complexType name="Parameter">
    <xsd:sequence>
      <xsd:element ecore:resolveProxies="true" minOccurs="0" name="parameterValue"
type="foundation:ParameterValue"/>
    </xsd:sequence>
    <xsd:attribute name="name" type="ecore:EString"/>
    <xsd:attribute name="description" type="ecore:EString"/>
    <xsd:attribute ecore:reference="foundation:Type" name="parameterType"
type="xsd:anyURI"/>
  </xsd:complexType>
  <xsd:complexType name="ParameterValue"/>
  <xsd:complexType name="CustomType">
    <xsd:complexContent>
      <xsd:extension base="foundation:Type">
        <xsd:sequence>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="definitions" type="foundation:Parameter"/>
          <xsd:element ecore:resolveProxies="true" maxOccurs="unbounded" minOccurs="0"
name="typeProperties" type="foundation:Parameter"/>
        </xsd:sequence>
        <xsd:attribute ecore:reference="foundation:Type" name="baseType"
type="xsd:anyURI"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:complexType name="Type">
    <xsd:attribute name="name" type="ecore:EString"/>
  </xsd:complexType>
  <xsd:complexType name="ValueElement">
    <xsd:complexContent>
```

```
      <xsd:extension base="foundation:ParameterValue">
        <xsd:attribute name="value" type="ecore:EString"/>
        <xsd:attribute ecore:reference="foundation:Type" name="type"
type="xsd:anyURI"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:simpleType ecore:name="COMMON_BASE_TYPE" name="COMMON_BASE_TYPE">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="REAL"/>
      <xsd:enumeration value="INTEGER"/>
      <xsd:enumeration value="STRING"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

Appendix G: ARIS Import/Export Schema

The following XSD specified the Import/Export Format in ARIS Business Architect to export the Models for automated testing and import the weaknesses back into the graphical model:

```
<?xml version="1.0" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="subComponentType">
    <xs:sequence maxOccurs="unbounded" minOccurs="0">
      <xs:element name="subComponentName" type="xs:string"/>
      <xs:element name="subComponent" type="subComponentType"/>
      <xs:element maxOccurs="unbound" minOccurs="0" name="cwe">
        <xs:simpleType>
          <xs:restriction base="xs:integer">
            <xs:minInclusive value="0"/>
            <xs:maxInclusive value="1000"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="vignetteEntryType">
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="0"/>
      <xs:maxInclusive value="9"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="productName" type="xs:string"/>
  <xs:element name="vignette">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="HA" type="vignetteEntryType"/>
        <xs:element name="RC" type="vignetteEntryType"/>
        <xs:element name="MD" type="vignetteEntryType"/>
        <xs:element name="UE" type="vignetteEntryType"/>
        <xs:element name="GP" type="vignetteEntryType"/>
        <xs:element name="EA" type="vignetteEntryType"/>
        <xs:element name="BP" type="vignetteEntryType"/>
        <xs:element name="RD" type="vignetteEntryType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element maxOccurs="unbounded" minOccurs="0" name="component">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="componentName" type="xs:string"/>
        <xs:element name="subComponent" type="subComponentType"/>
        <xs:element maxOccurs="unbound" minOccurs="0" name="cwe">
          <xs:simpleType>
            <xs:restriction base="xs:integer">
              <xs:minInclusive value="0"/>
              <xs:maxInclusive value="1000"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```