14

Domain Verification and Validation

- The **prerequisite** for studying this chapter is that you have a reasonable grasp of the previous stages of domain engineering: from domain acquisition, via analysis and concept formation, to domain description (i.e., domain modelling).
- The **aims** are to briefly introduce the concepts of domain verification (including model checking and testing) and validation, and to cover some of the attendant principles and techniques.
- The **objective** is to complete your education and training so as to become a professional domain engineer.
- The **treatment** is informal.

The Right Domain — The Domain Right [42]

- Domain Validation: Validate to get the right domain.
- Domain Verification: Verify (model check, test) to get the domain right.

14.1 Introduction

Let us first review where we are in the process of describing the domain development process and its method principles and techniques:

(i) First we focused on the core aspects of domain modelling: The "whats" and "hows" of a domain model. We could call this the "production technology". (i.1) We covered the concepts of abstraction of phenomena and concepts in Chap. 5, and (i.2) the attributes and facets of what is being described in domain models in Chaps. 10–11. (i.3) We covered, in between, the issues of stakeholders and their perspectives in Chap. 9. That coverage explained "what" a domain model should contain, the abstractions possible, the facets "mirrored", and — notably — with respect to the stakeholders and the perspectives to be dealt with.

(ii) Then we focused more on "how". In contrast to "production technology" we could call this "how" the "process technology". (ii.1) First, we focused on the process, principles and techniques of domain acquisition, Chap. 12, that which "begins" the domain development work. (ii.2) Then we covered the process, principles and techniques of domain analysis and concept formation in Chap. 13. After domain acquisition, domain analysis and concept formation follows the domain modelling proper. Finally, we focus on domain validation and verification — the topic of this chapter.

The purpose of the above review has been to put the somehow "reverse" ordering of the chapter sections "straight" with respect to the ordering of the domain development processes. We can now summarise the domain development process (even before we have covered the notions of verification and validation). After producing the appropriate informative documents: needs and ideas, concepts, scope and span, synopsis, and contracts, one proceeds to identifying domain stakeholders and establishing liaison with members of domain stakeholder groups. Then we move on to domain acquisition: interviews, studies, questionnaire formulation and domain stakeholders' replies to these, ending wit domain description unit indexing and an elicitation report. This acquisition is followed by domain analysis and concept formation. Then we do the actual domain modelling. And, finally, we perform domain verification and validation.

14.2 Domain Verification

In this chapter (as we shall also do in Chap. 22 on requirements validation and verification) we use the term verification to also cover the concepts of model checking and testing.

Characterisation. By *domain verification* we shall understand a process, and the resulting (analytic) documents, in which some domain descriptions are being analysed in order to ascertain whether what is being described satisfies certain (claimed or otherwise expected) properties. ∎

So what — really — is the difference between domain validation and domain verification?

- In validation we examine the domain model to make sure we are modelling what the domain stakeholders think that domain is: *Validation gets the right domain model.*
- In verification we examine whether our domain model "hangs together," such as the domain engineers want it to be: *Verification gets the domain model right.*

(The above, oft-quoted distinction was, it seems, first reported by Barry Boehm in [42].)

Verification is adjoint to validation: Both validation and verification are needed. Usually verification precedes validation. Verification work typically proceeds as follows: Desired properties of the domain model — properties that do not transpire immediately from the domain description — are formulated, informally or formally. Then "proofs" by "verbal" arguments, or some form of symbolic testing, or formal proofs, or model checking, are performed in order to check that the desired property holds of the domain model.

So verification, to us, includes, rearranging the terms a bit, (i) informal reasoning: (i.1) "proofs" by "verbal" arguments and (i.2) testing; (ii) formal reasoning: (ii.1) formal proofs and (ii.2) model checking. By informal reasoning we shall, however, mean "proofs" by "verbal" arguments.

14.2.1 Informal Reasoning

Characterisation. By *informal reasoning* we shall understand a carefully phrased series of arguments, which, as a whole, convinces an audience of the validity of what is concluded. ∎

Human beings often reason, but are not always careful in doing so. Informal reasoning demands great care.

14.2.2 Testing

Characterisation. By *testing* we shall understand that a domain description is provided with set values for all relevant arguments (the test data), with the description then being evaluated ("executed") for those arguments. The test then results in a "final value" of the description for those arguments. ∎

Such a "final value" may be a complicated quantity. Typical final values could be an execution sequence, or a trace of description points, with a set of variable values for each step in the sequence (i.e., a trace).

In another way of phrasing it: Testing is a systematic search for a counterexample to a claim (or proof) of correctness. Testing, till recently, has basically been a heuristics-based science. An important part of testing is text analysis. If domain description parts have been formalised, then theory-based testing technologies have been or can be developed and can be used for testing. Chapter 29, Sect. 29.5.3, covers testing in more detail.

14.2.3 Formal Proofs

Characterisation. By a *formal proof* we shall understand a given domain description, a statement (a theorem) to be proved and a proof that the domain description satisfies the statement: This proof refers to a proof system for the language in which the domain description is expressed (axioms and inference

rules), and is otherwise a sequence, composed from steps, where each step in the sequence is like a theorem (a lemma), a statement, and where pairs of steps in the proof sequence are related, i.e., are justified, by the axioms and the inference rules. ∎

14.2.4 Model Checking

Characterisation. By *model checking* we shall understand [55] *a method for formally verifying usually concurrent systems, whose usually extremely large, practically speaking infinite state systems, have been reduced to manageable finite-state systems.*

We augment this characterisation by the following: In model checking a somehow executable abstraction of the thing to be checked is programmed. That model is then subject to certain forms of executions in which specified properties are checked. These executions, for example, check whether the model is able to enter certain states or not. ∎

Domain descriptions about such finite-state systems are typically expressed as temporal logic formulas. Efficient symbolic algorithms are used to traverse the (state machine) model defined by the system and to check if the domain description holds or not, i.e., whether the model execution "enters" appropriate states, albeit for a "reduced" set of possible states of systems. Extremely large state-spaces can often be traversed in minutes. Seminal books on model checking are [26, 59, 173].

14.3 Domain Validation

Characterisation. By *domain validation* we shall understand a process, and the resulting (analytic) documents, in which some domain descriptive documents are being coinspected by domain stakeholders and domain engineers, and in which whatever is being described is being positively and/or negatively reviewed with reference to the elicitation report and with respect to whatever the stakeholders might now realise about their domain. This includes pointing out, if necessary, inconsistencies, incompletenesses, conflicts and errors of description that may change the elicitation report. ∎

Domain validation is possibly interwoven with domain verification work, see Sect. 14.2.

14.3.1 The Domain Validation Documents

In order to perform domain validations, the validators need the following (input) documents: (i) the list of domain stakeholders; (ii) the domain acquisition

documents: questionnaire, and the collection of indexed description units; (iii) the rough-sketch, terminology, narrative, and possibly — if produced — the formalisation documents that constitute the domain description proper; and (iv) the domain analysis and concept formation documents. That is, the validators need access to basically all documents produced (so far) in the domain modelling effort.

In order to complete domain validation, the validators produce the following (output) documents: (i) a possibly updated domain stakeholder document; (ii) possibly updated domain acquisition documents; (iii) possibly updated rough sketches, terminology, narrative, and — if relevant — the formalisation documents; (iv) possibly updated domain analysis and concept formation documents; and (v) a domain validation report. We now cover some aspects of the necessarily informal validation process.

14.3.2 The Domain Validation Process

Domain validation proceeds as follows: Domain engineers "sit together" with stakeholders and review, line by line, the domain model, holding it up against the previously elicited domain description units, while then noting down any discrepancies. In doing domain validation, domain stakeholders usually read the informal, yet precise and detailed narrative descriptions. No assumption is made as to their ability to read formalisations. On the contrary: It is assumed that they cannot read formal specifications.

For reasonably large-scale projects the customer may hire professional consultants who can also study the formalisations. This is just like future ship owners hiring Lloyd's Register of Shipping [224][1] to check ship designs in preparation for insurance companies to take on insurance risks.[2]

Domain validation (and verification) ends with a signed domain validation (and verification) report. This report either OKs the domain model, or points out required corrections in the elicitation report, in the domain analysis and concept formation report, and in the domain model.

14.3.3 Domain Development Iterations

Thus domain validation (and verification) can be an iterative process, alternating possibly with further domain verification, further elicitation report work, further domain analysis and concept formation work, and with further domain modelling work. The domain validation process may end with further domain validation (and verification) work.

[1] Or such similar companies as Norwegian Veritas [262], Bureau Veritas [51] or TÜV [356].

[2] The staff of these, and similar design quality assurance companies, are oftentimes very sophisticated software engineers, well-versed in formal software development and verification methods.

14.4 Discussion

14.4.1 General

This chapter is closely related to Chap. 22: Requirements Validation and Verification. The reader is advised to carefully review this material before reading Chap. 22. After having studied Chap. 22, the reader is encouraged to compare the two chapters: 14 and 22.

We have treated aspects of domain validation and verification — in the same chapter since they relate in many ways. And we have used the term verification, primarily to stand for formal proofs, but, secondarily, also for model checks and tests.

14.4.2 Principles, Techniques and Tools

We summarise:

Principle. *Domain Validation:* To ensure that the domain described is the right domain. ∎

Principle. *Domain Verification:* To uncover a domain theory, i.e., to get the domain descriptions right. ∎

Techniques. *Domain Validation:* In summary, human, collaborative document inspection (Sect. 14.3.2). ∎

Techniques. *Domain verification* techniques, based on formal descriptions, include those that enable formal verification (of posed lemmas and theorems), model checking, and tests, while domain verification techniques, based on informal descriptions, basically amount to informal, concise reasoning. ∎

Tools. Since *domain validation* is basically an informal process, the tools are those that support document cross-referencing between domain description units and narrative domain descriptions and domain terminologies, and data mining based on such documents. ∎

Tools. *Domain verification* based on formal descriptions requires such tools as, for example, proof assistants and theorem provers, model checkers, and test generators and tester monitors; whereas domain verification based on informal descriptions basically requires human reasoning. ∎

14.5 Exercises

14.5.1 A Preamble

We refer to Sect. 1.7.1 for the list of 15 running domain (requirements and software design) examples; and we refer to the introductory remarks of Sect. 1.7.2 concerning the use of the term "selected topic".

14.5.2 The Exercises

The first 4 exercises (14.1–14.4) of this chapter are *closed book* exercises. That means that you are to try write down a few lines of your solution before you check with the appropriate section for our answer to the questions.

Exercises 14.6 and 14.5 test the problem solver's ability to lead a group of two or more domain validators, respectively domain verifiers.

Exercise 14.1 *Domain Validation Documents.* Which are the domain development documents that are needed in order to commence proper domain validation, and which are the resulting documents?

Exercise 14.2 *Domain Validation Process.* Outline, in brief, i.e., in a few itemised lines, the domain validation process.

Exercise 14.3 *Domain Verification, Model Checking and Testing.* Explain, in brief, in a few itemised lines, the concepts of formal verification, of model checking and of testing.

Exercise 14.4 *Domain Validation Versus Domain Verification.* Explain in two itemised lines the difference between the objectives of domain validation and domain verification.

Exercise 14.5 *Domain-Specific Verification.* For the fixed topic, selected by you, and on the basis of your solutions to some of Exercises 10.1–10.7, and/or some of Exercises 11.1–11.7, suggest, preferably three or more, issues that may need special attention during domain verification.

Exercise 14.6 *Domain-Specific Validation.* For the fixed topic, selected by you, and on the basis of your solutions to some of Exercises 10.1–10.7, and/or some of Exercises 11.1–11.7, suggest, preferably three or more, issues that may need special attention during domain validation.