Chapter 5
Safety Case Patterns, Notations and GSN

What This Chapter Is About:

- Information regarding safety case patterns and notations.
- We explain how an agile approach can be combined with GSN.

5.1 Safety Case Patterns and Notations

5.1.1 The Contents of a Safety Case

Whatever notation we use for safety cases—standard prose, structured prose or diagrams—there are certain dos and don’ts that apply in order to arrive at a sound argument with corresponding evidences. First and foremost, we need to identify:

- Context—where are our arguments valid? This might again be split into safety context and use context.
- Assumptions—which assumptions have we made in order to arrive at our arguments?

The safety cases must have a top claim—“the system is safe because…” The top claim is then decomposed into sub-claims that support the top claim and so on. Another important topic that must be discussed early in the process is “What is a valid argument?” Let us look at an example:

“We have found all errors because we used test method X.” Many assessors will probably accept this if method X is the method specified in the standard and only require more information if another method is used. This is, however, a dangerous approach since there are several ways to use a method. In a safety case, it is important to show that the method or technique applied is (1) suitable for its intended purpose, (2) used in the right way (3) by people with the right competence and (4) that a sufficient amount of effort was used.

Example: Boundary Testing
As an example, consider “boundary testing.” There is a considerable difference between just running a couple of boundary tests and doing a solid analysis, identifying all boundaries and then testing them. The first gives no confidence; the other will give a lot.

Thus, “We have used method X” should not be considered a valid argument. What we need would be something like “We have found all errors because”:

- The participants have a long and documented experience in the methods used—see document named A.
- We used method Y to define the test cases and the expected results—see document named B.
- All the tests ran as expected—see test log named C.

In this case, we can inspect the test cases and accept or reject them. We can check both the process and its results. Thus, this approach will build confidence, mainly because it is possible to check all steps. Note that arguments need to be to the point. Long, complicated arguments will be difficult to read and understand and, in addition, they may create an impression that someone is trying to hide something. Really trying to hide something would be unethical. However, making people behave in an ethical way is not something that can be enforced—it has to be grounded in the company’s culture.

There is a strong connection between safety cases and development process. Thus, we need to plan in advance which evidence we need and when and how we will produce them. Some of these evidences are also required by the standard—e.g. the test report—but many are needed only for the safety case. The reasons for this are that:

- The safety case needs information to be used as evidence. Thus, the process must contain activities that produce this information.
- The process needs to perform certain activities due to standard requirements, assessor interpretation of requirements or customer requirement. The argument that these activities are safe must be included in the safety case.

### 5.1.2 Normal Prose Safety Case

It is always possible to present the safety case argument as unstructured prose even though this may make the arguments difficult to follow. Surveys done by SINTEF Digital indicate that most industrial safety cases are written using normal, unstructured prose. The following is an example of a structured safety case argument turned into an unstructured safety case argument—see Hawkins and Kelly (2011). The structured case can be found in the next subsections:

Potential hazards are acceptably managed. We claim this to be true because potentially dangerous design errors are not introduced—see list of potentially dangerous design errors (Fig. 5.1). Moreover, we will show that either the process...