"On the Inevitable Intertwining of Specification and Implementation" Revisited

David Gelperin
ClearSpecs Enterprises
Fountain Hills, USA
david@clearspecs.com

***Abstract* — This paper introduces a new form of requirements specification. The new form, *intentionally imprecise specifications*, should be used when aspects of the application, its domain, or its implementation technology are poorly understood or design alternatives and tradeoffs have not been identified. Intentionally imprecise specifications accurately state what is necessary, but not yet fully understood.**

***Index Terms* — intentionally imprecise specifications, excess precision, imperfect foresight, requirements intertwining, subjective verification.**

I. Introduction

In the Swartout and Balzer paper [1], the authors claim that specification and implementation **must** be intertwined. Unfortunately, by argument and example they only show that these processes **may** be intertwined. Their argument rests on the concept of "imperfect foresight". In fact, when stakeholders have deep understanding of the application and its domain, the weaknesses of the implementation technology, and the design alternatives and tradeoffs, intertwining is unlikely because foresight is nearly perfect. Developers, with years of experience building billboard websites, are likely to have nearly perfect foresight for their next one.

When stakeholders don't have deep understanding of these areas, specifications may change due to discoveries made during design or implementation. When foresight is imperfect, the analysis in [1] leads to the conclusion that any aspect of a specification may be changed because of deeper understanding i.e., any aspect may be wrong. Without deep understanding, the analysis implies that all specifications are merely pseudo-requirements until an implementation is thoroughly verified against its (possibly changed) specifications. This is based on the assumption that specifiers don't know what they don't know i.e., that all unknowns are unknown.

In practice, some unknowns are known. Consider the following specification.

> Any self-driving vehicle approved for use outside Australian cities must "recognize kangaroos" on or near the roadway and take proper action [2].

This statement marks some unknowns by the intentional use of imprecise words i.e., "near the roadway" and "proper action". It also indicates that the ability to accurately and cost-effectively recognize kangaroos at various distances and of various sizes and orientations is unknown. This ability will need to be explored via prototyping, possibly using deep learning.

A traditional view of the quality of a requirement's specification would consider this statement defective because it is imprecise and possibly unachievable. We choose to view it as an "intentionally imprecise specification" to be refined by improved understanding gained during research and development. In the example above, a refinement may take the form of a glossary entry for "kangaroo near the roadway".

Intentionally imprecise specifications state important restrictions while signaling the need for further analysis and refinement. Since some unknowns may be unknown, some traditional (precise) specifications may also need to be changed.

II. Traditional specifications

A **traditional** requirements specification is precise enough to clearly and accurately specify a complete, necessary, achievable, and objectively verifiable restriction on an implementation and includes pre and post conditions on behavior.

An **ideal** requirements specification is a traditional specification containing no design information. For example, "Develop a smart phone application that plays tic-tac-toe and never loses." is an ideal specification. The required use of a specific algorithm is a traditional, but not ideal, specification.

III. Excess precision

Requirements (restrictions) provide guidance when developing a solution. The challenge is to identify exactly which restrictions are necessary and to accurately describe these restrictions with necessary and sufficient precision.

The precision of a specification may be inadequate, sufficient, or excessive. Excess precision adds unnecessary restrictions that may endanger the development of optimal or even effective solutions by excluding effective designs.

Inadequate precision motivates discussion. Excess precision makes it appear that discussion is unnecessary. This means that care must be taken when asking stakeholders to be precise, because they may say more than they know. Excess precision may result from excess confidence, lack of confidence, inadequate understanding of this risk, or lack of validation. Excess precision may also express a want, rather than a need.

Sufficient precision may entail imprecise descriptions implying that important information is unknown at the moment. For example, when is a kangaroo "near the roadway"? How far and how fast can an average kangaroo hop? At what points on the roadway would a kangaroo hopping toward the roadway first arrive? How far must the kangaroo be from the roadway to assure a vehicle will be beyond the points of arrival? One might define less than this distance to be "near the roadway". Alternatively, "near the roadway" may be a function of the size of the kangaroo and the speed of the vehicle.

All of this means that imprecise statements may be the most accurate at a particular time in a project. Their use reduces the risk of excess precision and signals the need for analysis to understand their deeper meaning.

IV. Intentionally imprecise specifications

An **intentionally imprecise specification** states a complete, necessary, possibly achievable, and subjectively verifiable restriction on an implementation. **Subjective verification** entails agreement by a "sufficient majority" of the stakeholders, where the composition of the "sufficient majority" is precisely defined. To assess compliance with the restriction, subjective verification might entail agreement on the definition of a comprehensive set of initial situations to be handled during a simulation or agreement on a Planguage-like [3] measurement.

Intentionally imprecise specifications trigger a mixture of research and development. Their use invokes a social contract between stakeholders and system acquirers. Stakeholders provide a clear statement of objectives. Acquirers research alternative strategies or designs and their costs. The alternatives are presented to the stakeholders, who select one or more alternatives.

For example, imagine that a product manager for a new blood analyzer says the average cycle time must be "reduced". This is an imprecise specification and the product manager is the sufficient majority.

Assume the current cycle time is 3.5 minutes and the competitor's best is 3.1 minutes. Developers do research and report the average cycle time can be reduced as described in the following table.

TABLE I. Alternative meanings of "reduced cycle time"

Reduced cycle time	Strategy	Increased cost per unit	Increased development time
3.2 min.	Modify analysis algorithm	$0	4 weeks
2.8 min.	Faster standard processor	$150	6 weeks
2.5 min.	Faster custom processor	$400	16 weeks

Based on a set of marketing assumptions, the product manager chooses the precise meaning or meanings of "reduced cycle time" and the associated cost and development time increases.

For some imprecise requirements, as in the blood analyzer example, imprecision is easy to resolve during design. For other imprecise requirements, imprecision is harder to resolve. Consider the following example.

> Develop an elevator control system for tall buildings. When fire is detected, the control system must help to safely evacuate as many people as possible. It must also enable responders to safely reach the fire and endangered locations as quickly as possible.

"Safely evacuate", "safely reach", "endangered locations", and "as quickly as possible" are imprecise concepts in a poorly-understood corner of a well-understood domain.

Assume a *sufficient majority* is precisely defined for this example. Each stakeholder would have to assess alternative designs and subjectively decide which, if any, of the designs complies with the imprecisely-stated requirements. The stakeholders might help themselves by defining a comprehensive set of initial situations and running simulations of alternative designs. Finally, a sufficient majority would need to agree on one of the compliant alternatives.

The following might be part of a design description for this system.

> Whenever the system receives a verified fire warning, it must transition into evacuation mode and light the service indicators on each call button panel on each floor as follows. Whenever the integrity sensors in a shaft report possible problems, the service indicators on all floors for that shaft should be red. In addition, the car should be brought to the ground floor, emptied, and placed out-of-service. If a car is already out-of-service, its service indicators should be red. One in-service car in each bank should be brought to the ground floor, emptied, and placed in responder mode. Its service indicator on the ground floor should be green, while on higher floors, they should be red. All other service indicators should be green.

Deciding if this design fragment supports its imprecise specification may be difficult.

Sometimes, an imprecisely-stated requirement needs to be factored into specific situations. For example, consider "If a vehicle's motor dies on a roadway (e.g., out of gas), the self-driving guidance system must respond safely." There are many situations in which a motor might die (e.g., when stopped at a traffic light or when doing 75 mph on a highway). One (precise) derived requirement might be:

> If the motor dies on a roadway and the vehicle is stopped, then turn on the hazard lights and brake lights and sound the horn.

A second (imprecise) derived requirement might be:

> If the motor dies on a roadway and the vehicle is moving and a safe and traversable glide path is available, then turn on the hazard lights, signal

appropriately, steer onto the glide path, decrease speed as needed, and brake when stopping is safe.

Notice that the parent requirement and its factors assume that a collection of derived conditions can be recognized e.g., the availability of a safe and traversable glide path. These requirements also assume that safe responses can be precisely defined and carried out. Also notice that every hazard condition for a self-driving vehicle, excluding hazards in the software itself and its processors, can be factored as shown in this example.

In regulated industries, the following information should be recorded for each imprecisely-stated requirement to support the selection of its design:

- Identification of *sufficient majority*
- Definitions of imprecise terms
- Design alternatives considered and assessments of each
- Verification strategy used

V. All imprecise statements are not requirements

Some imprecise statements are not requirements because they are not achievable or not subjectively verifiable. "The system must be highly secure and very easy to use." is not achievable because these qualities conflict. An imprecise statement may not be subjectively verifiable because a "sufficient majority" has not been defined or because compliance is unknowable. For example, compliance with "The software shall be perfectly safe." is unknowable. If a set of hazards (H1, H2, …, Hn) has been identified, a statement such as "The software shall effectively mitigate hazards H1 through Hn." could be an imprecisely-stated requirement if a "sufficient majority" has been defined.

VI. Verification

A traditional view of verification entails assessing compliance with fixed specifications. Understanding the ways that specifications, designs, and implementations intertwine provides a different perspective. Some specifications will not change and therefore a traditional view of assessing compliance will be appropriate. Some specifications will change or be refined by deeper understanding. In these cases, verification entails assessing consistency between modified specifications and their implementations. This will entail objective or subjective verification.

VII. Conclusion

Imprecisely-stated requirements are a valuable supplement to precise requirements. When aspects of the application, its domain or its implementation technology are poorly understood or design alternatives and tradeoffs have not been identified, imprecise specifications can accurately state what is necessary, but not yet fully understood. They provide specific guidance on the analysis required to increase understanding of an acceptable solution.

Using imprecise specifications and accurately assessing the stability of precise specifications will enable requirements risk to be more accurately determined.

References

[1] Swartout, William and Balzer, Robert "On the Inevitable Intertwining of Specification and Implementation" Communications of the ACM July 1982 Vol. 25 No. 7

[2] https://www.theguardian.com/technology/2017/jul/01/volvo-admits-its-self-driving-cars-are-confused-by-kangaroos

[3] Simmons, Erik (2001) "Quantifying Quality Requirements Using Planguage" [Available as reference 2.22 at www.understandingrequirements.com]