

Aphorisms about Requirements

David Gelperin
LiveSpecs Software
dave@livespecs.com

The requirements puzzle will have pieces missing

Expect requirements development to be a shared learning experience

For wicked problems, the picture keeps changing

Right the first time can be the enemy of good enough

Risk requires higher resolution

The requirements puzzle will have pieces missing refers to the many stakeholder needs and preferences that need to be discovered, placed in the picture, and understood during requirements development. Discovery entails becoming aware of and learning about and from:

info sources

documents and existing systems as well as stakeholders

stakeholder requirements (needs & preferences)

application concepts, terminology, tasks, & frequencies

stakeholder goals & values

stakeholder opinions & critical assumptions

stakeholder ways of working

Stakeholder requirements are discovered in both a geographic sense (i.e., finding after crossing an expanse) and a scientific sense (i.e., piecing together clues and testing hypotheses).

Failure to find critical pieces can be very expensive – but you already knew that.

Expect requirements development to be a shared learning experience means that listening carefully and understanding stakeholder needs is always necessary, but often insufficient for designing effective system requirements. It is difficult for anyone to grasp the full implications; both positive and negative, of requested features. Pressures such as innovation [Ro] and change in the application domain and in technology make understanding even more difficult. Rare is the stakeholder who has a firm command of

all critical factors affecting a system solution. Therefore, each stakeholder should expect to gain important insights during the discovery and design process.

For wicked problems, the picture keeps changing refers to the existence of inherently wicked problems [CW] or parts of problems and the need to recognize when a system you are trying to describe contains wicked aspects.

Paraphrasing Conklin & Weil, wicked problems have no definitive statement, nor definitive solution. They are sets of interlocking issues and constraints, inevitably changing over time, that are adequately understood only after you have attempted several solutions. Wicked problem solving is inherently non-linear and ends when you run out of resources, not when a perfect solution emerges.

Dealing with the wicked aspect of products or projects requires humility, flexibility, and creativity, since linear approaches will always fail.

In addition to inherently wicked problems, poor development practices can turn inherently tame problems wicked. Unfortunately, neither wicked nor tame problem solving strategies will succeed in these cases, because the problem is not the problem.

Right the first time can be the enemy of good enough refers to the inevitability of change. Many requirements change a little or a lot because of changes in the understanding of stakeholder needs, the system design, the application domain, or the schedule. Right the first time requires specification when understanding is at a minimum and views subsequent change as a process failure. This encourages the hiding of new insight and discourages proposals for necessary change. The result may be what was originally specified, but not what is needed.

The inevitability of change should not be an excuse for undisciplined development, but evidence that a balanced, cost-effective approach to requirements development and management is needed. Change must be managed and judgment used to decide if the current requirements are “good enough” to proceed.

Risk requires higher resolution refers to the need for requirements elaboration which adds essential details to risky system requirements using one or more precise models. Among such models are:

- active glossaries
- full-dress [Co] or precise use cases [Ge2]
- action contracts [Ge1]
- acceptance test designs [CJ, Chapter 4] [KF, Chapter 7]
- formal languages (e.g. OCL, Z)

As others have observed, precision **always** happens (i.e., code & tests), the issue is not **if**, but **how soon** it starts.

For tamer problems or parts, early investments in precise modeling can avoid many late stage surprises (costs) otherwise obscured by ignorance or ambiguity. However, circumstances will dictate which precise models will be cost-effective.

References

- [CJ] Craig, Rick D. & Jaskiel, Stefan P. **Systematic Software Testing** Artech House Publishers 2002
- [Co] Cockburn, Alistair **Writing Effective Use Cases** Addison-Wesley 2001
- [CW] Conklin, E. Jeffery & Weil, William “**Wicked Problems: Naming the Pain in Organizations**” Available for download at <http://depts.washington.edu/cmweb/cero/wicked.pdf>
- [Ge1] Gelperin, David “**Specifying Consequences with Action Contracts**” Available for download at www.livespecs.com
- [Ge2] Gelperin, David “**Precise Use Cases**” Available for download at www.livespecs.com
- [KF] Kaner, Cem Falk, Jack Nguyen, Hung Quoc **Testing Computer Software** Van Nostrand Reinhold 1993
- [Ro] Robinson, James "Eureka! Why Analysts Should Invent Requirements" IEEE Software July/August 2002 Available for download at www.volere.co.uk/inventing.pdf