**The Historical Accident of Waterfall Validity?**

The waterfall model was a response to ad hoc code-and-fix development in the 1960s. Note—as has been explored in the "[Early Historical Project Evidence](mk:@MSITStore:C:\Documents%20and%20Settings\Administrador\Escritorio\0131111558.chm::/0131111558_ch06lev1sec3.html#ch06lev1sec3)" section on page 79—disciplined IID did exist as a contemporary alterative. It was not inevitable that the waterfall be widely promoted (rather than IID) starting in the 1970s; it has a bit of an accidental quality to it, as I discovered in my historical research.

Misunderstanding started early, with the article and author most often cited for the waterfall: "Managing the Development of Large Software Systems" by Winston Royce, in Proceedings of IEEE Westcon, 1970.

In this article—sometimes (incorrectly) identified as the paragon of single-pass waterfall—Royce actually recommends an approach different than what devolved into the popular notion of waterfall, with its strict single-pass sequence of requirements analysis, design, and development phases. He recommends to "do it twice." To quote:

If the computer program in question is being developed for the first time, arrange matters so that the version finally delivered to the customer for operational deployment is actually the second version insofar as critical design/operations areas are concerned.

He goes on to suggest that a 30-month project might have a 10-month throw-away "pilot model" and justifies its necessity when there are novel elements and unknown factors (hardly a unique case).

Winston's son Walker Royce described what his (deceased) father felt about the misinterpretation of the waterfall that was attributed to him, and the widespread promotion of a document-driven single-pass approach [[LB03](mk:@MSITStore:C:\Documents%20and%20Settings\Administrador\Escritorio\0131111558.chm::/0131111558_biblio01.html#biblio01entry69)]:

He was always a proponent of iterative, incremental, evolutionary development. His paper described the waterfall as the simplest description, but that it would not work for all but the most straightforward projects. The rest of his paper describes [iterative practices] within the context of the 60s/70s government contracting models (a serious set of constraints).

It is ironic that the author of the seminal waterfall paper was a proponent of iterative and evolutionary development, that his paper was only describing a process for the most straightforward projects, and that Royce did not subscribe to the simplified single-pass waterfall, as it is often described.

Note also that the paper was not based on research into various successful lifecycle choices for large, novel projects. It did not cite any evidence, trends, or other projects. Indeed, the paper opens with this sentence:

I am going to describe my personal views about managing large software development.

Even by the time DOD-STD-2167 was adopted in the 1980s there was a growing body of experience and recommendations by thought leaders to avoid rather than embrace the (misunderstood) waterfall, but this knowledge did not find its way into the 2167 standard. Why? One part of the reason is that these standards were usually combined with and influenced by MIL-STD-1521B, another standard that required a series of document-driven reviews, such as a requirements review. I found another part of the answer in Boston.

In 1996 I visited the Boston area and had lunch with the principal author of 2167. He expressed regret for the creation of the rigid single-pass waterfall standard. He said he was influenced by common knowledge and practice of the time, plus other standards (e.g., 1521B). He was not familiar with the practice of timeboxed iterative development and evolutionary requirements at the time, and in hindsight, said he would have made a strong IID recommendation, rather than what was in 2167.

It is no small irony that 2167 was then used as input to other standards, both within the United States and internationally. For example, the British JSP-188, German V-Model (also the basis of the Austrian and Swiss standards), and the French GAM-T-17 were influenced by 2167, with an emphasis on single-pass waterfall phases, and early large, signed-off specifications before construction.

Although DoD reports started to publicly caution against 2167 and the waterfall in 1987, and 2167 was replaced with an IID-promoting standard in 1994, other governments and standards bodies that had drawn from 2167 did not likewise update their standards, apparently unaware of the changes afoot.

As I uncovered these stories, I was struck by the unintended influence a small number of individuals had on a world of standards and projects. And, how the misunderstanding and speculation that actually lay behind these standards led to a kind of accidental perception of validity for the waterfall, making of it a mirage as the "obvious tried-and-true" best practice for large, novel, complex projects, when in fact this was not the case.

**Why Did Waterfall Promotion Continue?**

H. L. Mencken quipped, "For every complex problem, there is a solution that is simple, neat, and wrong." In the history of science it is normal that lesser ideas first hold the dominant position, even in the absence of results. Western cosmology's Earth-centric universe dominated Europe for over a millennium until enough evidence and brave souls accumulated to depose the model. Software development is a young field, so it is no surprise the simple formula of "requirements, design, implementation" held sway during the first attempts to create a skillful development process. Other reasons for the early and ongoing adoption of the single-pass waterfall idea include:

* Few actually read Royce's original waterfall paper [[Royce70](mk:@MSITStore:C:\Documents%20and%20Settings\Administrador\Escritorio\0131111558.chm::/0131111558_biblio01.html#biblio01entry87)]. Its iterative flavor was lost, and it devolved from the nuanced evolutionary description he gave, to a simple single-step lifecycle. This is seen in the many third-party diagrams supposedly depicting "Royce's waterfall," that do not correctly correspond to the original iterative picture Royce gave.
* Few realized that, as in the words of Royce's son, "[My father] was always a proponent of iterative, incremental, evolutionary development. His paper described the waterfall as the simplest description, but that it would not work for all but the most straightforward projects."
* The single-pass waterfall had simplicity of explanation and recall ("do the requirements, then design, and then implement"); IID is more complex to understand and describe. Note that even Royce's original two-iteration waterfall immediately devolved into a single step model by other adopters and writers.
* As discussed in the opening chapter, software projects have been inappropriately associated with a predictable manufacturing paradigm (such as mobile phones) that can be predictably specified and planned, rather than a new product development paradigm.
* Single-pass waterfall has been favored by some management because it gives the illusion of an orderly, predictable, accountable, and measurable process, with simple document-driven milestones (such as "requirements complete"). There is a special irony in choosing a simple-to-track process that yields higher levels of risk.
* Waterfall values and big up-front specification goals continued to be promoted by requirements engineering (and other groups) as appropriate or even ideal decades after large project experience, research, standards bodies, and the criticism of leading experts advised against it. Perhaps this was due to unfamiliarity with the evidence or with how disciplined iterative and evolutionary requirements could be made to work.
* The Capability Maturity Model (CMM) from the Software Engineering Institute (SEI) influenced some software process engineers in the late 1980s and 1990s towards gated, document-driven, waterfall practices [[SEI03](mk:@MSITStore:C:\Documents%20and%20Settings\Administrador\Escritorio\0131111558.chm::/0131111558_biblio01.html#biblio01entry93)]. Although an IID-approach can be certified as CMM-mature, early CMM discussions had a tone of document and plan-driven, phase-oriented, and predictive planning. Many CMM certifiers and consultants had a background in waterfall values and practices and prescriptive process, without experience in iterative and adaptive methods. More recently, SEI CMM thought leaders have promoted IID and agile methods [[Paulk01](mk:@MSITStore:C:\Documents%20and%20Settings\Administrador\Escritorio\0131111558.chm::/0131111558_biblio01.html#biblio01entry82)].