Brady Field

Prototyping: alternative systems development methodology

**Important points**

“In recent years, use of prototyping has increased dramatically for both the requirements definition phase of the systems development life-cycle and rapid building of end-user systems.” [159]

“The traditional software development approach has several inherent problems, which prototyping attempts to address.” [160]

“One confusion in defining prototyping arises from the existence of two distinct types of prototyping that are used by various companies. These two basic approaches are iterative and throwaway.” [161]

“If the prototype is a throwaway type, the end system may not be exactly like the prototype.” [162]

“Because prototyping application generators are relatively easy to use and produce quick results, analysts are tempted to plunge into prototyping before sufficient analysis has taken place.” [162]

“Even though prototyping provides an excellent method of analyst/user communication, there is nothing inherent in the prototyping tools to ensure adherence to good human-factors guidelines.” [163]

“The focus of any computer session should be on the work task rather than on the system itself.” [164]

“There are many other sources for user-interface design guidelines. Incorporating these guidelines into interface designs and using prototypes to communicate user requirements will help to ensure system success.” [164]

“As prototyping software tools become more and more sophisticated, the inefficiencies will be reduced dramatically.” [165]

“Prototyping is the process of quickly building a model of the final software system, which is used primarily as a communication tool to assess and meet the information needs of the user.” [165]

“Some advantages of prototyping include: faster development time, easier end use and learning, less human-power to develop systems, decreased backlogs, and enhanced user/analyst communication.” [166]

“Some disadvantages of prototyping include: the fostering of undue expectations on the part of the user, what the user sees may not be what the user gets, and availability of application-generator software may encourage end-user computing.” [166]

“Not all systems are good candidates for the prototyping approach. Care should be taken to determine whether the system in question exhibits characteristics that make prototyping a viable option.” [166]

“No current prototyping tools ensure that good human-factors guidelines will be exhibited in the final system. Analysts should be aware of these guidelines and build systems that adhere to them, regardless of the use of prototyping tools.” [166]

“Prototyping is a powerful and widely used approach to system development.” [166]

**Disagreements**

“Adherence to a strict methodology will help to ensure the success of the prototyping approach and will combat the ‘quick and dirty’ system development that sometimes results from prototyping in a haphazard manner.” [163]

Here they are talking about using the Type I or Type II prototyping methodologies. There is no mention of alternate methodologies, and in software development we never use one method exclusively for anything. We use hybrids. The reason we do this is due to the unique nature of each project that we work on. Something should be tailored specifically to the project at hand. In this way, we can help maximize the chances that our prototypes will garner valuable information.

**Questions**

I have no questions. I understood everything in the article.