

EVOLUTIONARY DELIVERY

Evolutionary delivery is a lifecycle model that supports delivery of selected portions of the software earlier than would otherwise be possible, but it does not necessarily deliver the final software package any faster.

Main Benefits	Increases visibility of progress, supports mid-course corrections, and supports for more frequent and predictable product releases.
Keys to Success	Ensure the system's core is carefully defined, develop an architecture that supports more than one possible product direction, and schedule the delivery stages to release functionality beginning with the "most certain".
When to Use	Evolutionary delivery works best for systems where the core functionality is well-understood, but a fair amount of uncertainty exists in other areas such as the user interface, the secondary feature set, etc.
Main Risks	The main risks associated with evolutionary delivery are the risk of feature creep and diminished project control.

Overview

Evolutionary delivery is a lifecycle model that strikes a balance between staged delivery's control and prototyping's flexibility. It provides some ability to change product direction mid-course in response to customer requests. Evolutionary delivery has been used successfully on in-house business software and shrink-wrap software. Used thoughtfully, it can lead to improved product quality, reduced code size, and more even distribution of development and testing resources.

To use evolutionary delivery the core functionality of the system must be well understood. If your system is poorly understood and you expect a lot of surprises, including surprises that are likely to affect the system in fundamental ways, you will be better off using prototyping.

CxOne Support

CxOne provides support for selecting of the most appropriate lifecycle for your project through the *project planning* materials. The project plan checklist, template, and guide provide a mechanism for selecting or defining the most suitable lifecycle.

Interactions with other Best Practices

Success of evolutionary delivery depends on the effective use of designing for change. Evolutionary delivery invites changes, and your system needs to be set up to accommodate them.

Further Reading

Gilb, Tom. *Principles of Software Engineering Management*. Wokingham, England: Addison-Wesley, 1988

McConnell, Steve. *Code Complete*. Redmond, WA: Microsoft Press, 1993.

McConnell, Steve. *Rapid Development*. Redmond WA: Microsoft Press. 1996.