Software Engineers Software Engineering Engineering

What is the difference between the prototype and the spiral model in software engineering? Which is the best?

Answer Follow · 2 Request

3 Answers

Deepali K. Nagrani

Answered Dec 21, 2017

In a spiral model of development,more emphasis is laid on the risk planning.Typically involves the following in order-

1. Planning
2. Risk Analysis
3. Engineering
4. Evaluation

Here,the project phases repeat through phases- spiral in iteration.

Related Questions

What is the difference between an incremental and evolutionary model of software engineering?

What is the difference between prototype and model?

What are the different software pricing models?

What is a spiral model?

What is the difference between modelling and engineering drawing?

What is the difference between software engineering and web engineering?

What's the difference between a quantitative software engineer and a technical software engineer?

(Homework) What are the differences between evolutionary, throw-away, and rapid prototyping techniques, in software engineering?

What is really the difference between a throw-away prototype and an evolutionary prototype in software development?

1. ***Baseline spiral starts in planning phase - requirements are gathered and risk is assessed.Each subsequent spiral builds on baseline spiral .***
2. ***Angular components represent progress and radius of spiral represents costs.***
3. ***Space,research and defence projects follow this ,where large mission critical projects require critical risk analysis.***
4. Software is produced earlier .
5. This is quite costly as compared to its counterparts.
6. Doesn't suit well to the smaller projects.
7. Project success largely depends on the risk analysis phase .A prototype is produced at the end of risk analysis phase .

Prototyping Model is a systems development method (SDM) in which aprototype (an early approximation of a final system or product) is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system or product can now be developed.

1.3k Views · View Upvoters

Upvote · 4 Share

Add a comment... Recommended All

What is the difference between software developer and software engineer?

+ Ask New Question

More Related Questions

In other languages

En español: ¿Cuál es la diferencia entre el prototipo y el modelo espiral en ingeniería de software? ¿Cuál es el mejor?
En français : Quelle est la différence entre un prototype et un modèle en spirale en génie logiciel ? Quel est le meilleur ?
Auf Deutsch: Was ist der Unterschied zwischen dem Prototyp und dem Spiralmodell in der Softwareentwicklung und welches davon ist das Beste?
In italiano: Qual è la differenza tra il prototipo e il modello a spirale nell'ingegneria del software? Qual è il migliore?

Question Stats

2 Public Followers

2,105 Views

Last Asked Dec 14, 2017

Edits

Joshua Gross, Assistant Professor of Computer Science at Blackburn College

Answered Dec 15, 2017

Prototyping is a process by which you intentionally develop a working model of the software that explicitly *does not meet* requirements; this prevents you from using the prototype as a draft, rather than an experiment, which is what it should be.

The spiral model was Barry Boehm's proposal to fix the waterfall method. You go through the main tasks of development in order:

- Analysis
- Design
- Implementation
- Testing

And then you repeat, adding features. You start with high-risk and core functionality to ensure you get it right. If you have a significant problem, you fix it in the next turn of the spiral.

The spiral model is the intellectual forebear of Unified Process (UP), Rational Unified Process (RUP), and agile methods. In UP and RUP, you are doing a little bit of each activity in each work unit, but the assumption is that you're distributing the major tasks (analysis, design, etc.) through each phase

(inception, elaboration, etc.), but unevenly; there's far more analysis and virtually no testing in the first phase of a UP project, called inception, and there's still a ton of planning. In agile, your timeframe is short (often one week) and chaotic, you commit to timeframe and not feature completion, and you don't do much planning.

I never saw the spiral model used in practice, and prototyping was rarely done properly; you have to use a tool that won't work for the real thing, or else management will require you to just use/improve your prototype (it works, after all, right?). The two are orthogonal; you can do spiral, or prototyping, or neither, or both.

It's entirely possible that the spiral model is in limited, modified use somewhere; it would be good in a situation where you needed control and planning (although still little or no predictability), but want some flexibility. It appeals to formalists because it's structured; agile is seen as complete chaos, and UP/RUP is too informal and focused on "soft" issues, not just "does it work?"

803 Views · View Upvoters · Answer requested by Sara Altuwaiya

Chuck Matthews, Experience in bridging the gap between engineering and manufacturing.

Answered Feb 26

This is a question of methodology. Neither is “best,” other than that one may be more applicable than the other given the state of understanding of the requirements, what the deliverables are, and the maturity of the system the team or organization is developing.

In general, the spiral model of development focuses on staged or phased implementation of a system. With each iteration of the spiral, typically more features and enhancements are added to the system (or application) in a formal and planned way. This generally applies to a mature system that is under configuration control.

The prototype method is generally used for newer, less mature systems (or applications) where requirements are not as well known. Agile is one of the more popular (currently) methods for prototyping software applications. When the sprint or scrum is completed, the software is baselined and put under configuration control.

I’m sure you will receive other answers to this question. All points of view are likely to be valid, so don’t just accept one perspective.

878 Views · Answer requested by Sara Altuwaiya

Upvote Share ooo

Add a comment...

Recommended All

Top Stories from Your Feed