

## Bruegge 99, Analysis, Chapter 6

Bernd Bruegge

Quiz No. 6 – Answer Key

October 5<sup>th</sup>, 1999

Assuming that each of the software life cycle models listed below could be a refinement of the previous one, explain briefly the characteristics of each model and how it adds features or is different from the previous.

<b>Software Life Cycle Model</b>	<b>Characteristics, added features or differences from the previous</b>
<b>Waterfall</b>	Activity-based life cycle that consists of a sequence of the software development activities. After each activity there is a review.
<b>V-model</b>	Activity based life cycle that adds to the Waterfall model the dependency between development activities and verification activities. There is a correspondence between a development activity that focuses on the representation of the system and a verification activity that validates the system. For example, the detailed design is validated during the unit tests.
<b>Boehm's spiral model</b>	Activity based life cycle that intends to incorporate the fact that there are changes during development, by adding risk management, reuse, and prototyping to each activity. It is called spiral because it consists of several rounds that always end with a review and add to the previous round.
<b>Sawtooth model</b>	Activity based life cycle that intends to integrate the user and the software developer's perceptions of the system, by adding prototypes and client demonstrations at certain stages of the software development life cycle.
<b>Shark tooth model</b>	Activity based life cycle that adds to the Sawtooth model, some internal reviews and demonstrations for project managers.
<b>Unified software development process</b>	Entity based life cycle composed of models of the system. It is similar to the Spiral model because it is based on the fact that a project consists of several cycles, that all end with a client deliverable, and each cycle has four phases, which are Inception, Elaboration, Construction and Transition.
<b>Issue-based life cycle model</b>	Entity based life cycle that aims at dealing with frequent changes. Each project starts with a set of issues that are stored in a database and mapped to activities of the software development life cycle. The status of each issue is used to track the status of each activity.