Jana Nash-Siegle
Networks and Server Structures
John Dalton
150901

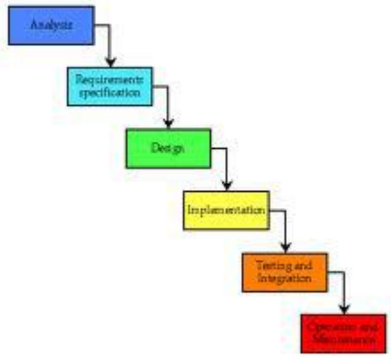
SDLC Comparison

In this paper we will compare and contrast two different processes of SDLC, aka Software/System Development Life Cycles. These two processes are known as the Waterfall and the Spiral. During my research of these two cycles, I ran across a paper that, in my opinion makes the most sense. This paper states within it that one should never subscribe to just one process for every job, but that teams should pick and choose that which works best for them. Perhaps that would be a bit of each process, or perhaps the process would change depending on the job. But the main idea I believe behind this theory is that the team should always remain flexible. Two quotes stood out to me and they are as follows, quoted from an article entitled "Enough of Processes – Lets Do Practices", by Ivar Jacobson, Pan Wei Ng and Ian Spence:

> "…it doesn't matter which process you adopt as long as it is adaptable, extensible, and capable of absorbing good ideas, even if they arise from other processes.
> To achieve this kind of flexibility things need to change. The focus needs to shift from the definition of complete processes to the capture of reusable practices. Teams should be able to mix-and-match practices and ideas from many different sources to create effective ways of working, ones that suit them and address their risks."

Waterfall Model

We will begin with the Waterfall Model, which is a very popular version of SDLC and considered the Cadillac classic approach to the development life cycle.



The Waterfall Model is both linear and sequential. It consists of 6 main phases these being: Analysis, Requirements/Specifications, Design, Implementation, Testing/Integration and Operation/Maintenance. Some teams may have this broke down into 5 stages or with vaguely different names, but the overall processes tend to achieve the same goal throughout. Waterfall Development are that it does allow for department and manager control simply due to the fact that because of its linear nature, one stage must be completed before it can move on to the next. This model is best used for small projects with very certain requirements. Reviews of each phase occur before the next stage is moved into. Phases/stages will never overlap and testing will only start once development is complete.

Advantages of the Waterfall Model:

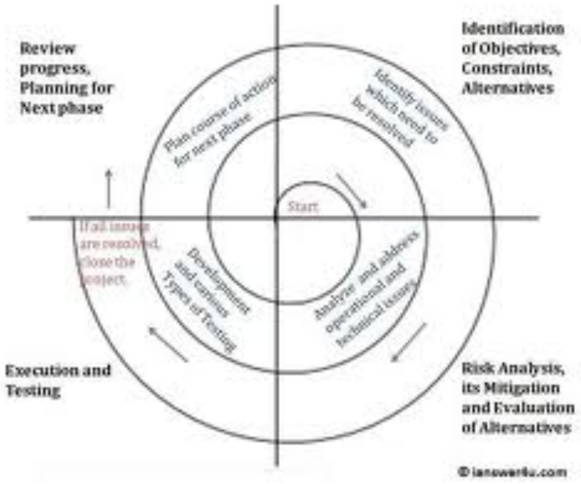
- Simple to use and easy to understand.
- Ease management, specific deliverables per phase and review process.
- No overlapping phases.

Disadvantages of the Waterfall Model:

- Once testing stage is reached, backward movement is difficult to accomplish.
- No working software until late in the cycle.
- High amounts of risk and uncertainty.
- Not appropriate for complex and object oriented projects.
- Not for long and ongoing projects
- Not suitable where requirements maintain a high risk of uncertainty and changeability.

Spiral Model

Barry Boehm first wrote about this model when he wrote “A Spiral Model of Software Development and Enhancement” in 1988. This model is characterized as a “risk-driven” process model. This term of risk-driven means that choices are made in regards to the development based upon the project’s risk. The four stages of this model are the Review/Planning, Modeling, Construction and Deployment.


Review progress, Planning for Next phase
Identification of Objectives, Constraints, Alternatives
Plan course of action for next phase
Identify issues which need to be resolved
Start
If all issues are resolved, close the project.
Development and various Types of Testing
Analyze and address operational and technical issues
Execution and Testing
Risk Analysis, its Mitigation and Evaluation of Alternatives
© ianswer4u.com

A little more definition of each stage gives us:

- Review/Planning
    - Estimating
    - Scheduling
    - Risk Analysis
- Modeling
    - Analysis
    - Design
- Construction
    - Code
    - Test
- Deployment
    - Delivery
    - Support
    - Feedback

Just as with the Spiral Model, these are flexible and depending on the team using them the definitions could vary slightly in detail. In this model the phases or stages follow an iterative order, one right after another. By doing this iteration analysis of problems associated with each phase are dealt with when the phase comes around with the next iteration.

Advantages of the Spiral Model:

- Due to high level of risk analysis, risk avoidance is enhanced.
- Works well for large and mission-critical projects.
- Strong control via document and approval.
- Flexible enough to allow more functionality to be added at later times.
- Production begins earlier in the life cycle.

Disadvantages of the Spiral Model:

- Costly to incorporate.
- Risk analysis requires very specific expertise.

- Success rate is very dependent upon the risk analysis phase.
- Definitely not compatible with smaller projects.

References:

What is Waterfall model- advantages, disadvantages and when to use it? (n.d.). In *ISTBQ Exam Certification*. Retrieved September 12, 2015, from Google.

Jacobson, I., Wei Ng, P., & Spence, I. (2007). "Enough of Processes—Lets Do Practices. N.p.: Journal of Object Technology.

Various Models Of SDLC (n.d.). In *Software Testing for Beginners*. Retrieved September 12, 2015, from Google.

Gordiyenko, S. (2014, November 3). Software Development Life Cycle (SDLC). Waterfall Model. In *XB Software*. Retrieved September 12, 2015, from Google.