Managing Software Development Projects,
The Sequence of the Project Phases

Felician ALECU, PhD, University Lecturer
Department of Economic Informatics
Academy of Economic Studies, Bucharest, Romania
E-mail: alecu[at]ase[dot]ro; Web Page: http://alecu.ase.ro

Abstract: Software development projects are logically divided into phases that are composing the project life cycle. Typically, the phases are scheduled sequentially but in some cases a project may take clear advantages by running the phases concurrently. The most common phase-to-phase relationships are sequential, overlapping, iterative and composite.

Keywords: project life cycle, project phases, project management like a game, sequence of phases.

The Sequence of Project Phases

Since a project is a temporary endeavor undertaken to create a unique product, service or result (PMBOK Guide), it is quite obvious most projects have their own life cycle, a progression (generally sequential) of phases that are industry dependent, so projects from various fields will implement different stages. In such a way, any project is usually divided into smaller logical parts/subsets that can be better managed, planned and controlled.
The life cycle is typically applied together with a project management methodology/process which includes, from a PMI’s point of view, the five Project Management Process Groups – Initiating, Planning, Executing, Monitoring and Controlling, Closing, as detailed into Figure 1. The Project Management Process Groups are industry independent and totally different than the project phases.

![PMI’s Project Management Process Groups](image)

Figure 1 – PMI’s Project Management Process Groups

Small projects usually have a single set of project management process groups covering the entire life cycle, as illustrated in Figure 2.
On the other hand, large projects are divided into phases, each phase having its own set of project management process groups (Figure 3).
Usually, the project phases are scheduled sequentially but in some cases a project may take clear advantages by running them in a concurrently manner. The most common phase-to-phase relationships are detailed below:

- **sequential** - one phase can start only after the previous one is finished, so it is clearly a step by step approach that doesn’t allow fast tracking, even if the level of improbability is very low (Figure 4). Each phase can be executed totally independent by the others since there are no overlapping areas. The waterfall model is still the most widely used sequential approach in software development where the problem to be solved is divided into phases that become milestones.

![Sequential Phases](image)

**Figure 4 – Sequential Phases**

- **overlapping** - that one phase can start even if the previous one is not completed yet, by assuming all the corresponding risks in order to speed up the project schedule. Usually, the overlapping occurs only at the edge of successive phases (Figure 5).

![Overlapping Phases](image)

**Figure 5 – Overlapping Phases**
• **iterative** – used where a high degree of uncertainty is encountered, like the research projects. The work for the current stage is performed in the same time as planning for the next phase (Figure 6), so it is not possible to provide long term planning. While the work progresses, new gathered data help to update the project schedule. In software development area, the iterative model is sometimes seen as a sequence of waterfalls with higher costs but reduced risks.

• **composite** - can be found in multiphase projects that are using several types of relationships during the life cycle. Figure 7 illustrates the case of a project that is implementing both sequential and overlapping relationship types during its life cycle. It is possible to use all three types of relationships during the phases of a single project.
Figure 6 – Iterative Phases

Figure 7 - Composite Relationship (Sequential and Overlapping)
Conclusions

How can we better exemplify these phase-to-phase models? An interesting approach could be the analogy with the sports since the software development itself can be seen as a game.

- The sequential approach can be met at relay races (athletics, swimming, skating – long or short distance).
- The overlapping model is characterizing the team sports in which any competitor is assigned to a dedicated field area and these areas are intersecting is some ways. Examples of such sports are team tennis, football, volleyball and so on.
- The best example for the iterative approach is rugby, a sport in which the entire team is progressing with the ball, so each team member is contributing to the game from the start until the end.

References


