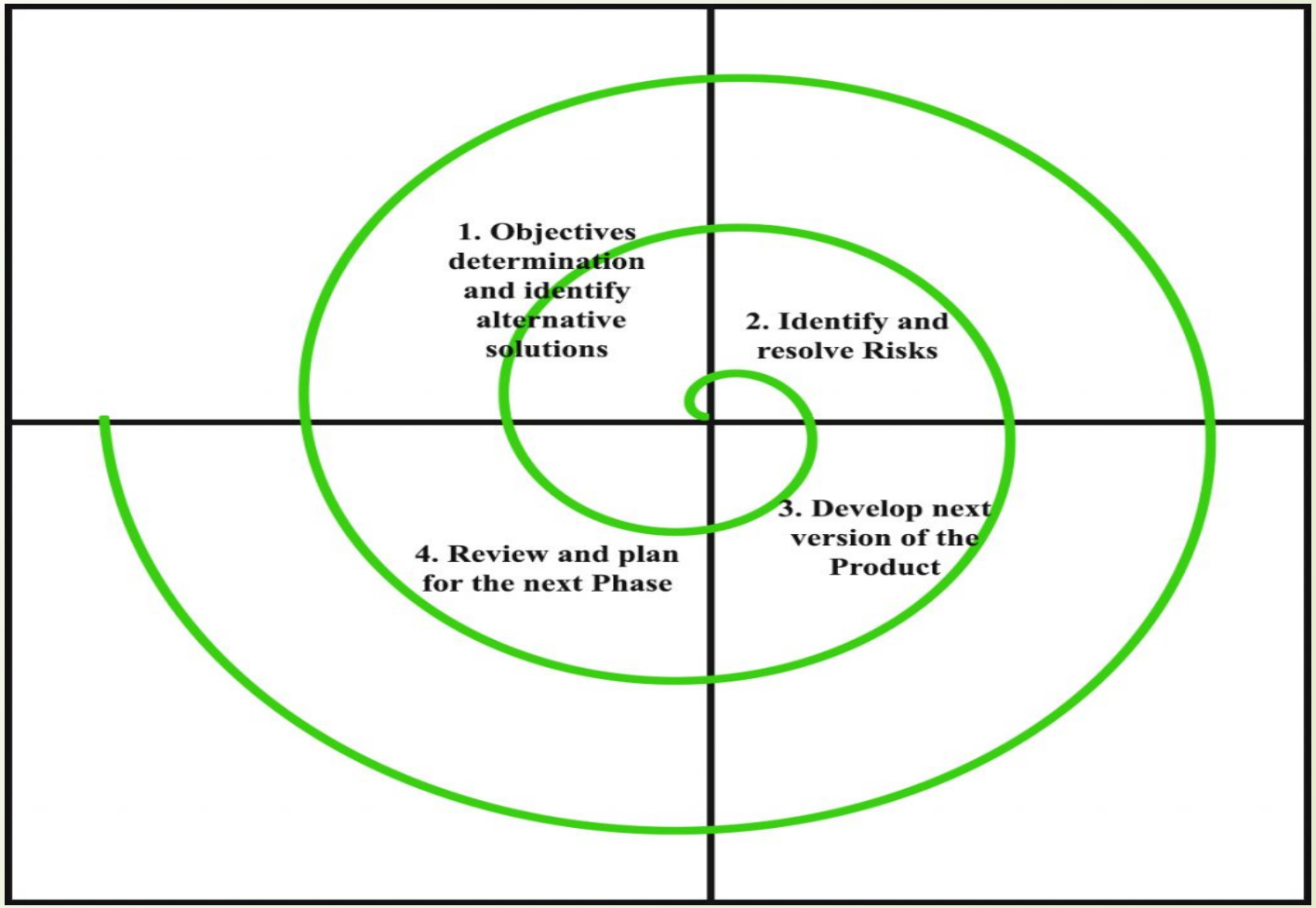
Software Process Model

Spiral Model

- **Spiral model** is one of the most important Software Development Life Cycle models, which provides support for **Risk Handling**.
- Each loop of the spiral is called a Phase of the software development process**.**
- The exact number of phases needed to develop the product can be varied by the project manager depending upon the project risks.
- The Radius of the spiral at any point represents the expenses(cost) of the project so far, and the angular dimension represents the progress made so far in the current phase.

Spiral Model



Four Phases of Spiral Model

- **Objectives and Constraints**: Define what exactly would be the objective for this cycle. And then you have to talk to all of your stakeholders so that it meets the need of all the stakeholders. And, at the same time, you define what are the constraints to achieve those objectives in that cycle. And find alternative solutions.

- **Identifying and resolving risk**: It is not just only about identify the risk, but then find ways to mitigate or do something about those risks. So sometimes you might be building a prototype, or you might be doing some surveys or whatever you need to do, or some research to identify what other things can go wrong and then you try to resolve what you can.

Four Phases of Spiral Model

- **Development and Testing** : If you are doing feasibility study then you do your feasibility study, or if you're doing requirements, then you write those and so on. And if you're doing actual development then you go through the design code integration and the result. So whatever is the objective of that cycle, the work needed to be done to achieve that objective is done here.

- **Review and Plan for Next**: The last is to plan the next iteration. So this is the step where you review the work done in this cycle and then, you say, should we continue or should we not continue. And If you will continue then you will decide the goal for next iteration.

Advantages

- The focus on the risk increases the chance of success of your product.
- Good for large products.
- Customer Satisfaction.

Disadvantages

- Complex
- Expensive
- More focus on risk management
- Time management
- Need to involve stakeholder in each iteration.

When to use?

Very large and high risk projects.

Scenarios

Scenario 3

- Defense organization of a country did a recent study, and the research recommends new capability the country should build to keep the country protected from potential conflicts in the region. So there's a study and the solution, the system that they came up with require to be build has never been attempted, and no literature exist for such system. So it's a very fairly new area, or fairly new adventure or things that they need to create. And it's fairly big and complex system, and potentially can take decades to build. A lot of loss of time, a lot of years to build this software. And scientists have vague idea about how to go about it, but no concrete plan exists. And there'll be a lot of organization stakeholders because this of course defense organization of the country and a lot of stakeholders involved. So a lot of constraints will also be there that will impact on this initiative. So there's a lot of risk, a lot of constraints. So let's try to analyze this situation and see which model will work in this situation.

Solution?

- Unknown need and outcomes.
- Very Risky.
- Very Large and Complex Project.

Which Model/s is/are suitable?