## Introduction

*Background:*

In this book, the main aim is to write the application with easy understandable, easy maintainable, portability, scalability, high efficiency, and do any expected behaviors.

*The main strategies can be divided into two types: General Design Strategy and Specific Language Feature with Specific Details.*

The former strategy is mainly focused on discussing Design which is mainly focused on “How to choose the Design to finish among two different behaviors”.

Such as *“Choose Inheritance or Templates”*, *“Public Inheritance or Private Inheritance”*, *“Private Inheritance or Composition”*, *“Member function or Non - Member function”*, *“Pass - By - Value or Pass - By - Reference”*.

It’s important to make the right decision on these points, since one bad decision may not bring any bad result, but would show some bad results, then at that time to correct, it would be difficult and time - consuming, and the cost would be very high.

Even you totally know what to do, then it may still be difficult to go to the right norm. *What is the right return type of assignment Operator ? When should make Destruction Function Virtual ? When operator new can not find enough main memory, then what should be the next step ?*

Software Design and Realization is complex, which has been constrained by Hardware, Operating System, Constraint Condition of Application.

*Terminology:*

In this Chapter, would introduce “Terminology” that each Programmer should get knowledge with.

1. Declaration - The so - called Declaration is to tell the Compiler of the name and type of somethings, but to neglect all Details.
2. Definition -
3. Initialization -
4. Copy Constructor -
5. Standard Template Library -
6. Undefined Behavior -
7. Interface -
8. Client -

*Naming Conventions:*

*Threading Consideration:*

*TR1 and Boost:*