CODE LAYOUT, REABILITY, REUSABILITY

Date	18 October 2022
Team ID	PNT2022TMID43265
Project Name	Efficient water quality analysis & predicition using machine learning
Maximum Marks	8 Marks

Introduction

In software programming, reusable code refers to the use of the same source code At its best, code reuse is accomplished by sharing common classes or collections of functions and procedures (this is possible in C++, but not in Smalltalk or Java). At its worst, code reuse is accomplished by copying and then modifying existing code. There are multiple ways to write reusable code. In this article we are going to review the most common reusable code patterns .Reusable code in Object Oriented Programming.

Back in the origins of object-oriented programming, object classes were thought to be reusable for any future project under any circumstance. This, however, hasn’t been the case as code reusability is a difficult technique that is not suitable for all programming Object Oriented Programming is based on objects. Objects are represented by interfaces which specify the properties of the objects. The goal to code reusability in Object-Oriented Programming is to design objects in a way that.

The truth about code reusability is that software projects frequently overrun their budgets and software is developed behind the planned schedule. This pressure on development can lead to code that is not clean becoming less reusable.

Reusable components

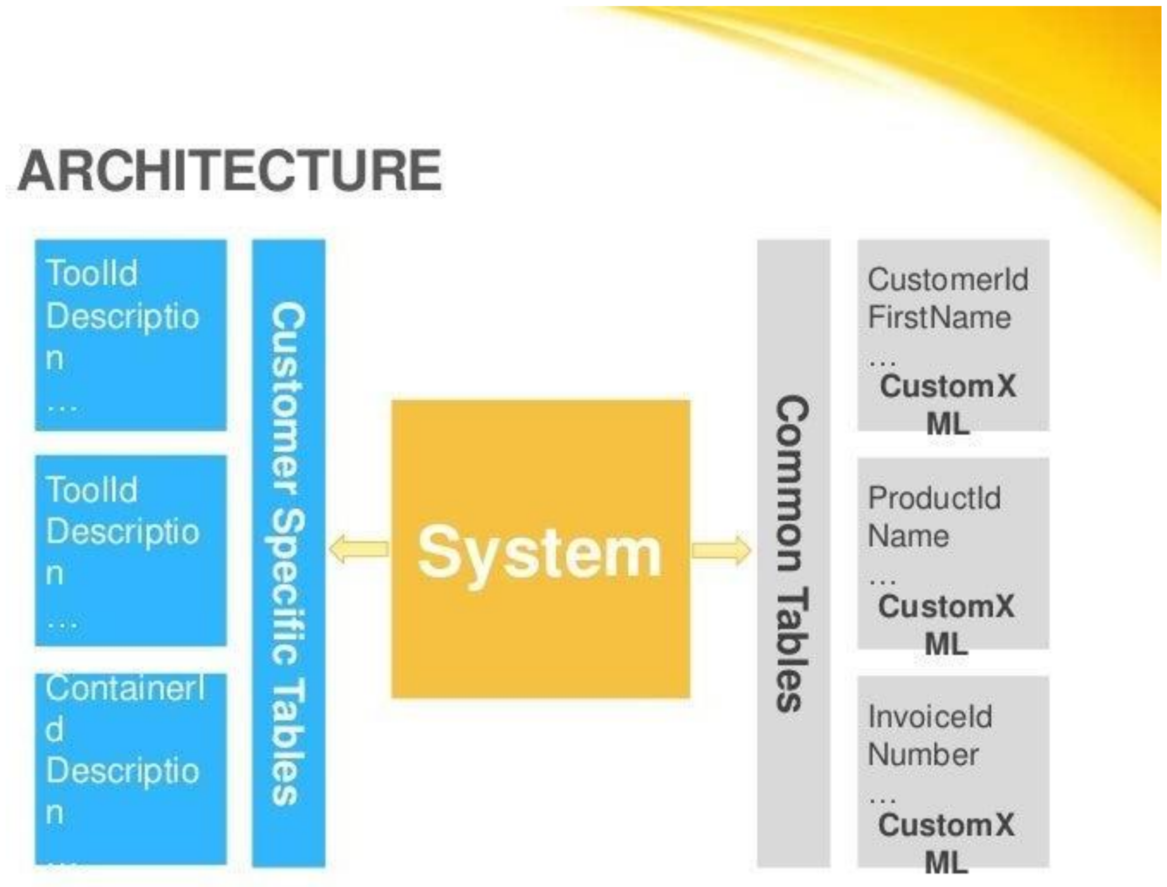
The evolution to object reusability is the concept of component reusability. A component is a unit that can be reused or replaced. The ideal software applications are built using connections of components.

According to the Catalysis Approach, a component can be reused when:

- High cohesion. Meaning the internal methods and properties of a component should have a strong relation and not have logic that is independent.

- Low coupling with the rest of the system. A component should not care about the environment where it is being used.

- A well-defined interface. This allows for external services to know how to interact with the component.

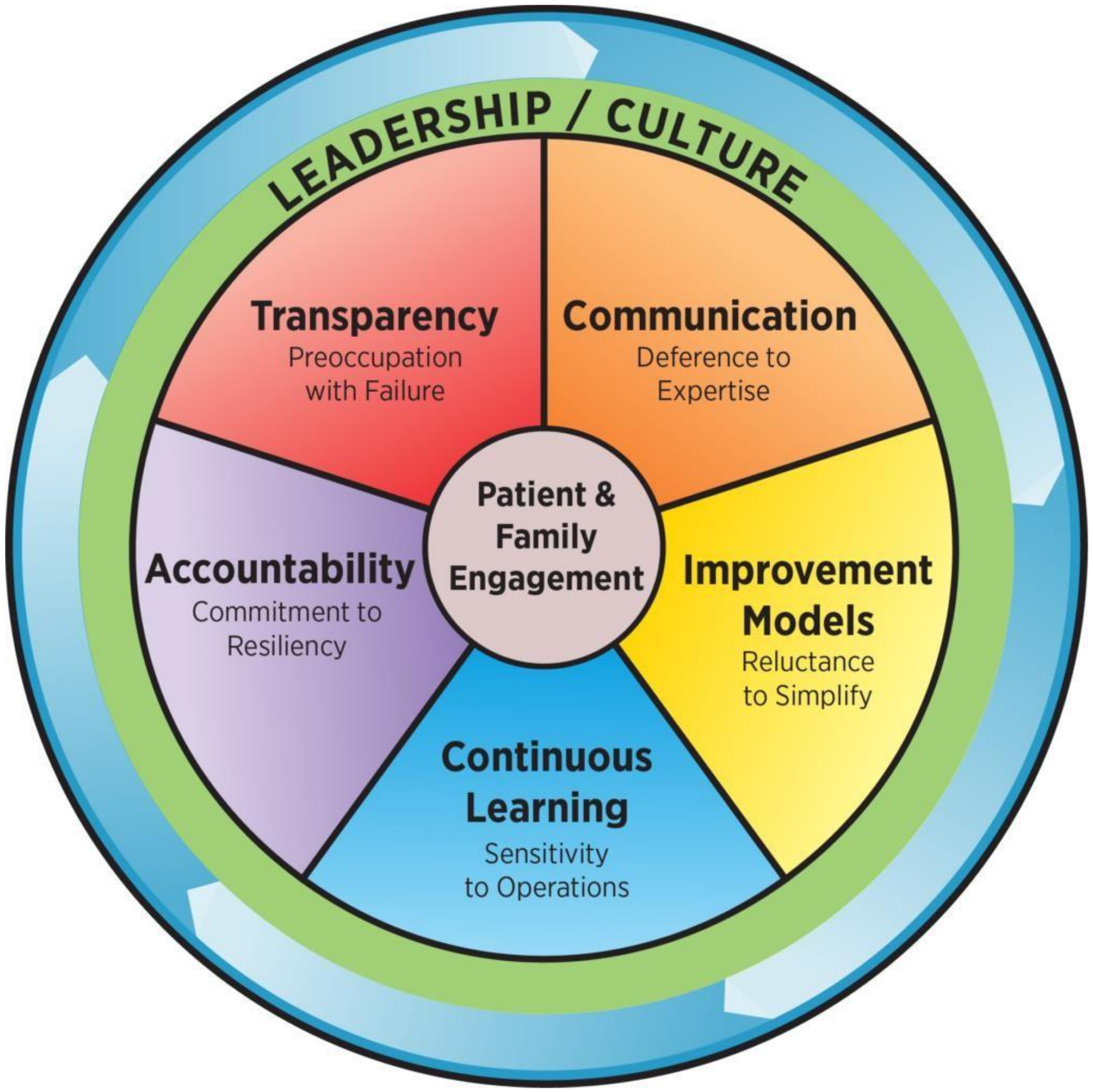


Reusable Knowledge in Software Programming

While doing my research for this article I came across the "Reusable Knowledge" term. This has changed my way of thinking about code reuse .Due to difficulties attached to code reusability the "reuse of knowledge" concept has been researched. Instead of reusing the lines of code what it is being reused is the knowledge on how to build a system taking into account past experiences.

An ongoing research is the Know Bench architecture. The concept lies on capturing knowledge from building a software system and applying it on future developments. Documentation is key to this type of reusability. This research uses Semantic Web technologies to generate annotations that capture knowledge and experiences that can prevent the repetition of past failures in future software development projects .Another example of knowledge reuse in software systems is the idea of applying program transformation to a set of knowledge concepts and transforming it into software.

The code reusability journey is far from over. It will be an exciting topic to following the upcoming years as new and powerful technologies are constantly raising.Back in the origins of object-oriented programming, object classes were thought to be reusable for any future project under any circumstance. This,



Raw score = 0.1579*(PDW) + 0.0496*(ASL) + 3.6365

Here,

PDW = Percentage of difficult words not on the Dale–Chall word list.

ASL = Average sentence length

The Gunning fog Formula

Grade level= 0.4 * ((average sentence length) + (percentage of Hard Words))

Here, Hard Words = words with more than two syllables.

Smog Formula

SMOG grading = 3 + √(polysyllable count).

Here, polysyllable count = number of words of more than two syllables in a

sample of 30 sentences.

Flesch Formula

Reading Ease score = 206.835 - (1.015 × ASL) - (84.6 × ASW)

Here,

ASL = average sentence length (number of words divided by number of sentences)

ASW = average word length in syllables (number of syllables divided by number of words)

Advantages

:

1. Readability formulas measure the grade-level readers must have to be to read a given text. Thus provides the writer of the text with much-needed information to reach his target audience.
2. Know beforehand if the target audience can understand your content.
3. Easy-to-use.
4. A readable text attracts more audience.

Disadvantages

1. Due to many readability formulas, there is an increasing chance of getting wide variations in results of a same text.
2. Applies Mathematics to Literature which isn't always a good idea.
3. Cannot measure the complexity of a word or phrase to pinpoint where you need to correct it.

Conclusion

Object Oriented Programming is based on objects. Objects are represented by

interfaces which specify the properties of the objects. The goal to code reusability in Object-Oriented Programming is to design objects in a way that can later on b

The truth about code reusability is that software projects frequently overrun their budgets and software is developed behind the planned schedule. This pressure on development can lead to code that is not clear.