**Polymorphism in Everyday Life**

Object Oriented Design, sometimes referred to as Object Oriented Programming, is a concept based around creating a system of interacting objects for the purpose of solving a problem. An "object," in this context, refers to a physical or digital item with its own unique properties and behavior.

OOD originated in the field of software development and has been used there for over 30 years. However, the principles of OOD can be applied to a wide variety of situations both in the workplace and at home.

The Pillars of OOD
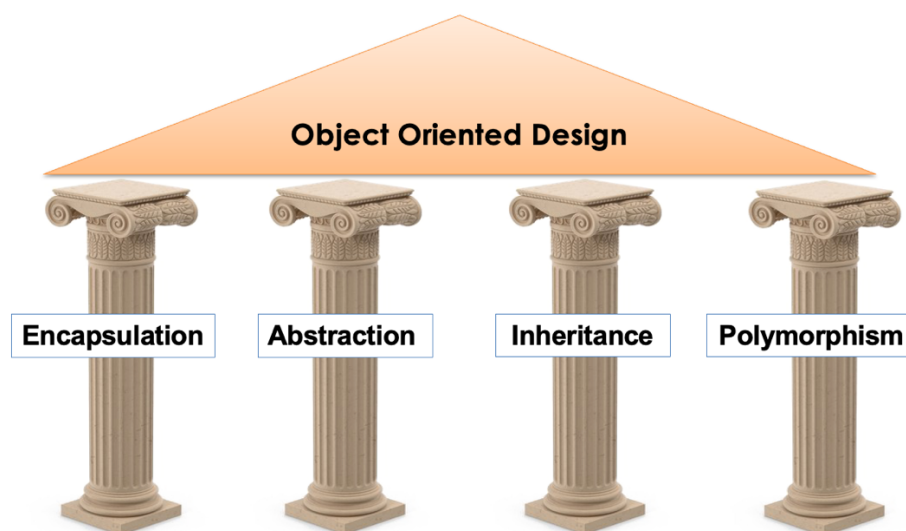
OOD is supported by four main concepts:
- Encapsulation: Bundling an object's properties and functions into one unit.
- Abstraction: Hiding complexities and showing users only what is necessary.
- Inheritance: Categorizes objects into "classes" based on shared and unique properties.
- Polymorphism: Accounts for different interpretations and responses to the same command.

In this post, we will focus on polymorphism.

Polymorphism in the Real World

Part of a Scrum Master's task is facilitating Retrospectives at the end of each Sprint. Traditionally, Retrospectives begin with the Scrum Master asking the Team Members an icebreaker question. Recently, during a Retrospective I was facilitating, I asked the Team Members what their first job had been. One person had been a waiter, another confessed that this was their first job, and another took about 10 minutes on a related topic, but then finally

stated that they could not remember their first job. They had all been asked the same question, and yet their responses were all different.

Discussions like this are a perfect example of **polymorphism.** This term, which literally translates to "many forms," describes the ability of different objects to react and respond to the same message in their own way given their purpose or mission.

In software development, polymorphism allows objects to have the same flexibility of communication that humans use daily, permitting the addition of future objects without changing the current messages. Because of differences in culture, language, and life experiences, people interpret questions, commands, and situations in many diverse ways.

Some everyday examples of polymorphism include:

- To a parent, "It's time for dinner" may mean "Come downstairs and wash your hands." A child, however, may interpret this as "It's time to eat immediately."
- In the United States, saying "I will try" is generally a positive response to a request. In Japan, "I will try" is often meant as a polite "no."
- As a response to a request for a meeting with their manager, some people would view it as an opportunity to express how great they are, while others might interpret it as a sign they may be getting fired.

As Agile Coaches, we work with clients who come from a wide variety of industries, professional backgrounds, and even different countries. As we work with someone, we begin to observe their own interpretations of questions and requests.

Call to Action

One of our most important tasks is to be aware that even the most innocent or simple request may result in undesirable behavioral side effects. As we start to engage with Team Members and Stakeholders, perhaps it is best that even the simplest requests are in the context of a dialogue, at least until we have an opportunity to understand their way of thinking and communicating.

Keeping in mind the principle of polymorphism, we can overcome communication barriers, build stronger Teams, and forge a stronger relationship with our clients.