UML Modeling

2018-03-02

This explanation of the concepts is based on how they are explained by Sommerville (2010).

1) **Generalization** is a technique for managing complexity by placing entities in more general classes such that the general classes hold the common characteristics of its members.
2) **Aggregation** is an association between parts of an object that is composed of many different parts.
3) **Activity diagram** is used to model processing of data. An activity diagram consists of activities of the system process and the flow of control between these activities. Each activity represents one process step.
4) **State diagram** is a model used for event-driven modeling. A state diagram shows system states and the events that cause transitions from one state to another without showing the flow of data within the system. The state diagram may include additional information on the computations carried out in each state.
5) **Design pattern** is a description of how to solve a problem in such a way that that the solution can be reused in many different situations. A design pattern is not a detailed specification and cannot be directly transformed into a machine code (Wikipedia 2018). To quote Sommerville (2010), design patterns can be thought as a *"description of accumulated wisdom and experience, a well-tried solution to a common problem"*.
6) In **Host-target development** software is developed on the *host* computer and deployed to another computer referred as the *target*.

Bibliography

Sommerville, I., 2010. *Software engineering*. 9th ed. USA: Addison-Wesley Publishing Company.

Wikipedia, 2018. *Software design pattern - wikipedia* [online]. Available from: https://en.wikipedia.org/wiki/Software_design_pattern [Accessed 2 Mar 2018].