Builder

The Builder pattern creates elements and a diagram that separates the construction of a complex object from its representation so that the same construction process can create different representations.

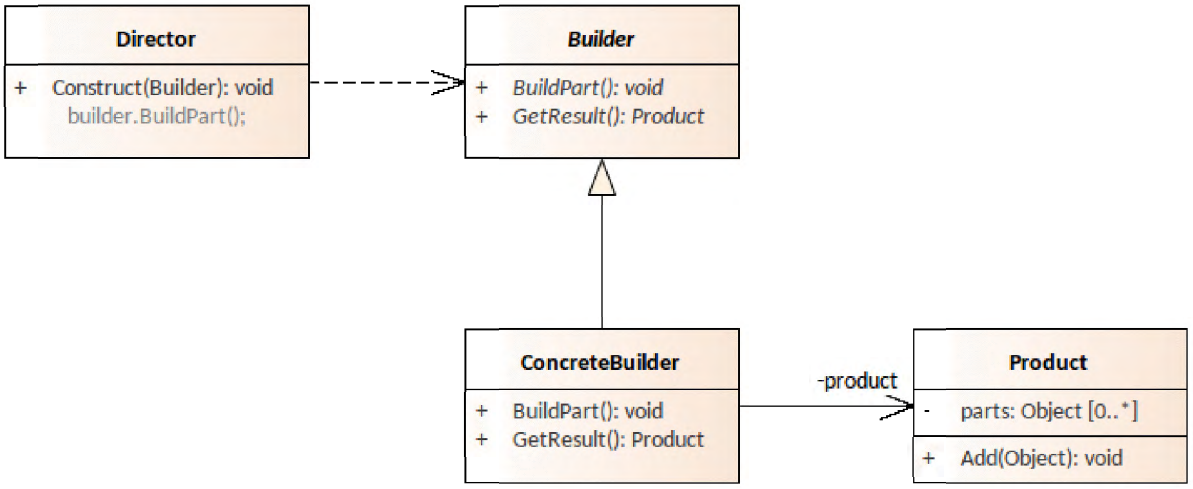

Figure 1. Shows a Class diagram that contains a number of Classes including a Director, Product, Builder and a Concrete Builder that specializes the Builder Class.

Discussion

The purpose of the pattern is to solve a problem with Object construction when multiple constructors are required. Using the Builder pattern an object delegates the construction of an object to the Builder which is sent a series of initialization parameters one at a time and then returns the resulting constructed object when it has been built.

The pattern can be used to:

- Allow a class to create different representations of a complex object?
- Simplify the creation of a complex object.

The following is a list of some things you may want to do when working with this pattern.

- Change the name of the Package and the diagram to suit the initiative.
- Change the name of the Classes to suit the initiative.
- Add Attributes and Operations to the Classes if needed.
- Change the names of the Roles and the Cardinalities described by the multiplicities.
- Create additional Classes to embellish the model where needed.

The following is a list of some of the next steps available when applying the pattern.

- Generate Source (Programming) Code directly by first selecting a target language.
- Synchronize the generated Source Code with the model.
- Edit the Source Code directly inside the model using the fully featured Code Editor.
- Add descriptions to the Classes to describe their role in the system description.
- Add one or more State machines to describe the discreet states a particular Class could exhibit.
- Automatically generate documentation with the Document Generator using built-in or user-defined templates.

Useful Workspace Layouts **Core | Core Modeling, Wide View**

Reference

The following help topics will assist you learn about how to work with this pattern.

GoF Patterns

Generate Source Code

Editing Source Code

Class

Class Diagram

Associations

Generalization

Composition

Source and Target Roles

Working with Diagrams

Changing Element Appearance

Changing Diagram Layout

The following are some of the tools that will be helpful when working with this pattern.

Source Code Generator

The Source Code Generator is an integrated set of tools that allows source code to be automatically generated from a range of UML diagrams including Class diagrams. Behavioral models including State Machine, Activity and Sequence diagrams can be used to generate code. The models can be generated to a wide range of programming languages and built in or user defined templates can be used to tailor the generated source code to meet development and corporate standards and coding guidelines. For more details see the Generate Source Code help topic.

Source Code Editor

The Source Code Editor is a fully featured programming source code editor. It has a structure tree for easy navigation of attributes, properties and methods. Line numbers can be displayed and syntax highlight options can be configured. Many of the features that software engineers are familiar with in their favorite IDE, such as Intelli-sense and code completion are included in the editor. Viewing the source code juxtaposed with the Models from which it is generated brings a great clarity to the design effort and its implementation. For more details see the Editing Source Code help topic.

Reviews

The Review facility is a collaborative and team tool that allows one or more models elements to be reviewed and the comments made visible directly in the model. The reviews can be conducted formally or informally and the review process can be augmented by further Discussions and Chat available from the same location in the User Interface. For more details see the Formal Review Elements help topic.

Element Discussions

The Element Discussion facility is a fully featured collaboration tool allowing modelers and model viewers and reviewers to communicate with each other directly inside the repository. Modelers using the full client or occasional viewers using WebEA can both post and reply to discussions and communicate and engage in chat. For more details see the Element Discussions help topic.

Document Generator

The Document Generator is a powerful facility in Enterprise Architect that allows a Database Engineer or other stakeholder to create high quality corporate or technical documentation directly from the model, suitable for internal or external audiences. For more details see the Documentation help topic or the more general topic on Model Publishing.

Diagram Layout

The Diagram Layout tool allows you to layout an entire diagram, selected elements or sections of a diagram to make it more visually appealing or meaningful to a particular audience. There are a wide range of layout types to choose from and some types have filters that can be applied. For more details see the Diagram Layout help topic.

Pan and Zoom

The Pan and Zoom facility is one of the tools that can be used to navigate around a large diagram. Often the resolution of a diagram must be reduced to ensure it is wholly visible but by using the Pan and Zoom window you can leave the diagram at a readable resolution and pan around to areas of interest zooming in when necessary. For more details see the Pan and Zoom help topic.

Diagram Legends

The Diagram Legend facility is useful for manually or automatically changing the appearance of elements and connectors on a diagram. A legend can be added from the Common Toolbox and configured to codify the fill and line color and line thickness. This is a powerful way to add meaning and expression to a diagram and is particularly expressive when applied automatically based on element or connector properties. It can be used with a number of specialized diagrams such as roadmaps to create a powerful visualization. For more details see the Diagram Legends help topic.

Relationship Matrix

The Relationship Matrix provides a spreadsheet like view of two groups of elements and the relationships that exist between them. It can be a used as a powerful analysis mechanism to visually indicate how elements are related to each other and to discover

which elements are missing relationships. For more details see the Relationship Matrix help topic.

Specification View

The Specification View can be used as a way of working with the Components and Interfaces particularly when there are a large number of elements as is typically the case when describing a system of any appreciable size. For more details see the Specification View help topic

Traceability Window

The Traceability Window automatically displays the relationships that exist between Use Cases and other model elements including up-process and down-process elements. The traceability tree view can be conveniently expanded to see deeper relationships and elements displayed in the window can be located in all diagrams in which they appear. For more details see the Traceability Window help topic.

Hand Drawn and Whiteboard Diagrams

The Hand Drawn and Whiteboard Mode are display options available for any diagram that changes a system-drawn diagram to appear as though it was drawn by hand and, optionally, hand drawn on a whiteboard. It is a powerful device to engage an audience by presenting the diagram in a rough and more immediate style giving the impression that it is just a sketch that can be changed. For more details see the Hand Drawn and Whiteboard Mode help topic.

Alternate and Images for Diagram Elements

Most standard elements allow an alternate image to be defined for an element that will be used in place of the graphical notation for the element either on a selected diagram or as a default on all diagrams. For more details see the Using the Image Manager help topic.