

Describing the capabilities and cases of uses for sequence diagrams in software engineering *

Shevcheko Serhii

Slovenská technická univerzita v Bratislave
Fakulta informatiky a informačných technológií
`xshevchenko@stuba.sk`

5. november 2021

Abstract

The article will describe the general principles of constructing sequence diagrams, and provide optimal cases of their use in the field of software engineering modeling.

The main structural elements of a sequence diagram will be described, the rules for their use with corresponding examples.

1 Introduction

In the well-known graphical description language for modeling in the field of software development, UML, there are several types of diagrams, each of which has its own advantages and disadvantages.

Among the popular types of diagrams are class diagrams, component diagrams, and activity diagrams.

The problem with all the above diagrams is that they cannot simulate the dynamics of the model, the behavior and the relationship between "actors" over time. They represent the structure, but not the behavior of the model on the timeline.

To solve this problem, a sequence diagram was created. It, unlike other types of diagrams, shows the interactions of "actors" on the timeline, the exchange of "messages" (data) between them. And, again, this is all shown in projection on the time axis. That is, it allows to explore the model in dynamics [5].

2 What is UML?

Before we talk about sequence diagrams, we should first understand what UML is, of which the sequence diagram is a part.

The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system [2].

*Semestrálny projekt v predmete Metódy inžinierskej práce, ak. rok 2021/22, vedenie: Shevchenko Serhii

In other words, we can say that UML is a set of rules and recommendations on how to create visual structural and relational models of software systems.

The creation of UML was originally motivated by the desire to standardize the disparate notational systems and approaches to software design. It was developed at Rational Software in 1994–1995, with further development led by them through 1996 [3].

2.1 What UML include?

The fact is UML consist of many types of diagrams 1. In order to understand the benefits of a sequence diagram, it is important first to understand what other diagrams exist in UML and for what purposes they are used.

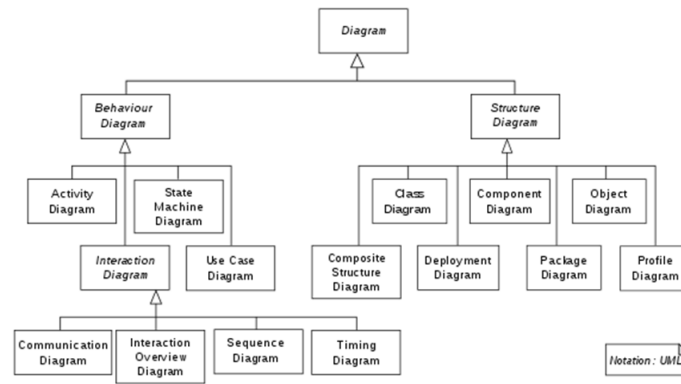


Figure 1: Diagram types [3].

Looking at Figure 1, we see that the structural tree of UML diagrams is rather voluminous. Not all of these diagrams are actively used in the development industry [2], but each was created for a specific purpose. In Table 1, we can see the 4 main types of diagrams actively used in the development industry, and a short summary of their purpose and structural composition.

	Uses for representation	Consist of
Sequence diagram	relations of the object in time sequence	objects and messages on lifeline.
Class diagram	static classes structure in a model and relations between them.	classes, that contain attributes and methods.
Activity diagram	of workflows of stepwise activities and actions.	activities and logic blocks.
Component diagram	how components are wired together to form larger components or software systems.	structural components.

3 When to use a sequence diagram

A sequence diagram should be used primarily to visualize relationships between objects, taking into account the sequence of these very relationships [1].

This diagram is very useful for modeling synchronous services, it allows you to think through all the interactions between "actors" from the beginning to the end of the service life cycle on the timeline.

4 The notation

4.1 Frame

The first notation element to know is the frame. It makes a graphical boundary of a sequence diagram and provide place for a diagram label (Frame 2).

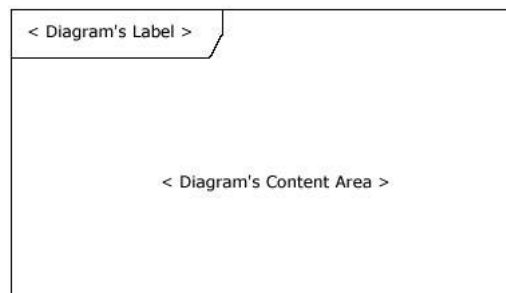


Figure 2: Frame template.

This element is not required in sequence diagrams, but it is better to use it for convenience. When using this element, it is necessary to place all subsequent elements, which will be described below, inside the frame.

4.2 Lifelines

Lifelines are designations of objects that make up the model, and between which information is exchanged. This is one of the key elements of a sequence diagram.

Lifelines are drawn as a box with a dashed line descending from the center of the bottom edge (Figure 3) [1].

4.3 Messages

Messages are illustrating calling a function from one object to another, data exchange between the objects. There are dispatch messages, that display calling informations, and return message, that display end of execution [4].

White rectangle is an Execution Specification, it is used to visually display the start and end of an operation.



Figure 3: Lifeline template.

5 Building a Sequence Diagram

6 Conclusion

References

- [1] Donald Bell. Explore the uml sequence diagram. <https://developer.ibm.com/articles/the-sequence-diagram/>.
- [2] Martin Fowler. *UML Distilled*. Addison-Wesley, 2003.
- [3] Ivar Jacobson Grady Booch, James Rumbaugh. *Unified Modeling Language User Guide*. Addison-Wesley, 2005.
- [4] Object Managment Group. Unified modeling language, v2.5.1. <https://www.omg.org/spec/UML/2.5.1/PDF>.
- [5] Susan F. Marseken Lambert M. Surhone, Miriam T. Timpledon. *Sequence Diagram*. VDM Publishing, 2010.

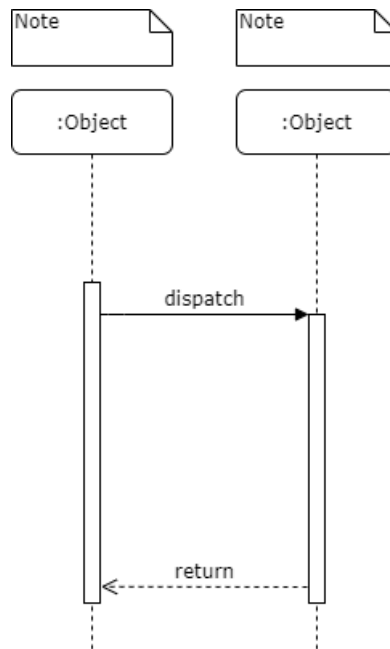


Figure 4: Messages template.