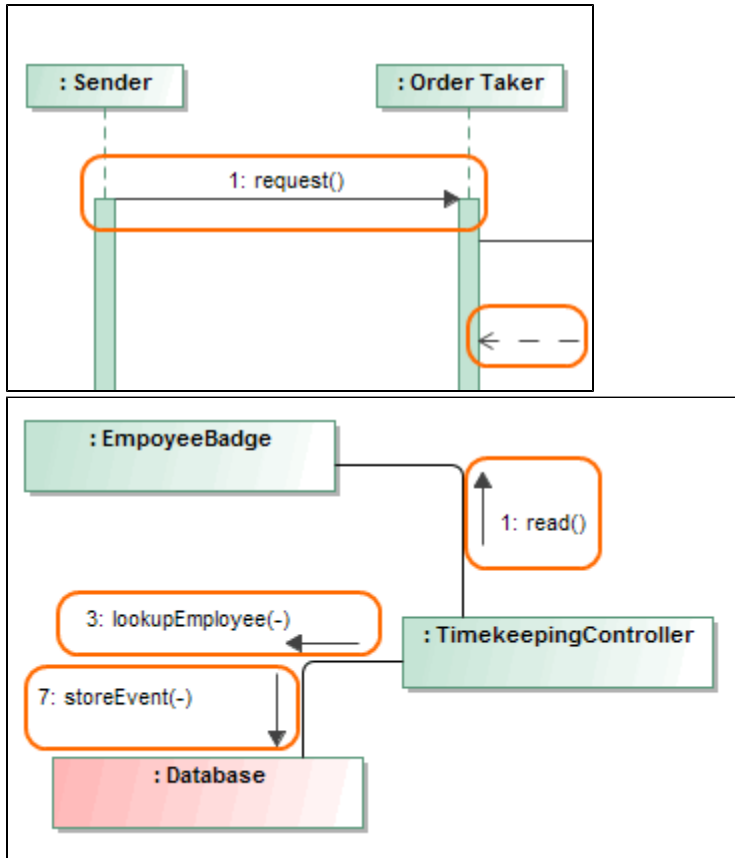


# Message

A Message defines a particular communication between [Lifelines](#) of an [Interaction](#), such as raising a Signal, invoking an [Operation](#), or creating or destroying an Instance. Messages specify the kind of communication, the sender, and the receiver.

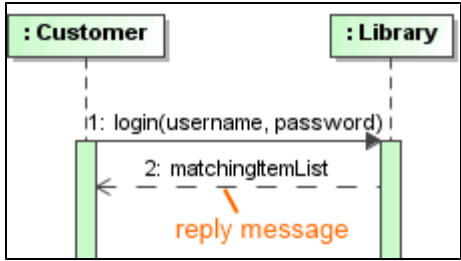
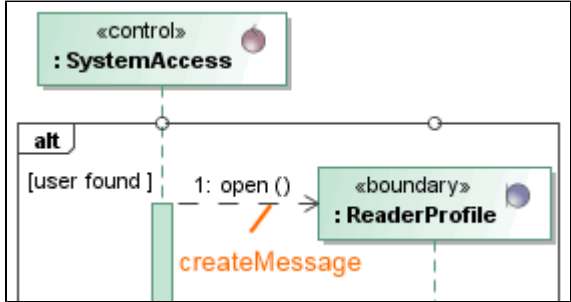
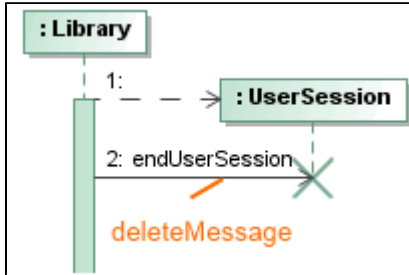
A Message is represented by arrows between the lifelines. The style of the Message line and arrowhead reflect the types of the Message.



You can see the description of a selected property in the description area of the [Specification window](#). To see descriptions, be sure the **Show Description** option is selected.

## Message sorts

Message Sort	Description	Example
synchCall	The Message was generated by a synchronous call to an Operation. Synchronous Messages can overtake each other. When a Message represents an Operation, the arguments of the Message must correspond to the parameters of the Operation.	
asynchCall	Asynchronous means that the caller continues immediately after the call. asynchCall Messages do not have reply Messages.	
asynchSignal	The Message was generated by an asynchronous send Action. The argument of the Message must correspond to the attributes of the Signal.	

reply	A reply Message to an Operation call.	 <pre> sequenceDiagram     participant Customer     participant Library     Customer-&gt;&gt;Library: 1: login(username, password)     activate Library     Library--&gt;&gt;Customer: 2: matchingItemList     deactivate Library     </pre> <p>reply message</p>
createMessage	The Message designates the creation of another Lifeline object. No other Messages on a given Lifeline in an Interaction operand may appear above a Lifeline.	 <pre> sequenceDiagram     participant SystemAccess as «control» : SystemAccess     alt [user found]         SystemAccess-&gt;&gt;ReaderProfile: 1: open()         activate ReaderProfile         createMessage ReaderProfile     end     </pre> <p>createMessage</p>
deleteMessage	The Message designates the termination of another Lifeline.	 <pre> sequenceDiagram     participant Library     participant UserSession as : UserSession     Library-&gt;&gt;UserSession: 1:      activate UserSession     Library-&gt;&gt;UserSession: 2: endUserSession     delete UserSession     </pre> <p>deleteMessage</p>

## Message syntax

The Message name is displayed on the message path on the diagram pane. The syntax of the message name is as follows:

`<messageid> ::= [<attribute> '='] <signal or operation name> ['(' [<argument> ','`

`<argument>]* ')'] ['<return value>]`

`<argument> ::= ([<parameter name> '=' <argument-value>]) / '-'`



### Example

`login(P1='a', P2=True):12`

## Common actions with Messages

To set an action type for a Message

Do one of the following:

- In the Message [Specification window](#), click the **Message Sort** property value cell and select the action type from the drop-down list.
- On the diagram pane, right-click the Message and select a desired action type from the shortcut menu.

To show / hide Message numbers

1. Right-click the diagram pane to open its shortcut menu.
2. Then do one of the following:
  - In a [Sequence](#) or Time diagram, select / clear **Show Message Numbers**.
  - In a [Communication diagram](#), select / clear **Numbering > Show Message Numbers**.

## Related pages

- [Assigning Operations to Messages](#)
- [Message name parsing](#)
- [Assigning Signals to Messages](#)
- [Creating Signal receptions for Messages](#)
- [Messages in Sequence and Time Diagrams](#)
- [Messages in Communication diagrams](#)
  
- [Model Elements](#)
- [Specification Window](#)
- [Lifelines](#)
- [Interaction](#)
- [Operation](#)