

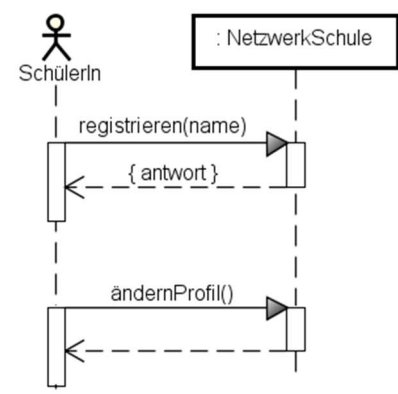
Max-Eyth-Schule	UML - Sequenzdiagramm	Name:
Tag:		Klasse:

Ablaufdarstellung - das Sequenzdiagramm

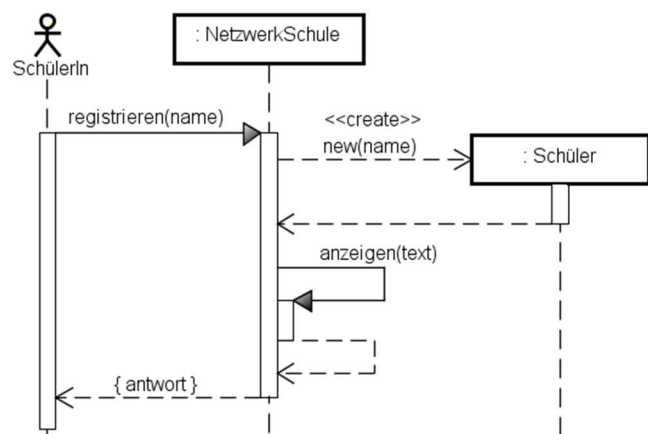
Sequenzdiagramme werden in der Entwurfsphase erstellt, sie stellen dynamische Abläufe des zu entwickelnden Systems dar. Ein Sequenzdiagramm zeigt die **Interaktionen zwischen Akteuren und Objekten**, indem die zeitliche Abfolge von Methodenaufrufen entlang einer gedachten Zeitachse von oben nach unten gezeichnet wird. Ein Sequenzdiagramm wird in vielen Fällen für einen Use-case oder einen Ausschnitt eines Use-cases (Szenario) oder auch für eine Methode erstellt. Sequenzdiagramme müssen mit dem statischen Klassendiagramm übereinstimmen. D. h. im Sequenzdiagramm können nur Methoden erscheinen, die auch im Klassendiagramm enthalten sind.

Die spätere Implementierung spielt beim Sequenzdiagramm schon eine gewisse Rolle, denn hier erscheinen die Methoden der Klassen, welche die Funktionalität des Systems implementieren. Beim Sequenzdiagramm geht es, anders als beim Use-case Diagramm schon darum, **wie** die Anforderungen im System umgesetzt werden. Ein Sequenzdiagramm ist eine Vorlage für die Implementierung.

In vielen Fällen startet ein Akteur den Ablauf, der Akteur wird mit dem bekannten Strichmännchen-Symbol dargestellt. Der Akteur sendet eine Nachricht an ein Objekt, was einen Operations- oder Methodenaufruf zur Folge hat. Das Objekt wird als Rechteck mit einer gestrichelten Lebenslinie (lifeline) dargestellt. Der Methodenaufruf als Pfeil vom Aufrufer zum Aufgerufenen, bezeichnet mit Methodennamen und Parameter. Die Methode beim Aufgerufenen ist aktiv und wird mit einem Aktivitätsbalken dargestellt. Dieser endet, wenn die Methode abgearbeitet ist und ein gestrichelter Pfeil führt zurück zum Aufrufer. An diesem Pfeil steht ggf. der Rückgabewert der Methode. Das Beispiel zeigt einen Akteur `SchülerIn`, die nacheinander zwei Methoden der Klasse `NetzwerkSchule` aufruft. Die Methode `registrieren()` hat einen Parameter `name` und einen Rückgabewert `antwort`.

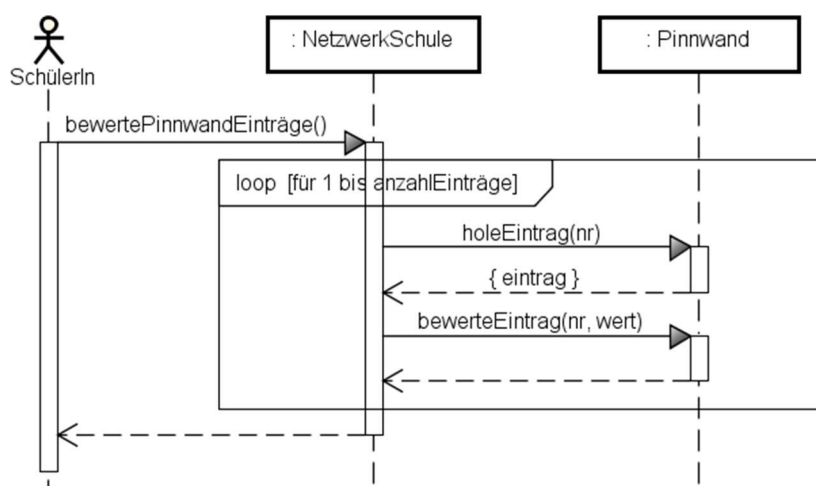


Eine Methode kann eine Methode eines Objekts einer anderen Klasse aufrufen, oder auch eine eigene Methode. Soll ein Objekt neu erzeugt werden, wird eine spezielle Methode verwendet, die `new`-Methode mit dem Vermerk `<<create>>`. Die `new`-Methode entspricht dem Konstruktor-Aufruf und wird mit Parametern gezeichnet. Im Beispiel soll die Methode `registrieren()` der Klasse `NetzwerkSchule` ein `Schüler`-Objekt erzeugen und anschließend die Methode `anzeigen()` der eigenen Klasse aufrufen, um einen Text auszugeben. Dazu das Sequenzdiagramm:

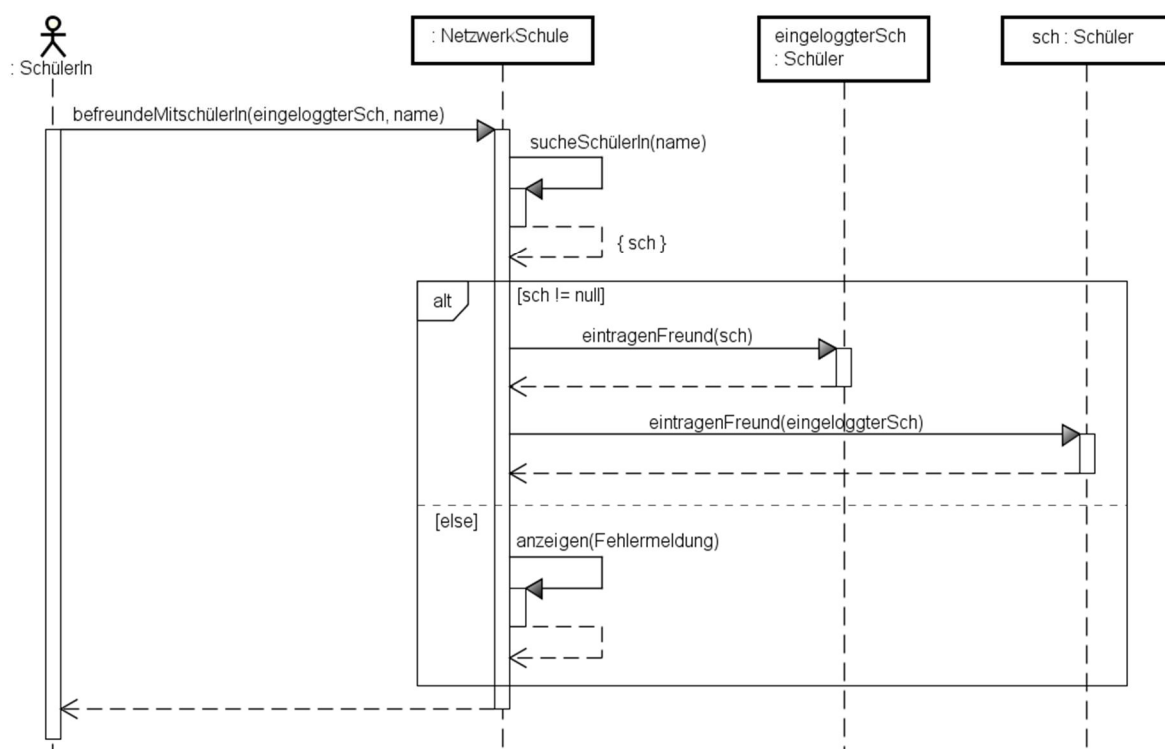


Max-Eyth-Schule	UML - Sequenzdiagramm	Name:
Tag:		Klasse:

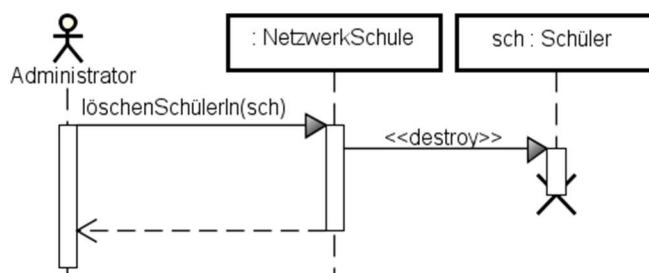
Eine, oder mehrere Methoden können mehrfach hintereinander aufgerufen werden. Im Sequenzdiagramm gibt es dafür das Schleifensymbol `loop`, gefolgt von einem Ausdruck, der die Anzahl der Durchläufe beschreibt. Im `loop`-Rechteck werden alle Methodenaufrufe gezeichnet, die wiederholt werden.



Wenn eine Methode nur in Abhängigkeit von einer Bedingung aufgerufen wird, wird das mit einem Verzweigungssymbol dargestellt, es gibt die einseitige Verzweigung `opt` (Option) ohne `else`-Zweig und die zweiseitige `alt` (Alternative) mit `else`-Zweig. Hier ein Beispiel einer zweiseitigen Verzweigung:



Zu guter Letzt werden Objekte auch gelegentlich gelöscht, wenn sie nicht mehr gebraucht werden. Das wird mit der `destroy`-Nachricht und dem Ende der Lebenslinie mit einem Kreuz dargestellt:



Alle Bausteine können in beliebiger Reihenfolge auftreten und beliebig ineinander verschachtelt werden.