

Semantic Variation Points der UML 2

“Semantic Variation Points” werden jene Bereiche in der Spezifikation der UML 2 genannt, die bewusst nicht vollständig spezifiziert wurden, um den domain-spezifischen Erweiterungen der UML (z.B. durch Profile und Stereotypen) mehr Freiheiten zu lassen.

In der folgenden Tabelle sind sämtliche Semantic Variation Points der UML 2 zusammengefasst.

Semantic Variation Point	Klasse im UML-Metamodell	Erläuterung
Definition einer Aggregation	AggregationKind (from Kernel)	Die genaue Definition einer Aggregation unterliegt dem Anwendungsgebiet bzw. dem Modellierer.
Erzeugung von Instanzen bei einer Aggregation	AggregationKind (from Kernel)	Die Reihenfolge und die Art der Erzeugung der aggregierten Part-Instanzen ist nicht definiert.
Erzeugung von Part-Instanzen bei einer Komposition	Association (from Kernel)	Die Reihenfolge und die Art der Erzeugung der in einer Komposition aggregierten Part-Instanzen ist nicht definiert.
Beziehung Assoziation-Assoziationsenden bei Ableitung	Association (from Kernel)	Die logische Beziehung zwischen der Ableitung einer Assoziation und der Ableitung ihrer Assoziationsenden ist nicht definiert.
Konkreter Lebenszyklus einer Aggregation	Classifier (from Kernel, Dependencies, PowerTypes)	Der konkrete Lebenszyklus einer Aggregation ist ein Semantic Variation Point.
Definitionsbereich von Datentypen	DataType (from Kernel)	Die Definition von Datentypen ist nicht eingeschränkt (beispielsweise existieren keine Einschränkungen hinsichtlich der erlaubten Typen ihrer Attribute).
Interpretation eines statischen Features	Feature (from Kernel)	Ein statisches Feature lässt sich auf zweierlei Weise interpretieren: Es kann verschiedene Werte für verschiedene featuringClassifier besitzen oder einen Wert für alle featuringClassifier. Daneben ist es erlaubt (aber nicht zwingend) einen Wert eines statischen Features zu vererben.
Nichterfüllte Vorbedingungen bei Operationsaufruf	Operation (from Kernel, Interfaces)	Das Verhalten eines Operationsaufrufs bei nicht erfüllten Vorbedingungen ist ein Semantic Variation Point.
Substitution bei Operationen	Operation (from Kernel, Interfaces)	Wenn Operationen bei einer Spezialisierung überschrieben werden, entscheiden Regeln hinsichtlich Invarianz, Kovarianz oder Kontravarianz ob der spezialisierte Classifier seinen Eltern-Classifier substituieren kann oder nicht. Diese Regeln sind in der UML nicht spezifiziert, sondern stellen Semantic Variation Points dar.

Semantic Variation Point	Klasse im UML-Metamodell	Erläuterung
Art der Kompatibilität zwischen überschriebenen und überschreibenden Element	RedefinableElement	Zwischen einem überschriebenen und einem überschreibenden Element sind verschiedene Abstufungen hinsichtlich ihrer Kompatibilität möglich, beispielsweise Namenskompatibilität, Sichtbarkeitskompatibilität oder Typkompatibilität. Die Einschränkung bei der Überschreibung und damit die Art der Kompatibilität ist in der UML nicht festgelegt.
Art der Kompatibilität in der Anwendung einer Kollaboration	CollaborationUse (from Collaborations)	Es ist in der UML nicht festgelegt, wann Client-Elemente und Supplier-Elemente zueinander kompatibel sind und wann somit die Kollaborationsanwendung "korrekt" angewendet wird.
Art der Kompatibilität zwischen verbindbaren Elementen	Connector (from InternalStructures)	Was verbindbare Elemente kompatibel (für eine Kommunikationsverbindung untereinander) macht, ist ein Semantic Variation Point.
Weiterleitung von Requests bei mehreren Links	Port (from Ports)	Wenn verschiedene Konnektoren auf einer Seite eines Ports angetragen sind, wird jede Anfrage, die diesen Port auf der anderen Seite erreicht, durch Links dieser Konnektoren weitergereicht. Dabei lässt die UML offen, ob die Anfrage durch alle Links oder nur durch einen Link weitergereicht wird. Im letzteren Fall wäre eine Möglichkeit, den Link zur Weitergabe der Anfrage willkürlich unter denen auszuwählen, die zu einer Instanz führen, die in der Lage ist, diese Anfrage zu bearbeiten.
Umsetzung der Multiplizitäten von Konnektorenden, Parts und Ports	StructuredClassifier (from InternalStructures)	Die Regeln, mithilfe derer die Multiplizitäten der Konnektor-Enden sowie der durch die Konnektoren verbundenen Parts und Ports eingehalten werden, ist ebenso wie die sich hieraus ergebende Topologie ein Semantic Variation Point. Das Star-Connector-Pattern oder das Array-Connector-Pattern bieten in bestimmten Konstellationen Möglichkeiten zur Abbildung (siehe UML 2 glasklar, 2. Auflage, Kap. 9.4.2).
Auswahl von Broadcast-Zielobjekten	BroadcastSignalAction (from IntermediateActions)	Die Auswahl des Sets von Broadcast-Zielobjekten ist nicht spezifiziert.
Auswahl der geeigneten Methode durch CallOperationAction	CallOperationAction (from BasicActions)	Der Mechanismus für die Auswahl der Methode, welche als Ergebnis der CallOperationAction aufgerufen wird, ist nicht spezifiziert.
Umfang des Extents	ReadExtentAction (from CompleteActions)	Das Set von Instanzen, das durch eine ReadExtentAction gelesen werden soll (Extent), umfasst nicht alle denkbaren Instanzen eines Classifiers, sondern lediglich ein Subset davon. Welches Subset von Instanzen gelesen werden soll, ist durch die UML nicht spezifiziert.

Semantic Variation Point	Klasse im UML-Metamodell	Erläuterung
Nichterfüllte Vor-oder Nachbedingungen einer Aktion	Action (from CompleteActivities, FundamentalActivities, StructuredActivities)	Die Implementierung eines Anwenders legt fest, wie die Einhaltung lokaler Vor- und Nachbedingungen von Aktionen durchgesetzt wird. Werden diese Bedingungen verletzt, bedeutet dies nicht, dass der Aufruf der Aktion nicht definiert im Sinne der UML ist, sondern lediglich, dass das Modell bzw. die Ausführung des Ablaufs nicht den Intentionen des Modellierers entspricht.
Spezifikation verschiedener Token	Activity (from BasicActivities, CompleteActivities, FundamentalActivities, StructuredActivities)	Die UML 2 lässt die Definition von Token weitestgehend offen, bietet aber die Möglichkeit, das Metamodell um spezifische Typen von Token zu erweitern.
Zugriff auf Variable durch Aktionen	Variable (from StructuredActivities)	Die UML 2 lässt offen, welche Aktionen auf eine Variable zugreifen können - dies muss bei der Modellierung, beispielsweise durch Kontrollflüsse oder Randbedingungen, festgelegt werden.
Transportwege von Anfragen	Behavior (from BasicBehaviors)	Die Wege, auf denen Anfragen zu ihrem Ziel transportiert werden sind abhängig von zahlreichen Faktoren wie beispielsweise der Typ der anfragenden Aktion, das Ziel, die Eigenschaften des Kommunikationsmediums usw. und daher nicht Gegenstand der UML 2 Spezifikation.
Kompatibilität zwischen Behavior und BehavioralFeature	BehavioralFeature (from BasicBehaviors, Communications)	Die UML 2 lässt offen, inwieweit die Parameter eines Verhaltensmerkmals (BehavioralFeature) mit den Parametern der zugeordneten Verhaltensspezifikation (Behavior) übereinstimmen müssen.
Verhalten bei mehreren passenden Triggern für einen Event	BehavioredClassifier (from BasicBehaviors, Communications)	Es ist ein Semantic Variation Point, ob ein oder mehrere Verhalten gestartet werden, sobald ein Event mehrere Trigger bei einem Classifier auslösen könnte.
Verarbeitung eines Events ohne passenden Trigger	BehavioredClassifier (from BasicBehaviors, Communications)	Wenn ein Event an einem Wartepunkt keinen der zur Verfügung stehenden Trigger auslösen kann, existiert ein Semantic Variation Point in Bezug darauf, wie dieser Event verarbeitet wird.
Reihenfolge der Events in der Eingangswarteschlange	BehavioredClassifier (from BasicBehaviors, Communications)	Die Reihenfolge der Events in der Eingangswarteschlange eines Objekts ist von der UML 2 Spezifikation nicht geregelt.
Auswertung einer ChangeExpression	ChangeEvent (from Communications)	Es existiert ein Semantic Variation Point bei der Frage, wann eine ChangeExpression ausgewertet wird. Beispielsweise könnte sie kontinuierlich solange ausgewertet werden, bis ihr Wert "true" ist und ein ChangeEvent erzeugt werden kann.
Lebensdauer eines ChangeEvents	ChangeEvent (from Communications)	Es ist in der UML 2 nicht spezifiziert, ob ein ChangeEvent bestehen bleibt bis er verbraucht wird, auch wenn der Wert der ChangeExpression zu "false" wird.

Semantic Variation Point	Klasse im UML-Metamodell	Erläuterung
Auswahl des geeigneten Verhaltens	Operation (from Communications)	Die UML 2 lässt offen, wie das aufzurufende Verhalten durch eine Operation und die übermittelten Argumente ausgewählt wird. Für objekt-orientierte Sprachen schlägt die UML 2 Spezifikation folgendes vor: Sobald ein call request vom Zielobjekt empfangen wird, wird die Klasse des Zielobjekts daraufhin überprüft, ob sie eine Operation mit passenden Parametern besitzt. Wird eine solche Operation gefunden, wird das entsprechende Verhalten ausgeführt. Andernfalls wird die Vererbungshierarchie weiter zur Wurzel hin durchsucht (bei Mehrfachvererbung werden alle Eltern überprüft). Wird genau eine Operation gefunden, wird das entsprechende Verhalten ausgeführt. Werden bei mehreren Superklassen geeignete Operationen gefunden, gilt das Modell als fehlerhaft (ill-formed).
Transportwege von Anfragen	SignalEvent (from Communications)	Die Wege, auf denen Anfragen zu ihrem Ziel transportiert werden sind abhängig von zahlreichen Faktoren wie beispielsweise der Typ der anfragenden Aktion, das Ziel, die Eigenschaften des Kommunikationsmediums usw. und daher nicht Gegenstand der UML 2 Spezifikation.
Verzögerungen zwischen Empfang und Ausführung eines TimeEvents	TimeEvent (from Communications, SimpleTime)	Es kann eine nicht bestimmbare Verzögerung zwischen dem Empfang und der Ausführung eines TimeEvents auftreten.
Zeitintervalle bei Zustandsübergängen von Events	Trigger (from Communications)	Die UML 2 Spezifikation trifft keine Annahmen bezüglich der Zeitintervalle zwischen dem Auftreten eines Ereignisses, seiner Ausführung und seinem Verbrauch, so dass bei der Modellierung beispielsweise absolut verzögerungsfreie Übergänge (Zeitintervall=0) festgelegt werden können.
Verarbeitung eines Events ohne geeigneten Trigger	Trigger (from Communications)	Die UML 2 Spezifikation lässt offen, ob ein Event, für das kein geeigneter Trigger modelliert wurde, verworfen werden soll.
Verarbeitung eines Events in einer unerwarteten Situation	ProtocolTransition (from ProtocolStateMachines)	In der UML 2 ist nicht festgelegt, wie mit einem Event, das in einer unerwarteten Situation (beispielsweise bei einer nicht erfüllten Vorbedingung oder Zustandsinvariante) empfangen wird, verfahren werden soll: Möglich wäre beispielsweise ein Ignorieren oder zurückweisen des Events, das Werfen einer Exception oder Stopp der Anwendung.
Unerwartetes Verhalten	ProtocolTransition (from ProtocolStateMachines)	In der UML 2 ist nicht festgelegt, wie mit einem unerwarteten Verhalten durch ein unerwartetes Ergebnis einer Transition (beispielsweise bei einem fehlerhaften Endzustand, einer nicht erfüllten Nachbedingung oder Zustandsinvariante) verfahren werden soll. Die UML 2 Spezifikation schlägt vor, dies als fehlerhafte Modellierung des Protokollzustandsautomaten zu interpretieren.

Semantic Variation Point	Klasse im UML-Metamodell	Erläuterung
Transition auf zusammengesetzten Zustand ohne Startzustand in den Regionen	State (from BehaviorStateMachines, ProtocolStateMachines)	Wenn eine Transition auf einen zusammengesetzten Zustand zeigt, dessen Regionen keinen Startzustand besitzen, wird dies von der UML 2 als Semantic Variation Point interpretiert. Teilweise wird ein solches Modell als fehlerhaft angesehen. Das Modell könnte man aber auch so interpretieren, dass der Zustandsautomat in diesem zusammengesetzten Zustand verbleibt, ohne die Regionen oder ihre Unterzustände zu betreten.
Ort des Erweiterungspunktes	ExtensionPoint (from UseCases)	Die UML 2 Spezifikation lässt offen, auf welche Weise die Stelle eines Erweiterungspunktes definiert wird.
Verwendung von Randbedingungen bei Parametrisierbaren Elementen	ParameterableElement (from Templates)	Die Durchsetzung von Randbedingungen von Parametrisierbaren Elementen ist ein Semantic Variation Point: Einerseits verlangt die Anwendung von Randbedingungen die Übereinstimmung der aktuellen mit den formalen Parametern, sichert aber ein "wohl-geformtes" Modell im Sinne der UML. Andererseits bietet der Verzicht auf Randbedingungen mehr Flexibilität, allerdings könnte es bei einigen aktuellen Parametern zu Komplikationen bei der Bindung führen.
Auswahl der Parameter bei Parameterbindung	TemplateBinding (from Templates)	Die UML 2 Spezifikation lässt bei der Bindung von Parametern offen, ob alle TemplateParameter gebunden werden (vollständige Bindung) oder nur ein Subset der TemplateParameter zur Bindung verwendet werden darf (teilweises Binden). Bei vollständigem Binden kann das gebundene Element eigene formale Templateparameter besitzen, welche als aktuelle Parameter für die Bindung verwendet werden können. Bei teilweisem Binden stellen die nicht verwendeten TemplateParameter formale TemplateParameter des gebundenen Elements dar.

Semantic Variation Point	Klasse im UML-Metamodell	Erläuterung
Verwendung von Randbedingungen bei Parameterbindung	Classifier (from Templates)	<p>Wenn Randbedingungen für TemplateParameter angewendet werden, wird die Verwendung eines Classifiers als aktueller Parameter wie folgt beschränkt:</p> <ul style="list-style-type: none"> - wenn der als TemplateParameter verwendete Classifier eine Generalisierung besitzt, muss auch der als aktueller Parameter verwendete Classifier eine Generalisierung mit dem selben Super-Classifier besitzen. - wenn der als TemplateParameter verwendete Classifier eine Substitution besitzt, muß auch der als aktueller Parameter verwendet Classifier eine Substitution mit den selben Verpflichtungen besitzen. - wenn der als TemplateParameter verwendete Classifier weder eine Generalisierung noch eine Substitution besitzt, kann ein beliebiger Classifier als aktueller Parameter verwendet werden. <p>Wenn keine Randbedingungen für TemplateParameter verwendet werden, kann ein beliebiger Classifier als aktueller Parameter verwendet werden.</p>
Interpretation von verschiedenen Icons eines Stereotyps	Stereotype (from Profiles)	<p>Die UML 2 sieht vor, dass einem Stereotyp verschiedene Bilder (beispielsweise je nach Kontext oder Ansicht) zugeordnet werden können. Die Interpretation dieser verschiedenen Bilder für einen Stereotyp stellt allerdings einen Semantic Variation Point dar.</p>