

# OPERATIONALE SEMANTIKDEFINITION FÜR KONZEPTUELLES REGELWISSEN

Bodo Kraft, Daniel Retkowitz

Lehrstuhl für Informatik III,  
RWTH Aachen, D-52056 Aachen

(kraft|daniel)@i3.informatik.rwth-aachen.de

**Kurzfassung:** *Mittels eines operationalen Ansatzes zur Semantikdefinition wird am Beispiel des konzeptuellen Gebäudeentwurfs ein Regelsystem formalisiert. Dazu werden zwei Teile, zum einen das Regelwissen, zum anderen ein konzeptueller Entwurfsplan zunächst informell eingeführt und dann formal beschrieben. Darauf aufbauend wird die Grundlage für eine Konsistenzprüfung des konzeptuellen Entwurfs gegen das Regelwissen formal angegeben.*

## 1 Einleitung

Als Grundlage für alle folgenden Phasen ist der konzeptuelle Gebäudeentwurf am Anfang des Entwurfsprozesses maßgeblich für den Erfolg eines Bauprojekts verantwortlich. In einem frühen Entwurfsplan erarbeitet der Architekt im Rahmen des konzeptuellen Entwurfs die Struktur und Organisation eines zukünftigen Gebäudes in einer abstrakten Sichtweise. Das Ziel besteht darin, unter Beachtung aller ökonomischen, rechtlichen, technischen, organisatorischen, sozialen und funktionalen Randbedingungen einen ersten groben Gebäudeplan zu entwickeln, der als Vorlage für die weitere Ausarbeitung dient. Für den konzeptuellen Entwurf spielen später wichtige Grundlagen wie Materialdefinition und exakte Dimensionen noch keine Rolle, eine zu detaillierte Sicht auf einen Entwurf verhindert sogar die kreative und bewusst ungenaue konzeptuelle Arbeitsweise. Bisher gibt es kaum Werkzeuge, die den Architekten in der konzeptuellen Entwurfsphase unterstützen. Die existierenden CAD-Werkzeuge bieten zwar umfassende Unterstützung für den detaillierten konstruktiven Entwurf, sie bieten aber weder die Möglichkeit zur Abstraktion auf einfachere Gebäudeelemente (wie z. B. den Raum) noch bieten sie die Möglichkeit zur Prüfung eines konzeptuellen Entwurfs gegen geltendes Fachwissen.

Das Condes-Projekt [2] an der RWTH Aachen beschäftigt sich mit der Unterstützung des konzeptuellen architektonischen Entwurfs. In einem Teilbereich wird die Möglichkeit zur Formalisierung von konzeptuellem Fachwissen untersucht und in einem zweiten Bereich Erweiterungsmöglichkeiten für industrielle CAD-Systeme erforscht. In diesem Beitrag wird ein Ausschnitt der theoretischen Grundlage für die Formalisierung konzeptuellen Fachwissens dargestellt. Dazu wird zunächst die Systemarchitektur des gesamten Projekts erläutert und eine anwenderfreundliche Sichtweise der internen Darstellung des formalisierten Fachwissens und des konzeptuellen Entwurfs geboten. Im Hauptteil wird schließlich die Semantik der Regeln mit Hilfe eines operationalen Ansatzes formal definiert.

## 2 Systemarchitektur

Die Systemarchitektur des Condes-Projekts besteht aus einer Komponente zur Wissensformalisierung (in Abbildung 1 links) und einer weiteren Komponente zur konzeptuellen Entwurfsunterstützung (in Abbildung 1 rechts). Auf Basis der Vereinigung beider Komponenten wird, unterstützt durch entsprechende Werkzeugfunktionalität, die Prüfung eines konzeptuellen Gebäudeentwurfs gegen das formalisierte Fachwissen ermöglicht.

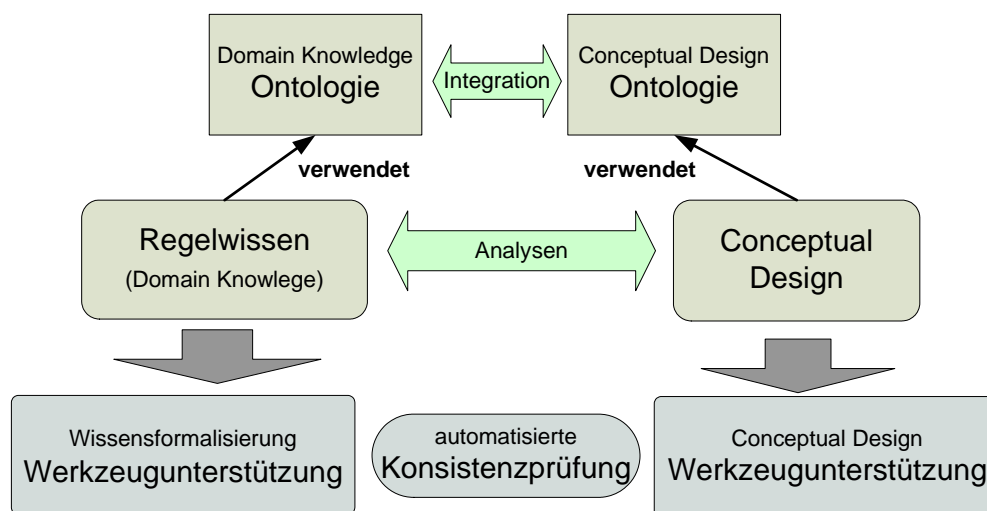


Abb. 1: Systemarchitektur, bestehend aus Wissensformalisierung und konzeptueller Entwurfsunterstützung

Beide Komponenten zeigen einen ähnlichen, mehrstufigen Aufbau. Im Bereich der Wissensformalisierung (links) kann ein Wissensingenieur, z. B. ein Architekt oder Bauingenieur, konzeptuelles Fachwissen in zwei Stufen eingeben. Im ersten Schritt definiert er dazu in der *Domain Knowledge Ontologie* die für eine Gebäudeklasse relevanten Konzepte [1; 5]. Als Strukturierungshilfe stehen hierzu die Elemente *Semantic Object*, *Relation* und *Attribute* zur Verfügung. Ein Semantisches Objekt beschreibt eine für den

Gebäudeentwurf konzeptuell relevante Einheit, diese wird spezifisch für eine Gebäudeklasse definiert. Sinnvolle Semantische Objekte für die Gebäudeklasse KFZ-Werkstatt wären z. B. eine Empfangshalle, Toilettenräume und eine Lagerhalle. Semantische Objekte können durch Vererbung und Aggregation klassifiziert und strukturiert werden. Relationen werden bei der Wissensformalisierung verwendet, um Beziehungen zwischen Semantischen Objekten zu fordern oder zu verbieten. In der Ontologie werden für die Formalisierung von Wissen über die Gebäudeklasse KFZ-Werkstatt z. B. die Relationen Kundenzugang, Personalzugang, Sicht, Nähe, KFZ-Einfahrt benötigt. Schließlich beschreiben Attribute im Bereich der Wissensformalisierung weitere Eigenschaften von Semantischen Objekten, z. B. Größenanforderungen oder die Notwendigkeit bzw. das Verbot einer bestimmten Ausstattung. Attribute werden unterschieden in Integerattribute zur Definition eines gültigen Wertebereichs und Boolesche Attribute. Die Möglichkeit zur Definition der relevanten Begriffe durch den Wissensingenieur zur Laufzeit in einer Ontologie erhöht die Flexibilität der Werkzeuge. Dadurch wird die Akzeptanz gesteigert. Die Qualität der Wissensformalisierung wird ebenfalls verbessert, da zum einen ein Domänenexperte Spezialwissen für die jeweilige Domäne definieren kann und zum anderen durch wiederholte Überarbeitung der Ontologie Fehler und Unvollständigkeiten behoben werden.

Im zweiten Schritt der Wissensformalisierung verwendet der Wissensingenieur die vorab in der Ontologie definierten Begriffe, um mit Hilfe von Regeln das eigentliche Fachwissen über die betrachtete Gebäudeklasse zu formalisieren. Die vordefinierten Regeltypen bieten die nötige Aussagekraft, um Wissen für einen architektonischen, konzeptuellen Entwurf darzustellen. Das Fachwissen besteht somit aus Gültigkeitsregeln, z. B. für die Größe, Anordnung und Existenz von Räumen, Bereichen und Teilbereichen oder die Erreichbarkeit bzw. Entfernung zwischen Räumen. Durch entsprechende Werkzeugfunktionalität wird der Wissensingenieur bei der Entwicklung der Ontologie und bei der Formalisierung des Fachwissens umfassend unterstützt.

Die zweite Komponente der Systemarchitektur (in Abbildung 1 rechts) beschreibt den Aufbau des konzeptuellen Entwurfs. Man kann wieder eine Ontologie zur Definition der relevanten Begriffe für einen konzeptuellen Entwurf erkennen. Auf dieser Ontologie baut, analog der Systemarchitektur zur Wissensdefinition, der eigentliche konzeptuelle Entwurfsplan auf. Durch Erweiterung des kommerziellen CAD-Systems ArchiCAD wird in diesem Bereich dem Architekten eine angemessene Werkzeugfunktionalität zur Verfügung gestellt.

Beide Teile, also die Wissensformalisierung und der konzeptuelle Entwurf werden durch die Integration beider Ontologien verknüpft, sodass auf den vordefinierten Regeltypen basierende Konsistenzanalysen den konzeptuellen Entwurf im CAD-System gegen das definierte Fachwissen prüfen können. Im Fall von Regelverletzungen werden Fehlermeldungen im CAD-Werkzeug dargestellt, um dem Architekten die Möglichkeit zur Überarbeitung seines Entwurfs zu geben.

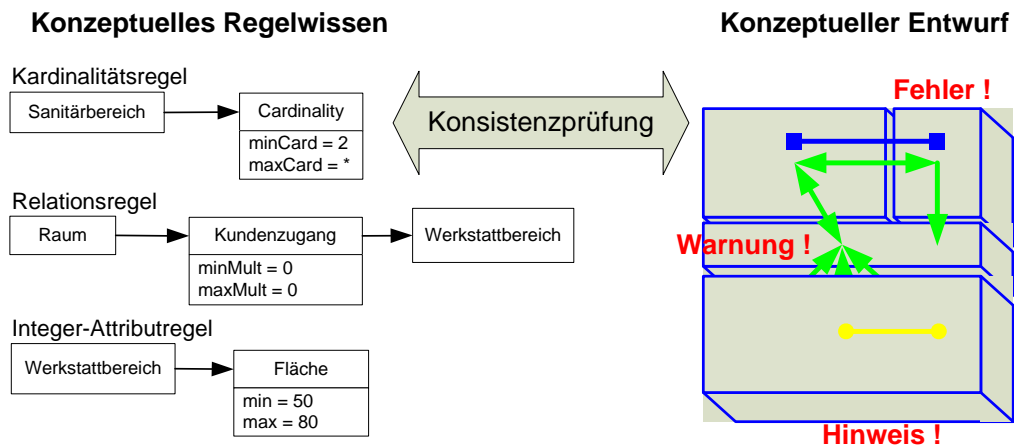


Abb. 2: Abstrakte Darstellung des Regelwissens und eines konzeptuellen Entwurfs

In Abbildung 2 ist eine schematische Abbildung beider Teile der konzeptuellen Entwurfsunterstützung dargestellt. Auf der linken Seite sind exemplarisch die Instanzen von drei Regeltypen, Kardinalitätsregel, Relationsregel und Integer-Attributregel abgebildet. Die Regeln entsprechen einem Ausschnitt aus dem in Abbildung 1 links eingeführten Regelwissen. Die zugrunde liegende Ontologie ist nicht dargestellt. Die Kardinalitätsregel in der Abbildung besteht aus einem Semantischen Objekt (Sanitärbereich) und einem Wertebereich, definiert durch *minCard* und *maxCard*. Die Kardinalitätsregel fordert, dass in einem konzeptuellen Entwurf, mindestens zwei Sanitärbereiche vorkommen müssen, die Obergrenze ist nicht eingeschränkt. Das zweite Beispiel zeigt eine Relationsregel, bestehend aus zwei semantischen Objekten (Raum, Werkstattbereich) und einer Relation (Kundenzugang). Durch die Einschränkung der minimalen und maximalen Multiplizität auf null, wird das Verbot eines Kundenzugangs zwischen einem beliebigen Raum und dem Werkstattbereich ausgedrückt. In diesem Fall wird das Semantische Objekt *Raum* als Wurzelknoten der objekt-orientiert strukturierten Ontologie angenommen, sodass die Regel ebenso für alle von *Raum* erbinden Semantischen Objekte gilt. Die dritte Regel in Abbildung 2 zeigt eine Integer-Attributregel, die für den Werkstattbereich eine Größenanforderung, im Beispiel von 50m<sup>2</sup> bis 80m<sup>2</sup> definiert. Diese Regel ist erfüllt, falls im konzeptuellen Entwurf einer KFZ-Werkstatt die Größe jedes Werkstattbereichs innerhalb dieser Werte liegt. Die hier dargestellten Regeltypen zeigen nur einen Ausschnitt auf die grundlegenden Konzepte zur Regeldefinition, weitere komplexe Regeltypen und erweiterte Ausdrucksmittel bieten eine umfassendere Aussagekraft zur Definition von konzeptuellem Fachwissen [4].

Der konzeptuelle Entwurf, dargestellt auf der rechten Seite in Abbildung 2, zeigt verschiedene Semantische Objekte, nun im konzeptuellen Entwurf. Auf die Einführung der zugrunde liegenden Ontologie wurde hier wiederum verzichtet. Die schematisch dargestellten Semantischen Objekte sind verknüpft durch Relationen, ein Pfeil stellt z. B. den Kundenzugang, die Sichtbeziehung oder den Personalzugang zwischen zwei Räumen

in diesem konzeptuellen Entwurf dar. Ebenfalls angedeutet wird, dass innerhalb von komplexen Bereichen weitere Räume oder Teilbereiche eingeschachtelt werden können, z. B. sind im Werkstattbereich, die Teilbereiche KFZ-Diagnose, Rad-Auswuchtung, Batterie-Ladestation usw. definiert. Diese Aggregation von Semantischen Objekten ermöglicht sowohl die Entwicklung eines konzeptuellen Entwurfs nach dem klassischen Top-Down Paradigma, als auch die Darstellung eines konzeptuellen Entwurfs in verschiedenen Abstraktionsstufen. Durch die Konsistenzanalyse wird der konzeptuelle Entwurf gegen das Regelwissen geprüft und Fehlermeldungen innerhalb des Entwurfsplans visualisiert.

### 3 Operationale Semantikdefinition

#### 3.1 Regelsemantik

In diesem Abschnitt wird die Semantik der Regeln formal definiert. Zur formalen Beschreibung der Semantik werden im Allgemeinen drei Hauptansätze unterschieden: die operationale Semantikdefinition, die denotationale Semantikdefinition und die axiomatische Semantikdefinition [3]. Die *operationale Semantikdefinition* entspricht am ehesten der Implementierungssicht. Die syntaktischen Elemente der Sprache werden als Mengen betrachtet, und die Semantik der Sprache wird durch Logikformeln über diesen Mengen definiert. Auf diese Weise wird die maschinelle Verarbeitung von Ausdrücken dieser Sprache bis ins Detail festgelegt, weshalb sich diese Art der Semantikdefinition besonders im Hinblick auf eine korrekte Implementierung eignet. Die *denotationale Semantikdefinition* entspricht der Sprachdesignersicht. Die Ein- / Ausgabefunktion der syntaktischen Elemente der Sprache wird induktiv über den Aufbau der Syntax definiert. Die *axiomatische Semantikdefinition* entspricht im Wesentlichen der Sicht des Programmierers. Es werden logische Eigenschaften der syntaktischen Konstrukte beschrieben. Daher ist die axiomatische Semantikbeschreibung gut für Korrektheitsbeweise geeignet.

Die folgende formale Beschreibung der Regelsemantik folgt hier dem operationalen Ansatz der Semantikbeschreibung. Zunächst werden einige Konstanten definiert, gefolgt von der Definition eines Regelsystems, zur Darstellung des Regelwissens und eines Designs, welches den konzeptuellen Entwurf beschreibt. Dann folgt die Erläuterung der Auswertung von Regelsystemen in Bezug auf ein Design, als Grundlage der Konsistenzprüfung.

##### 3.1.1 Konstanten

Zum Aufbau von Regeln werden bestimmte grundlegende Datentypen benötigt. Dazu sind die folgenden Konstanten definiert.

$C_{int}$  : Die Menge aller Integerwerte.

$C_{IntS}$  : Die Menge aller Integerwerte zuzüglich \* für unendlich.

$C_{String}$  : Die Menge aller Strings.

### 3.1.2 Regelsysteme

Ein Regelsystem besteht aus einer fachspezifischen Ontologie, die die betrachteten Konzepte definiert und einer Regelmenge, die das eigentliche Fachwissen in Form einzelner Regeln enthält. Die fachspezifische Ontologie ist in Abbildung 1 mit *Domain Knowledge Ontologie* bezeichnet, die Regelmenge mit *Regelwissen*. Ein solches Regelsystem  $R$  ist als Tupel  $R = (S_R, I_R, B_R, R_R, p_R, c_R, \Theta_R)$  definiert. Das Tupel besteht aus den folgenden Elementen.

$S_R$  : Die Menge der semantischen Objekte in der Ontologie.

$I_R$  : Die Menge der Integer-Attribute in der Ontologie.

$B_R$  : Die Menge der Booleschen Attribute in der Ontologie.

$R_R$  : Die Menge der Relationen in der Ontologie.

$\Theta_R$  : Die Menge aller Regeln.

$p_R$  : Die Parent-Funktion  $p_R : S_R \rightarrow \{\text{null}\} \cup S_R$  legt für jedes semantische Objekt ein anderes semantisches Objekt als Vaterobjekt fest. Dabei bedeutet *null*, dass es sich um einen Wurzelknoten der Ontologie handelt und daher kein Vaterobjekt vorhanden ist.

$c_R$  : Die Contained-In-Funktion  $c_R : S_R \rightarrow \{\text{null}\} \cup S_R$  dient der Festlegung der Aggregationsbeziehung und legt zu jedem semantischen Objekt fest, in welchem anderen semantischen Objekt es enthalten ist. Hier bedeutet *null*, dass das semantische Objekt in keinem anderen Objekt enthalten ist.

Die Regelmenge  $\Theta_R$  setzt sich aus mehreren Mengen für die verschiedenen Regeltypen zusammen. Jede dieser Mengen enthält Tupel, die die Regeln repräsentieren.

Die Menge **aller Regeln**  $\Theta_R$ :

$$\Theta_R = R_{card} \cup R_{rel} \cup R_{iattr} \quad (1)$$

Die Menge der **Kardinalitätsregeln**  $R_{card}$ :

$$R_{card} \subseteq \{(s, card_{min}, card_{max}) \in S_R \times C_{Int} \times C_{IntS}\} \quad (2)$$

Die Menge der **Relationsregeln**  $R_{rel}$ :

$$R_{rel} \subseteq \{(s_1, l, s_2, mult_{min}, mult_{max}) \in S_R \times R_R \times S_R \times C_{Int} \times C_{IntS}\} \quad (3)$$

Die Menge der **Integer-Attributregeln**  $R_{iattr}$  :

$$R_{iattr} \subseteq \{(s, i, val_{min}, val_{max}) \in S_R \times I_R \times C_{Int} \times C_{IntS}\} \quad (4)$$

Wir beschränken uns hier auf die drei oben aufgeführten, grundlegenden Regeltypen, um die Idee der Semantikformalisierung ausführlich demonstrieren zu können. Die Formalisierung der übrigen Regeltypen, die hier nicht vorgestellt wurden, erfolgt jedoch nach demselben Muster.

### 3.1.3 Designs

Ein Design entspricht einem konkreten konzeptuellen Gebäudeentwurf. Es besteht aus einer gebäudespezifischen Ontologie, die die für diesen Entwurf benötigten Konzepte enthält. Zudem besteht ein Design aus einer Menge von Instanzen der Konzepte, denen Attribute zugewiesen werden können und die mittels Relationen verbunden werden. Die gebäudespezifische Ontologie ist in Abbildung 1 mit *Conceptual Design Ontologie* bezeichnet, der eigentliche Entwurfsplan ist mit *Conceptual Design* bezeichnet. Ein Design  $D$  ist formal definiert als ein Tupel  $D = (S_D, I_D, B_D, R_D, \Delta_D, \delta_D, i_D, b_D, r_D)$ . Dieses Tupel besteht aus den folgenden Elementen.

$S_D$  : Die Menge der semantischen Objekte im Design.

$I_D$  : Die Menge der Integer-Attribute im Design.

$B_D$  : Die Menge der Booleschen Attribute im Design.

$R_D$  : Die Menge der Relationen im Design.

$\Delta_D$  : Die Menge der Instanzen von semantischen Objekten  $S_D$  im Design.

$\delta_D$  : Die Typ-Funktion  $\delta_D : \Delta_D \rightarrow S_D$  weist jeder Instanz im Design das entsprechende semantische Objekt der Designontologie zu, von dem es eine Instanz ist.

$i_D$  : Die Relation  $i_D \subseteq \Delta_D \times I_D \times C_{Int}$  ist die Zuordnung von Integer-Attributen und Integer-Werten zu Instanzen.

$b_D$  : Die Relation  $b_D \subseteq \Delta_D \times B_D$  ist die Zuordnung von Booleschen Attributen zu Instanzen.

$r_D$  : Die Relation  $r_D \subseteq \Delta_D \times I_D \times \Delta_D$  legt die Verbindungen zwischen semantischen Objekten durch Relationen fest.

### 3.1.4 Korrespondenzen

Die Korrespondenzen zwischen den Elementen der gebäudespezifischen Ontologie im konzeptuellen Entwurfsteil und den Elementen einer zugehörigen fachspezifischen On-

tologie im Regelwissen werden durch die untenstehenden Korrespondenz-Funktionen festgelegt. So wird eine Integration der beiden Ontologien erreicht.

$\text{corr}_{S_D}^{S_R} : S_D \rightarrow S_R$  : Jedem semantischen Objekt im Design wird ein semantisches Objekt im Regelsystem zugeordnet.

$\text{corr}_{I_D}^{I_R} : I_D \rightarrow I_R$  : Jedem Integer-Attribut im Design wird ein Integer-Attribut im Regelsystem zugeordnet.

$\text{corr}_{B_D}^{B_R} : B_D \rightarrow B_R$  : Jedem Booleschen Attribut im Design wird ein Boolesches Attribut im Regelsystem zugeordnet.

$\text{corr}_{R_D}^{R_R} : R_D \rightarrow R_R$  : Jeder Relation im Design wird eine Relation im Regelsystem zugeordnet.

## 3.2 Auswertung

In diesem Abschnitt wird die Konsistenzprüfung eines konzeptuellen Gebäudeentwurfs beschrieben. Die so genannten *Design Checks* prüfen, ob ein Design die Anforderungen, die durch das Regelsystem gestellt werden, erfüllt. Falls eine Regel durch das Design nicht erfüllt ist, wird dem Benutzer durch das Werkzeug eine Warnmeldung ausgegeben, so dass dieser entsprechende Änderungen vornehmen kann. Beim Entwerfen der Werkzeuge, stand im Vordergrund, dass die Kreativität des Architekten nicht eingeschränkt wird. Daher ist es möglich die Benachrichtigungen über nicht erfüllte Regeln zu ignorieren oder erst später entsprechende Veränderungen vorzunehmen. Es ist also aus oben genannten Gründen auch möglich nicht-regelkonforme Entwürfe zu erstellen. Um die Warnmeldungen zu erzeugen, müssen die Regeln bezüglich eines zu prüfenden Entwurfs ausgewertet werden. Dazu wird in diesem Abschnitt die Semantik der verschiedenen Regeltypen formal beschrieben.

### 3.2.1 Auswertung eines Gebäudeentwurfs

Eine Design  $D$  erfüllt ein Regelsystem  $R$  genau dann, wenn

$$\forall r \in \Theta_R : \langle r \rangle_D = \text{true}. \quad (5)$$

Dabei ist  $\langle r \rangle_D \in \{\text{true}, \text{false}\}$  die Auswertung einer Regel  $r$  bezüglich eines Designs  $D$ .

Die Auswertung einer Regel hängt vom Regeltyp ab. Der folgende Abschnitt beschreibt die Auswertung für verschiedene Regeltypen. Wir beschränken uns hier auf die drei Basisregeltypen, die Kardinalitätsregeln, die Relationsregeln und stellvertretend für Attributregeln, die Integer-Attributregeln.

### 3.2.2 Überprüfung einzelner Regeln

Für eine Regel  $r \in R$  und ein Design  $D$  gilt  $\langle r \rangle_D = \text{true}$  genau dann, wenn die folgenden Aussagen erfüllt sind.



1. Für **Kardinalitätsregeln**  $r = (s, card_{\min}, card_{\max}) \in R_{card}$ :

$$card_{\min} \leq \left| \left\{ d \in \Delta_D \mid \text{corr}_{S_D}^{S_R}(\delta_D(d)) = s \right\} \right| \leq card_{\max} \quad (6)$$

Die Anzahl der Instanzen  $d$  von semantischen Objekten im Design, deren Typ  $\delta_D(d)$  in der Designontologie mit dem in der Regel angegebenen Typ  $s$  in der Regelwissensontologie übereinstimmt, muss innerhalb des in der Regel angegebenen Bereichs liegen.

2. Für **Relationsregeln**  $r = (s_1, l, s_2, mult_{\min}, mult_{\max}) \in R_{rel}$ :

$$\forall d \in \Delta_D : \left( \text{corr}_{S_D}^{S_R}(\delta_D(d)) = s_1 \Rightarrow \left. \begin{array}{l} mult_{\min} \leq \left| \left\{ (s_{D1}, l_D, s_{D2}) \subseteq r_D \mid s_{D1} = d \wedge \text{corr}_{R_D}^{R_R}(l_D) = l \right\} \right| \\ \wedge \text{corr}_{S_D}^{S_R}(\delta_D(s_{D2})) = s_2 \end{array} \right\} \leq mult_{\max} \right) \quad (7)$$

Für alle Instanzen  $d$  von semantischen Objekten im Design, deren Typ  $\delta_D(d)$  in der Designontologie mit dem in der Regel angegebenen Typ  $s_1$  in der Regelwissensontologie übereinstimmt, muss die Anzahl der Relationstupel im Design, die von  $d$  ausgehen und deren Relationstyp mit dem in der Regel angegebenen korrespondiert und deren Zielobjekte mit  $s_2$  korrespondieren, im in der Regel angegebenen Bereich der Relationsmultiplizität liegen.

3. Für **Integer-Attributregeln**  $r = (s, i, val_{\min}, val_{\max}) \in R_{attr}$ :

$$\forall d \in \Delta_D : \left( \text{corr}_{S_D}^{S_R}(\delta_D(d)) = s \Rightarrow \left( \exists (s_x, i_x, v_x) \in i_D : (s_x = d \wedge \text{corr}_{I_D}^{I_R}(i_x) = i \wedge val_{\min} \leq v_x \leq val_{\max}) \right) \right) \quad (8)$$

Für alle Instanzen  $d$  von semantischen Objekten im Design, deren Typ  $\delta_D(d)$  in der Designontologie mit dem in der Regel angegebenen Typ  $s$  in der Regelwissensontologie übereinstimmt, muss ein Integer-Attribut-Tupel für das Objekt  $d$  existieren, dessen Attributtyp mit dem in der Regel angegebenen korrespondiert und dessen zugehöriger Wert im durch die Regel zugelassenen Bereich liegt.

## 4 Diskussion und Ausblick

In diesem Artikel wurde die operationale Semantik für einen Ausschnitt unserer visuellen Sprache zur Repräsentation von konzeptuellem Gebäudewissen definiert. Eine vollständige Betrachtung aller durch diese Sprache unterstützten Regeltypen findet sich in [4]. Dort findet sich auch eine Besprechung verwandter Arbeiten zum Thema, die hier aus Platzgründen entfällt. Die hier dargestellte Semantikdefinition bleibt dem Anwender der Werkzeuge zum konzeptuellen Entwurf verborgen. Der Anwender wird durch eine benutzerfreundliche Oberfläche beim Erstellen und Prüfen von Entwürfen unterstützt.

Die formale Semantikdefinition dient zum einen der Implementierung der Prüfmechanismen der Konsistenzanalyse und zum anderen auch dem Verständnis, in welchen Fällen und aus welchen Gründen dem Benutzer Warnmeldungen ausgegeben werden.

Zurzeit wird im Condes-Projekt daran gearbeitet, die Ausdruckskraft der visuellen Sprache dahingehend zu erweitern, dass einfache, atomare Regeln zu komplexen, zusammengesetzten Regeln verknüpft werden können. Dazu werden dem Benutzer Operatoren der Aussagen- und der Prädikatenlogik zur Verfügung gestellt. Die hier dargestellte formale Semantikdefinition wird entsprechend erweitert. Darüber hinaus wird an der Modularisierung des Regelwissens gearbeitet. Dadurch soll es ermöglicht werden, Regeln zu bestimmten Teildomänen in modularer Form zu spezifizieren und später einen Gebäudeentwurf gegen ein Gesamtregelwissen zu prüfen das aus einer Hierarchie von Modulen besteht.

## 5 Danksagung

Wir danken der Deutschen Forschungsgemeinschaft für die Förderung des Projekts im Rahmen des Schwerpunktprogramms „Vernetzt-kooperative Planungsprozesse im Konstruktiven Ingenieurbau“ (SPP 1103).

## Literatur

- [1] T. Berners-Lee, J. Hendler, O. Lassila: The Semantic Web. Scientific American, 284(5):34–43, 2001.
- [2] Condes Gruppe: Software-Werkzeuge zur Unterstützung des konzeptuellen Gebäude-Entwurfs. <http://www-i3.informatik.rwth-aachen.de/condes/> (05.03.2005), März 2005.
- [3] M. Erwig: Abstract Syntax and Semantics of Visual Languages. Journal of Visual Languages and Computing, 9(5):461–483, 1998.
- [4] B. Kraft, N. Wilhelms: Visual Knowledge Specification for Conceptual Design. In: 2005 Intl. Conf. on Computing in Civil Engineering (ICCC 2005), Seiten 1–14, 2005.
- [5] J. F. Sowa: Conceptual Structures: Information Processing in Mind and Machine. Addison-Wesley, Boston, MA, USA, 1984.