## Beispiel zur Partialität (2)

**Beachte**

spec kann man erweitern, damit $-$ „mehr definiert" ist.

(1) $0 - y = 0$ oder (2) $0 - y = y$

Bei der Erweiterung (1) wird $\mathcal{A}^C$ bis auf Isomorphie zu $(\mathbb{N}, \{0, +1, +, \dot{-}\})$.

Bei der Erweiterung (2) wird $\mathcal{A}^C$ bis auf Isomorphie zu $(\mathbb{N}, \{0, +1, +, d)$ mit $d(x, y) = |x - y|$.

In allen 3 Fällen ist $\mathcal{A}$ ein **Datenmodell** zu spec.

**Frage**: Was gilt in allen Datenmodellen von spec?

**Ziel**

Definition der Gültigkeit einer Klausel $C$ in spec, so dass die Gültigkeit unter Erweiterungen von spec erhalten bleibt.



## Beispiel negative Bedingungen

**Ansatz**

Die Gleichheit wird in $\mathcal{A}$ klassisch ausgewertet,
d. h. $\mathcal{A}$ erfüllt $t_1 = t_2 \text{ gdw } t_1^{\mathcal{A}} = t_2^{\mathcal{A}}$.

Die Ungleichheit wird „konstruktiv" ausgewertet,
d. h. $\mathcal{A}$ erfüllt $t_1 \neq t_2 \text{ gdw } t_1^{\mathcal{A}}, t_2^{\mathcal{A}} \in A^C$ und $t_1^{\mathcal{A}} \neq t_2^{\mathcal{A}}$.

**Sei** $F = \{0, s, \geq, d, \mathsf{true}, \mathsf{false}\}$, $C = \{0, s, \mathsf{true}, \mathsf{false}\}$.

$$
\begin{array}{lll}
E: & x \geq 0 = \mathsf{true} & d(x, y) = x \Leftarrow x \geq y = \mathsf{true} \\
 & s(x) \geq s(y) = x \geq y & d(x, y) = y \Leftarrow x \geq y \neq \mathsf{true}
\end{array}
$$

Sei $\mathcal{A}$ ein Modell von spec.

Es gilt $d(s^i(0), s^j(0)) = s^i(0)$ in $\mathcal{A}$ für $i \geq j$.

Für $i < j$ ist $s^i(0) \geq s^j(0)$ nicht notwendigerweise in $\mathcal{A}^C$, also muss in $\mathcal{A}$ nicht $d(s^i(0), s^j(0)) = s^j(0)$ gelten.

Zur $E$-Gleichheit:
Es gilt $s^i(0) \geq s^j(0) =_E \mathsf{true}$, also $d(s^i(0), s^j(0)) =_E s^i(0)$ für $i \geq j$.
Für $i < j$ gibt es kein $t \in \mathrm{Term}(C)$ mit $s^i(0) \geq s^j(0) =_E t$, also gilt nicht $s^i(0) \geq s^j(0) \neq_E \mathsf{true}$, also gilt nicht $d(s^i(0), s^j(0)) =_E s^j(0)$.

# Zwei Typen von Variablen

**Ansatz**

Wir unterscheiden Konstruktorvariablen und generelle Variable:
Die Konstruktorvariablen $x \in V^C$ können in $\mathcal{A}$ nur Werte aus $A^C$ annehmen.

Die generellen Variablen $x \in V^G$ können in $\mathcal{A}$ alle $a \in A$ als Wert annehmen, also auch „Fehlerelemente".

**Beachte**

In den bisherigen Beispielen waren alle Variablen Konstruktorvariablen. Also durfte man in einer Gleichung z. B. nicht $0 - s(0)$ oder auch $s(0 - s^2(0))$ für $x$ einsetzen.

**Beispiel** $F = \{0, s, -, \text{nil}, ., \text{app}\}$, $C = \{0, s, \text{nil}, .\}$.

Sei $x, y \in V^C$ und $X, L_1, L_2 \in V^G$.

$$
\begin{array}{lll}
E: & x - 0 = x & s(x) - s(y) = x - y \\
 & \text{app}(\text{nil}, L_2) = L_2 & \text{app}(X.L_1, L_2) = X.\text{app}(L_1, L_2)
\end{array}
$$

Dies definiert die Konkatenation von Listen auch für den Fall, dass sie Fehlerelemente der Form $0 - s^i(0)$, $i > 0$, als Elemente haben.



# 4.2 Syntax und Semantik von Spezifikationen

**Sei** $\text{sig} = (S, F, \alpha)$, $C \subseteq F$, $\text{sig}^C = (S, C, \alpha|_C)$.

Dann heißt sig eine **Signatur über den Konstruktoren** in $C$.

Wir schreiben auch $\text{sig} = (S, C, F, \alpha)$.

Sei $V = V^C \cup V^G$ mit $V^C = (V_s^C)_{s \in S}$ und $V^G = (V_s^G)_{s \in S}$.

Dann ist $V$ ein **Variablensystem** zu $\text{sig} = (S, C, F, \alpha)$.

Voraussetzung: $\text{Term}_s(C) \neq \varnothing$ für alle $s \in S$.

**Bezeichnung**

Eine Substitution ist $\sigma : V \longrightarrow \text{Term}(F, V)$.

$\sigma$ ist eine **Konstruktorsubstitution**, falls $\sigma(x) \in \text{Term}(C, V^C)$ für alle $x \in V^C$.

$\sigma$ ist eine **induktive Substitution**, falls $\sigma(x) \in \text{Term}(C)$ für $x \in V^C$ und $\sigma(x) \in \text{Term}(F, V^G)$ für $x \in V^G$.

$\operatorname{def}$ ist ein Prädikatssymbol mit der festen Interpretation:

$\operatorname{def}(t)$ gdw $t$ ist definiert.

Ein Atom $A$ hat also die Form $\operatorname{def}(t)$ oder $t_1 = t_2$.

Es ist $\bar{l}$ die Negation eines Literals $l$, also $\bar{A} \equiv \neg A$, $\overline{\neg A} \equiv A$.

## Spezifikationen

**Sei** sig $= (S, C, F, \alpha)$ und $V = V^C \cup V^G$.

**Bezeichnung**

Eine **bedingte Gleichung** ist eine Klausel $C$ mit einem ausgezeichneten Atom $t_1 = t_2 \in C$, also $C \equiv t_1 = t_2 \vee l_1 \vee \cdots \vee l_n$.

Schreibweise: $t_1 = t_2 \Leftarrow \bar{l}_1 \wedge \bar{l}_2 \wedge \cdots \wedge \bar{l}_n$.

**Definition 4.1**

Eine **Basisspezifikation** ist spec$^C$ = (sig$^C$, $E^C$) über $V^C$, wobei die Gleichungen aus $E^C$ nur positiv bedingt sind.

Eine **Spezifikation** über spec$^C$ ist spec = (sig, $E$) über $V$ mit $E = E^C \cup E^F$, wobei $E^F$ aus bedingten Gleichungen besteht.

**Beachte** zur Intention:

Die Elemente aus $C$ heißen Konstruktoren. $E^C$ definiert den Zusammenhang zwischen den Konstruktoren. Sie heißen **frei**, falls $E^C$ leer ist. (Dies ist in Anwendungen häufig der Fall.) $E^F$ definiert (partielle) Funktionen auf den Konstuktortermen.



## sig-Algebren

**Sei** sig $= (S, C, F, \alpha)$.

**Definition 4.2**

Eine **sig-Algebra** $\mathcal{A} = ((A_s)_{s\in S}, (f^{\mathcal{A}})_{f\in F})$ ist wie üblich definiert. Sei $t^{\mathcal{A}}$ die Auswertung von $t \in$ Term$(F)$ in $\mathcal{A}$.
Das $C$-**Redukt** $\mathcal{A}^C$ von $\mathcal{A}$ ist die sig$^C$-Algebra
$\mathcal{A}^C = ((A_s^C)_{s\in S}, (f^{\mathcal{A}^C})_{f\in C})$ mit

$A_s^C = \{t^{\mathcal{A}} \mid t \in \text{Term}_s(C)\}$ für alle $s \in S$

$f^{\mathcal{A}^C}(a_1, \ldots, a_n) = f^{\mathcal{A}}(a_1, \ldots, a_n)$ für alle $f \in C$

**Beachte**

Gilt $a_1, \ldots, a_n \in A^C$, so gibt es $t_i \in$ Term$(C)$ mit $a_i = t_i^{\mathcal{A}}$.

Ist $f \in C$, so ist $f(t_1, \ldots, t_n) \in$ Term$(C)$ und $f^{\mathcal{A}^C}(a_1, \ldots, a_n) = f(t_1, \ldots, t_n)^{\mathcal{A}} \in A^C$. Also sind $\mathcal{A}$ und $\mathcal{A}^C$ „totale Algebren", d. h., alle $f^{\mathcal{A}}$ mit $f \in F$ bzw. $f^{\mathcal{A}^C}$ mit $f \in C$ sind totale Funktionen.

**Lemma 4.1**

Seien $\mathcal{A}$ und $\mathcal{B}$ sig-Algebren und $\varphi = (\varphi_s)_{s\in S}$ ein sig-Homomorphismus $\varphi : \mathcal{A} \to \mathcal{B}$. Sei $\varphi^C = (\varphi_s^C)_{s\in S}$ die Einschränkung von $\varphi$ auf $\mathcal{A}^C$. Dann ist $\varphi^C$ ein sig$^C$-Homomorphismus $\varphi^C : \mathcal{A}^C \to \mathcal{B}^C$.

## sig-Algebren (Forts.)

Die Funktionen zu $f \in F - C$ sind als partielle Funktionen über $A^C$ erklärbar.

**Definition 4.3**

Das **erweiterte** $C$**-Redukt** $\mathcal{A}_e^C$ von $\mathcal{A}$ ist die sig-Algebra $\mathcal{A}_e^C = ((A_s^C)_{s \in S}, (f^{\mathcal{A}_e^C})_{f \in F})$ mit

$$f^{\mathcal{A}_e^C}(a_1, \ldots, a_n) = f^{\mathcal{A}^C}(a_1, \ldots, a_n) \qquad \text{für alle } f \in C$$

$$f^{\mathcal{A}_e^C}(a_1, \ldots, a_n) = \begin{cases} f^{\mathcal{A}}(a_1, \ldots, a_n) & \text{falls } f^{\mathcal{A}}(a_1, \ldots, a_n) \in A^C \\ \text{nicht definiert} & \text{sonst} \end{cases} \qquad \text{für alle } f \in F - C$$



## spec-Modelle

**Sei** spec $=$ (sig, $E$) eine Spezifikation über spec$^C$,
$\mathcal{A}$ eine sig-Algebra.

**Definition 4.4**

a) Eine **Variablen-Belegung** $\varphi = (\varphi^C, \varphi^G)$ in $\mathcal{A}$ ist eine Abbildung $\varphi : V \to A$ mit $\varphi^C : V^C \to A^C$ und $\varphi^G : V^G \to A$. Sie wird homomorph fortgesetzt zu $\varphi$ : Term$(F, V) \to A$.

b) $(\mathcal{A}, \varphi)$ erfüllt $t_1 = t_2$, falls $\varphi(t_1) = \varphi(t_2)$ in $\mathcal{A}$. $(\mathcal{A}, \varphi)$ erfüllt $t_1 \neq t_2$, falls $\varphi(t_1) \neq \varphi(t_2)$ in $\mathcal{A}$. $(\mathcal{A}, \varphi)$ erfüllt $\mathrm{def}(t)$, falls $\varphi(t) \in A^C$. $(\mathcal{A}, \varphi)$ erfüllt eine Klausel $D$, falls $(\mathcal{A}, \varphi)$ ein Literal aus $D$ erfüllt.

c) $\mathcal{A}$ **erfüllt** $D$ (oder $D$ gilt in $\mathcal{A}$), falls für jedes $\varphi$ gilt: $(\mathcal{A}, \varphi)$ erfüllt $D$. $\mathcal{A}$ erfüllt $E$, falls $\mathcal{A}$ jede bedingte Gleichung aus $E$ erfüllt. Dann heißt $\mathcal{A}$ ein **Modell** von spec.

**Schreibweise**

$\mathcal{A} \models E$, falls $\mathcal{A}$ $E$ erfüllt.

$\mathcal{K} \models E$, falls $\mathcal{A} \models E$ gilt für alle $\mathcal{A} \in \mathcal{K}$.

Mod(spec) ist die Klasse aller Modelle von spec.

## Bemerkungen

### 1. Bemerkung

Die Definition von „$\mathcal{A}$ erfüllt $D$“ legt fest, dass $\mathrm{def}$ stets als das Definiertheitsprädikat interpretiert wird.

Sie berücksichtigt auch, dass alle Gleichungen in $E$ implizit allquantifiziert sind.

### 2. Bemerkung

In Spezifikationen soll die Ungleichheit stets konstruktiv ausgewertet werden. Wir erreichen dies später dadurch, dass jede Ungleichung $t_1 \neq t_2$ durch $\mathrm{def}(t_1)$ und $\mathrm{def}(t_2)$ bewacht wird.

Gilt $t \in \mathsf{Term}(C, V^C)$, so gilt $\mathrm{def}(t)$ automatisch, kann also weggelassen werden.

### 3. Bemerkung

spec$^C$ ist eine positiv-bedingte Spezifikation, also ist $I(\mathsf{spec}^C)$ ein Modell von spec$^C$.

Mod(spec) ist nicht leer, denn das **finale Modell** $\mathcal{A}_{\mathsf{fin}}$ liegt immer in Mod(spec). Dabei ist $\mathcal{A} = \mathcal{A}_{\mathsf{fin}} = ((A_s)_{s\in S}, (f^{\mathcal{A}})_{f\in F})$ die sig-Algebra, in der jedes $A_s = \{a_s\}$ nur aus einem Element besteht. Es gilt $f^{\mathcal{A}}(a_{s_1}, \ldots, a_{s_n}) = a_s$ für $f : s_1 \ldots s_n \to s$.



## Datenmodelle

**Sei** spec eine Spezifikation über spec$^C$.

### Intention

Eine sig-Algebra $\mathcal{A} \in$ Mod(spec) ist ein Datenmodell, wenn das $C$-Redukt $\mathcal{A}^C$ aus den „Daten“ von $I(\mathsf{spec}^C)$ besteht, d. h., $\mathcal{A}^C$ isomorph zu $I(\mathsf{spec}^C)$ ist. Dazu müssen wir uns aber auf „zulässige“ Spezifikationen beschränken. Siehe später.

### Definition 4.5

$\mathcal{A} \in$ Mod(spec) heißt ein **Datenmodell**, falls für alle $t_1, t_2 \in \mathsf{Term}(C)$ gilt: $t_1^{\mathcal{A}} = t_2^{\mathcal{A}}$ impliziert $\mathsf{Mod(spec)} \models t_1 = t_2$. DMod(spec) ist die Klasse der Datenmodelle von spec.

### Lemma 4.2

Sei $\mathcal{A} \in$ Mod(spec) und $\mathcal{K}$ die Klasse der sig$^C$-Algebren $\mathcal{B}^C$ mit $\mathcal{B} \in$ Mod(spec). Dann gilt $\mathcal{A} \in$ DMod(spec) $\mathrm{gdw}$ $\mathcal{A}^C$ ist initial in $\mathcal{K}$.

Also: Gilt $\mathcal{A}, \mathcal{B} \in$ DMod(spec), so sind $\mathcal{A}^C$ und $\mathcal{B}^C$ sig$^C$-isomorph.

## Zulässige Spezifikationen (1)

**Intention**

Wir benötigen zusätzliche Bedingungen an Spezifikationen, so dass

a) man in spec effektiv rechnen kann,
b) der Begriff „Datenmodell“ stets sinnvoll ist, z. B. DMod(spec) nicht leer ist.

**Ansatz**

a) $E$ ist als (i. Allg. konvergentes) Regelsystem $R$ gegeben.
b) spec ist „konservative“ Erweiterung von spec$^C$.

**Definition 4.6**

spec ist eine **konservative Erweiterung** von spec$^C$, falls für alle $t_1, t_2 \in \mathsf{Term}(C)$ gilt: $t_1 =_E t_2 \text{ gdw } t_1 =_{E^C} t_2$.

**Beispiel** $F = \{c_1, c_2, d\}$, $C = \{c_1, c_2\}$, $E^C = \varnothing$.

(1) $d = c_1$ (2) $d = c_2$

Besteht $E$ nur aus (1), so ist spec $=$ (sig, $E$) eine konservative Erweiterung von spec$^C$. Dies gilt nicht, wenn $E$ aus (1) und (2) besteht.



## Zulässige Spezifikationen (2)

**Sei** sig $= (S, C, F, \alpha)$.

**Definition 4.7**

Eine **Konstruktorregel** ist $l \to r \Leftarrow u = v$ mit

- $l, r, u_i, v_i \in \mathsf{Term}(C, V^C)$.
- $\mathsf{Var}(r) \subseteq \mathsf{Var}(l)$ und $\mathsf{Var}(u_i, v_i) \subseteq \mathsf{Var}(l)$.

Eine **definierende Regel** ist $l \to r \Leftarrow \Delta$ mit

- $l \in \mathsf{Term}(F, V) - \mathsf{Term}(C, V)$.
- $\Delta$ ist Konjunktion von Literalen nicht von der Form $\neg\mathrm{def}(t)$.

**Also**

Eine Konstruktorregel ist nur positiv bedingt.

Eine definierende Regel ist positiv/negativ bedingt. Es gibt keine Bedingung der Form $\neg\mathrm{def}(t)$. Die linke Seite $l$ enthält ein Funktionssymbol $f \in F - C$.

## Zulässige Spezifikationen (3)

**Sei** $\text{sig} = (S, C, F, \alpha)$.

**Definition 4.8**

Eine **zulässige Spezifikation** ist $\text{spec} = (\text{sig}, R)$ mit

a) $R = R^C \cup R^D$, wobei $R^C$ eine Menge von Konstruktorregeln und $R^D$ eine Menge von definierenden Regeln ist.

b) $\rightarrow_R$ ist konfluent (zur Definition von $\rightarrow_R$ siehe unten).

c) Für jede Regel $l \rightarrow r \Leftarrow \Delta$ gilt: Ist $t_1 \neq t_2$ in $\Delta$, $i \in \{1, 2\}$ und $t_i \notin \text{Term}(C, V^C)$, so ist $\text{def}(t_i)$ in $\Delta$.

**Definition 4.9**

Sei $R = R^C \cup R^D$. (Beachte: $t_1 \neq t_2$ steht für $\neg t_1 = t_2$).

a) Es ist $\rightarrow_{R^C}$ wie üblich erklärt:
$t_1 \rightarrow_{R^C} t_2 \text{ gdw}$ es gibt $l \rightarrow r \Leftarrow u = v$ in $R_C$ und $\sigma$, so dass $t_1 \rightarrow t_2$ mit $l \rightarrow r$, $\sigma$ und $\sigma(u) \downarrow_{R^C} \sigma(v)$.

b) Zur Definition $\rightarrow_R$:
$\sigma$ erfüllt $t_1 = t_2 \text{ gdw } \sigma(t_1) \downarrow_R \sigma(t_2)$.
$\sigma$ erfüllt $\text{def}(t) \text{ gdw } \sigma(t) \xrightarrow{*}_R t_0$ mit $t_0 \in \text{Term}(C, V^C)$.
$\sigma$ erfüllt $\Delta \text{ gdw } \sigma$ erfüllt jedes Literal aus $\Delta$.
$t_1 \rightarrow_R t_2 \text{ gdw}$ i) $t_1 \rightarrow_{R^C} t_2$ oder
ii) es gibt $l \rightarrow r \Leftarrow \Delta$ in $R^D$ mit $t_1 \rightarrow t_2$ mit $l \rightarrow r, \sigma$ und $\sigma$ erfüllt $\Delta$.



## Bemerkungen

**1. Bemerkung**

$\rightarrow_R$ wird in zwei Schichten definiert:

1. Schicht: $\rightarrow_{R^C}$ benutzt nur $R^C$-Regeln.

2. Schicht: $\rightarrow_R$ benutzt alle $R$-Regeln.

**2. Bemerkung**

Die Definitionen von $\rightarrow_{R^C}$ und $\rightarrow_R$ sind rekursiv.

Also: $\rightarrow_{R^C}$ und $\rightarrow_R$ sind die kleinsten Relationen, die die Bedingungen der Definitionen erfüllen.

**3. Bemerkung**

Ist $l \rightarrow r \Leftarrow \Delta$ in $R^D$, so enthält $l$ ein $f \in F - C$. Die Regeln aus $R^D$ sind also nicht auf Terme $t \in \text{Term}(C, V^C)$ anwendbar.

Also gilt in einer zulässigen Spezifikation für alle $t_1, t_2 \in \text{Term}(C, V^C)$:

$$t_1 =_R t_2 \ \text{ gdw } \ t_1 \downarrow_R t_2 \ \text{ gdw } \ t_1 \downarrow_{R^C} t_2 \ \text{ gdw } \ t_1 =_{R^C} t_2$$

**Lemma 4.3**

Ist spec eine zulässige Spezifikation, so ist spec eine konservative Erweiterung zu $\text{spec}^C$.

## Beispiel

**Sei** $C = \{0, s, p\}$ $\quad D = \{f, \mathsf{half}\}$ $\quad F = C \cup D$
$R = R^C \cup R^D$

$R^C$: $s(p(x)) \to x \quad p(s(x)) \to x$

$R^D$: $\mathsf{half}(0) \to 0$
$\mathsf{half}(s(s(x))) \to s(\mathsf{half}(x))$
$\mathsf{half}(p(p(x))) \to p(\mathsf{half}(x))$
$s(\mathsf{half}(p(x))) \to \mathsf{half}(s(x))$
$p(\mathsf{half}(s(x))) \to \mathsf{half}(p(x))$
$f(x) \to 0 \Leftarrow \mathsf{half}(x) \neq x, \mathrm{def}(\mathsf{half}(x))$

**Dann** gilt:

a) Die Spezifikation ist zulässig.

b) Jedes Modell $\mathcal{A} = (\mathbb{Z}, F^{\mathcal{A}})$ aus Mod(spec) mit
$0^{\mathcal{A}} = 0 \qquad s^{\mathcal{A}}(x) = x + 1 \qquad p^{\mathcal{A}}(x) = x - 1$
ist ein Datenmodell. Beispiele dazu sind
$\mathsf{half}^{\mathcal{A}}(2x) = x \quad \mathsf{half}^{\mathcal{A}}(2x+1) = x + 1$
$f^{\mathcal{A}}(x) = 0$ und
$\mathsf{half}^{\mathcal{A}}(2x) = x \quad \mathsf{half}^{\mathcal{A}}(2x+1) = \uparrow$
$f^{\mathcal{A}}(2x) = 0 \quad f^{\mathcal{A}}(2x+1) = \uparrow$

c) Die Algebra $\mathcal{A} = (\mathbb{Z}, F^{\mathcal{A}})$ mit
$0^{\mathcal{A}} = 0 \qquad s^{\mathcal{A}}(x) = p^{\mathcal{A}}(x) = \mathsf{half}^{\mathcal{A}}(x) = f^{\mathcal{A}}(x) = x$
ist ein Modell von spec, aber kein Datenmodell. Es gilt
z. B. $\mathcal{A} \models s(0) = 0$, aber nicht Mod(spec) $\models s(0) = 0$.



## Das Standard-Datenmodell

**Sei** sig $= (S, C, F, \alpha)$,
spec $=$ (sig, $R$) eine zulässige Spezifikation.

**Definition 4.10**

Das **Standard-Datenmodell** zu spec ist die sig-Algebra
$\mathcal{M} = \mathcal{M}(\text{spec})$ mit

$$M_s = \{[t] \mid t \in \mathrm{Term}(F)\} \qquad \text{für } s \in S$$
$$f^{\mathcal{M}}([t_1], \ldots, [t_n]) = [f(t_1, \ldots, t_n)] \qquad \text{für } f \in F$$

Dabei ist $[t] = \{t' \mid t =_R t'\}$ die $R$-Äquivalenzklasse zu $t$.

**Beachte**

Ist $R$ konvergent, so ist $\mathcal{M}$(spec) isomorph zur kanonischen Termalgebra mit den $R$-irreduziblen Grundtermen als Elementen.

**Lemma 4.4**

a) $\mathcal{M} = \mathcal{M}$(spec) ist ein Datenmodell.

b) Ist $\mathcal{A} \in$ DMod(spec), so sind $\mathcal{A}^C$ und $\mathcal{M}^C$ isomorph.

c) $\mathcal{M}^C$ hat die $R^C$-Äquivalenzklassen der $t \in \mathrm{Term}(C)$ als Elemente.