Konservative Erweiterung in Isabelle

- *Isabelle*: Konstruktion von Theorien durch *konservative Erweiterungen*
- Konstantendefinition (zur Signatur Σ):

  $$c :: \sigma \qquad c \equiv t$$

  - $c \notin \Sigma$
  - $t \in T_\Sigma$ (enthält *nicht* c)
  - Typvariablen in t auch in σ
- *Lemma: Konstantendefinition* ist *konservative* Erweiterung
- Weitere konservative Erweiterungen: *Typdefinitionen*, *Datentypen*.

Zusammenfassung

Logik höherer Stufe (HOL)

- Syntax basiert auf dem *einfach getypten λ-Kalkül*
- *Drei* Basis-Operatoren, *Acht* Basis-Axiome
- *Rest* folgt durch konservative Erweiterung

Vorlesung vom 24.11.05: Grundlagen von Isabelle

Fahrplan

- *Isabelle*: *Systemarchitektur*
- *Beweis* in Isabelle: *Unifikation* und *Resolution*
- *Typdefinitionen*
- Nützliche *vordefinierte Datentypen* in Isabelle

Isabelle Systemarchitektur: LCF-Design

- *Isabelle*: ca. 150 Kloc SML, ca. 310 Kloc Beweise — *Korrektheit*?
- *Reduktion* des Problems:
  - Korrektheit eines *logischen Kerns*
  - Rest durch *Typisierung*
- *Abstrakter Datentyp* `thm`, *Inferenz-Regeln* als *Operationen*

```
val assume: cterm -> thm
val implies_intr: cterm -> thm -> thm
```

- *Logischer Kern*:
  - *Typcheck*, *Signaturen*, *Unifikation*, *Meta-Logik*: ca. 5500 LOC

Variablen

- *Meta-Variablen*: können *unifiziert* und beliebig *instantiiert* werden
- *freie Variablen* (fixed): *beliebig* aber *fest*
- *Gebundene* Variablen: Name beliebig (α-Äquivalenz)
- *Meta-Quantoren*: Isabelles *Eigenvariablen*

  $$\frac{\bigwedge x.P(x)}{\forall\, x.P(x)}\,allI$$ `!!x. P x ==> ALL x. P x`

  - Beliebig *instantiierbar*
  - *Gültigkeit* auf diese (Teil)-Formel *begrenzt*

Formeln

- *Formeln* in Isabelle:

  $$\frac{\phi_1, \ldots, \phi_n}{\psi} \qquad [\![\phi_1, \ldots, \phi_n]\!] \Longrightarrow \psi$$

  - $\phi_1, \ldots, \phi_n$ Formeln, ψ atomar
  - *Theoreme*: *ableitbare* Formeln
  - *Ableitung* von Formeln: *Resolution*, *Instantiierung*, *Gleichheit*
- *Randbemerkung*:
  - $\Longrightarrow$, $\bigwedge$, $\equiv$ formen *Meta-Logik*
  - Einbettung *anderer* Logiken als HOL möglich — *generischer* Theorembeweiser

Resolution

- Einfache Resolution: *vorwärts* (`THEN`), oder *rückwärts* (`rule`)
  - *Achtung: Lifting* von *Meta-Quantoren* und *Bedingungen*
  - Randbemerkung: Unifikation höherer Stufe *unentscheidbar*
- *Eliminationsresolution* (`erule`)
- *Destruktionsresolution* (`drule`)

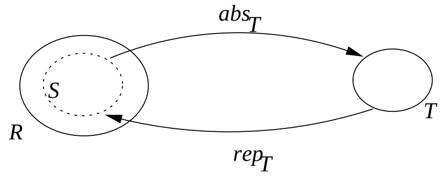
Vorwärts oder Rückwärts?

- *Vorwärtsbeweis:*
  - Modifikation von Theoremen mit *Attributen*:
  - Resolution (`THEN`), Instantiierung (`WHERE`)
- *Rückwärtsbeweis:*
  - Ausgehend von *Beweisziel* ψ
  - *Beweiszustand* ist $[\![\phi_1, \ldots, \phi_n]\!] \Longrightarrow \psi$
  - $\phi_1, \ldots, \phi_n$: *subgoals*
  - Beweisverfahren: *Resolution*, *Termersetzung*, *Beweissuche*

Typdefinitionen in Isabelle

Definition eines neuen *Typen*:

- Gegeben Typ R, Prädikat $S : R \Rightarrow Bool$
- Neuer Typ T mit $abs_T : R \Rightarrow T, rep_T : T \Rightarrow R$ so dass
  - S und T *isomorph:*
    $\forall\, t. abs_T(rep_T t) = t, \forall\, r. Sr \longrightarrow rep_T(abs_T r) = r$
  - T *nicht leer*: $\exists\, t. St$



- *Lemma:* Typdefinitionen sind *konservative Erweiterungen*.

Beispiel: Produkte

- *Ausgangstyp* $R_{\alpha,\beta} \equiv \alpha \Rightarrow \beta \Rightarrow Bool$
- *Neuer Typ* $T_{\alpha,\beta} \equiv \alpha \times \beta$
  - *Idee*: Prädikat $p : \alpha \Rightarrow \beta \Rightarrow Bool$ repräsentiert (a, b)
    falls $p(x, y) = true \longleftrightarrow x = a \wedge y = b$
- $S = \lambda f.\, \exists\, a\, b. f = \lambda\, x\, y. x = a \wedge y = b$
- *Abstraktionsfunktion*: $abs_\times p = \iota a\, b. p\, a\, b$
- *Repräsentationsfunktion*: $rep_\times(a, b) = p$ mit
  $p(x, y) = true \longleftrightarrow x = a \wedge y = b$