

Spezifikations- und Selektionsmethoden für Daten und Dienste

XQuery Semantik

Yukari Ishii



Motivation

- Formale Semantik für **XQuery 1.0** und **XPath 2.0** wurde von der **XML Query Working Group** und der **XSL Working Group** definiert
- stellt Ergänzung zur **XQuery** Spezifikation dar
- Ziel: Definition der Bedeutung von **XQuery** Ausdrücken
- **Statische Semantik:** Setzt XQuery Ausdrücke in Bezug zu dem zugrunde gelegten XML Schema Typ
- **Dynamische Semantik:** Setzt XQuery Ausdruck in Bezug zu dem Wert, der sich bei der Auswertung ergibt



Gliederung

- Wichtige Begriffe
 - Judgments
 - Inference Rules
- Logisches Verarbeitungsmodell
 - Phase 1 Static Analysis
 - Phase 2 Dynamic evaluation
- Zusammenfassung



Wichtige Begriffe

Judgments

Judgment

- Grundbaustein der formalen Spezifikation
- drückt aus, ob eine Eigenschaft gilt oder nicht

Beispiele:

1. $Expr \Rightarrow Value$

- Lies: Ausdruck $Expr$ ergibt Wert $Value$
- gilt, wenn Auswertung des Ausdrucks $Expr$ den Wert $Value$ ergibt

2. $Expr : Type$

- Lies: Ausdruck $Expr$ ist vom Typ $Type$
- gilt, wenn Ausdruck $Expr$ vom Typ $Type$ ist



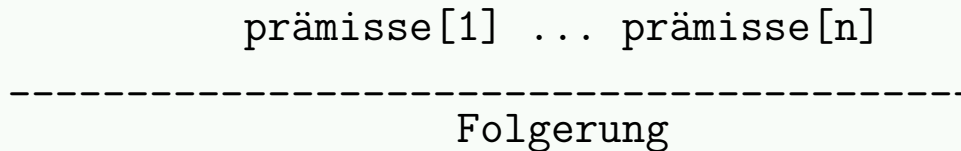
Wichtige Begriffe

Inference Rules (1)

Inference rule

- Sammlung von 0 oder mehr **Prämissen** und einer **Folgerung**
- beschreibt, wie **complex judgments** aus einfacheren **premise judgments** gefolgert werden können

Notation



Interpretation: Gelten alle Prämissen, so gilt auch die Folgerung.



Wichtige Begriffe

Inference Rules (2)

Beispiel 1:

$$\begin{array}{ccc} \$x \Rightarrow 0 & & 3 \Rightarrow 3 \\ \hline & & \$x + 3 \Rightarrow 3 \end{array}$$

Beispiel 2:

$$\begin{array}{ccc} \hline & & 3 \Rightarrow 3 \end{array}$$



Logisches Verarbeitungsmodell

Phase 1 Static Analysis

1. Parsing
2. Static Context Processing
3. Normalization
4. Static Type Analysis

Phase 2 Dynamic Evaluation

1. Dynamic Context Processing
2. Dynamic Evaluation



Phase 1 Static Analysis

Static Analysis:

- Unterteilt in 4 Teilschritte
- Ziel: Erkennung von Fehlern (z.B. Syntaxfehlern, Typfehlern) zur Compilierungszeit



Phase 1 Static Analysis

Überblick

1. Parsing:

- Generierung eines **internal operation tree**
- formale Semantik definiert kein formales Modell für Syntaxbäume, sondern lehnt sich an XQuery Syntax an

2. Static Context Processing: Generieren des statischen Eingabekontextes

3. Normalization:

- Abbildung von **XQuery Ausdrücken/Anfragen** auf **Ausdrücke/Anfragen in XQuery Core**

4. Static Type Analysis:

- prüft für jeden Ausdruck/jede Anfrage, ob sie typsicher ist



Phase 1 Static Analysis

Normalization

Normalization:

- Ziel: Abbilden von **XQuery Ausdrücken/Anfragen** auf **Ausdrücke/Anfragen in XQuery Core**
- XQuery Core Grammatik ist Teilmenge der XQuery Grammatik
- Abbildung erfolgt mittels **mapping rules**, die beschreiben, wie ein gegebener XQuery Ausdruck als XQuery Core Ausdruck geschrieben wird



Phase 1 Static Analysis

Static Type Analysis (1)

Static Type Analysis

- Jeder Ausdruck/jede Anfrage wird daraufhin überprüft, ob sie typsicher ist
- ist dies nicht der Fall, ergibt sich ein Typfehler
 - Beispiel: Ausdruck, der String mit Integer vergleicht
- Fehler, die sich erst während der Auswertung ergeben, werden hier nicht erkannt
 - Beispiel: Überlauf bei arithmetischer Berechnung auf 32bit integer `xs:integer`



Phase 1 Static Analysis

Static Type Analysis (2)

- Analyse erfolgt durch bottom-up Anwendung von **type inference rules** auf die Teilausdrücke eines betrachteten Ausdrucks
- somit hängt der Typ eines Ausdrucks nur vom Typ der Teilausdrücke ab
- erfolgreicher Abschluß der **static type analysis** gibt einen abstrakten Syntaxbaum aus, wo jeder Teilausdruck eine statische Typannotation besitzt



Phase 2 Dynamic Evaluation

Überblick

Phase 2 Dynamic Evaluation:

1. Dynamic Context Processing: Generieren des dynamischen Eingabekontextes, der für den dynamischen Auswertungsschritt (Dynamic Evaluation) benötigt wird
2. Dynamic Evaluation: Berechnet Wert für Ausdruck/Anfrage



Phase 2 Dynamic Evaluation

Teilschritt: Dynamic Evaluation (1)

Dynamic Evaluation:

- Berechnet Wert für Ausdruck/Anfrage
- Auswertungssemantik ist nur definiert für Core Ausdrücke/Anfragen
- Auswertung erfolgt mittels Auswertungsregeln (evaluation rules), welche bottom-up auf einen Ausdruck angewendet werden, angefangen bei den Literalen und Variablen
- Ergebnis: Wert oder dynamischer Fehler



Phase 2 Dynamic Evaluation

Teilschritt: Dynamic Evaluation (2)

Beispiel:

```
let $v := 3 return $v+5 ergibt als Wert 8
```

Herleitung:

- Literale 3 und 5 haben als Wert 3 bzw. 5
- Variable \$v hat als Wert 3
- $3+5=8$



Zusammenfassung

- Die formale Semantik hat als Ziel die Definition der Bedeutung von XQuery Ausdrücken
- sie orientiert sich dabei an einem in 2 Phasen untergliederten logischen Verarbeitungsmodell
- Implementierungen müssen nicht der vorgegebenen Reihenfolge folgen, Phasen können vermischt oder parallel ausgeführt werden

