



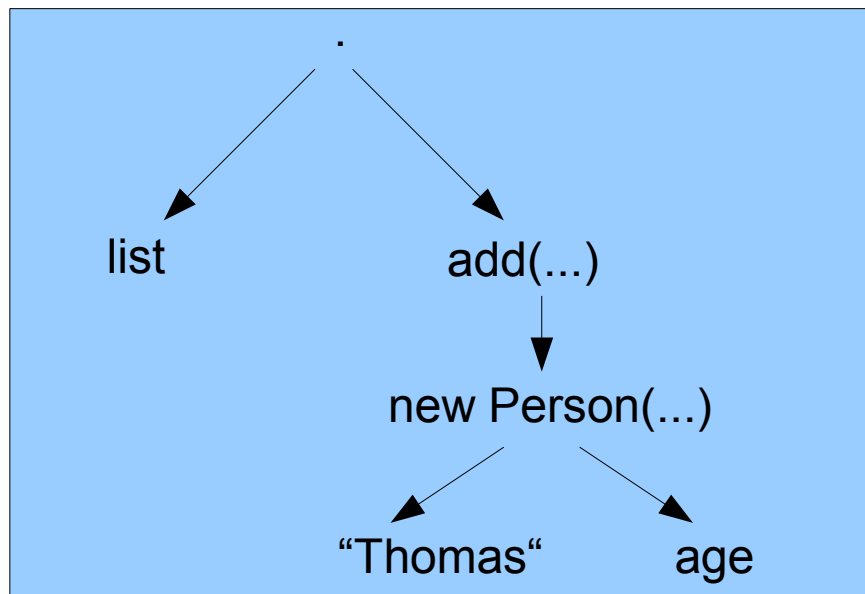
## SWP Übersetzerbau Sommer 2010

Markus Rudolph  
Institut für Informatik  
FU Berlin

- Ausdrücke abstrahieren
  - Untermenge des abstrakten Syntaxbaums
  - Dient zur weiteren Verarbeitung der Ausdrücke
- Parsen von Ausdrücken
  - Übersetzt betreffende Attributwerte aus dem DOM zu einem Teil des Syntaxbaumes
- Annotator
  - Traversiert über den aus dem DOM generierten Syntaxbaum
  - Prüft dabei die semantische Korrektheit des Programm

- Ziel: Abstraktion von Ausdrücken
- Findet Anwendung bei Wertzuweisungen und Funktionsaufrufen
- Beispiel: `<call exec="list.add(new Person('Thomas', age))"/>`

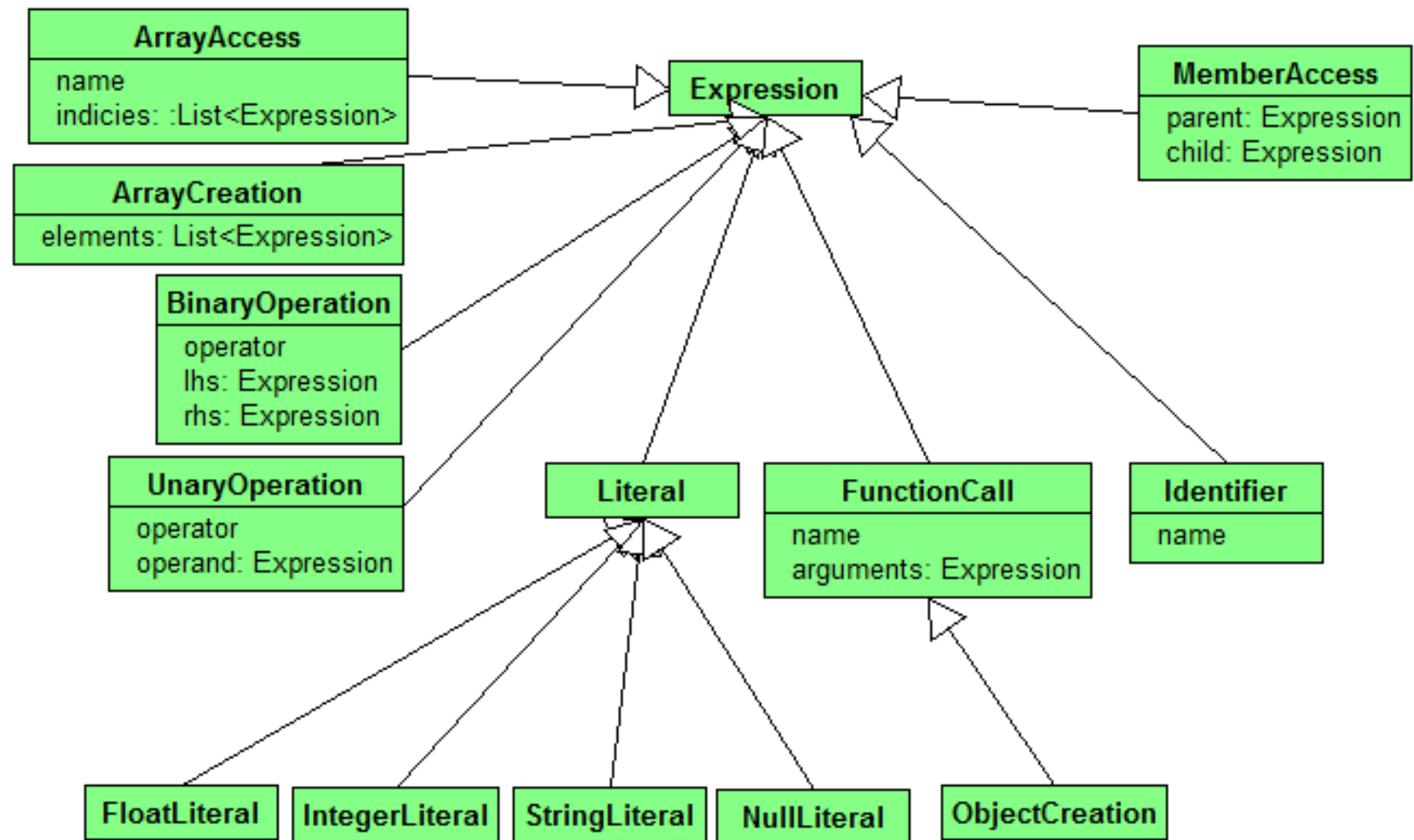
CallStatement



Statements

Ausdrücke

- Probleme:
  - Darstellung von Mitgliedsketten, wie  
`list.get(entryIndex).array[fieldIndex].execute()...`
- Lösung:
  - Eine Baumstruktur, die auch syntaktisch, ungültige Ausdrücke zulässt, wie „(a+1).(2+3)“
  - Der Erkennungsprozess (Parsen) baut aber nur syntaktisch korrekte Ausdrücke auf.
  - Alternative war eine Datenstruktur, welche auch nur die erlaubten Typen zuließ.
    - War in der Bedienung aber unzumutbar.



- Aufgaben
  - Umwandlung eines Ausdrucks in Form einer Zeichenkette zu einem Teilbaum im Syntaxbaum
  - Überprüfung auf syntaktische Korrektheit
  - Ggf. Fehlerausgabe
- Problem:
  - Umsetzung von Operatorprioritäten

```
<call exec="list.add(new Person('Thomas', age))"/>
```

der zu parsene Ausdruck

- Lösung:
  - Operatoren sind in einzelne Teilprozesse (Methoden) aufgegliedert
  - Aufrufreihenfolge entscheidet über die Priorität
  - Besser zu wartende und erweiterbare Lösung wäre möglich
- Weiteres Problem:
  - Der Parser muss verschiedene Arten von Ausdrücken erkennen können.
  - z.B. nur Funktionsaufrufe für das CallStatement oder L- und R-Werte bei Zuweisungen

```
<call exec="list.add(x)"/>
```

```
<set name="arr[index]" value="12 + offset"/>
```

- Lösung:
  - Dem Parser kann mitgeteilt werden, welche Art von Ausdruck geparkt werden soll.
  - Intern wird entsprechend ein anderer Ablauf zum Parsen gewählt...
  - ...und die Ausdrucksart am Ende des Parseprozess überprüft.



- Aufgabe
  - Bezeichner in Symboltabelle eintragen
  - Zusatzinformationen an den Syntaxbaum hängen, um Semantik zu prüfen
  - Ggf. Fehlerausgabe
- Problem
  - Syntaxbaum enthält Variablendeklarationen und -zugriffe
  - Aber sind diese Deklarationen gültig und in der richtigen Reihenfolge?

- Lösung
  - Durchlauf des gesamten Baumes um alle Typen und Variablen zu kennen.
  - Dabei werden alle Stellen mit Ausdrücken erfasst.
  - Weiterer Durchlauf aller Ausdrücke um sicherzugehen, dass die Variablen korrekt verwendet werden.

- Weitere Probleme:
  - Wie das Annotieren vom Syntaxbaum trennen?
    - Lösung: Jede Knotenart des Baumes hat im Annotator eine eigene Methode. Knoten erhalten eine Verbindung zur Symboltabelle.
    - Alternative wäre ein weiterer (annotierter) Syntaxbaum gewesen, der über die Knoten hätte traversiert werden können.

Danke!