

# Ein System für die Konstruktion objektorientierter Übersetzer

Präsentation des Promotionsthemas von



Dipl.-Inform. Peter Knauber

FB Informatik, AG Programmiersprachen und Übersetzer

## Inhalts-Übersicht

- Motivation, Anforderungen
- ❑ Traditionelle Verfahren, objektorientierte Variante
- ❑ Objektorientierte Bibliothek
- ❑ Das OCC-System
- ❑ Zusammenfassung
- ❑ Fragen, Vertiefung ...

## Ziele der Arbeit

Entwicklung eines Systems zur Unterstützung ...

- ... bei der Entwicklung neuer Sprachen
  - Sprachdefinition
  - Referenz-Implementierung
- ... bei der Portierung existierender Sprachen/Übersetzer

## Anforderungen ...

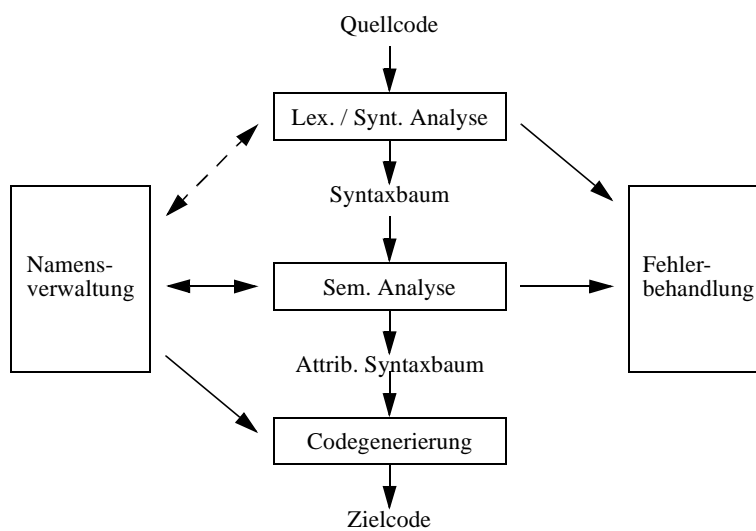
... an ein System zur Unterstützung bei der Sprachentwicklung:

- Unterstützen bei der Wiederverwendung bereits realisierter Konstrukte
- Freies Kombinieren von Sprachkonzepten
- Einfaches Integrieren neuer Sprachkonzepte
- Unterstützen bei der Konstruktion eines evtl. Zwischencodes
- Semantikdefinition auf hohem Abstraktionsniveau

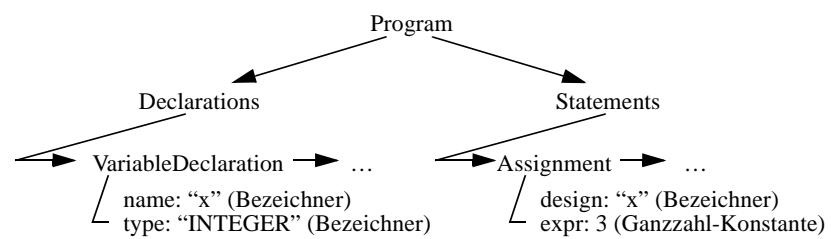
# Inhalts-Übersicht

- ❑ Motivation, Anforderungen
- ❑ Traditionelle Verfahren, objektorientierte Variante
- ❑ Objektorientierte Bibliothek
- Das OCC-System
- ❑ Zusammenfassung
- ❑ Fragen, Vertiefung ...

# Traditionelle Übersetzer



## Beispiel für einen Syntaxbaum



PROGRAM ...;

VAR x : INTEGER;

... x := 3;

END.

# Knoten im Syntaxbaum

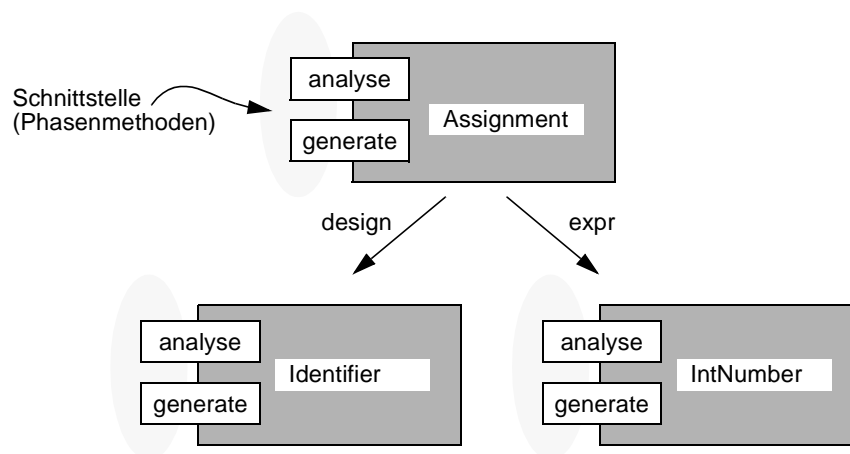
imperativ (C)

```
struct node {  
    int tag;  
    union {  
        ...  
    }  
}
```

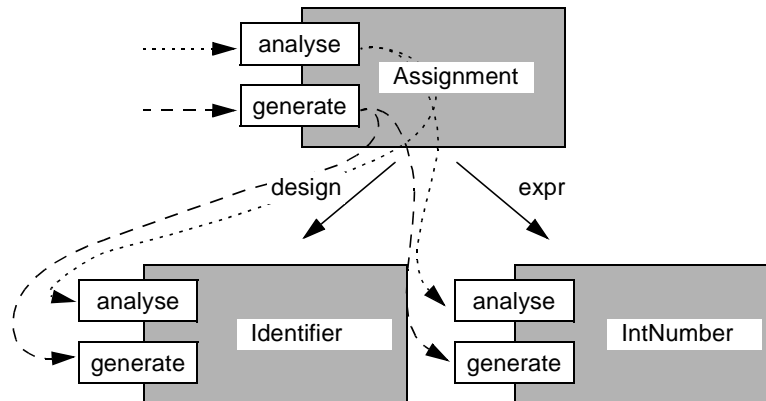
objektorientiert (W-Lisp)

```
CLASS Node IS  
    ...  
END-CLASS  
  
CLASS Assignment INHERIT Node IS  
    design  
    expr  
END-CLASS
```

## Idee der objektorientierten Übersetzung: Beziehungen



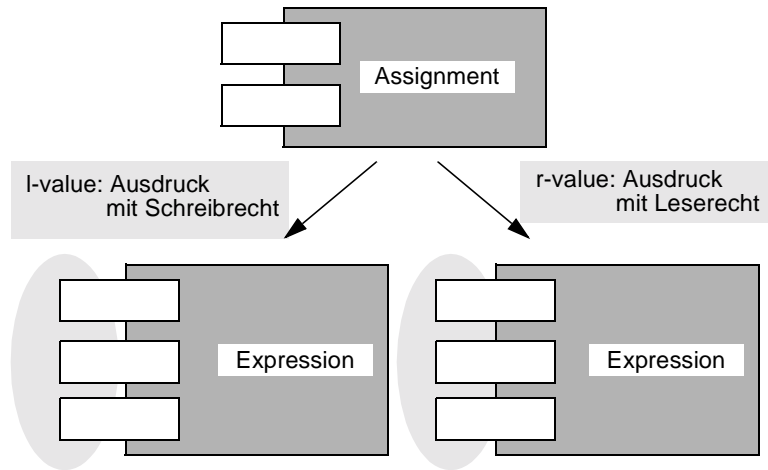
## Idee der objektorientierten Übersetzung: Ablauf



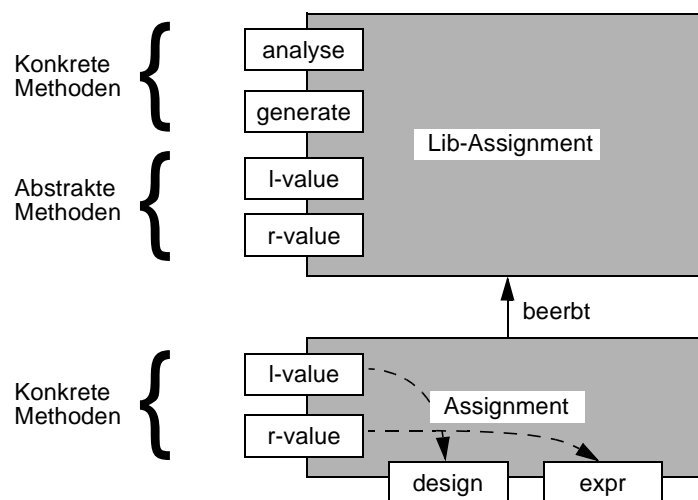
## Inhalts-Übersicht

- Motivation, Anforderungen
- Traditionelle Verfahren, objektorientierte Variante
- Objektorientierte Bibliothek
- Das OCC-System
- Zusammenfassung
- Fragen, Vertiefung ...

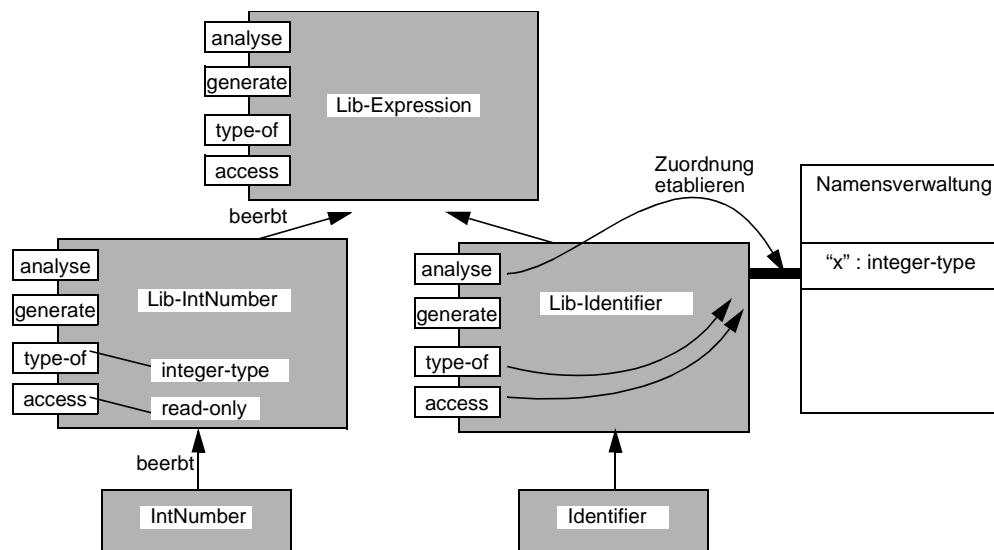
# Abstraktion von einer konkreten Sprache



# Objektorientierte Bibliothek mit Sprachkonstrukten: Zuweisung



# Objektorientierte Bibliothek mit Sprachkonstrukten: Ausdruck



## Inhalts-Übersicht

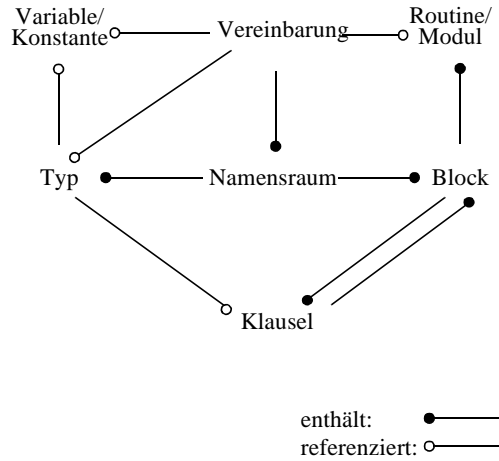
- Motivation, Anforderungen
- Traditionelle Verfahren, objektorientierte Variante
- Objektorientierte Bibliothek
- Das OCC-System
- Zusammenfassung
- Fragen, Vertiefung ...



# Grundlagen: Konzepte imperativer/objektorientierter Sprachen

Untersuchte Sprachen:

- FORTRAN, ALGOL, LISP, ...
- C, ANSI C, C++, Java
- Pascal, Modula-2, Oberon-2

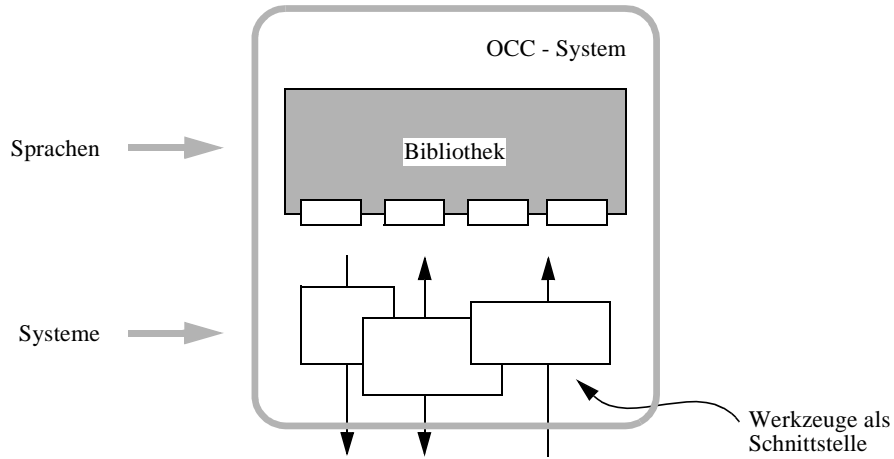


# Grundlagen: Existierende Systeme für den Übersetzerbau

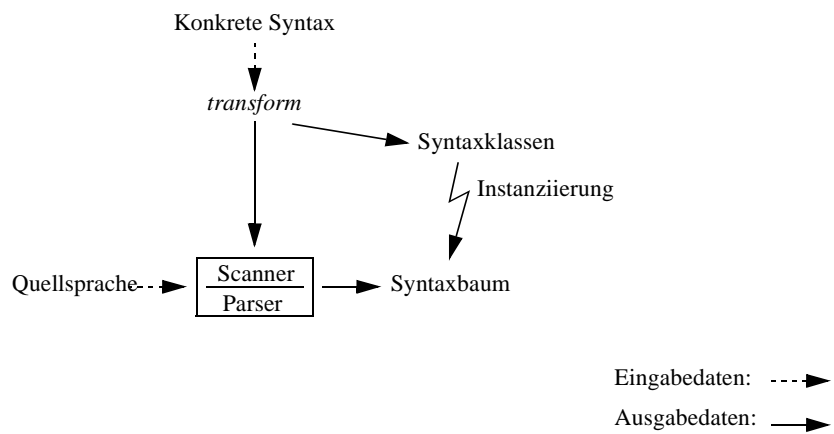
- |   |   |
|---|---|
| <input type="checkbox"/> lex, flex, yacc, bison | <input type="checkbox"/> LDL                |
| <input type="checkbox"/> SIC                    | <input type="checkbox"/> OCS                |
| <input type="checkbox"/> ACK                    | <hr style="width: 100px; margin-left: 0;"/> |
| <input type="checkbox"/> Cocktail               | <input type="checkbox"/> B. Meek            |
| <input type="checkbox"/> Eli                    | <input type="checkbox"/> J. Holmes          |
|   | <input type="checkbox"/> ...                |

# Die Entstehung des OCC-Systems

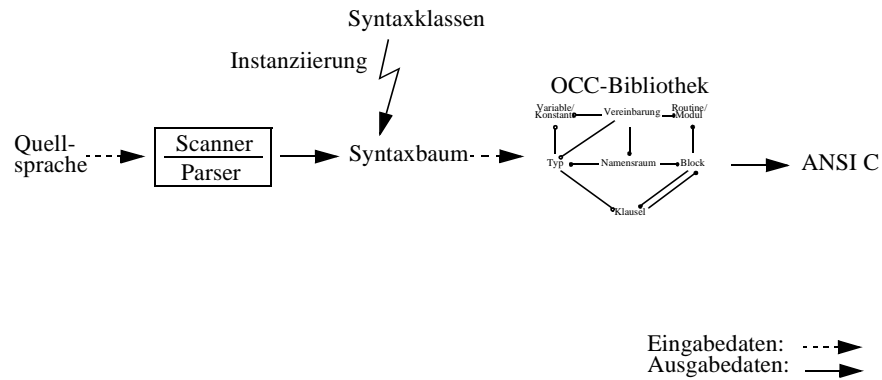
(OCC = Objectoriented Compiler Construction)



# Erzeugen eines objektorientierten Übersetzers



# Objektorientierte Übersetzer zur Laufzeit



## Zuordnung von Sprachkonstrukten zu Bibliotheksklassen

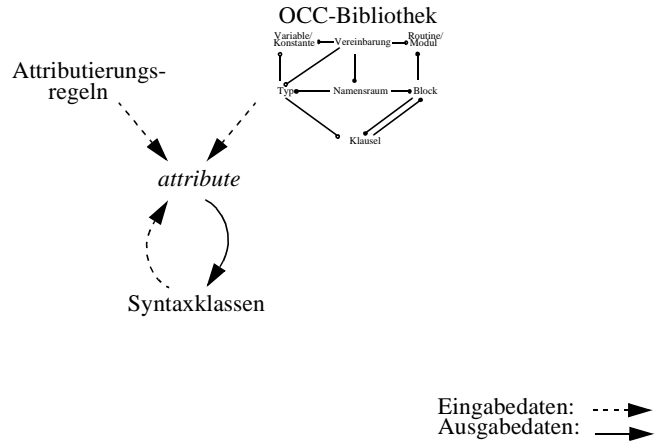
### □ Konkrete Syntax

- Assignment = Designator “:=” Expression.
- Designator = Ident ... .
- Ident = letter ... .

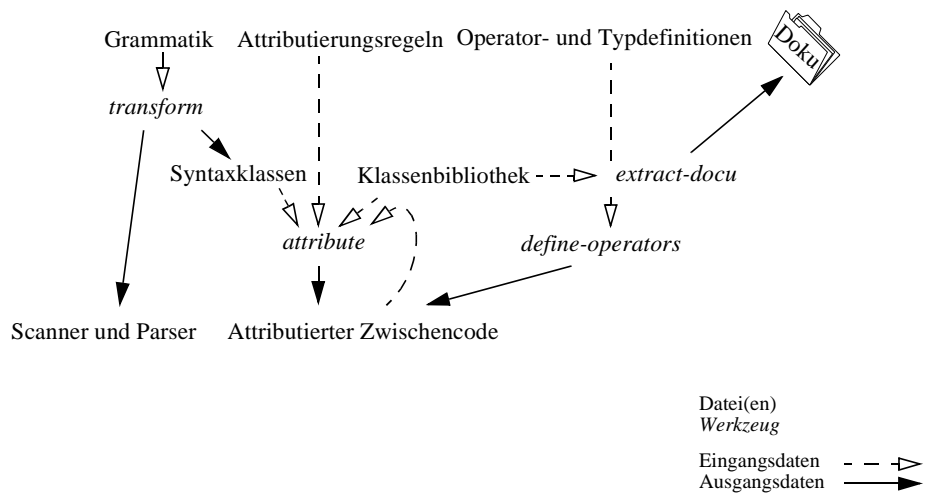
### □ Zuordnung

- Assignment IS-A Lib-Assignment.
- Designator IS-A Lib-Expression.
- Ident IS-A Lib-Identifizier, Lib-Terminal.

# Zuordnungsablauf



# Das OCC-System



## Inhalts-Übersicht

- ❑ Motivation, Anforderungen
- ❑ Traditionelle Verfahren, objektorientierte Variante
- ❑ Objektorientierte Bibliothek
- ❑ Das OCC-System
- Zusammenfassung
- ❑ Fragen, Vertiefung ...

## Daten der prototypischen Implementierung

- ❑ Anzahl der Klassen: 361
  - ❑ Methoden pro Klasse: max. 26; Ø 4,8
  - ❑ Attribute pro Klasse: max. 10; Ø 1,0
- ❑ Nachrichten (versch. Methoden): 130
- ❑ Als Hypertext aufgebaute Dokumentation
- ❑ Implementierungssprache: W-Lisp

## Erfüllung der Anforderungen (nicht lex/yacc, SIC, ACK)

Anforderungen		<i>Cocktail</i>	<i>Eli</i>	<i>LDL</i>	<i>OCS</i>	<i>OCC</i>
Verwenden existierender Konstrukte ...	... möglich	(✓) <sup>a</sup>	✓	✓	✓	✓
	... unterstützt	—	✓	(✓) <sup>b</sup>	—	✓
Austauschbarkeit von Konstrukten ...	... möglich	—	✓	✓	✓	✓
	... unterstützt	—	—	✓	—	✓
Integrieren neuer Konstrukte ...	... möglich	✓	✓	(✓) <sup>c</sup>	✓	✓
	... unterstützt	—	—	—	—	✓
Unterstützen bei der Zwischencode-Konstruktion		✓	—	entfällt	—	✓
Semantikdefinition auf hohem Niveau		✓ <sup>d</sup>	✓ <sup>d</sup>	✓ <sup>d</sup>	(✓) <sup>e</sup>	✓

a. nur funktionale Bibliotheken, z. B. für die Namensverwaltung.

b. geplante Unterstützung durch Wissensbasis

c. sehr aufwendig.

d. durch entsprechende Beschreibungssprachen.

e. Semantikdefinition durch Zwischencodekonstruktion.

## Ausblick

- Benutzungsschnittstelle
- Bedienungsanleitung
- Bibliotheksausbau
- Testprogramme als Referenz für andere Übersetzer
- (teil-) automatisch Dokumentation-/Report-Generieren

# Fragen, Vertiefung ...

# Bibliotheksgliederung

