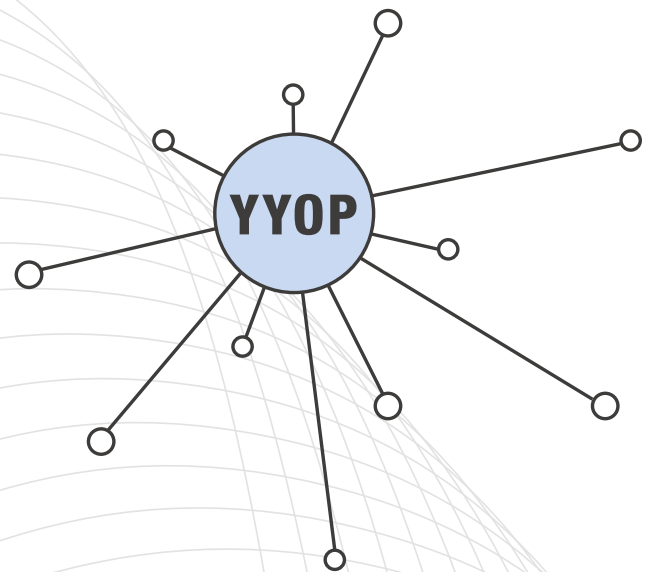


# FACTSHEET

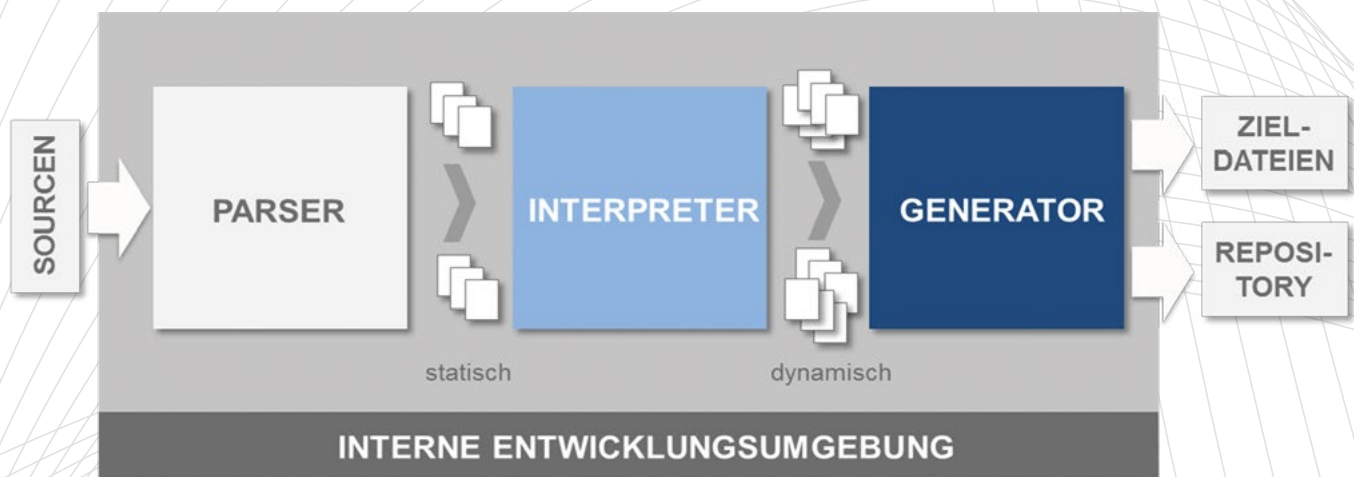
## YYOP UND DER REST DER WELT



### KOMPONENTEN VON YYOP

Mit YYOP stehen für alle Analyse- und Transformationsvorhaben die jeweils relevanten Komponenten zur Verfügung: Parser, Interpreter und Generator.

Nachfolgendes Schaubild stellt den gesamten möglichen Prozess anhand einer Sprachtransformation im Überblick dar:



**PARSER** sind die Basis des Erfolgs, denn sie erzeugen aus dem Sourcecode Abstract Syntax Trees (AST) und legen alle Informationen über Abhängigkeiten, Strukturen und Zusammenhänge der Programmquellen und Codeelemente in einem DB2-Repository sowie speziellen Filesystemdateien ab.

Mit unseren Parsern können wir sogenannten Dead-Code in einer Anwendung auffinden oder Abhängigkeitsanalysen von Programmen und Ressourcen durchführen.

Ein besonderes Herzstück unserer Technologie sind die **INTERPRETER**: damit können wir den Code einer Anwendung virtuell im Speicher ausführen.

Durch diese Interpretation kann festgestellt werden, in welcher Reihenfolge und in welchen Kombinationen Code teile durchlaufen werden. Dadurch ergeben sich Analysemöglichkeiten, die weit über eine statische Analyse hinausreichen. So kann beispielsweise der Datenfluss eines bestimmten Feldes durch das Programm hindurch verfolgt werden oder es wird möglich festzustellen, welchen Wert eine Variable zu einem bestimmten Zeitpunkt annehmen kann. Solche Funktionalitäten sind notwendig, um dynamische Calls auf Programme aufzulösen oder Registerinhalte im Rahmen einer Transformation von Assembler zu bestimmen.

Die Interpreter-Technologie ist ein Alleinstellungsmerkmal von YYOP!

Finale Komponente ist der **GENERATOR**.

Dieser generiert und optimiert den Zielcode. Die Generierung erfolgt regelbasiert, so dass die vom Kunden jeweils vorgegebenen Wartbarkeitskriterien erfüllt werden. Der Generator sorgt für eine stabile und performante Generierung, nimmt gewünschte Code-Optimierungen vor und kann auch für Code-Instrumentierungen sowie zur Testunterstützung wertvolle Beiträge leisten.

**YYOP bietet alle benötigten Hilfsmittel für die effiziente Entwicklung und Qualitätssicherung unserer Analyse- und Transformationstools. Graphische Debugger und Versionskontrolle für die zu implementierenden Grammatiken sind mächtige Werkzeuge, die uns in die Lage versetzen, auf Projektanforderungen effizient reagieren zu können.**

## WAS GENAU UNTERSCHIEDET YYOP VON ANDEREN KONVERTIERUNGS-TECHNOLOGIEN?

YYOP bietet zahlreiche Vorteile gegenüber anderen am Markt verfügbaren Technologien.

**Für technisch interessierte Leser stellen wir in diesem Factsheet einige Highlights von YYOP vor:**

- YYOP bietet dem Toolbauer eine hohe Abstraktion, dadurch erst wird die Konzentration auf die eigentliche Aufgabe möglich
- Mit YYOP sind komplexe Aufgaben schnell und sicher zu bewältigen
- YYOP bringt eine eigene Entwicklungssprache mit, mit der die Grammatik entwickelt wird:
  - keine sprachliche Trennung von Arbeitsschritten (Scanner, Parser, Verarbeitung, Generierung)
  - alles aus einem „Guss“, dadurch wird die Entwicklung wie auch die Wartung erleichtert
  - YYOP ist objektorientiert
  - Regeln, reguläre Ausdrücke und Variablen werden gekapselt
  - Regeln können gruppiert und in einer hierarchischen Aufrufreihenfolge abgebildet werden
  - YYOP ermöglicht eine modulare Entwicklung und damit einen hohen Wiederverwendungsgrad (Grammatik-Pool)
- Scanner/Parser, die auf Basis von YYOP entwickelt werden, sind sehr mächtig:
  - Top-Down-Verfahren:
    - LL(k), wobei k (Anzahl Lookahead Symbole) variabel
  - interne automatische Lookahead-Optimierung
  - Parsen mehrdeutiger oder kontext-abhängiger Sprachen (Backtracking)
  - Parse-/Syntax-Baum wird während dem Parsen automatisch erzeugt
  - beliebige Abstraktionen sind möglich (AST)
- Analyse/Verarbeitung/Konvertierung/Optimierung/Generierung ... diese Schritte können alle über Bäume erfolgen, also nicht rein textuell oder tabellarisch
- YYOP bietet umfangreiche Möglichkeiten zur Baum-Navigation und -Manipulation
- YYOP Symboltabellen sind frei definierbar:
  - dienen als interne Datenbank
  - können exportiert und importiert werden
  - sind transaktionsfähig, u.a. in Verbindung mit Backtracking
  - können Verweise auf Bäume enthalten
- YYOP verfügt über spezielle, integrierte Techniken, z.B.
  - Datenbank-Anbindungen (ODBC, SQLite)
  - Suffix Trees zur Clone-Detection
- Die YYOP-IDE: Visual YYOP unterstützt während der Entwicklung durch:
  - Modul- oder Ressource-Ansichten
  - Automatische Vervollständigung von Sprach-Elementen
  - Debugging
- Breakpoints in Grammatik oder Input-Modulen sind möglich
- Sichten auf Bäume, Symboltabellen und Variablen sowie Testunterstützung durch Grammatik-Versionskontrolle



Für ein weiterführendes Gespräch wenden Sie sich bitte gern direkt an unseren technischen Geschäftsführer **Roland Zurawka**:

Tel.: +49 751 56140-222  
Email: zurawka@pks.de  
Internet: www.pks.de

Rufen Sie einfach an oder schreiben Sie uns.



people knowing software.