

SDT is grammar + semantic rules.

Two Notations for semantic rules:

- Syntax Directed Definitions
- Syntax Directed Translation Schemes

Syntax Directed Definitions are a generalization of context-free grammars in which:

- Grammar symbols have an associated set of **Attributes**;
- Productions are associated with **Semantic Rules** for computing the Values of attributes.
- It starts at the root and recursively visits the children of each node in left-to-right order.
- The semantic rules at a given node are evaluated once all descendants of that node have been visited.
- A parse tree showing all the attribute values at each node is called annotated parse tree.

Annotated Parse Tree where each node of the tree is a record with a field for each attribute

Attribute Grammars: Each grammar symbol has an associated set of attributes. An attribute can represent anything we choose:

- The value of an expression when literal constants are used
- The data type of a constant, variable, or expression
- The location (or offset) of a variable in memory
- The translated code of an expression, statement, or function

Syntax Directed Translation Schemes: A translation scheme is a context free grammar in which

- Attributes are associated with grammar symbols
- Semantic actions are enclosed between braces {} and are inserted within the right-hand side of productions.

A translation scheme is like a syntax-directed definition except the order of evaluation of the semantic rules is explicitly shown.

Synthesized attributes have the desirable property that they can be evaluated during a single bottom-up traversal of a parse tree

Attributes. An attribute is any quantity associated with a programming construct. Examples of attributes are data types of expressions, the number of instructions in the generated code, or the location of the first instruction in the generated code for a construct, among many other possibilities. Since we use grammar symbols (non terminals and terminals) to represent programming constructs, we extend the notion of attributes from constructs to the symbols that represent them.

(Syntax-directed) translation schemes: A translation scheme is a notation for attaching program fragments to the productions of a grammar. The program fragments are executed when the production is used during syntax analysis. The combined result of all these fragment executions, in the order induced by the syntax analysis, produces the translation of the program to which this analysis/synthesis process is applied.

We shall adopt the following convention: the nonterminal appears unsubscripted in the head and with distinct subscripts in the body

Parsing is the process of determining how a string of terminals can be generated by a grammar