

2.DXL 対応 Hichart 生成規則一覧

DXL – Production Rule – 01

module_packet		
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[module_packet] 0</td> </tr> </table>	Production	[module_packet] 0
Production		
[module_packet] 0		
<table border="1"> <tr> <td>Semantic Rules</td> </tr> </table> <pre> x(1) = x(0) x(2) > x(1)+w(1) y(0) = y(1) y(1) = y(2) top(2)=top(0) bottom(0) = max(bottom(1),bottom(2)) bottom(1) = y(1) + h(1) id(1) = id(0) id(2) = id(1)+1 nc(0) = nc(2)+1 w(1) = MinW h(1) = get_height(["M_packet"]) cell(1) = "pre_defined_process" string(1) = get_str(["M_packet"]) lines(1) = get_line_begin(1,[2]) </pre>	Semantic Rules	
Semantic Rules		
<table border="1"> <tr> <td>DXL</td> </tr> </table> <pre> <モジュールパケット > ::= 'M_packet' <プロファイル句 > { <モジュール識別句 > <モジュール論理句 > } 'End_M_packet' ; ;</pre>	DXL	
DXL		

DXL – Production Rule – 02

profile_module_list(1)		
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[profile_module_list] 0</td> </tr> </table>	Production	[profile_module_list] 0
Production		
[profile_module_list] 0		
<table border="1"> <tr> <td>Semantic Rules</td> </tr> </table> <pre> x(1) = x(0) y(0) = y(1) top(1)=top(0) bottom(0) = bottom(1) id(1) = id(0) nc(0) = nc(1) </pre>	Semantic Rules	
Semantic Rules		
<table border="1"> <tr> <td>DXL</td> </tr> </table> <pre> <モジュールパケット > ::= 'M_packet' <プロファイル句 > { <モジュール識別句 > <モジュール論理句 > } 'End_M_packet' ; ;</pre>	DXL	
DXL		

DXL – Production Rule – 03

profile_module_list(2)
<p>Production</p> <pre>[profile_module_list] 0 ┌─────────┐ [profile] └─────────┘ 1 ┌─────────┐ [module_list] └─────────┘ 2</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) = x(1) y(0) = y(1) top(1) = top(0) top(2) > bottom(1) bottom(0) = bottom(2) id(1) = id(0) id(2) = id(1) + nc(1) nc(0) = nc(1) + nc(2) line(1) = get_line(1,[2])</pre>
<p>DXL</p> <pre><モジュールパケット> ::= 'M_packet' <プロファイル句> { <モジュール識別句> <モジュール論理句> } 'End_M_packet';</pre>

DXL – Production Rule – 04

profile
<p>Production</p> <pre>[profile] 0 ┌─────────┐ "Profile" └─────────┘ 1 ┌─────────┐ [explanation] └─────────┘ 2</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) > x(1) + w(1) y(0) = y(1) y(1) = y(2) top(2) = top(0) bottom(0) = max(bottom(1), bottom(2)) bottom(1) = y(1) + h(1) bottom(2) = y(2) + h(2) id(1) = id(0) id(2) = id(1) + 1 nc(0) = nc(2) + 1 w(1) = MinW h(1) = get_height(["Profile"]) cell(1) = "caption" string(1) = get_str(["Profile"]) line(1) = get_line(1,[2])</pre>
<p>DXL</p> <pre><プロファイル句> ::= 'Profile' [説明記述] 'End_Profile' ;</pre>

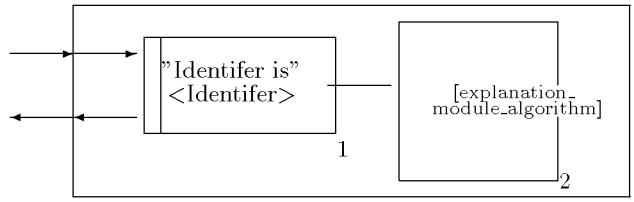
DXL – Production Rule – 05

module_list(1)		
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[module_list] 0</td> </tr> </table>	Production	[module_list] 0
Production		
[module_list] 0		
<table border="1"> <tr> <td>Semantic Rules</td> </tr> </table> <pre> x(1) = x(0) y(0) = y(1) top(1) = top(0) bottom(0) = bottom(1) id(1) = id(0) nc(0) = nc(1) </pre>	Semantic Rules	
Semantic Rules		
<table border="1"> <tr> <td>DXL</td> </tr> </table> <pre> <モジュールパケット > ::= 'M_packet' <プロファイル句 > { <モジュール識別句 > <モジュール論理句 > } 'End_M_packet' ; ;</pre>	DXL	
DXL		

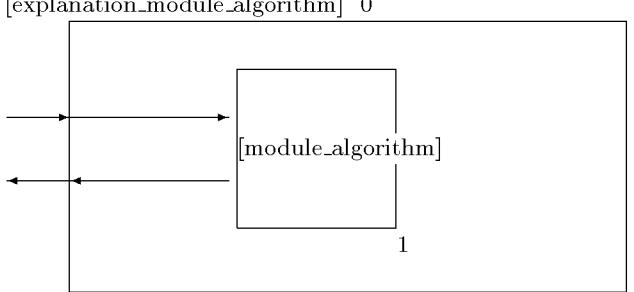
DXL – Production Rule – 06

module_list(2)		
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[module_list] 0</td> </tr> </table>	Production	[module_list] 0
Production		
[module_list] 0		
<table border="1"> <tr> <td>Semantic Rules</td> </tr> </table> <pre> x(1) = x(0) x(2) = x(1) y(0) = y(1) top(1) = top(0) top(2) > bottom(1) bottom(0) = bottom(2) id(1) = id(0) id(2) = id(1) + nc(1) nc(0) = nc(1) + nc(2) </pre> <p>line(1) = get_line(1,[2])</p>	Semantic Rules	
Semantic Rules		
<table border="1"> <tr> <td>DXL</td> </tr> </table> <pre> <モジュールパケット > ::= 'M_packet' <プロファイル句 > { <モジュール識別句 > <モジュール論理句 > } 'End_M_packet' ; ;</pre>	DXL	
DXL		

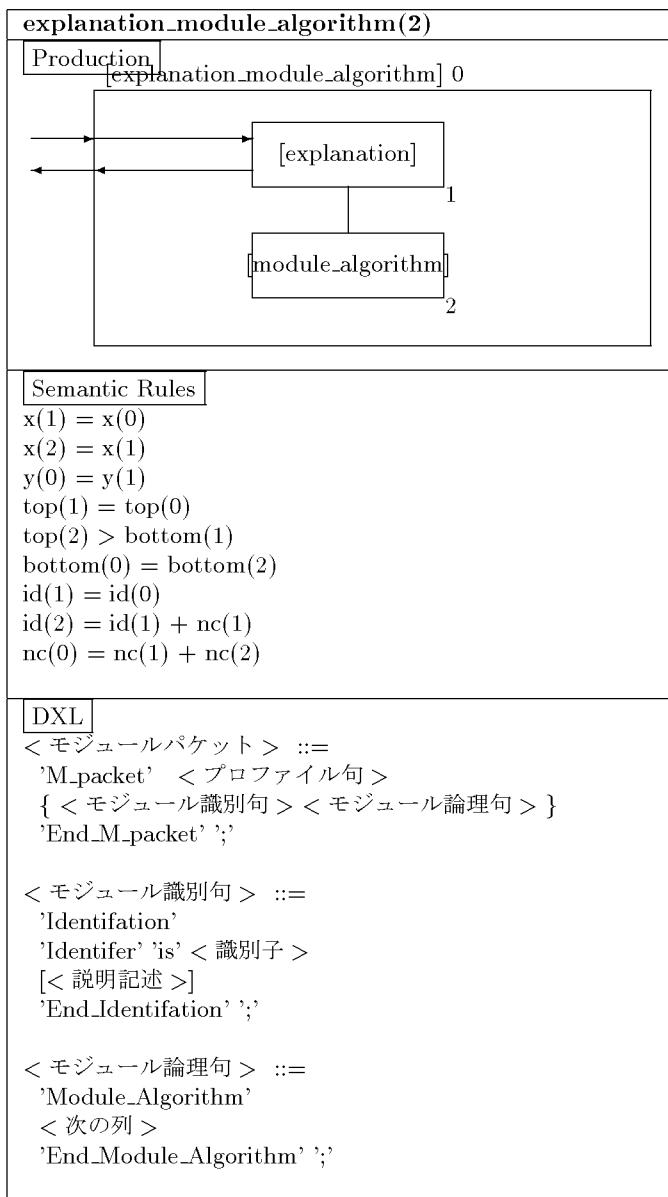
DXL – Production Rule – 07

module
Production[module] 0

Semantic Rules
$x(1) = x(0)$ $x(2) > x(1)+w(1)$ $y(0) = y(1)$ $y(1) = y(2)$ $\text{top}(2) = \text{top}(0)$ $\text{bottom}(0) = \max(\text{bottom}(1), \text{bottom}(2))$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{id}(2) = \text{id}(1) + 1$ $\text{nc}(0) = \text{nc}(2) + 1$
DXL
$<\text{モジュールパケット}> ::=$ $'\text{M_packet}' <\text{プロファイル句}>$ $\{ <\text{モジュール識別句}> <\text{モジュール論理句}> \}$ $'\text{End_M_packet}', ;$ $<\text{モジュール識別句}> ::=$ $'\text{Identification}'$ $'\text{Identifier}' 'is' <\text{識別子}>$ $[<\text{説明記述}>]$ $'\text{End_Identification}', ;$ $<\text{モジュール論理句}> ::=$ $'\text{Module_Algorithm}'$ $<\text{次の列}>$ $'\text{End_Module_Algorithm}', ;$

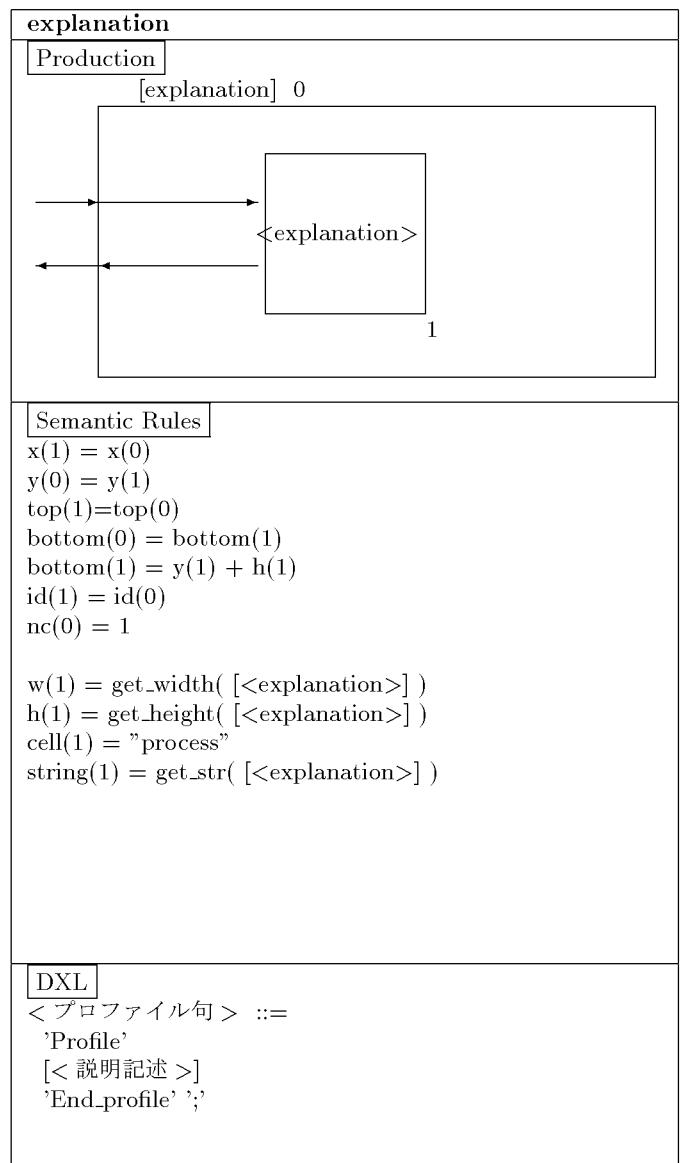
DXL – Production Rule – 08

explanation_module_algorithm(1)
Production
[explanation_module_algorithm] 0

Semantic Rules
$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$
DXL
$<\text{モジュールパケット}> ::=$ $'\text{M_packet}' <\text{プロファイル句}>$ $\{ <\text{モジュール識別句}> <\text{モジュール論理句}> \}$ $'\text{End_M_packet}', ;$ $<\text{モジュール識別句}> ::=$ $'\text{Identification}'$ $'\text{Identifier}' 'is' <\text{識別子}>$ $[<\text{説明記述}>]$ $'\text{End_Identification}', ;$ $<\text{モジュール論理句}> ::=$ $'\text{Module_Algorithm}'$ $<\text{次の列}>$ $'\text{End_Module_Algorithm}', ;$

DXL – Production Rule – 09



DXL – Production Rule – 10



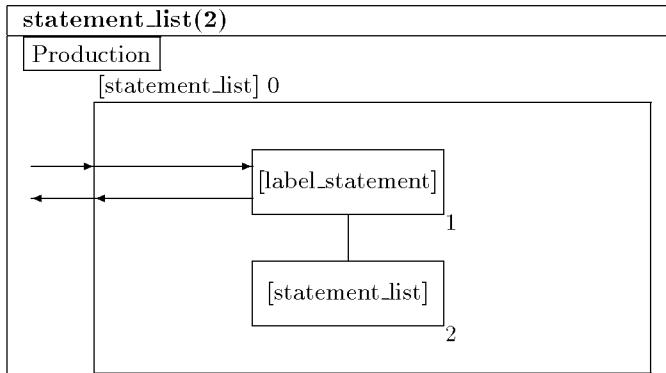
DXL – Production Rule – 11

module_algorithm
Production
<pre>[module_algorithm] 0 └── "Module_Algorithm" └── [statement_list] └── 1 └── 2</pre>
Semantic Rules
<pre>x(1) = x(0) x(2) > x(1) + w(1) y(0) = y(1) y(1) = y(2) top(2)=top(0) bottom(0) = max(bottom(1), bottom(2)) bottom(1) = y(1) + h(1) id(1) = id(0) id(2) = id(1) + 1 nc(0) = nc(2) + 1 w(1) = MinW h(1) = get_height("Module_Algorithm") cell(1) = "pre_defined_process" string(1) = get_str("Module_Algorithm") lines(1) = get_line_begin(1,[2])</pre>
DXL
<pre>< モジュール論理句 > ::= 'Module_Algorith' < 文の列 > 'End_Module_Algorithm' ;'</pre>

DXL – Production Rule – 12

statement_list(1)
Production
<pre>[statement_list] 0 └── [label_statement] └── 1</pre>
Semantic Rules
<pre>x(1) = x(0) y(0) = y(1) top(1) = top(0) bottom(0) = bottom(1) id(1) = id(0) nc(0) = nc(1)</pre>
DXL
<pre>< 文の列 > ::= { {<< 名札 >> ":"} < 文 > }</pre>

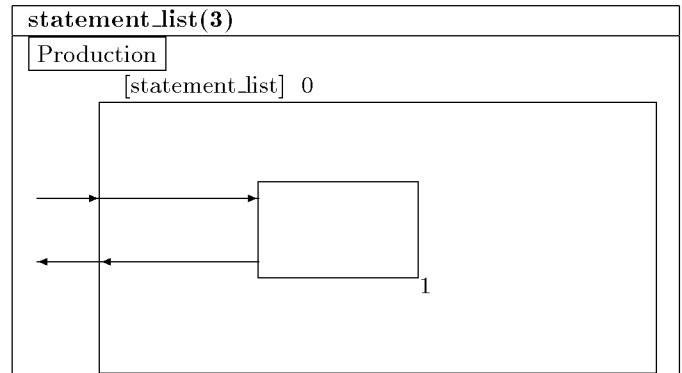
DXL – Production Rule – 13



Semantic Rules
$x(1) = x(0)$
$x(2) = x(1)$
$y(0) = y(1)$
$\text{top}(1) = \text{top}(0)$
$\text{top}(2) > \text{bottom}(1)$
$\text{bottom}(0) = \text{bottom}(2)$
$\text{id}(1) = \text{id}(0)$
$\text{id}(2) = \text{id}(1) + \text{nc}(1)$
$\text{nc}(0) = \text{nc}(1) + \text{nc}(2)$
$\text{lines}(1) = \text{get_line}(1,[2])$

DXL
 $\langle \text{文の列} \rangle ::= \{ \{ \ll \text{名札} \gg ":" \} \langle \text{文} \rangle \}$

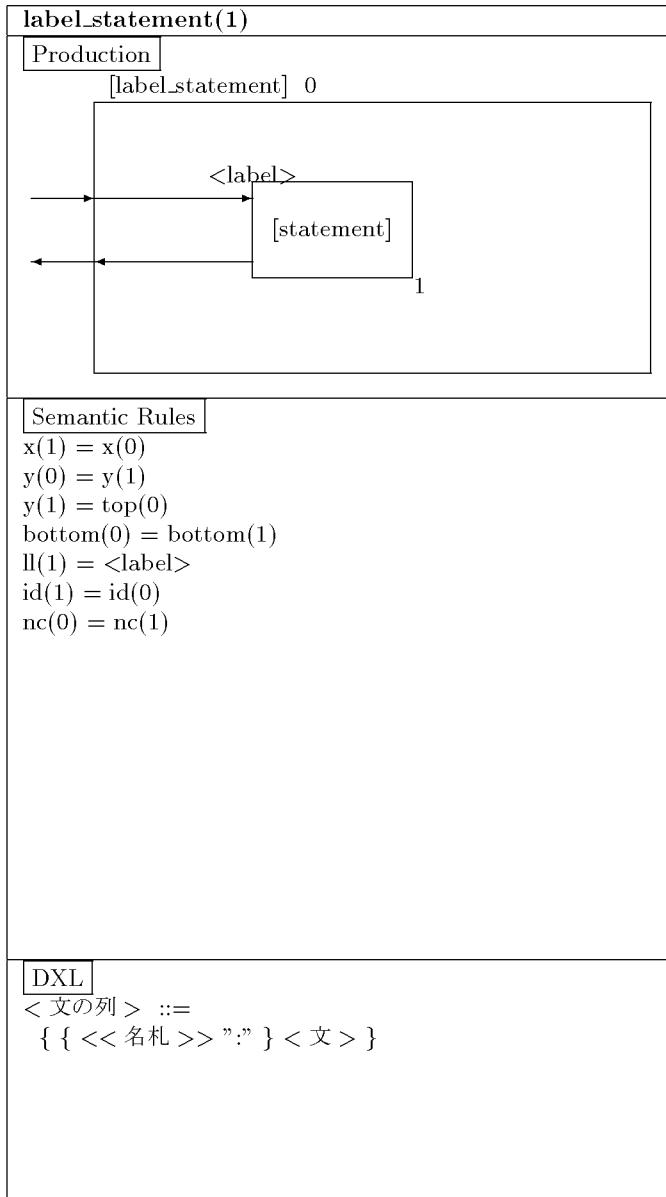
DXL – Production Rule – 14



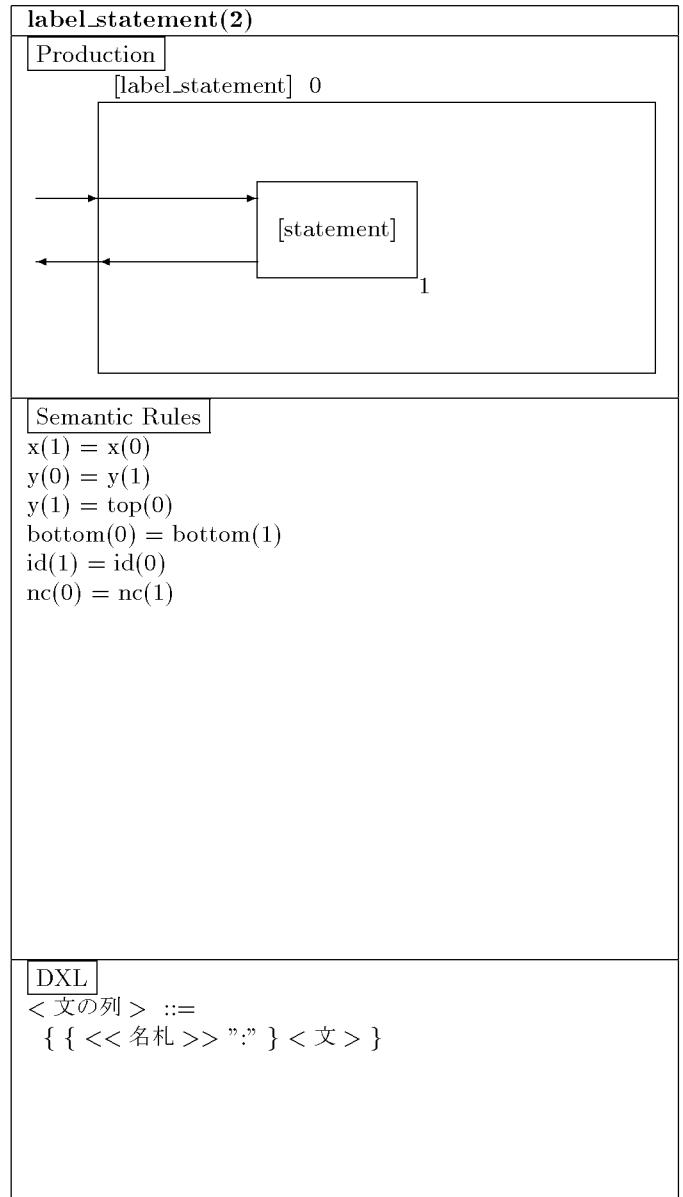
Semantic Rules
$x(1) = x(0)$
$y(0) = y(1)$
$y(1) = \text{top}(0)$
$\text{bottom}(0) = \text{bottom}(1)$
$\text{bottom}(1) = y(1) + h(1)$
$\text{id}(1) = \text{id}(0)$
$\text{nc}(0) = 1$
$w(1) = \text{MinW}$
$h(1) = \text{MinH}$
$\text{cell} = \text{"process"}$
$\text{string}(1) = \text{get_str}(\text{""})$

DXL
 $\langle \text{文の列} \rangle ::= \{ \{ \ll \text{名札} \gg ":" \} \langle \text{文} \rangle \}$

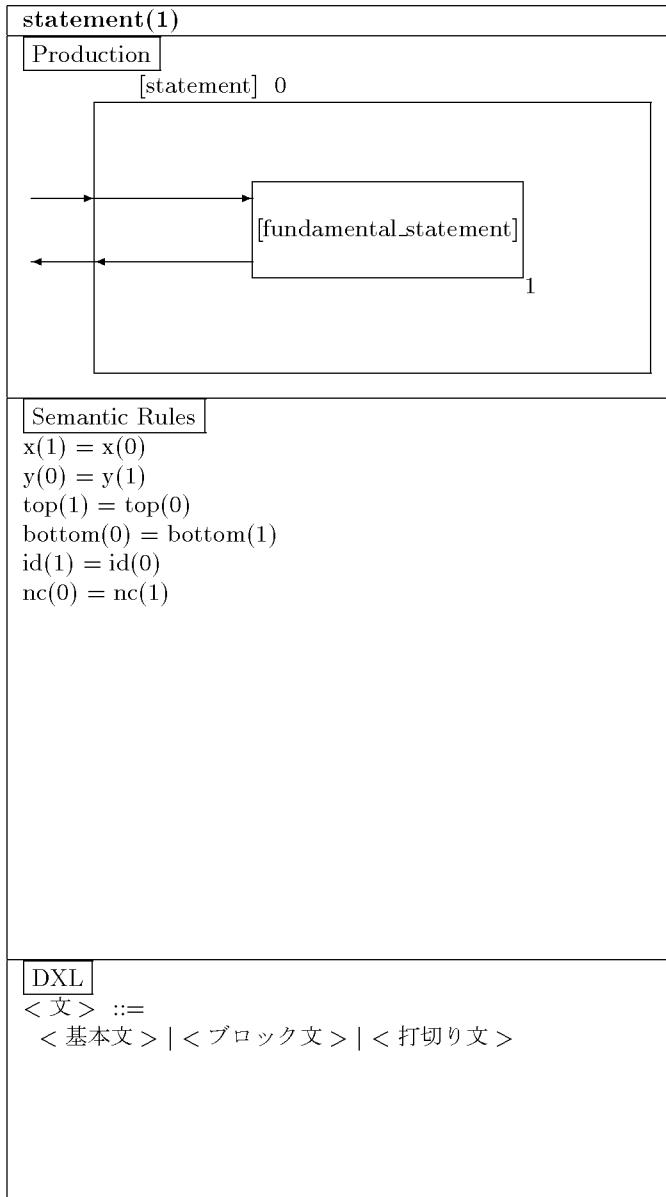
DXL – Production Rule – 15



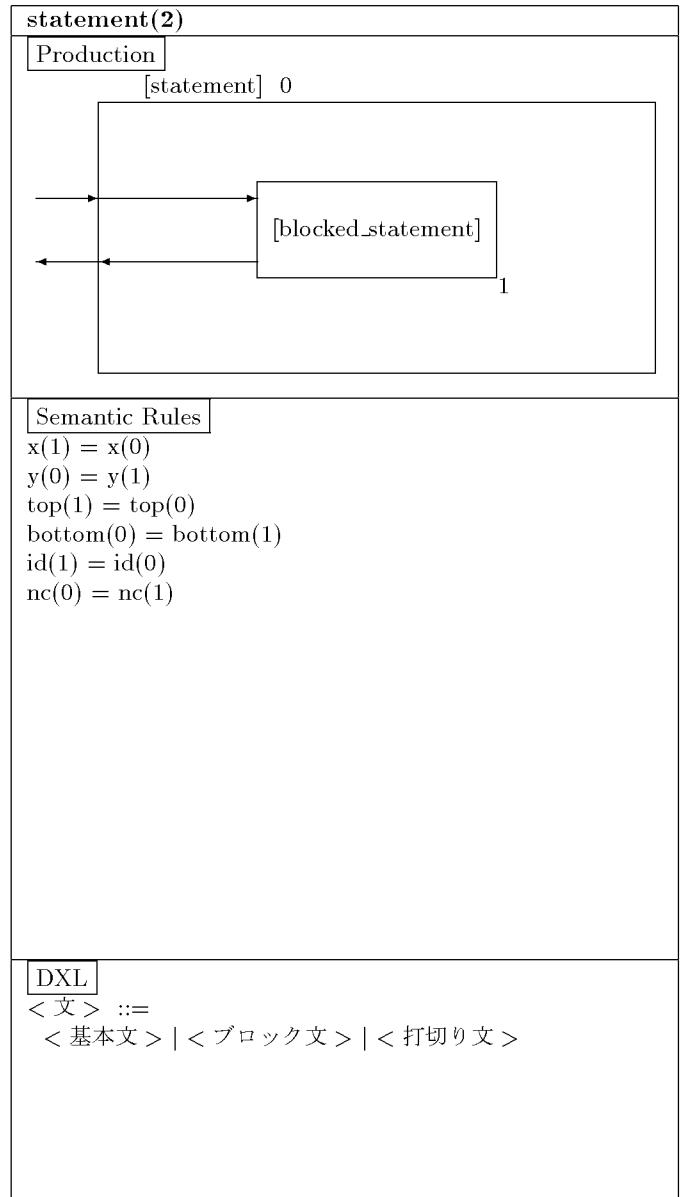
DXL – Production Rule – 16



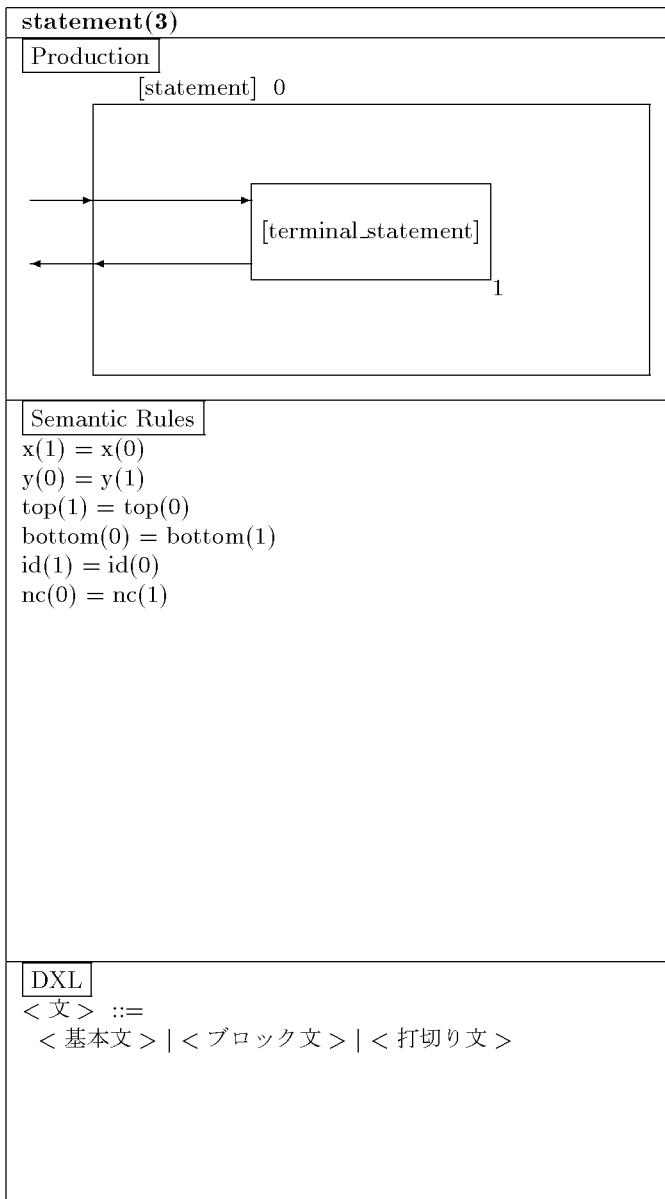
DXL – Production Rule – 17



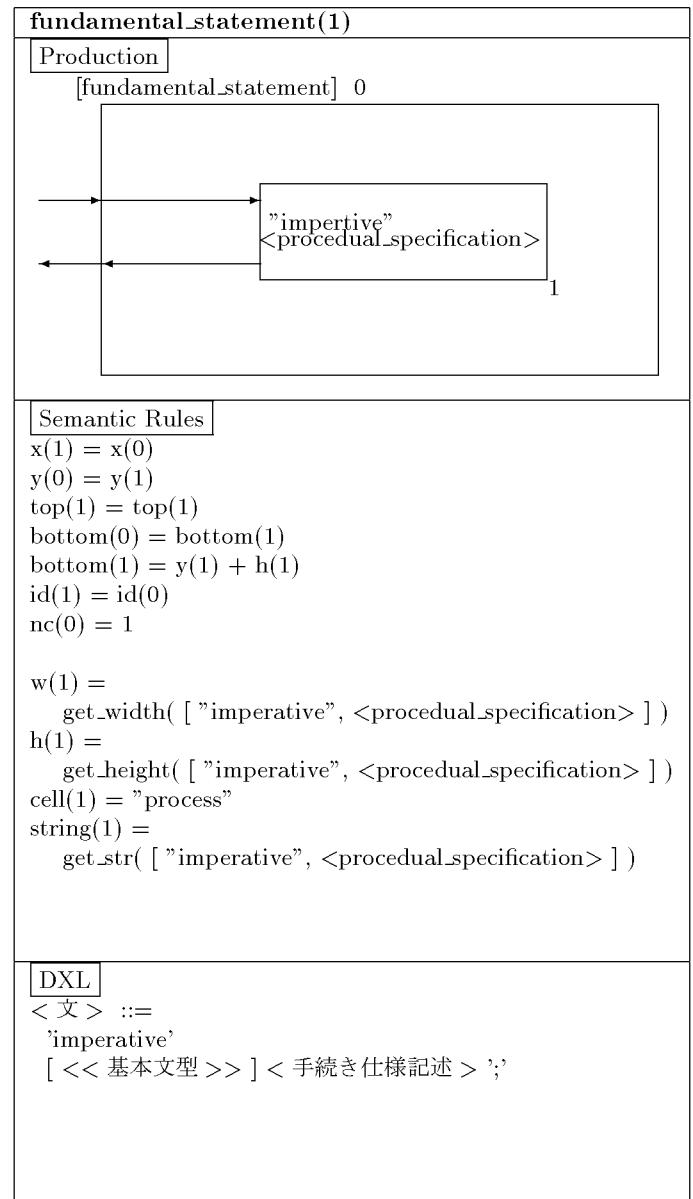
DXL – Production Rule – 18



DXL – Production Rule – 19



DXL – Production Rule – 20



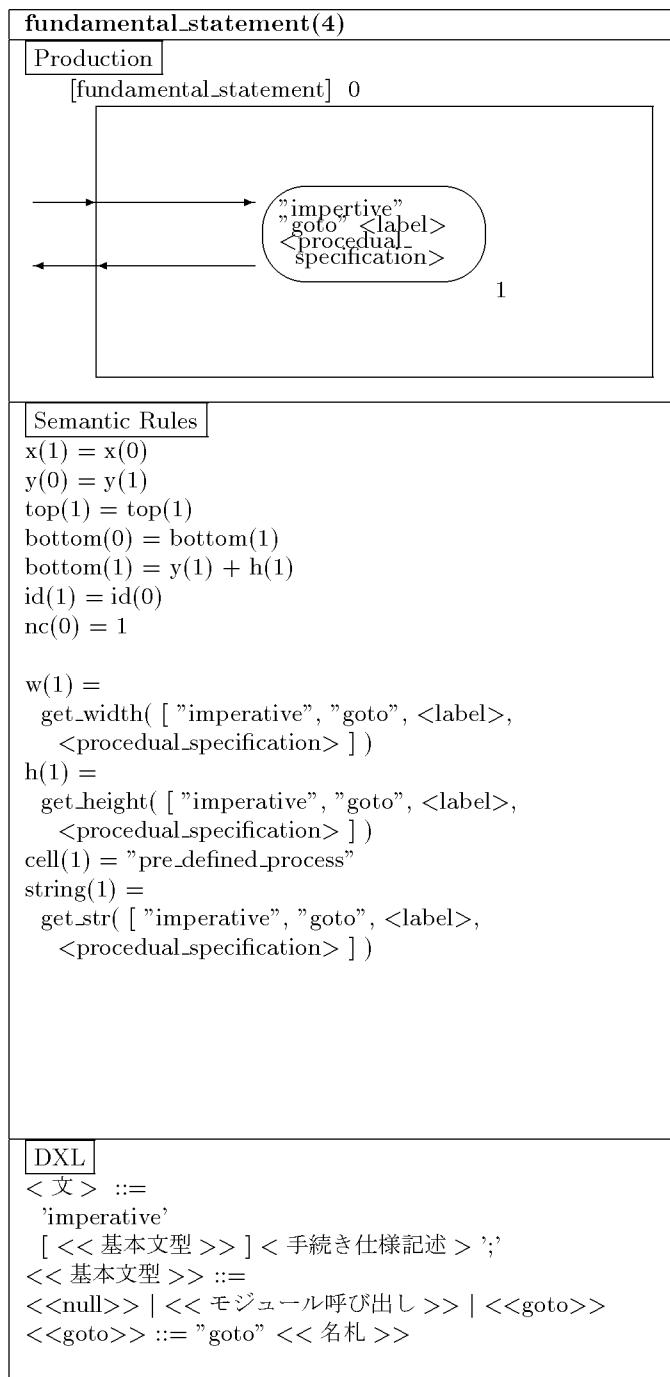
fundamental_statement(2)				
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[fundamental_statement] 0</td> </tr> <tr> <td> </td> </tr> <tr> <td>1</td> </tr> </table>	Production	[fundamental_statement] 0		1
Production				
[fundamental_statement] 0				
1				

| | | |---| | Semantic Rules | | $x(1) = x(0)$
$y(0) = y(1)$
$\text{top}(1) = \text{top}(1)$
$\text{bottom}(0) = \text{bottom}(1)$
$\text{bottom}(1) = y(1) + h(1)$
$\text{id}(1) = \text{id}(0)$
$\text{nc}(0) = 1$ | |
| | | |---| | DXL | | $<\text{文}> ::=$
$'\text{imperative}'$
$[<<\text{基本文型}>>] <\text{手続き仕様記述}>;$
$<<\text{基本文型}>> ::=$
$<<\text{null}>> <<\text{モジュール呼び出し}>> <<\text{goto}>>$
$<<\text{null}>> ::= "null"$ | |

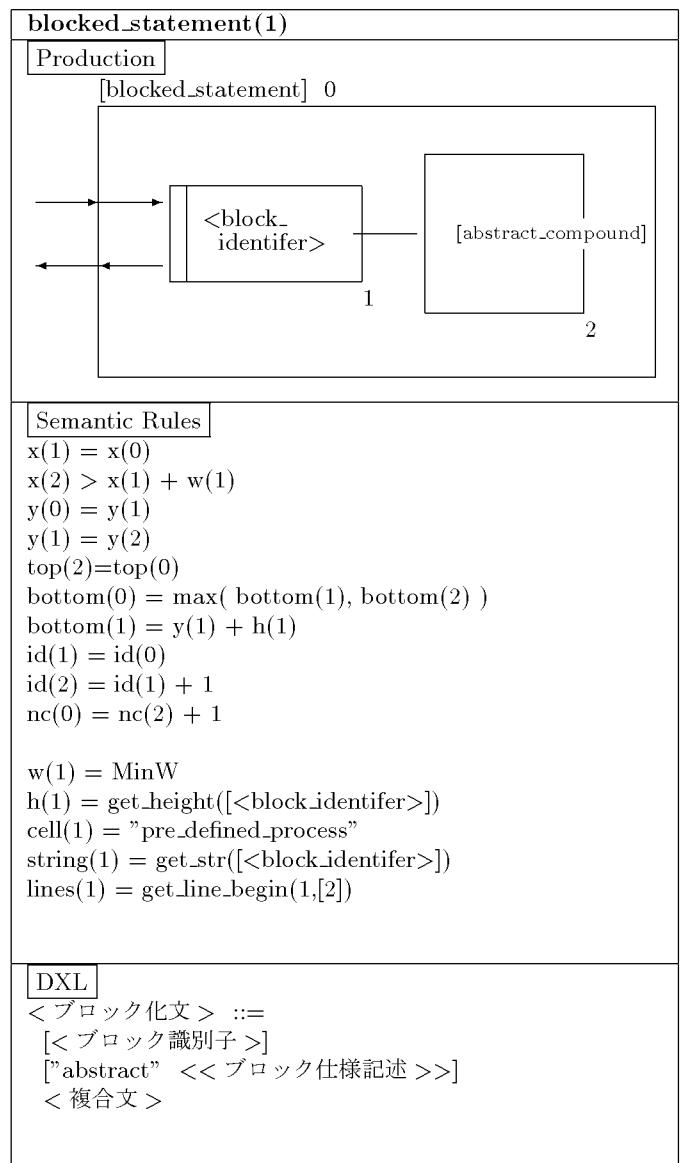
fundamental_statement(3)				
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[fundamental_statement] 0</td> </tr> <tr> <td> </td> </tr> <tr> <td>1</td> </tr> </table>	Production	[fundamental_statement] 0		1
Production				
[fundamental_statement] 0				
1				

| | | |---| | Semantic Rules | | $x(1) = x(0)$
$y(0) = y(1)$
$\text{top}(1) = \text{top}(1)$
$\text{bottom}(0) = \text{bottom}(1)$
$\text{bottom}(1) = y(1) + h(1)$
$\text{id}(1) = \text{id}(0)$
$\text{nc}(0) = 1$ | |
| | | |--| | DXL | | $<\text{文}> ::=$
$'\text{imperative}'$
$[<<\text{基本文型}>>] <\text{手続き仕様記述}>;$
$<<\text{基本文型}>> ::=$
$<<\text{null}>> <<\text{モジュール呼び出し}>> <<\text{goto}>>$
$<<\text{モジュール呼び出し}>> ::= "call" <<\text{モジュール識別子}>>$ | |

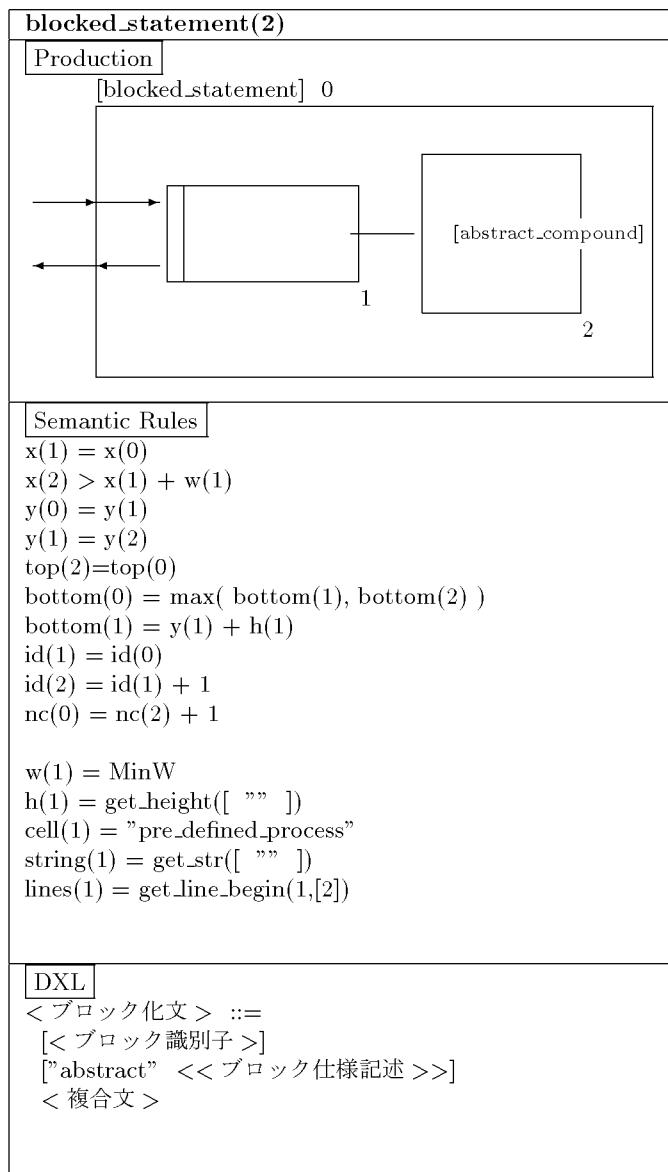
DXL – Production Rule – 23



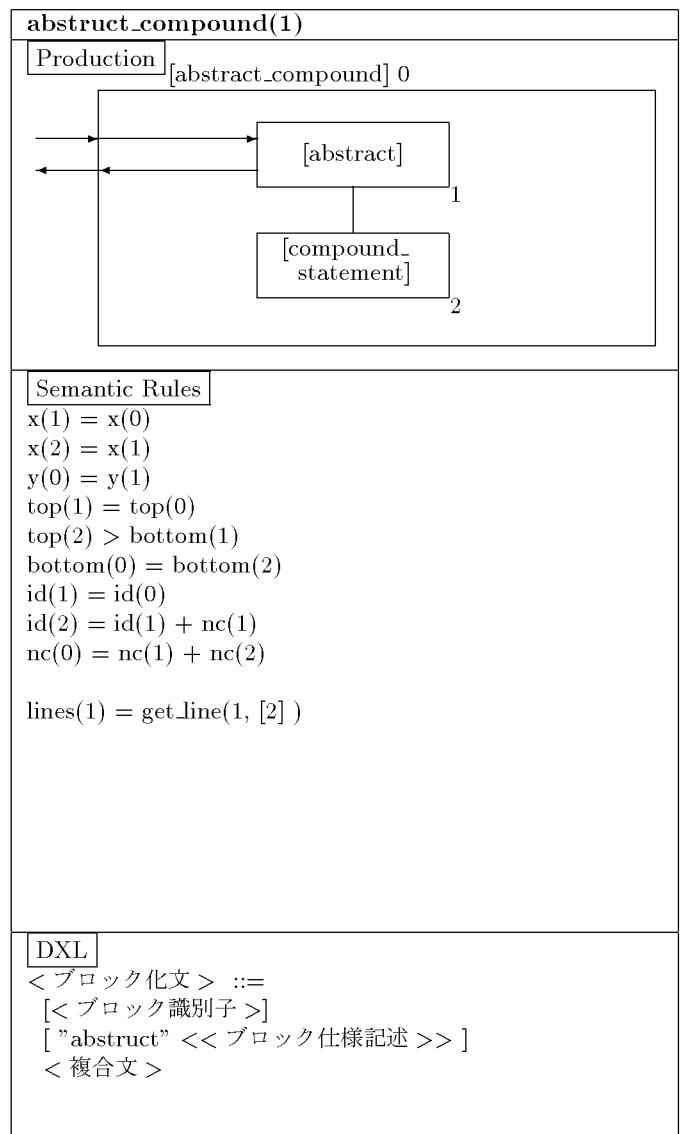
DXL – Production Rule – 24



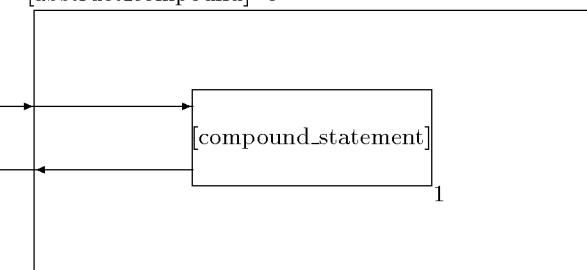
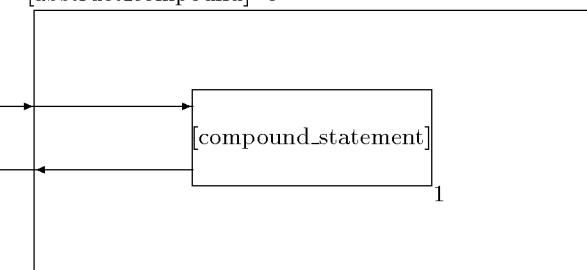
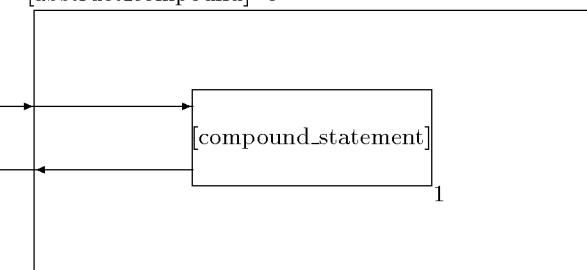
DXL – Production Rule – 25



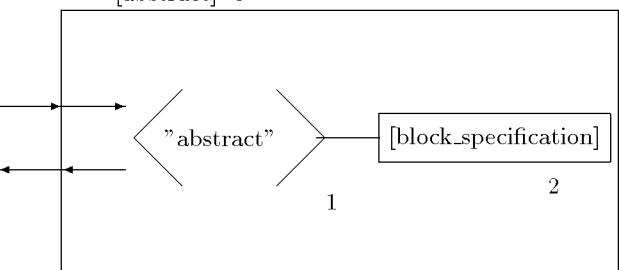
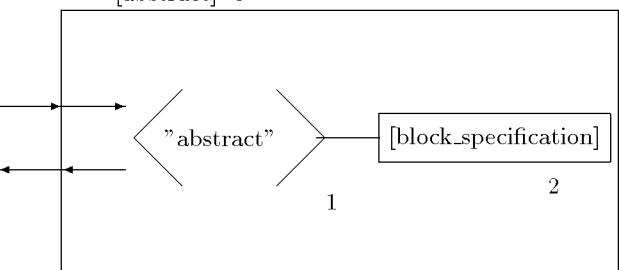
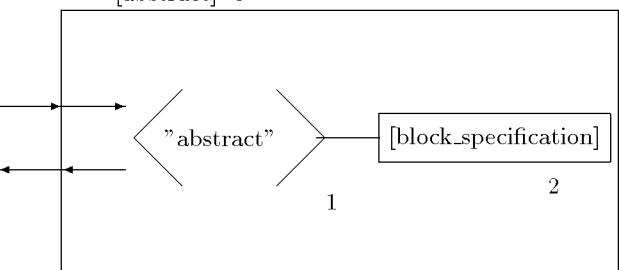
DXL – Production Rule – 26



DXL – Production Rule – 27

abstract_compound(2)			
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[abstract_compound] 0</td> </tr> <tr> <td></td> </tr> </table>	Production	[abstract_compound] 0	
Production			
[abstract_compound] 0			
			
<table border="1"> <tr> <td>Semantic Rules</td> </tr> <tr> <td> $x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$ </td> </tr> </table>	Semantic Rules	$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$	
Semantic Rules			
$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$			
<table border="1"> <tr> <td>DXL</td> </tr> <tr> <td> $<\text{ブロック化文}> ::=$ $[<\text{ブロック識別子}>]$ $[\text{"abstract"} <<\text{ブロック仕様記述}>>]$ $<\text{複合文}>$ </td> </tr> </table>	DXL	$<\text{ブロック化文}> ::=$ $[<\text{ブロック識別子}>]$ $[\text{"abstract"} <<\text{ブロック仕様記述}>>]$ $<\text{複合文}>$	
DXL			
$<\text{ブロック化文}> ::=$ $[<\text{ブロック識別子}>]$ $[\text{"abstract"} <<\text{ブロック仕様記述}>>]$ $<\text{複合文}>$			

DXL – Production Rule – 28

abstract			
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[abstract] 0</td> </tr> <tr> <td></td> </tr> </table>	Production	[abstract] 0	
Production			
[abstract] 0			
			
<table border="1"> <tr> <td>Semantic Rules</td> </tr> <tr> <td> $x(1) = x(0)$ $x(2) > x(1) + w(1)$ $y(0) = y(1)$ $y(1) = y(2)$ $\text{top}(2) = \text{top}(0)$ $\text{bottom}(0) = \max(\text{bottom}(1), \text{bottom}(2))$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{id}(2) = \text{id}(1) + 1$ $\text{nc}(0) = \text{nc}(2) + 1$ $w(1) = \text{MinW}$ $h(1) = \text{get_height}([\text{"abstract"}])$ $\text{cell}(1) = \text{"caption"}$ $\text{string}(1) = \text{get_str}([\text{"abstract"}])$ $\text{lines}(1) = \text{get_line_begin}(1,[2])$ </td> </tr> </table>	Semantic Rules	$x(1) = x(0)$ $x(2) > x(1) + w(1)$ $y(0) = y(1)$ $y(1) = y(2)$ $\text{top}(2) = \text{top}(0)$ $\text{bottom}(0) = \max(\text{bottom}(1), \text{bottom}(2))$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{id}(2) = \text{id}(1) + 1$ $\text{nc}(0) = \text{nc}(2) + 1$ $w(1) = \text{MinW}$ $h(1) = \text{get_height}([\text{"abstract"}])$ $\text{cell}(1) = \text{"caption"}$ $\text{string}(1) = \text{get_str}([\text{"abstract"}])$ $\text{lines}(1) = \text{get_line_begin}(1,[2])$	
Semantic Rules			
$x(1) = x(0)$ $x(2) > x(1) + w(1)$ $y(0) = y(1)$ $y(1) = y(2)$ $\text{top}(2) = \text{top}(0)$ $\text{bottom}(0) = \max(\text{bottom}(1), \text{bottom}(2))$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{id}(2) = \text{id}(1) + 1$ $\text{nc}(0) = \text{nc}(2) + 1$ $w(1) = \text{MinW}$ $h(1) = \text{get_height}([\text{"abstract"}])$ $\text{cell}(1) = \text{"caption"}$ $\text{string}(1) = \text{get_str}([\text{"abstract"}])$ $\text{lines}(1) = \text{get_line_begin}(1,[2])$			
<table border="1"> <tr> <td>DXL</td> </tr> <tr> <td> $<\text{ブロック化文}> ::=$ $[<\text{ブロック識別子}>]$ $[\text{"abstract"} <<\text{ブロック仕様記述}>>]$ $<\text{複合文}>$ </td> </tr> </table>	DXL	$<\text{ブロック化文}> ::=$ $[<\text{ブロック識別子}>]$ $[\text{"abstract"} <<\text{ブロック仕様記述}>>]$ $<\text{複合文}>$	
DXL			
$<\text{ブロック化文}> ::=$ $[<\text{ブロック識別子}>]$ $[\text{"abstract"} <<\text{ブロック仕様記述}>>]$ $<\text{複合文}>$			

DXL – Production Rule – 29

block_specification
<p>Production</p> <pre>[block_specification] 0 ┌─────────┐ <block_specification> └─────────┘ 1</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) y(0) = y(1) top(1) = top(0) bottom(0) = bottom(1) bottom(1) = y(1) + h(1) id(1) = id(0) nc(0) = nc(1) w(1) = get_width([<block_specification>]) h(1) = get_height([<block_specification>]) cell(1) = "process" string(1) = get_str([<block_specification>])</pre>
<p>DXL</p> <pre><ブロック化文> ::= [<ブロック識別子>] ["abstract" <<ブロック仕様記述>>] <複合文></pre>

DXL – Production Rule – 30

compound_statement(1)
<p>Production</p> <pre>[compound_statement] 0 ┌─────────┐ [sequential_statement] └─────────┘ 1</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) y(0) = y(1) top(1) = top(0) bottom(0) = bottom(1) id(1) = id(0) nc(0) = nc(1)</pre>
<p>DXL</p> <pre><複合文> ::= <順次文> <並列文> <繰り返し文> <選択文></pre>

DXL – Production Rule – 31

compound_statement(2)		
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[compound_statement] 0</td> </tr> </table>	Production	[compound_statement] 0
Production		
[compound_statement] 0		
<table border="1"> <tr> <td>Semantic Rules</td> </tr> <tr> <td> $x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$ </td> </tr> </table>	Semantic Rules	$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$
Semantic Rules		
$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$		
<table border="1"> <tr> <td>DXL</td> </tr> <tr> <td> $<\text{複合文}> ::=$ $<\text{順次文}> <\text{並列文}> <\text{繰り返し文}> <\text{選択文}>$ </td> </tr> </table>	DXL	$<\text{複合文}> ::=$ $<\text{順次文}> <\text{並列文}> <\text{繰り返し文}> <\text{選択文}>$
DXL		
$<\text{複合文}> ::=$ $<\text{順次文}> <\text{並列文}> <\text{繰り返し文}> <\text{選択文}>$		

DXL – Production Rule – 32

compound_statement(3)		
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[compound_statement] 0</td> </tr> </table>	Production	[compound_statement] 0
Production		
[compound_statement] 0		
<table border="1"> <tr> <td>Semantic Rules</td> </tr> <tr> <td> $x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$ </td> </tr> </table>	Semantic Rules	$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$
Semantic Rules		
$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{bottom}(1) = y(1) + h(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$		
<table border="1"> <tr> <td>DXL</td> </tr> <tr> <td> $<\text{複合文}> ::=$ $<\text{順次文}> <\text{並列文}> <\text{繰り返し文}> <\text{選択文}>$ </td> </tr> </table>	DXL	$<\text{複合文}> ::=$ $<\text{順次文}> <\text{並列文}> <\text{繰り返し文}> <\text{選択文}>$
DXL		
$<\text{複合文}> ::=$ $<\text{順次文}> <\text{並列文}> <\text{繰り返し文}> <\text{選択文}>$		

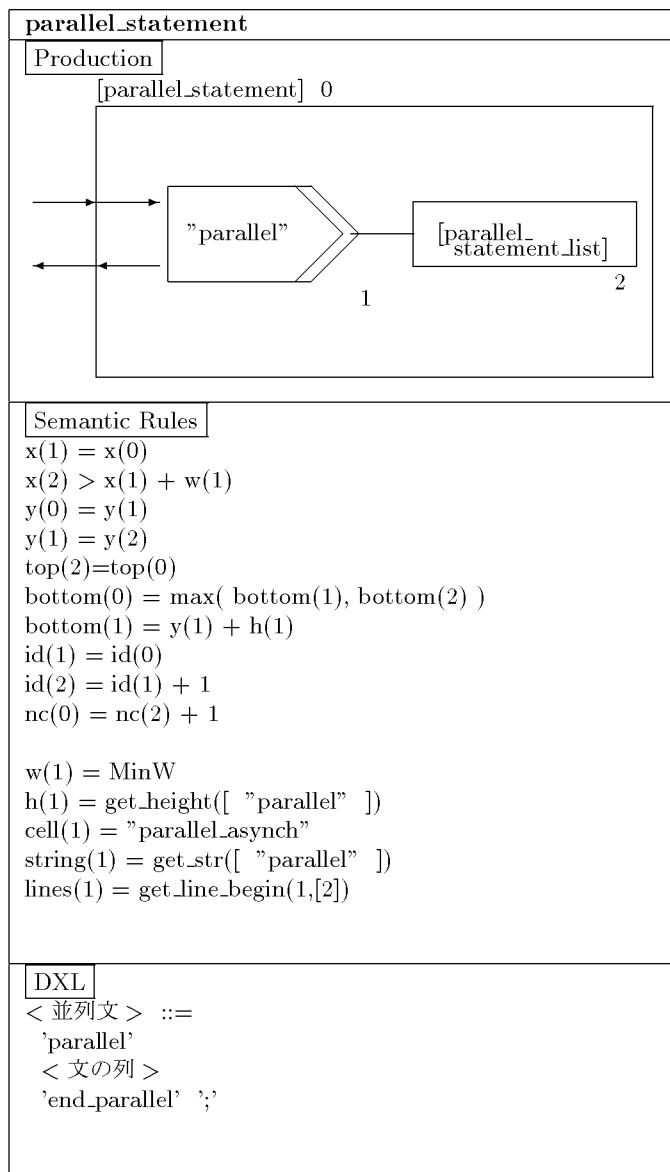
DXL – Production Rule – 33

compound_statement(4)
<p>Production</p> <pre>[compound_statement] 0 ┌─────────┐ [selective_statement] └─────────┘ 1</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) y(0) = y(1) top(1) = top(0) bottom(0) = bottom(1) bottom(1) = y(1) + h(1) id(1) = id(0) nc(0) = nc(1)</pre>
<p>DXL</p> <pre><複合文> ::=<順次文> <並列文> <繰り返し文> <選択文></pre>

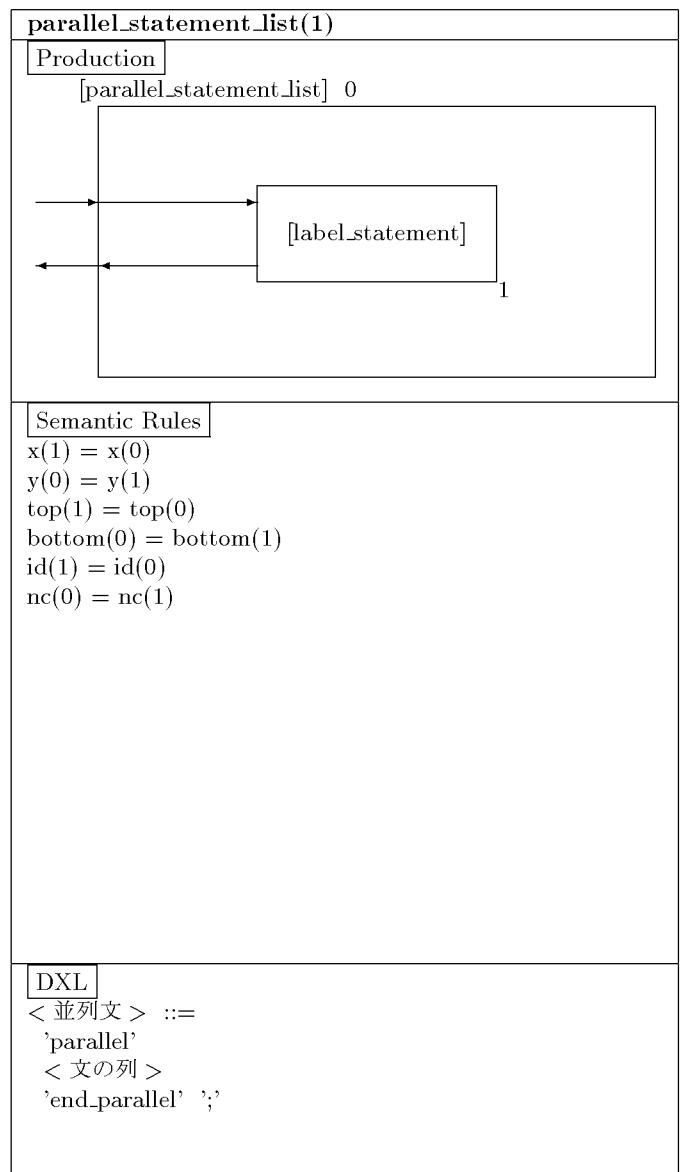
DXL – Production Rule – 34

sequential_statement
<p>Production</p> <pre>[sequential_statement] 0 ┌─────────┐ "begin" └─────────┘ 1</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) > x(1) + w(1) y(0) = y(1) y(1) = y(2) top(2)=top(0) bottom(0) = max(bottom(1), bottom(2)) bottom(1) = y(1) + h(1) id(1) = id(0) id(2) = id(1) + 1 nc(0) = nc(2) + 1</pre>
<p>w(1) = MinW h(1) = get_height(["begin"]) cell(1) = "conditions_iteration" string(1) = get_str(["begin"]) lines(1) = get_line_begin(1,[2])</p> <p>DXL</p> <pre><順次文> ::= 'begin' <文の列> 'end' ;</pre>

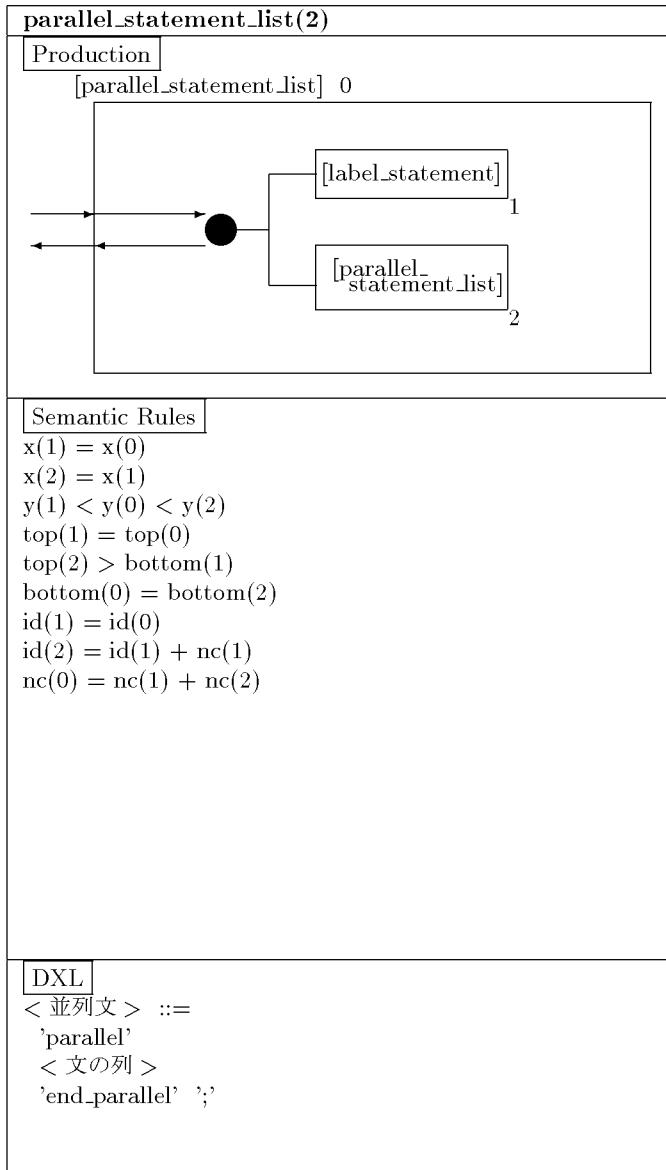
DXL – Production Rule – 35



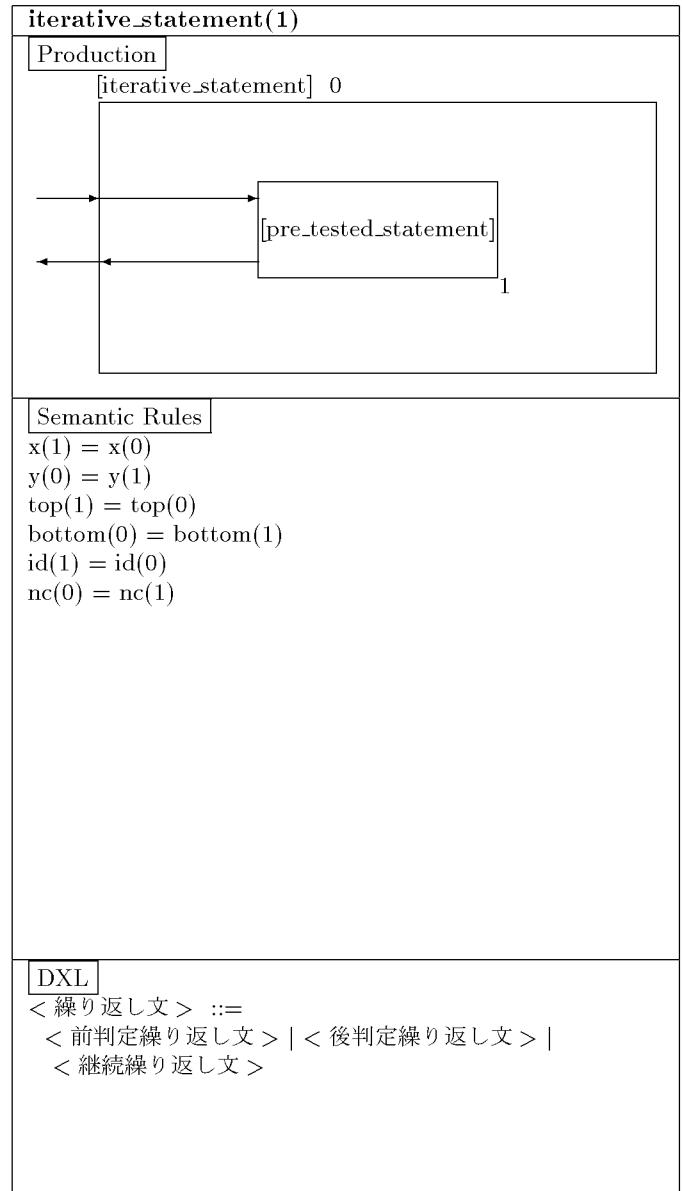
DXL – Production Rule – 36



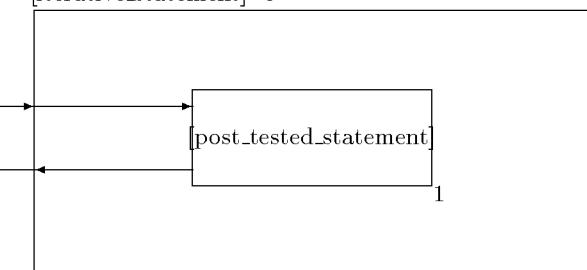
DXL – Production Rule – 37



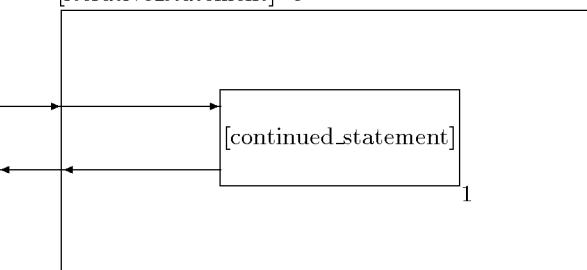
DXL – Production Rule – 38



DXL – Production Rule – 39

iterative_statement(2)	
Production	[iterative_statement] 0
	
Semantic Rules	$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$
DXL	$<\text{繰り返し文}> ::=$ $<\text{前判定繰り返し文}> <\text{後判定繰り返し文}> $ $<\text{継続繰り返し文}>$

DXL – Production Rule – 40

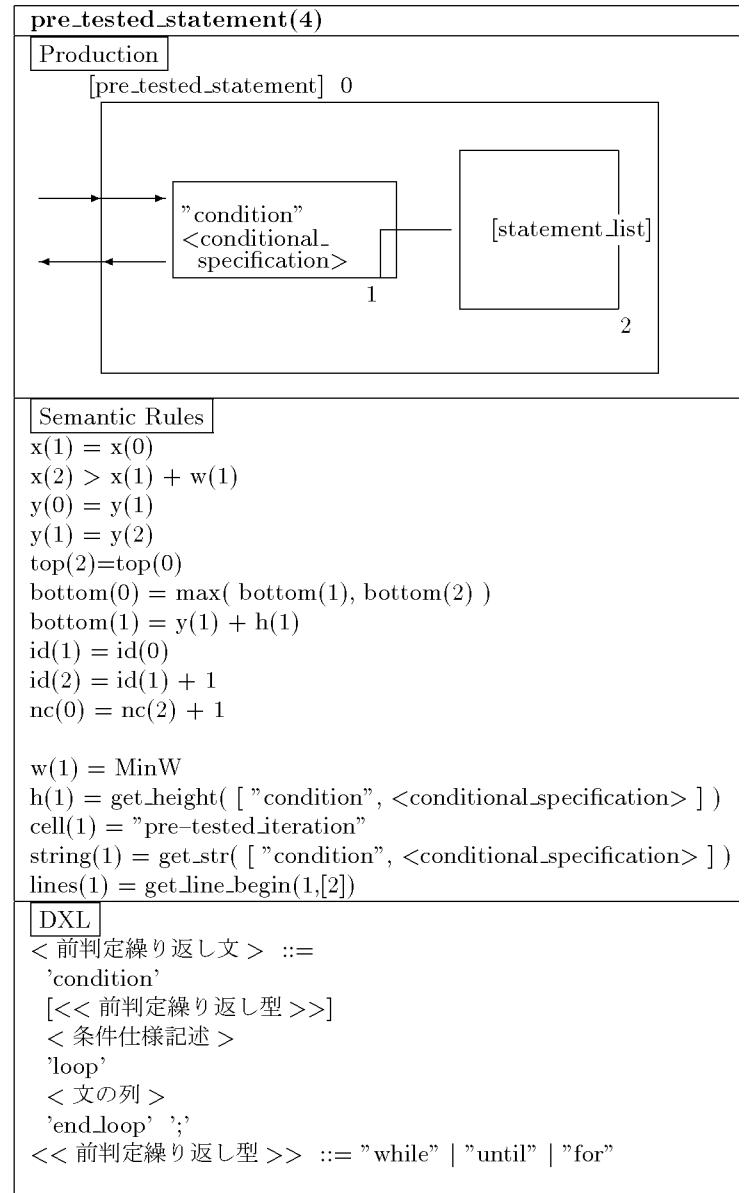
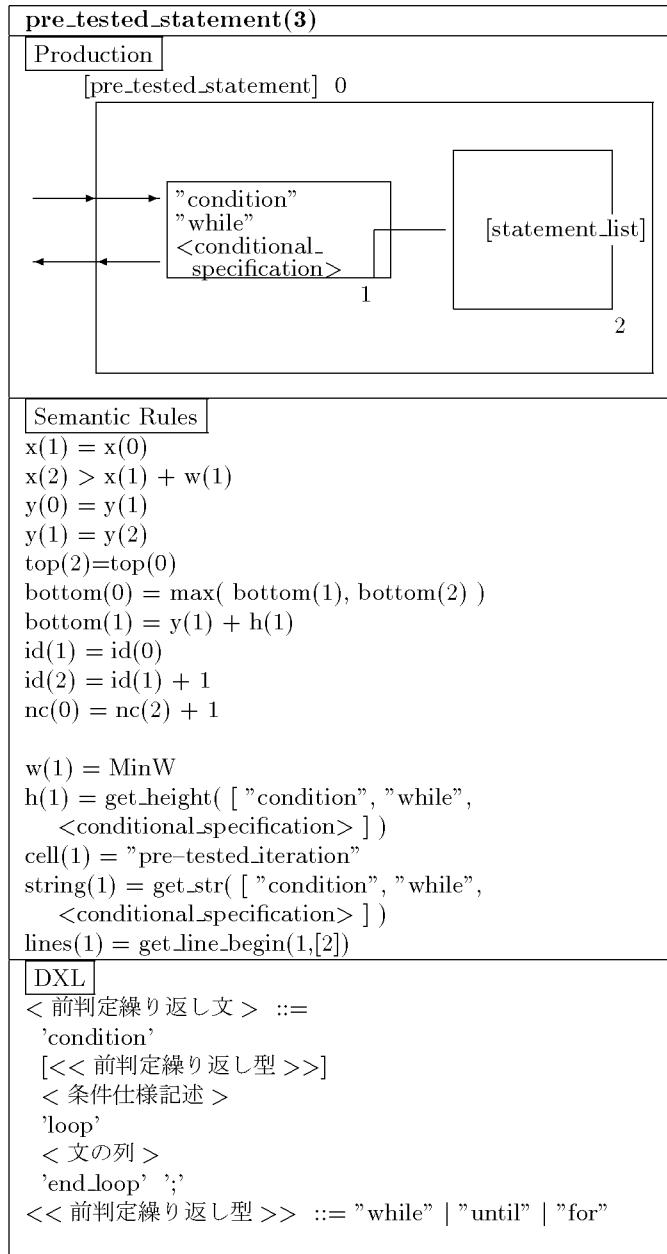
iterative_statement(3)	
Production	[iterative_statement] 0
	
Semantic Rules	$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$
DXL	$<\text{繰り返し文}> ::=$ $<\text{前判定繰り返し文}> <\text{後判定繰り返し文}> $ $<\text{継続繰り返し文}>$

DXL – Production Rule – 41

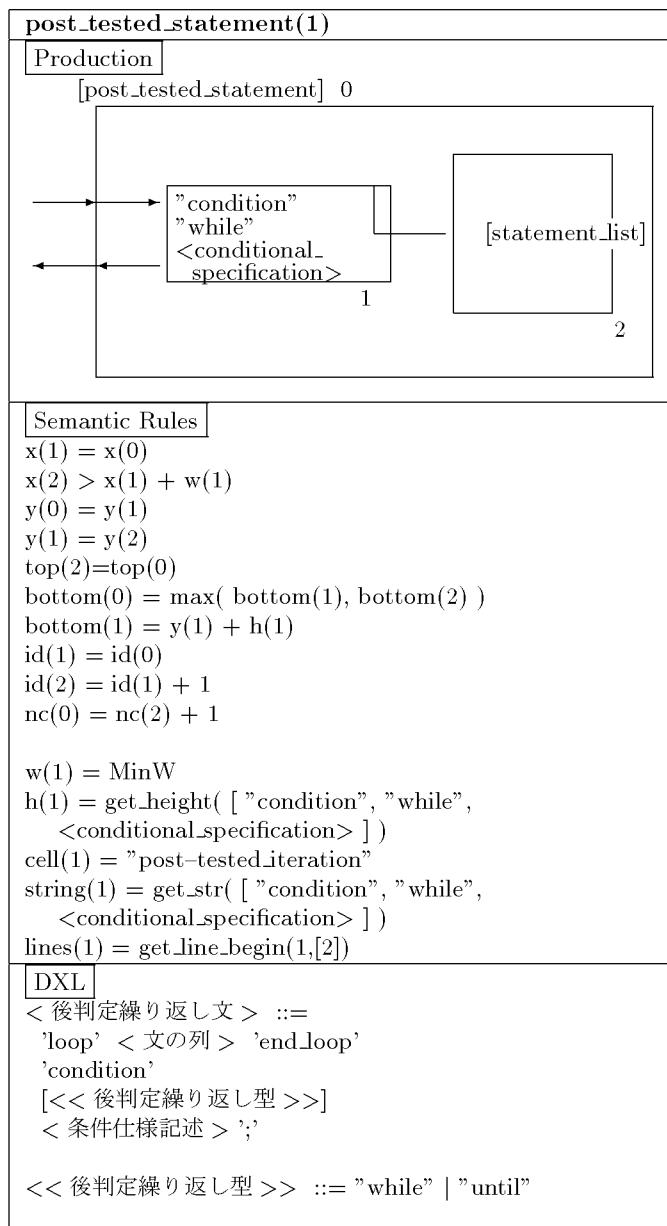
pre_tested_statement(1)
<p>Production</p> <pre>[pre_tested_statement] 0 graph LR A["'condition'"
'for'"
<conditional_specification>"] --> B["[statement_list]"] A --> C["1"] B --> D["2"]</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) > x(1) + w(1) y(0) = y(1) y(1) = y(2) top(2)=top(0) bottom(0) = max(bottom(1), bottom(2)) bottom(1) = y(1) + h(1) id(1) = id(0) id(2) = id(1) + 1 nc(0) = nc(2) + 1 w(1) = MinW h(1) = get_height(["condition", "for", <conditional_specification>]) cell(1) = "pre-tested_iteration" string(1) = get_str(["condition", "for", <conditional_specification>]) lines(1) = get_line_begin(1,[2])</pre>
<p>DXL</p> <pre><前判定繰り返し文> ::= 'condition' [<< 前判定繰り返し型 >>] <条件仕様記述> 'loop' <文の列> 'end_loop' ';' << 前判定繰り返し型 >> ::= "while" "until" "for"</pre>

DXL – Production Rule – 42

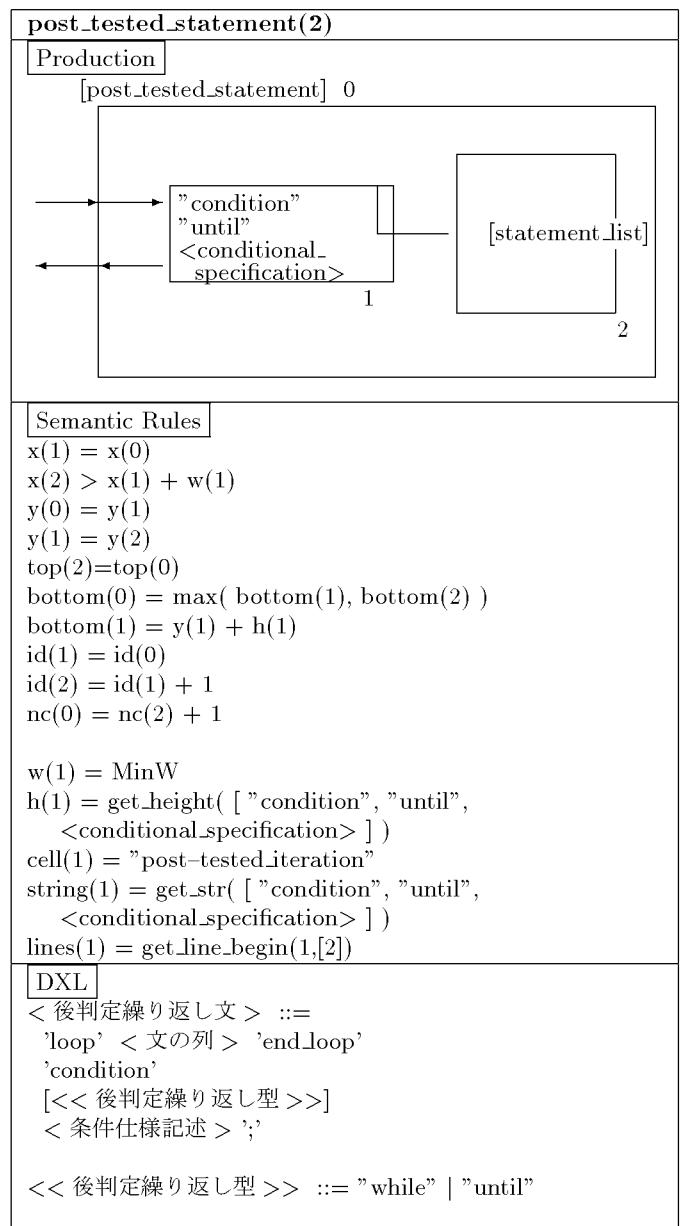
pre_tested_statement(2)
<p>Production</p> <pre>[pre_tested_statement] 0 graph LR A["'condition'"
'until'"
<conditional_specification>"] --> B["[statement_list]"] A --> C["1"] B --> D["2"]</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) > x(1) + w(1) y(0) = y(1) y(1) = y(2) top(2)=top(0) bottom(0) = max(bottom(1), bottom(2)) bottom(1) = y(1) + h(1) id(1) = id(0) id(2) = id(1) + 1 nc(0) = nc(2) + 1 w(1) = MinW h(1) = get_height(["condition", "until", <conditional_specification>]) cell(1) = "pre-tested_iteration" string(1) = get_str(["condition", "until", <conditional_specification>]) lines(1) = get_line_begin(1,[2])</pre>
<p>DXL</p> <pre><前判定繰り返し文> ::= 'condition' [<< 前判定繰り返し型 >>] <条件仕様記述> 'loop' <文の列> 'end_loop' ';' << 前判定繰り返し型 >> ::= "while" "until" "for"</pre>



DXL – Production Rule – 45



DXL – Production Rule – 46



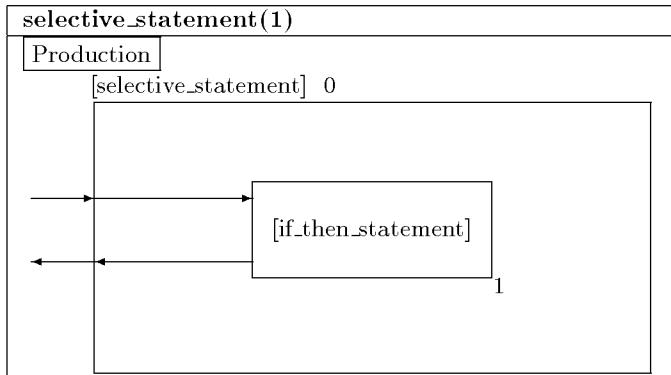
DXL – Production Rule – 47

post_tested_statement(3)
<p>Production</p> <pre>[post_tested_statement] 0 ┌─────────┐ "condition" <conditional_ specification> └─────────┘ 1 ┌─────────┐ [statement_list] └─────────┘ 2</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) > x(1) + w(1) y(0) = y(1) y(1) = y(2) top(2)=top(0) bottom(0) = max(bottom(1), bottom(2)) bottom(1) = y(1) + h(1) id(1) = id(0) id(2) = id(1) + 1 nc(0) = nc(2) + 1 w(1) = MinW h(1) = get_height(["condition", <conditional_specification>]) cell(1) = "post-tested_iteration" string(1) = get_str(["condition", <conditional_specification>]) lines(1) = get_line_begin(1,[2])</pre>
<p>DXL</p> <pre><後判定繰り返し文> ::= 'loop' <文の列> 'end_loop' 'condition' [<<後判定繰り返し型>>] <条件仕様記述> ;' <<後判定繰り返し型>> ::= "while" "until"</pre>

DXL – Production Rule – 48

continued_statement
<p>Production</p> <pre>[continued_statement] 0 ┌─────────┐ "loop" └─────────┘ 1 ┌─────────┐ [statement_list] └─────────┘ 2</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) > x(1) + w(1) y(0) = y(1) y(1) = y(2) top(2)=top(0) bottom(0) = max(bottom(1), bottom(2)) bottom(1) = y(1) + h(1) id(1) = id(0) id(2) = id(1) + 1 nc(0) = nc(2) + 1 w(1) = MinW h(1) = get_height(["loop"]) cell(1) = "continuous_iteration" string(1) = get_str(["loop"]) lines(1) = get_line_begin(1,[2])</pre>
<p>DXL</p> <pre><継続繰り返し文> ::= 'loop' <文の列> 'end_loop' ;'</pre>

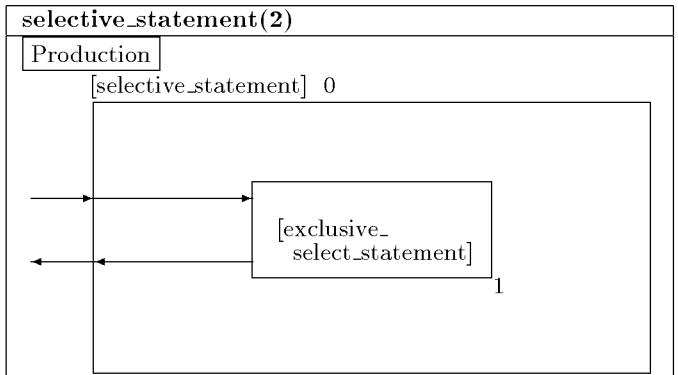
DXL – Production Rule – 49



Semantic Rules
$x(1) = x(0)$
$y(0) = y(1)$
$\text{top}(1) = \text{top}(0)$
$\text{bottom}(0) = \text{bottom}(1)$
$\text{id}(1) = \text{id}(0)$
$\text{nc}(0) = \text{nc}(1)$

DXL < 選択文 > ::=
< 簡単選択文 > | < 多岐選択文 > | < 多重岐選択文 >

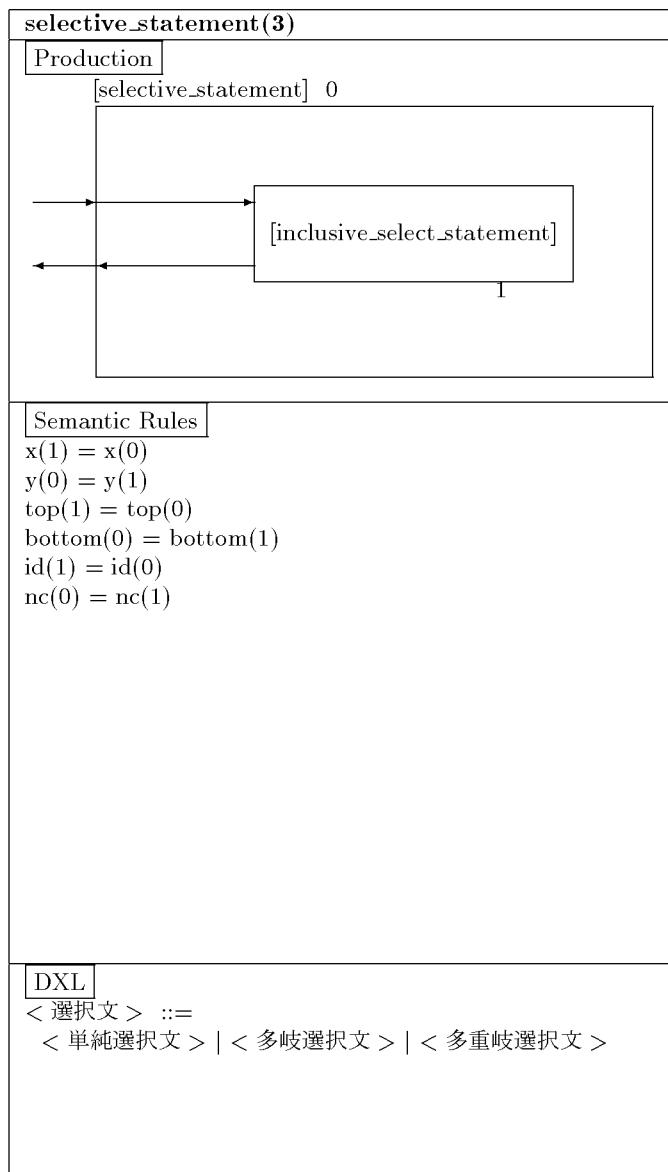
DXL – Production Rule – 50



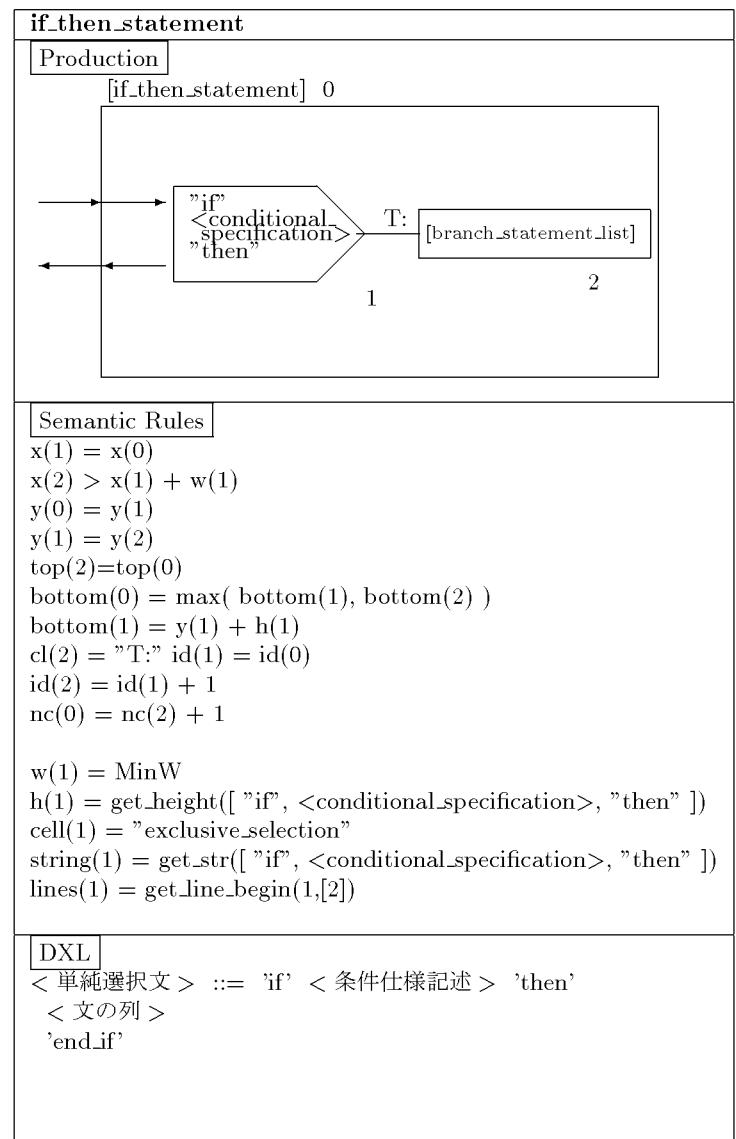
Semantic Rules
$x(1) = x(0)$
$y(0) = y(1)$
$\text{top}(1) = \text{top}(0)$
$\text{bottom}(0) = \text{bottom}(1)$
$\text{id}(1) = \text{id}(0)$
$\text{nc}(0) = \text{nc}(1)$

DXL < 選択文 > ::=
< 単純選択文 > | < 多岐選択文 > | < 多重岐選択文 >

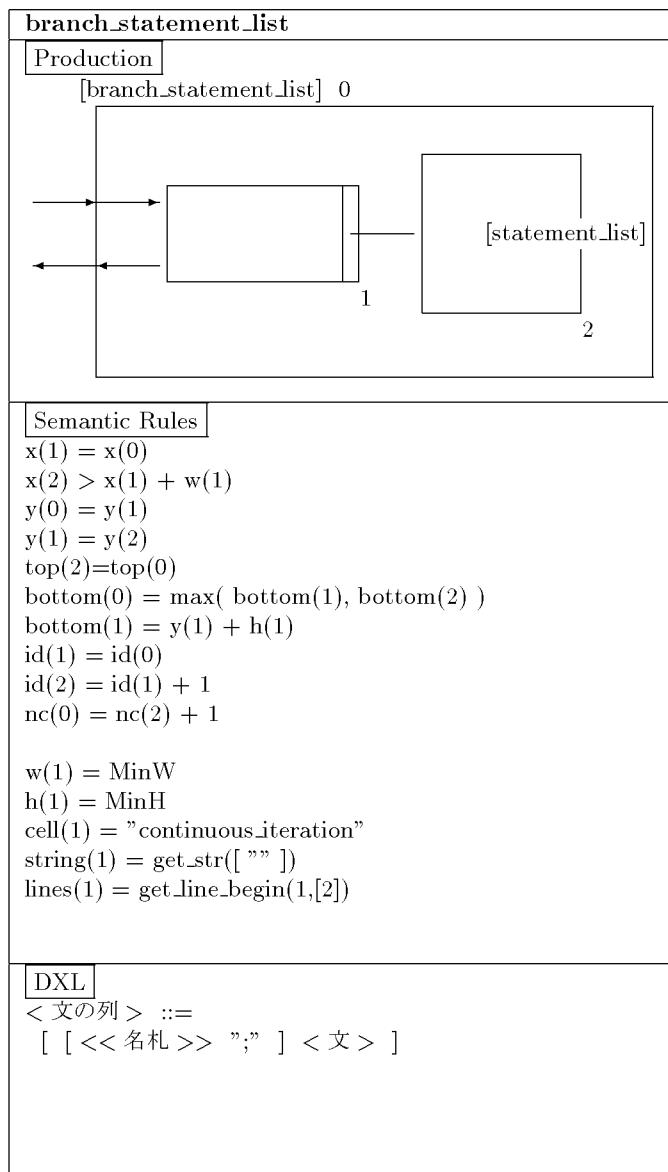
DXL – Production Rule – 51



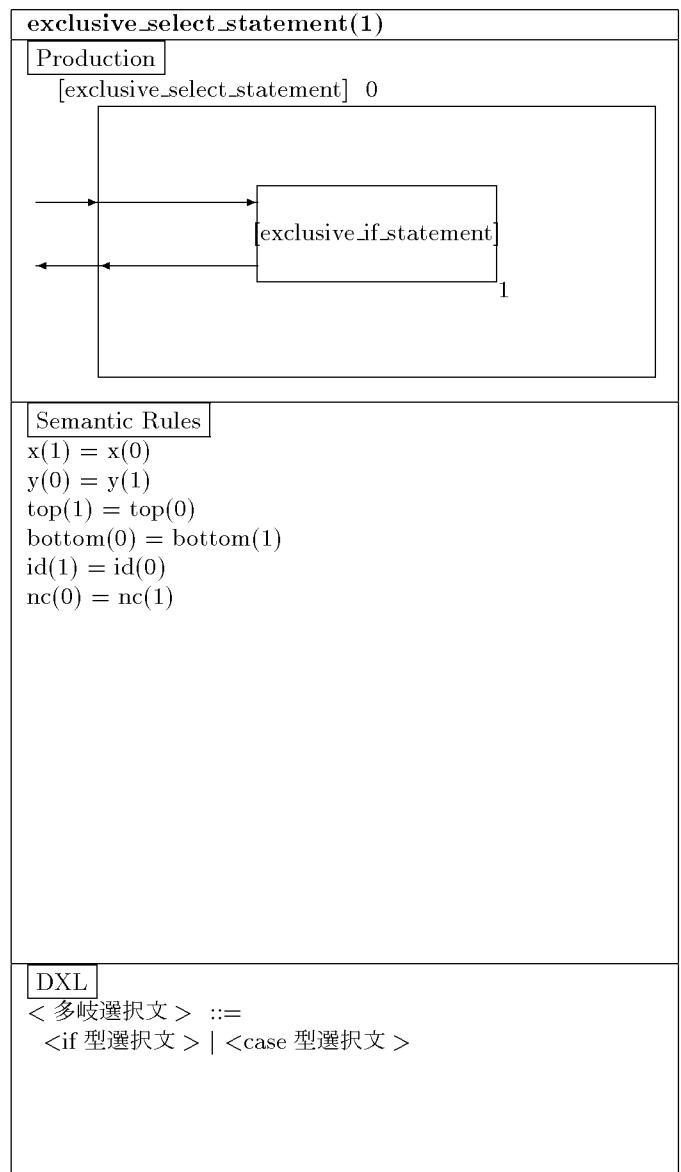
DXL – Production Rule – 52



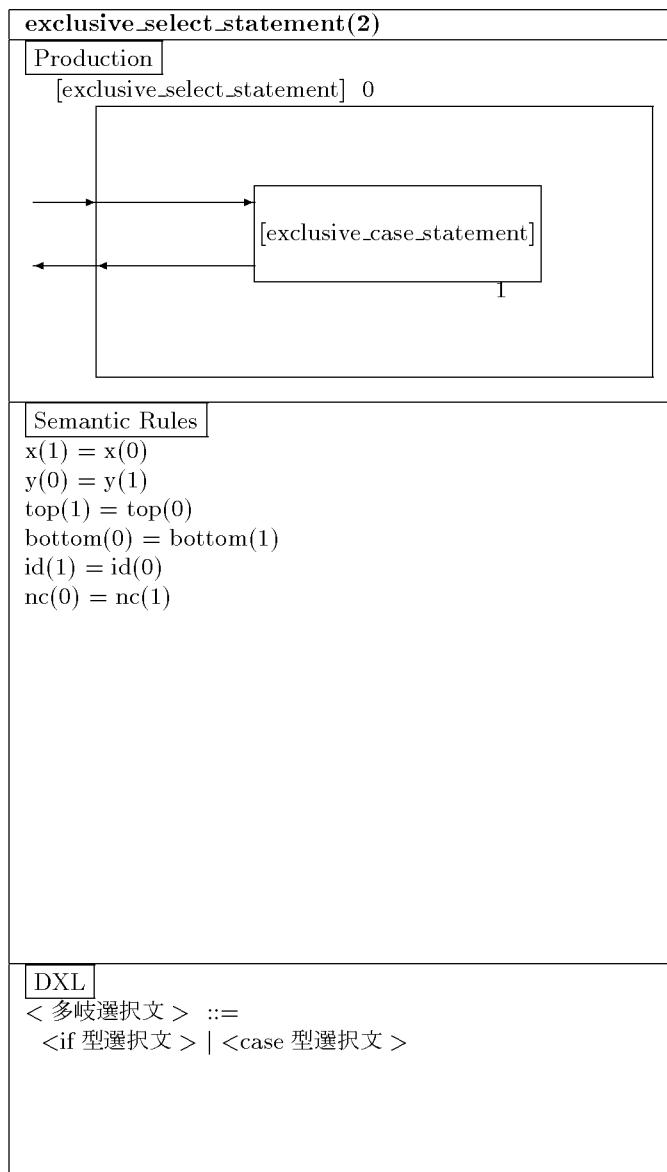
DXL – Production Rule – 53



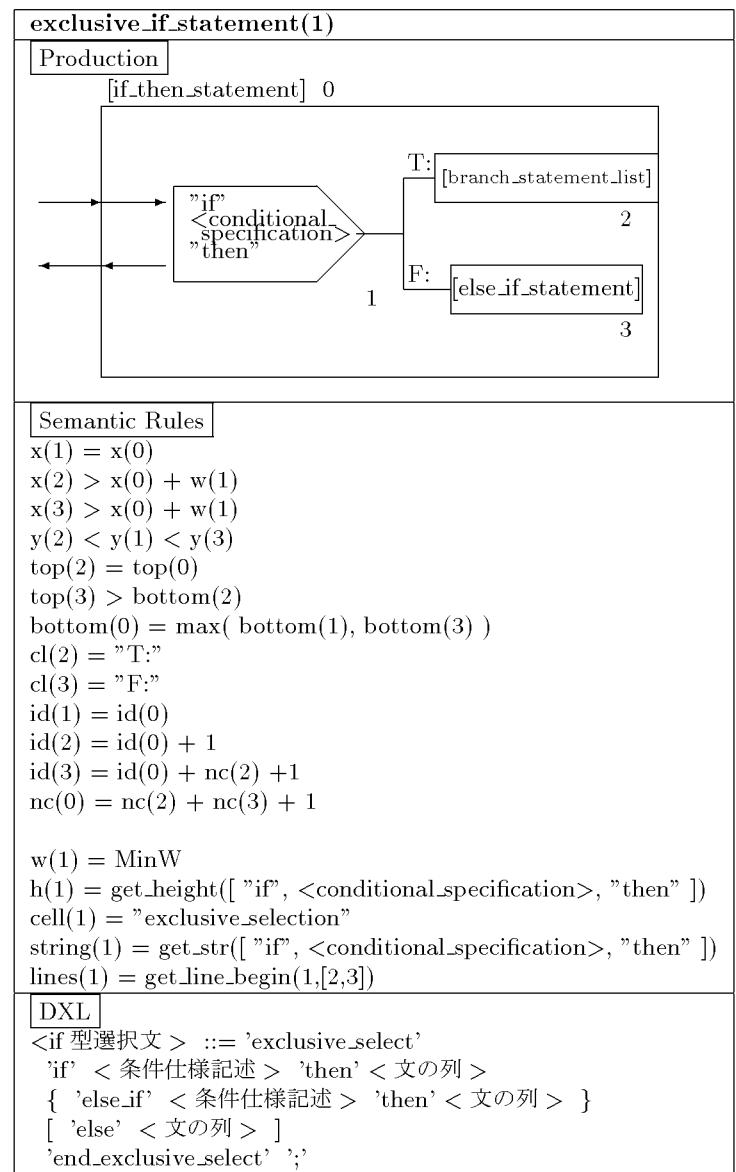
DXL – Production Rule – 54



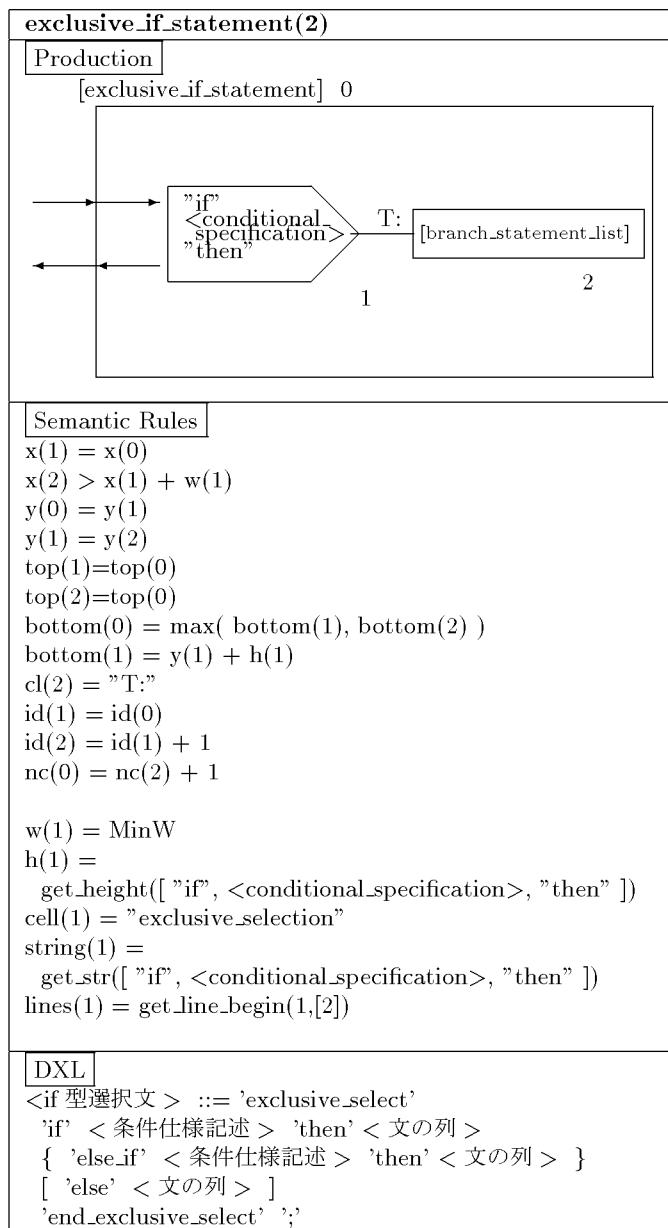
DXL – Production Rule – 55



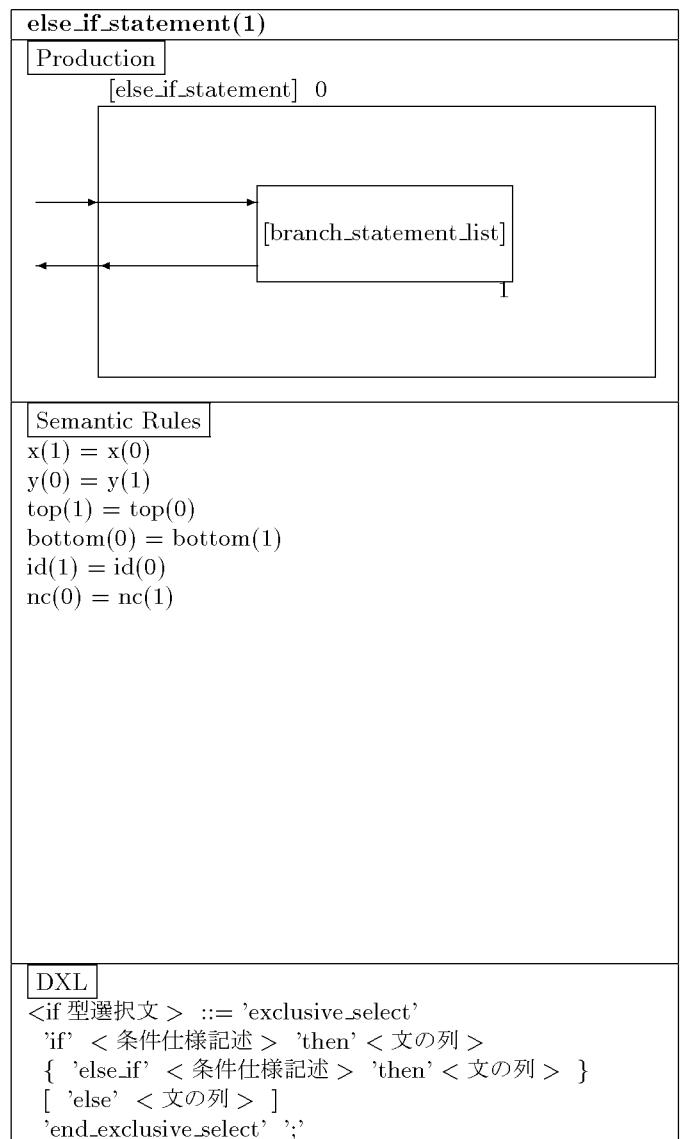
DXL – Production Rule – 56



DXL – Production Rule – 57



DXL – Production Rule – 58



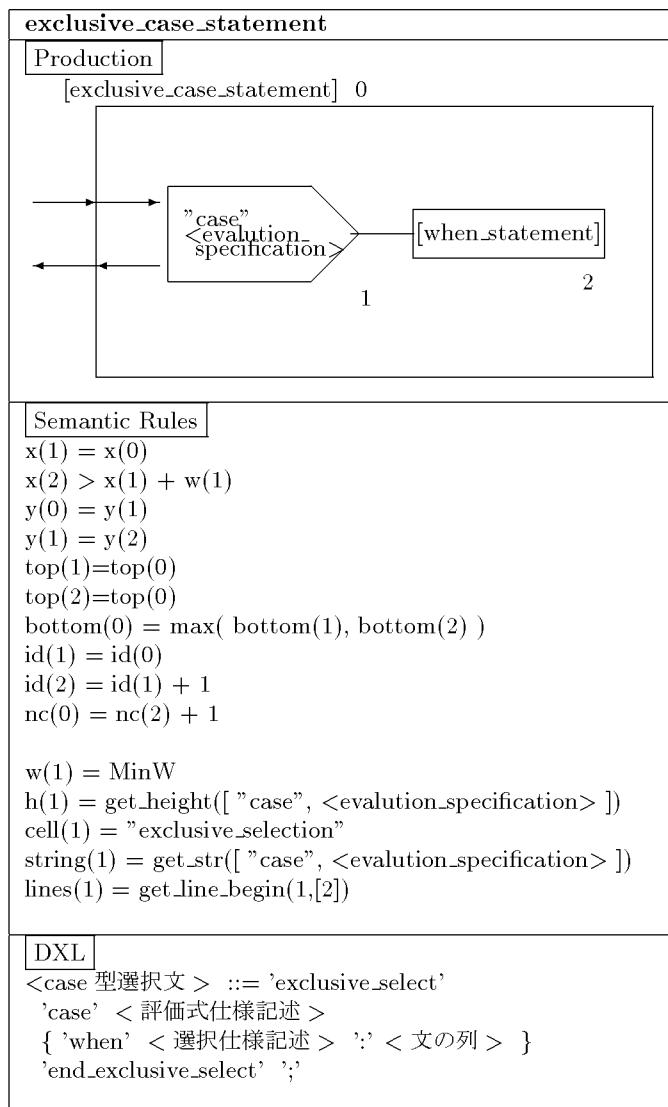
DXL – Production Rule – 59

else_if_statement(2)
<p>Production</p> <pre>[else_if_statement] 0 --> "else_if" <conditional_specification> "then" --> T: [branch_statement_list] 2 --> F: [else_if_statement] 3</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) > x(0) + w(1) x(3) > x(0) + w(1) y(2) < y(1) < y(3) top(2) = top(0) top(3) > bottom(2) bottom(0) = max(bottom(1), bottom(3)) cl(2) = "T." cl(3) = "F." id(1) = id(0) id(2) = id(0) + 1 id(3) = id(0) + nc(2) + 1 nc(0) = nc(2) + nc(3) + 1 w(1) = MinW h(1) = get_height(["if", <conditional_specification>, "then"]) cell(1) = "exclusive_selection" string(1) = get_str(["if", <conditional_specification>, "then"]) lines(1) = get_line_begin(1,[2,3])</pre>
<p>DXL</p> <pre><if 型選択文 > ::= 'exclusive_select' 'if' < 条件仕様記述 > 'then' < 文の列 > { 'else_if' < 条件仕様記述 > 'then' < 文の列 > } ['else' < 文の列 >] 'end_exclusive_select' ';'</pre>

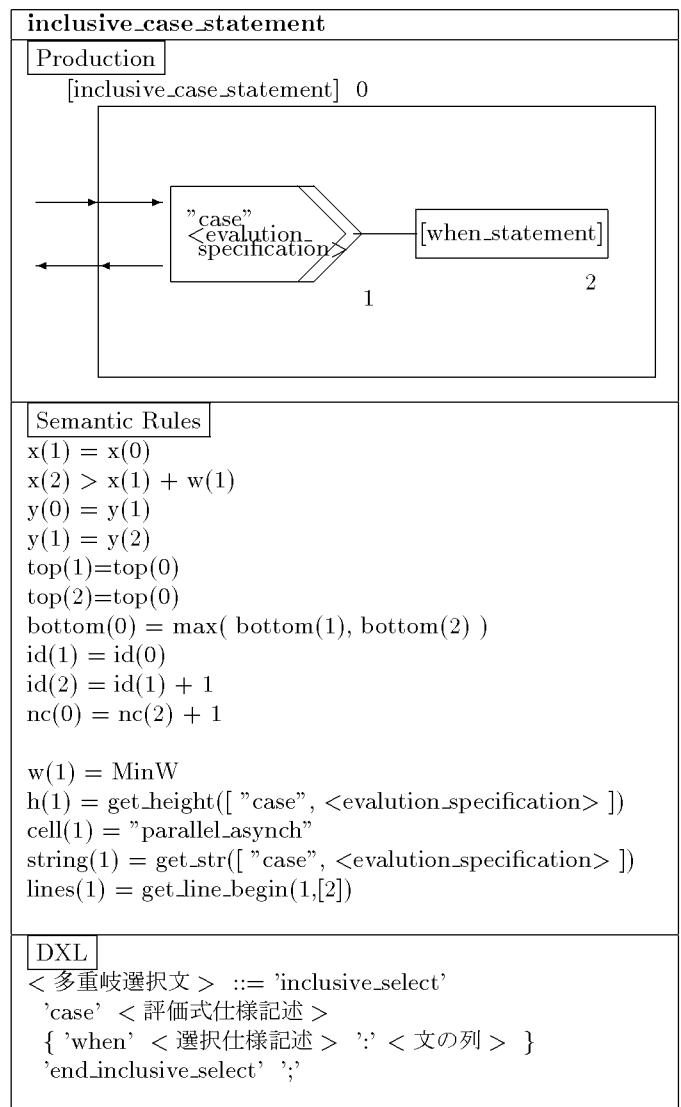
DXL – Production Rule – 60

else_if_statement
<p>Production</p> <pre>[else_if_statement] 0 --> "else_if" <conditional_specification> "then" --> T: [branch_statement_list] 2 --> F: [else_if_statement] 1</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) x(2) > x(1) + w(1) y(0) = y(1) y(1) = y(2) top(1)=top(0) top(2)=top(0) bottom(0) = max(bottom(1), bottom(2)) bottom(1) = y(1) + h(1) cl(2) = "T." id(1) = id(0) id(2) = id(1) + 1 nc(0) = nc(2) + 1 w(1) = MinW h(1) = get_height(["else_if", <conditional_specification>, "then"]) cell(1) = "exclusive_selection" string(1) = get_str(["else_if", <conditional_specification>, "then"]) lines(1) = get_line_begin(1,[2])</pre>
<p>DXL</p> <pre><if 型選択文 > ::= 'exclusive_select' 'if' < 条件仕様記述 > 'then' < 文の列 > { 'else_if' < 条件仕様記述 > 'then' < 文の列 > } ['else' < 文の列 >] 'end_exclusive_select' ';'</pre>

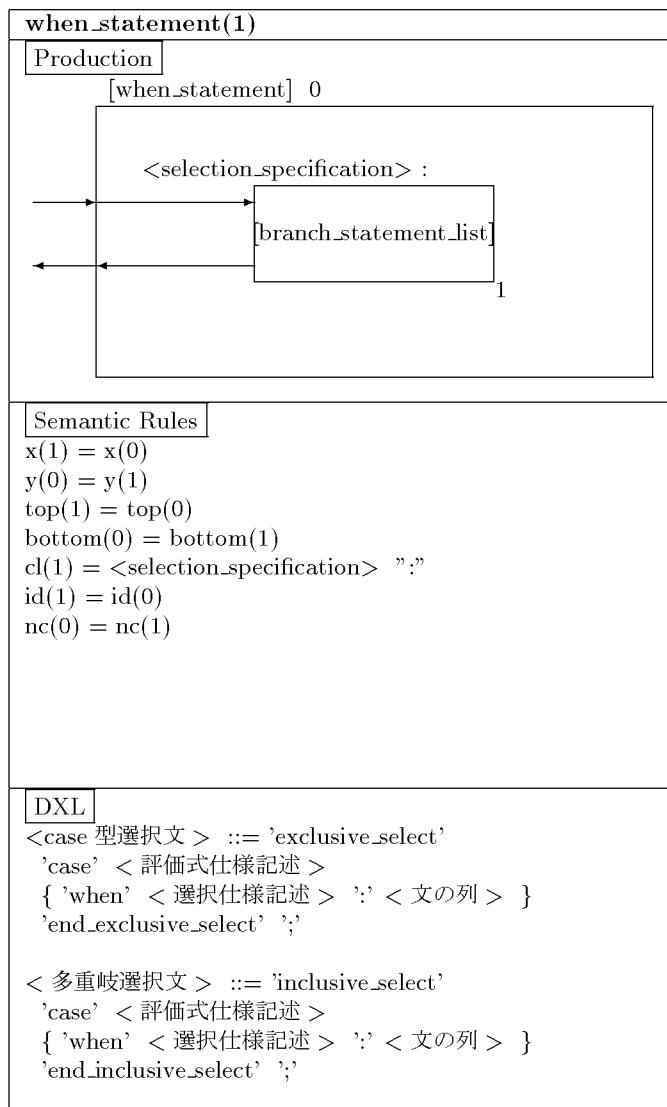
DXL – Production Rule – 61



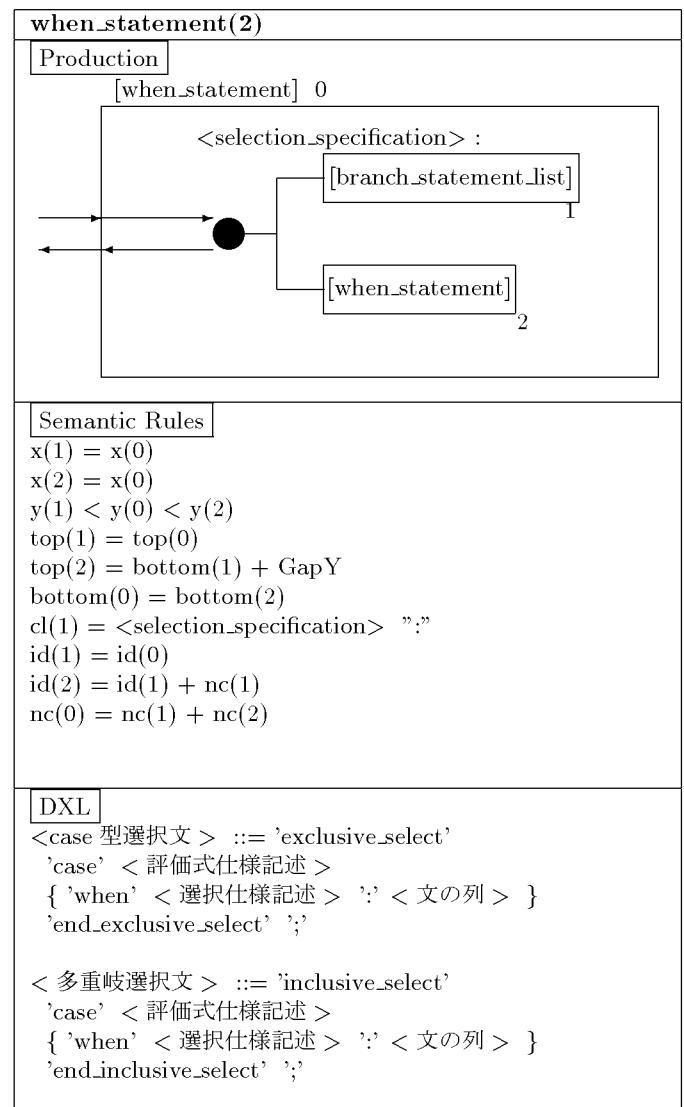
DXL – Production Rule – 62



DXL – Production Rule – 63



DXL – Production Rule – 64



DXL – Production Rule – 65

terminate_statement(1)
<p>Production</p> <pre>[terminate_statement] 0</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) y(0) = y(1) top(1) = top(0) bottom(0) = bottom(1) id(1) = id(0) nc(0) = nc(1) w(1) = get_width(["terminate", "system", <specification>]) h(1) = get_height(["terminate", "system", <specification>]) cell(1) = "terminal" string(1) = get_str(["terminate", "system", <specification>])</pre>
<p>DXL</p> <pre><打ち切り文 > ::= 'terminate' <打ち切り対象 > <仕様記述 > ; <打ち切り対象 > ::= 'system' 'module' 'block' < ブロック識別子 ></pre>

DXL – Production Rule – 66

terminate_statement(2)
<p>Production</p> <pre>[terminate_statement] 0</pre>
<p>Semantic Rules</p> <pre>x(1) = x(0) y(0) = y(1) top(1) = top(0) bottom(0) = bottom(1) id(1) = id(0) nc(0) = nc(1) w(1) = get_width(["terminate", "module", <specification>]) h(1) = get_height(["terminate", "module", <specification>]) cell(1) = "terminal" string(1) = get_str(["terminate", "module", <specification>])</pre>
<p>DXL</p> <pre><打ち切り文 > ::= 'terminate' <打ち切り対象 > <仕様記述 > ; <打ち切り対象 > ::= 'system' 'module' 'block' < ブロック識別子 ></pre>

terminate_statement(3)			
<table border="1"> <tr> <td>Production</td> </tr> <tr> <td>[terminate_statement] 0</td> </tr> <tr> <td> </td> </tr> </table>	Production	[terminate_statement] 0	
Production			
[terminate_statement] 0			
<table border="1"> <tr> <td>Semantic Rules</td> </tr> <tr> <td> $x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$ $w(1) = \text{get_width}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$ $h(1) = \text{get_height}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$ $\text{cell}(1) = \text{"terminal"}$ $\text{string}(1) = \text{get_str}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$ </td> </tr> </table>	Semantic Rules	$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$ $w(1) = \text{get_width}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$ $h(1) = \text{get_height}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$ $\text{cell}(1) = \text{"terminal"}$ $\text{string}(1) = \text{get_str}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$	
Semantic Rules			
$x(1) = x(0)$ $y(0) = y(1)$ $\text{top}(1) = \text{top}(0)$ $\text{bottom}(0) = \text{bottom}(1)$ $\text{id}(1) = \text{id}(0)$ $\text{nc}(0) = \text{nc}(1)$ $w(1) = \text{get_width}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$ $h(1) = \text{get_height}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$ $\text{cell}(1) = \text{"terminal"}$ $\text{string}(1) = \text{get_str}([\text{"terminate"}, \text{"block"}, \langle \text{block_identifier} \rangle, \langle \text{specification} \rangle])$			
<table border="1"> <tr> <td>DXL</td> </tr> <tr> <td> $<\text{打ち切り文}> ::= \text{'terminate'} \langle \text{打ち切り対象} \rangle \langle \text{仕様記述} \rangle ;$ $\langle \text{打ち切り対象} \rangle ::= \text{'system'} \mid \text{'module'} \mid \text{'block'} \langle \text{ブロック識別子} \rangle$ </td> </tr> </table>	DXL	$<\text{打ち切り文}> ::= \text{'terminate'} \langle \text{打ち切り対象} \rangle \langle \text{仕様記述} \rangle ;$ $\langle \text{打ち切り対象} \rangle ::= \text{'system'} \mid \text{'module'} \mid \text{'block'} \langle \text{ブロック識別子} \rangle$	
DXL			
$<\text{打ち切り文}> ::= \text{'terminate'} \langle \text{打ち切り対象} \rangle \langle \text{仕様記述} \rangle ;$ $\langle \text{打ち切り対象} \rangle ::= \text{'system'} \mid \text{'module'} \mid \text{'block'} \langle \text{ブロック識別子} \rangle$			