

0.1 EBNF Grammatik für DMF

$\langle \text{source_file} \rangle$	$::= \langle \text{dmf_declaration} \rangle \langle \text{new_line} \rangle \langle \text{model_declaration} \rangle \langle \text{new_line} \rangle$ $[\langle \text{import_block} \rangle] \langle \text{model_content} \rangle$
$\langle \text{dmf_declaration} \rangle$	$::= \text{'dmf'} \langle \text{version_number} \rangle$
$\langle \text{model_declaration} \rangle$	$::= \text{'model'} \langle \text{string_value} \rangle \text{'version'} \langle \text{version_number} \rangle$
$\langle \text{import_block} \rangle$	$::= \langle \text{import_statement} \rangle^+$
$\langle \text{import_statement} \rangle$	$::= \text{'import'} \langle \text{package_string} \rangle \text{'from'} \langle \text{string_value} \rangle \langle \text{new_line} \rangle$
$\langle \text{model_content} \rangle$	$::= \langle \text{package_content} \rangle^+$
$\langle \text{package_content} \rangle$	$::= [\langle \text{comment_block} \rangle] [\text{'expand'}] \langle \text{package_block} \rangle [\langle \text{override_block} \rangle]$ $[\langle \text{comment_block} \rangle] [\text{'expand'}] \langle \text{struct_block} \rangle [\langle \text{override_block} \rangle]$ $[\langle \text{comment_block} \rangle] [\text{'expand'}] \langle \text{enum_block} \rangle [\langle \text{override_block} \rangle]$ $[\langle \text{comment_block} \rangle] [\text{'expand'}] \langle \text{entity_block} \rangle [\langle \text{override_block} \rangle]$ $[\langle \text{comment_block} \rangle] [\text{'expand'}] \langle \text{interface_block} \rangle [\langle \text{override_block} \rangle]$
$\langle \text{comment_block} \rangle$	$::= \langle \text{comment} \rangle^+$
$\langle \text{comment} \rangle$	$::= \text{R'//.*\n'}$
$\langle \text{override_block} \rangle$	$::= \text{'override' '{' } (\langle \text{java_override} \rangle \langle \text{typescript_override} \rangle)^* \text{'}'}$
$\langle \text{java_override} \rangle$	$::= \text{'java' '{' } (\langle \text{java_annotation} \rangle \langle \text{java_extends} \rangle \langle \text{java_implements} \rangle$ $\langle \text{java_class} \rangle \langle \text{java_name} \rangle \langle \text{java_type} \rangle \langle \text{java_doc} \rangle)^* \text{'}'}$
$\langle \text{java_annotation} \rangle$	$::= \text{'annotations' } \langle \text{stringValue} \rangle$
$\langle \text{java_doc} \rangle$	$::= \text{'javaDoc' } \langle \text{stringValue} \rangle$
$\langle \text{java_extends} \rangle$	$::= \text{'extends' } \langle \text{stringValue} \rangle$
$\langle \text{java_implements} \rangle$	$::= \text{'implements' } \langle \text{stringValue} \rangle$
$\langle \text{java_class} \rangle$	$::= \text{'class' } \langle \text{stringValue} \rangle$

$\langle \text{java_name} \rangle$::= 'name' $\langle \text{stringValue} \rangle$
$\langle \text{java_type} \rangle$::= 'type' $\langle \text{stringValue} \rangle$
$\langle \text{package_block} \rangle$::= 'package' '{' $\langle \text{package_content} \rangle^*$ '}'
$\langle \text{struct_block} \rangle$::= 'struct' $\langle \text{identifier} \rangle$ [$\langle \text{extends_block} \rangle$] [$\langle \text{implements_block} \rangle$] '{' $\langle \text{struct_content} \rangle^*$ '}'
$\langle \text{extends_block} \rangle$::= 'extends' $\langle \text{reftype} \rangle$
$\langle \text{implements_block} \rangle$::= 'implements' $\langle \text{reftype} \rangle$ (',' $\langle \text{reftype} \rangle$)+
$\langle \text{struct_content} \rangle$::= [$\langle \text{comment_block} \rangle$] $\langle \text{arg_block} \rangle$ [$\langle \text{override_block} \rangle$] [$\langle \text{comment_block} \rangle$] $\langle \text{ref_block} \rangle$ [$\langle \text{override_block} \rangle$] [$\langle \text{comment_block} \rangle$] $\langle \text{multi_block} \rangle$ [$\langle \text{override_block} \rangle$] [$\langle \text{comment_block} \rangle$] $\langle \text{func_block} \rangle$ [$\langle \text{override_block} \rangle$]
$\langle \text{arg_block} \rangle$::= 'arg' $\langle \text{primitive_type} \rangle$ $\langle \text{identifier} \rangle$ ';' ;
$\langle \text{ref_block} \rangle$::= 'ref' $\langle \text{reftype} \rangle$ $\langle \text{identifier} \rangle$ ';' ;
$\langle \text{multi_block} \rangle$::= 'ref' $\langle \text{multi_name} \rangle$ '<' $\langle \text{primitive_type} \rangle$ [',' $\langle \text{primitive_type} \rangle$] >' $\langle \text{identifier} \rangle$ ';' ; 'ref' $\langle \text{multi_name} \rangle$ '<' $\langle \text{reftype} \rangle$ [',' $\langle \text{primitive_type} \rangle$] '>' $\langle \text{identifier} \rangle$ ';' ; 'ref' $\langle \text{multi_name} \rangle$ '<' $\langle \text{primitive_type} \rangle$ [',' $\langle \text{reftype} \rangle$] '>' $\langle \text{identifier} \rangle$ ';' ; 'ref' $\langle \text{multi_name} \rangle$ '<' $\langle \text{reftype} \rangle$ [',' $\langle \text{reftype} \rangle$] '>' $\langle \text{identifier} \rangle$ ';' ;
$\langle \text{func_block} \rangle$::= 'func' $\langle \text{reftype} \rangle$ $\langle \text{identifier} \rangle$ '(' [$\langle \text{param_definition} \rangle$ (',' $\langle \text{param_definition} \rangle$)*] ')' ';' ; 'func' $\langle \text{primitive_type} \rangle$ $\langle \text{identifier} \rangle$ '(' [$\langle \text{param_definition} \rangle$ (',' $\langle \text{param_definition} \rangle$)*] ')' ';' ; 'func' 'void' $\langle \text{identifier} \rangle$ '(' [$\langle \text{param_definition} \rangle$ (',' $\langle \text{param_definition} \rangle$)*] ')' ';' ;
$\langle \text{param_definition} \rangle$::= $\langle \text{reftype} \rangle$ $\langle \text{identifier} \rangle$ $\langle \text{primitive_type} \rangle$ $\langle \text{identifier} \rangle$

$\langle \text{enum_block} \rangle$	$::= \text{'enum' } \langle \text{identifier} \rangle \text{'{' } } \langle \text{enum_content} \rangle^* \text{'}'$
$\langle \text{enum_content} \rangle$	$::= [\langle \text{comment_block} \rangle] \langle \text{arg_block} \rangle [\langle \text{override_block} \rangle]$ $\quad [\langle \text{comment_block} \rangle] \langle \text{enum_constant} \rangle [\langle \text{override_block} \rangle]$
$\langle \text{enum_constant} \rangle$	$::= \langle \text{identifier} \rangle \text{'(' } \langle \text{enum_index} \rangle \text{' ' } \langle \text{primitive_value} \rangle \text{' ' } \text{'};'$
$\langle \text{enum_index} \rangle$	$::= \text{'_' } \langle \text{integerValue} \rangle$
$\langle \text{entity_block} \rangle$	$::= \text{'entity' } \langle \text{identifier} \rangle [\langle \text{extends_block} \rangle] [\langle \text{implements_block} \rangle] \text{'{' } }$ $\quad \langle \text{struct_content} \rangle^* \langle \text{identifier_statement} \rangle \text{'}'$
$\langle \text{identifier_statement} \rangle$	$::= \text{'identifier' '(' } \langle \text{identifier} \rangle \text{'(' } \langle \text{identifier} \rangle \text{' ' } \langle \text{identifier} \rangle^* \text{' ' } \text{'};'$
$\langle \text{interface_block} \rangle$	$::= \text{'interface' } \langle \text{identifier} \rangle [\langle \text{implements_block} \rangle] \text{'{' } } \langle \text{interface_content} \rangle^*$ $\quad \text{'}'$
$\langle \text{interface_content} \rangle$	$::= [\langle \text{comment_block} \rangle] \langle \text{func_block} \rangle [\langle \text{override_block} \rangle]$
$\langle \text{reftype} \rangle$	$::= \langle \text{dot} \rangle^* \langle \text{package_string} \rangle$
$\langle \text{number} \rangle$	$::= \text{R'[0-9]+'}$
$\langle \text{identifier} \rangle$	$::= \text{R'([a-zA-Z_])+'}$
$\langle \text{dot} \rangle$	$::= \text{'.'}$
$\langle \text{byte_content} \rangle$	$::= \text{R'[0-9A-F]{4}'}$
$\langle \text{new_line} \rangle$	$::= \text{R'\n'}$
$\langle \text{package_string} \rangle$	$::= \langle \text{identifier} \rangle$ $\quad \langle \text{identifier} \rangle (\langle \text{dot} \rangle \langle \text{identifier} \rangle)^+$ $\quad (\langle \text{dot} \rangle \langle \text{identifier} \rangle)^+$
$\langle \text{primitive_type} \rangle$	$::= \text{'byte' } \text{'int' } \text{'long' } \text{'double' } \text{'datetime' } \text{'date' } \text{'string' } $ $\quad \text{'boolean'}$
$\langle \text{primitive_value} \rangle$	$::= \langle \text{integerValue} \rangle$ $\quad \langle \text{doubleValue} \rangle$ $\quad \langle \text{stringValue} \rangle$ $\quad \langle \text{dateValue} \rangle$

	<i><dateTimeValue></i> <i><booleanValue></i> <i><byteValue></i> <i><longValue></i>
<i><version_number></i>	::= <i><number></i> <i><dot></i> <i><number></i> <i><dot></i> <i><number></i>
<i><integerValue></i>	::= <i><number></i> [<i>'_'</i> <i><number></i>]
<i><longValue></i>	::= <i><integerValue></i> <i>'L'</i>
<i><booleanValue></i>	::= <i>'true'</i> <i>'false'</i>
<i><byteValue></i>	::= <i>'0x'</i> <i><byte_content></i>
<i><doubleValue></i>	::= <i><integerValue></i> <i><dot></i> <i><integerValue></i>
<i><stringValue></i>	::= <i>""</i> (<i><string_content_double_quote></i> <i><escape_sequence></i>)* <i>""</i> <i>'''</i> (<i><string_content_single_quote></i> <i><escape_sequence></i>)* <i>'''</i>
<i><string_content_single_quote></i>	::= <i>R'^\'\'^+'</i>
<i><string_content_double_quote></i>	::= <i>R'^\"^+'</i>
<i><escape_sequence></i>	::= <i>'\'</i> <i>R'[\\"bfnr]</i> <i>'\'</i> <i>R'u[0-9a-fA-F]{4}'</i> <i>'\'</i> <i>R'x[0-9a-fA-F]{2}'</i> <i>'\'</i> <i>R'[0-7]{1,3}'</i>
<i><dateTimeValue></i>	::= <i><dateValue></i> <i>'T'</i> <i><timeValue></i>
<i><timeValue></i>	::= <i>R'(?:[01][0-9] 2[0-3])' ':' R'[0-5][0-9]' ':' R'[0-5][0-9]'</i>
<i><dateValue></i>	::= <i>'D'</i> <i>R'[0-9]{4}' '-' R'(?:[1-9] 1[0-2])' '-' R'(?:[01-9] 1[0-9] 2[0-9] 3[01])'</i>

0.2 Beispiel

<i><statement></i>	::= <i><ident></i> <i>'='</i> <i><expr></i> <i>'for'</i> <i><ident></i> <i>'='</i> <i><expr></i> <i>'to'</i> <i><expr></i> <i>'do'</i> <i><statement></i> <i>'{'</i> <i><stat-list></i> <i>'}'</i> <i><empty></i>
--------------------------	--

$\langle \text{stat-list} \rangle ::= \langle \text{statement} \rangle \text{';'} \langle \text{stat-list} \rangle \mid \langle \text{statement} \rangle$