# A.2 Modified TTCN-3 syntax BNF productions

This clause includes all BNF productions that are modifications of BNF rules defined in the TTCN-3 core language document ETSI ES 201 873-1 [1]. When using this package the BNF rules below replace the corresponding BNF rules in the TTCN-3 core language document. The rule numbers define the correspondence of BNF rules.

Additions to clause A.1.6 of ETSI ES 201 873-1 are identified by underlined font, deletions are identified by strikethrough font. In case of contradiction between the above clause of part 1 and this clause (i.e. parts of the productions not marked by ~~strikethrough~~ font is changed in part 1), part 1 takes precedence, i.e. tools supporting the present document shall apply the insertions and deletions of this clause to the actual part 1 production or static semantics rule automtically.

BNF changes in clause A.1.6.1.1 Type definitions

12. StructuredTypeDef ::= RecordDef |

UnionDef |

SetDef |

RecordOfDef |

SetOfDef |

EnumDef |

PortDef |

ComponentDef |

MapDef |

ClassDef

BNF Changes in clause A.1.6.1.3 Template Definitions

100.FieldSpec ::= ( FieldReference | FunctionInstance ) AssignmentChar (TemplateBody | Minus)

BNF changes in clause A.1.6.1.4 Function definitions

161. FunctionDef ::= FunctionKeyword [ DeterministicModifier | ControlModifier ]

IdentifierOrControl "(" [ FunctionFormalParList ] ")"

[ RunsOnSpec ] [ MtcSpec ] [ SystemSpec ] [ ReturnType ] [ ExceptionSpec ]

StatementBlock

172. StatementBlock ::= BasicStatementBlock [ CatchBlocks ] [ FinallyBlock ]

179. FunctionRef ::= [ ( Identifier | ObjectInstance ) Dot ]

( Identifier | PreDefFunctionIdentifier | ControlKeyword )

BNF changes in clause A.1.6.1.7 Altstep definitions

195. AltstepDef ::= AltstepKeyword [ ControlModifier ] [ InterleaveKeyword ]

Identifier "(" [ FunctionFormalParList ] ")"

[ RunsOnSpec ] [ MtcSpec ] [ SystemSpec ] [ ExceptionSpec ]

"{" AltstepLocalDefList AltGuardList "}" [ CatchBlocks ] [ FinallyBlock ]

BNF changes in clause A.1.6.1.10 External function definitions

247. ExtFunctionDef ::= ExtKeyword FunctionKeyword [ DeterministicModifier | ControlModifier ]

Identifier "(" [ FunctionFormalParList ] ")"

[ ReturnType ] [ ExceptionSpec ]

BNF changes in clause A.1.6.3.1 Variable instantiation

259.SingleVarInstance ::= [PropertyModifier] Identifier [ArrayDef]

[AssignmentChar Expression] [PropertyBody]

/\* STATIC PropertyBody shall only be present if PropertyModifier is also present \*/

262.SingleTempVarInstance ::= [PropertyModifier] Identifier [ArrayDef]

[AssignmentChar TemplateBody] [PropertyBody]  
/\* STATIC PropertyBody shall only be present if PropertyModifier is also present \*/

263. ValueRef ::= [ ThisOp Dot ] Identifier [ExtendedFieldReference]

BNF changes in clause A.1.6.4.1 Component Operations

269. CreateOp ::= Type Dot CreateKeyword [ActualParList]

[AliveKeyword |

ExternalKeyword ActualParList]

BNF changes in clause A.1.6.5 Type

405. PredefinedType ::= BitStringKeyword |

BooleanKeyword |

CharStringKeyword |

UniversalCharString |

IntegerKeyword |

OctetStringKeyword |

HexStringKeyword |

VerdictTypeKeyword |

FloatKeyword |

AddressKeyword |

DefaultKeyword |

AnyTypeKeyword |

TimerKeyword |

ObjectKeyword

BNF changes in clause A.1.6.6 Value

437. ReferencedValue ::= ( ( ExtendedIdentifier | ThisOp ) [ExtendedFieldReference] )

| ReferencedEnumValue

BNF changes in clause A.1.6.8.2 Behaviour statements

492. BehaviourStatements ::= TestcaseInstance |

FunctionInstance |

ReturnStatement |

AltConstruct |

InterleavedConstruct |

LabelStatement |

GotoStatement |

RepeatStatement |

DeactivateStatement |

AltstepInstance |

ActivateOp |

BreakStatement |

ContinueStatement |

RaiseExceptionStatement

BNF changes in clause A.1.6.8.3 Basic statements

562. RelOp ::= "<" | ">" | ">=" | "<=" | OfKeyword

# A.3 Additional TTCN-3 syntax BNF productions

This clause includes all additional BNF productions that needed to define the syntax introduced by this package. All rules start with the digits "033". The numbering of other rules start with number 033001.

Additional BNF rules related to clause A.1.6.1.1 Type definitions

033001. ClassDef ::= [ ExtKeyword ] ClassKeyword [ FinalModifier | AbstractModifier | TraitModifier]

Identifier [ ExtendsKeyword ClassTypeList ] [ RunsOnSpec ] [ MtcSpec ]

[ SystemSpec ]

"{" ClassMemberList "}"

[ FinallyKeyword BasicStatementBlock ]

033002. ClassKeyword ::= "class"

033003. ThisOp ::= "this"

033004. SuperOp ::= "super"

033005. FinalModifier ::= "@final"

033006. AbstractModifier ::= "@abstract"

033007. FinallyKeyword ::= "finally"

033008. ObjectKeyword ::= "object"

033008a. ClassType ::= ReferencedType | ObjectKeyword

/\* STATIC SEMANTICS – ReferencedType shall evaluate to a class. \*/

033009. ClassMemberList ::= { ClassMember [ WithStatement ] [ SemiColon ] }

033010. ClassMember ::= [ MemberVisibility ]

( VarInstance |

TimerInstance |

ClassConstDef |

ClassTemplateDef |

ClassFunctionDef |

ConstructorDef |

ClassDef )

033011. MemberVisibility ::= "public" | "private"

033012. ClassFunctionDef ::= [ ExtKeyword ] FunctionKeyword

[ FinalModifier | AbstractModifier ] [ DeterministicModifier ]

Identifier "(" [ FunctionFormalParList ] ")" [ ReturnType ]

[ StatementBlock ]

033013. ConstructorDef ::= CreateKeyword

"(" FunctionFormalParList ")"

[ ExternalKeyword "(" FunctionFormalParList ")" ]

[ ":" ReferencedType ActualParList ]

[ StatementBlock ]

/\* STATIC SEMANTICS – ReferencedType shall evaluate to a class. \*/

033013a. ClassConstDef ::= ConstKeyword Type ClassConstList

033013a ClassConstList ::= SingleClassConstDef {"," SingleClassConstDef}

033013a SingleClassConstDef ::= Identifier [ArrayDef] [ AssignmentChar ConstantExpression ]

033013b. ClassTemplateDef ::= TemplateKeyword [TemplateRestriction]

[FuzzyModifier [DeterministicModifier]]

BaseTemplate [DerivedDef] [ AssignmentChar BaseTemplateBody ]

033014. TraitModifier ::= "@trait"

003015. ClassTypeList ::= ClassType { "," ClassType }

Additional BNF rules related to clause A.1.6.1.4 Function definitions

033016. BasicStatementBlock ::= "{" [ FunctionDefList ] [ FunctionStatementList ] "}"

033017. CatchBlocks ::= CatchBlock { CatchBlock }

033018. CatchBlock ::= CatchOpKeyword "(" Type Identifier ")" BasicStatementBlock

033019. FinallyBlock ::= FinallyKeyword BasicStatementBlock

033020. ObjectInstance ::= ( ThisOp | ValueRef | FunctionInstance ) [ ExtendedFieldReference ]

Additional BNF related to clause A.1.6.3.1 Variable Instantiation

033021. PropertyModifier ::= "@property"

033022. PropertyBody ::= "{" ( PropertyGetterDef [PropertySetterDef] |

PropertySetterDef [PropertyGetterDef] ) "}"

033023. PropertyGetterDef ::= ["public"| "private"]

[AbstractModifier | FinalModifier] [DeterministicModifier]

"@get" [ "=>" TemplateBody | *StatementBlock* ] [";"]

033024. PropertySetterDef ::= ["public"| "private"]

[AbstractModifier | FinalModifier] [DeterministicModifier]

"@set" [ "=>" Assignment | StatementBlock] [";"]

Additional BNF rules related to clause A.1.6.8.2 Behaviour statements

033025. RaiseExceptionStatement ::= RaiseKeyword TemplateInstance

/\* STATIC SEMANTICS - The TemplateInstance shall evaluate to an explicit value. \*/