**Specification**

USPD: US.ECO.00010-01 90

Component: IEcoAST1

UGUID:

Status: Draft

Date: November 8, 2021

Version: 1.0

|  |  |
| --- | --- |
| **Authors** | **Company** |
| Vladimir Bashev | PEERF |
|  |  |
|  |  |

Content

[**1.** **Overview** 11](#_Toc180967356)

[**1.1.** **Introduction** 11](#_Toc180967357)

[**1.2.** **Note** 11](#_Toc180967358)

[**1.3.** **Links** 11](#_Toc180967359)

[**2.** **Eco.** **AST1 Component** 12](#_Toc180967360)

[**3.** **IEcoAST1 Interface** 13](#_Toc180967361)

[**3.1.** **IEcoAST1 IDL** 13](#_Toc180967362)

[**3.1.1.** **new\_NumberType function** 19](#_Toc180967363)

[**3.1.2.** **new\_StructOrUnionType function** 19](#_Toc180967364)

[**3.1.3.** **new\_EnumType function** 19](#_Toc180967365)

[**3.1.4.** **new\_ArrayType function** 19](#_Toc180967366)

[**3.1.5.** **new\_PointerType function** 19](#_Toc180967367)

[**3.1.6.** **new\_IntegerLiteral function** 19](#_Toc180967368)

[**3.1.7.** **new\_CharacterLiteral function** 19](#_Toc180967369)

[**3.1.8.** **new\_FloatLiteral function** 19](#_Toc180967370)

[**3.1.9.** **new\_StringLiteral function** 19](#_Toc180967371)

[**3.1.10.** **new\_VariableDefinition function** 20](#_Toc180967372)

[**3.1.11.** **new\_BinaryExpression function** 20](#_Toc180967373)

[**3.1.12.** **new\_MemberSelection function** 20](#_Toc180967374)

[**3.1.13.** **new\_UnaryExpression function** 20](#_Toc180967375)

[**3.1.14.** **new\_CastExpression function** 20](#_Toc180967376)

[**3.1.15.** **new\_ConditionalExpression function** 20](#_Toc180967377)

[**3.1.16.** **new\_SizeofType function** 20](#_Toc180967378)

[**3.1.17.** **new\_SizeofExpression function** 20](#_Toc180967379)

[**3.1.18.** **new\_NullExpression function** 20](#_Toc180967380)

[**3.1.19.** **new\_BlockExpressions function** 20](#_Toc180967381)

[**3.1.20.** **new\_BreakStatement function** 21](#_Toc180967382)

[**3.1.21.** **new\_ContinueStatement function** 21](#_Toc180967383)

[**3.1.22.** **new\_ReturnStatement function** 21](#_Toc180967384)

[**3.1.23.** **new\_GoToStatement function** 21](#_Toc180967385)

[**3.1.24.** **new\_IfStatement function** 21](#_Toc180967386)

[**3.1.25.** **new\_ForStatement function** 21](#_Toc180967387)

[**3.1.26.** **new\_WhileStatement function** 21](#_Toc180967388)

[**3.1.27.** **new\_DoWhileStatement function** 21](#_Toc180967389)

[**3.1.28.** **new\_SwitchStatement function** 21](#_Toc180967390)

[**3.1.29.** **new\_DefaultStatement function** 22](#_Toc180967391)

[**3.1.30.** **new\_CaseStatement function** 22](#_Toc180967392)

[**3.1.31.** **new\_LabelStatement function** 22](#_Toc180967393)

[**3.1.32.** **new\_BlockStatements function** 22](#_Toc180967394)

[**3.1.33.** **new\_BlockDeclarations function** 22](#_Toc180967395)

[**3.1.34.** **new\_BlockTranslationUnitObjects function** 22](#_Toc180967396)

[**3.1.35.** **new\_VariableDeclaration function** 22](#_Toc180967397)

[**3.1.36.** **new\_StructOrUnionDeclaration function** 22](#_Toc180967398)

[**3.1.37.** **new\_EnumDeclaration function** 22](#_Toc180967399)

[**3.1.38.** **new\_CompoundStatement function** 22](#_Toc180967400)

[**3.1.39.** **new\_FunctionType function** 23](#_Toc180967401)

[**3.1.40.** **new\_BlockTypes function** 23](#_Toc180967402)

[**3.1.41.** **new\_FunctionDeclaration function** 23](#_Toc180967403)

[**3.1.42.** **new\_FunctionDefinition function** 23](#_Toc180967404)

[**3.1.43.** **new\_FunctionCall function** 23](#_Toc180967405)

[**3.1.44.** **new\_TypedefDeclaration function** 23](#_Toc180967406)

[**3.1.45.** **new\_TypedefType function** 23](#_Toc180967407)

[**3.1.46.** **new\_GlobalScope function** 23](#_Toc180967408)

[**3.1.47.** **new\_Scope function** 23](#_Toc180967409)

[**3.1.48.** **new\_TranslationUnit function** 24](#_Toc180967410)

[**3.1.49.** **copy\_Object function** 24](#_Toc180967411)

[**3.1.50.** **get\_TranslationUnit) function** 24](#_Toc180967412)

[**4.** **IEcoAST1SyntaxObject Interface** 25](#_Toc180967413)

[**4.1.** **IEcoAST1SyntaxObject IDL** 25](#_Toc180967414)

[**4.1.1.** **get\_Id function** 26](#_Toc180967415)

[**4.1.2.** **get\_Token function** 26](#_Toc180967416)

[**4.1.3.** **get\_NameObject function** 26](#_Toc180967417)

[**4.1.4.** **CopyToAST function** 26](#_Toc180967418)

[**5.** **IEcoAST1BlockDeclarations Interface** 27](#_Toc180967419)

[**5.1.** **IEcoAST1BlockDeclarations IDL** 27](#_Toc180967420)

[**5.1.1.** **get\_Declaration function** 28](#_Toc180967421)

[**5.1.2.** **get\_Count function** 28](#_Toc180967422)

[**6.** **IEcoAST1BlockTypes Interface** 29](#_Toc180967423)

[**6.1.** **IEcoAST1BlockTypes IDL** 29](#_Toc180967424)

[**6.1.1.** **get\_Type function** 30](#_Toc180967425)

[**6.1.2.** **get\_Count function** 30](#_Toc180967426)

[**6.1.3.** **IsVariadic function** 30](#_Toc180967427)

[**7.** **IEcoAST1Declaration Interface** 31](#_Toc180967428)

[**7.1.** **IEcoAST1Declaration IDL** 31](#_Toc180967429)

[**7.1.1.** **get\_IdentifierName function** 32](#_Toc180967430)

[**8.** **IEcoAST1VariableDeclaration Interface** 33](#_Toc180967431)

[**8.1.** **IEcoAST1VariableDeclaration IDL** 33](#_Toc180967432)

[**8.1.1.** **get\_Initializator function** 34](#_Toc180967433)

[**9.** **IEcoAST1ExpressionableDeclaration Interface** 35](#_Toc180967434)

[**9.1.** **IEcoAST1ExpressionableDeclaration IDL** 35](#_Toc180967435)

[**9.1.1.** **get\_Type function** 36](#_Toc180967436)

[**10.** **IEcoAST1FunctionDeclaration Interface** 37](#_Toc180967437)

[**10.1.** **IEcoAST1FunctionDeclaration IDL** 37](#_Toc180967438)

[**10.1.1.** **get\_Params function** 38](#_Toc180967439)

[**10.1.2.** **get\_ReturnType function** 38](#_Toc180967440)

[**11.** **IEcoAST1EnumDeclaration Interface** 39](#_Toc180967441)

[**11.1.** **IEcoAST1EnumDeclaration IDL** 39](#_Toc180967442)

[**11.1.1.** **get\_Declarations function** 40](#_Toc180967443)

[**11.1.2.** **get\_Scope function** 40](#_Toc180967444)

[**12.** **IEcoAST1TypedefDeclaration Interface** 41](#_Toc180967445)

[**12.1.** **IEcoAST1TypedefDeclaration IDL** 41](#_Toc180967446)

[**12.1.1.** **get\_SourceType function** 42](#_Toc180967447)

[**13.** **IEcoAST1StructOrUnionDeclaration Interface** 43](#_Toc180967448)

[**13.1.** **IEcoAST1StructOrUnionDeclaration IDL** 43](#_Toc180967449)

[**13.1.1.** **get\_Params function** 44](#_Toc180967450)

[**13.1.2.** **get\_StructOrUnionType function** 44](#_Toc180967451)

[**13.1.3.** **get\_Scope function** 44](#_Toc180967452)

[**13.1.4.** **get\_Sizes function** 44](#_Toc180967453)

[**14.** **IEcoAST1TranslationUnit Interface** 45](#_Toc180967454)

[**14.1.** **IEcoAST1TranslationUnit IDL** 45](#_Toc180967455)

[**14.1.1.** **function** 46](#_Toc180967456)

[**15.** **IEcoAST1Statement Interface** 47](#_Toc180967457)

[**15.1.** **IEcoAST1Statement IDL** 47](#_Toc180967458)

[**15.1.1.** **The function function** 48](#_Toc180967459)

[**16.** **IEcoAST1BlockStatements Interface** 49](#_Toc180967460)

[**16.1.** **IEcoAST1BlockStatements IDL** 49](#_Toc180967461)

[**16.1.1.** **get\_Statement function** 50](#_Toc180967462)

[**16.1.2.** **get\_Count function** 50](#_Toc180967463)

[**17.** **IEcoAST1DoWhileStatement Interface** 51](#_Toc180967464)

[**17.1.** **IEcoAST1DoWhileStatement IDL** 51](#_Toc180967465)

[**17.1.1.** **get\_TestExpression function** 52](#_Toc180967466)

[**17.1.2.** **get\_InternalStatement function** 52](#_Toc180967467)

[**18.** **IEcoAST1WhileStatement Interface** 53](#_Toc180967468)

[**18.1.** **IEcoAST1WhileStatement IDL** 53](#_Toc180967469)

[**18.1.1.** **get\_TestExpression function** 54](#_Toc180967470)

[**18.1.2.** **get\_InternalStatement function** 54](#_Toc180967471)

[**19.** **IEcoAST1DefaultStatement Interface** 55](#_Toc180967472)

[**19.1.** **IEcoAST1DefaultStatement IDL** 55](#_Toc180967473)

[**19.1.1.** **get\_InternalStatement function** 56](#_Toc180967474)

[**20.** **IEcoAST1BreakStatement Interface** 57](#_Toc180967475)

[**20.1.** **IEcoAST1BreakStatement IDL** 57](#_Toc180967476)

[**20.1.1.** **function** 58](#_Toc180967477)

[**21.** **IEcoAST1CaseStatement Interface** 59](#_Toc180967478)

[**21.1.** **IEcoAST1CaseStatement IDL** 59](#_Toc180967479)

[**21.1.1.** **get\_Expression function** 60](#_Toc180967480)

[**21.1.2.** **get\_Statement function** 60](#_Toc180967481)

[**22.** **IEcoAST1LabelStatement Interface** 61](#_Toc180967482)

[**22.1.** **IEcoAST1LabelStatement IDL** 61](#_Toc180967483)

[**22.1.1.** **get\_LabelName function** 62](#_Toc180967484)

[**22.1.2.** **get\_InternalStatement function** 62](#_Toc180967485)

[**23.** **IEcoAST1CompoundStatement Interface** 63](#_Toc180967486)

[**23.1.** **IEcoAST1CompoundStatement IDL** 63](#_Toc180967487)

[**23.1.1.** **get\_BlockDeclarations function** 64](#_Toc180967488)

[**23.1.2.** **get\_BlockStatements function** 64](#_Toc180967489)

[**23.1.3.** **get\_Scope function** 64](#_Toc180967490)

[**24.** **IEcoAST1SwitchStatement Interface** 65](#_Toc180967491)

[**24.1.** **IEcoAST1SwitchStatement IDL** 65](#_Toc180967492)

[**24.1.1.** **get\_TestExpression function** 66](#_Toc180967493)

[**24.1.2.** **get\_Statement function** 66](#_Toc180967494)

[**25.** **IEcoAST1IfStatement Interface** 67](#_Toc180967495)

[**25.1.** **IEcoAST1IfStatement IDL** 67](#_Toc180967496)

[**25.1.1.** **get\_TestExpression function** 68](#_Toc180967497)

[**25.1.2.** **get\_TrueStatement function** 68](#_Toc180967498)

[**25.1.3.** **get\_FalseStatement function** 68](#_Toc180967499)

[**26.** **IEcoAST1ForStatement Interface** 69](#_Toc180967500)

[**26.1.** **IEcoAST1ForStatement IDL** 69](#_Toc180967501)

[**26.1.1.** **get\_InitExpression function** 70](#_Toc180967502)

[**26.1.2.** **get\_TestExpression function** 70](#_Toc180967503)

[**26.1.3.** **get\_IterExpression function** 70](#_Toc180967504)

[**26.1.4.** **get\_InternalStatement function** 70](#_Toc180967505)

[**27.** **IEcoAST1ReturnStatement Interface** 71](#_Toc180967506)

[**27.1.** **IEcoAST1ReturnStatement IDL** 71](#_Toc180967507)

[**27.1.1.** **get\_Expression function** 72](#_Toc180967508)

[**28.** **IEcoAST1GoToStatement Interface** 73](#_Toc180967509)

[**28.1.** **IEcoAST1GoToStatement IDL** 73](#_Toc180967510)

[**28.1.1.** **get\_LabelName function** 74](#_Toc180967511)

[**29.** **IEcoAST1ContinueStatement Interface** 75](#_Toc180967512)

[**29.1.** **IEcoAST1ContinueStatement IDL** 75](#_Toc180967513)

[**29.1.1.** **function** 76](#_Toc180967514)

[**30.** **IEcoAST1SingleStatement Interface** 77](#_Toc180967515)

[**30.1.** **IEcoAST1SingleStatement IDL** 77](#_Toc180967516)

[**30.1.1.** **function** 78](#_Toc180967517)

[**31.** **IEcoAST1Scope Interface** 79](#_Toc180967518)

[**31.1.** **IEcoAST1Scope IDL** 79](#_Toc180967519)

[**31.1.1.** **get\_ParentScope function** 81](#_Toc180967520)

[**31.1.2.** **get\_ChildScopes function** 81](#_Toc180967521)

[**31.1.3.** **add\_ChildScope function** 81](#_Toc180967522)

[**31.1.4.** **FindByIdentifier function** 81](#_Toc180967523)

[**31.1.5.** **FindByIdentifierOnCurrentLevel function** 81](#_Toc180967524)

[**31.1.6.** **RegistDeclaration function** 81](#_Toc180967525)

[**31.1.7.** **FindStructDeclarationByIdentifier function** 81](#_Toc180967526)

[**31.1.8.** **FindStructDeclarationByIdentifierOnCurrentLevel function** 81](#_Toc180967527)

[**31.1.9.** **FindEnumDeclarationByIdentifier function** 81](#_Toc180967528)

[**31.1.10.** **FindEnumDeclarationByIdentifierOnCurrentLevel function** 82](#_Toc180967529)

[**31.1.11.** **RegistUndeclaredType function** 82](#_Toc180967530)

[**31.1.12.** **RegistUndeclaredEnumType function** 82](#_Toc180967531)

[**31.1.13.** **DeclareType function** 82](#_Toc180967532)

[**31.1.14.** **DeclareEnumType function** 82](#_Toc180967533)

[**31.1.15.** **get\_Declarations function** 82](#_Toc180967534)

[**31.1.16.** **get\_StructOrUnionDeclarations function** 82](#_Toc180967535)

[**31.1.17.** **get\_EnumDeclarations function** 82](#_Toc180967536)

[**31.1.18.** **get\_UndeclaredTypes function** 82](#_Toc180967537)

[**31.1.19.** **get\_UndeclaredEnumTypes function** 83](#_Toc180967538)

[**32.** **IEcoAST1GlobalScope Interface** 84](#_Toc180967539)

[**32.1.** **IEcoAST1GlobalScope IDL** 84](#_Toc180967540)

[**32.1.1.** **FindFunctionDefenitionByIdentifier function** 85](#_Toc180967541)

[**32.1.2.** **RegistFunctionDefinition function** 85](#_Toc180967542)

[**32.1.3.** **get\_FunctionDefinitions function** 85](#_Toc180967543)

[**33.** **IEcoAST1Type Interface** 86](#_Toc180967544)

[**33.1.** **IEcoAST1Type IDL** 86](#_Toc180967545)

[**33.1.1.** **IsConst function** 87](#_Toc180967546)

[**33.1.2.** **get\_ClassTypeSpecifier function** 87](#_Toc180967547)

[**34.** **IEcoAST1TypedefType Interface** 88](#_Toc180967548)

[**34.1.** **IEcoAST1TypedefType IDL** 88](#_Toc180967549)

[**34.1.1.** **get\_TypeName function** 89](#_Toc180967550)

[**34.1.2.** **get\_Declaration function** 89](#_Toc180967551)

[**35.** **IEcoAST1StructOrUnionType Interface** 90](#_Toc180967552)

[**35.1.** **IEcoAST1StructOrUnionType IDL** 90](#_Toc180967553)

[**35.1.1.** **get\_Params function** 91](#_Toc180967554)

[**35.1.2.** **get\_TypeName function** 91](#_Toc180967555)

[**35.1.3.** **get\_Declaration function** 91](#_Toc180967556)

[**35.1.4.** **Declare function** 91](#_Toc180967557)

[**35.1.5.** **get\_StructOrUnionType function** 91](#_Toc180967558)

[**36.** **IEcoAST1PointerType Interface** 92](#_Toc180967559)

[**36.1.** **IEcoAST1PointerType IDL** 92](#_Toc180967560)

[**36.1.1.** **get\_IntegralType function** 93](#_Toc180967561)

[**37.** **IEcoAST1NumberType Interface** 94](#_Toc180967562)

[**37.1.** **IEcoAST1NumberType IDL** 94](#_Toc180967563)

[**37.1.1.** **IsSigned function** 95](#_Toc180967564)

[**37.1.2.** **get\_Size function** 95](#_Toc180967565)

[**37.1.3.** **get\_IntegralType function** 95](#_Toc180967566)

[**38.** **IEcoAST1FunctionType Interface** 96](#_Toc180967567)

[**38.1.** **IEcoAST1FunctionType IDL** 96](#_Toc180967568)

[**38.1.1.** **get\_ReturnType function** 97](#_Toc180967569)

[**38.1.2.** **get\_Params function** 97](#_Toc180967570)

[**38.1.3.** **get\_Scope function** 97](#_Toc180967571)

[**38.1.4.** **isVariadic function** 97](#_Toc180967572)

[**39.** **IEcoAST1EnumType Interface** 98](#_Toc180967573)

[**39.1.** **IEcoAST1EnumType IDL** 98](#_Toc180967574)

[**39.1.1.** **get\_Declaration function** 99](#_Toc180967575)

[**39.1.2.** **get\_TypeName function** 99](#_Toc180967576)

[**39.1.3.** **Declare function** 99](#_Toc180967577)

[**40.** **IEcoAST1ArrayType Interface** 100](#_Toc180967578)

[**40.1.** **IEcoAST1ArrayType IDL** 100](#_Toc180967579)

[**40.1.1.** **get\_IntegralType function** 101](#_Toc180967580)

[**40.1.2.** **get\_CountExpression function** 101](#_Toc180967581)

[**41.** **IEcoAST1SizeofType Interface** 102](#_Toc180967582)

[**41.1.** **IEcoAST1SizeofType IDL** 102](#_Toc180967583)

[**41.1.1.** **get\_InternalType function** 103](#_Toc180967584)

[**42.** **IEcoAST1Expression Interface** 104](#_Toc180967585)

[**42.1.** **IEcoAST1Expression IDL** 104](#_Toc180967586)

[**42.1.1.** **get\_Type function** 105](#_Toc180967587)

[**43.** **IEcoAST1BlockExpressions Interface** 106](#_Toc180967588)

[**43.1.** **IEcoAST1BlockExpressions IDL** 106](#_Toc180967589)

[**43.1.1.** **get\_Expression function** 107](#_Toc180967590)

[**43.1.2.** **get\_Count function** 107](#_Toc180967591)

[**44.** **IEcoAST1NullExpression Interface** 108](#_Toc180967592)

[**44.1.** **IEcoAST1NullExpression IDL** 108](#_Toc180967593)

[**44.1.1.** **The function function** 109](#_Toc180967594)

[**45.** **IEcoAST1UnaryExpression Interface** 110](#_Toc180967595)

[**45.1.** **IEcoAST1UnaryExpression IDL** 110](#_Toc180967596)

[**45.1.1.** **get\_Operator function** 111](#_Toc180967597)

[**45.1.2.** **get\_Operand function** 111](#_Toc180967598)

[**46.** **IEcoAST1SizeofExpression Interface** 112](#_Toc180967599)

[**46.1.** **IEcoAST1SizeofExpression IDL** 112](#_Toc180967600)

[**46.1.1.** **get\_InternalExpression function** 113](#_Toc180967601)

[**47.** **IEcoAST1ConditionalExpression Interface** 114](#_Toc180967602)

[**47.1.** **IEcoAST1ConditionalExpression IDL** 114](#_Toc180967603)

[**47.1.1.** **get\_TestOperand function** 115](#_Toc180967604)

[**47.1.2.** **get\_TrueOperand function** 115](#_Toc180967605)

[**47.1.3.** **get\_FalseOperand function** 115](#_Toc180967606)

[**48.** **IEcoAST1CastExpression Interface** 116](#_Toc180967607)

[**48.1.** **IEcoAST1CastExpression IDL** 116](#_Toc180967608)

[**48.1.1.** **get\_Operand function** 117](#_Toc180967609)

[**49.** **IEcoAST1BinaryExpression Interface** 118](#_Toc180967610)

[**49.1.** **IEcoAST1BinaryExpression IDL** 118](#_Toc180967611)

[**49.1.1.** **get\_Operator function** 119](#_Toc180967612)

[**49.1.2.** **get\_LeftOperand function** 119](#_Toc180967613)

[**49.1.3.** **get\_RightOperand function** 119](#_Toc180967614)

[**50.** **IEcoAST1FunctionCall Interface** 120](#_Toc180967615)

[**50.1.** **IEcoAST1FunctionCall IDL** 120](#_Toc180967616)

[**50.1.1.** **get\_Arguments function** 121](#_Toc180967617)

[**50.1.2.** **get\_FunctionExpression function** 121](#_Toc180967618)

[**51.** **IEcoAST1MemberSelection Interface** 122](#_Toc180967619)

[**51.1.** **IEcoAST1MemberSelection IDL** 122](#_Toc180967620)

[**51.1.1.** **get\_SourceExpression function** 123](#_Toc180967621)

[**51.1.2.** **get\_InternalExpression function** 123](#_Toc180967622)

[**51.1.3.** **isDirect function** 123](#_Toc180967623)

[**52.** **IEcoAST1VariableDefinition Interface** 124](#_Toc180967624)

[**52.1.** **IEcoAST1VariableDefinition IDL** 124](#_Toc180967625)

[**52.1.1.** **get\_IdentifierName function** 125](#_Toc180967626)

[**52.1.2.** **get\_Declaration function** 125](#_Toc180967627)

[**53.** **IEcoAST1Literal Interface** 126](#_Toc180967628)

[**53.1.** **IEcoAST1Literal IDL** 126](#_Toc180967629)

[**53.1.1.** **get\_Value function** 127](#_Toc180967630)

[**54.** **IEcoAST1CharacterLiteral Interface** 128](#_Toc180967631)

[**54.1.** **IEcoAST1CharacterLiteral IDL** 128](#_Toc180967632)

[**54.1.1.** **function** 129](#_Toc180967633)

[**55.** **IEcoAST1FloatLiteral Interface** 130](#_Toc180967634)

[**55.1.** **IEcoAST1FloatLiteral IDL** 130](#_Toc180967635)

[**55.1.1.** **function** 131](#_Toc180967636)

[**56.** **IEcoAST1IntegerLiteral Interface** 132](#_Toc180967637)

[**56.1.** **IEcoAST1IntegerLiteral IDL** 132](#_Toc180967638)

[**56.1.1.** **function** 133](#_Toc180967639)

[**57.** **IEcoAST1StringLiteral Interface** 134](#_Toc180967640)

[**57.1.** **IEcoAST1StringLiteral IDL** 134](#_Toc180967641)

[**57.1.1.** **get\_Value function** 135](#_Toc180967642)

[**58.** **IEcoAST1FunctionDefinition Interface** 136](#_Toc180967643)

[**58.1.** **IEcoAST1FunctionDefinition IDL** 136](#_Toc180967644)

[**58.1.1.** **get\_Declaration function** 137](#_Toc180967645)

[**58.1.2.** **get\_Body function** 137](#_Toc180967646)

[**59.** **IEcoAST1TranslationUnitObject Interface** 138](#_Toc180967647)

[**59.1.** **IEcoAST1TranslationUnitObject IDL** 138](#_Toc180967648)

[**59.1.1.** **function** 139](#_Toc180967649)

[**60.** **IEcoAST1BlockTranslationUnitObjects Interface** 140](#_Toc180967650)

[**60.1.** **IEcoAST1BlockTranslationUnitObjects IDL** 140](#_Toc180967651)

[**60.1.1.** **get\_Object function** 141](#_Toc180967652)

[**60.1.2.** **get\_Count function** 141](#_Toc180967653)

[**61.** **Коды ошибок** 142](#_Toc180967654)

[**Приложение А: Обучающие программы** 143](#_Toc180967655)

1. **Overview**

This document describes the requirements for the implementation of the Eco.AST1 component.

* 1. **Introduction**

Description.

* 1. **Note**
* Keywords
  1. **Links**

This paragraph contains links to information to help you understand this document:

[] – name of the link

Available by: http://address

1. **Eco.** **AST1 Component**

The Eco.AST1component is

The component has the following description:

1. **IEcoAST1 Interface**
   1. **IEcoAST1 IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1Type.idl"  import "IEcoAST1NumberType.idl"  import "IEcoAST1StructOrUnionType.idl"  import "IEcoAST1EnumType.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1BinaryExpression.idl"  import "IEcoAST1MemberSelection.idl"  import "IEcoAST1UnaryExpression.idl"  import "IEcoAST1IntegerLiteral.idl"  import "IEcoAST1CharacterLiteral.idl"  import "IEcoAST1FloatLiteral.idl"  import "IEcoAST1StringLiteral.idl"  import "IEcoAST1VariableDefinition.idl"  import "IEcoAST1CastExpression.idl"  import "IEcoAST1ConditionalExpression.idl"  import "IEcoAST1SizeofType.idl"  import "IEcoAST1SizeofExpression.idl"  import "IEcoAST1NullExpression.idl"  import "IEcoAST1BlockExpressions.idl"  import "IEcoAST1BreakStatement.idl"  import "IEcoAST1ContinueStatement.idl"  import "IEcoAST1ReturnStatement.idl"  import "IEcoAST1GoToStatement.idl"  import "IEcoAST1IfStatement.idl"  import "IEcoAST1ForStatement.idl"  import "IEcoAST1WhileStatement.idl"  import "IEcoAST1DoWhileStatement.idl"  import "IEcoAST1VariableDeclaration.idl"  import "IEcoAST1SwitchStatement.idl"  import "IEcoAST1DefaultStatement.idl"  import "IEcoAST1CaseStatement.idl"  import "IEcoAST1LabelStatement.idl"  import "IEcoAST1BlockStatements.idl"  import "IEcoAST1BlockDeclarations.idl"  import "IEcoAST1BlockTranslationUnitObjects.idl"  import "IEcoAST1ArrayType.idl"  import "IEcoAST1PointerType.idl"  import "IEcoAST1StructOrUnionDeclaration.idl"  import "IEcoAST1EnumDeclaration.idl"  import "IEcoAST1CompoundStatement.idl"  import "IEcoAST1FunctionType.idl"  import "IEcoAST1BlockTypes.idl"  import "IEcoAST1FunctionDeclaration.idl"  import "IEcoAST1FunctionDefinition.idl"  import "IEcoAST1FunctionCall.idl"  import "IEcoAST1TypedefDeclaration.idl"  import "IEcoAST1TypedefType.idl"  import "IEcoAST1GlobalScope.idl"  import "IEcoAST1TranslationUnit.idl" | | | |
| [  object,  uguid(E248639F-2296-469E-B1D5-2D75ABE95298),  ] | | | |
| interface IEcoAST1 : IEcoUnknown { | | | |
|  | | | |
| int16\_t | | ***new\_NumberType*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] int16\_t prim\_type,  [in] uint32\_t size,  [in] bool\_t isConst,  [in] bool\_t isSigned,  [in] uint16\_t classSpecifier,  [out] IEcoAST1NumberType\*\* ppIType); |
|  | | | |
| int16\_t | | ***new\_StructOrUnionType*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* name,  [in] bool\_t isConst,  [in] IEcoAST1StructOrUnionDeclaration\* pIStructDec,  [in] uint16\_t classSpecifier,  [in] bool\_t isDeclared,  [in] uint16\_t structOrUnionType,  [out] IEcoAST1StructOrUnionType\*\* ppIType); |
|  | | | |
| int16\_t | | ***new\_EnumType*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* name,  [in] bool\_t isConst,  [in] IEcoAST1EnumDeclaration\* pIStructDec,  [in] uint16\_t classSpecifier,  [in] bool\_t isDeclared,  [out] IEcoAST1EnumType\*\* ppIType); |
|  | | | |
| int16\_t | | ***new\_ArrayType*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Type\* type,  [in] IEcoAST1Expression\* pICountExpression,  [out] IEcoAST1ArrayType\*\* ppIType); |
|  | |  |  |
| int16\_t | | ***new\_PointerType*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Type\* type,  [in] bool\_t isConst,  [out] IEcoAST1PointerType\*\* ppIType); |
|  | | | |
| int16\_t | | ***new\_IntegerLiteral*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* literal,  [out] IEcoAST1IntegerLiteral\*\* ppIIntegerLiteral); |
|  | | | |
| int16\_t | | ***new\_CharacterLiteral*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* literal,  [out] IEcoAST1CharacterLiteral\*\* ppILiteral); |
|  | | | |
| int16\_t | | ***new\_FloatLiteral*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* literal,  [out] IEcoAST1FloatLiteral\*\* ppIFloatLiteral); |
|  | | | |
| int16\_t | | ***new\_StringLiteral*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* literal,  [out] IEcoAST1StringLiteral\*\* ppIStringLiteral); |
|  | |  |  |
| int16\_t | | ***new\_VariableDefinition*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1ExpressionableDeclaration\* pIDec,  [out] IEcoAST1VariableDefinition\*\* ppIVariableDefinition); |
|  | |  |  |
| int16\_t | | ***new\_BinaryExpression*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] uint16\_t BinaryOperator,  [in] IEcoAST1Expression\* leftOperand,  [in] IEcoAST1Expression\* rightOperand,  [out] IEcoAST1BinaryExpression\*\* ppIBinaryExpression); |
|  | |  |  |
| int16\_t | | ***new\_MemberSelection*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* pISourceExpression,  [in] IEcoAST1VariableDefinition\* pIInternalExpression,  [in] bool\_t isDirect,  [out] IEcoAST1MemberSelection\*\* ppIMemberSelection); |
|  | |  |  |
| int16\_t | | ***new\_UnaryExpression*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] uint16\_t unaryOperator,  [in] IEcoAST1Expression\* Operand,  [out] IEcoAST1UnaryExpression\*\* ppIUnaryExpression); |
|  | |  |  |
| int16\_t | | ***new\_CastExpression*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Type\* type,  [in] IEcoAST1Expression\* operand,  [out] IEcoAST1CastExpression\*\* ppICastExpression); |
|  | |  |  |
| int16\_t | | ***new\_ConditionalExpression*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* testOperand,  [in] IEcoAST1Expression\* trueOperand,  [in] IEcoAST1Expression\* falseOperand,  [out] IEcoAST1ConditionalExpression\*\* ppIConditionalExpression); |
|  | |  |  |
| int16\_t | | ***new\_SizeofType*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Type\* type,  [out] IEcoAST1SizeofType\*\* ppISizeofType); |
|  | |  |  |
| int16\_t | | ***new\_SizeofExpression*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* pIExpression,  [out] IEcoAST1SizeofExpression\*\* ppISizeofExpression); |
|  | |  |  |
| int16\_t | | ***new\_NullExpression*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [out] IEcoAST1NullExpression\*\* ppINullExpression) |
|  | |  |  |
| int16\_t | | ***new\_BlockExpressions*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoList1\* list,  [out] IEcoAST1BlockExpressions\*\* ppIBlockExpressions); |
|  | |  |  |
| int16\_t | | ***new\_BreakStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [out] IEcoAST1BreakStatement\*\* ppIBreakStatement); |
|  | |  |  |
| int16\_t | | ***new\_ContinueStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [out] IEcoAST1ContinueStatement\*\* ppIContinueStatement); |
|  | |  |  |
| int16\_t | | ***new\_ReturnStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* expression,  [out] IEcoAST1ReturnStatement\*\* ppIReturnStatement); |
|  | |  |  |
| int16\_t | | ***new\_GoToStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* label\_Name,  [out] IEcoAST1GoToStatement\*\* ppIGoToStatement); |
|  | |  |  |
| int16\_t | | ***new\_IfStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* expression,  IEcoAST1Statement\* true\_statement,  IEcoAST1Statement\* false\_statement,  [out] IEcoAST1IfStatement\*\* ppIIfStatement); |
|  | |  |  |
| int16\_t | | ***new\_ForStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* init,  IEcoAST1Expression\* test,  IEcoAST1Expression\* iter,  IEcoAST1Statement\* statement,  [out] IEcoAST1ForStatement\*\* ppIForStatement); |
|  | |  |  |
| int16\_t | | ***new\_WhileStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* test,  IEcoAST1Statement\* statement,  [out] IEcoAST1WhileStatement\*\* ppIWhileStatement); |
|  | |  |  |
| int16\_t | | ***new\_DoWhileStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* test,  IEcoAST1Statement\* statement,  [out] IEcoAST1DoWhileStatement\*\* ppIDoWhileStatement); |
|  | |  |  |
| int16\_t | | ***new\_SwitchStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* expression,  IEcoAST1Statement\* statement,  [out] IEcoAST1SwitchStatement\*\* ppISwitchStatement); |
|  | |  |  |
| int16\_t | | ***new\_DefaultStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Statement\* pIInternalStatement,  [out] IEcoAST1DefaultStatement\*\* ppIDefaultStatement); |
|  | |  |  |
| int16\_t | | ***new\_CaseStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* pIExpression,  [in] IEcoAST1Statement\* pIStatement,  [out] IEcoAST1CaseStatement\*\* ppICaseStatement); |
|  | |  |  |
| int16\_t | | ***new\_LabelStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* label\_name,  [in] IEcoAST1Statement\* pIStatement,  [out] IEcoAST1LabelStatement\*\* ppILabelStatement); |
|  | |  |  |
| int16\_t | | ***new\_BlockStatements*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoList1\* pIListStatements,  [out] IEcoAST1BlockStatements\*\* ppIBlockStatements); |
|  | |  |  |
| int16\_t | | ***new\_BlockDeclarations*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoList1\* pIListDeclarations,  [out] IEcoAST1BlockDeclarations\*\* ppIBlockDeclarations); |
|  | |  |  |
| int16\_t | | ***new\_BlockTranslationUnitObjects*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoList1\* pIListObjects,  [out] IEcoAST1BlockTranslationUnitObjects\*\* ppIBlockTranslationUnitObjects); |
|  | |  |  |
| int16\_t | | ***new\_VariableDeclaration*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Type\* pIType,  [in] char\_t\* name,  [in] IEcoAST1Expression\* pIInitExp,  [out] IEcoAST1VariableDeclaration\*\* ppIDeclaration); |
|  | |  |  |
| int16\_t | | ***new\_StructOrUnionDeclaration*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* name,  [in] IEcoAST1BlockDeclarations\* pIParams,  [in] IEcoAST1BlockExpressions\* pISizes,  [in] uint16\_t structOrUnionType,  [in] IEcoAST1Scope\* pIScope,  [out] IEcoAST1StructOrUnionDeclaration\*\* ppIDeclaration); |
|  | |  |  |
| int16\_t | | ***new\_EnumDeclaration*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* name,  [in] IEcoAST1BlockDeclarations\* pIDeclarations,  [in] IEcoAST1Scope\* pIScope,  [out] IEcoAST1EnumDeclaration\*\* ppIDeclaration); |
|  | |  |  |
| int16\_t | | ***new\_CompoundStatement*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1BlockDeclarations\* declarations,  IEcoAST1BlockStatements\* statements,  [in] IEcoAST1Scope\* pIScope,  [out] IEcoAST1CompoundStatement\*\* ppICompoundStatement); |
|  | |  |  |
| int16\_t | | ***new\_FunctionType*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Type\* pIReturnType,  [in] IEcoAST1BlockDeclarations\* pIParams,  [in] IEcoAST1Scope\* pIScope,  [in] bool\_t isVariadic,  [out] IEcoAST1FunctionType\*\* ppIFunctionType); |
|  | |  |  |
| int16\_t | | ***new\_BlockTypes*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoList1\* pIListTypes,  [in] bool\_t isVariadic,  [out] IEcoAST1BlockTypes\*\* ppIBlockTypes); |
|  | |  |  |
| int16\_t | | ***new\_FunctionDeclaration*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1FunctionType\* pIType,  [in] char\_t\* name,  [out] IEcoAST1FunctionDeclaration\*\* ppIFunctionDeclaration); |
|  | |  |  |
| int16\_t | | ***new\_FunctionDefinition*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1FunctionDeclaration\* pIDeclaration,  [in] IEcoAST1CompoundStatement\* pIBody,  [out] IEcoAST1FunctionDefinition\*\* ppIFunctionDefinition); |
|  | |  |  |
| int16\_t | | ***new\_FunctionCall*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Expression\* pIFunctionExpression,  IEcoAST1BlockExpressions\* pArguments,  [out] IEcoAST1FunctionCall\*\* ppIFunctionCall); |
|  | |  |  |
| int16\_t | | ***new\_TypedefDeclaration*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] char\_t\* name,  [in] IEcoAST1Type\* pISourceType,  [out] IEcoAST1TypedefDeclaration\*\* ppIDeclaration); |
|  | |  |  |
| int16\_t | | ***new\_TypedefType*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] bool\_t isConst,  [in] IEcoAST1TypedefDeclaration\* pIDeclaration,  [in] uint16\_t classSpecifier,  [out] IEcoAST1TypedefType\*\* ppIType); |
|  | |  |  |
| int16\_t | | ***new\_GlobalScope*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [out] IEcoAST1GlobalScope\*\* ppIGlobalScope); |
|  | |  |  |
| int16\_t | | ***new\_Scope*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1Scope\* pIParentScope,  [out] IEcoAST1Scope\*\* ppIScope); |
|  | |  |  |
| int16\_t | | ***new\_TranslationUnit*** | ([in] IEcoLexicalAnalyzer1Token\* token,  [in] IEcoAST1GlobalScope\* pIScope,  IEcoAST1BlockTranslationUnitObjects\* pIObjects,  [out] IEcoAST1TranslationUnit\*\* ppITranslationUnit); |
|  | |  |  |
| IEcoAST1SyntaxObject\* | | ***copy\_Object*** | ([in] IEcoAST1SyntaxObject\* pIObject); |
|  | |  |  |
| IEcoAST1TranslationUnit\* | | ***get\_TranslationUnit)*** | (); |
|  | |  |  |
| } | |  |  |

* + 1. **new\_NumberType function**

The function

* + 1. **new\_StructOrUnionType function**

The function

* + 1. **new\_EnumType function**

The function

* + 1. **new\_ArrayType function**

The function

* + 1. **new\_PointerType function**

The function

* + 1. **new\_IntegerLiteral function**

The function

* + 1. **new\_CharacterLiteral function**

The function

* + 1. **new\_FloatLiteral function**

The function

* + 1. **new\_StringLiteral function**

The function

* + 1. **new\_VariableDefinition function**

The function

* + 1. **new\_BinaryExpression function**

The function

* + 1. **new\_MemberSelection function**

The function

* + 1. **new\_UnaryExpression function**

The function

* + 1. **new\_CastExpression function**

The function

* + 1. **new\_ConditionalExpression function**

The function

* + 1. **new\_SizeofType function**

The function

* + 1. **new\_SizeofExpression function**

The function

* + 1. **new\_NullExpression function**

The function

* + 1. **new\_BlockExpressions function**

The function

* + 1. **new\_BreakStatement function**

The function

* + 1. **new\_ContinueStatement function**

The function

* + 1. **new\_ReturnStatement function**

The function

* + 1. **new\_GoToStatement function**

The function

* + 1. **new\_IfStatement function**

The function

* + 1. **new\_ForStatement function**

The function

* + 1. **new\_WhileStatement function**

The function

* + 1. **new\_DoWhileStatement function**

The function

* + 1. **new\_SwitchStatement function**

The function

* + 1. **new\_DefaultStatement function**

The function

* + 1. **new\_CaseStatement function**

The function

* + 1. **new\_LabelStatement function**

The function

* + 1. **new\_BlockStatements function**

The function

* + 1. **new\_BlockDeclarations function**

The function

* + 1. **new\_BlockTranslationUnitObjects function**

The function

* + 1. **new\_VariableDeclaration function**

The function

* + 1. **new\_StructOrUnionDeclaration function**

The function

* + 1. **new\_EnumDeclaration function**

The function

* + 1. **new\_CompoundStatement function**

The function

* + 1. **new\_FunctionType function**

The function

* + 1. **new\_BlockTypes function**

The function

* + 1. **new\_FunctionDeclaration function**

The function

* + 1. **new\_FunctionDefinition function**

The function

* + 1. **new\_FunctionCall function**

The function

* + 1. **new\_TypedefDeclaration function**

The function

* + 1. **new\_TypedefType function**

The function

* + 1. **new\_GlobalScope function**

The function

* + 1. **new\_Scope function**

The function

* + 1. **new\_TranslationUnit function**

The function

* + 1. **copy\_Object function**

The function

* + 1. **get\_TranslationUnit) function**

The function

1. **IEcoAST1SyntaxObject Interface**
   1. **IEcoAST1SyntaxObject IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(F0BC0E8A-A598-42E1-9676-801CA6B18F1D),  ] | | | |
| interface IEcoAST1SyntaxObject : IEcoUnknown { | | | |
|  | | | |
| const UGUID\* | | ***get\_Id*** | ([in] void); |
|  | | | |
| IEcoLexicalAnalyzer1Token\* | | ***get\_Token*** | ([in] void); |
|  | | | |
| char\_t\* | | ***get\_NameObject*** | ([in] void); |
|  | |  |  |
| IEcoAST1SyntaxObject\* | | ***CopyToAST*** | ([in] IEcoAST1\* pIAST); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Id function**

The function

* + 1. **get\_Token function**

The function

* + 1. **get\_NameObject function**

The function

* + 1. **CopyToAST function**

The function

1. **IEcoAST1BlockDeclarations Interface**
   1. **IEcoAST1BlockDeclarations IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoSystem1.idl"  import "IEcoAST1Declaration.idl" | | | |
| [  object,  uguid(F0BC0E8A-A598-42E1-9676-801CA6B1FF98),  ] | | | |
| interface IEcoAST1BlockDeclarations : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Declaration\* | | ***get\_Declaration*** | ([in] uint32\_t\* index); |
|  | | | |
| uint32\_t | | ***get\_Count*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Declaration function**

The function

* + 1. **get\_Count function**

The function

1. **IEcoAST1BlockTypes Interface**
   1. **IEcoAST1BlockTypes IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoSystem1.idl"  import "IEcoAST1Type.idl" | | | |
| [  object,  uguid(F0BC0E8A-A598-42E1-9676-801CA6B1F808),  ] | | | |
| interface IEcoAST1BlockTypes: IEcoAST1SyntaxObject { | | | |
|  | | | |
| IEcoAST1Type\* | | ***get\_Type*** | ([in] uint32\_t\* index); |
|  | | | |
| uint32\_t\* | | ***get\_Count*** | ([in] void); |
|  | |  |  |
| bool\_t | | ***IsVariadic*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Type function**

The function

* + 1. **get\_Count function**

The function

* + 1. **IsVariadic function**

The function

1. **IEcoAST1Declaration Interface**
   1. **IEcoAST1Declaration IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CEC4E83),  ] | | | |
| interface **IEcoAST1Declaration** : IEcoAST1SyntaxObject { | | | |
|  | | | |
| char\_t\* | | ***get\_IdentifierName*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_IdentifierName function**

The function

1. **IEcoAST1VariableDeclaration Interface**
   1. **IEcoAST1VariableDeclaration IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1Expression.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CEC475D),  ] | | | |
| interface IEcoAST1VariableDeclaration : IEcoAST1ExpressionableDeclaration { | | | |
|  | | | |
| IEcoAST1Expression \* | | ***get\_Initializator*** | ([in] void); |
|  | | | |
| } | |  |  |

* + 1. **get\_Initializator function**

The function

1. **IEcoAST1ExpressionableDeclaration Interface**
   1. **IEcoAST1ExpressionableDeclaration IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1Type.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CEC4E80),  ] | | | |
| interface IEcoAST1ExpressionableDeclaration: IEcoAST1VariableDeclaration { | | | |
|  | | | |
| IEcoAST1Type\* | | ***get\_Type*** | ([in] void); |
|  | | | |
| } | |  |  |

* + 1. **get\_Type function**

The function

1. **IEcoAST1FunctionDeclaration Interface**
   1. **IEcoAST1FunctionDeclaration IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1FunctionType.idl"  import "IEcoAST1BlockDeclarations.idl"  import "IEcoAST1Scope.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CEC656D),  ] | | | |
| interface IEcoAST1FunctionDeclaration : IEcoAST1ExpressionableDeclaration { | | | |
|  | | | |
| IEcoAST1BlockDeclarations\* | | ***get\_Type*** | ([in] void); |
|  | |  |  |
| IEcoAST1Type\* | | ***get\_ReturnType*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Params function**

The function

* + 1. **get\_ReturnType function**

The function

1. **IEcoAST1EnumDeclaration Interface**
   1. **IEcoAST1EnumDeclaration IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1VariableDefinition.idl"  import "IEcoAST1BlockDeclarations.idl"  import "IEcoAST1BlockExpressions.idl"  import "IEcoAST1Scope.idl"  import "IEcoAST1Expression.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CEC4458),  ] | | | |
| interface IEcoAST1EnumDeclaration : IEcoAST1Declaration{ | | | |
|  | | | |
| IEcoAST1BlockDeclarations\* | | ***get\_Declarations*** | ([in] void); |
|  | |  |  |
| IEcoAST1Scope\* | | ***get\_Scope*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Declarations function**

The function

* + 1. **get\_Scope function**

The function

1. **IEcoAST1TypedefDeclaration Interface**
   1. **IEcoAST1TypedefDeclaration IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1Type.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CEC0AAA),  ] | | | |
| interface IEcoAST1TypedefDeclaration : IEcoAST1Declaration{ | | | |
|  | | | |
| IEcoAST1Type\* | | ***get\_SourceType*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_SourceType function**

The function

1. **IEcoAST1StructOrUnionDeclaration Interface**
   1. **IEcoAST1StructOrUnionDeclaration IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1VariableDefinition.idl"  import "IEcoAST1BlockDeclarations.idl"  import "IEcoAST1BlockExpressions.idl"  import "IEcoAST1Scope.idl"  import "IEcoAST1Expression.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CEC4610),  ] | | | |
| interface IEcoAST1StructOrUnionDeclaration : IEcoAST1Declaration{ | | | |
|  | | | |
| IEcoAST1BlockDeclarations\* | | ***get\_Params*** | ([in] void); |
|  | |  |  |
| uint16\_t | | ***get\_StructOrUnionType*** | ([in] void); |
|  | |  |  |
| IEcoAST1Scope\* | | ***get\_Scope*** | ([in] void); |
|  | |  |  |
| IEcoAST1BlockExpressions\* | | ***get\_Sizes*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Params function**

The function

* + 1. **get\_StructOrUnionType function**

The function

* + 1. **get\_Scope function**

The function

* + 1. **get\_Sizes function**

The function

1. **IEcoAST1TranslationUnit Interface**
   1. **IEcoAST1TranslationUnit IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1GlobalScope.idl"  import "IEcoAST1BlockTranslationUnitObjects.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CEC6ABA),  ] | | | |
| interface IEcoAST1TranslationUnit : IEcoAST1Declaration{ | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoAST1Statement Interface**
   1. **IEcoAST1Statement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(F0BC0E8A-A598-42E1-9676-801CA6B18A25),  ] | | | |
| interface IEcoAST1Statement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **The function function**

The function

1. **IEcoAST1BlockStatements Interface**
   1. **IEcoAST1BlockStatements IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoSystem1.idl"  import "IEcoAST1Statement.idl" | | | |
| [  object,  uguid(F0BC0E8A-A598-42E1-9676-801CA6B18301),  ] | | | |
| interface IEcoAST1BlockStatements: IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Statement\* | | ***get\_Statement*** | (uint32\_t index); |
|  | |  |  |
| uint32\_t | | ***get\_Count*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Statement function**

The function

* + 1. **get\_Count function**

The function

1. **IEcoAST1DoWhileStatement Interface**
   1. **IEcoAST1DoWhileStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4054),  ] | | | |
| interface IEcoAST1DoWhileStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_TestExpression*** | ([in] void); |
|  | |  |  |
| IEcoAST1Statement\* | | ***get\_InternalStatement*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_TestExpression function**

The function

* + 1. **get\_InternalStatement function**

The function

1. **IEcoAST1WhileStatement Interface**
   1. **IEcoAST1WhileStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C456F),  ] | | | |
| interface IEcoAST1WhileStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_TestExpression*** | ([in] void); |
|  | |  |  |
| IEcoAST1Statement\* | | ***get\_InternalStatement*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_TestExpression function**

The function

* + 1. **get\_InternalStatement function**

The function

1. **IEcoAST1DefaultStatement Interface**
   1. **IEcoAST1DefaultStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl "  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl "  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4FDB),  ] | | | |
| interface IEcoAST1DefaultStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Statement\* | | ***get\_InternalStatement*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_InternalStatement function**

The function

1. **IEcoAST1BreakStatement Interface**
   1. **IEcoAST1BreakStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4332),  ] | | | |
| interface IEcoAST1BreakStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoAST1CaseStatement Interface**
   1. **IEcoAST1CaseStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4EAA),  ] | | | |
| interface IEcoAST1CaseStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_Expression*** | ([in] void); |
|  | |  |  |
| IEcoAST1Statement\* | | ***get\_Statement*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Expression function**

The function

* + 1. **get\_Statement function**

The function

1. **IEcoAST1LabelStatement Interface**
   1. **IEcoAST1LabelStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4DB4),  ] | | | |
| interface IEcoAST1LabelStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| char\_t\* | | ***get\_LabelName*** | ([in] void); |
|  | |  |  |
| IEcoAST1Statement\* | | ***get\_InternalStatement*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_LabelName function**

The function

* + 1. **get\_InternalStatement function**

The function

1. **IEcoAST1CompoundStatement Interface**
   1. **IEcoAST1CompoundStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1BlockDeclarations.idl"  import "IEcoAST1BlockStatements.idl"  import "IEcoAST1Scope.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4443),  ] | | | |
| interface IEcoAST1CompoundStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1BlockDeclarations\* | | ***get\_BlockDeclarations*** | ([in] void); |
|  | |  |  |
| IEcoAST1BlockStatements\* | | ***get\_BlockStatements*** | ([in] void); |
|  | |  |  |
| IEcoAST1Scope\* | | ***get\_Scope*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_BlockDeclarations function**

The function

* + 1. **get\_BlockStatements function**

The function

* + 1. **get\_Scope function**

The function

1. **IEcoAST1SwitchStatement Interface**
   1. **IEcoAST1SwitchStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4106),  ] | | | |
| interface IEcoAST1SwitchStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_TestExpression*** | ([in] void); |
|  | |  |  |
| IEcoAST1Statement\* | | ***get\_Statement*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_TestExpression function**

The function

* + 1. **get\_Statement function**

The function

1. **IEcoAST1IfStatement Interface**
   1. **IEcoAST1IfStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4F4D),  ] | | | |
| interface IEcoAST1IfStatement : IEcoAST1SyntaxObject { | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_TestExpression*** | ([in] void); |
|  | |  |  |
| IEcoAST1Statement\* | | ***get\_TrueStatement*** | ([in] void); |
|  | |  |  |
| IEcoAST1Statement\* | | ***get\_FalseStatement*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_TestExpression function**

The function

* + 1. **get\_TrueStatement function**

The function

* + 1. **get\_FalseStatement function**

The function

1. **IEcoAST1ForStatement Interface**
   1. **IEcoAST1ForStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4FF5),  ] | | | |
| interface IEcoAST1ForStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_InitExpression*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression\* | | ***get\_TestExpression*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression\* | | ***get\_IterExpression*** | ([in] void); |
|  | |  |  |
| IEcoAST1Statement\* | | ***get\_InternalStatement*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_InitExpression function**

The function

* + 1. **get\_TestExpression function**

The function

* + 1. **get\_IterExpression function**

The function

* + 1. **get\_InternalStatement function**

The function

1. **IEcoAST1ReturnStatement Interface**
   1. **IEcoAST1ReturnStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4AAA),  ] | | | |
| interface IEcoAST1ReturnStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_Expression*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Expression function**

The function

1. **IEcoAST1GoToStatement Interface**
   1. **IEcoAST1GoToStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4DA3),  ] | | | |
| interface IEcoAST1GoToStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| char\_t\* | | ***get\_LabelName*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_LabelName function**

The function

1. **IEcoAST1ContinueStatement Interface**
   1. **IEcoAST1ContinueStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4462),  ] | | | |
| interface IEcoAST1ContinueStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoAST1SingleStatement Interface**
   1. **IEcoAST1SingleStatement IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Type.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(937549A5-DFF7-41AE-AD21-166A74B0CC5A),  ] | | | |
| interface IEcoAST1SingleStatement : IEcoAST1SyntaxObject{ | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoAST1Scope Interface**
   1. **IEcoAST1Scope IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1Declaration.idl"  import "IEcoAST1StructOrUnionDeclaration.idl"  import "IEcoAST1StructOrUnionType.idl"  import "IEcoAST1EnumDeclaration.idl"  import "IEcoAST1EnumType.idl" | | | |
| [  object,  uguid(EFE01C87-3641-420F-8C24-DF2668126111),  ] | | | |
| interface IEcoAST1Scope : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Scope\* | | ***get\_ParentScope*** | ([in] void); |
|  | |  |  |
| IEcoList1\* | | ***get\_ChildScopes*** | ([in] void); |
|  | |  |  |
| int16\_t | | ***add\_ChildScope*** | ([in] IEcoAST1ScopePtr\_t pIChildScope); |
|  | |  |  |
| IEcoAST1Declaration\* | | ***FindByIdentifier*** | ([in] char\_t\* name); |
|  | |  |  |
| IEcoAST1Declaration\* | | ***FindByIdentifierOnCurrentLevel*** | ([in] char\_t\* name); |
|  | |  |  |
| int16\_t | | ***RegistDeclaration*** | ([in] IEcoAST1Declaration\* pIDeclaration); |
|  | |  |  |
| IEcoAST1StructOrUnionDeclaration\* | | ***FindStructDeclarationByIdentifier*** | ([in] char\_t\* name); |
|  | |  |  |
| IEcoAST1StructOrUnionDeclaration\* | | ***FindStructDeclarationByIdentifierOnCurrentLevel*** | ([in] char\_t\* name); |
|  | |  |  |
| IEcoAST1EnumDeclaration\* | | ***FindEnumDeclarationByIdentifier*** | ([in] char\_t\* name); |
|  | |  |  |
| IEcoAST1EnumDeclaration\* | | ***FindEnumDeclarationByIdentifierOnCurrentLevel*** | ([in] char\_t\* name); |
|  | |  |  |
| int16\_t | | ***RegistUndeclaredType*** | ([in] IEcoAST1StructOrUnionType\* pIType); |
|  | |  |  |
| int16\_t | | ***RegistUndeclaredEnumType*** | ([in] IEcoAST1EnumType\* pIType); |
|  | |  |  |
| int16\_t | | ***DeclareType*** | ( [in] IEcoAST1StructOrUnionDeclaration\* pIDeclaration); |
|  | |  |  |
| int16\_t | | ***DeclareEnumType*** | ([in] IEcoAST1EnumDeclaration\* pIDeclaration); |
|  | |  |  |
| IEcoList1\* | | ***get\_Declarations*** | ([in] void); |
|  | |  |  |
| IEcoList1\* | | ***get\_StructOrUnionDeclarations*** | ([in] void); |
|  | |  |  |
| IEcoList1\* | | ***get\_EnumDeclarations*** | ([in] void); |
|  | |  |  |
| IEcoList1\* | | ***get\_UndeclaredTypes*** | ([in] void); |
|  | |  |  |
| IEcoList1\* | | ***get\_UndeclaredEnumTypes*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_ParentScope function**

The function

* + 1. **get\_ChildScopes function**

The function

* + 1. **add\_ChildScope function**

The function

* + 1. **FindByIdentifier function**

The function

* + 1. **FindByIdentifierOnCurrentLevel function**

The function

* + 1. **RegistDeclaration function**

The function

* + 1. **FindStructDeclarationByIdentifier function**

The function

* + 1. **FindStructDeclarationByIdentifierOnCurrentLevel function**

The function

* + 1. **FindEnumDeclarationByIdentifier function**

The function

* + 1. **FindEnumDeclarationByIdentifierOnCurrentLevel function**

The function

* + 1. **RegistUndeclaredType function**

The function

* + 1. **RegistUndeclaredEnumType function**

The function

* + 1. **DeclareType function**

The function

* + 1. **DeclareEnumType function**

The function

* + 1. **get\_Declarations function**

The function

* + 1. **get\_StructOrUnionDeclarations function**

The function

* + 1. **get\_EnumDeclarations function**

The function

* + 1. **get\_UndeclaredTypes function**

The function

* + 1. **get\_UndeclaredEnumTypes function**

The function

1. **IEcoAST1GlobalScope Interface**
   1. **IEcoAST1GlobalScope IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Scope.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1FunctionDefinition.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4967),  ] | | | |
| interface IEcoAST1GlobalScope : IEcoAST1Scope{ | | | |
|  | | | |
| IEcoAST1FunctionDefinition\* | | ***FindFunctionDefenitionByIdentifier*** | ([in] char\_t\* name); |
|  | |  |  |
| int16\_t | | ***RegistFunctionDefinition*** | ([in] IEcoAST1FunctionDefinition\* pIDefinition); |
|  | |  |  |
| IEcoList1\* | | ***get\_FunctionDefinitions*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **FindFunctionDefenitionByIdentifier function**

The function

* + 1. **RegistFunctionDefinition function**

The function

* + 1. **get\_FunctionDefinitions function**

The function

1. **IEcoAST1Type Interface**
   1. **IEcoAST1Type IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(EFE01C87-3641-420F-8C24-DF2668126349),  ] | | | |
| interface IEcoAST1Type : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| bool\_t | | ***IsConst*** | ([in] void); |
|  | |  |  |
| uint16\_t | | ***get\_ClassTypeSpecifier*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **IsConst function**

The function

* + 1. **get\_ClassTypeSpecifier function**

The function

1. **IEcoAST1TypedefType Interface**
   1. **IEcoAST1TypedefType IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Type.idl"  import "IEcoAST1TypedefDeclaration.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(950EC30B-18C9-4CA4-BB6D-63277A296555),  ] | | | |
| interface IEcoAST1TypedefType : IEcoAST1Type{ | | | |
|  | | | |
| char\_t\* | | ***get\_TypeName*** | ([in] void); |
|  | |  |  |
| IEcoAST1TypedefDeclaration\* | | ***get\_Declaration*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_TypeName function**

The function

* + 1. **get\_Declaration function**

The function

1. **IEcoAST1StructOrUnionType Interface**
   1. **IEcoAST1StructOrUnionType IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Type.idl"  import "IEcoAST1StructOrUnionDeclaration.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(950EC30B-18C9-4CA4-BB6D-63277A29CD76),  ] | | | |
| interface IEcoAST1StructOrUnionType : IEcoAST1Type{ | | | |
|  | | | |
| IEcoAST1BlockDeclarations\* | | ***get\_Params*** | ([in] void); |
|  | |  |  |
| char\_t\* | | ***get\_TypeName*** | ([in] void); |
|  | |  |  |
| IEcoAST1StructOrUnionDeclaration\* | | ***get\_Declaration*** | ([in] void); |
|  | |  |  |
| int16\_t | | ***Declare*** | ([in] IEcoAST1StructOrUnionDeclaration\* pIDeclaration); |
|  | |  |  |
| uint16\_t | | ***get\_StructOrUnionType*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Params function**

The function

* + 1. **get\_TypeName function**

The function

* + 1. **get\_Declaration function**

The function

* + 1. **Declare function**

The function

* + 1. **get\_StructOrUnionType function**

The function

1. **IEcoAST1PointerType Interface**
   1. **IEcoAST1PointerType IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Type.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(950EC30B-18C9-4CA4-BB6D-63277A29C111),  ] | | | |
| interface IEcoAST1PointerType : IEcoAST1Type{ | | | |
|  | | | |
| IEcoAST1Type\* | | ***get\_IntegralType*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_IntegralType function**

The function

1. **IEcoAST1NumberType Interface**
   1. **IEcoAST1NumberType IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Type.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(950EC30B-18C9-4CA4-BB6D-63277A29C13F),  ] | | | |
| interface IEcoAST1NumberType : IEcoAST1Type{ | | | |
|  | | | |
| bool\_t | | ***IsSigned*** | ([in] void); |
|  | |  |  |
| uint32\_t | | ***get\_Size*** | ([in] void); |
|  | |  |  |
| int16\_t | | ***get\_IntegralType*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **IsSigned function**

The function

* + 1. **get\_Size function**

The function

* + 1. **get\_IntegralType function**

The function

1. **IEcoAST1FunctionType Interface**
   1. **IEcoAST1FunctionType IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoSystem1.idl"  import "IEcoAST1Type.idl"  import "IEcoAST1BlockDeclarations.idl"  import "IEcoAST1Scope.idl" | | | |
| [  object,  uguid(F0BC0E8A-A598-42E1-9676-801CA6B1F55F),  ] | | | |
| interface IEcoAST1FunctionType : IEcoAST1Type{ | | | |
|  | | | |
| IEcoAST1Type\* | | ***get\_ReturnType*** | ([in] void); |
|  | |  |  |
| IEcoAST1BlockDeclarations\* | | ***get\_Params*** | ([in] void); |
|  | |  |  |
| IEcoAST1Scope\* | | ***get\_Scope*** | ([in] void); |
|  | |  |  |
| bool\_t | | ***isVariadic*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_ReturnType function**

The function

* + 1. **get\_Params function**

The function

* + 1. **get\_Scope function**

The function

* + 1. **isVariadic function**

The function

1. **IEcoAST1EnumType Interface**
   1. **IEcoAST1EnumType IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Type.idl"  import "IEcoAST1EnumDeclaration.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1NumberType.idl" | | | |
| [  object,  uguid(950EC30B-18C9-4CA4-BB6D-63277A29C538),  ] | | | |
| interface IEcoAST1EnumType : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1EnumDeclaration\* | | ***get\_Declaration*** | ([in] void); |
|  | |  |  |
| char\_t\* | | ***get\_TypeName*** | ([in] void); |
|  | |  |  |
| int16\_t | | ***Declare*** | (IEcoAST1EnumDeclaration\* pIDeclaration); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Declaration function**

The function

* + 1. **get\_TypeName function**

The function

* + 1. **Declare function**

The function

1. **IEcoAST1ArrayType Interface**
   1. **IEcoAST1ArrayType IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Type.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1Expression.idl" | | | |
| [  object,  uguid(950EC30B-18C9-4CA4-BB6D-63277A29C740),  ] | | | |
| interface IEcoAST1ArrayType : IEcoAST1Type{ | | | |
|  | | | |
| IEcoAST1Type\* | | ***get\_IntegralType*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression\* | | ***get\_CountExpression*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_IntegralType function**

The function

* + 1. **get\_CountExpression function**

The function

1. **IEcoAST1SizeofType Interface**
   1. **IEcoAST1SizeofType IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4690),  ] | | | |
| interface IEcoAST1SizeofType : IEcoAST1Expression{ | | | |
|  | | | |
| IEcoAST1Type\* | | ***get\_InternalType*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_InternalType function**

The function

1. **IEcoAST1Expression Interface**
   1. **IEcoAST1Expression IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Type.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(937549A5-DFF7-41AE-AD21-166A74B0CA20),  ] | | | |
| interface IEcoAST1Expression : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1Type\* | | ***get\_Type*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Type function**

The function

1. **IEcoAST1BlockExpressions Interface**
   1. **IEcoAST1BlockExpressions IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4783),  ] | | | |
| interface IEcoAST1BlockExpressions : IEcoAST1Expression{ | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_Expression*** | ([in] uint32\_t index); |
|  | |  |  |
| uint32\_t | | ***get\_Count*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Expression function**

The function

* + 1. **get\_Count function**

The function

1. **IEcoAST1NullExpression Interface**
   1. **IEcoAST1NullExpression IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C47A3),  ] | | | |
| interface IEcoAST1NullExpression : IEcoAST1Expression{ | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **The function function**

The function

1. **IEcoAST1UnaryExpression Interface**
   1. **IEcoAST1UnaryExpression IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Statement.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(02D70CE4-24FF-4A08-9FBB-601C91E64A43),  ] | | | |
| interface IEcoAST1UnaryExpression : IEcoAST1Expression{ | | | |
|  | | | |
| uint16\_t | | ***get\_Operator*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression \* | | ***get\_Operand*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Operator function**

The function

* + 1. **get\_Operand function**

The function

1. **IEcoAST1SizeofExpression Interface**
   1. **IEcoAST1SizeofExpression IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1VariableDefinition.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4AD1),  ] | | | |
| interface IEcoAST1SizeofExpression : IEcoAST1Expression { | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_InternalExpression*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_InternalExpression function**

The function

1. **IEcoAST1ConditionalExpression Interface**
   1. **IEcoAST1ConditionalExpression IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4FD4),  ] | | | |
| interface IEcoAST1ConditionalExpression : IEcoAST1SyntaxObject { | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_TestOperand*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression \* | | ***get\_TrueOperand*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression\* | | ***get\_FalseOperand*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_TestOperand function**

The function

* + 1. **get\_TrueOperand function**

The function

* + 1. **get\_FalseOperand function**

The function

1. **IEcoAST1CastExpression Interface**
   1. **IEcoAST1CastExpression IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C456C),  ] | | | |
| interface IEcoAST1CastExpression : IEcoAST1Expression{ | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_Operand*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Operand function**

The function

1. **IEcoAST1BinaryExpression Interface**
   1. **IEcoAST1BinaryExpression IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Statement.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(02D70CE4-24FF-4A08-9FBB-601C91E64C68),  ] | | | |
| interface IEcoAST1BinaryExpression : IEcoAST1Expression { | | | |
|  | | | |
| int16\_t | | ***get\_Operator*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression\* | | ***get\_LeftOperand*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression\* | | ***get\_RightOperand*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Operator function**

The function

* + 1. **get\_LeftOperand function**

The function

* + 1. **get\_RightOperand function**

The function

1. **IEcoAST1FunctionCall Interface**
   1. **IEcoAST1FunctionCall IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1FunctionDeclaration.idl"  import "IEcoAST1BlockExpressions.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4358),  ] | | | |
| interface IEcoAST1FunctionCall : IEcoAST1Expression { | | | |
|  | | | |
| IEcoAST1BlockExpressions\* | | ***get\_Arguments*** | ([in] void); |
|  | |  |  |
| IEcoAST1Expression\* | | ***get\_FunctionExpression*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Arguments function**

The function

* + 1. **get\_FunctionExpression function**

The function

1. **IEcoAST1MemberSelection Interface**
   1. **IEcoAST1MemberSelection IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Statement.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(02D70CE4-24FF-4A08-9FBB-601C91E64E76),  ] | | | |
| interface IEcoAST1MemberSelection : IEcoAST1Expression { | | | |
|  | | | |
| IEcoAST1Expression\* | | ***get\_SourceExpression*** | ([in] void); |
|  | |  |  |
| IEcoAST1VariableDefinition\* | | get\_InternalExpression | ([in] void); |
|  | |  |  |
| bool\_t | | ***isDirect*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_SourceExpression function**

The function

* + 1. **get\_InternalExpression function**

The function

* + 1. **isDirect function**

The function

1. **IEcoAST1VariableDefinition Interface**
   1. **IEcoAST1VariableDefinition IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Expression.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1ExpressionableDeclaration.idl" | | | |
| [  object,  uguid(985EF5C8-9E6A-4781-8C54-AE18C31C4ADB),  ] | | | |
| interface IEcoAST1VariableDefinition : IEcoAST1Expression{ | | | |
|  | | | |
| char\_t\* | | ***get\_IdentifierName*** | ([in] void); |
|  | |  |  |
| IEcoAST1ExpressionableDeclaration\* | | ***get\_Declaration*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_IdentifierName function**

The function

* + 1. **get\_Declaration function**

The function

1. **IEcoAST1Literal Interface**
   1. **IEcoAST1Literal IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1Type.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(49BA0D4B-4C5A-42FF-856C-3E6176D91D50),  ] | | | |
| interface IEcoAST1Literal : IEcoAST1Expression { | | | |
|  | | | |
| char\_t\* | | ***get\_Value*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Value function**

The function

1. **IEcoAST1CharacterLiteral Interface**
   1. **IEcoAST1CharacterLiteral IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1Literal.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(86E98192-2759-4F47-ABCC-C68C7A456962),  ] | | | |
| interface IEcoAST1CharacterLiteral : IEcoAST1Literal { | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoAST1FloatLiteral Interface**
   1. **IEcoAST1FloatLiteral IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1Literal.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(86E98192-2759-4F47-ABCC-C68C7A456570),  ] | | | |
| interface IEcoAST1FloatLiteral : IEcoAST1Literal { | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoAST1IntegerLiteral Interface**
   1. **IEcoAST1IntegerLiteral IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1Literal.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(86E98192-2759-4F47-ABCC-C68C7A456962),  ] | | | |
| interface IEcoAST1IntegerLiteral : IEcoAST1Literal{ | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoAST1StringLiteral Interface**
   1. **IEcoAST1StringLiteral IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoAST1Expression.idl"  import "IEcoAST1SingleStatement.idl"  import "IEcoAST1Statement.idl"  import "IEcoAST1Literal.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(86E98192-2759-4F47-ABCC-C68C7A456AD4),  ] | | | |
| interface IEcoAST1StringLiteral : IEcoAST1SyntaxObject { | | | |
|  | | | |
| char\_t\* | | ***get\_Value*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Value function**

The function

1. **IEcoAST1FunctionDefinition Interface**
   1. **IEcoAST1FunctionDefinition IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoAST1FunctionDeclaration.idl"  import "IEcoAST1CompoundStatement.idl"  import "IEcoAST1Scope.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CE56834),  ] | | | |
| interface IEcoAST1FunctionDefinition : IEcoAST1SyntaxObject { | | | |
|  | | | |
| IEcoAST1FunctionDeclaration\* | | ***get\_Declaration*** | ([in] void); |
|  | |  |  |
| IEcoAST1CompoundStatement\* | | ***get\_Body*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Declaration function**

The function

* + 1. **get\_Body function**

The function

1. **IEcoAST1TranslationUnitObject Interface**
   1. **IEcoAST1TranslationUnitObject IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoAST1SyntaxObject.idl"  import "IEcoLexicalAnalyzer1.idl" | | | |
| [  object,  uguid(82AA8D91-4E04-44BD-B3DF-EA237CE94876),  ] | | | |
| interface IEcoAST1TranslationUnitObject : IEcoAST1SyntaxObject { | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoAST1BlockTranslationUnitObjects Interface**
   1. **IEcoAST1BlockTranslationUnitObjects IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoLexicalAnalyzer1.idl"  import "IEcoSystem1.idl"  import "IEcoAST1TranslationUnitObject.idl" | | | |
| [  object,  uguid(F0BC0E8A-A598-42E1-9676-801CA6BFFFAA),  ] | | | |
| interface IEcoAST1BlockTranslationUnitObjects : IEcoAST1SyntaxObject{ | | | |
|  | | | |
| IEcoAST1TranslationUnitObject\* | | ***get\_Object*** | ([in] uint32\_t index); |
|  | |  |  |
| uint32\_t | | ***get\_Count*** | ([in] void); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Object function**

The function

* + 1. **get\_Count function**

The function

1. **Коды ошибок**

Следующая таблица содержит коды ошибок.

|  |  |  |
| --- | --- | --- |
| **Код ошибки** | **Значение** | **Описание** |
| ERR\_ECO\_SUCCESES | 0x0000 | Выполнено успешно. Ошибок нет. |
| ERR\_ECO\_UNEXPECTED | 0xFFFF | Непредвиденное условие. |
| ERR\_ECO\_POINTER | 0xFFEE | Было передано неправильное значение указателя. |
| ERR\_ECO\_NOINTERFACE | 0xFFED | Такой интерфейс не поддерживается. |
| ERR\_ECO\_COMPONENT\_NOTFOUND | 0xFFE9 | Компонент не найден. |
|  |  |  |
|  |  |  |

# **Приложение А: Обучающие программы**