
hybrid

Release 0.0.1

Iain Moncrief, Nick Burrell

Feb 09, 2022

CONTENTS:

1	API Reference	1
1.1	hybrid	1
2	Indices and tables	11
	Python Module Index	13
	Index	15

API REFERENCE

This page contains auto-generated API reference documentation¹.

1.1 hybrid

1.1.1 Submodules

`hybrid.language`

Module Contents

Functions

<code>definition_libraries(ast)</code>	→
<code>list[hybrid.syntax.LibraryDefinition]</code>	
<code>definition_calling_programs(ast)</code>	→
<code>list[hybrid.syntax.CallingProgramDefinition]</code>	

`hybrid.language.definition_libraries(ast)` → `list[hybrid.syntax.LibraryDefinition]`

`hybrid.language.definition_calling_programs(ast)` → `list[hybrid.syntax.CallingProgramDefinition]`

`hybrid.latex`

Module Contents

Classes

<code>Stylizer</code>
<code>Relation</code>
<code>HighlightExpression</code>

continues on next page

¹ Created with sphinx-autoapi

Table 2 – continued from previous page

Link

Interchangable

Indistinguishable

Functions

bin_op_tex(op: str) → str

texify_relation(relation: str) → str

texify(ast: hybrid.syntax.AST, queries_only=True) → str

link(prog: hybrid.syntax.AST, lib: hybrid.syntax.LibraryDefinition, glowing=False) → str

equiv(left: hybrid.syntax.AST, right: hybrid.syntax.AST, align=False) → str

save_file(tex_source, file)

Attributes

context

is_in_math_block

depth

class hybrid.latex.StylizerBases: *hybrid.syntax.AST***class** hybrid.latex.RelationBases: *Stylizer***class** hybrid.latex.HighlightExpressionBases: *Stylizer***expression** :hybrid.syntax.AST**class** hybrid.latex.LinkBases: *Relation***lhs** :hybrid.syntax.AST**rhs** :hybrid.syntax.AST**class** hybrid.latex.InterchangableBases: *Relation*

```

    lhs :hybrid.syntax.AST
    rhs :hybrid.syntax.AST
class hybrid.latex.Indistinguishable
    Bases: Relation
    lhs :hybrid.syntax.AST
    rhs :hybrid.syntax.AST
hybrid.latex.bin_op_tex(op: str) → str
hybrid.latex.texify_relation(relation: str) → str
hybrid.latex.context
hybrid.latex.is_in_math_block = False
hybrid.latex.depth = 0
hybrid.latex.texify(ast: hybrid.syntax.AST, queries_only=True) → str
hybrid.latex.link(prog: hybrid.syntax.AST, lib: hybrid.syntax.LibraryDefinition, glowing=False) → str
hybrid.latex.equiv(left: hybrid.syntax.AST, right: hybrid.syntax.AST, align=False) → str
hybrid.latex.save_file(tex_source, file)

```

hybrid.parser

Module Contents

Classes

RawSyntaxTreeVisitor

Functions

get_rule_name(*node: antlr4.tree.Tree.Tree*) → str

get_children_derivation_names(*node:*
antlr4.tree.Tree.Tree)
copy_context(*node*)

expand_all_derivations(*node:*
antlr4.ParserRuleContext.ParserRuleContext)
non_terminal_children(*ctx*)

file_to_ast_dict(*file*)

Attributes

NonTerminalName

`hybrid.parser.NonTerminalName`

`hybrid.parser.get_rule_name(node: antlr4.tree.Tree.Tree) → str`

`hybrid.parser.get_children_derivation_names(node: antlr4.tree.Tree.Tree)`

`hybrid.parser.copy_context(node)`

`hybrid.parser.expand_all_derivations(node: antlr4.ParserRuleContext.ParserRuleContext)`

`hybrid.parser.non_terminal_children(ctx)`

class `hybrid.parser.RawSyntaxTreeVisitor`

Bases: `hybrid.parsing.HybLangVisitor.HybLangVisitor`

visitProgram(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.ProgramContext`)

visitProcedure(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.ProcedureContext`)

visitCall_prog_def(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Call_prog_defContext`)

visitLibrary_def(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Library_defContext`)

visitBlock(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.BlockContext`)

visitStatement(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.StatementContext`)

visitAssignment(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.AssignmentContext`)

visitSample(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.SampleContext`)

visitSet_type(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Set_typeContext`)

visitSet_lit(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Set_litContext`)

visitExpression(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.ExpressionContext`)

visitBin_op(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Bin_opContext`)

visitFunction_call(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Function_callContext`)

visitReturn_stmt(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Return_stmtContext`)

visitFunction_arguments(self, ctx:
`hybrid.parsing.HybLangParser.HybLangParser.Function_argumentsContext`)

visitFunction_argument(self, ctx:
`hybrid.parsing.HybLangParser.HybLangParser.Function_argumentContext`)

visitFunction_def(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Function_defContext`)

visitTerminal(self, node: `antlr4.tree.Tree.TerminalNodeImpl`)

visitIf_stmt(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.If_stmtContext`)

visitQuery_statement(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Query_statementContext`)

visitShow_query_statement(self, ctx: `hy-`
`brid.parsing.HybLangParser.HybLangParser.Show_query_statementContext`)

visitQuery_relation(self, ctx: `hybrid.parsing.HybLangParser.HybLangParser.Query_relationContext`)


```

visitQuery_expression(self, ctx:
    hybrid.parsing.HybLangParser.HybLangParser.Query_expressionContext)
visitString_literal(self, ctx: hybrid.parsing.HybLangParser.HybLangParser.String_literalContext)
visitWrite_query_statement(self, ctx: hy-
    brid.parsing.HybLangParser.HybLangParser.Write_query_statementContext)
aggregateResult(self, aggregate, nextResult)
defaultResult(self)
hybrid.parser.file_to_ast_dict(file)

```

hybrid.syntax

Module Contents

Classes

<i>AST</i>
<i>Statement</i>
<i>NonStatement</i>
<i>Block</i>
<i>Definitions</i>
<i>Expression</i>
<i>Literal</i>
<i>Identifier</i>
<i>Number</i>
<i>String</i>
<i>BinaryOperation</i>
<i>FunctionCall</i>
<i>Assignment</i>
<i>Sample</i>
<i>Return</i>
<i>SetAtom</i>

continues on next page

Table 8 – continued from previous page

NumberSetAtom

IdentifierSetAtom

SetLiteral

FunctionArgument

FunctionArguments

FunctionDefinition

IfStatement

CallingProgramDefinition

LibraryDefinition

QueryExpression

QueryIdentifier

QueryStatement

ProgramReference

LibraryReference

QueryRelation

Write

ShowQueryStatement

WriteQueryStatement

Functions

construct_ast(parse_tree) → Any

class hybrid.syntax.AST**class** hybrid.syntax.StatementBases: *AST***class** hybrid.syntax.NonStatementBases: *AST***class** hybrid.syntax.BlockBases: *AST*

```

    procedure :list[Statement | NonStatement]

class hybrid.syntax.Definitions
    Bases: AST

    body :Block

class hybrid.syntax.Expression
    Bases: Statement

class hybrid.syntax.Literal
    Bases: Expression

class hybrid.syntax.Identifier
    Bases: Literal

    id :str

class hybrid.syntax.Number
    Bases: Literal

    num :int

class hybrid.syntax.String
    Bases: Literal

    string :str

class hybrid.syntax.BinaryOperation
    Bases: Expression

    lhs :Expression

    bin_op :str

    rhs :Expression

class hybrid.syntax.FunctionCall
    Bases: Expression

    fun_name :str

    args :list[Expression]

class hybrid.syntax.Assignment
    Bases: Statement

    lhs :Identifier

    rhs :Expression

class hybrid.syntax.Sample
    Bases: Statement

    lhs :Identifier

    rhs :Expression | Any

class hybrid.syntax.Return
    Bases: Statement

    ret :Expression

class hybrid.syntax.SetAtom
    Bases: AST

```

```
class hybrid.syntax.NumberSetAtom
    Bases: SetAtom

    num :int

class hybrid.syntax.IdentifierSetAtom
    Bases: SetAtom

    id :str

class hybrid.syntax.SetLiteral
    Bases: AST

    data :list[SetAtom]

class hybrid.syntax.FunctionArgument
    Bases: AST

    arg_id :str
    arg_type :Optional[SetLiteral]

class hybrid.syntax.FunctionArguments
    Bases: AST

    args :list[tuple[int, FunctionArgument]]

class hybrid.syntax.FunctionDefinition
    Bases: NonStatement

    fun_name :str
    args :FunctionArguments
    block :Block

class hybrid.syntax.IfStatement
    Bases: NonStatement

    condition :Expression
    then_block :Block
    else_block :Optional[Block]

class hybrid.syntax.CallingProgramDefinition
    Bases: NonStatement

    name :str
    body :Block

class hybrid.syntax.LibraryDefinition
    Bases: NonStatement

    name :str
    exposing :list[tuple[str, str]]
    body :Block

class hybrid.syntax.QueryExpression
    Bases: AST

class hybrid.syntax.QueryIdentifier
    Bases: QueryExpression
```

```
class hybrid.syntax.QueryStatement
  Bases: NonStatement

class hybrid.syntax.ProgramReference
  Bases: QueryIdentifier
  name :str

class hybrid.syntax.LibraryReference
  Bases: QueryIdentifier
  name :str

class hybrid.syntax.QueryRelation
  Bases: QueryExpression
  lhs :QueryExpression
  relation :str
  rhs :QueryExpression

class hybrid.syntax.Write
  Bases: QueryExpression
  body :String

class hybrid.syntax.ShowQueryStatement
  Bases: QueryStatement
  expression :QueryExpression

class hybrid.syntax.WriteQueryStatement
  Bases: QueryStatement
  body :Write

hybrid.syntax.construct_ast(parse_tree) → Any
```


INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

h

- `hybrid`, 1
- `hybrid.language`, 1
- `hybrid.latex`, 1
- `hybrid.parser`, 3
- `hybrid.syntax`, 5

A

`aggregateResult()` (*hybrid.parser.RawSyntaxTreeVisitor* method), 5
`arg_id` (*hybrid.syntax.FunctionArgument* attribute), 8
`arg_type` (*hybrid.syntax.FunctionArgument* attribute), 8
`args` (*hybrid.syntax.FunctionArguments* attribute), 8
`args` (*hybrid.syntax.FunctionCall* attribute), 7
`args` (*hybrid.syntax.FunctionDefinition* attribute), 8
`Assignment` (class in *hybrid.syntax*), 7
`AST` (class in *hybrid.syntax*), 6

B

`bin_op` (*hybrid.syntax.BinaryOperation* attribute), 7
`bin_op_tex()` (in module *hybrid.latex*), 3
`BinaryOperation` (class in *hybrid.syntax*), 7
`Block` (class in *hybrid.syntax*), 6
`block` (*hybrid.syntax.FunctionDefinition* attribute), 8
`body` (*hybrid.syntax.CallingProgramDefinition* attribute), 8
`body` (*hybrid.syntax.Definitions* attribute), 7
`body` (*hybrid.syntax.LibraryDefinition* attribute), 8
`body` (*hybrid.syntax.Write* attribute), 9
`body` (*hybrid.syntax.WriteQueryStatement* attribute), 9

C

`CallingProgramDefinition` (class in *hybrid.syntax*), 8
`condition` (*hybrid.syntax.IfStatement* attribute), 8
`construct_ast()` (in module *hybrid.syntax*), 9
`context` (in module *hybrid.latex*), 3
`copy_context()` (in module *hybrid.parser*), 4

D

`data` (*hybrid.syntax.SetLiteral* attribute), 8
`defaultResult()` (*hybrid.parser.RawSyntaxTreeVisitor* method), 5
`definition_calling_programs()` (in module *hybrid.language*), 1
`definition_libraries()` (in module *hybrid.language*), 1
`Definitions` (class in *hybrid.syntax*), 7
`depth` (in module *hybrid.latex*), 3

E

`else_block` (*hybrid.syntax.IfStatement* attribute), 8
`equiv()` (in module *hybrid.latex*), 3
`expand_all_derivations()` (in module *hybrid.parser*), 4
`exposing` (*hybrid.syntax.LibraryDefinition* attribute), 8
`Expression` (class in *hybrid.syntax*), 7
`expression` (*hybrid.latex.HighlightExpression* attribute), 2
`expression` (*hybrid.syntax.ShowQueryStatement* attribute), 9

F

`file_to_ast_dict()` (in module *hybrid.parser*), 5
`fun_name` (*hybrid.syntax.FunctionCall* attribute), 7
`fun_name` (*hybrid.syntax.FunctionDefinition* attribute), 8
`FunctionArgument` (class in *hybrid.syntax*), 8
`FunctionArguments` (class in *hybrid.syntax*), 8
`FunctionCall` (class in *hybrid.syntax*), 7
`FunctionDefinition` (class in *hybrid.syntax*), 8

G

`get_children_derivation_names()` (in module *hybrid.parser*), 4
`get_rule_name()` (in module *hybrid.parser*), 4

H

`HighlightExpression` (class in *hybrid.latex*), 2
`hybrid`
 module, 1
`hybrid.language`
 module, 1
`hybrid.latex`
 module, 1
`hybrid.parser`
 module, 3
`hybrid.syntax`
 module, 5

I

`id` (*hybrid.syntax.Identifier* attribute), 7

id (*hybrid.syntax.IdentifierSetAtom* attribute), 8
 Identifier (*class in hybrid.syntax*), 7
 IdentifierSetAtom (*class in hybrid.syntax*), 8
 IfStatement (*class in hybrid.syntax*), 8
 Indistinguishable (*class in hybrid.latex*), 3
 Interchangable (*class in hybrid.latex*), 2
 is_in_math_block (*in module hybrid.latex*), 3

L

lhs (*hybrid.latex.Indistinguishable* attribute), 3
 lhs (*hybrid.latex.Interchangable* attribute), 2
 lhs (*hybrid.latex.Link* attribute), 2
 lhs (*hybrid.syntax.Assignment* attribute), 7
 lhs (*hybrid.syntax.BinaryOperation* attribute), 7
 lhs (*hybrid.syntax.QueryRelation* attribute), 9
 lhs (*hybrid.syntax.Sample* attribute), 7
 LibraryDefinition (*class in hybrid.syntax*), 8
 LibraryReference (*class in hybrid.syntax*), 9
 Link (*class in hybrid.latex*), 2
 link() (*in module hybrid.latex*), 3
 Literal (*class in hybrid.syntax*), 7

M

module
 hybrid, 1
 hybrid.language, 1
 hybrid.latex, 1
 hybrid.parser, 3
 hybrid.syntax, 5

N

name (*hybrid.syntax.CallingProgramDefinition* attribute), 8
 name (*hybrid.syntax.LibraryDefinition* attribute), 8
 name (*hybrid.syntax.LibraryReference* attribute), 9
 name (*hybrid.syntax.ProgramReference* attribute), 9
 non_terminal_children() (*in module hybrid.parser*), 4
 NonStatement (*class in hybrid.syntax*), 6
 NonTerminalName (*in module hybrid.parser*), 4
 num (*hybrid.syntax.Number* attribute), 7
 num (*hybrid.syntax.NumberSetAtom* attribute), 8
 Number (*class in hybrid.syntax*), 7
 NumberSetAtom (*class in hybrid.syntax*), 7

P

procedure (*hybrid.syntax.Block* attribute), 6
 ProgramReference (*class in hybrid.syntax*), 9

Q

QueryExpression (*class in hybrid.syntax*), 8
 QueryIdentifier (*class in hybrid.syntax*), 8
 QueryRelation (*class in hybrid.syntax*), 9

QueryStatement (*class in hybrid.syntax*), 8

R

RawSyntaxTreeVisitor (*class in hybrid.parser*), 4
 Relation (*class in hybrid.latex*), 2
 relation (*hybrid.syntax.QueryRelation* attribute), 9
 ret (*hybrid.syntax.Return* attribute), 7
 Return (*class in hybrid.syntax*), 7
 rhs (*hybrid.latex.Indistinguishable* attribute), 3
 rhs (*hybrid.latex.Interchangable* attribute), 3
 rhs (*hybrid.latex.Link* attribute), 2
 rhs (*hybrid.syntax.Assignment* attribute), 7
 rhs (*hybrid.syntax.BinaryOperation* attribute), 7
 rhs (*hybrid.syntax.QueryRelation* attribute), 9
 rhs (*hybrid.syntax.Sample* attribute), 7

S

Sample (*class in hybrid.syntax*), 7
 save_file() (*in module hybrid.latex*), 3
 SetAtom (*class in hybrid.syntax*), 7
 SetLiteral (*class in hybrid.syntax*), 8
 ShowQueryStatement (*class in hybrid.syntax*), 9
 Statement (*class in hybrid.syntax*), 6
 String (*class in hybrid.syntax*), 7
 string (*hybrid.syntax.String* attribute), 7
 Stylizer (*class in hybrid.latex*), 2

T

texify() (*in module hybrid.latex*), 3
 texify_relation() (*in module hybrid.latex*), 3
 then_block (*hybrid.syntax.IfStatement* attribute), 8

V

visitAssignment() (*hybrid.parser.RawSyntaxTreeVisitor* method), 4
 visitBin_op() (*hybrid.parser.RawSyntaxTreeVisitor* method), 4
 visitBlock() (*hybrid.parser.RawSyntaxTreeVisitor* method), 4
 visitCall_prog_def() (*hybrid.parser.RawSyntaxTreeVisitor* method), 4
 visitExpression() (*hybrid.parser.RawSyntaxTreeVisitor* method), 4
 visitFunction_argument() (*hybrid.parser.RawSyntaxTreeVisitor* method), 4
 visitFunction_arguments() (*hybrid.parser.RawSyntaxTreeVisitor* method), 4

```

visitFunction_call()          (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitFunction_def()          (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitIf_stmt() (hybrid.parser.RawSyntaxTreeVisitor
    method), 4
visitLibrary_def()          (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitProcedure()            (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitProgram() (hybrid.parser.RawSyntaxTreeVisitor
    method), 4
visitQuery_expression()      (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitQuery_relation()        (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitQuery_statement()       (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitReturn_stmt()          (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitSample() (hybrid.parser.RawSyntaxTreeVisitor
    method), 4
visitSet_lit() (hybrid.parser.RawSyntaxTreeVisitor
    method), 4
visitSet_type() (hybrid.parser.RawSyntaxTreeVisitor
    method), 4
visitShow_query_statement()  (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitStatement()            (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
4
visitString_literal()        (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
5
visitTerminal() (hybrid.parser.RawSyntaxTreeVisitor
    method), 4
visitWrite_query_statement()  (hy-
    brid.parser.RawSyntaxTreeVisitor    method),
5

```

W

Write (class in hybrid.syntax), 9

WriteQueryStatement (class in hybrid.syntax), 9