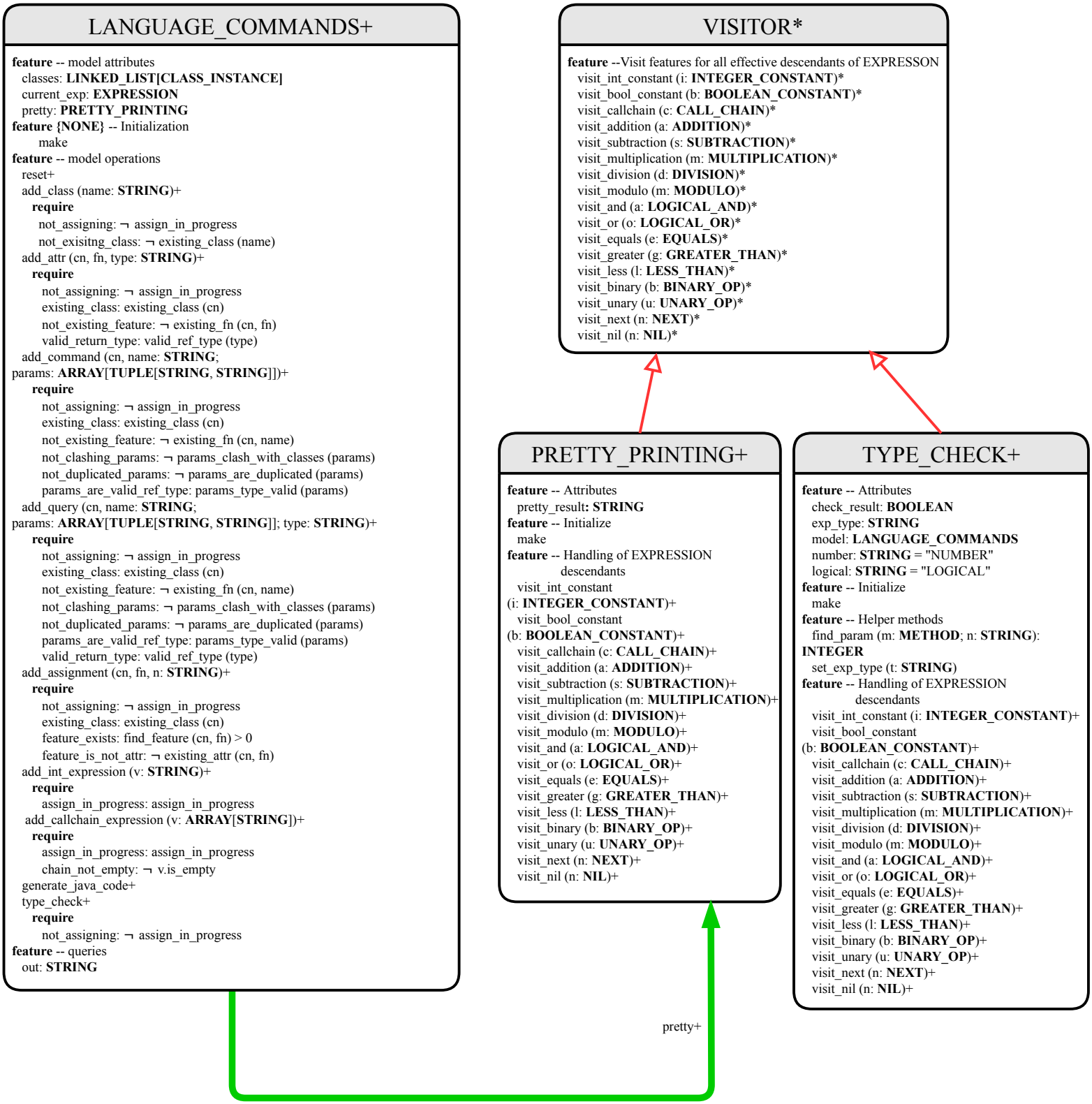
LANGUAGE_COMMANDS+
feature -- model attributes
classes: LINKED_LIST[CLASS_INSTANCE]
current_exp: EXPRESSION
pretty: PRETTY_PRINTING
feature {NONE} -- Initialization
make
feature -- model operations
reset+
add_class (name: STRING)+
require
not_assigning: ¬ assign_in_progress
not_exisitng_class: ¬ existing_class (name)
add_attr (cn, fn, type: STRING)+
require
not_assigning: ¬ assign_in_progress
existing_class: existing_class (cn)
not_existing_feature: ¬ existing_fn (cn, fn)
valid_return_type: valid_ref_type (type)
add_command (cn, name: STRING;
params: ARRAY[TUPLE[STRING, STRING]])+
require
not_assigning: ¬ assign_in_progress
existing_class: existing_class (cn)
not_existing_feature: ¬ existing_fn (cn, name)
not_clashing_params: ¬ params_clash_with_classes (params)
not_duplicated_params: ¬ params_are_duplicated (params)
params_are_valid_ref_type: params_type_valid (params)
add_query (cn, name: STRING;
params: ARRAY[TUPLE[STRING, STRING]]; type: STRING)+
require
not_assigning: ¬ assign_in_progress
existing_class: existing_class (cn)
not_existing_feature: ¬ existing_fn (cn, name)
not_clashing_params: ¬ params_clash_with_classes (params)
not_duplicated_params: ¬ params_are_duplicated (params)
params_are_valid_ref_type: params_type_valid (params)
valid_return_type: valid_ref_type (type)
add_assignment (cn, fn, n: STRING)+
require
not_assigning: ¬ assign_in_progress
existing_class: existing_class (cn)
feature_exists: find_feature (cn, fn) > 0
feature_is_not_attr: ¬ existing_attr (cn, fn)
add_int_expression (v: STRING)+
require
assign_in_progress: assign_in_progress
add_callchain_expression (v: ARRAY[STRING])+
require
assign_in_progress: assign_in_progress
chain_not_empty: ¬ v.is_empty
generate_java_code+
type_check+
require
not_assigning: ¬ assign_in_progress
feature -- queries
out: STRING
VISITOR*
feature --Visit features for all effective descendants of EXPRESSON
visit_int_constant (i: INTEGER_CONSTANT)*
visit_bool_constant (b: BOOLEAN_CONSTANT)*
visit_callchain (c: CALL_CHAIN)*
visit_addition (a: ADDITION)*
visit_subtraction (s: SUBTRACTION)*
visit_multiplication (m: MULTIPLICATION)*
visit_division (d: DIVISION)*
visit_modulo (m: MODULO)*
visit_and (a: LOGICAL_AND)*
visit_or (o: LOGICAL_OR)*
visit_equals (e: EQUALS)*
visit_greater (g: GREATER_THAN)*
visit_less (l: LESS_THAN)*
visit_binary (b: BINARY_OP)*
visit_unary (u: UNARY_OP)*
visit_next (n: NEXT)*
visit_nil (n: NIL)*
PRETTY_PRINTING+
feature -- Attributes
pretty_result: STRING
feature -- Initialize
make
feature -- Handling of EXPRESSION
descendants
visit_int_constant
(i: INTEGER_CONSTANT)+
visit_bool_constant
(b: BOOLEAN_CONSTANT)+
visit_callchain (c: CALL_CHAIN)+
visit_addition (a: ADDITION)+
visit_subtraction (s: SUBTRACTION)+
visit_multiplication (m: MULTIPLICATION)+
visit_division (d: DIVISION)+
visit_modulo (m: MODULO)+
visit_and (a: LOGICAL_AND)+
visit_or (o: LOGICAL_OR)+
visit_equals (e: EQUALS)+
visit_greater (g: GREATER_THAN)+
visit_less (l: LESS_THAN)+
visit_binary (b: BINARY_OP)+
visit_unary (u: UNARY_OP)+
visit_next (n: NEXT)+
visit_nil (n: NIL)+
TYPE_CHECK+
feature -- Attributes
check_result: BOOLEAN
exp_type: STRING
model: LANGUAGE_COMMANDS
number: STRING = "NUMBER"
logical: STRING = "LOGICAL"
feature -- Initialize
make
feature -- Helper methods
find_param (m: METHOD; n: STRING):
INTEGER
set_exp_type (t: STRING)
feature -- Handling of EXPRESSION
descendants
visit_int_constant (i: INTEGER_CONSTANT)+
visit_bool_constant
(b: BOOLEAN_CONSTANT)+
visit_callchain (c: CALL_CHAIN)+
visit_addition (a: ADDITION)+
visit_subtraction (s: SUBTRACTION)+
visit_multiplication (m: MULTIPLICATION)+
visit_division (d: DIVISION)+
visit_modulo (m: MODULO)+
visit_and (a: LOGICAL_AND)+
visit_or (o: LOGICAL_OR)+
visit_equals (e: EQUALS)+
visit_greater (g: GREATER_THAN)+
visit_less (l: LESS_THAN)+
visit_binary (b: BINARY_OP)+
visit_unary (u: UNARY_OP)+
visit_next (n: NEXT)+
visit_nil (n: NIL)+
pretty+