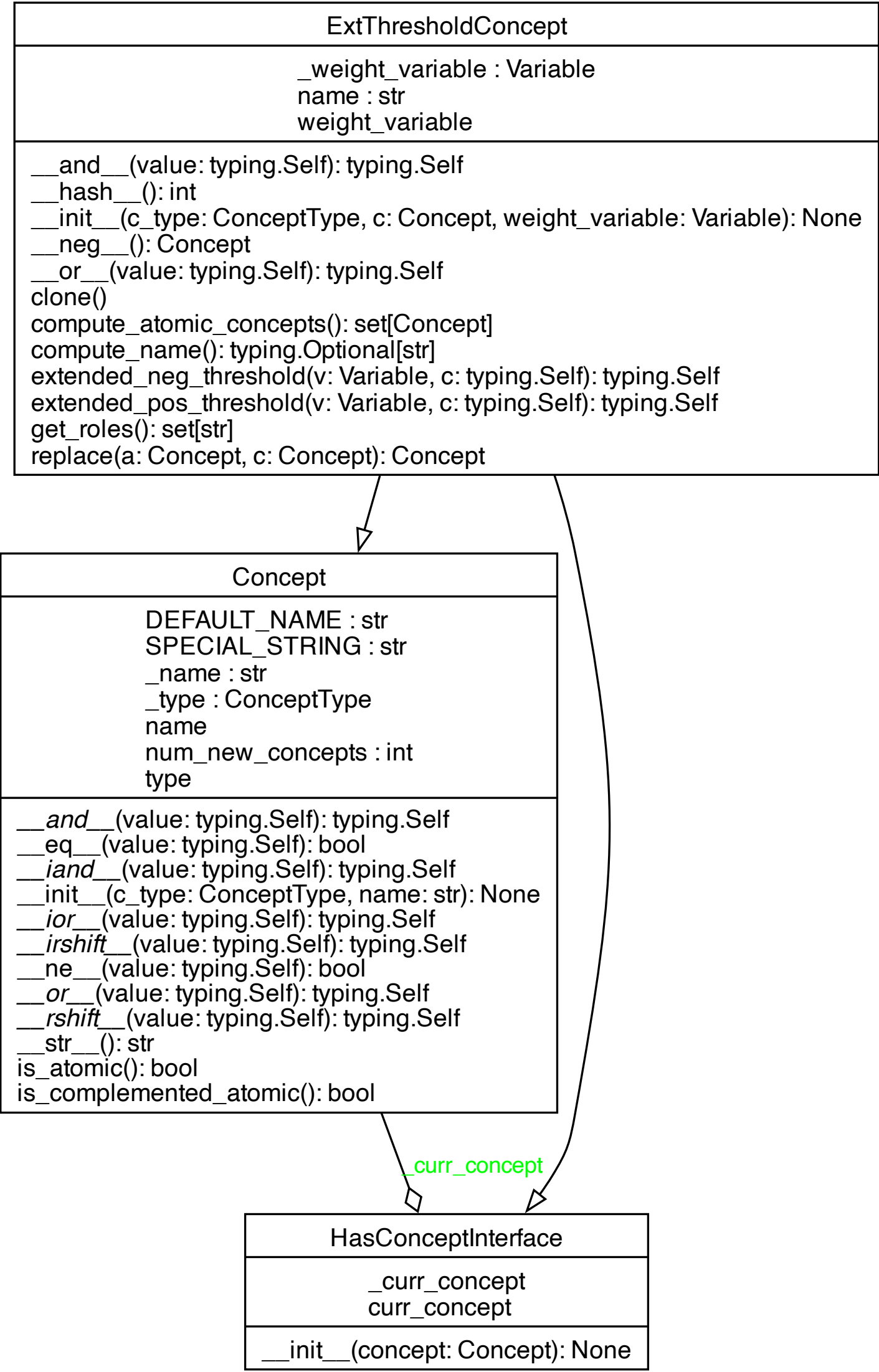
ExtThresholdConcept
_weight_variable : Variable
name : str
weight_variable
__and__(value: typing.Self): typing.Self
__hash__(): int
__init__(c_type: ConceptType, c: Concept, weight_variable: Variable): None
__neg__(): Concept
__or__(value: typing.Self): typing.Self
clone()
compute_atomic_concepts(): set[Concept]
compute_name(): typing.Optional[str]
extended_neg_threshold(v: Variable, c: typing.Self): typing.Self
extended_pos_threshold(v: Variable, c: typing.Self): typing.Self
get_roles(): set[str]
replace(a: Concept, c: Concept): Concept
Concept
DEFAULT_NAME : str
SPECIAL_STRING : str
_name : str
_type : ConceptType
name
num_new_concepts : int
type
__and__(value: typing.Self): typing.Self
__eq__(value: typing.Self): bool
__iand__(value: typing.Self): typing.Self
__init__(c_type: ConceptType, name: str): None
__ior__(value: typing.Self): typing.Self
__irshift__(value: typing.Self): typing.Self
__ne__(value: typing.Self): bool
__or__(value: typing.Self): typing.Self
__rshift__(value: typing.Self): typing.Self
__str__(): str
is_atomic(): bool
is_complemented_atomic(): bool
_curr_concept
HasConceptInterface
_curr_concept
curr_concept
__init__(concept: Concept): None