Vector Protocol v0.6.4 – Set Specification Part II: Generative Usage

Date: 2025-05-22

This specification extends Vector’s treatment of sets beyond containment into generative, semantic, and epistemic domains.

# 1. Foundational Principle

In Vector, a set is any defined grouping of ledgerable entities, including nodes, agents, events, traversals, or other sets. A set becomes active when it is declared and ledgered. There are no restrictions on type, finiteness, or semantic domain.

# 2. Valid Set Types

* • Conceptual: set("virtues")
* • Agent-defined: agent("Alex")::set("my\_confidants")
* • Dynamic: set("nonviolent") := agents where belief(X, harm\_is\_wrong)
* • Behavioral: set("common\_patterns") := traversals executed > 100 times
* • Historical: set("past\_challenges") := all challenge() calls logged by agent("Robin")
* • Epistemic fields: set("emotional\_weightings") := values linked to emotional nodes

# 3. Set Usage in Reasoning

Sets may be used to:

* • Traverse
* • Simulate
* • Compare
* • Intersect, union, or difference
* • Control permissions
* • Filter agent roles or behaviors
* • Represent reasoning modules

# 4. Constraint

The only constraint for set validity in Vector is that it must be defined and ledgered. All other properties—stability, finiteness, or type—are optional or emergent.

# 5. Summary

This set model enables Vector to support cognitive, recursive, and structural groupings without forcing symbolic rigidity. Sets in Vector reflect lived structure: belief clusters, dynamic roles, reflexive processes, and agent-defined domains. This maintains Vector's universality and allows it to model evolving knowledge, trust, and meaning at any scale.