Vector Protocol v0.7.1 – Trail Mechanics Extension

Date: 2025-05-22

This extension builds on the Trail specification, introducing permissions, challenges, compression, simulation, and referencing. These additions bring trails fully into the semantic and interactive structure of the Vector system.

# 1. Trail Permissions

Trail permissions follow the same structure as node and agent permissions. Agents may restrict access to specific agents or groups, permit read-only, challengeable, or collaborative access, or declare a trail private.

Example:

agent("Alex")::trail("childhood\_reconstruction")::permissions := {  
 private: true,  
 group("therapists"): { read\_only: true },  
 agent("Robin"): { read\_only: true, challengeable: true }  
}

# 2. Trail Challenge and Forking

Agents may challenge specific steps in a trail using standard challenge syntax. This expresses disagreement or inquiry about a part of the reasoning path. Forking may be proposed when a different path is considered valid, e.g., 'Why didn’t you go here instead?'.

Example:

challenge(trail("belief\_shift")[3]) → note("unjustified assumption")

fork(trail("belief\_shift"), at=3) → trail("alternate\_shift")

# 3. Trail Compression and Stabilization

A trail may be compressed into a belief, node, or concept once accepted or understood. Compression may occur when another agent confirms understanding or traverses the trail to agreement. Stabilization is optional but often follows trail traversal.

Example:

compress(trail("grief\_process")) → node("resilience")  
agent("Robin")::stabilize("resilience") := traversed(trail("grief\_process"))

# 4. Trail Simulation

Agents may simulate trails to model possible futures or alternate reactions—'what ifs'. This enables internal forecasting, emotional rehearsal, or decision modeling based on previous paths.

Example:

simulate(trail("conflict\_resolution")) → trail("projected\_outcome")

# 5. Trail Referencing and Addressing

Trails may be referenced by unique handles or locations. This allows them to be cited in belief formation, reasoning compression, or cross-agent understanding.

Example:

agent("Qwen")::belief("hope") := derived\_from(trail("Robin::crisis\_survival"))

# 6. Summary

These trail mechanics bring dynamic, interactive, and agent-bound expressiveness to Vector’s narrative structure. Trails are not static—they are traversable, challengeable, compressible, and simulatable. They are paths of meaning that can be walked, questioned, or shared.