TDM-Native Sequential Reasoning: A White Paper for the ::RS- Token

Version: 0.2.0 (Draft)

Status: Proposed

Abstract

This document provides a comprehensive specification for the ::RS- (Reasoning Step) protocol within the Token Decoder Maps (TDM) framework. The objective is to establish a TDM-native method for an AI agent to perform and log an explicit, step-by-step "chain of thought" when executing complex tasks. This protocol addresses the need for greater transparency and auditability in agentic workflows by externalizing the AI's cognitive process into a structured, human-readable log. This system replicates the functionality of external sequential reasoning servers directly within the TDM ecosystem, using a new ::RS- token type and a suite of orchestrating ::SY- and ::FX- tokens.

1. The Problem: The "Black Box" of AI Reasoning

While TDM's ::FX- tokens can guide an AI's reasoning style, the actual step-by-step process the AI follows to reach a conclusion often remains a "black box". For complex, high-stakes, or long-running tasks, this lack of transparency makes it difficult to debug errors, audit the AI's decisions, or trust its output. A mechanism is needed to force the agent to "show its work" in a structured and persistent format.

2. The Solution: A New Token Prefix ::RS- (Reasoning Step)

To address this, we propose a new token prefix to semantically separate reasoning logs from project tasks (::MX-) and knowledge entities (::EN-).

* ::RS- (Reasoning Step): A token that represents a single, discrete step in a sequential reasoning process. These tokens function as entries in a cognitive log.

::RS- Token Template:

::RS-STEP-ID::

- **Parent-Process:** [The ID of the overall reasoning session]
- **Sequence:** [An integer representing the order of the step, e.g., 1, 2, 3]
- **Thought:** [The AI's articulated thought or plan for this step, in natural language.]
- **Action:** [The specific tool or TDM token the AI decides to use to execute the thought.]
- **Observation:** [The result, output, or data returned from executing the Action.]
- **Status:** [Completed | Revised | Branched]

3. The Orchestration Protocol

The ::RS- protocol is managed by a suite of dedicated ::SY- and ::FX- tokens that control the agent's behavior.

The Cognitive Driver:

This ::FX- token instructs the AI to adopt a sequential reasoning methodology.

::FX-SEQUENTIAL-REASONING-LOOP::

- **Type:** ProblemSolvingMethod
- **Summary:** When this token is active, the agent MUST break down the user's goal into a sequence of discrete steps. For each step, it must articulate a "Thought," choose an "Action," and record the "Observation." It must log each completed step using

`::SY-REASONING-ADD-STEP::` and review the entire log before deciding on the next step.
- **Tags:** #Cognitive, #Agentic, #Sequential, #CoT

The System Tools:
These ::SY- tokens are the tools the agent uses to create and manage the reasoning log.
::SY-REASONING-START-PROCESS::
- **Type:** System Protocol
- **Summary:** Creates a new, dedicated reasoning log file (e.g., `reasoning_log_[TIMESTAMP].md`) and a parent entry to track the overall goal.
- **Tags:** #RS, #Logging, #System

::SY-REASONING-ADD-STEP::
- **Type:** System Protocol
- **Summary:** Appends a new, fully formatted `::RS-STEP-ID::` token to the active reasoning log, using the Thought, Action, and Observation from the current step.
- **Tags:** #RS, #Logging, #System

4. The Complete Workflow in Practice
 * A user gives the gemini-cli agent a complex goal, such as "Analyze the attached document and produce a summary," with the ::FX-SEQUENTIAL-REASONING-LOOP:: active.
 * The agent, following the ::FX- instructions, first calls ::SY-REASONING-START-PROCESS::, creating a new log file.
 * Step 1: The agent generates its first thought and logs it by calling ::SY-REASONING-ADD-STEP::. The resulting token in the log file might look like this:
   ::RS-STEP-001::
- Parent-Process: RS-LOG-2025083001
- Sequence: 1
- Thought: I need to read the document to understand its contents before I can summarize it.
- Action: `ReadFile('path/to/document.txt')`
- Observation: "File content successfully read. 2,450 words."
- Status: Completed

 * Step 2: The agent reviews the log, sees the file is read, and formulates its next step, logging it again.
   ::RS-STEP-002::
- Parent-Process: RS-LOG-2025083001
- Sequence: 2
- Thought: The document is about the TDM framework. I should use a critical analysis framework to identify its key arguments.
- Action: `::FX-CRITICAL-ANALYSIS-DEEP::`
- Observation: "The analysis identified three core arguments: [argument 1...]"
- Status: Completed

* This process continues until the final summary is produced. The user is left with not only the answer but a complete, auditable transcript of the agent's thought process.

5. Conclusion

The ::RS- protocol provides a powerful, TDM-native solution for creating explicit and auditable "chains of thought." It enhances agent transparency and control by leveraging the existing Metrica-like structure to log cognitive processes. This moves the TDM framework closer to its goal of enabling truly agentic, yet fully understandable, AI systems.