









Pagina 5: Derivazione Stringhe MIU
Responsabilità: Derivazione Nuove Stringhe MIU da Esistenti

Input: Stringhe MIU esistenti nel database

Riferimento da Pagina 1:

[Schema Generale - Pagina 1]
Blocco: Motore Semantico

Interazione Principale per l'Input: DataManager (Pagina 3)

- La logica di derivazione legge le stringhe MIU esistenti dalla tabella `MIU_Searches` (e altre tabelle pertinenti come statistiche di regole/transizioni) tramite `MIUDatabaseManager` in `EvolutiveSystem.SQL.Core.csproj`.
- Questo fornisce i "semi" su cui si basano gli algoritmi BFS/DFS per derivare nuove stringhe.

Flusso Input:

Database (SQLite) --- (tramite EvolutiveSystem.SQL.Core) ---> DataManager --- (tramite IMIUDatabaseManager / IMIURepository) ---> Motore di Derivazione MIU

Persistenza degli Esiti (del Motore Semantico in generale):

- Il `SemanticProcessorService` (il "cervello centrale" di Pagina 2) orchestra l'elaborazione e persiste i suoi risultati (come regole, statistiche, anomalie, stati MIU) nel database SQLite utilizzando il `MIUDatabaseManager` in `EvolutiveSystem.SQL.Core.csproj`.
- Questa operazione non è diretta derivazione, ma persistenza dell'elaborazione generale del "Motore Semantico".

Output per il Prossimo Strato:

Fornisce le Stringhe MIU derivate e arricchite (risultato di BFS/DFS) al blocco "Analisi Topologica + Tassonomia (analisi non semantica)" (come definito in Pagina 1 e dettagliato in EvolutiveSystem.Taxonomy.cs

Componenti e Flussi Interni

[Input/Recupero Stringhe MIU Esistenti]

(Recupera le stringhe MIU dal database tramite DataManager)

- Query su tabella MIU_Searches
- Caricamento in memoria delle stringhe per l'elaborazione proj).

Stringhe MIU Caricate

[Motore di Derivazione MIU]

(Classe: RegoleMIU.cs in MIU.Core.csproj)

(Applica logiche combinatoriali per derivare nuove stringhe MIU)

- Algoritmi di ricerca: Depth-First Search (DFS)
- Algoritmi di ricerca: Breadth-First Search (BFS) (versione "Intelligent" con coda a priorità e euristiche)
- Logica di selezione algoritmo (TrovaDerivazioneAutomatica) basata su euristiche (es. lunghezza stringhe).
- Utilizzo di CurrentTransitionStatistics e CurrentRuleStatistics per guidare l'esplorazione.
- Potatura (pruning) tramite MAX_STRING_LENGTH e penalità lunghezza stringa.
- Tracciamento nodi visitati (HashSet<string> visitedStandard).
- Interazioni chiave: UpsertMIUState (persistenza stati), pubblicazione eventi (OnSolutionFound, OnRuleApplied, OnNewMiuStringDiscoveredInternal).

Nuove Stringhe MIU Derivate e Arricchite

[Output / Interfaccia per Elaborazione Successiva]

Prepara le stringhe MIU derivate per l'Analisi Topologica)

- Potenziale persistenza delle stringhe MIU derivate (tramite MIU.Core che usa IMIUDatabaseManager).
- Interfaccia per il blocco "Analisi Topologica + Tassonomia"