**CoDeR Model**

A Minimal Algebra and Data Model for Continuous Deductive Reasoning
Why CODER?

• Motivations
  • Short Term
    • Reduce streamed data traffic
  • Long Term
    • Support research into the expression of stream reasoning

• Aim
  • Represent continuous reasoning over streamed RDF as a first class problem
Existing Solutions

- Traditional reasoners
  - Window maintenance is as complex as reasoning
  - Entailments are Instantaneous, not Continuous

- Stream queries
  - Semantics of windows do not survive iterated application
Continuous Datalog

- Semantics
  - Theorem proving
  - Sliding window
- Continuous Datalog Programs: \( CDP = \{A, S, R\} \)
Continuous Datalog

- Semantics
  - Theorem proving
  - Sliding window

- Continuous Datalog Programs: \( CDP = \{A,S,R\} \)

- Instantaneous truth:
  - \( CDP_t = W^{S,R}_t \cup A \)
Continuous Datalog

- Semantics
  - Theorem proving
  - Sliding window

- Continuous Datalog Programs: $CDP = \{A, S, R\}$

- Instantaneous truth:
  - $CDP_t = W^{S,R}_t \cup A$

- Continuous entailments:
  - $CDP \models \{A, P, N\}$
  - $CDP_t \models (W^{P,R}_t \setminus W^{N,R}_t) \cup A$
CODER Model

- Expresses the semantics of Continuous Datalog
  - Streamed Data Model
  - Minimal Functional Algebra
CODER: Streamed Data Model

- Input/Output Streams
  - Sequences of RDF triples
    - May be Annotated with Entailment and Negation times: \( \{ T, t_E, t_N \} \)
CO DER: Streamed Data Model

- Input/Output Streams
  - Sequences of RDF triples
    - May be Annotated with Entailment and Negation times: \( \{ T, t_E, t_N \} \)

- Inter-Operator Streams
  - Each composed of matches to a single graph pattern
    - Valid-Time Annotated RDF Graphs: \( \{ G, t_E, t_N \} \)
CoDER Minimal Algebra

- Based on the minimal operators of Positive Datalog: \{\Vdash, \bot, \land\}
- Includes triple-stream filter operators: \{\Vdash, \bot, \land, \sigma_T\}
  - Forward Chaining
- Conjunction expressed by Sequential Conjunction: \{\Vdash, \bot, \textbf{seq}, \sigma_T\}
  - Supports Event Processing/Reasoning
    \[
    \{ A \land B \models C \} \equiv \{ (A \text{ seq } B) \lor (B \text{ seq } A) \models C \} \\
    \equiv \{ A \text{ seq } B \models C, B \text{ seq } A \models C \}
    \]
- RIF-Core built-in functions: \{\Vdash, \bot, \textbf{seq}, \sigma_T, \sigma_B\}
Thank You for Listening

Key Points

- Continuous Datalog: Streamed Entailment Semantics
- CODER Data Model: Duality of Stream Semantics
- CODER Algebra: Minimal Operators for Event Reasoning