ITEM #221 - **Full Application Stack of Time-Series IR Systems: From Offline Structural Discovery to Online Metric-Space Decision Loops**

Conversation：Time-Series IR Requirements

20260109

Authors: Sizhe Tan & GPT-Obot

DBM-COT ITEM #221

Full Application Stack of Time-Series IR Systems

From Offline Structural Discovery to Online Metric-Space Decision Loops

---

Abstract

This ITEM formalizes the complete **10-step application stack** for DBM-based Time-Series IR systems.
Unlike traditional time-series pipelines centered on prediction or regression, this stack treats time-series as **structural evidence streams**, transformed into IR, organized in metric space, indexed structurally, and continuously evaluated through an online evidence loop.

The stack explicitly separates **Offline structural intelligence formation** and **Online real-time recognition and decision**, while enforcing a strict IR contract to guarantee metric consistency, interpretability, and evolution capability.

Overview of the 10-Step Stack

The system is divided into two tightly coupled phases:

- **Offline Phase (Steps 1–5):**
  Structural discovery, pattern formation, and indexing.
- **Online Phase (Steps 6–10):**
  Real-time recognition, decision emission, and evidence-driven evolution.

Offline Phase — Structural Intelligence Formation

Step 1. Offline Input Time-Series Streams

Inputs include:

1. Historical time-series (e.g., stock price curves)
2. Known signal patterns (human-designed, legacy systems, or prior models)

Characteristics:

- High noise
- Non-stationarity
- Multi-scale and shape-variant
- Context-dependent effectiveness

The purpose of this step is **data acquisition**, not signal detection.

Step 2. Time-Series → IR Generation (Offline)

All offline time-series streams are transformed into **Time-Series IR** using DBM-designed IR generation algorithms.

Key principles:

- Structural events instead of raw numeric points
- Direction, delta, ladder/tier, and event causality
- Pattern constraints over threshold rules
- Full interpretability and metric compatibility

Outcome:
A structurally normalized representation suitable for metric-space reasoning.

Step 3. KnownPatterns / CCC Generation and Statistical Evaluation

Two parallel tasks are performed:

1. **Discovery of new KnownPatterns / CCCs** from IR streams
2. **Re-evaluation of existing KnownPatterns** using the same IR logic

Statistical evidence includes:

- Hit frequency
- RHS (Right-Hand-Side) behavior distributions

- Conditional effectiveness
- Failure regions

Patterns are treated as **conditional structures**, not absolute signals.

Step 4. User-Specific Structural Utilization

Users (traders, analysts, or higher-level systems) pull **structures**, not raw signals.

Users may:

- Recombine patterns
- Impose additional constraints
- Perform secondary structural analysis
- Explore relational or contextual dependencies

This step explicitly separates **structural intelligence** from **decision preference**.

Step 5. Structural Indexing with Pattern Family Augmentation

Before indexing, each Known Signal Pattern is expanded into a **Pattern Family**:

- A bounded set of **property-preserving IR variants**
- Variations in speed, amplitude, phase, and noise
- Controlled by mutation policies and cost budgets

Structural indices are then built over pattern families using:

- Metric Differential Trees
- Two-Phases Search Engines

This converts pattern matching into **neighborhood search in metric space**, improving recall without sacrificing correctness.

Online Phase — Real-Time Recognition and Evidence Loop

Step 6. Real-Time Input Time-Series Streams

Live time-series streams are ingested in real time.

Properties:

- Partial and noisy

- Asynchronous
- Latency-tolerant

No assumption is made that online data is "clean" or complete.

Step 7. Time-Series → IR Generation (Online)

Real-time streams are converted into IR using **the same IR generation logic** as offline processing.

This enforces the **IR contract**:

Metric distance is only meaningful if IR generation is identical across time.

Step 8. Metric Distance Computation

The real-time IR is compared against KnownPatterns / CCC pattern families using:

1. Differential Tree localization, or
2. Two-Phases Search (coarse prune → fine match)

The goal is not exact matching, but **structural proximity estimation** in metric space.

Step 9. Thresholding and Signal Emission

Decisions are made based on:

- Pattern-family-level distance distributions
- Structural consistency across variants
- Contextual and RHS-aware thresholds

Signals may be emitted to:

- Traders
- Automated decision engines
- Situation-awareness dashboards (war rooms)

This step outputs **evidence-backed structural alerts**, not raw predictions.

Step 10. Post-Event Update and Evolution

After signals are emitted and outcomes observed:

- Pattern scores are updated
- RHS statistics are refined
- Pattern family confidence is adjusted
- New variants may be promoted to offline discovery

This closes the **structural evidence loop**, enabling continuous evolution.

Core Design Principles

1. **IR is the contract** between Offline and Online
2. **Patterns are structural families, not templates**
3. **Metric distance carries semantic meaning**
4. **Indexing precedes real-time decision**
5. **Evidence accumulation dominates one-shot prediction**

Summary Statement

This 10-step stack transforms time-series intelligence from numeric prediction pipelines into a living structural reasoning system, where signals are discovered, indexed, recognized, and evolved within a unified metric-space framework.

DBM-COT ITEM #221（中文版）

时间序列 IR 系统的完整应用**栈**

从离线结构发现到在线度量空间决策闭**环**

摘要

本 ITEM 正式固化 DBM 时间序列 IR 系统的 **完整 10 步应用流程**。

该体系不同于以预测或回归为中心的传统时间序列方法，而是将时间序列视为**结构证据流**，通过 IR 表达、度量空间组织、结构索引与在线证据闭环，实现可解释、可演化的智能系统。

流程明确区分：

- **离线结构智能形成（Steps 1–5）**
- **在线实时识别与决策（Steps 6–10）**

并通过统一的 IR 合约保证系统一致性。

十步流程总览

- **Offline（1–5）：结构发现与索引**
- **Online（6–10）：实时识别与演化**

Offline 阶段 —— 结构智能形成

Step 1. 离线时间序列输入

输入包括：

1. 历史时间序列（如股票曲线）
2. 已知信号模式（人工经验或既有系统）

目标是**获取结构证据来源**，而非直接产出信号。

Step 2. 时间序列 → IR（离线）

所有时间序列被转换为 **Time-Series IR**：

- 事件化
- 方向与梯次结构
- 因果与约束表达
- 完全可解释、可度量

Step 3. KnownPatterns / CCC 生成与统计

并行执行：

- 新结构模式发现
- 已有模式的统一 IR 重评估

模式被视为**条件结构**而非绝对信号。

Step 4. 用户侧结构使用

用户拉取的是**结构本身**，而不是固定买卖点。

用户可进行：

- 结构组合
- 约束强化
- 二次分析

Step 5. 结构索引 + 模式族扩增

每个模式先生成**保持结构性质的模式族**，再进行索引构建：

- 差分树
- 两阶段搜索

实现稳健的度量空间邻域搜索。

Online 阶段 —— 实时识别与证据闭环

Step 6. 实时数据输入

接收实时、不完整、含噪的时间序列流。

Step 7. 时间序列 → IR（在线）

在线 IR 生成逻辑必须与离线完全一致。

Step 8. 度量距离计算

在结构索引中计算实时 IR 与模式族的结构距离。

Step 9. 阈值判断与信号发射

基于模式族一致性与 RHS 上下文作出决策，并输出结构化信号。

Step 10. 事后更新与演化

根据结果更新：

- 模式评分
- RHS 统计
- 模式族结构

形成持续演化的证据闭环。

核心设计原则

1. IR 是 Offline/Online 的唯一契约

2. 模式是结构族，不是模板
3. 距离具有语义
4. 索引优先于决策
5. 证据累积优先于一次预测

总结陈述

该 10 步体系将时间序列分析从数值预测升级为结构化度量空间推理系统，是 DBM 时间序列智能得以工程落地的核心架构。
