KgSyncController

September 4, 2026

KgSyncController

Kind: Controller

Source: `atloria-monorepo/apps/api/src/graph/kg-sync.controller.ts`

Knowledge Graph Sync Controller

Handles CLI dynamic mode sync operations.
Provides endpoints for bulk entity and relationship sync.

`KgSyncController` exposes backend endpoints used by the CLI dynamic mode to synchronize knowledge graph data. It coordinates bulk upserts of entities and relationships, delegating validation and persistence to the underlying knowledge graph sync services.

Diagram

```
graph LR
  CLI[CLI Dynamic Mode] -->|Bulk entity payload| Controller[KgSyncController]
  CLI -->|Bulk relationship payload| Controller
  Controller -->|Validate and dispatch| SyncService[Knowledge Graph Sync Service]
  SyncService -->|Upsert entities| EntityStore[Entity Storage]
  SyncService -->|Upsert relationships| RelationshipStore[Relationship Storage]
```

Usage

```
const apiBaseUrl = process.env.API_URL ?? "http://localhost:3000";

async function syncKnowledgeGraph() {
  await fetch(`${apiBaseUrl}/kg-sync/entities`, {
    method: "POST",
    headers: { "Content-Type": "application/json" },
    body: JSON.stringify({
      entities: [
        {
```

```
          id: "customer-123",
          type: "Customer",
          properties: {
            name: "Acme Corp",
            industry: "Technology",
          },
        },
      ],
    }),
  });

  await fetch(`${apiBaseUrl}/kg-sync/relationships`, {
    method: "POST",
    headers: { "Content-Type": "application/json" },
    body: JSON.stringify({
      relationships: [
        {
          sourceId: "customer-123",
          targetId: "project-456",
          type: "OWNS",
          properties: {
            since: "2024-01-01",
          },
        },
      ],
    }),
  });
}

void syncKnowledgeGraph();
```

AI Coding Instructions

- Keep controller handlers thin: parse request DTOs, invoke the sync service, and return consistent HTTP responses.
- Preserve bulk-sync semantics; avoid per-record database operations in the controller when a service-level batch operation is available.
- Validate entity identifiers, relationship source/target identifiers, types, and required payload structure before processing.
- Ensure entity sync occurs before relationship sync when relationships reference newly created entities.
- Maintain compatibility with CLI dynamic mode payload formats; coordinate DTO or route changes with the CLI integration.

Relationships

- MODULE_DECLARES → `getEntities`
- MODULE_DECLARES → `getEntityById`
- MODULE_DECLARES → `syncEntities`
- MODULE_DECLARES → `getRelationships`
- MODULE_DECLARES → `getRelationshipsForEntity`
- MODULE_DECLARES → `syncRelationships`
- DEPENDS_ON → `KgSyncService`
- DEPENDS_ON → `WorkflowDiscoveryService`
- DEPENDS_ON → `WorkflowStorageService`

Referenced By

- `GraphModule` (MODULE_DECLARES)