

L10nAutomationService

September 4, 2026

L10nAutomationService

Kind: Service

Source: `atloria-monorepo/apps/api/src/l10n/l10n-automation.service.ts`

B6 orchestration on top of the translate queue:

- `bulk-translate` — enqueue one job per page of a target language (missing / stale / all).
- `auto-retranslate` — the S3/S4 delta-completion hook: machine-translated

PUBLISHED variants

whose source moved under them are re-queued. DOUBLE-gated: the

`L10N_AUTO_RETRANSLATE`

kill-switch env AND the project's own `autoRetranslate` opt-in must both be on. Called fire-and-forget from the delta pipelines — it must never fail or slow a delta.

`L10nAutomationService` orchestrates translation automation on top of the translate queue. It supports bulk translation by enqueueing one job per page for a target language, and automatically re-enqueues machine-translated published variants when their source content changes. Auto-retranslation is protected by both the `L10N_AUTO_RETRANSLATE` environment kill switch and the project's `autoRetranslate` opt-in.

Methods

Method	Signature	Returns	Description
<code>bulkTranslate</code>	<code>`bulkTranslate(projectId: string, language: string, mode: 'missing'</code>	<code>'stale'</code>	<code>'all', publish: unknown)`</code>
<code>onDeltaCompleted</code>	<code>onDeltaCompleted(projectId: string)</code>	<code>Promise<{ enqueued: number }></code>	Delta-completion hook (fire-and-forget at every call site).

Dependencies

- TranslationsService
- TranslateQueueService

Where it refuses work

- L10nAutomationService stops the work with an early return when `mode === 'all'` .
- L10nAutomationService stops the work with an early return when `mode === 'stale'` .
- L10nAutomationService stops the work with an early return when `process.env.L10N_AUTO_RETRANSLATE !== 'true'` .
- L10nAutomationService stops the work with an early return when `!settings.autoRetranslate` .

When something fails

- L10nAutomationService handles failure in 1 place: it turns it into a return value in all 1.

Diagram

```
sequenceDiagram
    participant Caller
    participant Automation as L10nAutomationService
    participant Project
    participant Queue as Translate Queue
    participant Delta as Delta Pipeline

    Caller->>Automation: bulkTranslate(targetLanguage, mode)
    Automation->>Project: Find pages matching missing/stale/all mode
    loop For each matching page
        Automation->>Queue: Enqueue translation job
    end
    Automation-->>Caller: { enqueued, total }

    Delta->>Automation: onDeltaCompleted(delta)
    Automation->>Automation: Check L10N_AUTO_RETRANSLATE
    Automation->>Project: Check project.autoRetranslate
    alt Both gates enabled
        Automation->>Project: Find stale machine-translated PUBLISHED variants
        loop For each affected variant
            Automation->>Queue: Re-enqueue translation job
        end
        Automation-->>Delta: { enqueued }
    else Either gate disabled
```

```
Automation-->>Delta: { enqueued: 0 }  
end
```

Usage

```
import { L10nAutomationService } from './l10n-automation.service';  
  
// Typically injected by NestJS into a controller, resolver, or workflow service.  
export class TranslationAdminService {  
  constructor(  
    private readonly l10nAutomationService: L10nAutomationService,  
  ) {}  
  
  async translateFrenchPages(projectId: string) {  
    const result = await this.l10nAutomationService.bulkTranslate({  
      projectId,  
      targetLanguage: 'fr',  
      mode: 'missing',  
    });  
  
    return {  
      message: `Queued ${result.enqueued} of ${result.total} pages for translation.`,  
      ...result,  
    };  
  }  
  
  async handleSourceDeltaCompleted(deltaId: string) {  
    // Delta callers should invoke this fire-and-forget so translation  
    // automation cannot delay or fail the primary delta pipeline.  
    void this.l10nAutomationService  
      .onDeltaCompleted({ deltaId })  
      .catch(() => undefined);  
  }  
}
```

AI Coding Instructions

- Keep `bulkTranslate()` focused on selecting eligible pages and enqueueing exactly one translation job per page; return both the successfully enqueued count and total candidates.
- Treat `onDeltaCompleted()` as best-effort automation: callers in delta pipelines must invoke it fire-and-forget and must not allow its failures to block delta completion.
- Preserve the double gate for automatic retranslation: require both `L10N_AUTO_RETRANSLATE` and the project's `autoRetranslate` setting before enqueueing jobs.

- Only requeue eligible machine-translated `PUBLISHED` variants whose source content has moved or become stale; do not overwrite manual translations through this path.
- Integrate through the existing translate queue rather than performing translation work inline in this service.

Relationships

- `DEPENDS_ON` → `TranslationsService`
- `DEPENDS_ON` → `TranslateQueueService`

Referenced By

- `DocAutomationService` (`DEPENDS_ON`)
- `L10nController` (`DEPENDS_ON`)
- `L10nModule` (`MODULE_PROVIDES`)
- `L10nModule` (`MODULE_EXPORTS`)
- `TechnicalDocsGenerationService` (`DEPENDS_ON`)