ISWC IPI Context Search – Quick Guide

1   Introduction

The ISWC IPI Context Search allows publishers to identify an interested party (including their IPI Name Number) by searching for their name and providing work titles for additional context, as is currently possible via CIS-Net RHA. Simplified work metadata can also be provided in the results, with the user having the ability to specify the level of detail returned.

Before accessing these services in production, users will first undergo a test cycle to ensure the required specifications are being met.

This document will help to guide users through the initial test cycle for the IPI Context Search, before being onboarded to the live service after the test cycle has been completed successfully.

NOTE: ISWCs returned during the test cycle are not genuine ISWCs and should not be disseminated

2   Documentation and Sample Files

Documentation for the ISWC IPI Context Search is available to users on the ISWC Developer Portal, through the following link:
https://cisaciswcuat.developer.azure-api.net/publisher-ipi-context-search

3   Onboarding Overview

To commence UAT for the ISWC IPI Context Search, Publishers should register their interest in the service by completing the form available on the ISWC web page, and they will then be invited to sign the required Terms and Conditions with CISAC.

Once the Terms and Conditions are signed, users will undergo the following steps:

1. Sign up to the developer portal in UAT: https://cisaciswcuat.developer.azure-api.net/
2. Subscribe to the ISWC IPI Context Search API through the 'Products' page on the portal.
3. CISAC approves the subscription, at which point the user can begin testing. Documentation is available on the portal under the link to the API.
4. User submits test API data.
5. User verifies that their software can ingest the API responses.

Once the publisher has successfully performed a test cycle, a recommendation will be made to proceed to the Production environment. The user can then sign-up to the developer portal in Production and subscribe to the IPI Context Search API.

4   Onboarding Checklist

UAT

Publisher has signed T&Cs and subscribed to the API via the developer portal (UAT)	☐
CISAC has approved the subscription	☐
CISAC has informed the publisher that ISWCs returned during UAT are not genuine ISWCs and should not be disseminated	☐
Publisher has made test submissions and is receiving results	☐

<u>Go Live</u>

Recommendation has been made for the publisher to go live	☐
Publisher has been advised to sign up to the developer portal (Production) and subscribe to the API	☐
Subscription has been approved	☐
Publisher has confirmed that their subscription is working, and their submissions are generating valid results / errors.	☐

5   API Performance & Throughput Guidance

5.1 Rate Limiting

The ISWC IPI Context Search API does not enforce a fixed rate limit.

There is also no hard limit on the number of items per batch request for the Publisher API.

However, for optimal system performance and stability, we provide the following recommendations based on internal testing.

5.2 Recommended Batch Size

For best performance, batch requests should contain:

    10 items per request

While larger batch sizes are technically supported, performance testing has shown diminishing returns as batch size increases beyond 10 items.

Publishers are encouraged to:

- Configure their systems to send batches of 10 items
- Adjust parallel processing levels to achieve desired throughput rather than increasing batch size

5.3 Parallelism & Throughput Recommendations

Testing indicates that throughput improves significantly when using parallel requests with moderate batch sizes, rather than fewer requests containing large batches.

For example:

- 2–4 parallel threads
- Each requesting batches of 10 items

5.4 Implementation Best Practice

To achieve optimal performance:

- Set batch size to 10
- Use 2–3 concurrent threads as a starting point
- Increase parallelism gradually if higher throughput is required
- Avoid very large batch payloads, as these may:

    - Increase response time
    - Increase timeout risk
    - Reduce overall system efficiency

Each publisher should validate optimal settings within their own infrastructure, as performance may vary depending on network configuration and system architecture.