REST, GraphQL and JSON-LD

Digital Product Passport Documentation

Guide for Magento administrators, technical project owners, and implementation teams.

LAST UPDATED
2026-10-13

SOURCE
Open this article online

EXTENSION
Magento 2 Extension Digital Product Passport

REST, GraphQL and JSON-LD

The public APIs expose only passports with status `published`. They are anonymous endpoints, so treat every published payload field as public information.

While `dpp/general/enabled` is off, the API surface is closed alongside the storefront: the by-SKU endpoints and the GraphQL query return null, and the listing endpoint returns an empty result. There is no distinct "module disabled" error — integrations should treat the response the same as "no published passport".

REST by SKU

```
GET /rest/V1/dpp/passport/{sku}
GET /rest/V1/dpp/passport/{sku}/jsonld
```

The first endpoint returns the passport service object. The second returns a JSON string containing formatted JSON-LD. An optional `storeId` query value selects store context.

If the SKU does not exist or no published record matches, the service returns null.

REST listing

```
GET
/rest/V1/dpp/passports?searchCriteria[pageSize]=20&searchCriteri
a[currentPage]=1
```

Magento search criteria can filter, sort and paginate the result. The service always adds a Published filter, even if the caller requests another status.

GraphQL

```
query PublishedPassport {
  dppPassportBySku(sku: "NORTHSTAR-TEXTILE-410", storeId: 1) {
    passport_id
    template_code
    version
    espr_category
    country_of_origin
    dpp_payload
    completeness_score
  }
}
```

The query returns null when no published passport matches. The payload field is a JSON

string rather than a nested GraphQL object.

JSON-LD content

Formatted JSON-LD contains:

- a schema-oriented context plus template-specific vocabulary entries;
- a public identifier URL;
- SKU, GTIN, manufacturer, batch and serial identifiers when available;
- store issuer information;
- status, version and publication dates;
- category, origin and HS code when available;
- the decoded passport payload;
- public page and QR links.

Integration checklist

- URL-encode SKUs used in REST paths.
- Pass the intended store ID in multi-store integrations.
- Handle null and 404 outcomes without retry loops.
- Treat `dpp_payload` as untrusted external JSON in consumers.
- Cache published data only as long as the integration can tolerate stale passport updates.
- Test that Draft, Active and Archived records remain excluded.