Reading a node

Read one item for a detail screen through the REST API - by node id or object id, the metadata keys, the links map, locations and fields on request, and how to follow links without building URLs by hand.

Source file: doc/eng-us/help/mobile-app-developers/rest-api/reading-a-node.md Last changed 2026-11-04
https://doc.exponential.earth/learninminutes/exphelp/viewer/help/mobile-app-developers/rest-api/reading-a-node.md

This page is for app developers who build a detail screen: one article, one product, one event. It shows the read routes for a single item, what the answer contains and how to ask for more.

Contents

1. By node or by object
2. The metadata
3. Links to the fields
4. Locations
5. A detail screen in one call

By node or by object

Shell

```
curl -s -H "Authorization: Bearer $EXP_API_KEY" https://example.com/api/ezp/v2/content/node/<node
 id>
curl -s -H "Authorization: Bearer $EXP_API_KEY" https://example.com/api/ezp/v2/content/object/<ob
ject id>
```

Use the node route when you came from a list (lists give `nodeId`) or a link of the site. Use the object route when you stored an object id, for example from a relation. Both run the same controller action, `viewContent`, and both check `content/read` for the current user first: `403 access_denied` if the user may not read it, `404` if it does not exist.

The metadata

The `Metadata` response group is always included. The keys, from `ezpRestContentModel`:

Key	Type	Meaning
`objectName`	string	The name of the object in the language read
`classIdentifier`	string	Its content class, for example `article`
`datePublished`	integer	First publication, Unix timestamp
`dateModified`	integer	Last change, Unix timestamp
`objectRemoteId`	string	The remote id of the object
`objectId`	integer	The object id

Key	Type	Meaning
`nodeId`	integer	Node reads only: the node id
`nodeRemoteId`	string	Node reads only: the remote id of the node
`fullUrl`	string	Node reads only: the full address of the node's page on the site

Use `dateModified` to decide whether a cached copy in the app is still current (Caching in the app), and `fullUrl` for a "share" or "open in browser" button.

Links to the fields

Every read has a `links` map: one entry per field identifier with the URL of that field, and `*` with the URL of all fields. The URLs repeat the `Translation` and `OutputFormat` parameters of the request. Follow them instead of building paths in the app; it keeps the app working if a path ever changes.

Locations

An object can sit in several places of the tree. Ask for them with `?ResponseGroups=Locations`:

JSON

```
"locations": [
  {"nodeId": 61, "nodeRemoteId": "...", "fullUrl": "https://example.com/news/launch", "isMain": t
rue},
  {"nodeId": 142, "nodeRemoteId": "...", "fullUrl": "https://example.com/products/launch", "isMai
n": false}
]
```

(The ids and addresses are invented for the example; the keys are those of `getLocationsByContent()`.) Use the main location for shared links, so every user shares the same address.

A detail screen in one call

Shell

```
curl -s -H "Authorization: Bearer $EXP_API_KEY" \
     "https://example.com/api/ezp/v2/content/node/<node id>?ResponseGroups=Metadata,Fields"
```

This gives the metadata and every field in one answer, which is usually all a detail screen needs. Add `Locations` only if the screen uses them. The field values are described in Fields and field values; images and rich text need extra care (Images in REST answers, Rich text in apps).

Related pages

- Response groups
- Listing children and pagination
- How Exponential content reaches an app
- Free documentation: this page is free to copy, translate and improve under the GNU FDL 1.3.

Become an Exponential Partner

Partners share their information, provide funding, contribute back and develop the products with us, and that is how Exponential grows. A partner website and dashboard are planned for mid-2027: please become a partner today to speed up our growth rate and land new features faster.

Partners: start here | How to apply

Exponential Copyright © 1998-2026 Exponential Foundation and 7x
Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".
This documentation is free documentation, licensed under the GNU Free Documentation License 1.3 or later:
https://www.gnu.org/licenses/fdl-1.3.html.