Australian Government
Department of Infrastructure, Transport, Regional Development, Communications and the Arts

Harmonised National Roadworks and Road Closures dataset API User Guide

National Freight Data Hub

December 2024

1. Purpose

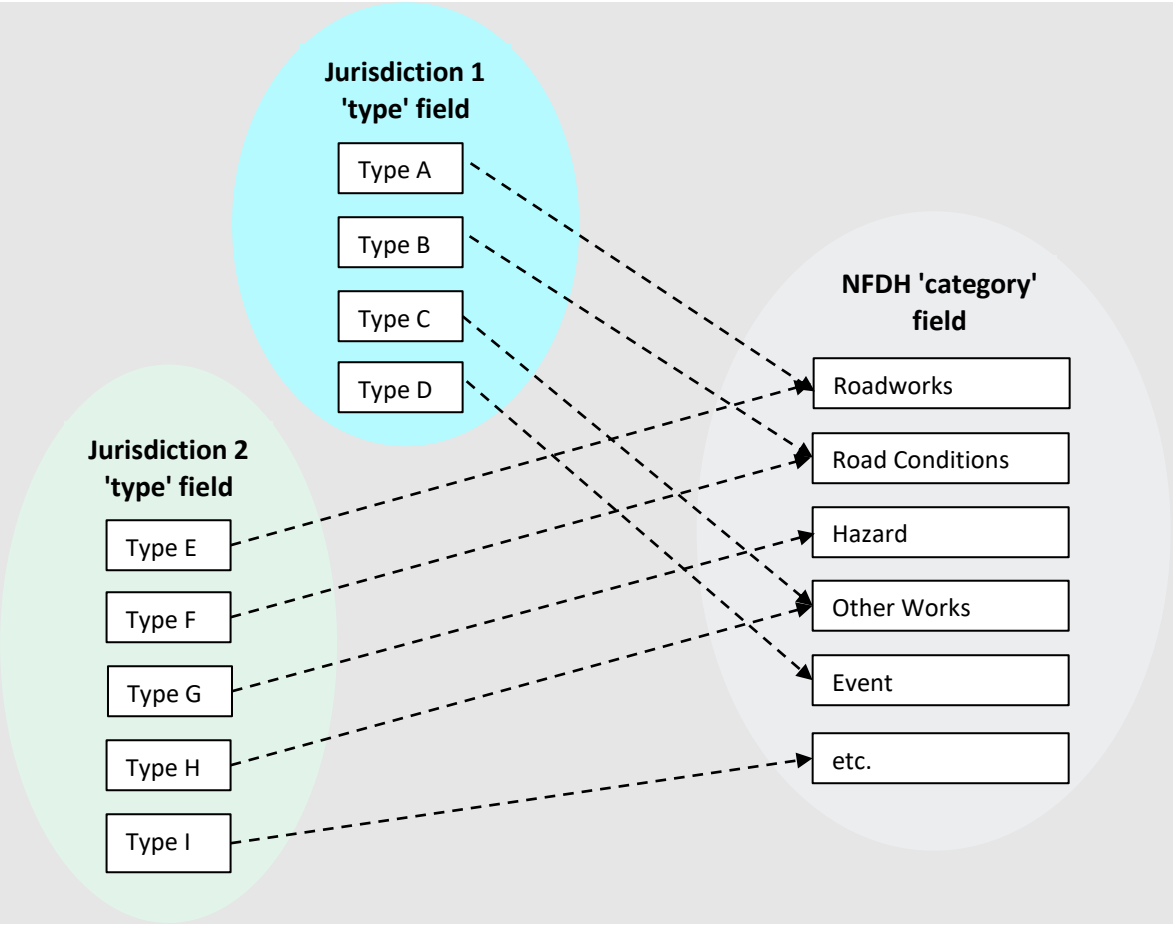
This document is a guide for users of the National Freight Data Hub's (NFDH) Harmonised National Roadworks and Road Closures dataset API. After reading this guide, users should be able to interact with the data and construct queries to return samples of the data to serve their particular use cases.

2. Introduction

The Harmonised National Roadworks and Road Closures dataset uses daily updates from state/territory roadworks systems to compile a set of historical roadworks for all of Australia. The geospatial coverage of the dataset is all state-managed roads in Australia. Most jurisdictions include planned and unplanned works and closures, however data for Tasmania currently includes planned roadworks only. The data commences from 2016 - some earlier records are present but the dates do not appear to be accurate.

Each state/territory uses a different set of roadwork 'types' to classify the works. The harmonisation process groups like types into a set of categories which are consistent across all jurisdictions, for reporting and visualisation purposes. Where the data indicates the road is closed (via text description), a category of Road Closure is applied, irrespective of the state/territory 'type'.

Figure 1: Example diagram of the NFDH process to harmonise the various reported roadwork types from the jurisdictions



Data sourced from states and territories provide different levels of detail about the roadworks. Therefore, not all columns will be populated for all records.

Data is refreshed on a nightly basis.

The Harmonised National Roadworks and Road Closures data is also visualised in the Roadworks and Road Closures interactive on the NFDH website, and can be downloaded in csv format or as a geoJSON file from the DITRDCA Data Catalogue.

2.1 Input datasets

Data is sourced from state and territory open data roadworks APIs. Links to each of these datasets are available through the DITRDCA Data Catalogue: Roadworks endpoints (state/territory datasets)

3. Dataset structure

The API calls the NFDH's Harmonised National Roadworks and Road Closures dataset. The table below describes the structure of this harmonised dataset. Users can consult this table as a reference document to better understand the data returned by the API.

Field name	Field description
id	Unique identifier of the record in NFDH's ArcGIS database.
unique_identifier	Unique identifier provided by the source agency.
status	Status of the roadwork (i.e. Open, Active, Closed). This field is included for the user's information only and is not harmonised. It is not consistently applied between jurisdictions.
category	The category of road event the record relates to, determined by the NFDH. The categories in the dataset are as follows: • Event • Hazard • Other • Other Works • Road Closure • Road Conditions • Roadworks Any newly introduced 'types' (see below) which have not yet been mapped to one of the above categories sit in the 'Unknown' category. When identified, these new types will be mapped to one of the above categories by the NFDH team at earliest convenience. • Unknown

Field name	Field description
updated_category	An updated field from the above simple "category" and includes more specific terms, determined by the NFDH. The categories in the dataset are as follows: • Planned Roadworks • Unplanned Roadworks • Flood Hazard • Crash Hazard • Fire Hazard • Other Hazard • Planned Event • Utilities / Construction Any newly introduced 'types' (see below) which have not yet been mapped to one of the above categories sit in the 'Unknown' category. When identified, these new types will be mapped to one of the above categories by the NFDH team at earliest convenience. • Unknown
type	Classification of the type of roadwork, provided by the source agency. Each of the types provided by the states/territories are mapped to a 'category' to standardise the data. This field is included for the user's information only and is not harmonised. It is not consistently applied between jurisdictions.
from_date	Date and time the roadwork starts, in UTC format (milliseconds).
to_date	Date and time the roadwork ends, in UTC format (milliseconds).
planned_start_date	Date and time the roadwork is planned to start – this field is unpopulated for all records in the dataset as the source APIs do not include this information.
planned_end_date	Date and time the roadwork is planned to end – this field is unpopulated as the source APIs do not include this information for all records in the dataset.
modified_date	The date the record was last modified/updated in the source data, in UTC format (milliseconds).
description	Textual description of the roadwork.

Field name	Field description
street_name	Name of the road on which the work or closure is occurring.
side_street	Human readable description of intersecting street nearest the start of the roadworks (could also be a motorway/freeway ramp) .
end_side_street	Human readable description of intersecting street nearest the end of the roadworks (could also be a motorway/freeway ramp) – this field is unpopulated for all records in the dataset as the source APIs do not include this information.
direction	Direction/s of travel affected – this field is unpopulated for all records in the dataset as the source APIs do not include this information.
state	The state/territory which has provided the record via their source API. Note that this does not always align with the state/territory managing the road, or the state/territory in which the road is located.
capture_date	The date the record was first captured in the NFDH dataset, in UTC format (milliseconds).
point	Geospatial location of roadwork – this field returns X and Y coordinates corresponding to longitude and latitude respectively.
hv_access	True (1) or False (0) value to indicate whether the road remains usable for heavy vehicles e.g. semitrailers. Derived field based on Road Closure status.
lv_access	True (1) or False (0) value to indicate whether the road remains usable for light vehicles e.g. passenger vehicles. Derived field based on Road Closure status.
source_url	The state/territory source API where the record originated.

4. Accessing and querying the dataset

The Harmonised National Roadworks and Road Closures dataset can be accessed and queried through the Department's ArcGIS web service using the following URL: https://spatial.infrastructure.gov.au/server/rest/services/Hosted/RADAR_Curated_Prod_roadworks/FeatureServer/0/query?

The data can be queried either by constructing a URL with specified parameters or through the dialog box interface. See section 5 for more detail.

5. Constructing queries

5.1 Base URL

https://spatial.infrastructure.gov.au/server/rest/services/Hosted/RADAR_Curated_Prod_roadworks/FeatureServer/0/query?

In order to construct a query URL to return only the records that meet our specified criteria, we can append specific parameters to the base URL. However, in order to do so, we need to understand what each potential parameter in the URL represents and what values it can take.

Please note:

- Parameters can be specified in any order
- Parameters are case sensitive
- Field names (in the dataset) are **not** case sensitive – e.g. the following will all give the same output:
  - *Category = 'Roadworks'*
  - *category = 'Roadworks'*
  - *CATEGORY = 'Roadworks'*
- Field values **are** case sensitive – e.g.
  - *category = 'Roadworks'* will give the correct output, but *category = 'ROADWORKS'* will **not**.
  - Fields defined by NFDH have tried to retain consistent capitalisation of field names and values for ease of use, and apply regular checks to avoid duplication of similar field values.

5.2 Parameter definitions and example queries

The following parameters can be used to filter the query responses:

where	spatialRel	maxAllowableOffset	returnDistinctValues	outStatistics	returnTrueCurves
objectIds	distance	geometryPrecision	returnIdsOnly	returnZ	returnCentroid
time	units	outSR	returnCountOnly	returnM	sqlFormat

geometry	relationParam	havingClause	returnExtentOnly	multipatchOption	resultType
geometryType	outFields	gdbVersion	orderByFields	resultOffset	datumTransformation
inSR	returnGeometry	historicMoment	groupByFieldsForStatistics	resultRecordCount	f

Use the above links to view parameter definitions and example queries in the below table:

Parameter	Details	Example query URLs (where <Base URL> = https://spatial.infrastructure.gov.au/server/rest/services/Hosted/RADAR_Curated_Prod_roadworks/FeatureServer/0/query?)

where	A `WHERE` clause for the query filter, can be used on all fields in the dataset. Operations supported:
	`( '<=' | '>=' | '<' | '>' | '=' | '<>' | LIKE )`
	`(AND | OR)`
	Example:
	• `where=category = 'Road Closure' AND state = 'VIC'`
	• `where= from_date >= DATE '2023-01-01'`
	Note that the `Query` will always return date values in UTC. However, if we want to filter on particular date-time fields, we will need to specify them in local time. There are two ways this can be done, by `DATE` and `TIMESTAMP` date functions. Each type of date-time query must include a date-time function to ensure the
		<Base URL>where=state = 'VIC' and to_date < DATE '2022-01-01'&outFields=*&resultOffset=2000
		
		<Base URL>where=state = 'NSW' and category = 'Roadworks'&outFields=*&f=pjson
		
		<Base URL>where=from_date < CURRENT_TIMESTAMP&returnCountOnly=true

query is treated properly. The syntax is as follows:

- `<DateField> = DATE 'YYYY-MM-DD'` (this is equivalent to `TIMESTAMP 'YYYY-MM-DD 00:00:00'` – only exact matches will be returned)
- `<DateField> = TIMESTAMP 'YYYY-MM-DD HH:MI:SS'`

Please note when querying `<DateField> = DATE 'YYYY-MM-DD'` this will not necessarily return all records on the specified day. To specify all records on a day (or for another time period) it is recommended to use `'<'` and `'>'` to specify a range.

`from_date, to_date, modified_date` and `capture_date are` the date-time fields in the dataset. Example queries:

- `where= from_date >= DATE '2023-01-01'`
- `where= capture_date < DATE '2022-07-01'`
- `where= from_date >= DATE '2022-01-01' AND from_date < DATE '2023-01-01'`

You can also use `CURRENT_DATE` and/or `CURRENT_TIMESTAMP` in your queries:

- `where= from_date <= CURRENT_TIMESTAMP and state = 'NSW'`
- `where= to_date > CURRENT_DATE`

	Back to parameter list	
objectIds	The object IDs of this layer or table to be queried. Syntax: `objectIds=<objectId1>, <objectId2>` Example: `objectIds=37, 462` Back to parameter list	<Base URL>outFields=*&f=pjson&objectIds=37, 462
time	The time instant or the time extent to query. This will filter on the `from_date` field in the dataset. **Time instant** Syntax: `time=<timeInstant>` Example: time= 1670665500000 (10 December 2022 9:45:00 AM GMT) This will return all results where `from_date = <timeInstant>` (exact match). **Time extent** Syntax: `time=<startTime>, <endTime>` Example: `time=1670665500000, 1670765500000` (10 December 2022 9:45:00 AM GMT to 11 December 2022 1:31:40 PM GMT) This will return all records where from_date lies within the specified interval.	<Base URL>outFields=*&f=pjson&time=1670665500000, 1670765500000

	A null value specified for start time or end time will represent infinity for start or end time, respectively. Example: `time=null, 1670765500000` Back to parameter list	
geometry	The geometry to apply as the spatial filter. This can be used when users wish to return all records located within an envelope/close to a particular point location etc. and works with the geometryType parameter. The geometry of envelopes and points can be specified with a simple comma-separated syntax. Syntax: • Envelope simple syntax: `geometryType=esriGeometryEnvelope&geometry=<xmin>,<ymin>,<xmax>,<ymax>` • Point simple syntax: `geometryType=esriGeometryPoint&geometry=<x>,<y>` Back to parameter list	<Base URL>geometryType=esriGeometryEnvelope&geometry=148.641921, -35.164522, 148.713332, -35.986594
geometryType	The type of geometry specified by the `geometry` parameter. The geometry type can be an envelope, a point, a line, or a polygon. The default geometry type is an envelope. Values: `esriGeometryPoint | esriGeometryMultipoint |`	<Base URL>geometryType=esriGeometryEnvelope&geometry=148.641921, -35.164522, 148.713332, -35.986594

	`esriGeometryPolyline` | `esriGeometryPolygon` | `esriGeometryEnvelope` Back to parameter list	
inSR	The spatial reference of the input geometry. If the `inSR` is not specified, the geometry is assumed to be the same as the native layer spatial reference (EPSG4326). Back to parameter list	
spatialRel	The spatial relationship to be applied to the input geometry while performing the query. The supported spatial relationships include intersects, contains, envelope intersects, within, and so on. The default spatial relationship is intersects (`esriSpatialRelIntersects`). Values: `esriSpatialRelIntersects` | `esriSpatialRelContains` | `esriSpatialRelCrosses` | `esriSpatialRelEnvelopeIntersects` | `esriSpatialRelIndexIntersects` | `esriSpatialRelOverlaps` | `esriSpatialRelTouches` | `esriSpatialRelWithin` Back to parameter list	<Base URL>geometry=148.641921, -35.164522, 148.713332, -35.986594&geometryType=esriGeometryEnvelope&spatialRel=esriSpatialRelIntersects&distance=100&units=esriSRUnit_Meter
distance	The buffer distance for the input geometries. The distance unit is specified by `units`. For	<Base URL>geometry=148.641921, -35.164522, 148.713332, -35.986594&geometryType=esriGeometryEnvelop

	example, if the distance is 100, the query geometry is a point, `units` is set to `meters`, and all points within 100 meters of the point are returned. The geodesic buffer is created based on the datum of the output spatial reference if it exists. If there is no output spatial reference, the input geometry spatial reference is used. Otherwise, the native layer spatial reference is used to generate the geometry buffer used in the query. This is not recommended as the native spatial reference system (WGS84) uses decimal degrees as the unit (not metres) which may return unexpected results. Back to parameter list	e&spatialRel=esriSpatialRelIntersects&distance=100&units=esriSRUnit_Meter
units	The unit for calculating the buffer distance. If `unit` is not specified, the default will be `esriSRUnit_Foot` Values: `esriSRUnit_Meter|esriSRUnit_StatuteMile|esriSRUnit_Foot|esriSRUnit_Kilometer|esriSRUnit_NauticalMile|esriSRUnit_USNauticalMile` Back to parameter list	<Base URL>geometry=148.641921, -35.164522, 148.713332, -35.986594&geometryType=esriGeometryEnvelope&spatialRel=esriSpatialRelIntersects&distance=100&units=esriSRUnit_Meter
relationParam	The spatial relate function that can be applied while performing the query operation. An example for this spatial relate function is "FFFTTT***". For more information on this	

	spatial relate function, see the documentation for the spatial relate function. Back to parameter list	
outFields	The list of fields to be included in the returned result set. This list is a comma delimited list of field names. You can also specify the wildcard "*" as the value of this parameter. In this case, the query results include all the field values. Back to parameter list	<Base URL>geometryType=esriGeometryEnvelope&geometry=148.641921, -35.164522, 148.713332, -35.986594&outFields=state, category, type
returnGeometry	If `true`, the result includes the geometry associated with each feature returned. The default is `true`. Values: `true|false` Back to parameter list	<Base URL>geometry=148.641921, -35.164522, 148.713332, -35.986594&geometryType=esriGeometryEnvelope&spatialRel=esriSpatialRelIntersects&distance=100&units=esriSRUnit_Meter&returnGeometry=false&outFields=*
maxAllowableOffset	This option can be used to specify the `maxAllowableOffset` to be used for generalizing geometries returned by the query operation. The `maxAllowableOffset` is in the units of `outSR`. If `outSR` is not specified, `maxAllowableOffset` is assumed to be in the unit of the spatial reference of the map (EPSG4326).	<Base URL>geometry=148.641921, -35.164522, 148.713332, -35.986594&geometryType=esriGeometryEnvelope&spatialRel=esriSpatialRelIntersects&distance=100&units=esriSRUnit_Meter&outFields=*&maxAllowableOffset=2 <Base URL>geometry=148.641921, -35.164522, 148.713332, -35.986594&geometryType=esriGeometryEnvelop

	Back to parameter list	e&spatialRel=esriSpatialRelIntersects&distance=100&units=esriSRUnit_Meter&outFields=*&maxAllowableOffset=0.05
geometryPrecision	This option can be used to specify the number of decimal places in the response geometries returned by the Query operation. This applies to X and Y values only (not m or z-values). Back to parameter list	<Base URL>where=Category = 'Roadworks'&outFields=*&f=pjson&geometryPrecision=3
outSR	The spatial reference of the returned geometry. If the `outSR` is not specified, the geometry is assumed to be the same as the native layer spatial reference (EPSG4326). Back to parameter list	
havingClause	This option is a condition used with `outStatistics` that limits the query result to groups which satisfy the aggregation function used. The `havingClause` parameter is used with the `groupBy` and `outStatistics` parameters	

	and allows you to filter results from `outStatistics`. Values: `AVG` | `COUNT` | `SUM` | `STDDEV` | `MIN` | `MAX` | `VAR` Back to parameter list	
gdbVersion	The geodatabase version to query. Suggest leaving this blank to query the most recent version of the database. Back to parameter list	
historicMoment	This is the historic moment to query. Suggest leaving this blank to query the most recent version of the database. Back to parameter list	
returnDistinctValues	If `true`, it returns distinct values based on the fields specified in `outFields`. In other words, this removes duplicate values. Users should be aware that the geometry field is considered, so set `returnGeometry` to `false` when `returnDistinctValues` is `true` to ensure reliable results are returned, unless you also want to query for distinct locations. Values: `true` | `false`	<Base URL>returnGeometry=false&where=Category = 'Road Conditions' and state = 'NT'&outFields=type, category, state&f=pjson&returnDistinctValues=true

	Back to parameter list	
returnIdsOnly	If `true`, the response only includes an array of object IDs. Otherwise, the response is a feature set. The default is `false`. While there is a limit to the number of features included in the feature set response, there is no limit to the number of object IDs returned in the ID array response. Clients can exploit this to get all the query conforming object IDs by specifying `returnIdsOnly=true` and subsequently requesting feature sets for subsets of object IDs. Values: `true` | `false` Back to parameter list	<Base URL>where=Category = 'Roadworks' and state = 'WA'&f=pjson&returnIdsOnly=true
returnCountOnly	If `true`, the response only includes the count (number of features/records) that would be returned by a query. Otherwise, the response is a feature set. The default is `false`. This option supersedes the `returnIdsOnly` parameter. If `returnCountOnly = true`, the response will return both the count and the extent. This parameter can be used with `returnDistinctValues` to return the count of distinct values of subfields. Values: `true` | `false`	<Base URL>where=Category = 'Roadworks' and state = 'WA'&f=pjson&returnCountOnly=true

	Back to parameter list	
returnExtentOnly	If `true`, the response only includes the extent of the features that would be returned by the query. If `returnCountOnly=true`, the response will return both the count and the extent. The default is `false`. Values: `true | false` Back to parameter list	<Base URL>where=Category = 'Roadworks' and state = 'WA'&f=pjson&returnCountOnly=true&returnExtentOnly=true
orderByFields	One or more field names on which the features/records need to be ordered. Use `ASC` or `DESC` for ascending or descending, respectively, following every field to control the ordering. `orderByFields` defaults to `ASC` (ascending order) if `<ORDER>` is unspecified. Back to parameter list	<Base URL>where=state = 'NSW'&outFields=*&orderByFields=from_date DESC, state, category&f=pjson
groupByFieldsForStatistics	One or more field names on which the values need to be grouped for calculating the statistics. `groupByFieldsForStatistics` is valid only when the `outStatistics` parameter is used. Syntax:	<Base URL>outFields=*&orderByFields=dateEarliestRecordCaptured&f=pjson&groupByFieldsForStatistics=state&outStatistics=%5B{"statisticType":"count","onStatisticField":"state","outStatisticFieldName":"numberTotalRecords"},{"statisticType":"min","onStatisticField":"capture_date","outStatisticFieldName":"dateEarliestRecordCaptured"}%5D

	groupByFieldsForStatistics= type, category Back to parameter list	
outStatistics	The definitions for one or more field-based statistics to be calculated. When using `outStatistics`, the only other parameters that can be used are `groupByFieldsForStatistics`, `orderByFields`, `time`, and `where`. Note: If `outStatisticFieldName` is empty or missing, the map server assigns a field name to the returned statistic field. A valid field name can only contain alphanumeric characters and an underscore. If the `outStatisticFieldName` is a reserved keyword of the underlying DBMS, the operation can fail. Try specifying an alternative `outStatisticFieldName`. Syntax: [{ "statisticType": "<count | sum | min | max | avg | stddev | var>", "onStatisticField": "Field1", "outStatisticFieldName": "Out_Field_Name1" }, {	<Base URL>outFields=*&orderByFields=dateEarliestRecordCaptured&f=pjson&groupByFieldsForStatistics=state&outStatistics=%5B{"statisticType":"count","onStatisticField":"state","outStatisticFieldName":"numberTotalRecords"},{"statisticType":"min","onStatisticField":"capture_date","outStatisticFieldName":"dateEarliestRecordCaptured"}%5D NOTE: The square brackets '[' and ']' need to be changed to '%5B' and '%5D' in the URL query.

	"statisticType": "<count | sum | min | max | avg | stddev | var>",
	"onStatisticField": "Field2",
	"outStatisticFieldName": "Out_Field_Name2"
	}
	]
	
	Example syntax to obtain the total number of records and the date of the earliest record captured for each state in the dataset:
	
	`[{"statisticType":"count","onStatisticField":"state","outStatisticFieldName":"numberTotalRecords"},{"statisticType":"min","onStatisticField":"capture_date","outStatisticFieldName":"dateEarliestRecordCaptured"}]`
	
	Back to parameter list

returnZ	If `true`, Z values are included in the results if the features have Z values. Otherwise, Z values are not returned. The default is `false`. This parameter only applies if `returnGeometry` is `true`, and the layer's `hasZ` property is `true`. Back to parameter list
returnM	If `true`, M values are included in the results if the features have M values. Otherwise, M values are not returned. The default is `false`.

	This parameter only applies if `returnGeometry` is `true`, and the layer's `hasM` property is `true`. Back to parameter list	
multipatchOption	This option dictates how the geometry of a multipatch feature will be returned. This parameter only applies if the layer's `geometryType` property is `esriGeometryMultiPatch`. The default is `xyFootprint`. You do not need to specify this parameter. Values: `xyFootprint|stripMaterials|embedMaterials|externalizeTextures` Back to parameter list	
resultOffset	This option can be used for fetching query results by skipping the specified number of records and starting from the next record (that is, `resultOffset` + 1th). The default is 0. You can use this option to fetch records that are beyond `maxRecordCount`. Back to parameter list	<Base URL>where=state = 'NSW'&outFields=*&orderByFields=from_date DESC, state, category&f=pjson&resultOffset=2000
resultRecordCount	This option can be used for fetching query results up to the `resultRecordCount` specified. When `resultOffset` is specified but this parameter is not, the map service defaults it to `maxRecordCount` (2000). The maximum value	<Base URL>where=state = 'NSW'&outFields=*&orderByFields=from_date DESC, state, category&f=pjson&resultRecordCount=100&resultOffset=2000

	for this parameter is the value of the layer's `maxRecordCount` property. The minimum value entered for this parameter cannot be below 1. Back to parameter list
returnTrueCurves	When set to `true`, returns true curves in output geometries. When set to `false`, curves are converted to densified polylines or polygons. The default is `false`. Values: `true` | `false` Back to parameter list
returnCentroid	Used to return the geometry centroid associated with each feature returned. If `true`, the result includes the geometry centroid. The default is `false`. Currently, this parameter is not supported for polygon data and is ignored for `count` and `objectID` queries. This parameter is only supported on layer-level queries. Values: `true` | `false` Back to parameter list
sqlFormat	The sqlFormat parameter can be either standard SQL92 standard or it can use the native SQL of the underlying data store native. This parameter

	should always be set to `None`. You can leave this blank. Values: `none` | `standard` | `native` Back to parameter list	
resultType	The `resultType` parameter can be used to control the number of features returned by the query operation. The `tile` value is used when the client is using a virtual tiling scheme when querying features, which works similarly to tiles in a tiled map service layer. The `standard` value is used with a non-tiled query where the client will send only one query for the full extent. Pagination queries also support `resultType` query parameter. If the `resultType` is specified and the `resultRecordCount` is not specified with the `resultOffset`, the server will determine the `maxRecordCount` relevant to the `resultType` query parameter. The client can supply the `resultRecordCount` parameter in the request. This cannot be greater than the standard/tile `maxRecordCount` value if `resultType` is used. The limits are as follows: • when `resultType` = `none`, the maximum number of results that can be returned is 2,000	<Base URL>returnGeometry=false&where=1=1&outFields=*&orderByFields=state&f=pjson&resultType=tile <Base URL>returnGeometry=false&where=1=1&outFields=*&orderByFields=state&f=pjson&resultType=standard

	• when `resultType` = `standard`, the maximum number of results that can be returned is 16,000 • when `resultType` = `tile`, the maximum number of results that can be returned is 4,000 Values: `none|standard|tile` Back to parameter list	
datumTransformation	This parameter applies a datum transformation while projecting geometries in the results when `outSR` is different than the layer's spatial reference. You can leave this blank. Back to parameter list	
f	The response format. The default response format is `html`. Values: `html|json|geojson|pbf` Back to parameter list	<Base URL>where=Category = 'Roadworks'&outFields=*&f=pjson&geometryPrecision=3

6. Limitations

6.1 Record count

Only the first 2,000 results are returned when running a default query. The actual number of total records which match the query parameters may be more than 2,000.

To display more than 2,000 records we can specify the resultType parameter:

- When resultType = standard, this will return 16,000 records
    - NOTE: Running this takes longer, and depending on how many attributes you want to return is prone to crashing the webpage.
- When resultType = none, this will return the MaxRecordCount of 2,000 records
- When resultType = tile, this will return 4,000 records

It should be noted that none of these options will allow a user to display all records, as there are more than 170,000 records in the dataset.

If a large number (more than 16,000) records are required to be outputted, users can also make use of the resultOffset parameter. For example, running the following queries in succession will allow the most recently captured 32,000 records for QLD to be obtained:

1. https://spatial.infrastructure.gov.au/server/rest/services/Hosted/RADAR_Curated_Prod_roadworks/FeatureServer/0/query?returnGeometry=false&where=state = 'QLD'&outFields=*&orderByFields=capture_date DESC&f=pjson&resultType=standard
2. https://spatial.infrastructure.gov.au/server/rest/services/Hosted/RADAR_Curated_Prod_roadworks/FeatureServer/0/query?returnGeometry=false&where=state = 'QLD'&outFields=*&orderByFields=capture_date DESC&f=pjson&resultOffset=16000&resultType=standard

6.2 Output formats

ArcGIS REST API only supports the following formats for the Harmonised National Roadworks and Road Closures dataset:

- HTML
- PBF
- JSON
- GeoJSON