ERC Redesign - Technical implementations

Contents

[Requirements: 2](#_Toc15378009)

[Drupal CMS Dev activities: 2](#_Toc15378010)

[How to create RESTful APIs to expose contents using Views? 3](#_Toc15378011)

[Workflows: 4](#_Toc15378012)

[Views: How to get related articles of current node? 5](#_Toc15378013)

[How to configure Views Exposed Filter for RESTful API? 8](#_Toc15378014)

[Views – how to combine multiple views together as a single view output? 9](#_Toc15378015)

[Views: how to create search RESTful API? 10](#_Toc15378016)

[Views: How to setup pagination for RESTful API? 11](#_Toc15378017)

[How to separate all fields of paragraph fieldgroup? 11](#_Toc15378018)

# Requirements:

1. Manual content migration from existing WordPress site.
2. RESTful API creation and expose data to UI team.
3. UI team will develop the mobile app.

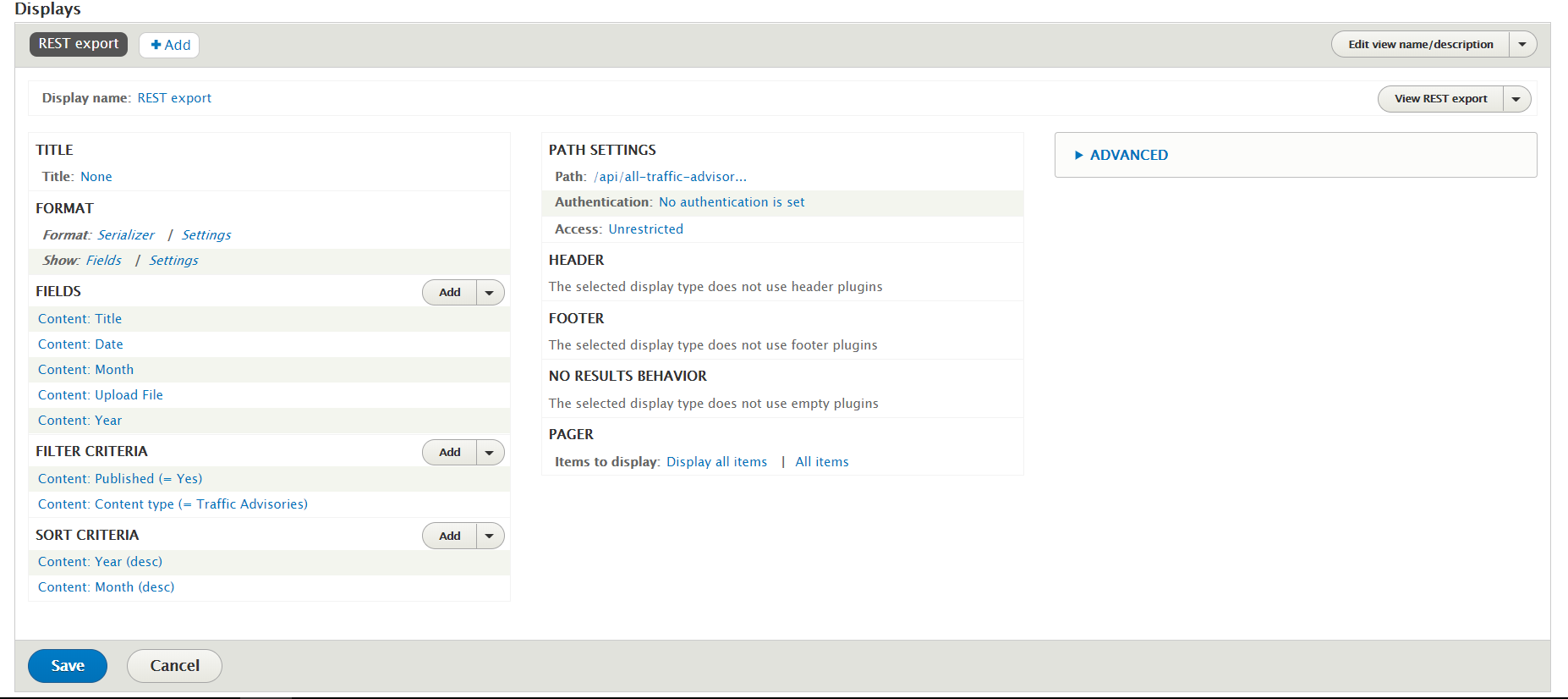
# Drupal CMS Dev activities:

1. Create content types with all the necessary fields.
2. Create contents as per the requirements.
3. Modules: Paragraphs, REST UI, RESTful Web Services, Workflows.
4. Create views to expose contents.

# How to create RESTful APIs to expose contents using Views?

Modules: REST UI, RESTful Web Services.

Views: create a view as REST export.



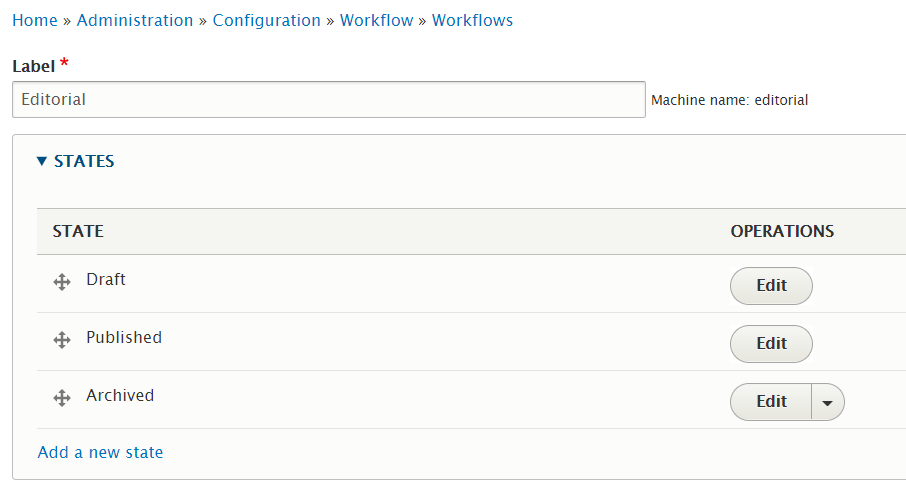
API Url: api/all-traffic-advisory

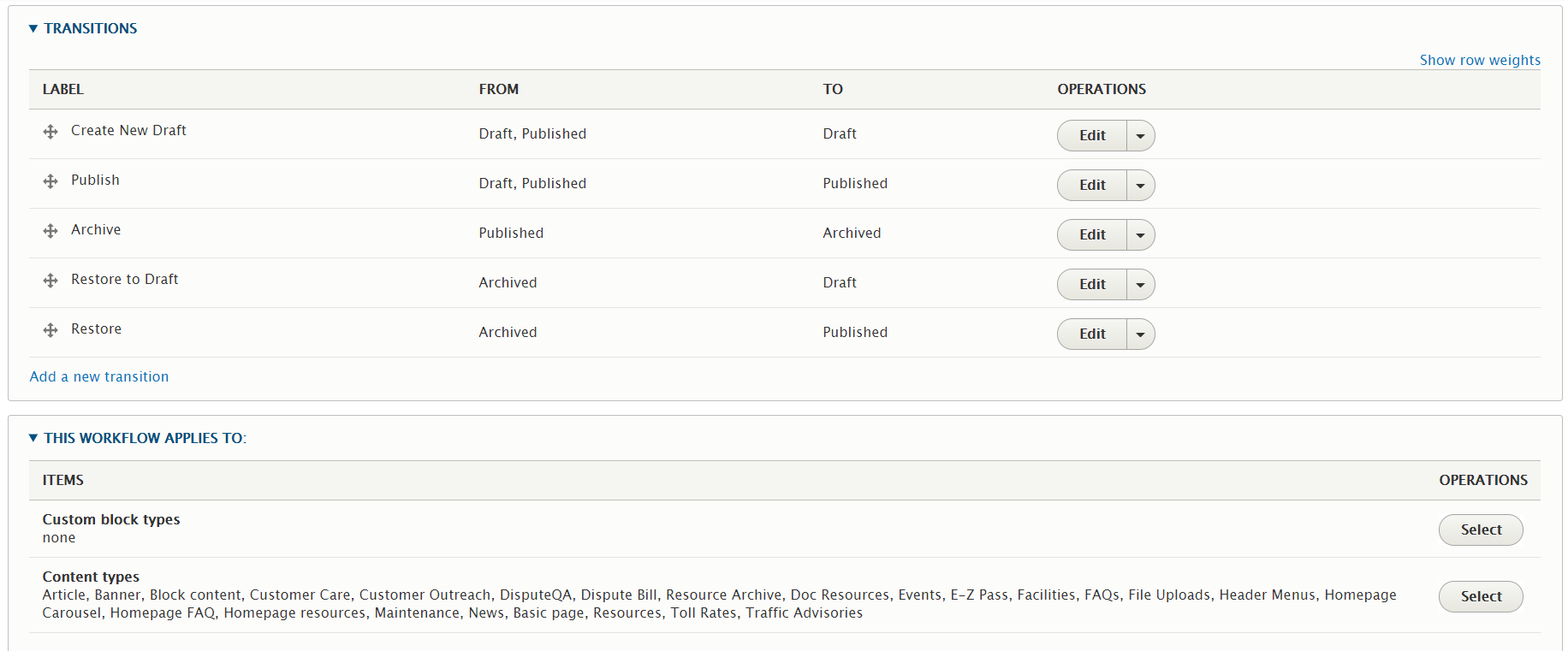
Serializer settings: Accepted request formats: json

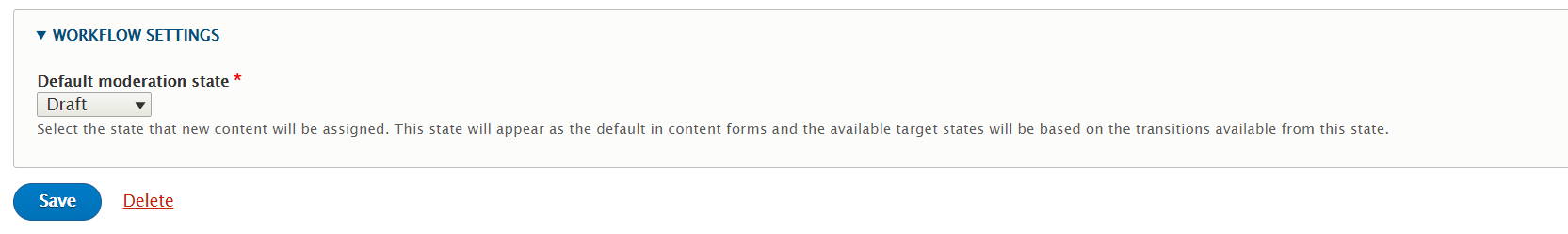
For all the fields: Uncheck Add default classes, Remove whitespaces (if necessary) and check Strip HTML tags.

# Workflows:

This module works along with Content moderation module.







# Views: How to get related articles of current node?

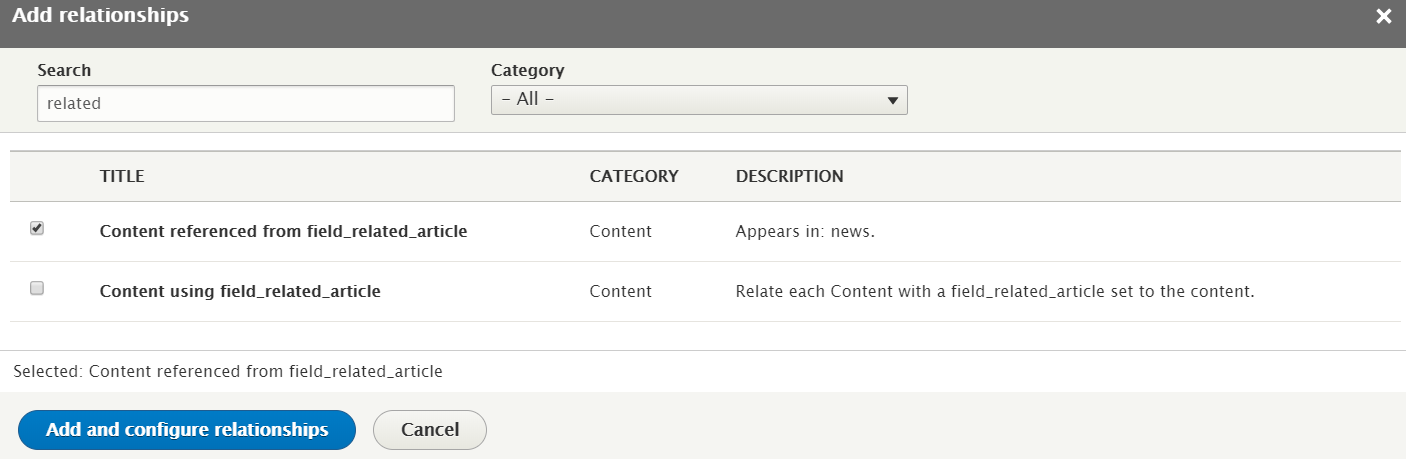
Content reference:

News 3 node is having related articles: News 1 and News 2.



Views configuration:

Add relationship:



Here, “field\_related\_article:” is the field name which we created in content type as Entity reference.



Uncheck “Require this relationship”



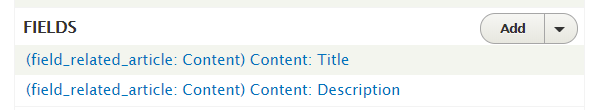
No need to add “CONTEXTUAL FILTERS”

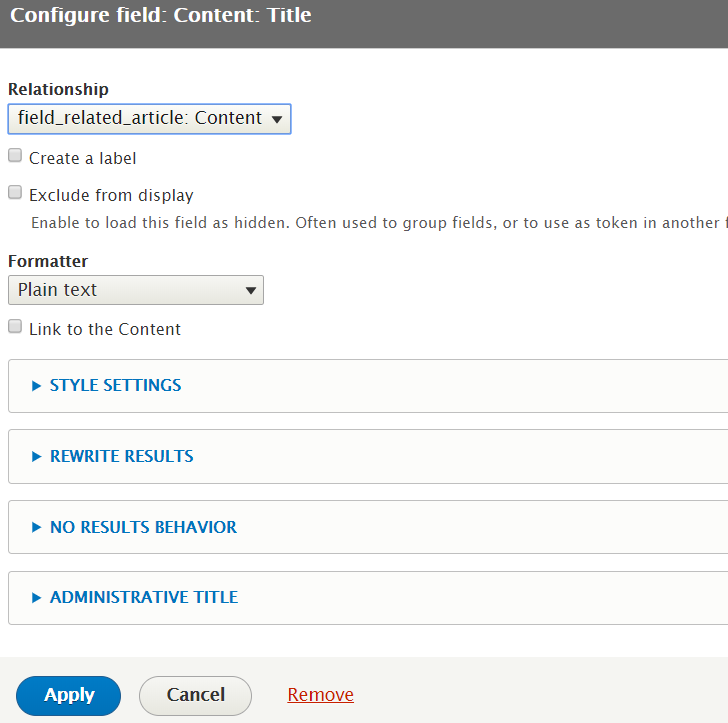
Filter the node id = 1436.



We are going to get all the node details which are marked as related article for the particular node 1436.

Add all the fields with the relationship “field\_related\_article: Content”





JSON Result:

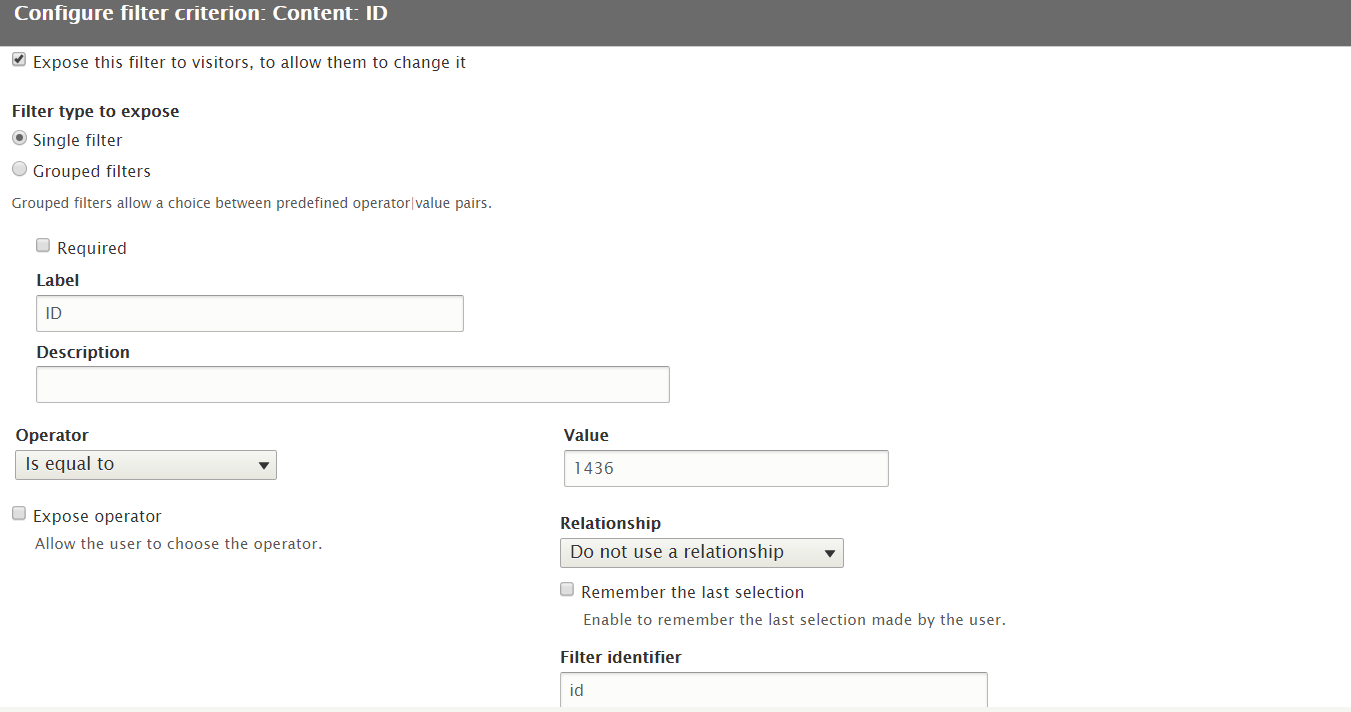
[{"title":"News 1","body":"News 1 content.\u00a0News 1 content.\u00a0News 1 content.\u00a0News 1 content.\u00a0News 1 content.\u00a0News 1 content.\u00a0News 1 content.\u00a0News 1 content.\u00a0News 1 content.\u00a0"},{"title":"News 2","body":"News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0News 2 content.\u00a0"}]

The result will display News 1 and News 2 contents when we visit News 3 API.

# How to configure Views Exposed Filter for RESTful API?

Service Url: api/news-related?id=1436

Check Exposed filter and enter filter identifier.



No contextual filter required.

# Views – how to combine multiple views together as a single view output?

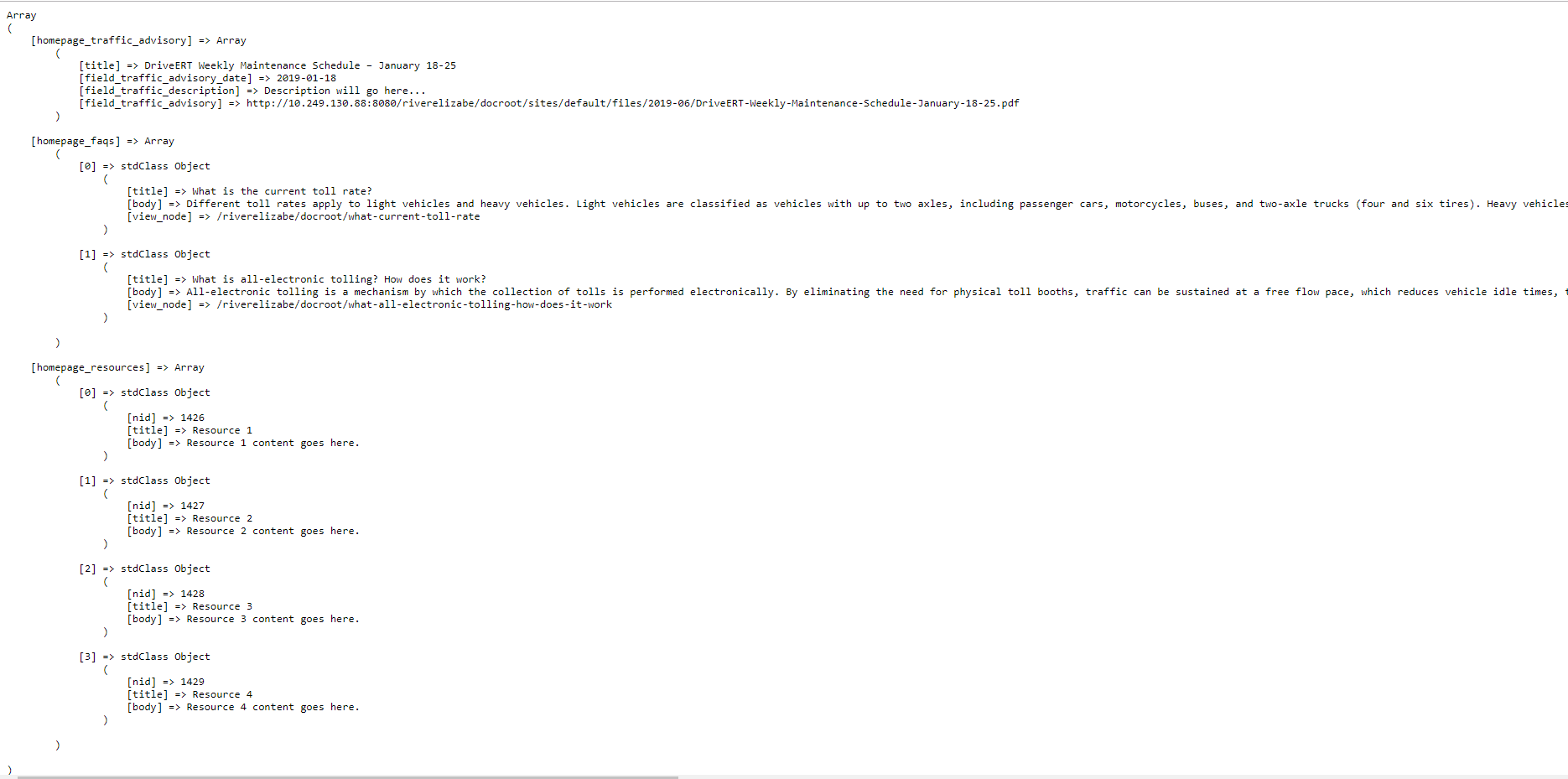
Hook function:

function twitter\_feed\_views\_pre\_render(\Drupal\views\ViewExecutable $view) {

}

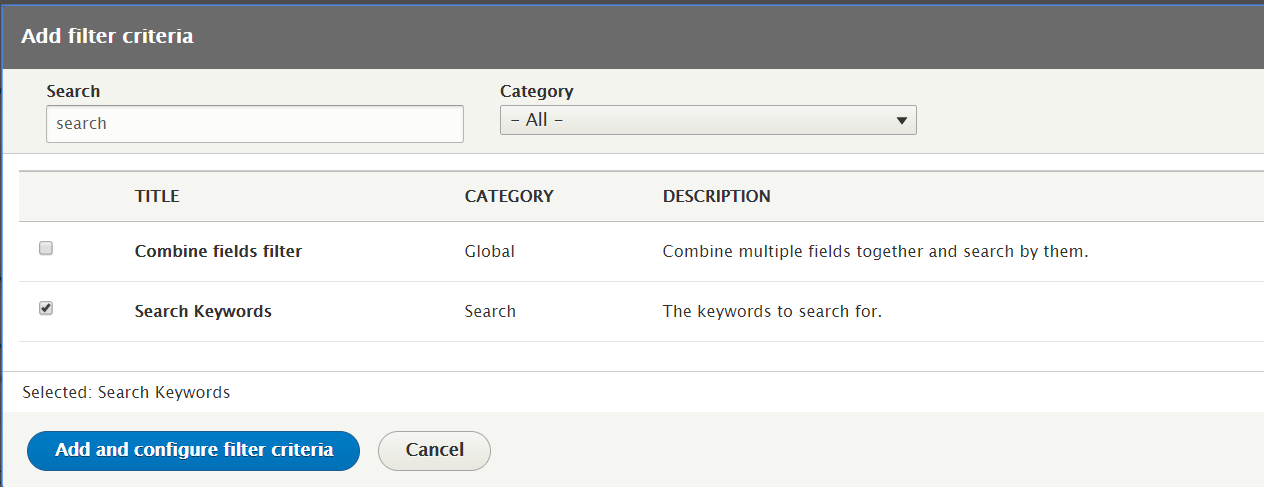


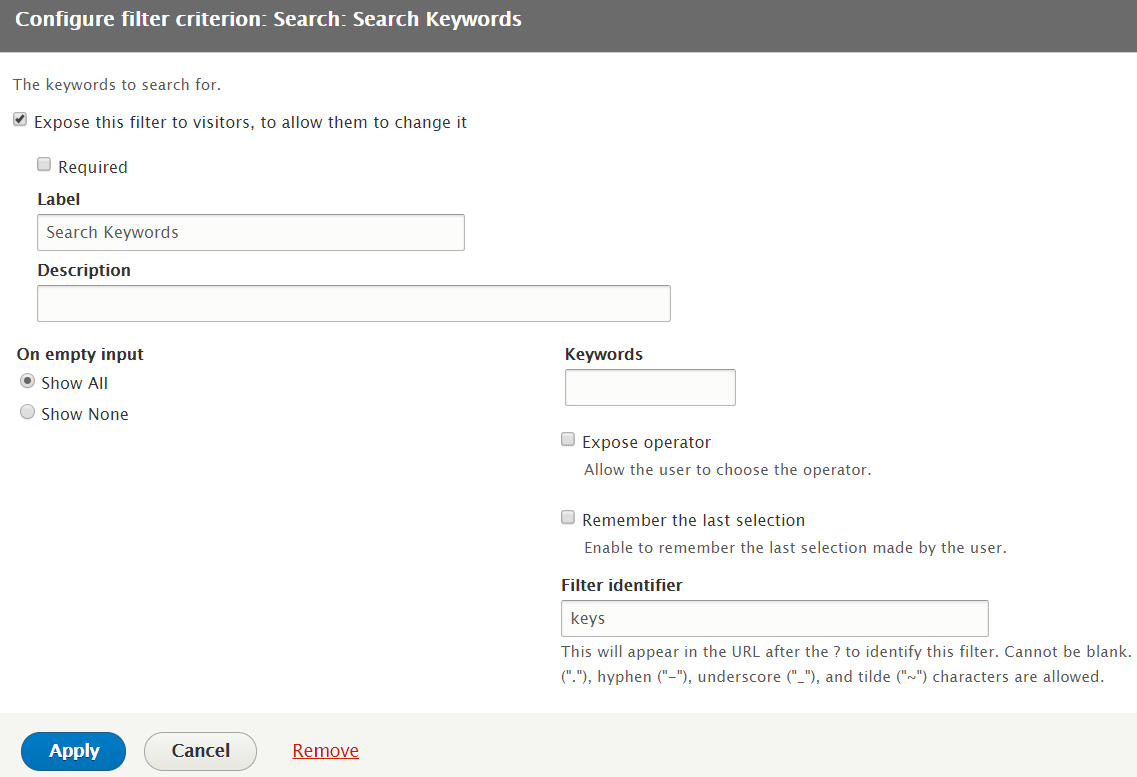
View output:



# Views: how to create search RESTful API?

Add filter criteria and expose it.





Url: ?keys=[text to search]

# Views: How to setup pagination for RESTful API?



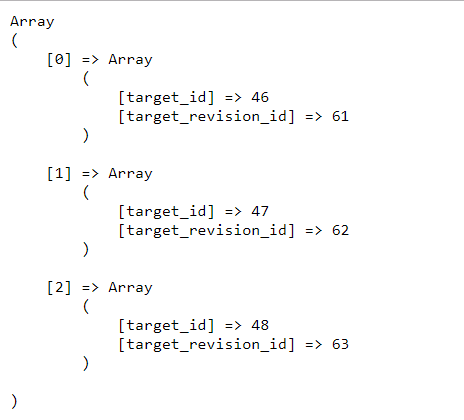
Url: ?page=1

# How to separate all fields of paragraph fieldgroup?

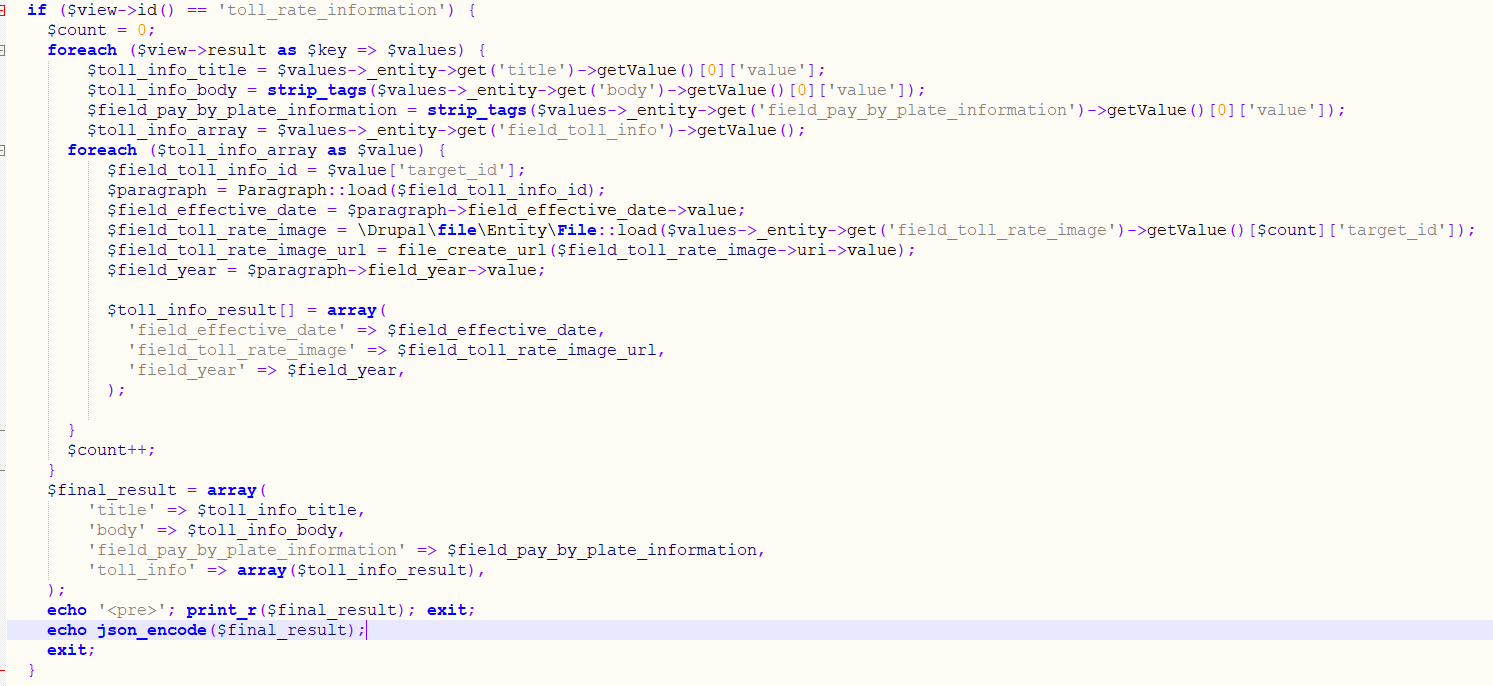
The paragraph fields is:

$toll\_info\_array = $values->\_entity->get('field\_toll\_info')->getValue();

Echo ‘<pre>’; print\_r($toll\_info\_array); exit;



The complete code is:



The output is:

