

Getting started

After [installing](#) the Designer plug-in, you can see a new design element called `XRest API Routes` under the `Application Configuration` category.

It is used for editing [routes](#) that your application will be listening on for REST calls.

Basic syntax of routes.groovy

Each endpoint is defined in following way:

```
router.METHOD('route/path/{variable}') {
    strategy(STRATEGY_NAME){
        ... strategy parameters ...
    }
    mapJson 'notesFieldName', json:'jsonFieldName', type:
'String|Integer|Double|Datetime|Array_of_String|Array_of_Integer|Array_of_Double|Array_of_Datetime', isformula:
true/false, formula: '', readonly: true/false, writeonly: true/false
    mapJson ...
    accessPermission 'groupName1', 'groupName2', '[roleName1]', '[roleName2]', ...
}
```

where `METHOD` is one of:

- GET
- POST
- PUT
- DELETE

and `STRATEGY` is one of:

- `DOCUMENT_BY_UNID`
- `DOCUMENT_FROM_VIEW_BY_KEY`
- `DOCUMENTS_BY_SEARCH_FT_PAGED`
- `DOCUMENTS_BY_FORMULA_PAGED`
- `DOCUMENTS_BY_VIEW_PAGED`
- `DOCUMENTS_FROM_VIEW_BY_KEY_PAGED`
- `VIEWENTRIES_PAGED`
- `VIEWENTRIES_BY_CATEGORY_PAGED`
- `ATTACHMENT`

Non-paged strategies are available as well, but consider using these just for small data sets, because these strategies return all relevant data at once:

- `DOCUMENTS_BY_SEARCH_FT`
- `DOCUMENTS_BY_FORMULA`
- `DOCUMENTS_BY_VIEW`
- `DOCUMENTS_FROM_VIEW_BY_KEY`
- `VIEWENTRIES`
- `VIEWENTRIES_BY_CATEGORY`

`accessPermission`: list of groups and/or roles that the user must be a member of to be allowed to use the endpoint

`mapJson`: list of fields that you want to read from documents, or formulas that you want to be executed in the context of current document

Example route

Let's try to create a simple `GET` route: assuming we have a view named (`Topics`) in our test database, and it contains some documents with the `Topic` field in them, then we can enter this in our `routes.groovy` file:

```
router.GET('docs') {
    strategy(DOCUMENTS_BY_VIEW) {
        viewName('(Topics)')
    }
    mapJson "date", json:'date', type:'DATETIME', isformula:true, formula:'@Created'
    mapJson 'Topic', json: 'topic', type: 'STRING'
}
```

and access our new REST point at:

`http://server.name/path/database.nsf/xsp/.xrest/docs`

Returned data should look like this:

```
{
  [
    { "topic": "Topic 1", "date": "2017-03-03T18:28+00:00" },
    { "topic": "Topic 2", "date": "2017-03-03T18:28+00:00" }
    ...
  ]
}
```

You can see it is basically returning the content of the `Topic` field for each document in the `(Topics)` view.

This example used a non-page strategy, which returns all relevant data at once. When the application contains more than a few documents, it might be better not to read all of them with one call.

If we change the example to use a paged strategy such as `DOCUMENTS_BY_VIEW_PAGED`, like this:

```
router.GET('docs') {
  strategy(DOCUMENTS_BY_VIEW_PAGED) {
    viewName('(Topics)')
  }
  mapJson 'date', json:'date', type:'DATETIME', isformula:true, formula:'@Created'
  mapJson 'Topic', json: 'topic', type: 'STRING'
}
```

we can now pass the parameters `start` and `count` in our URL to get just a subset of documents. For example, calling:

`http://server.name/path/database.nsf/xsp/.xrest/docs?start=100&count=5`

will return 5 documents starting at position 100 in the view. The returned JSON is a bit different:

```
{
  "start": 100,
  "count": 5,
  "total": 24900,
  "entries": [
    {
      "topic": "Topic 100",
      "date": "2017-03-03T18:28+00:00"
    },
    {
      "topic": "Topic 101",
      "date": "2017-03-03T18:28+00:00"
    },
    {
      "topic": "Topic 102",
      "date": "2017-03-03T18:28+00:00"
    },
    {
      "topic": "Topic 103",
      "date": "2017-03-03T18:28+00:00"
    },
    {
      "topic": "Topic 104",
      "date": "2017-03-03T18:28+00:00"
    }
  ]
}
```

✔ Congratulations, you've just created REST service to access data in Domino database using SmartNSF project!

Related articles

- [Strategy: CUSTOM](#)
- [Strategy: DOCUMENT_BY_UNID](#)
- [Strategy: DOCUMENT_FROM_VIEW_BY_KEY](#)
- [Strategy: DOCUMENTS_BY_FORMULA](#)
- [Strategy: DOCUMENTS_BY_FORMULA_PAGED](#)
- [Strategy: DOCUMENTS_BY_SEARCH_FT](#)
- [Strategy: DOCUMENTS_BY_SEARCH_FT_PAGED](#)
- [Strategy: DOCUMENTS_BY_VIEW](#)
- [Strategy: DOCUMENTS_BY_VIEW_PAGED](#)
- [Strategy: DOCUMENTS_FROM_VIEW_BY_KEY](#)
- [Strategy: DOCUMENTS_FROM_VIEW_BY_KEY_PAGED](#)
- [Strategy: VIEWENTRIES](#)
- [Strategy: VIEWENTRIES_BY_CATEGORY](#)
- [Strategy: VIEWENTRIES_BY_CATEGORY_PAGED](#)
- [Strategy: VIEWENTRIES_PAGED](#)