

CouchDB advanced features and tools for large-scale applications

Author

Tomasz Kisiel

Supervisors

Hubert Odziemczyk

Artur Wiecek



Table of Contents

01

Introduction

02

Integration with CERN SSO

03

Fine-grained access management

04

Managing design documents

05

Data replication

06

Inspecting documents

07

Monitoring and alerting

Introduction

What is CouchDB?

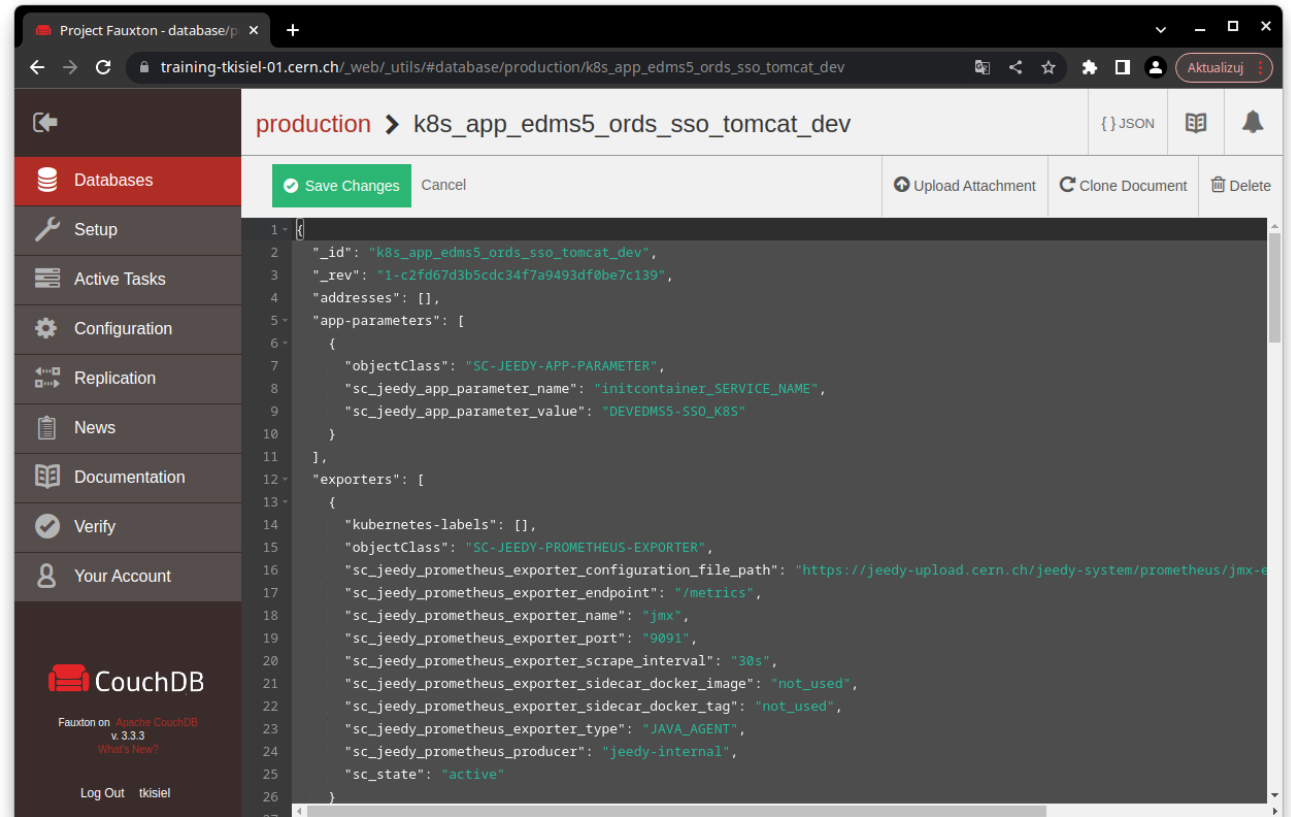
Document-Oriented No-SQL

Scalable and Reliable Solution

How we use it?

Metadata storage for Kubernetes clusters and Java applications

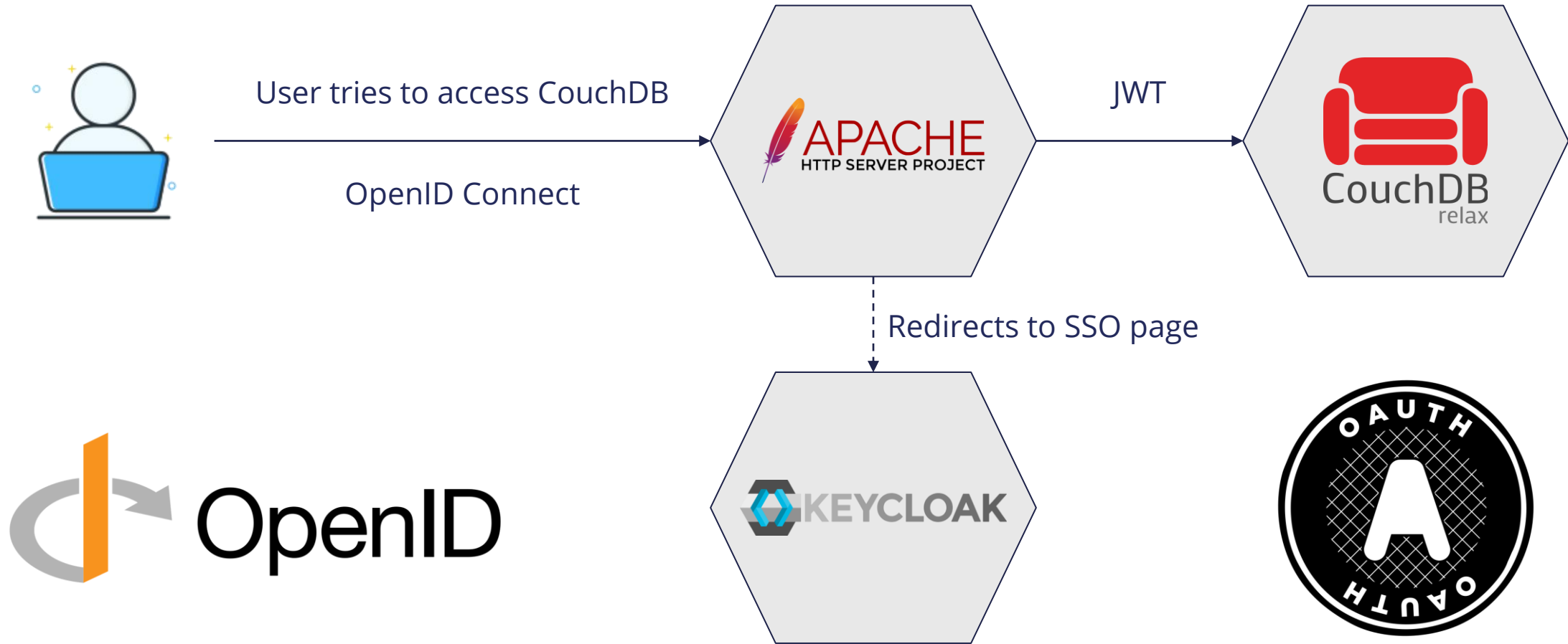
Managed with Puppet outside of a Kubernetes cluster



The screenshot shows the Project Fauxton web interface for a CouchDB database. The browser address bar shows the URL: `training-tkisiel-01.cern.ch/_utils/#database/production/k8s_app_edms5_ordms_sso_tomcat_dev`. The page title is `production > k8s_app_edms5_ordms_sso_tomcat_dev`. The interface includes a sidebar with navigation options: Databases, Setup, Active Tasks, Configuration, Replication, News, Documentation, Verify, and Your Account. The main content area displays a document with the following JSON structure:

```
1- |
2- |  "_id": "k8s_app_edms5_ordms_sso_tomcat_dev",
3- |  "_rev": "1-c2fd67d3b5cdc34f7a9493df0be7c139",
4- |  "addresses": [],
5- |  "app-parameters": [
6- |    {
7- |      "objectClass": "SC-JEEDY-APP-PARAMETER",
8- |      "sc_jeedy_app_parameter_name": "initcontainer_SERVICE_NAME",
9- |      "sc_jeedy_app_parameter_value": "DEVEDMS5-SSO_K8S"
10- |    }
11- |  ],
12- |  "exporters": [
13- |    {
14- |      "kubernetes-labels": [],
15- |      "objectClass": "SC-JEEDY-PROMETHEUS-EXPORTER",
16- |      "sc_jeedy_prometheus_exporter_configuration_file_path": "https://jeedy-upload.cern.ch/jeedy-system/prometheus/jmx-...",
17- |      "sc_jeedy_prometheus_exporter_endpoint": "/metrics",
18- |      "sc_jeedy_prometheus_exporter_name": "jmx",
19- |      "sc_jeedy_prometheus_exporter_port": "9091",
20- |      "sc_jeedy_prometheus_exporter_scrape_interval": "30s",
21- |      "sc_jeedy_prometheus_exporter_sidecar_docker_image": "not_used",
22- |      "sc_jeedy_prometheus_exporter_sidecar_docker_tag": "not_used",
23- |      "sc_jeedy_prometheus_exporter_type": "JAVA_AGENT",
24- |      "sc_jeedy_prometheus_exporter_producer": "jeedy-internal",
25- |      "sc_state": "active"
26- |    }
27- |  ]
```

Integration with CERN SSO



Fine-grained access management

```
</> YAML
1 hg_jeedy::jeeman::couchdb_production_rbac:
2   - role: '_admin'
3     documents: ['*']
4     properties: ['*']
5   - role: 'rths-dev'
6     documents:
7     - k8s_app_rths_tomcat_dev
8     - k8s_app_rths_tomcat_test
9     - k8s_app_rths_tomcat_preprod
10    - k8s_app_rths_tomcat_prod
11   properties:
12   - 'sc_category'
13   - 'sc_subcategory'
14   - 'webapps[*].objectClass'
15   - 'webapps[*].sc_jeedy_sso_webapp_config.enabled'
16   - 'webapps[*].tomcat_valves[*].objectClass'
17   - 'some-array[*].of-arrays[*]'
```

User

Attempts to
modify the
document

Validation function

Checks whether the user
can modify the specified
properties for the given
document

Allows or deny

Managing design documents

Problems:

- Difficult to understand and manage as the codebase grows.
- Lack of version control leads to potential configuration drift.
- Manual updates are cumbersome and error-prone.
- Lack of automation to ensure consistency between environments.
- Increased risk of conflicts when multiple developers work on the same design documents.

```
</> JSON
1  {
2    "_id": "_design/document-name",
3    "validate_doc_update": "function(newDoc, oldDoc,
                             userCtx) {  const RBAC = require('views/lib/rbac')
                             ['RBAC'];  const diff = require('views/lib/diff')
                             ['diff'];    // Access Control    const
                             listOfAllowedDocuments = [...new Set(RBAC.filter(rbac
                             => userCtx.roles.includes(rbac.role)).flatMap(rbac =>
                             rbac.documents || []))];    const
                             listOfAllowedProperties = [...new Set(RBAC.filter(rbac
                             => userCtx.roles.includes(rbac.role)).flatMap(rbac =>
                             rbac.properties || []))];    if
                             (listOfAllowedDocuments.includes(\"*\") &&
                             listOfAllowedProperties.includes(\"*\")) {
                             return;    }    if
                             (!listOfAllowedDocuments.includes(newDoc._id) &&
                             !listOfAllowedDocuments.includes(\"*\")) {
                             throw({ forbidden: \"You are not authorized to modify
                             document: \" + newDoc._id });    }    const
                             listOfChanges = diff(oldDoc, newDoc).filter(change =>
                             !change.path[0].startsWith('_')); // and more... }"
4  }
```

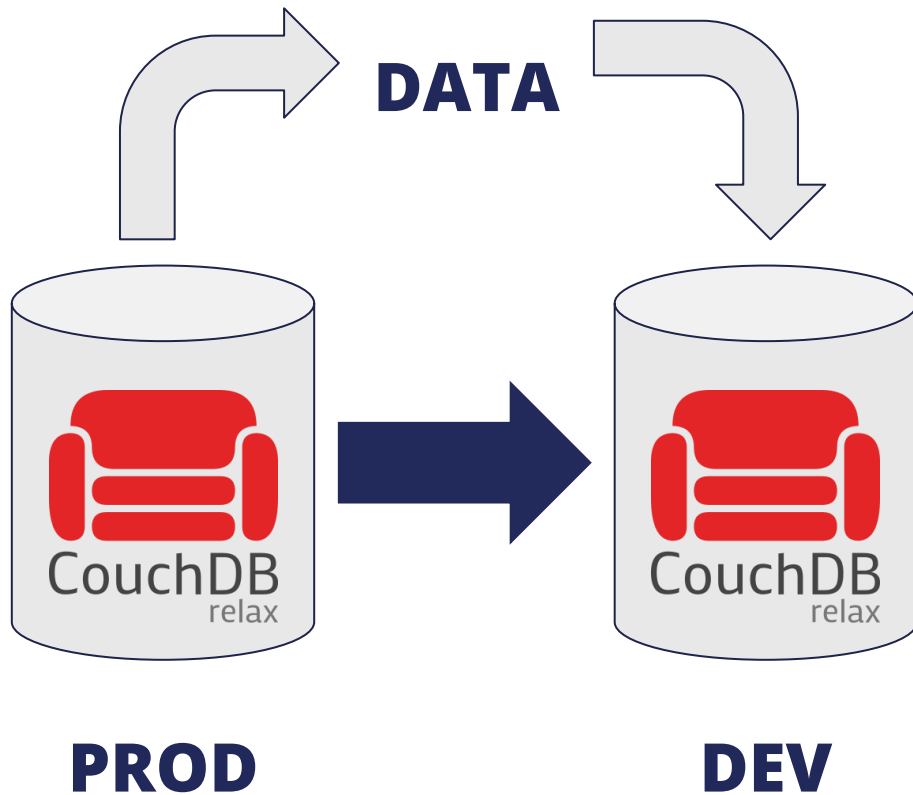
Managing design documents

```
1 database-name
2   └─ _design
3     └─ document-name
4         └─ filters
5             └─ filter-1.js
6                 └─ filter-2.js
7         └─ updates
8             └─ update-1.js
9                 └─ update-2.js
10        └─ views
11            └─ view-1
12                └─ map.js
13                    └─ reduce.js
14            └─ view-2
15                └─ map.js
16                    └─ reduce.js
17        └─ validate_doc_update.js
```



```
</> JSON
1  {
2    "_id": "_design/document-name",
3    "filters": {
4      "filter-1": "function(doc, req) { ... }",
5      "filter-2": "function(doc, req) { ... }"
6    },
7    "updates": {
8      "update-1": "function(doc, req) { ... }",
9      "update-2": "function(doc, req) { ... }"
10   },
11   "views": {
12     "view-1": {
13       "map": "function(doc) { ... }",
14       "reduce": "function(keys, values) { ... }"
15     },
16     "view-2": {
17       "map": "function(doc) { ... }",
18       "reduce": "function(keys, values) { ... }"
19     }
20   },
21   "validate_doc_update": "function(...) { ... }"
22 }
```

Data replication



```
</> JSON
1  {
2    "source": {
3      "url": "<%= $replication_source_url %>/production",
4      "headers": {
5        "Authorization": "%TEIGI__source_credentials__%"
6      }
7    },
8    "target": {
9      "url": "<%= $replication_target_url %>/production",
10     "headers": {
11       "Authorization": "%TEIGI__target_credentials__%"
12     }
13   },
14   "create_target": true,
15   "continuous": true,
16   "owner": "%TEIGI__couchdb_username__%"
17 }
```


Inspecting documents

Jeeman Search

Filters

edms.*dev_2

sc_subcategory: K8S_APP

Databases (5)

production

Documents (9)

sc_subcategory x sc_category x sc_entity x sc_type x exporters.0.sc_jeedy_prometheus_exporter_port x

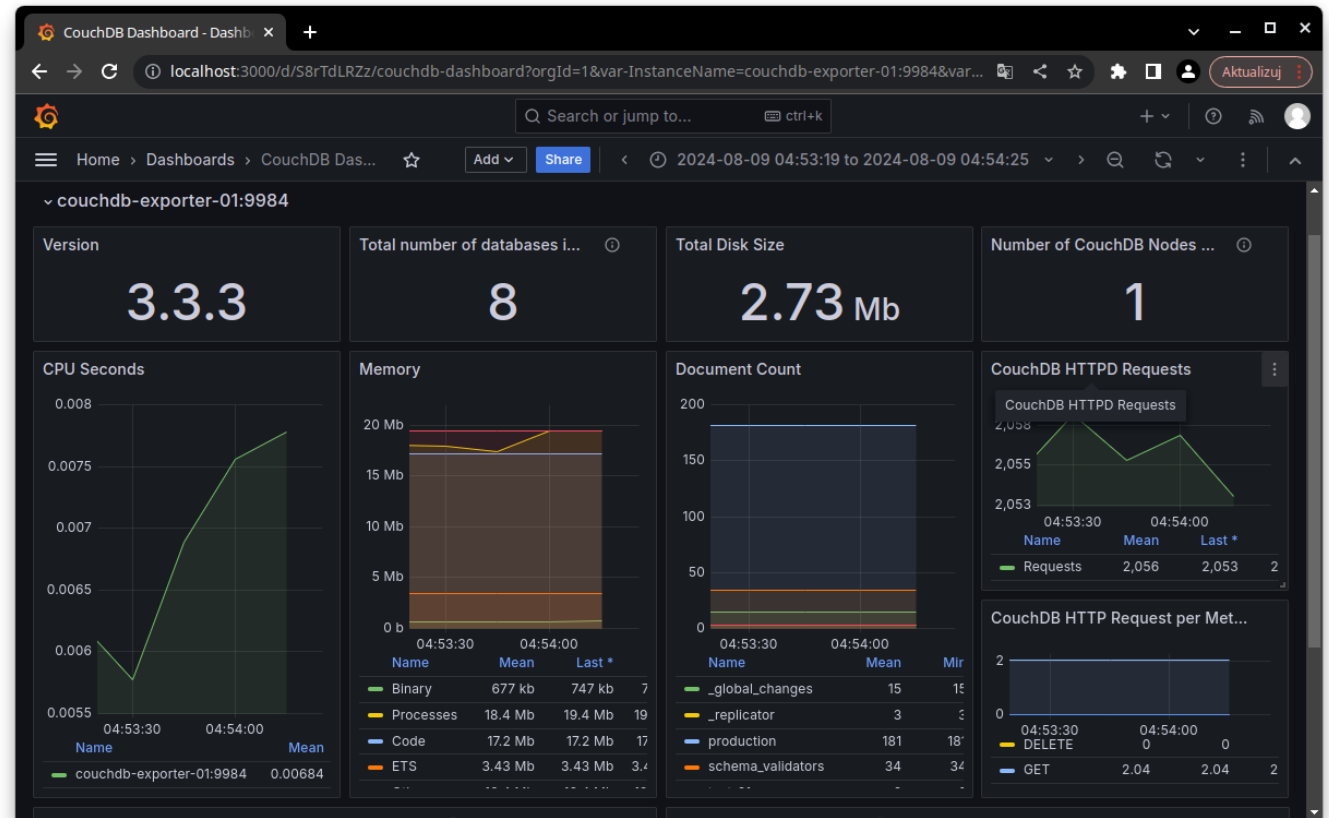
	_id	exporters.0.sc_jeedy_prometheus_exporter_port	sc_category	sc_entity	sc_subcategory	sc_type
Edit	k8s_app_edms5_ords_sso_to_mcat_dev_2	9091	APPSRV	k8s_app_edms5_ords_sso_to_mcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms5_ords_tomcat_dev_2	9091	APPSRV	k8s_app_edms5_ords_tomcat_dev_2	K8S_APP	DEV

	_id	exporters.0.sc_jeedy_prometheus_exporter_port	sc_category	sc_entity	sc_subcategory	sc_type
Edit	k8s_app_edms5_ords_sso_to_mcat_dev_2	9091	APPSRV	k8s_app_edms5_ords_sso_to_mcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms5_ords_tomcat_dev_2	9091	APPSRV	k8s_app_edms5_ords_tomcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms_components_tomcat_dev_2	9091	APPSRV	k8s_app_edms_components_tomcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms_fileserver_to_mcat_dev_2	9091	APPSRV	k8s_app_edms_fileserver_to_mcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms_micado_tomcat_dev_2	9091	APPSRV	k8s_app_edms_micado_tomcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms_panoramas_tomcat_dev_2	9091	APPSRV	k8s_app_edms_panoramas_tomcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms_ui_tomcat_dev_2	9091	APPSRV	k8s_app_edms_ui_tomcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms_ws_tomcat_dev_2	9091	APPSRV	k8s_app_edms_ws_tomcat_dev_2	K8S_APP	DEV
Edit	k8s_app_edms_z_root_tomcat_dev_2	9091	APPSRV	k8s_app_edms_z_root_tomcat_dev_2	K8S_APP	DEV

Monitoring and alerting

Metrics:

- VM-specific metrics
- Number of databases
- Size on disk
- CPU usage
- Memory usage
- Documents count per database
- Number of requests
- Number of requests per method
- Number of clients
- Open OS files
- And more...



Further reading

CouchDB with Keycloak SSO – Part 1

<https://kishieel.github.io/posts/couchdb-keycloak-sso-1/>

CouchDB with Keycloak SSO – Part 2

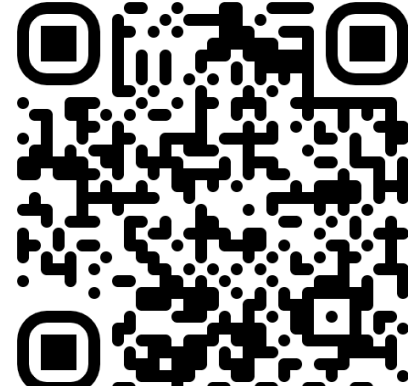
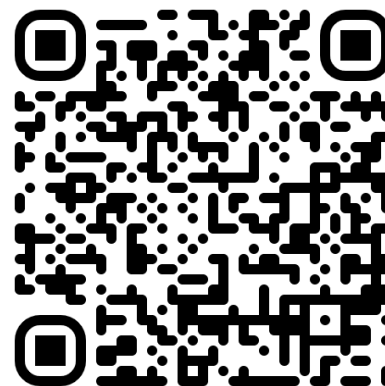
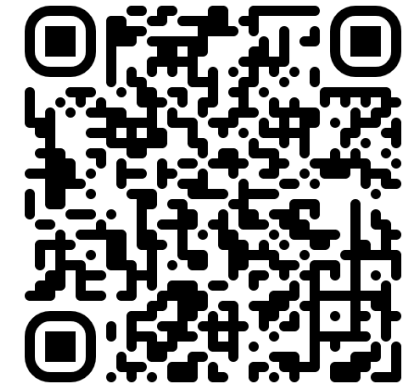
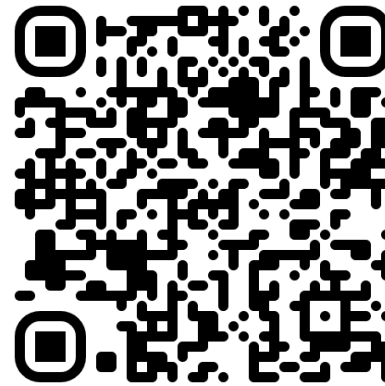
<https://kishieel.github.io/posts/couchdb-keycloak-sso-2/>

CouchGO! — Golang Query Server

<https://kishieel.github.io/posts/couchdb-query-server-go/>

Design document management utility

<https://github.com/kishieel/couchdb-sync>



Thank you!

Questions?

