

# Módulo 6

## Hyperledger Fabric

(finalização)



# Queries no channel

As buscas são feitas dentro do peer através de operações de pesquisa.

As pesquisas são realizadas nas bases de *state* escolhida (**LevelDb** ou **CouchDb**)

**LevelDb** é a base default e roda dentro do container peer

Pesquisas em **LevelDb** são realizadas para as chaves (primárias ou compostas)

**CouchDb** é um modelo mais poderoso e roda em um container separado do *peer*.

*Rich queries* e indexação são permitidas para **CouchDb**

# Trabalhando com CouchDb

O uso do CouchDb pode ser realizado com a configuração dos endorsing peer

Váriáveis de ambiente:

`CORE_LEDGER_STATE_STATEDATABASE=CouchDB`

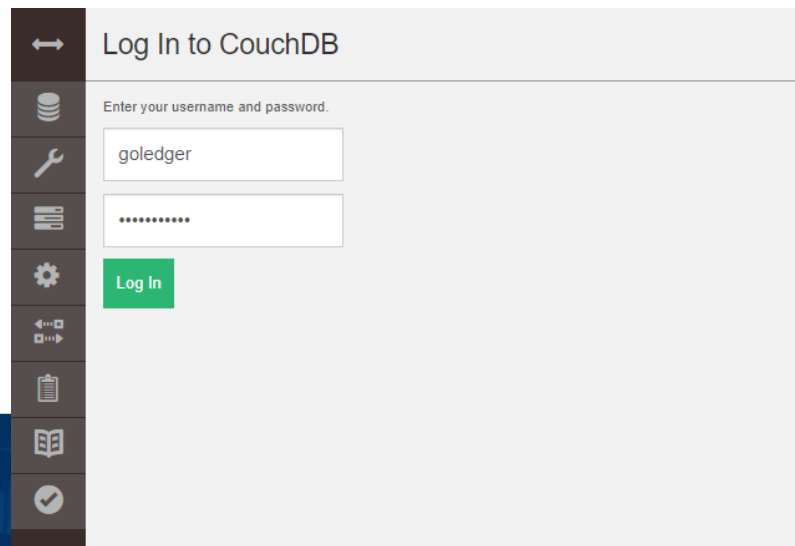
`CORE_LEDGER_STATE_COUCHDBCONFIG_COUCHDBADDRESS=couch.peer0.org.example.com:5984`

`CORE_LEDGER_STATE_COUCHDBCONFIG_USERNAME=admin`

`CORE_LEDGER_STATE_COUCHDBCONFIG_PASSWORD=adminpw`

Exemplo de acesso ao couchdb via interface Web (*Fauxion*)

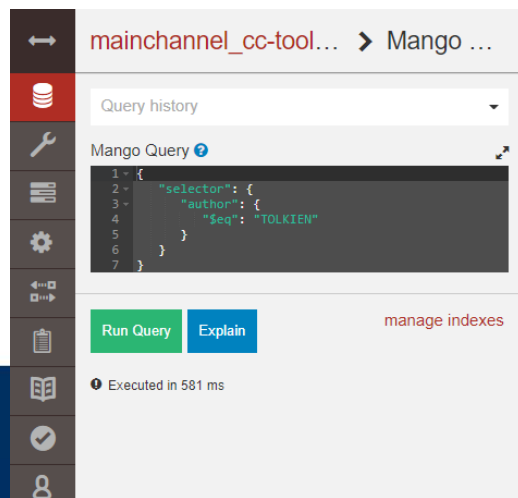
[http://localhost:5984/\\_utils/#login](http://localhost:5984/_utils/#login)



# Usando a interface do CouchDB

O base do *state* é identificada pelo nome do *channel* junto com o nome do *chaincode*.

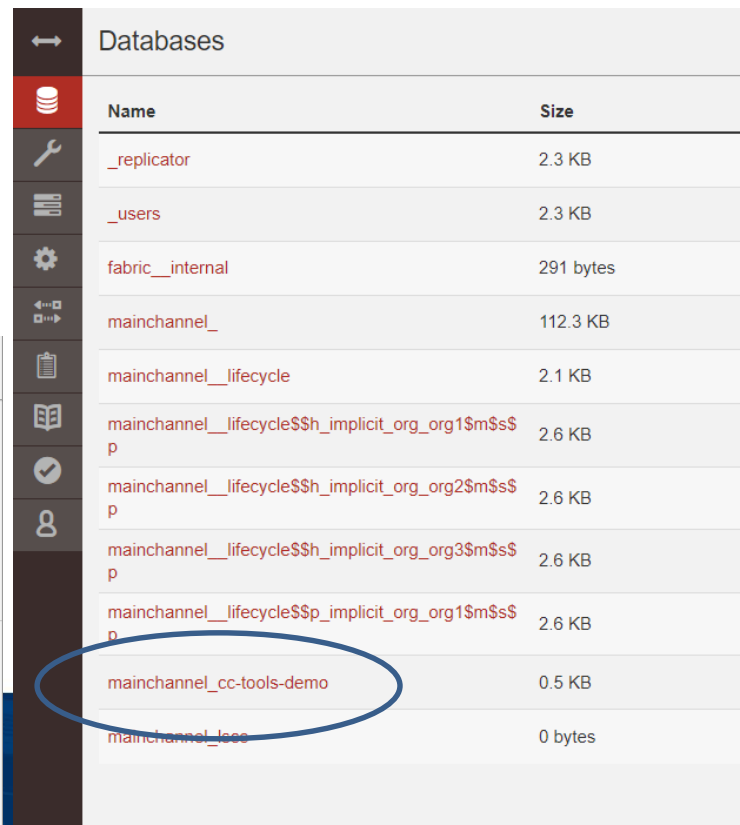
Queries podem ser executadas da interface através do item *Run Query with Mango*



The screenshot shows the Mango query interface. The breadcrumb path is 'mainchannel\_cc-tool... > Mango ...'. The 'Query history' dropdown is open. The 'Mango Query' section contains a query:

```
1 {
2   "selector": {
3     "author": {
4       "$eq": "TOLKIEN"
5     }
6   }
7 }
```

Below the query, there are buttons for 'Run Query' (green) and 'Explain' (blue). To the right of these buttons is a link 'manage indexes'. Below the buttons, it says 'Executed in 581 ms'.



The screenshot shows the 'Databases' interface. It lists several databases with their names and sizes. The database 'mainchannel\_cc-tools-demo' is circled in blue.

Name	Size
_replicator	2.3 KB
_users	2.3 KB
fabric__internal	291 bytes
mainchannel__	112.3 KB
mainchannel__lifecycle	2.1 KB
mainchannel__lifecycle\$\$h_implicit_org_org1\$m\$\$p	2.6 KB
mainchannel__lifecycle\$\$h_implicit_org_org2\$m\$\$p	2.6 KB
mainchannel__lifecycle\$\$h_implicit_org_org3\$m\$\$p	2.6 KB
mainchannel__lifecycle\$\$p_implicit_org_org1\$m\$\$p	2.6 KB
mainchannel_cc-tools-demo	0.5 KB
mainchannel__logs	0 bytes

# Tabela de Query CouchDb

Queries podem ser realizadas através de uma sintaxe. Maiores detalhes na documentação oficial do CouchDb

<https://docs.couchdb.org/en/3.2.2-docs/api/database/find.html>

Exemplo:

Busca documentos com status "draft"

```
"selector": {  
  "status": { "$eq": "draft" }  
}
```

*Busca asset com year maior ou igual a 2020*

```
"selector": {  
  "year": { "$gte": 1900 }  
}
```

# Função *Search*

Função CC-Tools utiliza a função *Search* do pacote *assets* para realizar busca no *CouchDb*

***func Search(stub \*sw.StubWrapper, request map[string]interface{}, privateCollection string, resolve bool) (\*SearchResponse, errors.ICCError)***

*request* – selector da query

*privateDataCollection* – pesquisa na base do PVC

*resolve* – resolve as referências dentro do asset.

Exemplo:

```
query := map[string]interface{}{
    "selector": map[string]interface{}{
        "@assetType": "book",
        "author":    authorName,
    },
}
```

```
response, err := assets.Search(stub, query, "", true)
```

# Utilizando funções de pesquisa

O Hyperledger Fabric possui diversas funções para realizar queries dentro do chaincode. Essas funções trabalham com iterators dentro de loops.

Exemplos:

*GetQueryResult*

*GetQueryResultWithPagination*

*GetStateByPartialCompositeKey*

*GetHistoryForKey*

Exemplo:

```
resultsIterator, _ = stub.GetQueryResult(queryString)
for resultsIterator.HasNext() {
    queryResponse, err := resultsIterator.Next()
    ...
}
```

...

# Private Data Collections

A biblioteca CC-Tools possui mecanismos para trabalhar com PDCs.

Assets que terão as informações gravadas em PDCs devem ter o operador *Readers*.



# Exemplo de um asset em PDC

```
var Secret = assets.AssetType{
    Tag:      "secret",
    Label:    "Secret",
    Description: "Secret between Org2 and Org3",
    Readers: []string{"org2MSP", "org3MSP"},
    Props: []assets.AssetProp{
        {
            IsKey: true,
            Tag:   "secretName",
            Label: "Secret Name",
            DataType: "string",
            Writers: []string{"org2MSP"}, // This means only org2 can create the asset (org3 can edit)
        },
        {
            Tag: "secret",
            Label: "Secret",
            DataType: "string",
        },
    },
}
```

# Arquivo collections2.json

Cada *asset PDC (Readers)* deve ter um elemento correspondente dentro do arquivo *collections2.json*

```
[  
  {  
    "name": "secret",  
    "requiredPeerCount": 0,  
    "maxPeerCount": 3,  
    "blockToLive": 1000000,  
    "memberOnlyRead": true,  
    "policy": "OR('org2MSP.member', 'org3MSP.member')"  
  }  
]
```



# **Escalabilidad Hyperledger Fabric**

# Operações de escalabilidade

Um rede Hyperledger Fabric é uma rede viva.

Diversas operações de rede podem ser realizadas, por exemplo:

- Adicionar novo peer em uma organização
- Adicionar nova organização em um channel
- Adicionar/atualizar um novo chaincode em um channel.
- Adicionar um novo orderer em uma rede
- Remover um peer de uma organização
- Adicionar novo cliente (web service) de uma organização.

# Adicionando uma organização no fabric-samples

Dentro da pasta *fabric-samples/test-network* subir uma rede com channel *channel1*

```
./network.sh up
```

```
./network.sh createChannel -c channel1
```

Após a criação da rede ir para a pasta *addOrg3*

O script de adicionar org

```
/addOrg3.sh up -c channel1
```

# Adicionar nova org

## Operação AddOrg

1. Criação dos artefatos criptográficos (MSP) na nova organização
2. Configuração do arquivo configtx.yaml para definição da nova org
3. Geração da transação de entrada de nova org no channel
4. Atualização do ultimo bloco de configuração para entrada da nova org no channel
5. Assinatura da transação pelas organizações administradoras do channel de acordo com a Policy de Admin
6. Aplicação do novo bloco de configuração no channel
7. Subir os containers dos peers da nova org
8. Pedido de entrada da nova org no channel.
9. Channel atualizado.

# configtx.yaml da nova org

Organizations:

- &Org3

Name: Org3MSP

ID: Org3MSP

MSPDir: ../organizations/peerOrganizations/org3.example.com/msp

Policies:

Readers:

Type: Signature

Rule: "OR('Org3MSP.admin', 'Org3MSP.peer', 'Org3MSP.client')"

Writers:

Type: Signature

Rule: "OR('Org3MSP.admin', 'Org3MSP.client')"

Admins:

Type: Signature

Rule: "OR('Org3MSP.admin')"

Endorsement:

Type: Signature

Rule: "OR('Org3MSP.peer')"

# Geração de MSP e definição da nova organização

Geração dos artefatos criptográficos da nova organização

```
cryptogen generate --config=org3-crypto.yaml --output="../organizations"
```

Geração da definição da nova organização

```
configtxgen -printOrg Org3MSP > ../organizations/peerOrganizations/org3.example.com/org3.json
```

Subindo o novo peer da nova organização

```
docker-compose -f docker/docker-compose-org3.yaml up -d
```



# Atualizando o bloco de configuração

Recuperar o ultimo bloco de configuração do channel através de um peer de uma org que já esteja no peer.

```
peer channel fetch config channel-artifacts/config_block.pb -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com -c channel1 --tls --cafile "${PWD}/organizations/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem"
```

O bloco de configuração é gravado como protobuf (binário)

Para atualizar o bloco de configuração deve-se usar a comando configtxlator transformar os dados para JSON e de volta para protobuf

O novo bloco é trabalhado externamente para poder conceder a entrada da nova org no channel.

# Aplicando o novo bloco de configuração no channel

O novo bloco gerado precisa ser assinado por todas as organizações definidas na configuração do channel.

```
setGlobals $ORG
```

```
peer channel signconfigtx -f channel-artifacts/org3_update_in_envelope.pb
```

A atualização é aplicada ao *channel*

```
peer channel update -f channel-artifacts/org3_update_in_envelope.pb -c channel1 -o localhost:7050 --ordererTLSHostnameOverride  
orderer.example.com --tls --cafile  
"${PWD}"/organizations/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem"
```

# Pedido de entrada no channel pela nova org

Nova org deve obter o *genesis block* para pedir entrada

```
peer channel fetch 0 channel-artifacts/channel1.block -o localhost:7050 --ordererTLSHostnameOverride orderer.example.com -c channel1 --tls --cafile "${PWD}/organizations/ordererOrganizations/example.com/orderers/orderer.example.com/msp/tlscacerts/tlsca.example.com-cert.pem"
```

Realizar operação de *joinPeer*

```
peer channel join -b channel-artifacts/channel1.block
```

Após a operação bem sucedida a nova org se encontra definitivamente no *channel*



# **Orquestradores Blockchain**

# Orquestradores Hyperledger Fabric

Estão disponíveis diversos orquestrados para facilitar a operação em redes Hyperledger Fabric

- Orquestradores open source
- Orquestradores em nuvem
- Orquestradores licenciados

# Orquestradores open-source

Disponibilizados pela Hyperledger Foundation

Hyperledger Cello

<https://www.hyperledger.org/use/cello>

Hyperledger Bevel

<https://www.hyperledger.org/use/bevel>

Fabric Operator (Hyperledger Labs)

<https://labs.hyperledger.org/labs/hlf-operator.html>

# Orquestradores em nuvem e licenciados

Disponíveis na nuvens públicas

Amazon Managed Blockchain

<https://aws.amazon.com/pt/managed-blockchain/>

IBM Blockchain Platform

<https://www.ibm.com/products/blockchain-platform-hyperledger-fabric>

Oracle Blockchain Platform Service

<https://www.oracle.com/br/blockchain/cloud-platform/>

Binario Cloud Blockchain (GoFabric engine)

<https://binario.cloud/produtos/blockchain>

# GoFabric

Orquestrador disponibilizado pela empresa GoLedger

<https://gofabric.io/>

- Lista de redes Hyperledger Fabric
- Dashboard de rede
- Gestão multichannel
- Operações de escalabilidade da rede (*addPeer*, *AddOrg*, etc)
- Integração automática com chaincodes desenvolvidos com a biblioteca CC-Tools
- Atualização de chaincodes
- Atualização de APIs
- Mapeamento de dados dos chaincodes utilizando *Templates* de dados




# GoFabric

Orquestrador disponibilizado pela empresa GoLedger

<https://gofabric.io/>

- Lista de redes Hyperledger Fabric
- Dashboard de rede
- Gestão multichannel
- Operações de escalabilidade da rede (*addPeer*, *AddOrg*, etc)
- Integração automática com chaincodes desenvolvidos com a biblioteca CC-Tools
- Atualização de chaincodes
- Atualização de APIs
- Mapeamento de dados dos chaincodes utilizando *Templates* de dados

# GoFabric – Dashboard

GOFABRIC

DASHBOARD

NETWORK

NODES

CHAINCODES

API

SETUP MACHINES

?

🔔

🌐

MSARRES

M

GoFabric networks

✓ Networks loaded

digital-certificate-id-

cert-channel

id-channel

+ New network

DIGITAL-CERTIFICATE-ID-

CHANNELS

cert-channel

Orgs3Peers4Orderers5

id-channel

Orgs3Peers4Orderers5

ORGANIZATIONS

international-student-id.org

Peers2Orderers1APIs2

Joined Channels

- cert-channel
- id-channel

social-admin.gov

Peers1Orderers1APIs1

Joined Channels

- id-channel

## CERT-CHANNEL

4

3

5

FABRIC VERSION

ACTIVE PEERS

ORGS

ORDERERS

2.2

CHAINCODE

VERSION

certificates

1.0

international-student-id.org

ca: 45.225.25.191

• orderer0: 45.225.25.191

• peer0: 45.225.25.191

• peer1: 45.225.25.240

ccapi:

none

[45.225.25.240](#)

ENTER GRAFANA

stanford-university.edu

ca: 45.225.25.74

• orderer0: 45.225.25.74

• peer0: 45.225.25.74

ccapi:

[45.225.25.74](#)

ENTER GRAFANA

ucla.edu

ca: 45.225.25.218

• orderer0: 45.225.25.218

• peer0: 45.225.25.218

ccapi:

[45.225.25.218](#)

# GoFabric – Data Mapping Templates

The screenshot displays the GoFabric web application interface. The top navigation bar includes links for DASHBOARD, NETWORK, NODES, CHAINCODES, API, SETUP MACHINES, and a user profile icon labeled 'MSARRES'. The left sidebar shows the 'GoFabric networks' section with a dropdown menu currently set to 'digital-certificate-id-'. Below this, there are links for 'cert-channel', 'id-channel', and a '+ New network' button. The main content area is titled 'TEMPLATES' and is divided into two panels: 'TEMPLATE LIST' and 'EDIT TEMPLATE'.

**TEMPLATE LIST**

Template Name	Edit	Delete
certificateTemplate		
certificates		
idTemplate		

**EDIT TEMPLATE**

**Name**  
idTemplate

**Description**

**Assets** CREATE NEW

Label	Tag	
Passport	passport	

**Fields**

Label	Tag	Is key	Required	Read only	
Passport ID	id	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Name	name	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

**Data type**  
String

**Dropdown menu:**

- Number
- String
- Boolean

**Field types:**

- List of Number
- List of String
- List of Boolean

# GoFabric – deployment na nuvem



## ORGANIZATIONS

international-student-id × social-admin × stanford-university × state-department × >

ADD MORE

ORGANIZATION BASIC INFO

Organization name: international-student-id Domain name: org

CA INFO

CA address: 45.225.25.191 CA user: admin

IP or domain address for CA: You must save this

CA password: Repeat password

You must save this Must be equal to CA password

INSTANCES

Machine configuration

Instance address: +

☒ Peer ☐ Orderer

Instance address: 45.225.25.191 ✓ Peer ✓ Orderer

Instance address: 45.225.25.240 ✓ Peer

New EC2 Experience Tell us what you think

EC2 Dashboard

EC2 Global View

Events

Tags

Limits

▼ Instances

Instances New

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances New

Dedicated Hosts

Capacity Reservations

▼ Images

AMIs New

Instances (3) info

Search

Instance state = running X Clear filters

<input type="checkbox"/>	Name	Instance ID	Instance state	Public IPv4 ...
<input type="checkbox"/>	-	i-0009f0a932eddf9c5	Running	3.129.44.196
<input type="checkbox"/>	goprocess-audit	i-02cdb10f594c407be	Running	3.135.62.89
<input type="checkbox"/>	goprocess-regi...	i-077650e507dbb8c94	Running	18.221.179.150

Select an instance

# GoFabric – multichannel

**GoFabric** DASHBOARD NETWORK NODES CHAINCODES API SETUP MACHINES ? MSARRES M

GoFabric networks  
✓ Networks loaded

digital-certificate-id- ▾

cert-channel

id-channel

+ New network

Define Organizations ✓ 2 Define Channels 3 Define Chaincodes 4 Start network

CHANNELS

cert-channel x id-channel x

ADD CHANNEL

CHANNEL BASIC INFO

Channel name  
cert-channel


Peers Available


- international-student-id
- social-admin
- 45.225.25.120 +
- stanford-university
- state-department


Peers in Channel


- international-student-id
- 45.225.25.191 ✗
- 45.225.25.240 ✗
- stanford-university
- 45.225.25.74 ✗


# GoFabric – Chaincode marketplace


 **GOFABRIC**


 GoFabric networks  
✓ Networks loaded



 digital-certificate-id- ▾

 cert-channel

 id-channel

 New network

cert-channel

id-channel

certificates ×

ADD CHAINCODE

Chaincode name

certificates

TEMPLATE

CLOUD CHAINCODES

CHAINCODE FILE

Select the template chaincode ⓘ

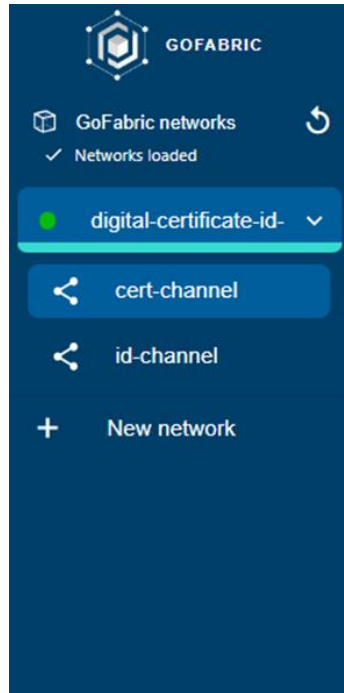
GOSHARE

Select the user template chaincode ⓘ

GOBIO

GOPROCESS

# GoFabric – Private Chaincode



## PERMISSIONS

**i** If no permissions are set, every organization in the network will be able to read and write to every asset

Passport

Asset

international-student-id, stanford-university

☒ Private data **i**

☒ international-student-id

☐ social-admin

☒ stanford-university

☐ state-department

☐ ucla


Private data

Passport ID string

Prop

All enabled

# GoFabric – Atualizar Chaincode

GOFABRIC

DASHBOARD

NETWORK

NODES

CHAINCODES

API

SETUP MACHINES

GoFabric networks

✓ Networks loaded

digital-certificate-id-

cert-channel

id-channel

+ New network

UPGRADE CHAINCODE

certificates

Chaincode version

2.0

☒ AUTO VERSION UPGRADE

CHANNEL PEERS UPGRADE

international-student-id

☒ peer0.international-student-id.org - 45.225.25.191

☒ peer1.international-student-id.org - 45.225.25.240

stanford-university

Select the template chaincode

GOSHARE

Select the user template chaincode

GOBIO

GOPROCESS

Add your template

certificates

PERMISSIONS

If no permissions are set, every organization in the network will be able to read and write to every asset

Document Hash

All

Asset

Course

All

Private data

Private



# Tarefa

