Develop & Deploy Hyperledger Fabric Apps Using KrypCore

Hyperledger Fabric Layer 2 Platform

Key Features

Quick & Easy

Multiple Deployment Options

Cost Management & Optimization

Operation / Governance / Security

Maintenance

Quick & Easy Hyperledger Fabric Starter

Interactive GUI driven feature for

Network Creation

Network interaction

Chaincode & Channel Management

Operation & Administration

Identity Management - Wallet

No Prerequisites:

Docker, Docker Compose YAML

Fabric Tools , Fabric Commands

Linux, Shell

Expected Skill Set

Chaincode Coding

Rest API – Consume using Preferred Programming Language

Knowledge of Endorsement Policies

Simplified & Focused

Focus on use case design, development and logic validation

Abstracts Hyperledger Fabric administration, configuration

Knowledge of Rest API is enough to perform transaction

Fabric SDK’s experience not required

Simplified Report Generation

Notification Service provides easy alerts and messages integration

Deploy On

Virtual Machines

Docker Supported

Linux, Windows

For development

For production

Managed Kubernetes Clusters

Azure AKS

AWS EKS

Private Clouds

Deploy For

Development

- Design use case
- Perform Integrations
- Deploy Chain Codes

POC / Pilot

- Validate

Production

- Manage & Scale

Deploy Via

- GUI – Super Easy
  - Simple & Interactive. Configure cloud resources & execute
  - Ideal for anyone – Cloud Skills not required
- Terraform, Helm Charts, Scripts – Power Users
  - Powerful, seamless deployment via commands and automation frameworks
  - Ideal for enterprises, devops, secOps engineer
- ARM Templates – Azure Ready
  - Simple & Interactive. Seamless Azure native deployment
  - Ideal for Azure ARM

Cost Optimized Blockchain

- Project Execution shortened by 1/3rd time & 1/3rd Cost
- Forecast Infrastructure Upgrade –Using data lake size analysis
- Reduced Integrations: Leverage on API & Email Notifications – Pub Sub
- Reduced report development time frame : Using data lake report builder
- Reduced Administration Charges: Fabric Administration simplified - cutting down the deep learning curve of Hyperledger Fabric and its nuances
- Low Code Option – No chaincode development Required
- Optimized cloud resource usage with tried and tested configurations provides lower consumption fees.

Operation/Governance/Security

Stratified user management where each strata has pre-defined privileges within the network and the application.

User blocks with limited permissions in blockchain operations.

Segregation of deployments at Kubernetes level via namespaces.

Certificates and Keys managed using tried and tested cloud solutions for key management.

Maintenance

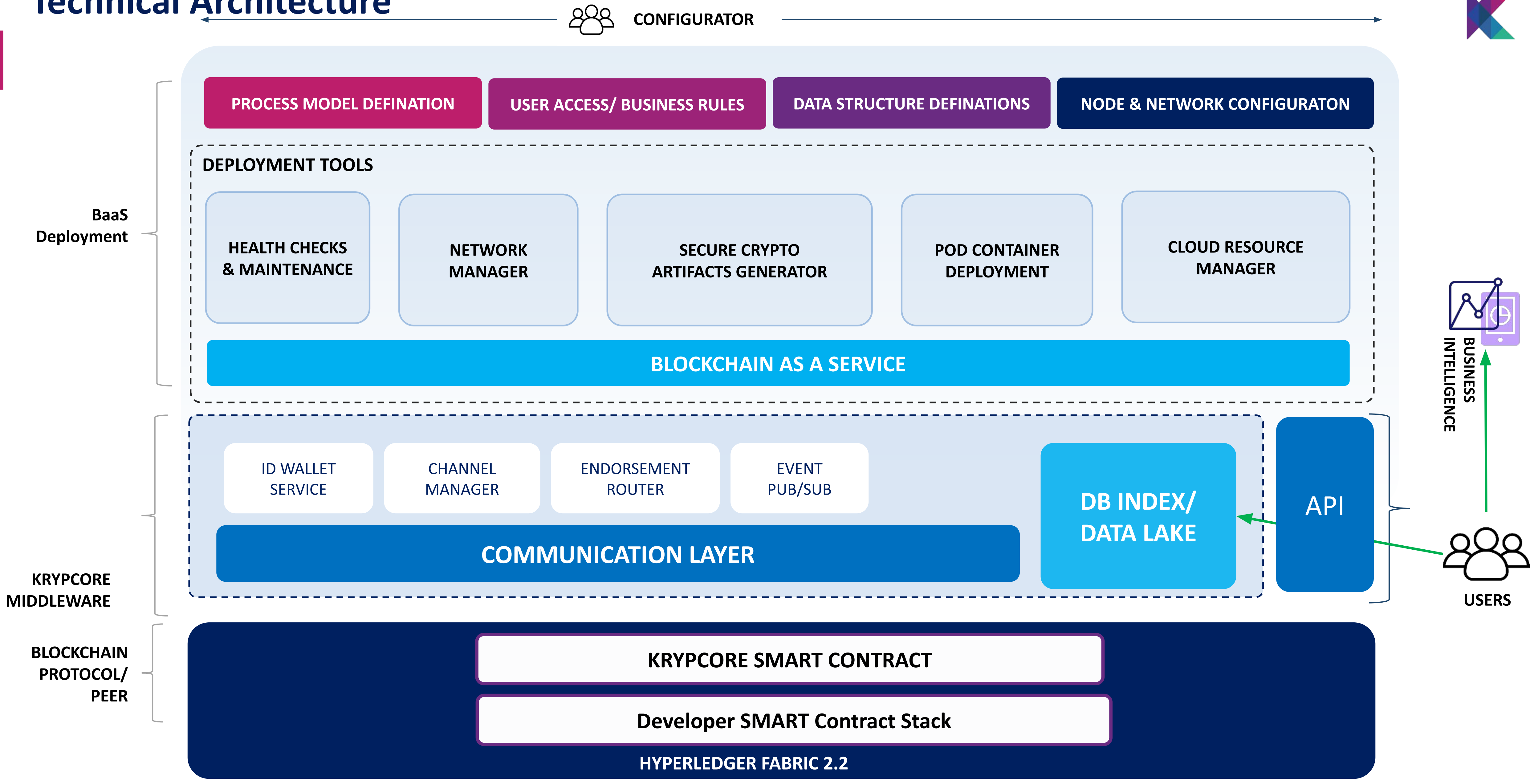
- Easy maintenance of production systems with top-of-the-line open-source monitoring tools leveraging Hyperledger Metrics.
  - Prometheus – Grafana
- Alerts set for blockchain data and activity.
- Alert set for Kubernetes dashboard for cluster resource monitoring.
  - Kubernetes Dashboard
- Alerts set of cloud infrastructure monitoring for costing and control of resources.
- Log Monitoring and Analytics.
  - Azure Log Analytics Workbook
  - Elasticsearch-FluentD-Kibana

End of Presentation

Thank You

Support Slides

Technical Architecture



CONFIGURATOR

BaaS Deployment

PROCESS MODEL DEFINATION

USER ACCESS/ BUSINESS RULES

DATA STRUCTURE DEFINATIONS

NODE & NETWORK CONFIGURATON

DEPLOYMENT TOOLS

HEALTH CHECKS & MAINTENANCE

NETWORK MANAGER

SECURE CRYPTO ARTIFACTS GENERATOR

POD CONTAINER DEPLOYMENT

CLOUD RESOURCE MANAGER

BLOCKCHAIN AS A SERVICE

KRYPCORE MIDDLEWARE

ID WALLET SERVICE

CHANNEL MANAGER

ENDORSEMENT ROUTER

EVENT PUB/SUB

COMMUNICATION LAYER

DB INDEX/ DATA LAKE

API

BUSINESS INTELLIGENCE

USERS

BLOCKCHAIN PROTOCOL/ PEER

KRYPCORE SMART CONTRACT

Developer SMART Contract Stack

HYPERLEDGER FABRIC 2.2

KrypCore vs. Bespoke

Development Comparison

BLOCKCHAIN SETUP

Set up, Docker
Docker Compose
Golang | Node JS, IDE
Download and install Hyperledger using docker image file
Fabric CA setup

CONFIGURATION

Configure Fabric SDK to interact with your application
Selection and configuration of IDE based on your business requirement

APPLICATION CONFIGURATION

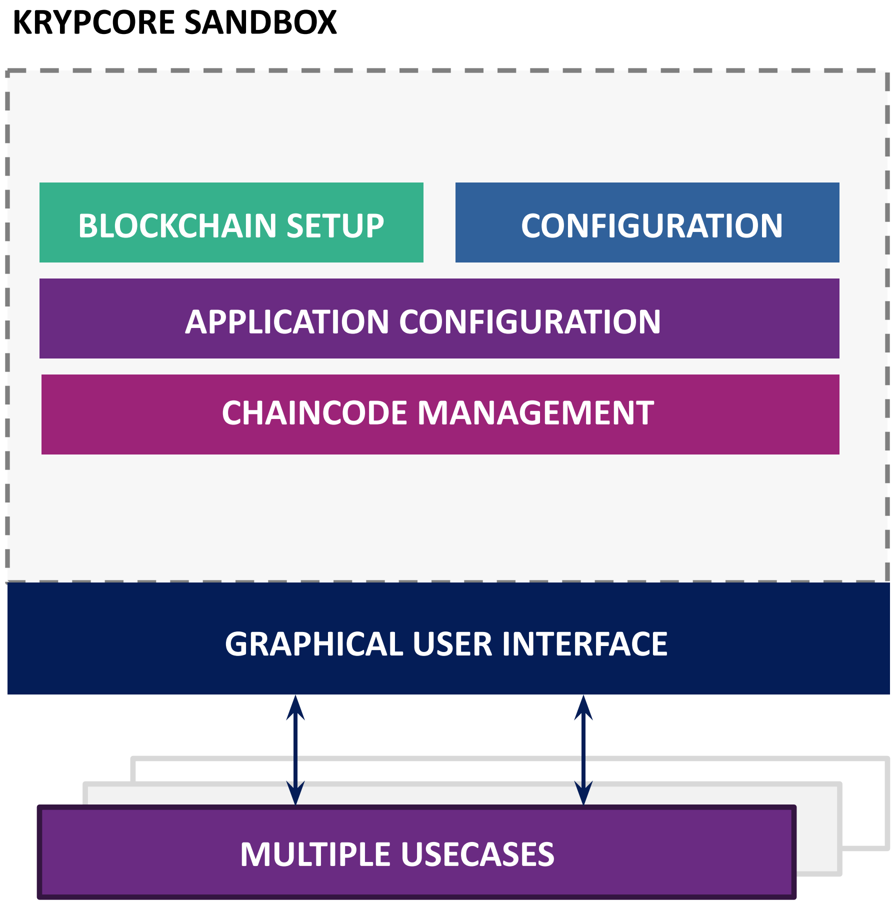
Configure PEER, ORDERER, CA, ORGANIZATION and CHANNELS

CHAINCODE MANAGEMENT

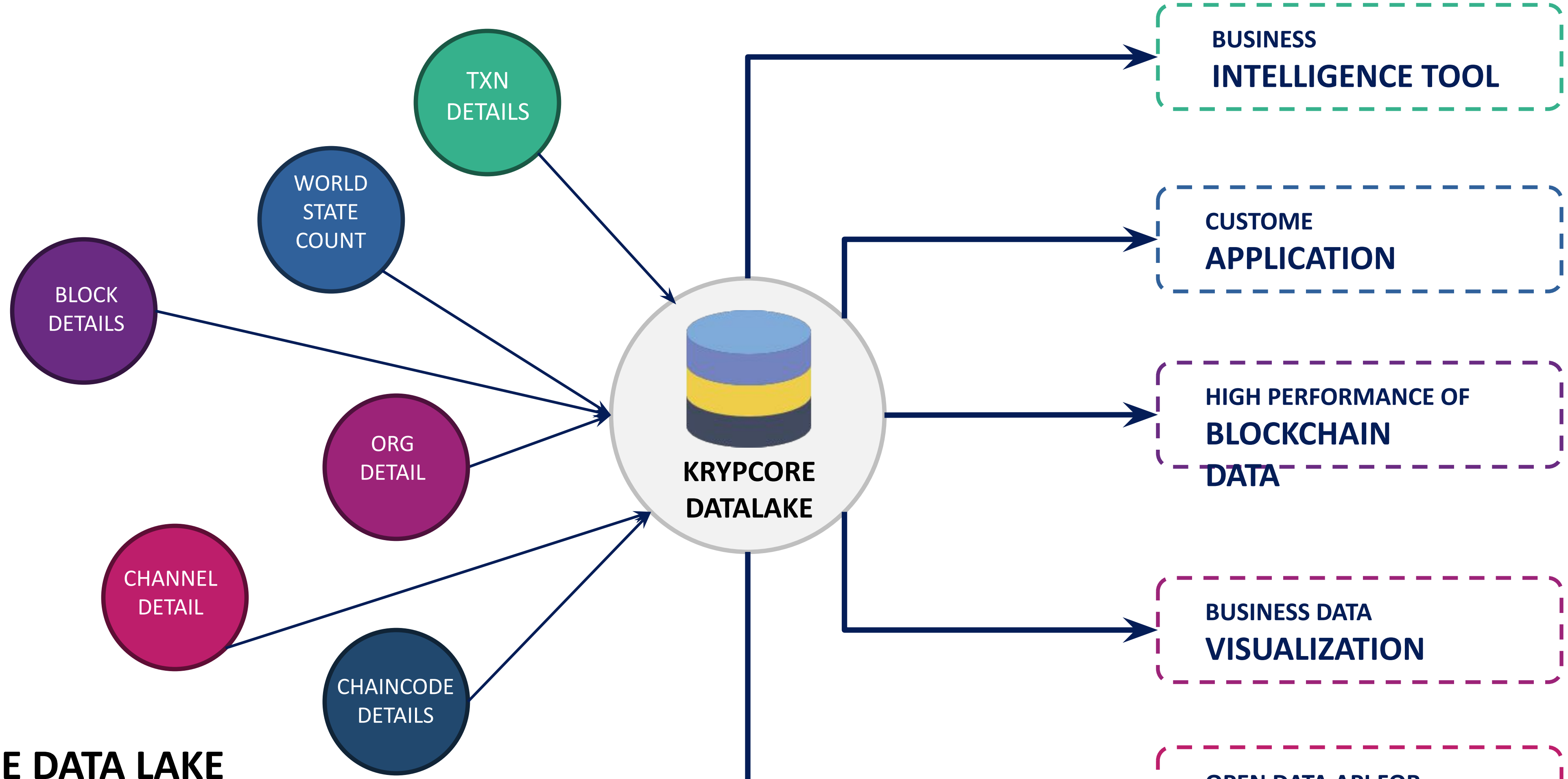
Install, Instantiate - Business logic coding
Visualization of chaincode activity, post analysis of transaction of blockchain

USER INTERFACE

Custom user interface to manage and interact the application



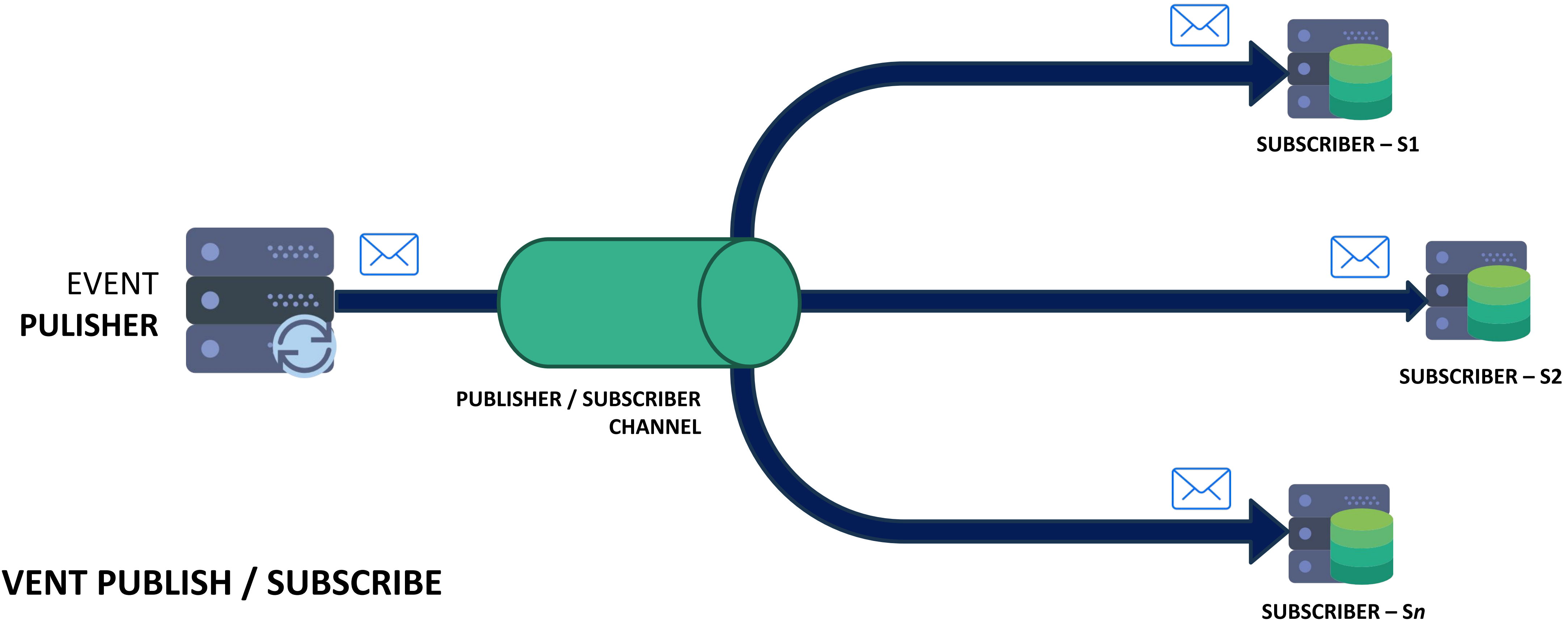
Data Management



KRYPCORE DATA LAKE

Enables high performance searches of the Blockchain data.
Component facilitates to configure business data that needs to be indexed.
Block data will be synchronized periodically with index database.
As part of this process business data will be extracted from the transaction payload and maintained in database to support BI tool.

Notification Services



EVENT PUBLISH / SUBSCRIBE

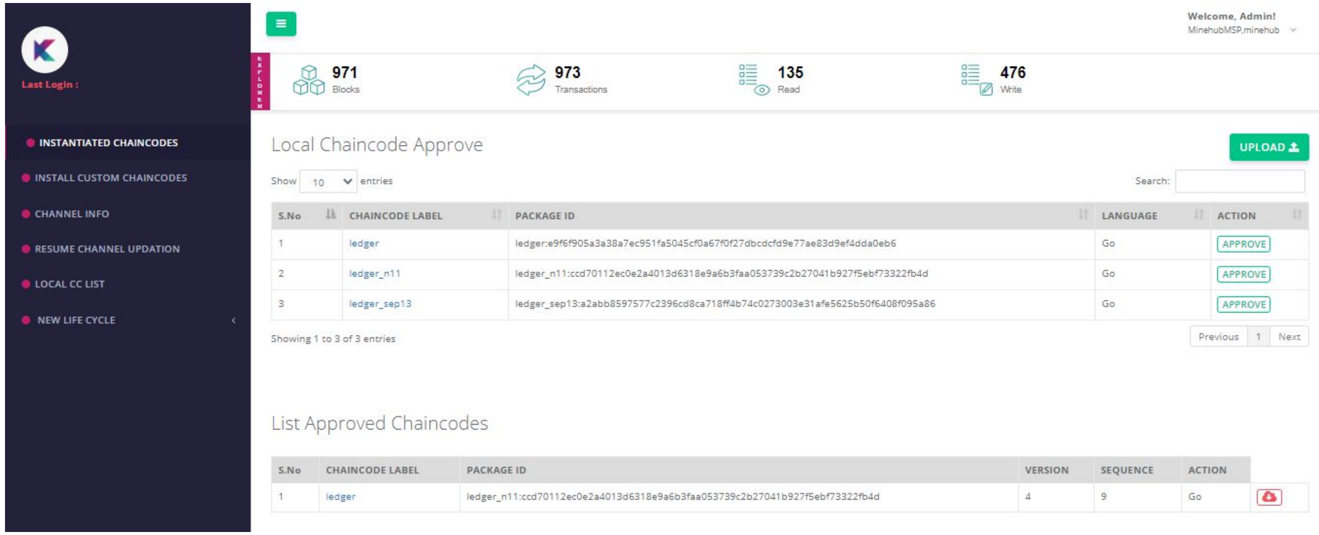
Notification for participants upon occurrence of an event/condition in the Blockchain network allows controlling with ease, the event sequences enabling better notification management for users.

Chaincode Life-Cycle Management

Manage chaincode lifecycle methods

Chaincode Lifecycle Management

- Chaincode Packaging
    - Supports GoLang / JS / Java Chaincode Packaging
    - Package supported for 1.4 and 2.2 formats
- Installing on Peers
- Approve, Committing
    - Perform approvals and commit of new versions
- View Configuration
    - View existing configuration of the chaincodes
- Define PDC Config
    - Defined PDC Collections (Explicit)
- Define Endorsement Policies
    - Define Custom Endorsement Polices



Local Chaincode Approve

Show 10 entries

Search:

S.No	CHAINCODE LABEL	PACKAGE ID	LANGUAGE	ACTION
1	ledger	ledger:e9f6f905a3a38a7ec951fa5045cf0a67f0f27dbcdcfd9e77ae83d9ef4dda0eb6	Go	APPROVE
2	ledger_n11	ledger_n11:ccd70112ec0e2a4013d6318e9a6b3faa053739c2b27041b927f5ebf73322fb4d	Go	APPROVE
3	ledger_sep13	ledger_sep13:a2abb8597577c2396cd8ca718ff4b74c0273003e31afe5625b50f6408f095a86	Go	APPROVE

Showing 1 to 3 of 3 entries

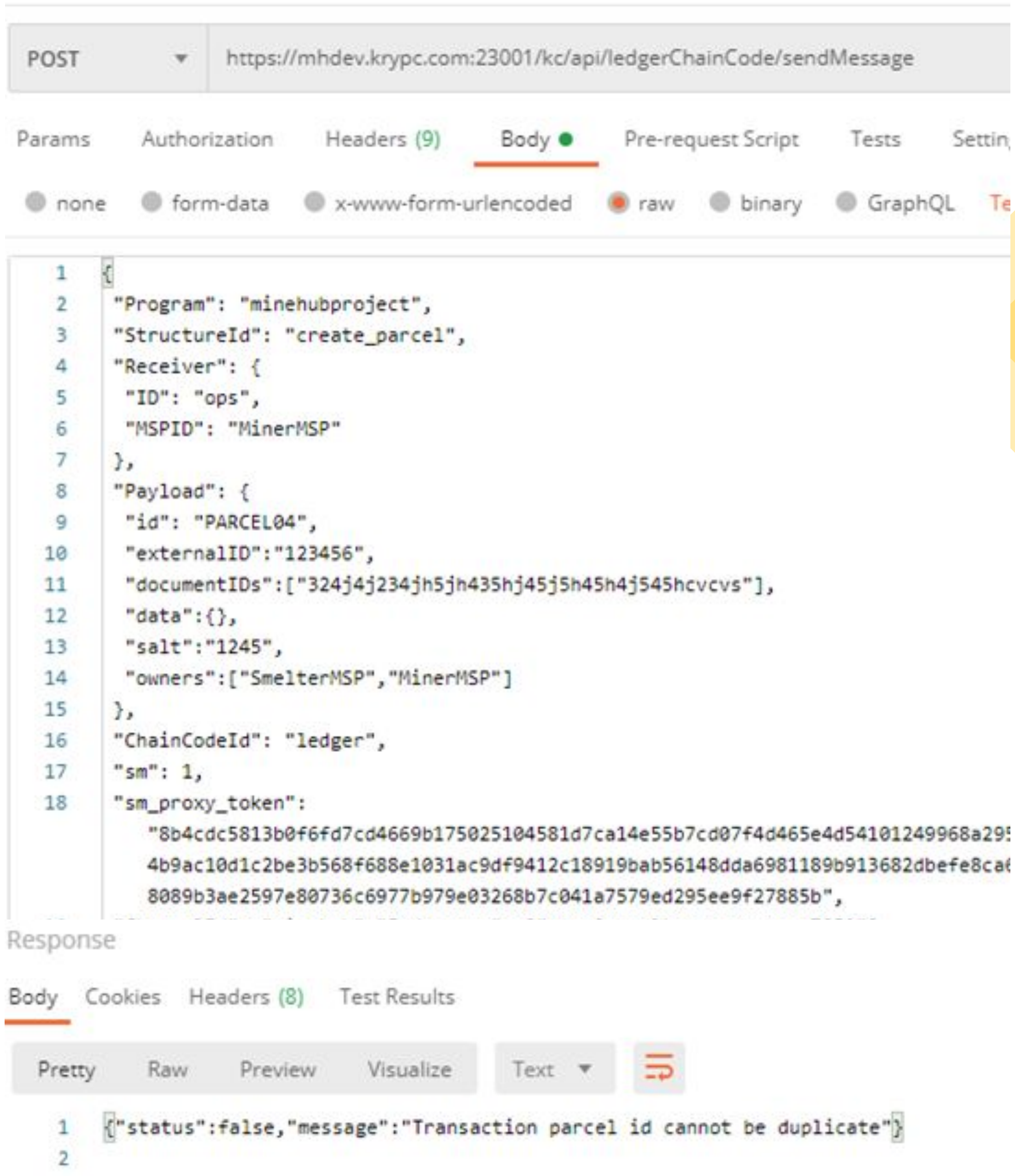
Previous 1 Next

List Approved Chaincodes

S.No	CHAINCODE LABEL	PACKAGE ID	VERSION	SEQUENCE	ACTION	
1	ledger	ledger_n11:ccd70112ec0e2a4013d6318e9a6b3faa053739c2b27041b927f5ebf73322fb4d	4	9	Go	

Transaction & Query API’s

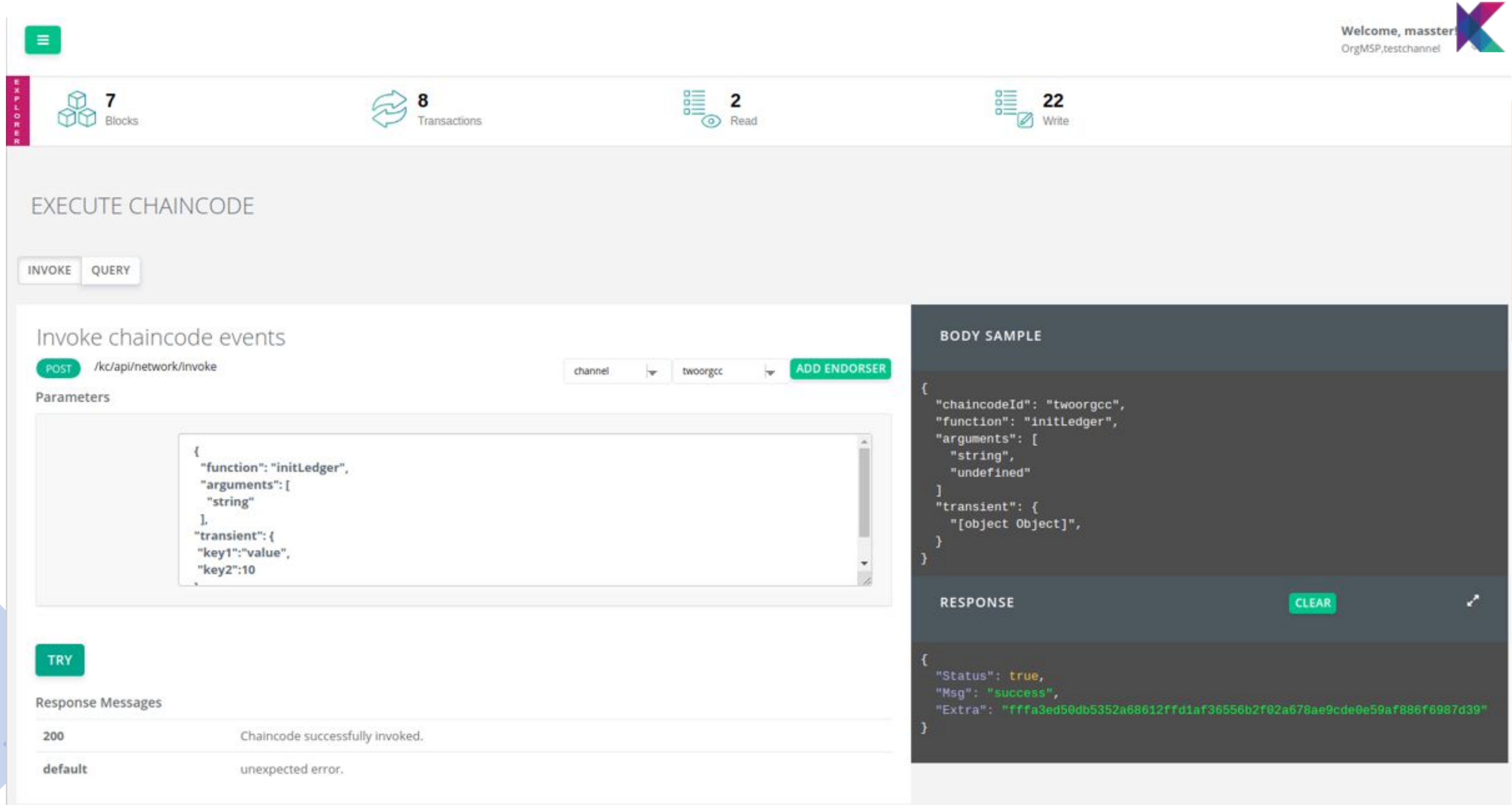
- REST based HTTP API’s for Invoke & Query
- JSON Payload
- Payload Attributes
  - channel name,
  - chaincode name,
  - function name and
  - CC Arguments.
  - Authentication Params
- All arguments are strings
- Optional Attributes:
  - Proxy
  - Endorsers List
- Automatic Endorser Selection is enabled if no endorsers are provided
- API Authentication based on:
  - Wallet Credentials Or
  - Authentication token



POST https://mhdev.krypc.com:23001/kc/api/ledgerChainCode/sendMessage

```
{
 "Program": "minehubproject",
 "StructureId": "create_parcel",
 "Receiver": {
  "ID": "ops",
  "MSPID": "MinerMSP"
 },
 "Payload": {
  "id": "PARCEL04",
  "externalID":"123456",
  "documentIDs":["324j4j234jh5jh435hj45j5h45h4j545hcvcvs"],
  "data":{},
  "salt":"1245",
  "owners":["SmelterMSP","MinerMSP"]
 },
 "ChainCodeId": "ledger",
 "sm": 1,
 "sm_proxy_token":
    "8b4cdc5813b0f6fd7cd4669b175025104581d7ca14e55b7cd07f4d465e4d54101249968a295
    4b9ac10d1c2be3b568f688e1031ac9df9412c18919bab56148dda6981189b913682dbefe8ca
    8089b3ae2597e80736c6977b979e03268b7c041a7579ed295ee9f27885b",
```

```
{"status":false,"message":"Transaction parcel id cannot be duplicate"}
```



Welcome, masster!
OrgMSP,testchannel

7 Blocks

8 Transactions

2 Read

22 Write

EXECUTE CHAINCODE

Invoke chaincode events

POST /kc/api/network/invoke

Parameters

```
{
 "function": "initLedger",
 "arguments": [
  "string"
 ],
"transient": {
 "key1":"value",
 "key2":10
```

BODY SAMPLE

```
{
  "chaincodeId": "twoorgcc",
  "function": "initLedger",
  "arguments": [
    "string",
    "undefined"
  ]
  "transient": {
    "[object Object]",
  }
}
```

RESPONSE

```
{
  "Status": true,
  "Msg": "success",
  "Extra": "fffa3ed50db5352a68612ffd1af36556b2f02a678ae9cde0e59af886f6987d39"
}
```

Response Messages

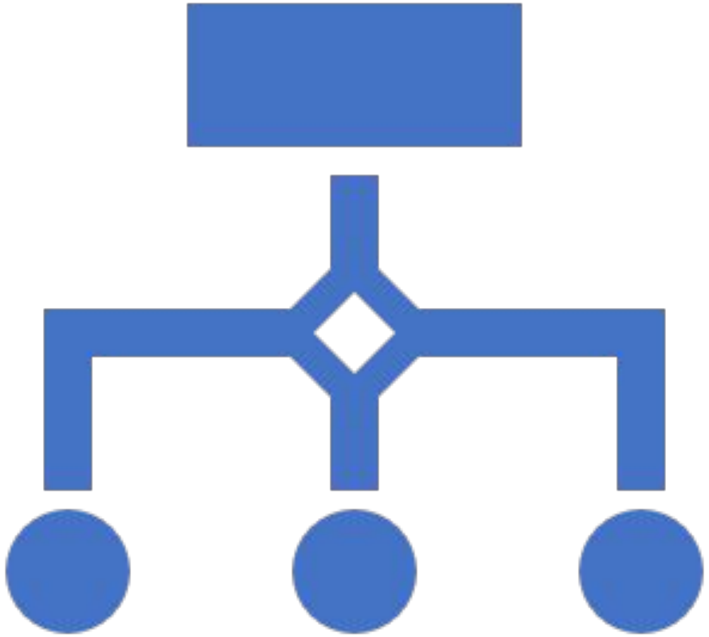
200	Chaincode successfully invoked.
default	unexpected error.

Network Governance

Managing Channel Attributes

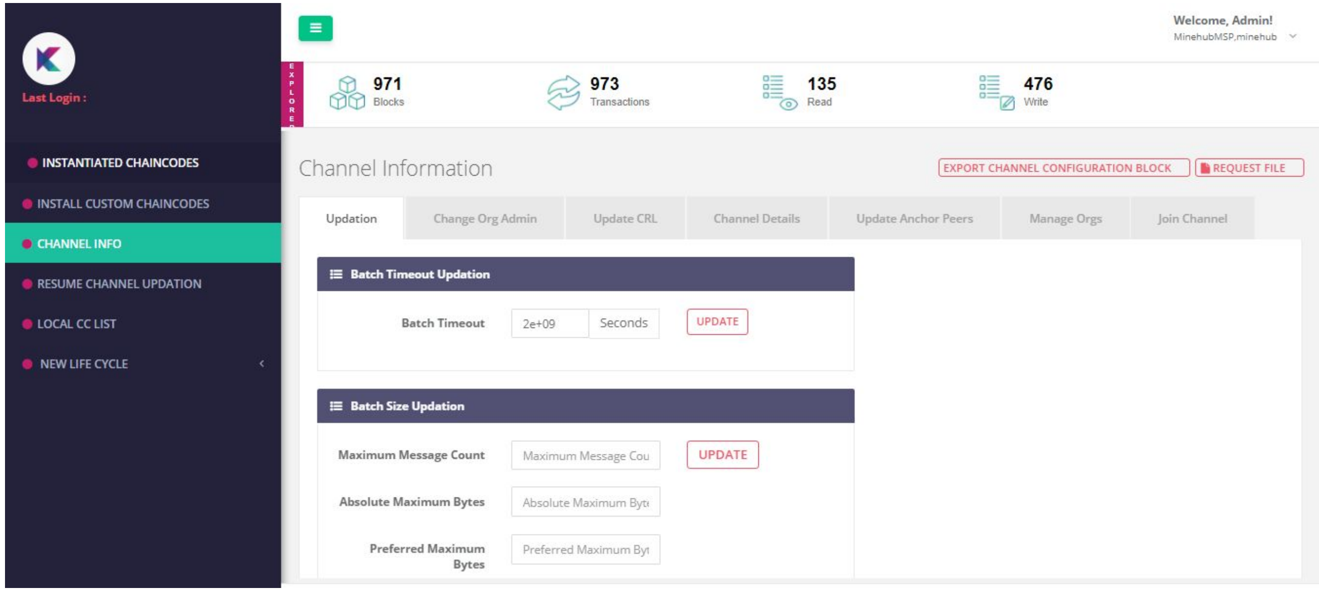
Channel Manager

- Allows Organization to manage the channel parameters in production
- Work-Flow driven UI to assist multiple members to sign and agree on the change
- Onward members can see / analyze the changes in the request file
- Simple changes like batch size to addition of a new organization can be performed
- Can be customized to support different file formats of other providers



Channel Manager

- Modify Anchor Peer List
- Modify Organization Admins
- Update CRL
- Add new channel
- Join new channel
- Change batch size, block time
- Change Policies



Welcome, Admin!
MinehubMSP,minehub

971 Blocks

973 Transactions

135 Read

476 Write

Channel Information

Batch Timeout Updation

Batch Timeout: 2e+09 Seconds

Batch Size Updation

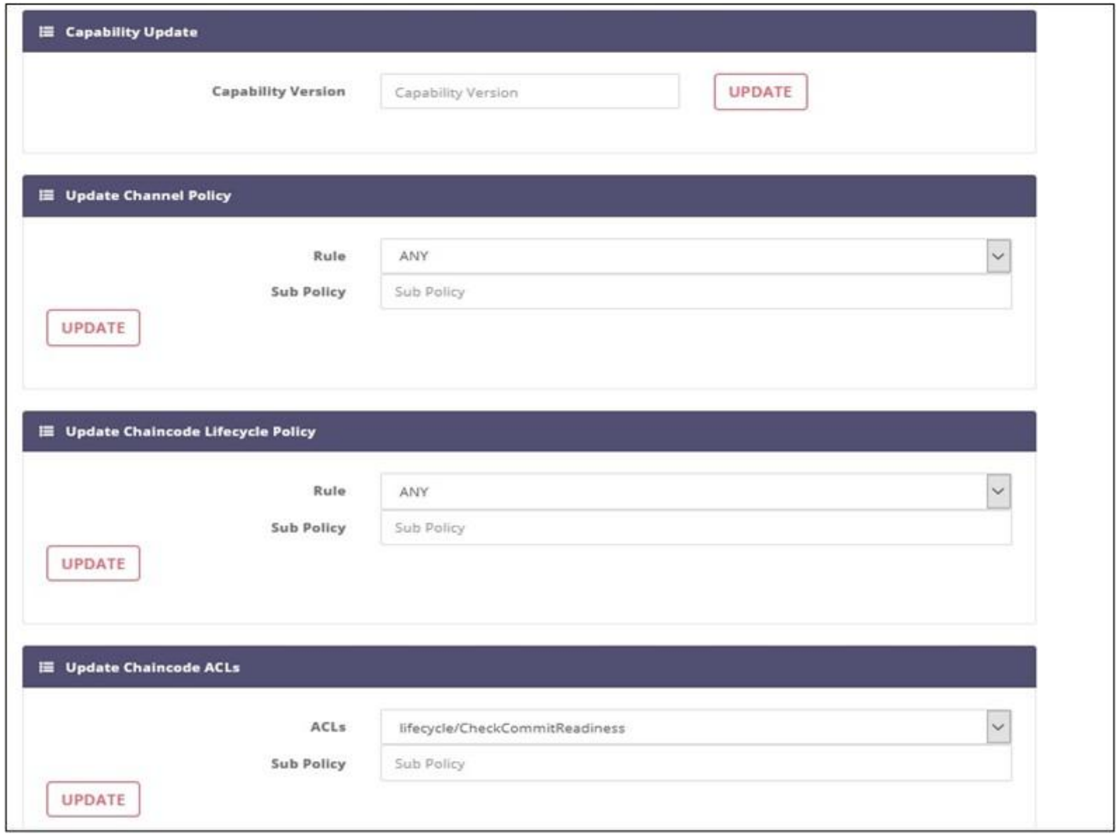
Maximum Message Count: Maximum Message Cou

Absolute Maximum Bytes: Absolute Maximum Byt

Preferred Maximum Bytes: Preferred Maximum Byt

Channel Manager – Upgrading Fabric Version

- Hyperledger version upgrades are activated using the capabilities attributes.
- This must be performed by majority of the members.
- Once done, the features of newer version are activated in the protocol



Capability Update

Capability Version: Capability Version

Update Channel Policy

Rule: ANY

Sub Policy: Sub Policy

Update Chaincode Lifecycle Policy

Rule: ANY

Sub Policy: Sub Policy

Update Chaincode ACLs

ACLs: lifecycle/CheckCommitReadiness

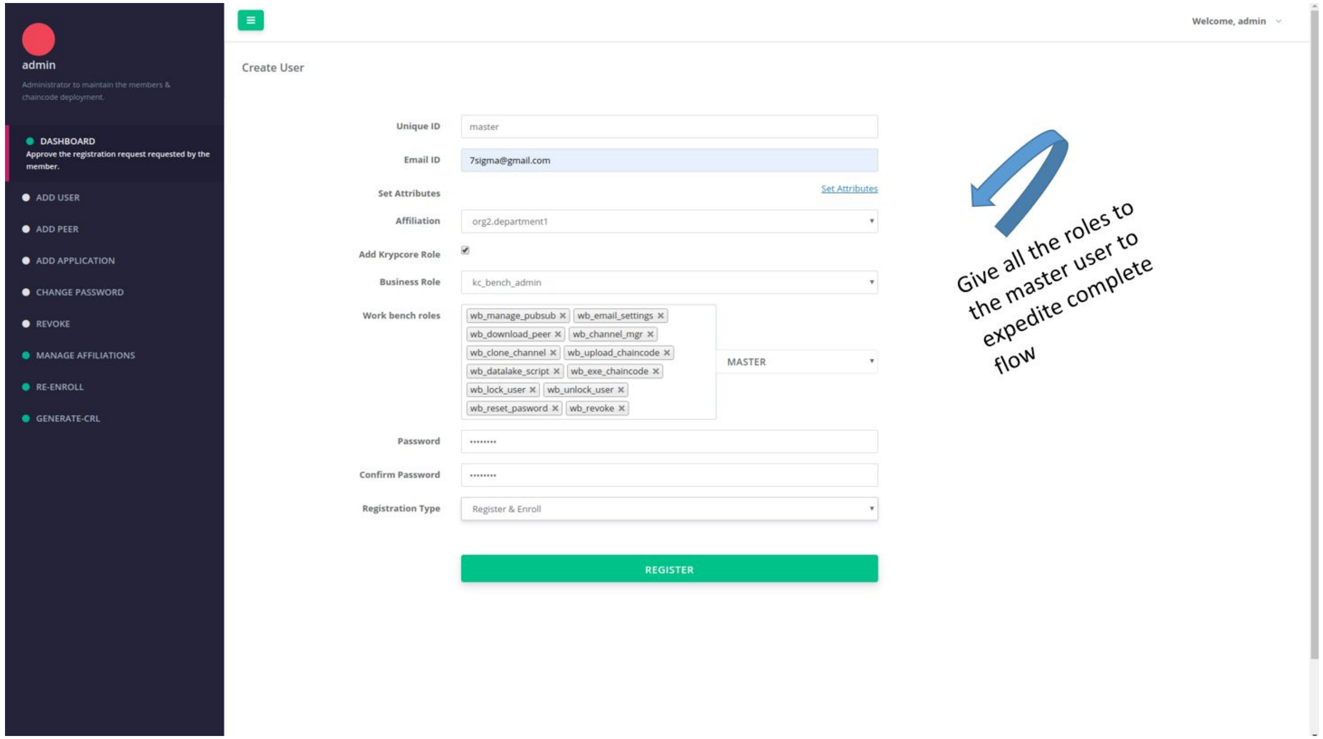
Sub Policy: Sub Policy

Identity Management

Create Identities & Manage lifecycle

Registrar & Workbench Administration

- Each Organization have registrar interface for managing organization user identities
- Identities are created for
  - Performing transactions in blockchain
  - Managing the workbench
  - Network Governance
- Multiple Affiliations can be created and managed
- Revocation & CRL Generation is also available



Create User

Unique ID	master
Email ID	7sigma@gmail.com
Set Attributes	Set Attributes
Affiliation	org2.department1
Add Krypcore Role	☑
Business Role	kc_bench_admin
Work bench roles	wb_manage_pubsub × wb_email_settings × wb_download_peer × wb_channel_mgr × wb_clone_channel × wb_upload_chaincode × wb_datalake_script × wb_exe_chaincode × wb_lock_user × wb_unlock_user × wb_reset_pasword × wb_revoke × MASTER
Password	••••••••
Confirm Password	••••••••
Registration Type	Register & Enroll

Give all the roles to the master user to expedite complete flow

User Wallet

- Each user / application can access the wallet to
- Manage credentials & access tokens
- View transactions
- Download private key and fabric folders for SDK integrations (if enabled)

Integrations, Connectors & Reporting

REST API's	Easy to integrate JSON Payload
Notifications (Pub-Sub)	Email REST End Points
Authentication	Wallet Credentials Token
Data Explorer & Reporting	Visualize data Full state, world state and PDC
Manage Channel	Anchor Peer Block Properties