

Enhancing Usability and Accessibility in ResilientDB

ResilientDB Ecosystem Development and Support



Apache
ResilientDB
Incubating



ExpoLab
Creativity Unfolded

- Gopal Nambiar -

UCDAVIS

Introduction

- Apache ResilientDB (incubating) is a decentralized and secure database designed to ensure transparent, tamper-proof record-keeping.
- High-performance, fault-tolerant blockchain database, now under the Apache incubation umbrella, supporting ongoing development and wider community engagement.
- Our lab's focus is to make Apache ResilientDB easier to use and accessible, enabling more students and developers to engage with blockchain.

Project Goals and Impact

- To improve usability, scalability, and development efficiency within the Apache ResilientDB ecosystem.
- **Goals:**
 - Developed Docker images for over 200 users to simplify Apache ResilientDB deployment.
 - Created CLI tools and APIs to facilitate blockchain interactions.
 - Implemented CI/CD to automate deployments and testing.
- **Impact:** Our work has streamlined teaching and research, allowing students and developers to focus more on experiments than setup.

Docker Images for Apache ResilientDB

- **Problem:** Setting up Apache ResilientDB was complex, often requiring extensive configuration.
- **Solution:** Developed Docker images for Apache ResilientDB to automate and simplify deployment across platforms.
- **Impact:** Around 300 students have used these images, and the download count is still increasing.

Docker Images

expolab / **resdb**

Contains: Image • Last pushed: 10 days ago

☆ 0

↓ 300

🔓 Public

🛡 Scout inactive



expolab/resdb:amd64

INDEX DIGEST sha256:0e9ddcb14ca705458001c262d4c693c3a527dc9334848ca56f792c4c29e0d143

OS/ARCH

linux/amd64

COMPRESSED SIZE ⓘ

975.22 MB

LAST PUSHED

10 days ago by [expolab](#)

TYPE

Image

MANIFEST DIGEST

sha256:f17830ffe...



expolab/resdb:arm64

INDEX DIGEST sha256:458be2f41cffb3853203433394d526e8dc0706fbb0f188b0b83744d77a57f638

OS/ARCH

linux/amd64

COMPRESSED SIZE ⓘ

1.34 GB

LAST PUSHED

a month ago by [expolab](#)

TYPE

Image

MANIFEST DIGEST

sha256:d1124122...

ResContract (Smart Contracts CLI)

- ResContract is a command-line interface to compile, deploy, and manage smart contracts.
- **Problem:** Interacting with smart contracts on Apache ResilientDB was initially challenging without dedicated tooling.
- **Solution:** ResContract provides an easy-to-use CLI for handling smart contracts, removing technical barriers.
- **Core Features:**
 - Compile: Converts Solidity files into deployable code.
 - Deploy: Allows contract deployment on Apache ResilientDB.
 - Manage: Offers commands to interact with deployed contracts.
- **Impact:** Enhanced developer productivity, greater ease in experimenting with smart contracts.

ResContract

```
gopuman@node0:~/ResContract$ rescontract
Usage: rescontract [options] [command]

ResContract CLI - Manage smart contracts in ResilientDB

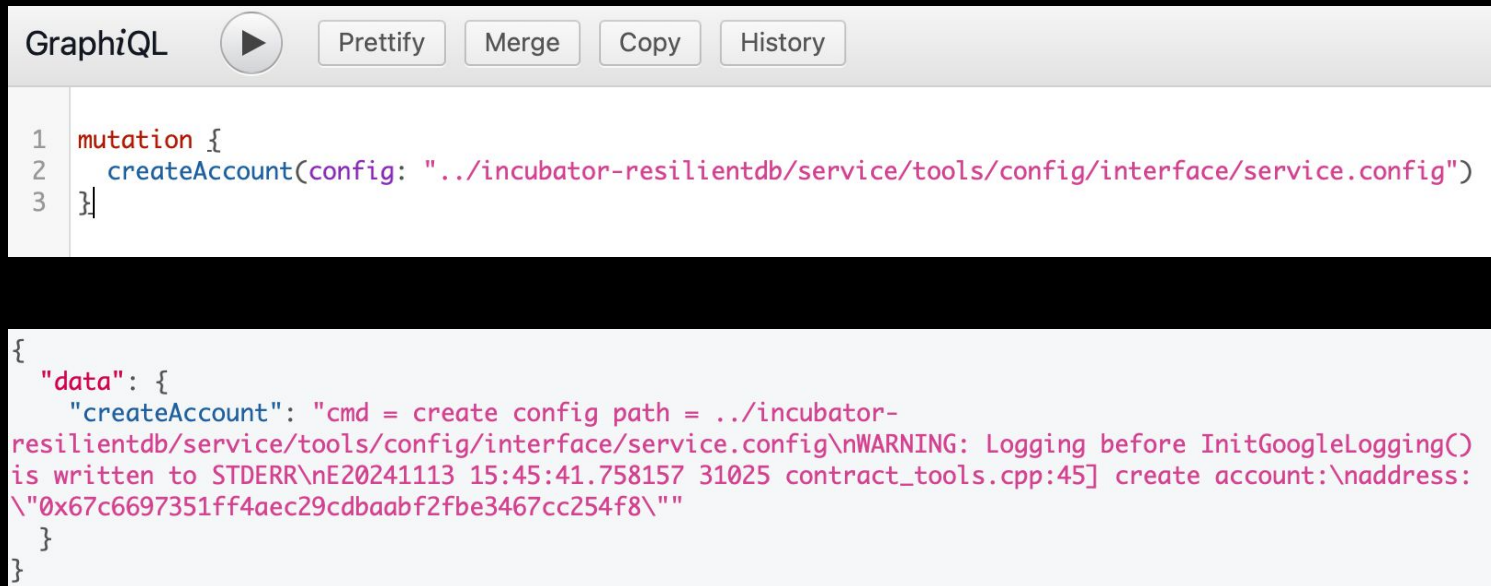
Options:
  -V, --version      output the version number
  -h, --help         display help for command

Commands:
  create [options]   Create a new account
  compile [options]  Compile a .sol file to a .json file
  deploy [options]   Deploy a smart contract
  execute [options]  Execute a smart contract function
  help [command]     display help for command
Usage: rescontract [options] [command]
```

Smart Contracts GraphQL API

- SmartContractsGraphQL API provides a flexible way to query and manage smart contracts.
- **Problem:** Traditional query methods for Apache ResilientDB were limited in flexibility and accessibility.
- **Solution:** The API enables developers to interact with smart contracts seamlessly, integrating easily with front-end applications.
- **Impact:** Simplified access to blockchain data, increased potential for application integration.

Smart Contracts GraphQL API



The screenshot shows the GraphiQL interface. At the top, there is a toolbar with a play button, a 'Prettify' button, a 'Merge' button, a 'Copy' button, and a 'History' button. Below the toolbar, the query editor contains the following GraphQL query:

```
1 mutation {  
2   createAccount(config: "../incubator-resilientdb/service/tools/config/interface/service.config")  
3 }
```

Below the query editor, the JSON response is displayed:

```
{  
  "data": {  
    "createAccount": "cmd = create config path = ../incubator-  
resilientdb/service/tools/config/interface/service.config\nWARNING: Logging before InitGoogleLogging()  
is written to STDERR\nE20241113 15:45:41.758157 31025 contract_tools.cpp:45] create account:\naddress:  
\\\"0x67c6697351ff4aec29cdbaabf2fbe3467cc254f8\\\""  
  }  
}
```

ResDB-ORM Library

- ResDB-ORM is an Object-Relational Mapping (ORM) library to simplify database interactions.
- **Problem:** Interacting directly with Apache ResilientDB's key-value store was time-consuming and error-prone.
- **Solution:** The ORM allows for CRUD operations with minimal code, supporting both version-based and non-version-based functions.
- **Core Features:**
 - Create, Read, Update, Delete (CRUD) operations.
 - Support for both basic and versioned database interactions.
- **Impact:** Streamlined development, easier data management, improved prototyping.

ResDB-ORM Library

```
1 # test.py
2 import requests
3 from resdb_orm.orm import ResDBORM
4
5 db = ResDBORM()
6
7 # Create records
8 data = {"name": "abc", "age": 123}
9 create_response = db.create(data)
10 print("Create Response:", create_response)
11
12 # Retrieve records
13 read_response = db.read(create_response)
14 print("Read Response:", read_response)
15 print("Data:", read_response["data"])
```

```
16
17 # Update records
18 update_response = db.update(create_response, {"name": "def", "age": 456})
19 print("Update Response:", update_response)
20 read_response = db.read(create_response)
21 print("Read Response:", read_response)
22 print("New Data:", read_response["data"])
23
24 # Delete records
25 delete_response = db.delete(create_response)
26 print("Delete Response:", delete_response)
27 read_response = db.read(create_response)
28
```

```
(venv) gopuman@node0:~/ResDB-ORM$ python tests/test.py
Create Response: c8e738a35294112bef1143193f72c9de346ce11fbee10361f6dda06befe10b13
Read Response: {'id': 'c8e738a35294112bef1143193f72c9de346ce11fbee10361f6dda06befe10b13', 'data': {'name': 'abc', 'age': 123}}
Data: {'name': 'abc', 'age': 123}
Update Response: {'status': 'update successful'}
Read Response: {'id': 'c8e738a35294112bef1143193f72c9de346ce11fbee10361f6dda06befe10b13', 'data': {'name': 'def', 'age': 456}}
New Data: {'name': 'def', 'age': 456}
Delete Response: {'status': 'delete successful'}
```

CI/CD Support for ResilientEcosystem

- Continuous Integration and Continuous Deployment (CI/CD) automate testing and deployment, improving code quality and efficiency.
- **Problem:** Manual deployment processes made updates slower and more error-prone.
- **Solution:** Implemented CI/CD pipelines for ResilientEcosystem using GitHub Actions automation.
- **Impact:**
 - Faster, more reliable deployments.
 - Improved code quality due to consistent testing.
 - Enabled quicker iterations, fostering a productive development cycle.

CI/CD Examples

test.yml

on: push

✓ build-and-test 16m 4s

npm-publish.yml

on: push

✓ publish-npm 7s

ci.yml

on: push

✓ build 10m 14s

test-publish.yml

on: push

✓ test 17m 49s — ✓ publish 17s

pages-build-deployment


on: dynamic



✓ build 5s — ✓ report-build-status 6s
— ✓ deploy 8s
<https://resilientecosystem.github.io/resili...>


Quickstart Website

- The Quickstart website provides a user-friendly guide for new users to get started with Apache ResilientDB and its associated tools.
- **Problem:** Setting up Apache ResilientDB and related tools required technical knowledge, often challenging for new users.
- **Solution:** The Quickstart website offers a streamlined, guided experience for installation and configuration.
- **Core Features:**
 - Step-by-Step Installation Guides.
 - Code Samples and Usage Examples.
 - Integration Guides.
 - Link to Blog posts.
 - Interactive installation script download

Quickstart Website

 **ResilientDB Quickstart Guide**


  Search

 ResilientEcosystem/resilien...
☆ 2 🗨 1



Home Blog Installation Usage API Reference Contributors About


Home

Home




Welcome to the **ResilientDB Quickstart Guide**—your one-stop shop for all ResilientDB projects and how to get started with them. This documentation provides comprehensive guides to install, use, and contribute to the ResilientDB ecosystem.


 Stars **133**  Forks **210** license **Apache-2.0**


 **What is ResilientDB?**


ResilientDB is a high-performance, scalable, and secure blockchain platform designed to meet the demands of modern distributed applications. It offers a suite of tools and projects that enable developers and organizations to build robust blockchain solutions efficiently.


Table of contents


 What is ResilientDB?


 Get Started

 Main Repositories

 ResilientDB Ecosystem

 ResilientApps

 Why Choose ResilientDB?

 Get Involved

Cumulative Impact and Future Directions

- Docker images, CLI tools, API, ORM library, and CI/CD support collectively make Apache ResilientDB accessible and usable for developers and students.
- Over 300 students using Docker, increasing downloads, feedback from the academic community.
- Future Plans:
 - Expand CLI capabilities.
 - Enhance API features.
 - Continue developing CI/CD for scalability.

DEMO