

IBM Watson IoT

Hyperledger Composer Application Demo
using Node-RED

Watson IoT[™]

John Walicki

Watson IoT Developer Advocate

walicki@us.ibm.com

@johnwalicki



Hyperledger Composer App

Hyperledger Fabric running on IBM Container Service

I created this application to learn about:

1. Blockchain use cases
2. Kubernetes
3. IBM Container Service
4. Hyperledger Fabric
5. Hyperledger Composer
6. Hyperledger Composer REST APIs
7. Node-RED Dashboard



Storm Alert Node-RED Dashboard

Blockchain Storm Demo

Create a Storm Event

Enter New Storm name

Storm Name *

1

Initial Severity *



SUBMIT

CANCEL

Issue Storm Alert

Harvey

2

Create Alert

Description

5

SUBMIT

CANCEL

Alerts for Storm Harvey (CAT 2)

High Winds

Power Outages

3

Flooding

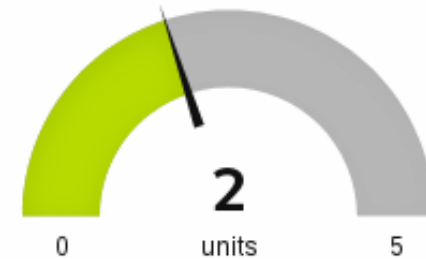
Severity Warning

4

Change Storm Severity



Storm Severity



Hyperledger Composer Playground

- Create a Blockchain Business Network
- Track Storms
 - (Assets)
- Track Alerts
 - (Transactions)
- Change Severity
 - Transactions that modify asset properties



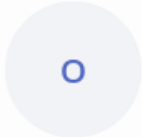
Hyperledger Composer Playground

My Business Networks



Import Business Network Card

Create Business Network Card

Connection: hlfv1




ollie@basic-storm-network



USER ID
ollie
BUSINESS NETWORK
basic-storm-network

Connect now →



You must import identity cards with the correct admin rights for deploying new business networks.

[Learn more.](#)

Legal

GitHub

Playground v0.16.0

Tutorial

Docs

Community

Hyperledger Composer Storm Model

- Define
 - Assets
 - Transactions
 - Properties
 - Participants
- Chaincode

The screenshot displays the Hyperledger Composer Storm Model interface. The top navigation bar shows the project name 'hlfv1 basic-storm-network' and the 'Define' tab is selected. The left sidebar lists the project files: 'About README.md', 'Model File models/org.acme.sample.cto' (which is selected), 'Script File lib/sample.js', and 'Access Control permissions.acl'. Below the file list is a '+ Add a file...' button and an 'Update' button. At the bottom of the sidebar is an 'Import/Replace' button. The main area shows the 'Model File models/org.acme.sample.cto' with a code editor containing the following CTO model definition:

```
1  /**
2   * Sample business network definition.
3   */
4   namespace org.acme.sample
5
6   /**
7    Participants
8   */
9   abstract participant SampleParticipant identified by participantId
10  {
11    o String participantId
12    o String orgName
13    o Boolean notified
14  }
15  participant FirstAlert extends SampleParticipant { }
16  participant Hospital extends SampleParticipant { }
17  participant Bank extends SampleParticipant { }
18  participant FireDepartment extends SampleParticipant { }
19  participant InsuranceCompany extends SampleParticipant { }
20  participant PoliceDepartment extends SampleParticipant { }
```

Below the code editor, a green checkmark icon and the text 'Everything looks good!' indicate that the model is valid. A message below states: 'Any problems detected in your code would be reported here'.

Hyperledger Composer Storm REST APIs

- Swagger OpenAPI documentation



Hyperledger Composer REST server			
Alert : A transaction named Alert	Show/Hide	List Operations	Expand Operations
Bank : A participant named Bank	Show/Hide	List Operations	Expand Operations
BeginStorm : A transaction named BeginStorm	Show/Hide	List Operations	Expand Operations
EndStorm : A transaction named EndStorm	Show/Hide	List Operations	Expand Operations
FireDepartment : A participant named FireDepartment	Show/Hide	List Operations	Expand Operations
FirstAlert : A participant named FirstAlert	Show/Hide	List Operations	Expand Operations
Hospital : A participant named Hospital	Show/Hide	List Operations	Expand Operations
InsuranceCompany : A participant named InsuranceCompany	Show/Hide	List Operations	Expand Operations
PoliceDepartment : A participant named PoliceDepartment	Show/Hide	List Operations	Expand Operations
Storm : An asset named Storm	Show/Hide	List Operations	Expand Operations
System : General business network methods	Show/Hide	List Operations	Expand Operations
updateStormSeverity : A transaction named updateStormSeverity	Show/Hide	List Operations	Expand Operations
UpdateStormStatus : A transaction named UpdateStormStatus	Show/Hide	List Operations	Expand Operations
[BASE URL: /api , API VERSION: 1.0.0]			

<https://openblockchain.readthedocs.io/en/latest/API/CoreAPI/>

Storm Alert Node-RED Dashboard

Blockchain Storm Demo

Create a Storm Event

Enter New Storm name

Storm Name * 1

Initial Severity * 2

SUBMIT CANCEL

Issue Storm Alert

Harvey 2

Create Alert

Description 5

SUBMIT CANCEL

Alerts for Storm Harvey (CAT 2)

High Winds

Power Outages 3

Flooding

Severity Warning 4

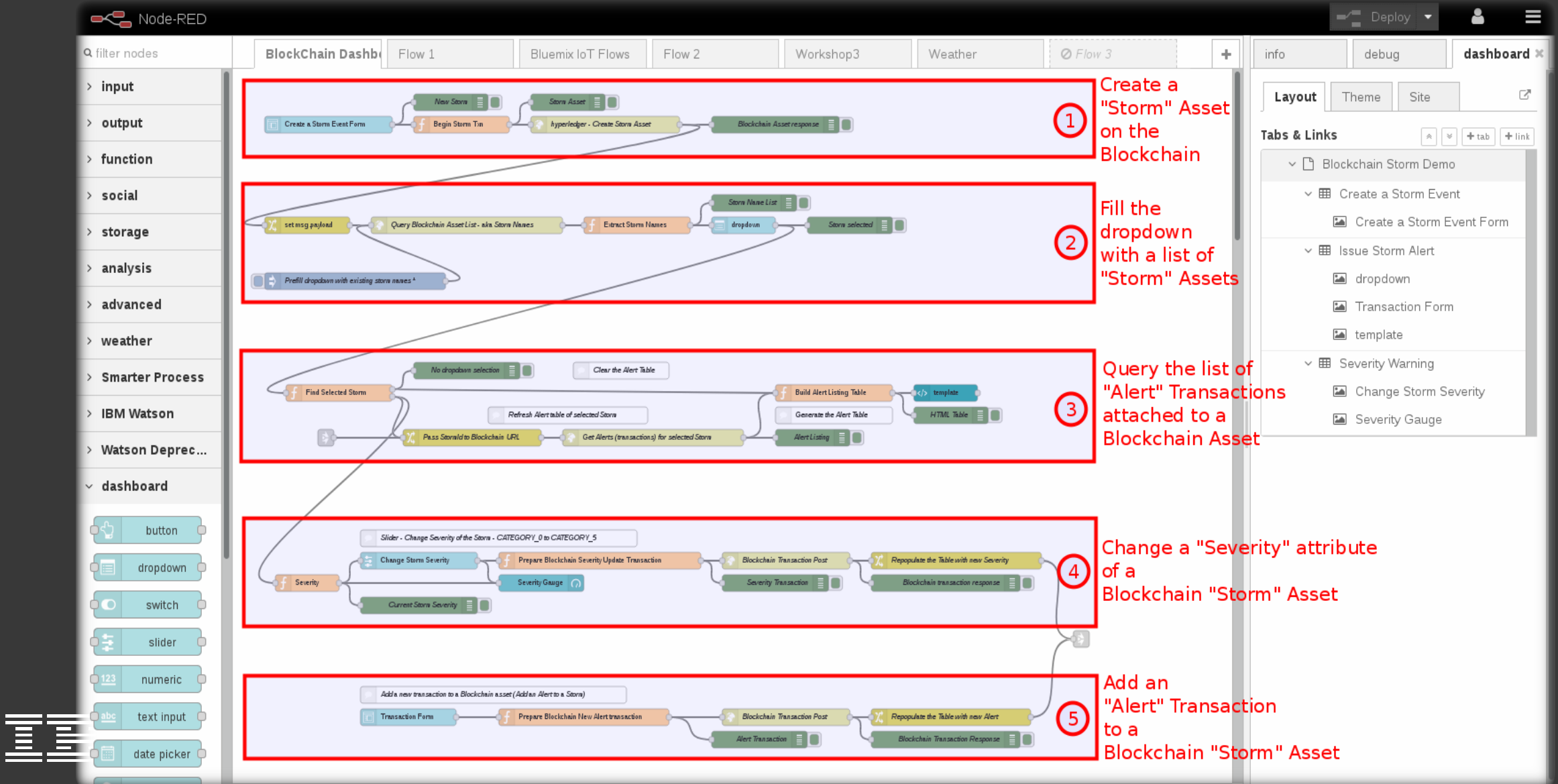
Change Storm Severity 4

Storm Severity

0 2 units 5



Blockchain Node-RED Flow



Node-RED and Hyperledger APIs

Live demo / code review of a Node-RED flow which calls Hyperledger Composer REST APIs and creates a dashboard

- Node-RED Dashboard
 - Forms
 - Dropdown
 - Slider
 - Gauge
 - Template
- Node-RED HTTP Request Nodes



Build with Watson

References:

- <https://ibm-blockchain.github.io/>
- <https://hyperledger.github.io/composer/reference/glossary.html>
- <https://nodered.org/>
- <https://console.bluemix.net/containers-kubernetes/catalog/cluster>
- <https://www.coursera.org/learn/developer-nodered>

John Walicki Twitter: @johnwalicki

