



Cloning Running Servers

# Ross Boucher

@boucher

dockercon 16

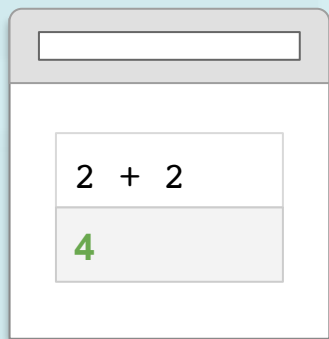
# Tonic



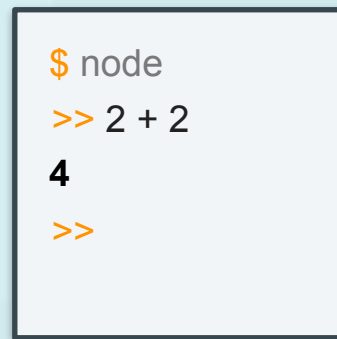
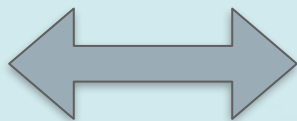
The screenshot shows a web browser window with the URL `tonicdev.com/tonic/city-forecast`. The page title is "Forecast". Below the title, it shows the environment: "node v0.12.9" and "version: 5.0.0". The main text explains that `forecastCities` is an asynchronous function that returns a day forecast for cities. Below this is a code editor with the following code:

```
1 var Promise = require("bluebird");
2 var R = require("ramda");
3
4 await forecastCities("New York", "San Francisco");
```

Below the code editor, there is a tabbed interface with "Object" selected and "Chart" as an alternative view. The "Chart" view displays a bar chart with orange bars. The y-axis has a label "60". The chart shows approximately 15 bars of varying heights, mostly between 50 and 70.



Browser

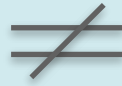


Server

```
[1] x = 1
    1

[2] ++x
    2

[3] x == 2
    true
```



```
[4] x = 1
    1

[2] ++x
    2

[3] x == 2
    true
```

# Tonic

Forecast

node v0.12.9 version: 5.0.0

forecastCities is an asynchronous function (defined below) that will return a day forecast of the cities passed in. Use the object viewer pop up to select and compare them:

```
1 var Promise = require("bluebird");
2 var R = require("ramda");
3
4 await forecastCities("New York", "San Francisco");
```

Object Chart

# CRIU

(Checkpoint Restore In Userspace)

# CRIU

+

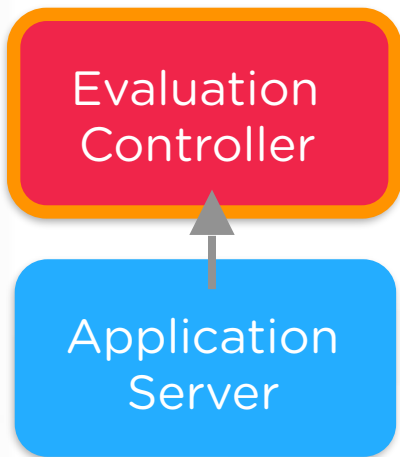


## Public Internet



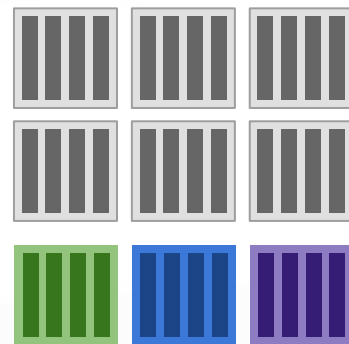
User's Browser

## Internal Network



Primary Docker

## Isolated Network

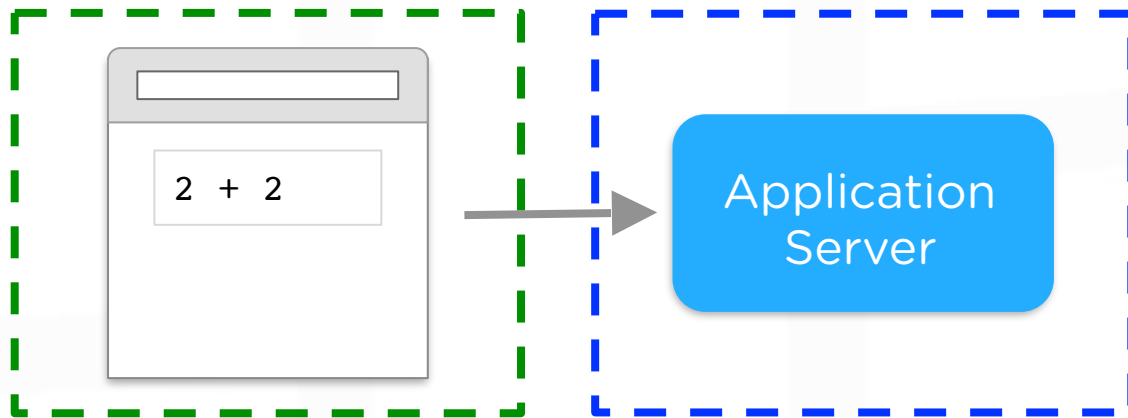


Container Pool

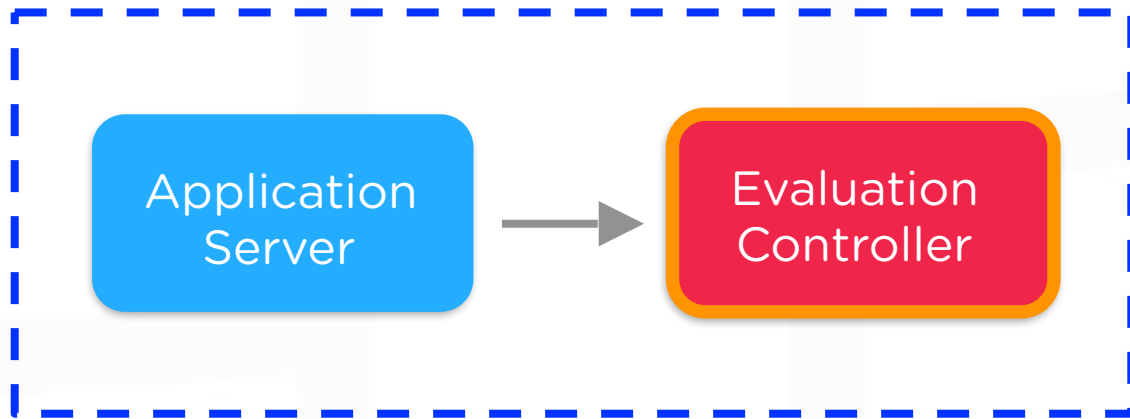


Eval Docker

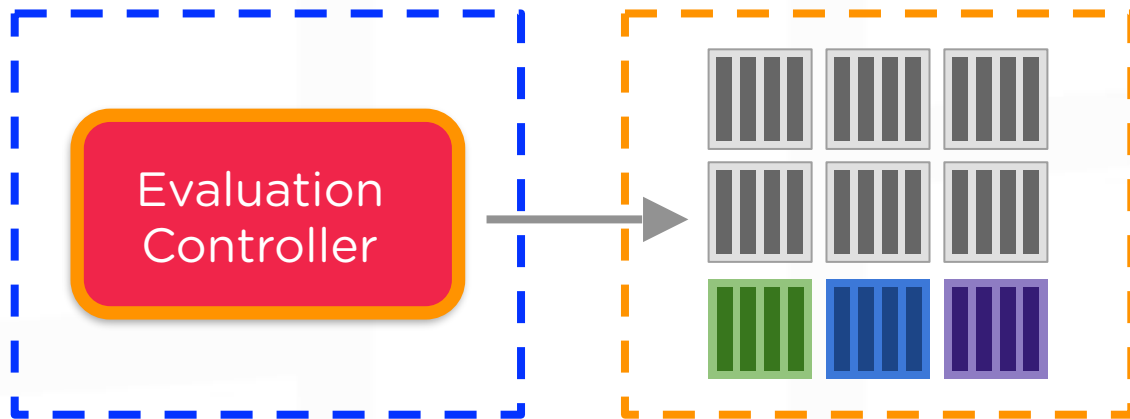




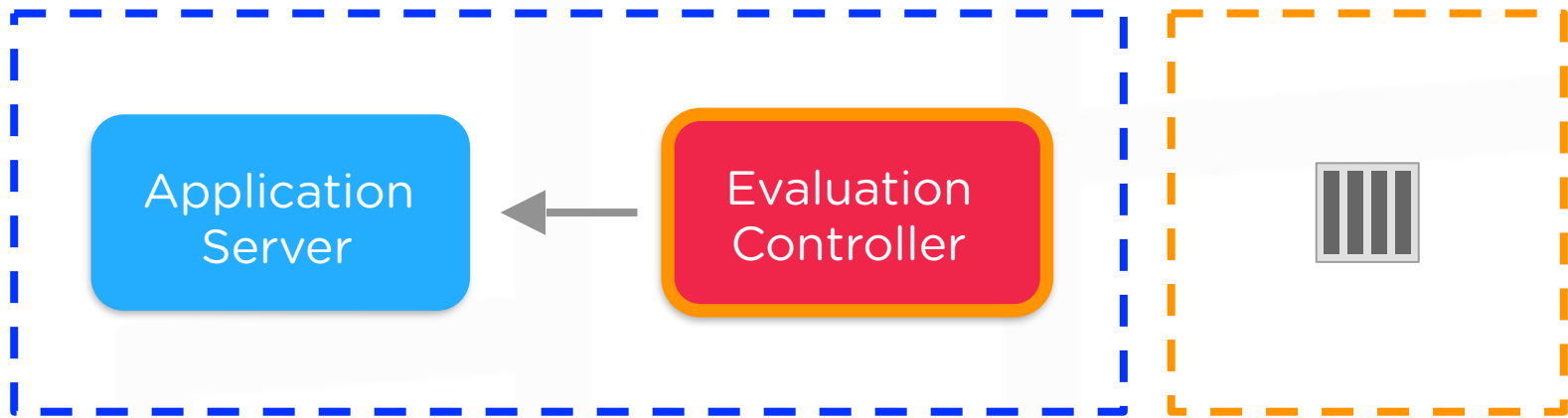
```
message: "evaluate"  
nodeVersion: "4.x.x"  
sources: [{  
  type: "source",  
  text: "var x = 2 + 2",  
  checksum: "a8efd"  
}]  
url: "/users/boucher/repositories/12345/branches/master"
```



```
message: "get-evaluator"  
nodeVersion: "4.x.x"  
checksums: [ "a8efd" ]  
url: "/users/boucher/repositories/12345/branches/master"
```



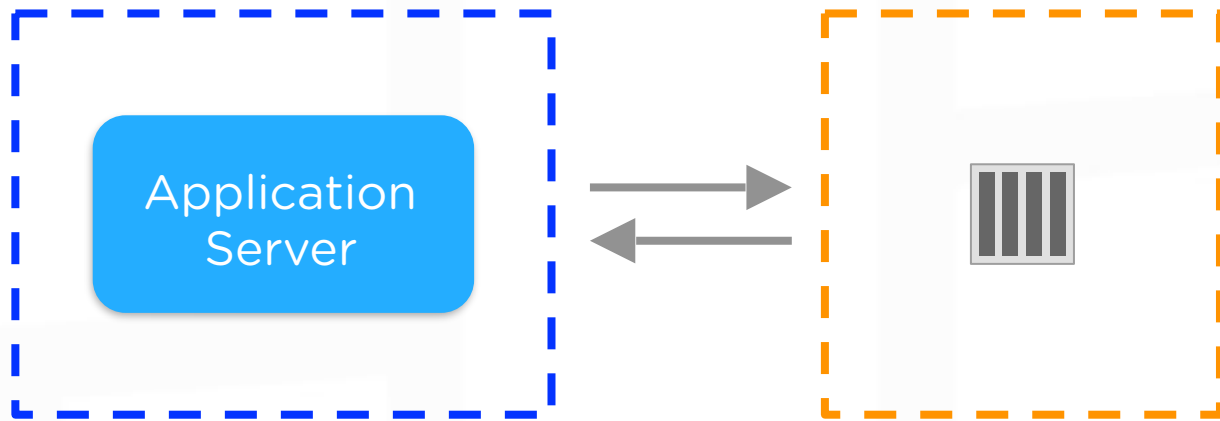
```
function get_evaluator(configuration) {  
  if (cached_containers[configuration]) {  
    return cached_containers[configuration]  
  }  
  if (checkpoint_exists(configuration)) {  
    return restored_container(configuration);  
  }  
  return pooled_container(configuration)  
}
```



message: "found-evaluator"

IP: "172.0.1.201"

port: "7777"



```
message: "establish-connection"
```

```
message: "source-at-index"
```

```
index: 0
```

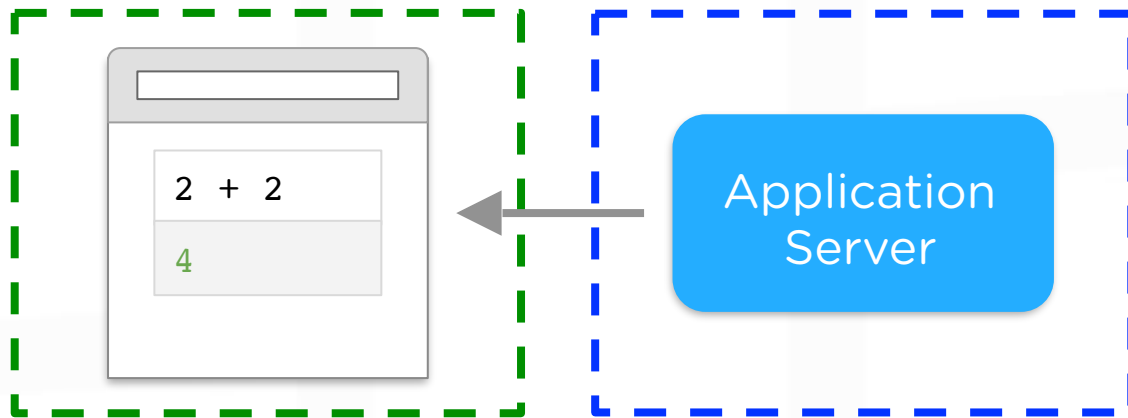
```
source: "var x = 2+2;"
```

```
message: "get-source"
```

```
index: 0
```

```
message: "output"
```

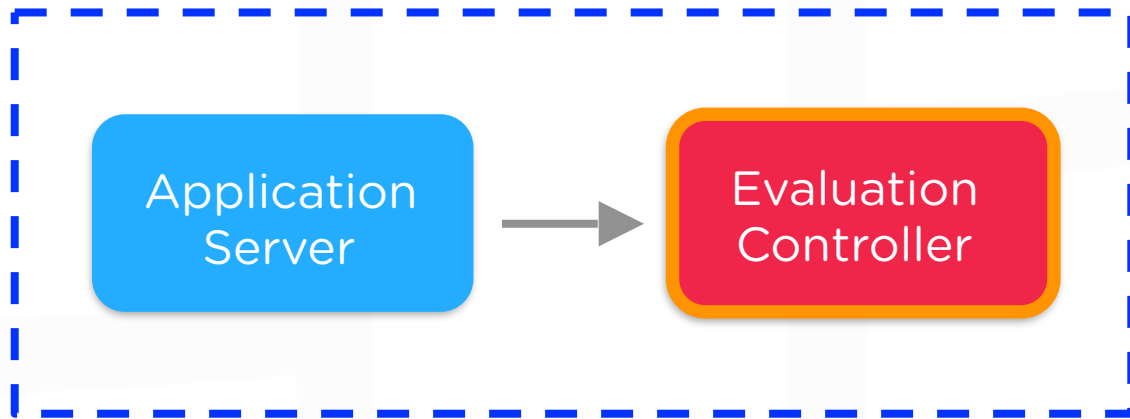
```
value: 4
```



message: "output"

index: 0

value: 4



message: "checkpoint-evaluator"

checksum: "a8efd"

url: "/users/boucher/repositories/12345/branches/master"



Evaluation  
Controller



```
await container.checkpoint({
  LeaveRunning: true,
  ImagesDirectory: "/checkpoints/<document_id>/<checksum>/"
})
if (await container.changes().length > 0) {
  metadata.image = await container.commit({ pause: false })
}
await fs.writeFile(metadata,
  "/checkpoints/<document_id>/<checksum>/metadata.txt")
```





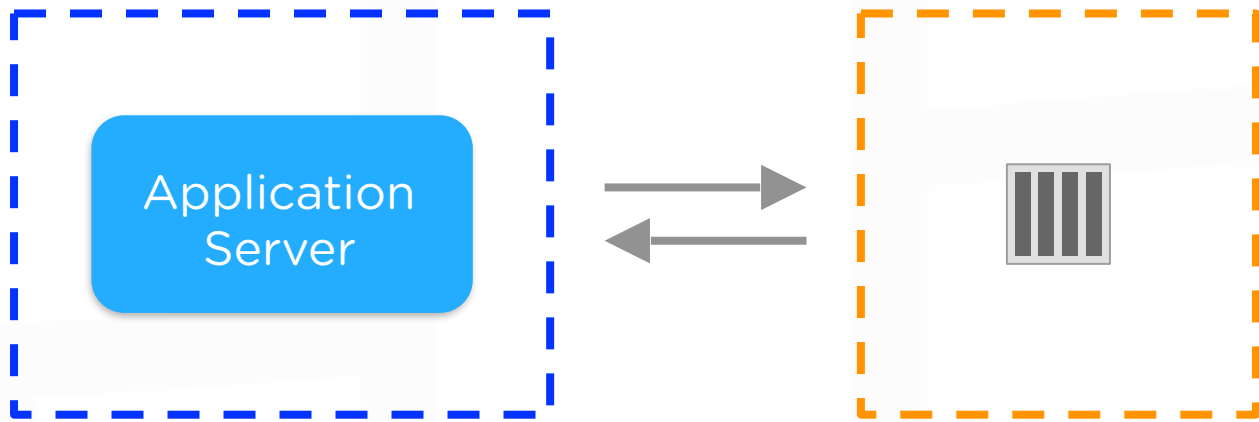
```
# current cli
```

```
$ docker checkpoint --leave-running --image-dir=/checkpoint/path  
<container_id>
```

```
$ docker commit --pause=false <container_id>
```

```
# new cli
```

```
$ docker checkpoint --exit=false <container_id> <checkpoint_id>
```



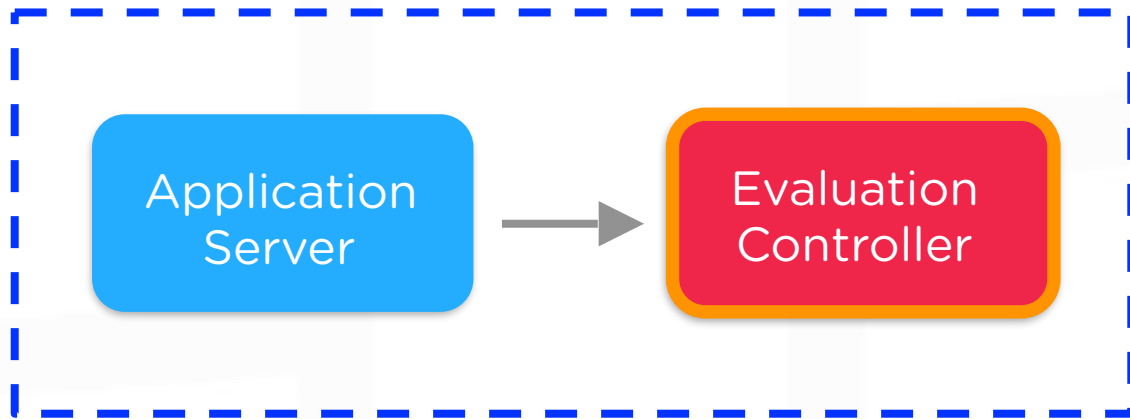
message: "source-at-index"

...

message: "get-source"

...

message: "finished"  
(process exited)



message: "predict-evaluator"

checksums: [ "a8efd" ]

url: "/users/boucher/repositories/12345/branches/master"



```
function predict_evaluator(configuration) {  
  var cpPath = "/checkpoints/<document_id>/<checksum>/" ;  
  var metadata = await fs.readFile(metadata, cpPath + "/metadata.txt");  
  var container = await docker.restore({  
    ImagesDirectory: cpPath  
    ...  
  });  
  cached_containers[configuration] = container;  
}
```



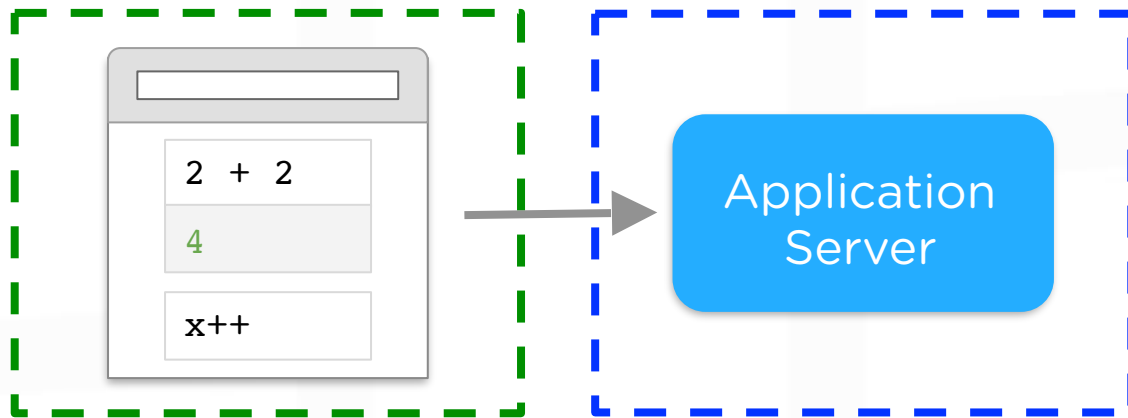
```
$ docker create tonic/worker  
<container_id>
```

```
# current cli
```

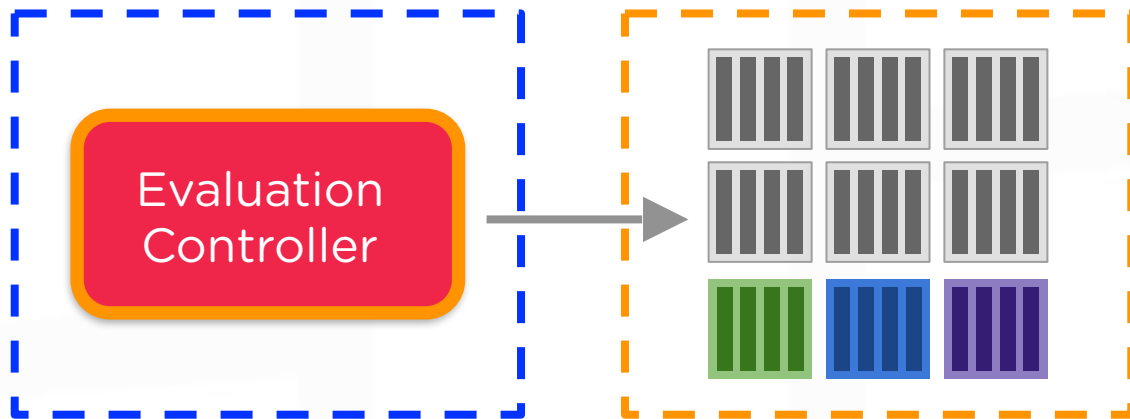
```
$ docker restore --force -image-dir=/checkpoint/path <container_id>
```

```
# new cli
```

```
$ docker start --checkpoint <checkpoint_id> <container_id>
```



```
message: "evaluate"  
nodeVersion: "4.x.x"  
sources: [  
  { type: "source", text: "var x = 2 + 2", checksum: "a8fed" },  
  { type: "source", text: "x++", checksum: "b9ccc" }  
]  
url: "/users/boucher/repositories/12345/branches/master"
```



```
function get_evaluator(configuration) {  
    if (cached_containers[configuration]) {  
        return cached_containers[configuration]  
    }  
    if (checkpoint_exists(configuration)) {  
        return restored_container(configuration);  
    }  
    return pooled_container(configuration)  
}
```

# Security Concerns

- Drop any capabilities you don't need
- Set CPU, memory, and network constraints
- User Namespaces
- Network Isolation
- Seccomp



# Security Concerns

- AUFS errors
- CRIU failures
- Race conditions
- Zombie processes
- Docker daemon restarts
- Filesystem management

# DockWorker

[github.com/boucher/dockworker](https://github.com/boucher/dockworker)

<> Code

Issues 1,501

**Pull requests 70**

Wiki

Pulse

Graphs

# Add checkpoint/restore to docker API #13602

**Closed**

boucher wants to merge 13 commits into `docker:master` from `boucher:cr-combined`

Conversation 320

Commits 13

Files changed 37



boucher commented on May 29, 2015



Adds support in the daemon and driver for checkpoint and restore, and exposes that in the API, but not in the CLI yet.

❤️ 2

# [WIP] Implement containerd API for checkpoints #22049

**Open** boucher wants to merge 2 commits into `docker:master` from `boucher:docker-checkpoint-restore`

Conversation 11

Commits 2

Files changed 25



boucher commented on Apr 14



This is an *incomplete* implementation of the containerd checkpoint API for docker. It depends on [docker/engine-api#202](#), but you can check out a branch with those changes vendored in (in order to build) here: <https://github.com/boucher/docker/tree/docker-checkpoint-restore-vendored>

Primarily hoping to start the discussion of the best way to get checkpoint/restore working now that 1.11 has shipped with containerd.

(ping #20300)

# Container Migration

(and other potential use cases)

# Further Reading

- [Pre-compiled Release](#) (based on Docker 1.10)
- [Checkpoint/Restore Pull Request](#)
- [Saied Kazemi's Linux Plumber's Talk](#)
- [CRIU Homepage](#)
- [DockerCon Doom Demo](#)
- [Tonic Blog](#) on checkpoint/restore
- [DockWorker](#) on Github
- [Using P.Haul with Docker](#)
- [DockerScript](#) is a cool tool we use to manage our images

# Thanks!

@boucher • rboucher@gmail.com

