

Ray API: Tasks. (remote funcs).

`@ray.remote`

```
def zeros(shape): //(numpy as np)  
    return np.zeros(shape)
```

`@ray.remote`

```
def dot(a, b):  
    return np.dot(a, b).
```

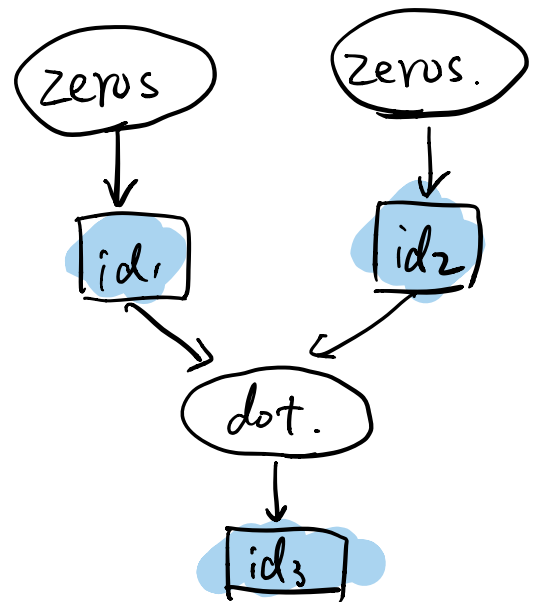
```
id1 = zeros.remote([4, 4]).
```

```
id2 = zeros.remote([4, 4]).
```

```
id3 = dot.remote(id1, id2).
```

```
obj = ray.get(id3).
```

DAG (directed acyclic graph).



Ray APZ: Actors (remote).

`@ray.remote`. (num_gpus = 4).

class Counter (object):

```
def __init__(self):  
    self.value = obj.val.
```

```
def inc (self):  
    self.value += 1  
    return self.value.
```

```
c = Counter.remote(5),
```

```
id1 = c.remote.inc()
```

```
id2 = c.remote.inc()
```

```
id3 = c.remote.inc()
```

```
ray.get([id1, id2, id3]).  
// [6, 7, 8]
```

