

Side Effect Handling in re-frame Apps

Ertuğrul Kağan Erfidan
Software Engineer

tugh@pm.me

Hands-on Approach

6 Dominoes



6 Dominoes

1. Event dispatch
2. Event handling
3. Effect handling
4. Query
5. View
6. DOM





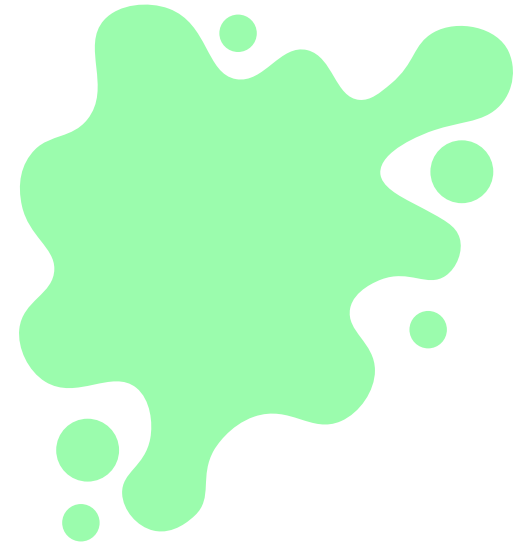
Effects are...

what your app does (*a.k.a. hidden outputs*)

Coeffects are...

what the world does (*a.k.a. hidden inputs*)

Do not write effectful code
in event handlers

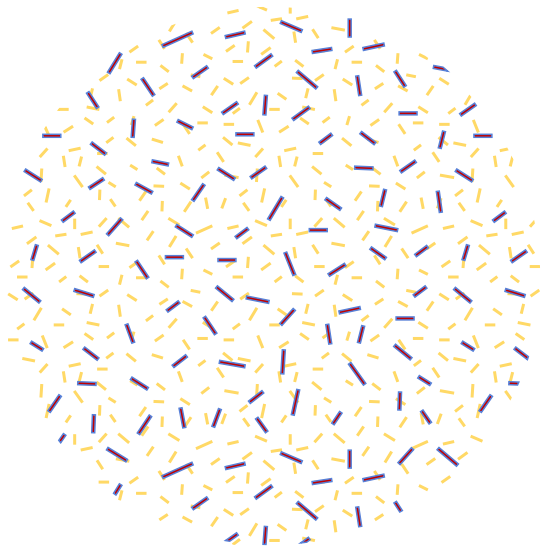




```
(reg-event-db  
  ::some-event  
  (fn [db]  
    (dispatch [::another-event])  
    (assoc db :some-field :some-value)))
```



```
(reg-event-fx  
  ::some-event  
  (fn [{:keys [db]}]  
    {:db (assoc db :some-field :some-value)  
     :dispatch [::another-event]}))
```



All effects should happen inside
effect and *co-effect* handlers

```
(reg-fx
 :db
 (fn [value]
  (reset! re-frame.db/app-db value)))
```

```
(reg-cofx
 :now
 (fn [coeffacts _]
  (assoc coeffacts :now (js.Date.)))))
```



```
(reg-event-fx
  ::register
  [(inject-cofx :now)]

  (fn [{:keys [db now]} [_ username]]
    {:db (assoc db :username username
                  :registered-at now)}))
```



[tugh/reframe-side-effects](https://github.com/tugh/reframe-side-effects)