UI

Visible Todo List

fetchTodos → C (Middleware) → Saga → D

'all'
{meta-filter}
fetchTodosRequest

byFilter
isFetching

fetching=false
filter='all'

all.isFetching=true
active.isFetching=state
completed.isFetching=state

'all'
{meta-filter}
fetchTodosSuccess
D

all.ids=resp
active.ids=state
completed.ids=state

byFilter
ids

addTodo → C (Middleware) → Saga → addTodoSuccess → D

byFilter
ids

all.ids = resp
active.ids = resp

completed.ids=state

---

fetchTodos, fetchTodosRequest,
fetchTodosSuccess, fetchTodosFailure,
addTodo, addTodoSuccess, toggleTodo,
toggleTodoSuccess.

fetchTodosRequest

fetchTodos → setLoading → setTodos
              toggleFetching

addTodo → setAddTodo

toggleTodo → setToggleTodo
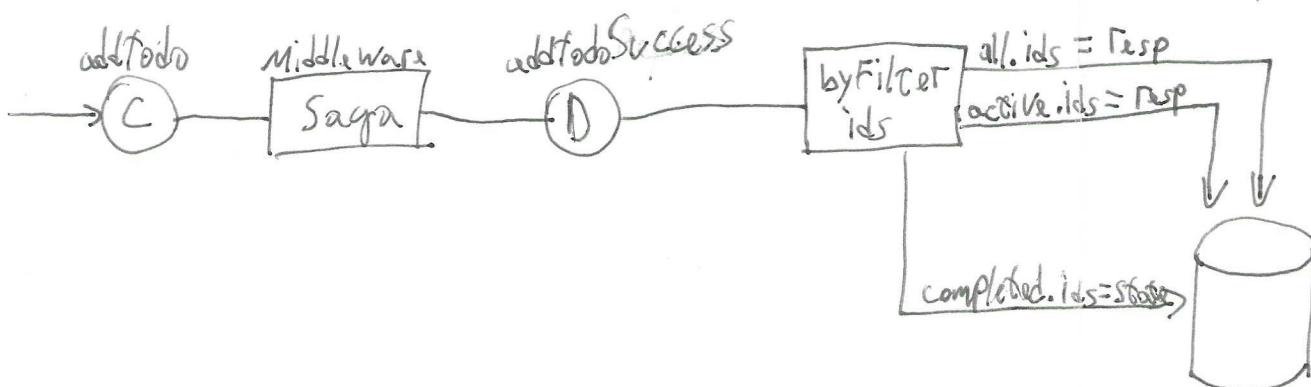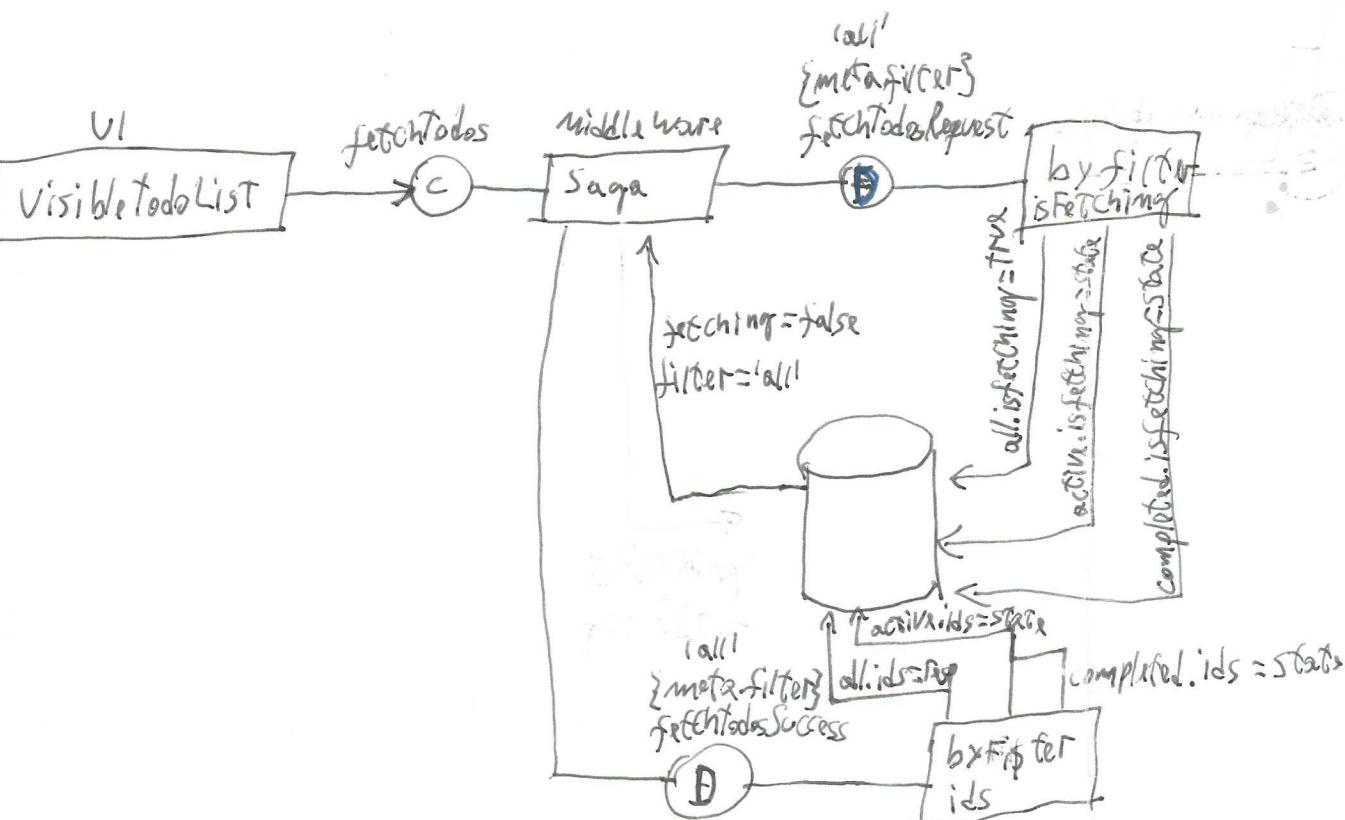
```
const action = {
  type: setTodos,
  error: true,
  payload: new Error ('Booom!')
}
```
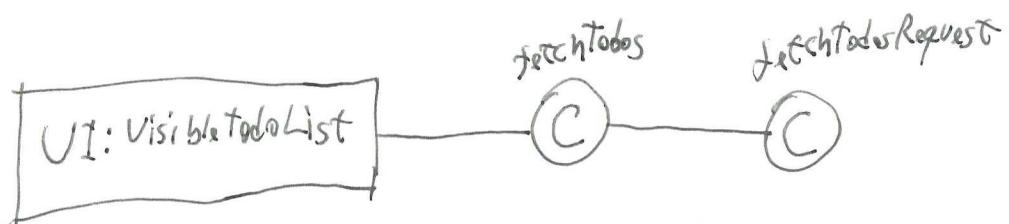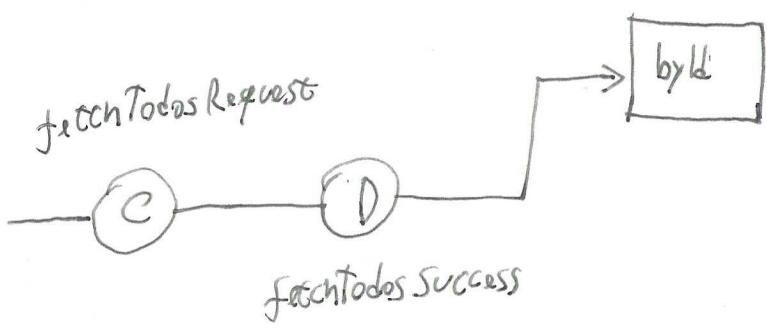
- if the action were in the error state,
the payload would hold the error itself,
be it an Error object.

fetchTodos Request

by Id

C ─── D

fetchTodos Success

UI: Visible todo List ─── C (fetchTodos) ─── C (fetchTodos Request)

Redux Patterns

Actions

Reducers
+
Selectors

Store

<<uses>>

<<call>> reducer (state, Action)

getState()

<<uses>>

Thunks

dispatch(action)

<<call>> Thunks()

UI

dispatch(thunk)

<<uses>>

UI

dispatch(action)

<<uses>>

Actions

Reducers
+
Selectors

Store

<<uses>>

<<call>> reducer (state, Action)

getState()

<<uses>>

WorkersSaga

<<put>> dispatch(action)

<<call>> (action)

<<uses>>

WatchersSaga