

# OrigoDB Workshop

## Module 6 – Immutability

### Immutability and blocking



## Example immutable model

```
[Serializable]
public class TodoModel : Model
{
    private String[] _tasks = {};

    public TodoModel AddTask(string task)
    {
        var newState = new TodoModel();
        var temp = _tasks.ToList();
        temp.Add(task);
        newState._tasks = temp.ToArray();
        return newState;
    }

    public IEnumerable<String> Tasks()
    {
        return _tasks.ToArray();
    }
}
```

## Example immutable entity

```
public class Task
{
    public readonly string Title;
    public readonly DateTime? Completed;

    public Task(string title, DateTime? completed = null)
    {
        Title = title;
        Completed = completed;
    }

    public Task Complete(DateTime completed)
    {
        return new Task(Title, completed);
    }
}
```

## Immutable command example

```
[Serializable]
public class AddTaskCommand : ImmutableCommand<TodoModel>
{
    public readonly string Task;

    public AddTaskCommand(string task)
    {
        Task = task;
    }

    public override void Execute(TodoModel model, out TodoModel next)
    {
        next = model.AddTask(task);
    }
}
```

## Configuration

```
var config = EngineConfiguration.Create();
config.Kernel = Kernels.Immutability;
config.Synchronization = SynchronizationMode.None;

// or to get both settings
var config = EngineConfiguration.Create().WithImmutability();

//pass config when creating engine
var engine = Engine.For<TodoModel>(config);

//execute commands as usual
engine.Execute(new AddTaskCommand("500 backhand volley drop shots"));
```

## Lab

- Try out the example from the slides