
Table of Contents

Preface	ix
1. Introduction	1
2. What Is a Pattern?	3
We Already Use Patterns Every Day	4
3. "Pattern"-ity Testing, Proto-Patterns, and the Rule of Three	7
4. The Structure of a Design Pattern	9
5. Writing Design Patterns	11
6. Anti-Patterns	13
7. Categories of Design Patterns	15
Creational Design Patterns	15
Structural Design Patterns	16
Behavioral Design Patterns	16
8. Design Pattern Categorization	17
A Brief Note on Classes	17
9. JavaScript Design Patterns	21
The Constructor Pattern	22
Object Creation	22
Basic Constructors	24
Constructors with Prototypes	25
The Module Pattern	26
Object Literals	26

The Module Pattern	27
Module Pattern Variations	32
The Revealing Module Pattern	37
Advantages	38
Disadvantages	38
The Singleton Pattern	39
The Observer Pattern	43
Differences Between the Observer and Publish/Subscribe Pattern	47
Advantages	49
Disadvantages	50
Publish/Subscribe Implementations	50
The Mediator Pattern	60
Basic Implementation	61
Advanced Implementation	62
Example	67
Advantages and Disadvantages	69
Mediator Versus Observer	69
Mediator Versus Facade	70
The Prototype Pattern	70
The Command Pattern	73
The Facade Pattern	75
Notes on Abstraction	78
The Factory Pattern	78
When to Use the Factory Pattern	81
When Not to Use the Factory Pattern	81
Abstract Factories	81
The Mixin Pattern	82
Subclassing	82
Mixins	84
Advantages and Disadvantages	87
The Decorator Pattern	87
Pseudoclassical Decorators	91
Interfaces	91
Abstract Decorators	92
Decorators with jQuery	95
Advantages and Disadvantages	97
Flyweight	97
Using Flyweights	98
Flyweights and Sharing Data	98
Implementing Classical Flyweights	99
Converting Code to Use the Flyweight Pattern	102
A Basic Factory	104
Managing the Extrinsic States	105

The Flyweight Pattern and the DOM	106
10. JavaScript MV* Patterns	111
MVC	111
Smalltalk-80 MVC	111
MVC for JavaScript Developers	112
Models	113
Views	115
Controllers	117
Controllers in Another Library (Spine.js) Versus Backbone.js	118
What Does MVC Give Us?	120
Smalltalk-80 MVC in JavaScript	120
Delving Deeper	121
Summary	121
MVP	121
Models, Views, and Presenters	122
MVP or MVC?	123
MVC, MVP, and Backbone.js	123
MVVM	125
History	126
Model	127
View	127
ViewModel	130
Recap: The View and the ViewModel	132
Recap: The ViewModel and the Model	132
Pros and Cons	132
Advantages	132
Disadvantages	132
MVVM with Looser Data Bindings	133
MVC Versus MVP Versus MVVM	137
Backbone.js Versus KnockoutJS	138
11. Modern Modular JavaScript Design Patterns	141
A Note on Script Loaders	142
AMD	142
Getting Started with Modules	143
AMD Modules with Dojo	146
AMD Module Design Patterns (Dojo)	147
AMD Modules with jQuery	149
AMD Conclusions	151
CommonJS	152
Getting Started	152
Consuming Multiple Dependencies	153

Loaders and Frameworks that Support CommonJS	153
Is CommonJS Suitable for the Browser?	154
Related Reading	154
AMD and CommonJS: Competing, but Equally Valid Standards	155
UMD: AMD and CommonJS-Compatible Modules for Plug-ins	155
ES Harmony	160
Modules with Imports and Exports	161
Modules Loaded from Remote Sources	162
Module Loader API	162
CommonJS-like Modules for the Server	162
Classes with Constructors, Getters, and Setters	163
ES Harmony Conclusions	164
Related Reading	164
Conclusions	164
12. Design Patterns in jQuery	167
The Composite Pattern	167
The Adapter Pattern	168
The Facade Pattern	170
The Observer Pattern	172
The Iterator Pattern	175
Lazy Initialization	176
The Proxy Pattern	178
The Builder Pattern	179
13. jQuery Plug-in Design Patterns	181
Patterns	182
A Lightweight Start Pattern	183
Complete Widget Factory Pattern	185
Nested Namespacing Plug-in Pattern	187
Custom Events Plug-in Pattern (with the Widget Factory)	189
Prototypal Inheritance with the DOM-to-Object Bridge Pattern	191
jQuery UI Widget Factory Bridge Pattern	193
jQuery Mobile Widgets with the Widget Factory	195
RequireJS and the jQuery UI Widget Factory	198
Usage	200
Globally and Per-Call Overridable Options (Best Options Pattern)	201
A Highly Configurable and Mutable Plug-in Pattern	202
What Makes a Good Plug-in Beyond Patterns?	205
Quality	205
Code Style	205
Compatibility	205
Reliability	205

Performance	206
Documentation	206
Likelihood of maintenance	206
Conclusions	206
Namespacing Patterns	207
Namespacing Fundamentals	207
Single Global Variables	207
Prefix Namespacing	208
Object Literal Notation	208
Nested Namespacing	212
Immediately Invoked Function Expressions (IIFE)s	213
Namespace Injection	215
Advanced Namespacing Patterns	217
Automating Nested Namespacing	217
Dependency Declaration Pattern	219
Deep Object Extension	220
Recommendation	223

14. Conclusions	225
------------------------	------------

Appendix: References	227
-----------------------------	------------

Index	231
--------------	------------