

# Indice generale

<b>Prefazione</b> .....	<b>ix</b>
<b>Introduzione</b> .....	<b>xiii</b>
Per chi è questo libro?.....	xiv
Che cosa fa di un programmatore un pragmatic programmer? .....	xiv
Pragmatici singoli, grandi squadre.....	xv
È un processo continuo.....	xvi
Come è organizzato il libro.....	xvi
Che cosa c'è in un nome?.....	xvi
Codice sorgente e altre risorse.....	xvii
Mandateci un feedback .....	xvii
Ringraziamenti.....	xvii
<b>Capitolo 1 Una filosofia pragmatica</b> .....	<b>1</b>
Il gatto mi ha mangiato il codice sorgente .....	2
Prendersi la responsabilità .....	2
Entropia del software .....	3
Estinguere l'incendio .....	4
Zuppa di pietre e rane bollite .....	5
Dalla parte degli abitanti del villaggio.....	6
Software abbastanza buono.....	7
Coinvolgete gli utenti nel compromesso .....	8
Sapere quando fermarsi.....	8
Il portafoglio delle conoscenze .....	9
Il vostro portafoglio di conoscenze.....	9
Costruire il proprio portafoglio.....	10
Obiettivi.....	10
Occasioni di apprendimento .....	11
Pensiero critico.....	12
Comunicate!.....	13
Dovete aver chiaro che cosa volete dire .....	13

Dovete conoscere il vostro pubblico.....	14
Scegliete il momento giusto.....	15
Scegliete uno stile.....	15
Che siano di bell'aspetto.....	15
Coinvolgete il vostro pubblico.....	16
Imparate ad ascoltare.....	16
Rispondete alle persone.....	17
Riepilogo.....	17

## **Capitolo 2 Un approccio pragmatico.....19**

I mali della duplicazione.....	20
Come nasce la duplicazione?.....	21
Duplicazione imposta.....	21
Duplicazione inavvertita.....	22
Duplicazione per impazienza.....	24
Duplicazione fra sviluppatori.....	24
Ortogonalità.....	25
Che cos'è l'ortogonalità?.....	26
Benefici dell'ortogonalità.....	26
Team di progetto.....	28
Progetto.....	28
Toolkit e librerie.....	29
Codifica.....	30
Test.....	31
Documentazione.....	31
Vivere con l'ortogonalità.....	32
Reversibilità.....	33
Reversibilità.....	34
Architettura flessibile.....	35
Proiettili traccianti.....	36
Codice che brilla nel buio.....	36
I proiettili traccianti non sempre colpiscono il bersaglio.....	38
Codice tracciante contro prototipazione.....	38
Prototipi e Post-it.....	39
Cose da prototipare.....	40
Come usare i prototipi.....	40
Prototipi di architettura.....	41
Come non usare i prototipi.....	41
Linguaggi di settore.....	42
Implementare un mini-linguaggio.....	44
Linguaggi di dati e linguaggi imperativi.....	45
Linguaggi standalone e incorporati.....	46
Sviluppo facile o manutenzione facile?.....	47
Stime.....	48
Quanto accurato è abbastanza accurato?.....	48
Da dove arrivano le stime?.....	49

Capire la domanda.....	49
Costruire un modello del sistema.....	50
Suddividere il modello in componenti.....	50
Dare a ciascun parametro un valore.....	50
Calcolare le risposte.....	51
Tener traccia della propria abilità nelle stime.....	51
Stimare i tempi di un progetto.....	51
Che cosa dire quando viene chiesta una stima.....	52

### **Capitolo 3    Gli strumenti di base.....53**

Il potere del puro testo.....	54
Che cos'è il puro testo?.....	54
Svantaggi.....	55
Il potere del testo.....	56
Il minimo comun denominatore.....	57
Giochi di shell.....	58
Utility di shell e sistemi Windows.....	60
Power Editing.....	61
Un editor.....	61
Caratteristiche degli editor.....	62
Produttività.....	63
E adesso?.....	64
Quali editor sono disponibili?.....	64
Controllo del codice sorgente.....	65
Controllo del codice sorgente e build.....	66
Ma il mio team non usa il controllo del codice sorgente.....	66
Prodotti per il controllo del codice sorgente.....	66
Debug.....	67
Psicologia del debug.....	67
Una mentalità da debug.....	68
Da dove iniziare.....	68
Strategie di debug.....	69
Lista di controllo per il debug.....	73
Manipolazione del testo.....	74
Generatori di codice.....	76
Generatori di codice passivi.....	77
Generatori di codice attivi.....	77
I generatori di codice non devono essere per forza complessi...79	
I generatori di codice non devono per forza generare codice....79	

### **Capitolo 4    Paranoia pragmatica .....81**

Progettare per contratto.....	82
DBC.....	83
DBC e crash precoci.....	86
Altri usi delle invarianti.....	87
Contratti dinamici e agenti.....	89

I programmi morti non mentono.....	91
Programmazione assertiva .....	92
Lasciate attive le asserzioni .....	93
Quando usare le eccezioni .....	95
Che cos'è eccezionale?.....	96
I gestori degli errori sono un'alternativa.....	97
Come equilibrare le risorse .....	98
Annidare le allocazioni.....	100
Oggetti ed eccezioni.....	100
Equilibri ed eccezioni.....	100
Quando non si possono equilibrare le risorse .....	102
Controllare l'equilibrio .....	103

## Capitolo 5 **Piegarsi o spezzarsi .....**105

Disaccoppiamento e legge di Demetra .....	106
Ridurre al minimo l'accoppiamento .....	106
La legge di Demetra per le funzioni .....	107
Cambia davvero qualcosa?.....	108
Metaprogrammazione .....	110
Configurazione dinamica.....	110
Applicazioni guidate dai metadati.....	111
Logica di business .....	112
Configurazione cooperativa .....	113
Non scrivere codice dodo.....	114
Accoppiamento temporale .....	114
Flusso di lavoro.....	115
Architettura .....	116
Progettazione per la concorrenza .....	118
Interfacce più pulite.....	118
Messa in esercizio .....	119
È solo una vista.....	120
Pubblicare/abbonarsi .....	120
Modello-Vista-Controllo.....	122
Oltre le GUI .....	123
Ancora accoppiati (dopo tutti questi anni).....	125
Lavagne .....	125
Implementazioni a lavagna .....	127
Un esempio applicativo .....	128

## Capitolo 6 **Mentre codificate.....**131

Programmazione per coincidenza.....	132
Come programmare per coincidenza.....	132
Come programmare consapevolmente .....	134
Velocità degli algoritmi .....	136
Che cosa si intende per stima degli algoritmi? .....	136
Stima a buon senso .....	138

La velocità degli algoritmi in pratica.....	139
Il meglio non è sempre il meglio.....	140
Refactoring.....	141
Refactoring: quando? .....	142
Come si procede al refactoring?.....	143
Codice facile da sottoporre a test.....	145
Unit test.....	146
Test rispetto al contratto .....	146
Scrivere unit test.....	148
Test harness .....	149
Costruire una finestra di test .....	150
Una cultura del test .....	151
Maghi cattivi .....	152

## **Capitolo 7**    **Prima del progetto.....155**

La fossa dei requisiti .....	156
Scavare alla ricerca dei requisiti .....	156
Documentare i requisiti .....	158
Solo una cosina ancora.....	162
Mantenere un glossario.....	162
Fate girare parola .....	163
Risolvere rompicapo impossibili.....	164
Gradi di libertà .....	164
Deve esserci un modo più facile!.....	165
Non finché non siete pronti.....	166
Sano giudizio o procrastinazione? .....	166
La trappola delle specifiche .....	167
Cerchi e frecce.....	169

## **Capitolo 8**    **Progetti pragmatici .....173**

Team pragmatici .....	174
Niente finestre rotte.....	174
Rane bollite .....	174
Comunicare.....	175
Non ripetetevi.....	175
Ortogonalità.....	176
Automazione.....	177
Sapere quando smettere di aggiungere vernice .....	177
Automazione onnipresente .....	178
Tutto in automatico.....	179
Compilazione del progetto .....	179
Automazione delle build.....	180
Automazione delle attività amministrative .....	181
I figli del calzolaio .....	182
Test senza pietà .....	183
Che cosa sottoporre a test.....	184

Come sottoporre a test .....	186
Quando sottoporre a test .....	189
Stringere la rete .....	190
È tutta scrittura .....	190
Commenti nel codice .....	191
Documenti eseguibili .....	193
Technical writer .....	194
O lo stampi o lo intrecci .....	194
Linguaggi di marcatura .....	195
Grandi speranze .....	196
Comunicare aspettative .....	196
Il miglio extra .....	197
Orgoglio e pregiudizio .....	198

## **Appendice A Risorse .....199**

Associazioni professionali .....	199
Costruirsi una biblioteca .....	200
Periodici .....	200
Settimanali .....	201
Libri .....	201
Risorse Internet .....	202
Editor .....	202
Compilatori, linguaggi e strumenti di sviluppo .....	203
Strumenti Unix sotto DOS .....	205
Strumenti di controllo del codice sorgente .....	205
Altri strumenti .....	206
Articoli e pubblicazioni .....	206
Vari .....	207
Bibliografia .....	207

## **Appendice B Risposte agli esercizi .....211**

## **Indice analito .....235**