

Table des matières

Avant-propos	XIII
Préface	XVII
À propos de l'auteur.....	XIX
Remerciements	XX
Remarques de l'édition française.....	XXI
Terminologie	XXI
Remerciements du traducteur.....	XXI

Partie I - Introduction

1 Que sont la conception et l'architecture ?	3
Dans quel but ?	4
Une étude de cas.....	5
La signature d'un gâchis.....	7
Le point de vue de la direction.....	8
Un Titanic d'optimisme	9
Conclusion	11
2 Une histoire de deux valeurs	13
Le comportement	14
L'architecture	14
Choisir la plus grande valeur.....	15
La matrice d'Eisenhower	16
Un combat pour l'architecture.....	17

Partie II - Briques de base ou paradigmes de programmation

3 Aperçu des paradigmes	21
Programmation structurée.....	22
Programmation orientée objets.....	22

IV Architecture logicielle propre

Programmation fonctionnelle.....	22
Pas de liberté sans contraintes	23
Conclusion	23
4 Programmation structurée.....	25
Trouver la logique	26
Une mise en garde solennelle.....	28
Décomposition fonctionnelle	28
Où est la vérification formelle ?.....	29
La science vient au secours des mathématiques	29
Test et vérification	30
Conclusion	30
5 Programmation orientée objets.....	33
L'encapsulation	34
L'héritage	37
Le polymorphisme	39
Puissance du polymorphisme	41
Inversion des dépendances.....	42
Conclusion	44
6 Programmation fonctionnelle.....	45
La preuve par l'exemple	46
Immutabilité et architecture	47
Ségrégation de la mutabilité	48
Sourçage d'événements	49
Conclusion	51

Partie III - Principes de conception

7 Principe SRP de responsabilité unique.....	57
Symptôme 1 : duplication accidentelle.....	58
Symptôme 2 : changements concurrents.....	60
Solution	61
Conclusion	63

8	Principe OCP ouvert/fermé	65
	Une expérience de pensée.....	66
	Contrôle directionnel.....	70
	Masquage des informations.....	70
	Conclusion.....	70
9	Principe LSP de substitution de Liskov	71
	Une aide à l'utilisation de l'héritage.....	72
	Le problème du rectangle carré.....	72
	LSP et architecture.....	73
	Exemple de violation du principe LSP.....	74
	Conclusion.....	75
10	Principe ISP de ségrégation d'interface	77
	Principe ISP et langage.....	79
	Principe ISP et architecture.....	79
	Conclusion.....	80
11	Principe DIP d'inversion des dépendances	81
	Abstractions stables.....	82
	Fabriques (<i>factories</i>).....	83
	Composants concrets.....	84
	Conclusion.....	84
Partie IV - Principes des composants		
12	Nature des composants	87
	Bref historique des composants.....	88
	Des binaires relogeables.....	91
	Conclusion.....	93
13	Cohésion des composants	95
	Le principe d'équivalence réutilisation/déploiement REP.....	96
	Le principe de clôture commune CCP.....	97
	Le principe de réutilisation commune CRP.....	98
	Diagramme des tensions de cohésion des composants.....	99
	Conclusion.....	101

14 Couplage des composants	103
Le principe des dépendances acycliques ADP	104
La production hebdomadaire (<i>weekly build</i>)	104
Suppression des cycles de dépendances.....	105
Apparition d'un cycle dans le graphe des dépendances.....	107
Rupture du cycle infernal	108
Une structure frissonnante	109
Conception descendante ou montante ?	109
Le principe des dépendances stables SDP	110
Concept de stabilité	111
Métérologie de la stabilité.....	112
Certains composants ne doivent pas être stables.....	114
Composants abstraits	116
Le principe d'abstraction stable SAP.....	116
Où placer les règles de haut niveau ?	116
Approche du principe SAP	117
Mesure du niveau d'abstraction.....	117
La séquence principale	118
La zone de peine	119
La zone d'inutilité.....	120
Pour éviter les zones d'exclusion.....	120
Distance de la séquence principale.....	120
Conclusion	122

Partie V - Architecture

15 Qu'est-ce que l'architecture ?	125
Développement	127
Déploiement	127
Exploitation (opérations)	128
Maintenance.....	128
Garder les options ouvertes	129
Indépendance des périphériques.....	130
Pour des pourriels efficaces.....	132
De l'adressage physique	133
Conclusion	134

16 Indépendances	135
Cas d'usage.....	136
Développement	137
Déploiement	137
Laisser les options ouvertes.....	137
Découplage des couches	138
Découplage des cas d'usage	139
Modes de découplage d'exploitation.....	139
Indépendance du développement.....	140
Indépendance du déploiement.....	140
Duplications désirées.....	140
Modes de découplage (encore).....	141
Conclusion	143
17 Tracé des frontières	145
Deux histoires tristes.....	146
FitNesse.....	148
Où et quand tracer des lignes ?.....	150
L'être et le paraître d'un logiciel.....	153
Architecture à modules enfichables (<i>plug-ins</i>)	154
L'argument des enfichables.....	155
Conclusion	156
18 Anatomie des frontières	157
Franchissement de frontière	158
L'horrible monolithe	158
Composants de déploiement.....	160
Exétrons (<i>threads</i>).....	161
Processus locaux	161
Services.....	162
Conclusion	162
19 Règles et niveaux	163
Niveaux	164
Conclusion et exercice pratique	167

20 Règles métier	169
Entités.....	170
Cas d'usage.....	171
Modèles requête/réponse	173
Conclusion	174
21 Une architecture hurlante	175
Le thème d'une architecture.....	176
Le but d'une architecture	176
Architecture et modèle web.....	177
Un châssis est un outil, pas une philosophie	177
Des architectures testables	178
Conclusion	178
22 Pour une architecture épurée	179
La règle de dépendance.....	181
Entités	182
Cas d'usage	182
Adaptateurs d'interface.....	182
Châssis et pilotes.....	183
Seulement quatre cercles ?.....	183
Franchissement de frontière.....	183
Nature des données franchissantes.....	184
Un scénario typique	184
Conclusion	186
23 Présentateurs et objets humbles	187
Le patron objet humble	188
Présentateurs et Vues.....	188
Tests et architecture	189
Passerelles d'accès à la base	189
Mappeurs de données	190
Auditeurs de services (<i>listeners</i>).....	190
Conclusion	190

24 Frontières partielles	191
Omission de la dernière étape.....	192
Frontières unidirectionnelles.....	193
Façades.....	193
Conclusion	194
25 Frontières et couches	195
À la chasse au Wumpus	196
Une architecture épurée ?.....	197
Un flot de flux	200
Des flux en couches	200
Conclusion	202
26 Le composant principal main	205
Le détail ultime	206
Conclusion	209
27 Services grands et petits	211
Une architecture de services ?	212
Avantages des services ?.....	212
La promesse du découplage.....	212
La promesse de développement et de déploiement indépendants	213
Le problème Chatadopt des chatons en taxi.....	213
Des objets comme solution.....	215
Services basés sur les composants	217
Problèmes de transversalité.....	218
Conclusion	218
28 Frontières de test	219
Les tests sont des composants.....	220
Une conception favorisant les tests	220
Une interface API de tests	221
Un couplage structurel.....	221
Sécurité.....	222
Conclusion	222

29	Architecture propre et embarquée	223
	Un <i>test d'app-titude</i>	226
	Le piège de la dépendance matérielle.....	228
	Une architecture pour embarqué propre est testable	229
	Masquer les détails matériels au client de la couche HAL.....	231
	Programmation avec interface et interchangeabilité.....	236
	Directives de compilation conditionnelle répétées	237
	Conclusion	238

Partie VI - Détails

30	La base de données est un détail	241
	Bases de données relationnelles.....	242
	Cause de la domination des systèmes de bases de données.....	242
	Et s'il n'y avait plus de disques ?	244
	Détails magnétiques.....	244
	Mais qu'en est-il des performances ?.....	245
	Une anecdote.....	245
	Conclusion	246
31	Le Web est un détail	247
	Un progrès pendulaire	248
	Résultats.....	249
	Conclusion	250
32	Le châssis est un détail	251
	Motivation des auteurs	252
	Un mariage déséquilibré.....	252
	Les risques d'un mariage	253
	La solution	253
	Je vous déclare unis... ..	254
	Conclusion	254
33	Étude de cas : vente de vidéos	255
	Le projet.....	256
	Analyse des cas d'usage.....	256
	Architecture en composants	258

Gestion des dépendances	259
Conclusion	260
34 Le chapitre bonus	261
Paquetages par couches.....	262
Paquetages par domaines fonctionnels.....	264
Ports et adaptateurs.....	265
Paquetages par composants.....	267
Le diable est dans les détails d'implémentation.....	272
Organisation ou encapsulation.....	273
Autres modes de découplage.....	276
Conclusion.....	277
Postface	279

Partie VII - Annexe

Archéologie architecturale	285
Comptabilité syndicale.....	285
Ablation de résistances.....	292
Suivi de fonderie aluminium.....	295
4-TEL	296
L'ordinateur central SAC	301
Déterminé à distribuer.....	301
L'architecture du SAC.....	302
Une reconception non maculée	303
Le SAC de l'Europe	304
Une conclusion en cul-de-SAC.....	305
Le langage C	305
Le système BOSS.....	307
pCCU.....	307
Le piège de l'horloge.....	308
Le double affichage DLU/DRU.....	309
Architecture de DLU/DRU	310
Contrôle vocal avec VRS.....	311
Le choix du nom	312
L'architecture de VRS.....	312

XII Architecture logicielle propre

Conclusion de VRS	313
Le standardiste électronique ER	314
Le système d'attribution des interventions CDS	315
La start-up Clear Communications	318
Une configuration mentale	319
Oncle Bob	320
Un appel pour vous.....	320
ROSE.....	321
La discussion continue pendant les travaux	322
Le mieux et le bien	322
Les examens de l'ordre des architectes	323
Conclusion	325
Index	327