

ÉTUDES & DÉVELOPPEMENT

ARCHITECTURE LOGICIELLE

**Concevoir des applications simples,
sûres et adaptables**



Jacques Printz

Préface de Yves Caseau

2^e édition

DUNOD



2-004-366-1

ARCHITECTURE LOGICIELLE

**Concevoir des applications simples,
sûres et adaptables**

Jacques Printz

Professeur au Cnam, titulaire de la chaire de génie logiciel

Préface de Yves Caseau

2^e édition

DUNOD

Table des matières

Avant-propos	1
Partie 1 – Qu'est-ce que l'architecture du logiciel ?	
Chapitre 1 – L'architecture dans les sciences de l'ingénieur	9
1.1 L'architecture au sens littéral	9
1.2 Limites des métaphores architecturales	12
1.3 Architecture de l'information	15
1.3.1 <i>Le contenu informationnel d'une application</i>	20
1.3.2 <i>Description de l'architecture par un langage – Machine abstraite</i>	22
1.3.3 <i>Point de vue systémique et contraintes</i>	24
Chapitre 2 – Les matériaux de l'architecture logicielle	31
2.1 De quoi sont faits les programmes informatiques ?	31
2.2 Nature sémantique des constructions informatiques	39
2.2.1 <i>Organisation hiérarchique – Niveaux d'abstraction</i>	41
2.2.2 <i>Relations, interactions et couplages des entités architecturales</i>	49
2.2.3 <i>Synthèse</i>	52
2.3 Indépendances des données et des programmes	54
2.4 Tentative de définition de l'architecture	59
2.5 Terminologie introduite dans ce chapitre	65
Chapitre 3 – Propriétés indésirables des entités architecturales	71
3.1 Défauts et anomalies de fonctionnement des entités architecturales	71
3.2 Comportements dégénératifs des entités architecturales en cours d'exécution	78
3.3 Contrôles associés aux défaillances – Système de surveillance – Administration	83
Chapitre 4 – Représentations de l'architecture – Symboles architecturaux – Diagrammes d'architecture	85
4.1 Introduction – Différentes vues de l'architecture	85
4.1.1 <i>La vue acquisition</i>	85
4.1.2 <i>La vue engineering</i>	86

4.1.3	<i>La vue projet</i>	87
4.1.4	<i>Exigences concernant la représentation de l'architecture</i>	87
4.2	Les premières notations – Le monde de la programmation structurée	90
4.3	Les notations récentes – Le monde objet	93
4.3.1	<i>La notation SDL</i>	94
4.3.2	<i>La notation UML</i>	97
4.4	La liberté de l'architecte – La pragmatique des représentations	107
4.5	Organisation du référentiel d'architecture – Le référentiel comme méta-langage	110
Chapitre 5 – Place de l'architecture dans les projets informatiques		117
5.1	Cycle de vie d'un système – Cycle de développement	117
5.2	Rôle et place de l'architecte dans la relation MOA/MOE	120
5.3	Influence de l'architecte sur le retour sur investissement ROI	121

Partie 2 – Analyse de deux chefs-d'œuvre d'architecture

Chapitre 6 – Principes d'architecture des compilateurs		129
6.1	Le problème de la traduction des langages informatiques	129
6.1.1	<i>Analyse lexicale</i>	132
6.1.2	<i>Syntaxe concrète – Syntaxe abstraite</i>	134
6.1.3	<i>Les types</i>	136
6.1.4	<i>La génération de code et l'optimisation</i>	138
6.1.5	<i>La notion de langage intermédiaire pivot</i>	141
6.2	Cas des méta-informations	144
Chapitre 7 – Architecture des processus et de leurs interactions dans une machine		149
7.1	Le concept de processus	150
7.1.1	<i>Topologie d'un processus</i>	152
7.1.2	<i>Organisation de l'espace d'adressage d'un processus</i>	155
7.2	Les sémaphores et la communication inter-processus	161
7.2.1	<i>Dynamique des opérations de synchronisation</i>	164
7.2.2	<i>Synchronisation des processus – les sémaphores</i>	165
7.3	Les leçons : les contraintes systèmes et la recherche d'un équilibre économique	168

Partie 3 – Architecture fonctionnelle logique

Chapitre 8 – Principes et règles de construction des architectures fonctionnelles logiques		175
8.1.	Les processus du monde réel	175
8.2	Comment informatiser les processus métier	177
8.3	Les contraintes de l'automatisation et de la machinerie informatique	183

8.4	Organisation hiérarchique des intégrats – Vision statique de la machine informationnelle	191
8.5	Enchaînement des intégrats – Vision dynamique de la machine informationnelle	196
Chapitre 9 – Propriétés sémantiques des intégrats – Transactions – Services		201
9.1	Transactions	201
9.1.1	Le modèle classique des transactions ACID	203
9.1.2	Transactions longues – Compensation	211
9.2	Fonctions de services – Fonctions primitives	214
9.2.1	Généralités	214
9.2.2	Synthèse simplifiée des mécanismes d'appels des services	216
9.3	Sémantique de couplages et des interactions entre les intégrats	219
9.3.1	Nature des couplages	220
9.3.2	Couplage par les données	221
9.3.3	Couplage par l'ordonnancement et les événements	222
9.3.4	Couplage par l'environnement	223
Chapitre 10 – Quelques modèles d'architectures		225
10.1	Notion de machines informationnelles – Intégration de l'information	226
10.1.1	Le puits ou Machine réceptrice (MR)	226
10.1.2	La source ou Machine émettrice (ME)	227
10.1.3	La traduction ou Machine transductrice (MT)	228
10.1.4	Intégration de l'information – Couplage des machines	230
10.2	Architecture en couche	231
10.2.1	Les deux types de couches	232
10.2.2	Identification des fonctions de services et des interfaces entre couches	235
10.2.3	Modèle ETL (Extract, Transform, Load)	241
10.2.4	Modèle CRUD (Create, Retrieve, Update, Delete)	242
10.2.5	Modèle Client – Serveur/Services	245
10.3	Le modèle générique traducteur-transducteur TT	246
10.4	Modèle générique d'un moniteur système	254
10.5	Architecture REST	263
Chapitre 11 – Clients et serveurs		267
11.1	Machine informationnelle basée sur le pattern MVC	267
11.2	Machine informationnelle MVC en architecture distribuée	270
11.3	Structure des organes de la machine informationnelle	271
11.3.1	Structure Actions-Opérations	272
11.3.2	La nomenclature des types de données	272
11.3.3	La structure de contrôle	275
11.3.4	La structure de surveillance	276
11.4	Architecture SOA (Service-Oriented Architecture)	276

Partie 4 – Propriétés d'une bonne architecture

Chapitre 12 – Simplicité – Complexité	283
12.1 Fondements des mesures de complexité textuelle	283
12.1.1 <i>Complexité et/ou complication</i>	283
12.1.2 <i>Indicateur de complexité/complication</i>	290
12.2 Avantages et inconvénients des mesures textuelles	295
12.2.1 <i>Légitimité de la simplification</i>	298
12.2.2 <i>Les limites de l'approximation hiérarchique</i>	303
12.3 La complexité dans le quotidien des projets	305
12.3.1 <i>Complexité des données, fonctions, événements</i>	305
12.3.2 <i>Complexité de la construction des modèles de données</i>	307
12.3.3 <i>Complexité dynamique, interactions et couplages</i>	312
Chapitre 13 – Disponibilité – Sûreté de fonctionnement	319
13.1 Introduction	319
13.2 Notion d'intégrat testable – Testabilité	330
13.3 Reconstruire l'histoire d'une défaillance	335
13.3.1 <i>Analyse des prédécesseurs</i>	335
13.3.2 <i>Analyse des données</i>	335
13.3.3 <i>Analyse de l'environnement</i>	336
Chapitre 14 – Adaptabilité – Évolutivité	339
14.1 Introduction	339
14.2 Adaptabilité du point de vue des métiers et de la maîtrise d'ouvrage	340
14.3 Adaptabilité du point de vue de l'architecte	343
14.3.1 <i>Cas des adaptabilités métier</i>	346
14.3.2 <i>Cas des adaptabilités aux plates-formes et aux socles techniques (architecture physique)</i>	350
Chapitre 15 – Interfaces	353
15.1 Introduction	353
15.2 Rappel sur la notion d'interface	354
15.3 Cycle de vie et mise en œuvre	362
15.4 Évolution et compatibilité ascendante des interfaces	368
15.5 Interfaces externes et internes d'un intégrat agrégé	372
Chapitre 16 – Le métier de l'architecte : complexité, logique, intuition.	375
16.1 Comment poser et résoudre les problèmes d'architecture	375
16.2 L'architecte face à la complexité du réel	380
16.2.1 <i>Généralités sur la mesure de la complexité des systèmes et des architectures</i>	380
16.2.2 <i>La mesure : analogie physique</i>	381
16.2.3 <i>Quantité d'informations et codage</i>	387

16.2.4	<i>Vers une approche de la mesure de la complexité textuelle</i>	389
16.2.5	<i>Le problème de l'étalon de mesure</i>	392
16.2.6	<i>Considération sur les couplages et les graphes de couplages entre modules</i>	395
16.2.7	<i>Tests et complexité</i>	398
16.3	<i>La logique de l'architecte</i>	401
16.3.1	<i>Le concept d'ensemble prédictif de Russel-Poincaré</i>	402
16.3.2	<i>Rôle de la logique en programmation</i>	406
16.3.3	<i>Le concept de vérité en univers incertain</i>	409
16.3.4	<i>Logique et stratégie en cas de conflit d'accès aux ressources</i>	415
16.3.5	<i>Retour sur les couplages hiérarchiques et les architectures en couches : une logique pour le nommage</i>	421
16.4	<i>Synthèse : la complexité dans les projets – Guide de survie de l'architecte débutant</i>	423
	<i>Conclusion</i>	425
	<i>Sigles et acronymes utilisés</i>	430
	<i>Glossaire commenté</i>	435
	<i>Bibliographie</i>	443
	<i>Index</i>	449