

软件工程 2012 年 索引

2012 年 6 月 8 日 8:00-10:00 东上院 10\_

~	<b>4</b>	4+1	6.2-9
	<b>5</b>	5W1H	10-7
~	<b>A</b>	Abstraction	4.2-8
		Action (State)	6.2-40
~	<b>A</b>	Activity (State)	6.2-40
		Activity Diagram	4.2-24
~	<b>A</b>	Actor 定义	4.2-17
		Actor 识别	4.2-17
~	<b>A</b>	Ada	8-3
		Adaptable System	5-20
~	<b>A</b>	Aggregation	4.2-44, 4.2-45, 4.2-49
		ALGO	8-3
~	<b>A</b>	Alpha 测试	9-11
		Alternative Flow	4.2-21
~	<b>A</b>	Analysis Class	4.2-59
		APL	8-3
~	<b>A</b>	Application Layer	6.2-13
		Architectural Layer	6.2-11
~	<b>A</b>	Architectural Patten	5-19
		Assembler	8-3
~	<b>A</b>	Association	4.2-40, 4.2-55
		Association Class	6.2-43
~	<b>A</b>	Attribute Representation	6.2-42
		Awk	8-4
~	<b>A</b>	$\alpha$ 测试	9-11
~	<b>B</b>	Baseline	8-15
		Bash	8-4
~	<b>B</b>	BASIC	8-3
		Basic Flow	4.2-21
~	<b>B</b>	Beta 测试	9-11
		Blackboard	5-22
~	<b>B</b>	Blackboard 模式	5-19
		Boundary Class	4.2-61, 6.2-35
~	<b>B</b>	BPR	11-14
		Broker 模式	5-19
~	<b>B</b>	Brooks 定律	10-19
		Business Process Reengineering	11-14
~	<b>B</b>	Business Services Layer	6.2-13
		$\beta$ 测试	9-11
~	<b>C</b>	C	8-3
		C/S 架构	6.2-22
~	<b>C</b>	C++	8-3
		Check - in	8-17
~	<b>C</b>	Check - out	8-17
		Chief programmer team	10-19
~	<b>C</b>	CIM	1-33, 2-18
		Class	4.2-12, 6.2-3
~	<b>C</b>	Class (Operation)	4.2-70

~	<b>C</b>	Class (Relationship)	4.2-39, 4.2-70
		Class Diagram	4.2-34
~	<b>C</b>	Client/Server Architectures	6.2-22
		CM Repository	8-16
~	<b>C</b>	COBOL	8-3
		COCOMO	10-16
~	<b>C</b>	Code Documentation	8-11
		Collaborate	4.2-65
~	<b>C</b>	Command Labeling	7-17
		Communication Diagram	4.2-68
~	<b>C</b>	Communications Association	4.2-19
		Component	6.2-25
~	<b>C</b>	Component Diagram	6.2-25
		Composition	4.2-45, 4.2-45
~	<b>C</b>	Conceptual Class	4.2-36
		Conceptual Model	4.2-36
~	<b>C</b>	Connector	6.2-23
		Control Class	4.2-62, 6.2-36
~	<b>C</b>	CSH	8-4
~	<b>D</b>	Data View	6.2-26
		debugging	9-2
~	<b>D</b>	Defect density	9-6
		Delegation	4.2-52
~	<b>D</b>	Democratic team	10-19
		Dependency	4.2-54
~	<b>D</b>	Deployment View	6.2-20
		Design Classes	6.2-4
~	<b>D</b>	Design Pattern	5-19, 5-23
		Design Recovery	11-12
~	<b>D</b>	Design Reuse	5-17
		Distributed System	5-19
~	<b>D</b>	Distribution Patterns	6.2-21
~	<b>E</b>	Eiffel	8-3
		Encapsulation	4.2-9
~	<b>E</b>	Entity Class	4.2-59, 6.2-36
		Error handling	7-16
~	<b>E</b>	Expected Result	9-21
		Extend (Use - Case)	4.2-30
~	<b>F</b>	Facade Pattern	5-24
		Factoring	4.2-51
~	<b>F</b>	Final State	6.2-39
		Forth	8-3
~	<b>F</b>	FORTRAN	8-3
		Forward Engineering	6.2-44, 8-12
~	<b>F</b>	Framework	5-25
		Functionality	3-4
~	<b>F</b>	FURPS	3-4
~	<b>G</b>	Gang of Four	5-23
		Generalization (Actor)	4.2-33
~	<b>G</b>	Generalization (Class)	4.2-45
		Generalization (Use-Case)	4.2-32

~	<b>G</b>	GoF	5-23
		GOTO 语句	8-10
		Graphics User Interface	7-8
		GUI	7-8
~	<b>H</b>	Haskell	8-3
		Help facility	7-14
		Hierarchical team	10-19
		Hierarchy	4.2-11
~	<b>I</b>	Id	8-3
		Idiom	5-19
		Implementation	4.2-52
		Inheritance	4.2-52
~	<b>I</b>	Implementation View	6.2-24
		Include (Use - Case)	4.2-29
		Inherit	4.2-48
		Initial State	6.2-39
~	<b>I</b>	Interaction Diagram	4.2-66
		Interactive System	5-20
		Interface	6.2-5
		Internationalization	7-19
~	<b>J</b>	IT 黑洞	3-13
		J2EE	5-27
		Java	8-3
		JavaScript	8-5
~	<b>J</b>	JDBC	6.2-16
		JSF	5-28
		JSP Model II	5-26
		JSP Model II	5-26
~	<b>L</b>	Layered Architecture	5-21
		Layering Approach	6.2-10
		Layers 模式	5-19
		Liskov 替换原则	4.2-53
~	<b>L</b>	LISP	8-3
		Logical View	6.2-9
		LSP	4.2-53
		LSP	4.2-53
~	<b>M</b>	Menu	7-17
		Messages	4.2-65
		Microkernel	5-20
		ML	8-3
~	<b>M</b>	Model - View - Controller	5-20
		Modula	8-3
		Modularity	4.2-10
		Multimodal User Interface	7-9
~	<b>M</b>	Multiple Inheritance	4.2-48
		Multiplicity	4.2-40
		MVC	5-20, 5-20
		MVC 框架	5-25
~	<b>N</b>	Navigability	4.2-43
		Node (Deployment View)	6.2-22
		Node (Deployment View)	6.2-22
		Node (Deployment View)	6.2-22
~	<b>O</b>	Object Technology	4.2-2
		Object - orientation	4.2-4
		OO	4.2-4
		Operation (Class)	4.2-70

~	<b>O</b>	Operation Signature	6.2-37
		O-R Mapping	6.2-28
		ORM 框架	5-25
		O-R 映射	6.2-28
~	<b>P</b>	PAC	5-20
		Package	4.2-13, 6.2-6
		Partitioning	6.2-12
		Partitions	4.2-25
~	<b>P</b>	Performance	3-6
		Perl	8-5
		Persistence Design Pattern	6.2-15
		PHP	8-5
~	<b>P</b>	PIM	1-33, 2-18, 3-23
		Pipes and Filters	5-22
		PL/I	8-3
		PMBOK	10-3, 10-6
~	<b>P</b>	Presentation - abstraction - control	5-20
		Process	6.2-19
		Process View	6.2-19
		Prolog	8-3
~	<b>P</b>	PSM	1-33, 2-18
		Python	8-5
		Python	8-5
		Python	8-5
~	<b>R</b>	Rayleigh - Norden 曲线	10-18
		Reengineering	11-13
		Refactoring	5-16, 11-12
		Relational Data Model	6.2-27
~	<b>R</b>	Relationship (Class)	4.2-39, 4.2-70
		Reliability	3-5
		Response time	7-14
		Restructuring	11-12
~	<b>R</b>	Reuse	5-17
		Reverse Engineering	6.2-44, 8-12, 11-12
		Role (Class Diagram)	4.2-42
		Round - Trip Engineering	6.2-44, 8-12
~	<b>R</b>	Ruby	8-6
		RUP	2-19
		RUP	2-19
		RUP	2-19
~	<b>S</b>	Sandwich testing	9-8
		Scenario	4.2-22
		Scheme	8-3
		Script Language	8-4
~	<b>S</b>	Sequence Diagram	4.2-67
		Shell	8-4
		Significant State	6.2-42
		Single Inheritance	4.2-47
~	<b>S</b>	Smalltalk	8-3
		SNOBOL	8-3
		Software Evolution	11-3
		Software Maintenance	11-3
~	<b>S</b>	Special State	6.2-39
		Spreadsheets	8-3
		Spreadsheets	8-3
		Spreadsheets	8-3

~	S	SRS	3-25
		SRS 模板	3-32
		State	6.2-38
		State Machine Diagram	6.2-38
		Struts	5-27
~	S	Subsystem	6.2-5
		Subsystem Interactions	6.2-32
		Supportability	3-5
		SWEBOK	1-19
~	T	Tcl	8-5
		Test Case	9-21
		Test Factor	9-21
		Test Input	9-21
		testing	9-1
~	T	Thread	6.2-19
~	U	UI 类	6.2-35
		UML 对象表达	4.2-7
		UML 建模工具	1-38
		UML 类表达	4.2-12
~	U	UML 历史	4.2-4
		UML 模型	1-37
		UML 软件模型视图	1-33
		URPS	3-4
~	U	Usability	3-6
		Use - Case	4.2-15
		Use - Case Realization	4.2-58
		Use - Case Specification	4.2-20
~	U	Use - Case 技术	4.2-15
		Use - Case 技术 SRS 模板	3-32
		Use - Case 间的关系	4.2-28
		Use - Case 建模步骤	4.2-16, 4.2-16
~	U	Use - Case 模型组成	4.2-15
		Use - Case 识别	4.2-17
		Use - case 图	4.2-27
~	V	Val	8-3
		Version	8-15
		Visibility	4.2-55
		Vision	3-15, 3-25
~	W	WBS	10-14
		Web 页面建模	6.2-35
~	X	XSLT	8-3
		Y	Yes... But 现象
~	Y	Yes... But 现象	3-13
A	安	安全性	5-5
B	白	白盒测试	9-13
		版本	8-15
		版本控制	8-16
B	版	包	6.2-6
		包定义	4.2-13
		包含 (用例)	4.2-29
B	包	备选流	4.2-21
		边界类	4.2-61, 6.2-35

B	编	编程	1-22
		编程的误区	1-23
		编码的风格	8-9
		编码准则	8-8
B	变	变更控制程序	3-36
		并行开发	8-17
		波音 747	1-28
B	布	布局	7-17
		布鲁克斯定律	10-19
B	部	部署	1-24
		部署视图	6.2-20
C	采	采购管理	10-5
		菜单	7-17
C	参	参与者 定义	4.2-17
		参与者 识别	4.2-17
C	操	操作 (类)	4.2-70
		操作签名	6.2-37
		操作信息反馈	7-15
C	测	测试	1-23, 9-1
		测试 生命周期	9-4
		测试覆盖	9-13
		测试过程	9-20
C	测	测试计划	9-20
		测试技术	9-13
		测试目的	9-2
		测试驱动开发	9-1
C	测	测试输入	9-21
		测试因素	9-21
		测试用例	9-21
		测试终止	9-23
C	测	测试准则	9-2
		测试自动化	9-4
		测试组	9-3
C	层	层次架构风格	5-21
		层次型组织结构	10-19
C	产	产品定位	3-26
		产品特性	3-27
		产业现状	1-10
C	常	常用术语	3-18
		唱片播放器	6.2-40
C	成	成本管理	10-5
		成功的标准	10-4
		成功率	1-8
		成例	5-19
C	持	持久设计模式	6.2-15
		重构	5-16, 11-12
		重用	5-17
C	抽	抽象	4.2-8
		抽象 (设计)	5-8
C	出	出错处理	7-16

C	初	初始状态	6.2-39
		初学型	7-12
	传	传统技术 SRS 模板	3-32
D	单	单继承	4.2-47
		单元测试	9-5
	导	导览	4.2-43
	迭	迭代	2-20
D	定	定义	1-15
		定制软件	1-3
	动	动作 (状态机图)	6.2-40
		对象表达 UML	4.2-7
D	对	对象层次	4.2-11
		对象定义	4.2-6
		对象技术	4.2-2
		对象 - 类关系	4.2-13
D	多	多重度	4.2-40
		多继承	4.2-48
		多媒体用户界面	7-8
		多通道用户界面	7-9
E	二	二义性	3-9
F	发	发展	1-26
	反	反馈操作信息	7-15
F	泛	泛化 (参与者)	4.2-33
		泛化 (类)	4.2-45
		泛化 (类的关系)	4.2-0
		泛化 (用例)	4.2-32
	范	范围管理	10-5
	方	方案	4.2-22
F	防	防错处理	7-16
	非	非功能需求	3-4
F	费	费用分配比例	1-25
		费用管理	10-4
F	分	分布模式	6.2-21
		分布式系统	5-19
		分层	6.2-10
		分层模式	5-19
F	分	分工	10-18
		分解	4.2-51, 5-10
		分配比例	1-24, 1-25
		分析建模准则	3-22
F	分	分析类	4.2-59
		分析模型	3-23, 6.2-2
F	风	风格 (界面)	7-7
		风险管理	10-5
		风险驱动	2-19
	封	封装	4.2-9
F	冯	冯诺依曼语言	8-3
	服	服务对象 (界面)	7-3
	复	复合框架	5-25
		复用	5-17

F	复	复杂性	1-18
		覆盖 (测试)	9-13
G	概	概念类	4.2-36
		概念模型	4.2-36
		概要需求	3-25
G	杠	杠杆作用点	2-2
		高级程序设计语言	8-3
		工程定义	1-14
G	工	工作量分配比例	1-24
		工作细化结构	10-14
		功能需求	3-4
G	功	沟通管理	10-5
		构件	2-14, 6.2-25
		构件级设计	5-7, 6.2-29
G	构	构件图	6.2-25
		估算	10-15
		故障曲线	1-4
G	估	关键状态	6.2-42
		关联关系	4.2-40, 4.2-55
		关联类	6.2-43
G	关	关系 (类)	4.2-39, 4.2-70
		关系 (用例)	4.2-28
		关系数据模型	6.2-27
G	管	管道和过滤器风格	5-22
		管理过程	10-6
		管理活动	1-20
G	国	管理人员的误区	1-11
		国际化	7-19
G	过	过程抽象	5-9
		函数式语言	8-3
H	合	合成关系	4.2-45, 4.2-45
		合同项目的立项管理	10-12
		核心开发计划	10-14
H	核	核心准则	1-17
		黑板风格	5-22
		黑板模式	5-19
H	黑	黑盒测试	9-9, 9-17
		后置条件	4.2-26
		黄金规则 (界面)	7-4
H	后	回归测试	9-19
		混合方式测试	9-8
H	活	活动	1-20
		活动 (状态机图)	6.2-40
		活动图	4.2-24, 4.2-27
J	基	基本流	4.2-21
		基线	8-15
		基于构件的开发	2-15
J	基	基于构件组装	1-3
		基于约束的语言	8-3
		集成测试	9-7

J	集	集成管理	10-5
	计	计算无关模型	1-33, 2-18
J	记	记忆 (界面)	7-5
	技	技术活动	1-20
J	技	技术实现 (界面)	7-1
	继	继承	4.2-48
J	继	继承 实现	4.2-52
	驾	驾驭软件 (界面)	7-4
J	架	架构风格	5-4, 5-19, 6.2-10
	架	架构模式	5-19
J	架	架构设计	5-4
	监	监控 (项目)	10-23
J	检	检出	8-17
	检	检入	8-17
J	建	建模	1-31
	建	建模 Web 页面	6.2-35
J	建	建模的方法	1-34
	建	建模的重要性	1-31
J	交	交互图	4.2-27, 4.2-66
	交	交互系统	5-20
J	角	角色 (类图)	4.2-42
	脚	脚本语言	8-4
J	教	教育 (界面)	7-3
	阶	阶段	2-19
J	接	接口	6.2-5
	接	接口设计	3-31, 5-6
J	节	节点 (部署视图)	6.2-22
	结	结构 (团队)	10-18
J	结	结构化的分析模型	3-23
	结	结构化方法	1-36
J	解	解决方案	3-14
	界	界面框架	5-25
J	界	界面设计 典型错误	7-2
	界	界面设计 黄金规则	7-4
J	界	界面设计风格	7-7
	界	界面设计过程	7-11
J	界	界面设计模式	7-9
	界	界面设计模型	7-11
J	界	界面设计要素	7-13
	金	金三角	1-19
J	进	进程	6.2-19
	经	经验教训	10-26
J	经	经验模型	10-16
	经	经验模型 (估算)	10-16
J	净	净室软件工程	2-15
	竞	竞赛	8-13, 8-0
J	纠	纠错性维护	1-26
	聚	聚合关系	4.2-44, 4.2-45, 4.2-49

K	开	开发人员的误区	1-12
	可	可测试性	5-5, 11-9
K	可	可见度	4.2-55
	可	可靠性	3-5, 5-4
K	可	可理解性	3-10, 11-9
	可	可维护性	5-5, 11-8
K	可	可修改性	11-10
	可	可验证性	3-11, 3-11
K	可	可移植性	11-10
	可	可用性	3-6, 5-4
K	客	客户的误区	1-12
	控	控制复杂性	1-18
K	控	控制类	4.2-62, 6.2-36
	快	快速原型模型	2-11
K	框	框架	5-25
	扩	扩展 (用例)	4.2-30
L	扩	扩展点	4.2-27
	类	类	6.2-3
L	类	类 操作	4.2-70
	类	类 关系	4.2-39, 4.2-70
L	类	类表达 UML	4.2-12
	类	类定义	4.2-12
L	类	类 - 对象关系	4.2-13
	类	类设计	5-3
L	类	类图	4.2-34
	立	立项管理	10-8
L	联	联结器	6.2-23
	良	良好的工程	1-14
L	六	六拍项目	10-7
	逻	逻辑式语言	8-3
L	逻	逻辑视图	6.2-9
	螺	螺旋模型	2-12
M	冒	冒烟测试	9-18
	面	面向对象	4.2-5
M	面	面向对象步骤	4.2-5
	面	面向对象的分析模型	3-24
M	面	面向对象的基本原则	4.2-8
	面	面向对象方法	1-37
M	面	面向对象分析	4.2-4
	面	面向对象设计	6.2-2
M	面	面向对象语言	8-3
	面	面向方面软件开发	2-17
M	描	描述用户界面	3-31
	民	民主开发小组	10-19
M	命	命令标记	7-17
	命	命令式语言	8-3
M	模	模板 (语言)	8-3
	模	模块的优劣标准	5-14
M	模	模块独立	5-12
	模	模块化	4.2-10, 5-10

M	M	模	模式	5-18
		模	模型	1-30
		模	模型 (界面)	7-11
		模	模型的定义	1-30
		模	模型的多个视图	1-32
		模	模型的功能	1-31
		模	模型的特征	1-32
		模	模型的转换	1-34
		模	模型驱动开发	2-14
		磨	磨损	1-3
N	内	内聚 (定义)	5-12	
		内聚 (级别)	5-13	
O	耦	逆向工程	6.2-44, 8-12, 11-12	
		耦合 (定义)	5-12	
P	配	耦合 (级别)	5-13	
		配置复查	9-10	
P	喷	配置库	8-16	
		喷泉模型	2-13	
P	平	平台无关模型	1-33, 2-18, 3-23	
		评审方法	3-33	
P	瀑	瀑布型	2-5	
Q	期	期望结果	9-21	
		启动项目	10-8	
Q	前	前景文档	3-15, 3-26	
		前置条件	4.2-26	
Q	求	求助设施	7-14	
		缺陷报告	9-22	
Q	缺	缺陷密度	9-6	
		确认测试	9-9	
R	人	人力资源管理	10-5	
		人员时间权衡定律	10-19	
R	人	人员组织	10-19	
		软件产业现状	1-10	
R	软	软件的定义	1-2	
		软件的分类	1-5	
R	软	软件的特征	1-3	
		软件复杂性	1-7	
R	软	软件工程定义	1-15	
		软件工程核心准则	1-17	
R	软	软件工程战略性学科	1-17	
		软件故障曲线	1-4	
R	软	软件过程	2-3	
		软件过程的杠杆作用点	2-2	
R	软	软件过程知识体系	1-19	
		软件过程组成	2-3	
R	软	软件开发模型	2-4	
		软件面临的新挑战	1-6	
R	软	软件模式	5-18	

R	R	软	软件配置	8-14
		软	软件生存周期模型	2-4
		软	软件实践的问题	1-9
		软	软件事故	1-9
		软	软件危机的表现	1-8
		软	软件危机的原因	1-10
		软	软件误区	1-11
		软	软件项目	10-3
		软	软件项目计划	10-15
		软	软件需求规约	3-7, 3-30
R	R	软	软件演化	11-3
		色	色彩	7-18
S	S	设	设计	1-21, 1-22
		设	设计重用	5-17
		设	设计复用	5-17
		设	设计工程	5-1
		设	设计恢复	11-12
		设	设计接口	3-31
		设	设计类	6.2-4
		设	设计模式	5-4, 5-19, 5-23, 6.2-14
		设	设计模型	5-3, 6.2-2
		设	设计评审	5-16
S	S	设	设计用户界面	3-31
		设	设计约束	3-7
		设	设计质量	5-15
		涉	涉众定义	3-16
		涉	涉众需求	3-7
		沈	沈备军	每页
		生	生存周期模型	2-4
		生	生命周期 测试	9-4
		时	时间管理	10-5
		时	时序图	4.2-67
S	S	实	实践的问题	1-9
		实	实施 (项目)	10-22
		实	实体类	4.2-59, 6.2-36
		实	实现视图	6.2-24
		事	事故	1-9
		事	事件流	4.2-21
		视	视图	1-32
		适	适应系统	5-20
		适	适应性维护	1-26
		收	收尾 (项目)	10-25
S	S	书	书写用例	4.2-21
		熟	熟练型	7-12
		属	属性表示	6.2-42
		术	术语	3-18
		术	术语表	3-18
		数	数据抽象	5-8
		数	数据流语言	8-3

S	数	数据设计	5-3
		数据视图	6.2-26
	双	双向工程	6.2-44, 8-12
	说	说明式语言	8-3
T	特	特定平台模型	1-33, 2-18
		特殊需求 (用例)	4.2-27
		特殊状态	6.2-39
		特性的优先级	3-29
	调	调试	9-2
	通	通信 - 关联	4.2-19
		通信图	4.2-68
	统	统一软件过程	2-19
	图	图形用户界面	7-8
	团	团队结构和分工	10-18
W	外	外观模式	5-24
		外行型	7-12
		完善性维护	1-26
	微	微内核	5-20
	维	维护	1-25, 11-2
		维护成本	11-5
		维护工作量	11-6
		维护过程	11-7
	维	维护类型	1-26, 11-4
		维护问题	11-5
		维护无形代价	11-6
		维护组织结构图	11-7
	委	委派	4.2-52
	文	文档化	8-11
		文档需求	3-28
		无二义性	3-9
误	误区	1-11	
	误区 (编程)	1-23	
X	系	系统测试	9-11
		系统工程	1-20, 3-15
		系统响应时间	7-14
X	细	细化需求	3-30
	线	线程	6.2-19
		线性顺序模型	2-5
X	详	详细需求	3-25
	项	项目	10-3
		项目成功率	1-8
		项目的约束	3-17
		项目范围	3-28
	项	项目工作细化结构	10-14
		项目估算	10-15
		项目管理	10-1
		项目管理知识体系	10-3
	X	项	项目规模
项目监控			10-23
		项目结束	10-25

X	项	项目经理分级	10-21
		项目经理技能	10-20
		项目经理职能	10-21
		项目启动	10-7
X	项	项目前景文档	3-7
		项目实施	10-22
		项目收尾	10-25
		消息	4.2-65
X	消	消息框	7-2
		协作	4.2-65
		信息隐藏	5-11
		形式化方法	1-35
X	协	形式化方法模型	2-15
		行为建模	4.2-64
		性能	5-4
		性能需求	3-6
X	性	需求	1-21, 1-21
		需求变更管理	3-36
		需求出错	3-11
		需求的来源	3-19
X	需	需求定义	3-3, 3-25
		需求跟踪	3-36
		需求工程	3-2
		需求管理	3-35
X	需	需求获取	3-12, 3-20
		需求具有的特性	3-9
		需求评审	3-34
		需求驱动开发	3-8
X	需	需求收集	3-20
		需求验证	3-33
X	选	选择语言	8-6
		演化型	2-9
		演化型类型	2-9
		演化性	11-2
Y	演	验收测试	9-12
		业务服务层	6.2-13
		业务过程再工程	11-14
		业务解决方案	3-14
Y	业	一次性集成测试	9-8
		一致性 (界面)	7-5
		依赖关系	4.2-54
		艺术设计 (界面)	7-1
Y	异	异常结束	10-25
		易用性	5-4
		易用性设计 (界面)	7-1
		应用层	6.2-13
Y	应	应用领域 (语言)	8-3
		硬件故障曲线	1-4
		泳道	4.2-25
		用户分类 (界面)	7-12

Y	用	用户界面	7-1
		用户界面框图	4.2-27
Y	用	用户界面类	6.2-35
		用户界面描述	3-31
Y	用	用户界面设计	3-31
		用例	4.2-15
Y	用	用例规约	4.2-20
		用例技术	4.2-15
Y	用	用例建模步骤	4.2-16
		用例模型组成	4.2-15
Y	用	用例识别	4.2-17
		用例实现	4.2-58
Y	优	优化类	6.2-34
		由底向上测试	9-8
Y	有	有效性测试	9-9
		有用模型的特征	1-32
Y	愚	愚公移山	10-2
		语言选择	8-6
Y	浴	浴缸曲线	1-4
		预防性维护	1-26
Y	源	源程序的文档化	8-11
		约束	3-17
Y	运	运行	1-25
		再工程	11-13
Z	再	再工程过程	11-15
		增量型	2-7
Z	战	战略性学科	1-17
		正常结束	10-25
Z	正	正向工程	6.2-44, 8-12

Z	支	支持计划	10-14
		支持性	3-5
Z	知	知识体系	1-19
		质量	5-15
Z	质	质量范围	3-27
		质量管理	10-5
Z	中	质量因素	5-4
		中国的软件产业现状	1-10
Z	终	终止测试	9-23
		逐步求精	5-9
Z	主	主程序员小组	10-19
		专家型	7-12
Z	专	专用过程模型	2-14
		状态	6.2-38
Z	状	状态机图	6.2-38
		资产检查	10-26
Z	资	子系统	6.2-5
		子系统接口	6.2-32
Z	子	子系统设计	6.2-33
		自顶向下测试	9-8
Z	自	自动化测试	9-4
		自主研发的立项管理	10-9
Z	字	字符用户界面	7-7
		综合管理	10-4
Z	综	综合评估	10-26
		总结经验教训	10-26
Z	总	最广泛的模型	2-10
		最终状态	6.2-39
Z	最	作用点	2-2
		作	