

# Web 2.0与 信息系统复杂性变革

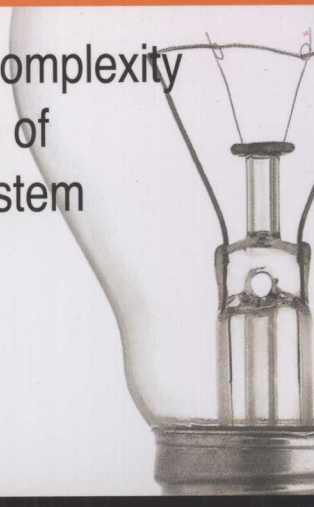
张树人 方美琪 著

Web 2.0 and Complexity  
Transformation of  
Information System



科学出版社

[www.sciencep.com](http://www.sciencep.com)



# Web 2.0 与信息系统 复杂性变革

张树人 方美琪 著

科学出版社

北京

## 内 容 简 介

随着信息技术的快速发展,信息系统也逐渐由低级到高级、由简单到复杂、由封闭孤立到开放协同地发展,信息系统的复杂性在近年来兴起的 Web 2.0 中表现得尤为突出。本书详尽分析了大量 Web 2.0 系统中的动态演化机制,提出了复杂适应信息系统范式;并结合一般性的系统思维和其他的系统理论与系统研究方法,提出了相应的概念模型、研究框架、设计的一般原则和具体方法、策略,并对复杂信息系统产生的一些新问题进行了专项研究。

本书可为电子商务、信息管理等相关专业的学者、研究生提供具有价值的研究借鉴,亦适合 IT 经理人、互联网产业研究与分析人员、技术媒体编辑以及对复杂性研究感兴趣的人士阅读和参考。

### 图书在版编目(CIP)数据

Web 2.0 与信息系统复杂性变革/张树人,方美琪著. —北京:科学出版社, 2008

ISBN 978-7-03-022697-6

I. W… II. ①张…②方… III. 主页制作—程序设计 IV. TP393.092

中国版本图书馆 CIP 数据核字(2008)第 119107 号

责任编辑:李欢 陈亮 苏雪莲/责任校对:张琪

责任印制:张克忠/封面设计:耕者设计工作室

科学出版社 出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

双青印刷厂印刷

科学出版社发行 各地新华书店经销

\*

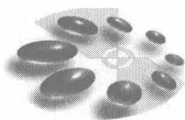
2008 年 12 月第 一 版 开本: B5 (720×1000)

2008 年 12 月第一次印刷 印张: 12 3/4

印数: 1—2 500 字数: 232 000

定价: 32.00 元

(如有印装质量问题,我社负责调换)



## 序

自 20 世纪 80 年代以来，系统复杂性作为一种世界规模的学科思潮，已经渗入到各门具体的学科中，成为科学技术领域内最热门的课题，体现了人类科学史、认识史上的重大转折。

圣菲研究所约翰·霍兰博士提出的复杂适应系统（CAS）理论，是一种认识、理解、管理、控制复杂系统的全新的思维方法体系，在过去十几年里，在各门具体的学科应用中获得了许多成效。对于信息管理与信息系统学科来说，把信息系统与互联网看做一个复杂适应系统、多主体组成的复杂协作关系网络，从而研究信息系统的基础架构与系统演化之间的关系——在国际上已经有了许多专门的研究。在美国，每年一度的“信息系统的哲学基础研究”学术会议，就是以此为专题的；在欧洲，欧盟合作技术框架专门组织了一批学者撰写复杂性研究对信息技术挑战的报告，指出在未来十年，复杂性科学的兴起将会支撑起具有巨大价值的新一代技术，欧洲应该在语义互联网领域内抓住这一机会等。一些从事复杂网络研究的学者也开始对信息系统，尤其是互联网中的开放性信息系统中的复杂性及其演化算法表示出极大的兴趣。例如，瑞士华人物理学家张翼成教授近年来一直对社会性网络服务中的信息复杂性问题投入极大的热情，并主持组织了中印欧复杂性研究暑期学校，推动这个交叉研究领域的合作。在国内，过去相关的研究大量是基于宏观信息化规划相关的组织管理研究，对于信息系统架构本身的复杂性研究一直以来还非常欠缺。正是这种欠缺，造成学术界在面临 Web 2.0 等互联网变革时，显得有些被动，这种被动甚至引发了业界对于学术界的失望与不满。

这本著作是方美琪教授和她的博士生张树人同志根据后者于 2005 年 11 月在信息系统协会中国分会成立暨第一次学术年会上提交的两篇论文——《社会性软件与复杂适应性信息系统范式》及《从发生与演化的角度研究社会性软件》——基础上扩展而成的，但绝不是一般性的扩展，而是具有更多的创新研究。无论是内容的广度与深度，还是与信息系统前沿实践相结合方面都大大推进了这两篇论文的研究，而这两篇论文在该次学术会议讨论中就获得了与会同仁的极大兴趣。

张树人同志在中国人民大学攻读博士学位期间，曾以“从社会性软件、Web 2.0 到复杂适应信息系统研究”为题进行了深入研究，其论文是国内第一篇自觉把复杂适应理论、复杂网络分析与社会网络分析应用到社会性软件与 Web 2.0 等新兴信息系统研究中的专门论文，可以看做是对这一领域内理论研究严重落

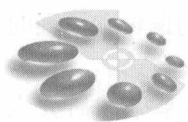
后于实践状况的适时回应。与以往研究信息系统复杂性时，人们常常人为地把信息系统中人的因素造成的管理复杂性、信息系统本身构造的复杂性，以及信息系统之间同创互生的生态复杂性分开进行研究不同，张树人同志提出的复杂适应信息系统范式的概念，把信息系统三个层次的复杂性纳入统一的研究范式中，并提出用社会网络分析方法、复杂网络分析方法、多主体建模方法研究信息系统三个复杂性的一般研究框架。这一理论与研究框架尽管可能会有些不成熟，但对于当前人们对 Web 2.0、社会性软件以及基于互联网信息系统演化发展变化的认识，却非常及时。其中对大量实例应用系统的非线性涌现现象的分析、涌现机制的模拟，以及社会复杂性造成信息系统算法评价的困难及解决办法等，都很值得本领域同行认真思考与严肃关注。

这一研究与传统的信息系统研究路向相比，具有很强的综合性、理论性和实践上的前沿性等特征，我与张树人同志的导师方美琪教授长期以来一直支持年轻的学者要大胆尝试跨学科的研究，并在这方面倾注了大量的心血。在该书付梓之际，我们也备感欣慰。

该书对不同学科与领域背景的读者可能会有不同的阅读角度，书中的部分内容在网络上公布后，短时间内就得到许多来自不同学科背景的读者自发的引用、转载和讨论，甚至引发一些争议，这也说明了大家对张树人同志这一研究关注的力度。该书对每位读者究竟有多大的帮助，我想关键取决于大家是否用心去读它。我的体会是，你用的时间越多，对信息系统在实践领域的发展前沿与发展趋势越了解，可能从该书中获得有益的思想启迪也就越多。我认为该书填补了信息系统架构设计中复杂性分析的空白，对实际信息系统项目的创新设计与评价也有很好的应用价值。当然，读者与作者一样，也有责任把它转化为实际的生产力，最终推动中国信息化的发展，特别是在这一轮的互联网变革中，提升我国信息产业在前沿领域创新的竞争力。

陈 禹

2008年5月于中国人民大学



## 前 言

复杂性研究被称为 20 世纪的科学思维革命,正在导致科学范式从二元论、简化还原论、线性思维到整体论、系统论和复杂性思维的转向<sup>[1~3]</sup>。同样,在信息系统研究领域,对日趋复杂的信息系统应用复杂思维方式思考、引入复杂性研究方法进行系统研究和设计,也必然导致一场观念的变革。近年来,互联网界关注的热点:Web 2.0(或互联网 2.0)正是这一观念变革在信息系统开发设计中的具体体现。

与科学领域中的复杂性范式主要由少数科学巨子发起与主导推动的情形不同,发生在信息技术和信息系统领域内的复杂性范式是自底向上发生的,由技术狂热群体推动发展起来的,具有明显的自发无序特征。正因为这场观念革命并非由学界或思想界发起,而是在实践中偶然发展起来的,在这个领域内理论研究远落后于实践,几乎没有正统的、前瞻性的相关学术研究,而很少的一些相关学术研究也都是对一些复杂技术形态以及涌现出的各类复杂现象的对象式分析。

尽管 Web 2.0 变革已经涌现出大量新的互联网应用模式,且已极大地改变了互联网的秩序与形态,甚至改变了人们在网络上的行为方式,但对于这一涌现变革发生的原因、未来发展的方向乃至对这一变革本身如何界定,都依然存在着很多争议。

这主要是由于这一变革是自底向上发生的,是由技术狂热群体推动发展起来的,具有明显的自发无序特征。大多数论述相关观念变革的文字资料,都来自于学院派之外的所谓草根阶层,以个人 Blog 或 Wiki 等非正式文本的方式呈现,整体表现为既缺乏逻辑条理性,又缺乏深度(相对于正式的学术论文而言)。在这一领域内,理论研究严重滞后于实践。

本书是尝试改变这个领域内理论研究落后于实践这一现状的一个努力。与其他一些著作将 Web 2.0 视为一种单纯的技术变革或一种纯粹的观念变革不同,本书的主要特点是用复杂性科学的系统思维方式,把信息技术和信息系统的研究纳入到复杂性研究的系统架构中去,进而对 Web 2.0 的动态复杂性产生的机制进行分析,并总结出这场变革的性质——开放式 Web 信息系统发展日趋复杂引发的必然变革。

从学科分类上看,本书属于信息技术与系统科学的交叉论著,除在信息系统架构设计与分析中借鉴复杂思维和复杂性科学研究方法,以提高信息系统设计与工程人员的系统思维素养外,本书为解决复杂信息系统中的一些实际问题发展了

部分复杂系统的研究方法和理论，并明晰提炼出多主体建模方法的逻辑和原理的分析，因此对系统科学专业研究人员理解多主体建模方法也有启迪意义。所以，本书建议的读者对象包括信息系统架构设计人员、信息系统哲学基础的研究人员、系统科学与工程类科研人员，以及对系统科学和复杂性研究感兴趣的工程类专业人士等。对读者的背景知识要求来看，一方面要求读者能对最前沿的互联网技术及应用有一些基本的了解，另一方面由于涉及大量系统科学和复杂性科学的理论，也需要读者具有基本的系统思维素养。

专业分工的细化会带来“见树不见林”的问题，一些专业研究的问题很可能在另外一些专业领域中有可借鉴的解决方案，不同专业间的互不了解（需要一定深度的专业背景才能理解）已越来越发展成为综合创新的障碍。是“专业余”（Pro-Am<sup>①</sup>）而不是“专门家”，更适合于跨专业的综合创新。本书内容跨越了不同学科分支的“专业余”，希望本书提出的理论体系能够在不同学科的专家之间搭建一个桥梁，让不同学科的专家们有一个共同的对话基础。

最后需要说明的是，虽然本书在写作过程中一直保持着对 Web 2.0 与互联网应用技术发展的前沿热点进行追踪，并不断地用复杂性科学系统思维方法对这些新技术形态、新应用发展趋势进行分析，同时跟踪关注国际 IT 思想界和相关学术界对这方面的研究现状，然而这一领域太新、发展太快、涉及各门学科的知识太多、涉及太多观念的转变，又如同一片芜杂丛生的热带丛林，除了一些真实的观念革新外，还充满了大量的杂音和商业上的炒作和噱头，而罕有严肃认真的深度学术思考供参考。作者虽然希望能做些理论开创性工作，并为此付出了许多的心血与努力，但毕竟各方面学识有限、力有不逮，最后完成的工作也只是初步的、不完备的尝试。希望本书能抛砖引玉，引起业界专家的重视，在本领域引发更深刻的、更有价值的理论研究。

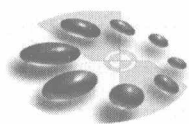
本书能够顺利出版，受到了杭州电子科技大学专著出版资金的资助，更得到了科学出版社的大力支持，在此一并表示诚挚的感谢！本书设置了网上交流园地 (<http://weblivinglab.com/cais>)，并提供书中部分模型源程序下载，我们热忱期待与各位读者的交流，以期共同进步！

作者

2008年10月于杭州

---

<sup>①</sup> 在 Web 2.0 混合文化中，“专业余”的作用十分瞩目。有人甚至提出 20 世纪是专家的世纪，21 世纪是“专业余”的世纪。Leadbeater 和 Miller 等对“专业余”的理念、力量、组织形式、未来发展趋势、对社会的影响等进行了全面分析。



# 目 录

序

前言

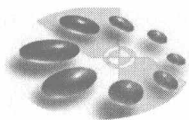
<b>第 1 章 绪论</b> .....	1
1.1 信息系统的系统研究概述 .....	3
1.1.1 信息技术与信息系统的的发展趋势 .....	3
1.1.2 系统科学在信息系统研究中的应用 .....	7
1.2 从 Web 2.0 热到复杂系统研究 .....	14
1.2.1 Web 2.0: 从狂热到困惑 .....	14
1.2.2 Web 2.0 与复杂适应系统理论 .....	17
1.2.3 本书的基本理论和方法 .....	18
1.3 本章小结 .....	19
<b>第 2 章 交互与涌现——Web 2.0 相关技术的深度解析</b> .....	20
2.1 Web 2.0 中的适应性和涌现 .....	21
2.1.1 Blog: 万维网中的适应性主体 .....	21
2.1.2 Wiki: “共同规范”的涌现 .....	33
2.1.3 社会性标签系统: 分众分类的涌现 .....	34
2.1.4 网络社群的自组织与社会关系网络的涌现 .....	37
2.2 Web 2.0 系统的经典架构的系统动力学分析 .....	39
2.2.1 主体参与式架构的典范——豆瓣评论 .....	40
2.2.2 开放式架构的典范——可编程的 Web 与 Web 2.0 组合工厂 .....	43
2.2.3 代表未来开源系统架构的 Ning .....	46
2.3 从社会性软件到 Web 2.0 .....	48
2.3.1 社会性软件的由来 .....	49
2.3.2 社会性软件的定义 .....	50
2.3.3 社会性软件的发生过程及判别标准 .....	51
2.3.4 社会性软件分类 .....	53
2.4 Web 2.0 的范畴 .....	54
2.5 本章小结 .....	58
<b>第 3 章 复杂适应信息系统范式</b> .....	60
3.1 系统复杂性研究与信息系统的复杂性 .....	61



3.1.1	系统复杂性研究产生的背景及研究的对象	61
3.1.2	信息系统的复杂性分析	62
3.1.3	信息系统复杂性所带来的机遇与挑战	65
3.2	复杂范式及复杂适应信息系统范式	66
3.2.1	科学研究的复杂范式	66
3.2.2	信息系统的范式	67
3.2.3	复杂适应信息系统范式	68
3.2.4	CAIS 范式与简单信息系统范式的比较	75
3.2.5	CAIS 设计的指导原则	76
3.2.6	CAIS 范式的适用范畴	81
3.3	本章小结	82
<b>第 4 章</b>	<b>复杂适应信息系统与复杂网络</b>	<b>84</b>
4.1	复杂网络分析	86
4.1.1	复杂网络分析的背景	86
4.1.2	复杂网络研究的相关概念	86
4.2	复杂适应信息系统中的复杂网络分析算法	89
4.2.1	2 模式网络及拆分算法研究	89
4.2.2	3 模式网络及其分析策略	97
4.3	复杂网络分析与复杂适应信息系统的架构设计	104
4.4	本章小结	111
<b>第 5 章</b>	<b>复杂适应信息系统的动态原型与建模</b>	<b>112</b>
5.1	多主体建模与 CAIS 的动态原型概述	113
5.1.1	多主体系统及多主体建模	113
5.1.2	多主体模型的原理及特点	114
5.1.3	两种层次的 CAIS 动态建模	116
5.2	CAIS 建模实例	118
5.2.1	实例 1: 适应性与非适应性页面	119
5.2.2	实例 2: 适应性规则	124
5.2.3	模型结论的应用推广	130
5.3	多主体建模在 CAIS 算法测试中的应用	131
5.3.1	模型设计	132
5.3.2	模拟实验	136
5.4	本章小结	137
<b>第 6 章</b>	<b>复杂适应信息系统生态及衍生模式</b>	<b>139</b>
6.1	信息系统生态概述	140

6.1.1	信息系统生态的学缘分析	140
6.1.2	信息系统生态的基本概念	142
6.2	信息系统的衍生模式及其应用	144
6.2.1	“收割者”模式及“ $N:1 \rightarrow N:M \rightarrow N:1:M$ ”中介产生模式	144
6.2.2	“收割者”模式及中介产生模式的应用	148
6.2.3	基因混合模式	151
6.2.4	系统功能综合集成模式及系统协作生态网络	153
6.2.5	系统功能综合集成模式与 SOA 的区别	157
6.3	本章小结	159
<b>第 7 章</b>	<b>涌动的变革——包罗万象的 Web 2.0 应用</b>	<b>162</b>
7.1	社会计算与群体智能类	162
7.1.1	社会化个性新闻服务:SpotBack	162
7.1.2	社会化音乐分享服务:Last.fm 和 Pandora	163
7.1.3	社会化软件推荐服务:Wakoopa 和 Soft1001	164
7.1.4	网络行为的社会化聚合服务:BlueDot	164
7.1.5	基于社会群体智能和社会化协作的 RSS 萃取服务	165
7.1.6	增加社会化评价、评论或注释的服务	165
7.1.7	增强社会化网络服务功能的插件:Widget	166
7.1.8	集合群体智力解释词条的社会化在线字典服务:WordSource	168
7.2	综合衍生类	168
7.2.1	在线标签管理服务:Tagsahoy	168
7.2.2	社会性书签提交代理	169
7.2.3	在线 RSS 发布代理	170
7.2.4	社会性书签、社会性网络和 RSS 在线阅读的综合集成服务	170
7.2.5	聚合各种社会化软件服务的服务:Mugshot	171
7.2.6	Widgy 插件综合服务	171
7.2.7	专注于提供各类基于 RSS 的操作和接口转换的服务	171
7.2.8	其他 RSS 衍生服务	172
7.3	手机和即时通信类	173
7.3.1	手机定位及分享的社会性服务	173
7.3.2	手机作为未来 Web 2.0 的应用终端	173
7.3.3	即时通信的插件和社会化扩展	174
7.4	信息系统生态群落类	174
7.4.1	Twitter 及其模仿克隆系列	174
7.4.2	Twitter 衍生服务	175

7.4.3	在线知识问答类的细分衍生	176
7.4.4	社会性书签的模仿与衍生	176
7.4.5	从社会性标签到社会性搜索引擎	177
7.4.6	Digg 类的模仿与改进	178
7.5	在线办公处理类	178
7.5.1	常见办公处理软件替代服务	178
7.5.2	在线日程和事务管理类	179
7.5.3	在线图形、图像处理类	180
7.5.4	在线流程图制作工具:DrawAnywhere 等	180
7.5.5	在线多媒体混烧服务:Sayjoy、Eyespot、Jumpcut 等	181
7.5.6	其他办公相关处理类	181
7.6	Web 2.0 的企业级应用	181
7.6.1	Yahoo 的 Pipes	181
7.6.2	Adobe 的 Apollo	182
7.6.3	微软的 Silverlight 与 Popfly	182
7.6.4	BEA 的 WebLogic Portal 10	182
7.7	本章小结	183
<b>参考文献</b>		184



自幼我们就被教导把问题加以分解，把世界拆成片片片段来理解。这显然能够使复杂的问题容易处理，但是无形中，我们却付出了巨大的代价——全然失掉对“整体”的连属感，也不了解自身行动所带来的一连串后果。

——彼得·圣吉（《第五项修炼》之“破镜重圆”<sup>[4]</sup>）

信息技术是一个快速更新和迅猛发展的领域，新技术和新思想层出不穷，系统模式和系统架构日新月异，项目管理、软件工程和系统开发设计方法等不断地推陈出新。随着互联网的兴起、知识生产全球化分工合作的进一步细化和深化，不同学科的知识综合与技术扩散也更加方便快捷，信息技术的发展呈现出“自催化”加速发展的态势。信息系统的复杂度与信息技术的发展密切相关，随着技术的进步和累积，信息系统不断地由低级到高级、由简单到复杂、由封闭孤立到开放协同地发展。

面对越来越复杂多样化的信息技术和信息系统，人们逐渐发现习惯性的片段思维和传统的基于“高效源于控制”的简单思维方式越来越难以处理和适应这种日益增长的复杂性<sup>[4]</sup>。具体表现有：限于技术细节，只见树木不见森林，而忽视信息系统设计的整体目标，甚至患上“高技术气量狭窄”综合征（The “Hi-Tech Hidebound” Syndrome）（Malhotra 1999）<sup>[5]</sup>，对问题的非技术解决策略持敌对或偏见态度；限于局部思考，不能充分考虑信息系统外部环境的动态变化，很少考虑或很难科学地考虑同其他系统间互操作与协同工作，造成项目设计成功后在应用推广与实施中失败；对信息技术和信息系统的的发展趋势缺乏认识，选择了错误的技术方案或落后的设计模式与理念；简单思维难以理解社会性软件和 Web 2.0<sup>①</sup> 领域内涌现的各种复杂<sup>②</sup>现象及系统内的非线性系统动力学机制，习惯于孤立地看事物的思维方式，不能清楚地认识 Blog 等简单系统中一些同样非常简单的技术中蕴涵着改变万维网拓扑结构和互联网秩序的巨大潜力，等等。

① 社会性软件和 Web 2.0 这两类新生事物非常复杂，不同的人有不同的理解，因而难以简单概括解释。在第 2 章我们会在分析的基础上再作具体的解释说明。

② 涌现是一种自行组织起来的结构、模式、形态，或者它们所呈现的特性、行为、功能，不是系统要素所固有的，而是相互作用的产物（效应）。涌现是复杂系统的特征和复杂性研究的中心词。后文还会结合具体信息系统中的涌现进行解释说明。

这些问题不仅造成了大量信息化项目的失败（或因为无法满足不断变动发展的用户需求和不断变化发展的应用环境造成的成效不高，或信息孤岛问题而妨碍了信息资源的有效利用，或陷入无休止的“设计—用户培训—需求和技术应用环境变化的反馈—重新设计或升级”循环的泥沼），还遮蔽了人们的眼睛，让人们对新技术的发展动向和信息系统的发展趋势缺乏前瞻性；在大量涌现的新观念、新技术、新架构、新概念面前难以作出审慎明智的判断抉择，表现出亦步亦趋地盲目跟风或不知所措，在真正的机遇到来之时却表现得完全漠然或无动于衷。

全面考察信息技术的发展规律和信息系统的一般演化规律，并自觉地借鉴其他系统演化发展的规律，应用一般系统论的研究方法来归纳和总结现有信息技术的发展规律及趋势，重新审视现有信息系统的一般理念、开发设计和管理的一般方法和原则，成为一些信息系统理论研究者追求的目标，也成为信息系统领域最有挑战性的研究议题<sup>①</sup>。欧盟技术合作框架项目（frame project, FP）的60多位研究者在“回应当复杂性科学给信息技术带来的机遇和挑战”的工作报告中指出：目前的方法已经无法应对信息技术未来的发展，只有应用基于复杂性思维方式的新技术才能给出一个解决方案，在未来十年，复杂性科学的兴起将会支撑起具有巨大价值的新一代技术，欧洲不能承受失去这次机会的可能<sup>②</sup>。这份报告对信息技术未来发展的可能路径（future pathways for IT）进行了系统分析<sup>[6]</sup>。

本书以近两年突然爆发兴起的社会性软件和 Web 2.0 为应用研究背景，以系统思维和系统理论为指导，把系统研究（特别是复杂性系统研究）的一些研究方法应用到信息系统的架构设计中去。在系统剖析社会性软件和 Web 2.0 发生、发展的内在成因基础上，重点研究如何从各种社会性软件和 Web 2.0 中抽取出系统间共同的行为规范、共同的动态演化机制和系统形态的演进模式，具有普遍适用性的设计原则，可推广的设计方法、设计理念以及适用技术等，并上升到理论高度，结合系统思维、系统理论和方法加以整合，参照复杂科学范式，提出一个新的信息系统范式；旨在为日趋复杂的信息系统提供规范化的开发方法、指导性的设计原则和启发性的技术创新依据，并为研究信息系统生态<sup>③</sup>提供基础理论框架和一般研究方法。

---

① 这类研究的专门分类称谓是信息系统中的哲学基础研究（philosophical foundations of information systems, PFIS）。2005年，美国第11届信息系统会议对9年来的此类研究进行了总结和回顾。见：Proceedings of the Eleventh Americas Conference on Information Systems, Omaha, NE, USA August 11~14th 2005, [http://aisel.isworld.org/subject.asp?Publication\\_ID=36](http://aisel.isworld.org/subject.asp?Publication_ID=36)。

② 原文在该报告的开篇：The emerging science of complex systems will underpin new generations of technology in the next decade of enormous value. Europe cannot afford to miss out on this opportunity（参见文献[6]）。

③ 信息系统生态指信息系统之间越来越错综复杂的交互协作和依赖关系让信息系统群体呈现出类生态系统的种群形态。

在信息系统的定义中,不同角色的用户也是系统的构成要素。“信息系统”这一称谓本身就有社会-技术综合性系统工程的含义,但片段思维和局部思考<sup>[4]</sup>让人们在设计信息系统或实施信息系统工程时常常不自觉地忽略了系统中人和社会的因素,表现出只重视技术和算法而轻视对用户群体智力和经验输入的综合,只重视系统功能设计而忽略用户之间协同协作的需要,只重视系统本身功能的设计而轻视系统之间的交互协作等倾向。社会性软件和 Web 2.0 的出现正在开始改变这种状况。社会性软件和 Web 2.0 在设计中包含了主体参与式架构 (participative architecture<sup>①</sup>)、社会关系网络和复杂网络的分析与设计、复杂的非线性协作机制、多层次的反馈循环机制以及开放的系统间交互机制等,把人的行为设计、社会关系、组织设计和人群协作机制、资源网络的动力学演化机制纳入到系统之内。然而,也正是这种人-机综合设计的整体系统观,让信息系统无论是内部结构的演化机制,还是外部由系统间多样化协作带来的协同进化机制,都变得复杂起来,给人们的科学理解和认识带来很多困难,从而制约了这方面的研究,也造成这个领域无序盲目的发展现状。本书正是对这一问题的回应,试图在信息系统的架构研究与设计中,全面引入系统思维、系统理论和系统方法,以加强对日趋复杂的信息系统的研究及开发设计的支持。正如当代著名思想家、法国国家科学研究中心名誉导师埃德加·莫兰 (Edgar Morin) 所言:“对复杂事物的研究和设计必须要用复杂性思维方式”<sup>[2,3]</sup>,对 Web 2.0 这一综合用户社会关系网络、知识关联网络、系统交互网络,并存在多层次间反馈关系复杂对象的研究和设计,也必须要用复杂性思维方式进行思考与设计。例如,应用群体社会心理分析进行以用户社会交互体验为中心的设计,以社群关系网络的分析计算 (社会计算) 来实现注重用户参与之后群体智能的涌现设计,以复杂网络分析算法实现系统内知识网络与内容网络关联及拓扑结构的优化,用系统动力学分析和设计系统各个要素和环节间存在的综合反馈 (如人-机智能的反馈) 等。

## 1.1 信息系统的系统研究概述

### 1.1.1 信息技术与信息系统的的发展趋势

信息技术相关的知识更新是最快的,没有哪项技术能够有信息技术那样的发

---

<sup>①</sup> 可参照 Tim O'Reilly 的 *An Architecture of Participation*, <http://www.oreillynet.com/pub/wlg/3017>。参与式架构原指系统架构设计面向公众开放,允许公众参与系统的改进或设计,描述开源软件的开发模式。本书重新限制了这一术语,强调系统功能中人的重要性,在系统运行中允许公众参与,可同时丰富系统的内容并改进系统功能的系统架构。Tim O'Reilly 是美国 O'Reilly 出版社的总裁、美国 IT 业界公认的传奇式人物、“开放源码”概念的缔造者,一直倡导开放标准,并活跃在开放源码运动的最前沿,也是 Web 2.0 概念的提出者。

展速度，信息系统及其开发设计方法都处在快速演变的进程当中。

从信息系统形态的演化发展来看，信息系统从最早的单用户、单机系统（或一台主机带多台终端），发展到主要应用于局域网或封闭网络环境的客户机/服务器（C/S）模式的系统，再到主要应用于开放式互联网络环境的浏览器/服务器（B/S）模式的系统，以及在 B/S 基础上扩充的三层架构和多层架构，最近又开始出现所谓的胖客户端模式及去中心化的 P2P 对等网络模式，信息系统正朝着越来越个性化、智能化、分布式去中心化的方向发展。

在信息系统演变的同时，信息系统的开发设计方法也经历了从面向过程、面向对象、组件设计、面向主体到面向服务（SOA<sup>①</sup>）的发展。以前在结构化设计中，模块是一个不能独立工作的函数。随着模块向可重用的对象、可方便集成的组件、可分布式组合的独立子系统的方向发展，系统组分（即系统的组成部分，下同）的独立性越来越高。系统架构由完全的自顶向下设计，到越来越多地选用自底向上的方法构建；从只注重系统本身的功能完备性和独有专属的信息格式，到越来越注重与其他系统间的协作能力和信息处理格式的标准化。这里有两个看上去背向发展的趋势：一方面，系统越来越注重平台无关性，减少自己对底层系统的依赖，增加了系统自身的独立性（如可跨平台运行的基于 Java 开发的软件、基于 Web OS<sup>②</sup> 的 Javascripts 开发的信息系统等）；另一方面，则采用方便交互标准交换格式和标准协议，为第三方系统软件形成互动提供方便，从而增强与其他信息系统之间的协作能力，且在增加了与别的信息系统之间的相互依赖的同时，部分降低了自身的完整性和独立性（如寄生在 Outlook 上的 Plaxo 软件<sup>③</sup>，就是在 Outlook 开放的功能扩展接口之上构建的）。其实这两个趋势并不矛盾，前者增加系统的独立自主性，后者增加系统的自主交互性。系统只有具有平台无关的独立性，才可能和更大范围内别的信息系统协作交互；系统只有尽可能地开放交互协作接口，才可能获得来自各种平台的支持，从而能广泛应用于各种平台而不受具体平台的制约。

信息系统及设计方法的发展可总结为以下几个趋势。

(1) 从系统开发到系统集成（from system development to system integration）<sup>[7,8]</sup>。程序设计和系统架构由最初的面向过程、面向对象（object oriented

---

① 面向服务的体系结构（service-oriented architecture, SOA）是一个组件模型，它将应用程序的不同功能单元（称为服务）通过这些服务之间定义良好的接口和契约联系起来。接口采用中立的方式进行定义，应该独立于实现服务的硬件平台、操作系统和编程语言。这使得构建在各种这样的系统中的服务可以以一种统一和通用的方式进行交互。

② 参见 <http://en.wikipedia.org/wiki/webos>。

③ <http://www.plaxo.com/>，中文介绍可参考 <http://www.chedong.com/blog/archives/000643.html>。

program, OOP) 和组件设计到面向主体 (agent oriented program, AOP), 再到面向服务的体系结构 (service-oriented architecture, SOA), 系统的基本构件越来越大: 从函数到对象库, 到控件、中间件或分布式控件 (active X)、插件 (plug-in) 等, 再到智能 Agent, 最后是 Web-Services 和完全独立的子系统, 系统的设计与开发越来越依赖于集成。

(2) 从以程序为中心到以信息内容 (数据) 为中心 (from program centered systems to information contents centered systems) 或以用户为中心 (human-centered systems)<sup>[8]</sup>。一些人甚至提出要用迅件 (infoware) 来取代软件 (software) 这一概念<sup>①</sup>。

(3) 从孤立封闭的集中控制式系统到协同开放的分布式系统。以前 C/S 体系是端到端封闭的体系结构, 必须使用特定的客户端; B/S 体系虽然打破了对客户端的限制, 但服务端的集中控制体系仍然没有开放, 功能或规模的扩展依然受限制; P2P 体系结构则把系统从集中控制的服务体系中解放出来, 打破了服务器专有、专属的限制, 成为开放式分布的体系, 所有的组成结点都可以是对等的, 任何结点都可以自由地加入和退出; 传统的 P2P 体系依赖于特定的协议和服务类别, 系统虽然是开放的, 但集成的结点依然限制为同质 (homogeneous) 的; 多智能主体系统 MAS (multi agent system) 通过智能主体间交互协议语言的标准化、面向服务的架构 SOA 通过 Web-Services 间互操作 (inter-operability) 标准的制定打破了这个限制<sup>②</sup>, 使开放的异质结点 (heterogeneous) 的集成体系成为可能。

(4) 系统的外在表现方面, 从设计者固定的界面到用户可调节的界面, 再到适应性用户界面等。

表 1-1 以系统的组分、组分的耦合度及组分的可重用性和独立性为指标, 对各系统设计方法进行比较。

表 1-1 系统设计方法对应组分的特征对照

系统设计方法	组分	耦合度	重用性和独立性
面向过程	函数 function	很高	很低
面向对象	对象 object	较高	低; 依赖于特定的程序开发语言
组件设计	控件 component	中	较低; 依赖宿主语言, 集成需要二次开发
面向主体	主体 Agent	较低	中; 交互性依赖 Agent 的交互协议

① 参见 <http://www.infoware.com/>.

② 参见 <http://ws-i.org>.



续表

系统设计方法	组分	耦合度	重用性和独立性
面向服务 SOA	Web-Services	低	较高；接口标准化，但非开放，Web-Services 的实现和升级控制在原开发者手中，由系统架构师或系统集成的技术工程是负责集成
社会性软件/ Web 2.0 时代 的系统设计	“人件” (humanware) 和独立功能子系统 (Web 2.0 部件)	很低	高；接口标准化且开放，与底层实现无关，系统具有生态多样性，功能细分为用户视角中的一个独立任务，最终用户的操作行为决定其相互之间的操作与组合方式

信息系统的组分在规模上越来越大、功能上越来越独立、组分间的耦合度越来越低。这符合系统由低级向高级发展的一般规律：越低层次的系统，其组分越简单，各组分之间相互依存性越高，组分间的关联越紧密固化、难以分解，且一旦分解还难以恢复，如无机化合物、单细胞生物体、机械系统等；越高层次的系统，其组分越复杂，组分间的独立性越高，组分间的关联协作模式越灵活自主，容易分解并且能够在分解后重新组合、恢复系统，如有机化合物、多细胞生物体、生物群体、人类社会经济系统等<sup>[9]</sup>。信息系统的复杂性也在这个发展过程中不断升级，在智能 Agent 和 Web-Services 之前的早期信息系统可以看做是一些函数的机械焊接，或一些模块构件的固定装配，或同质个体的重复叠加，信息系统更多地具有机械系统的特征，可称为信息系统的初级阶段；智能 Agent 和 Web-Services 大规模应用时期，信息系统则可以看做是一些具有完整功能，并有一定独立性的系统之间的松散联盟，信息系统具有相当的复杂度和灵活度，初步具有一些类生命系统的特征。

在社会性软件和 Web 2.0 中，由于采用了主体参与式架构，更加具有自主性的“人件”<sup>①</sup>引入系统设计，让这些新信息系统发展成为社会-技术混合系统，从而具有了社会系统的复杂性。

随着信息系统组分独立性和自主性的增强，组分间更灵活方便的搭配组合让系统具有越来越高的适应性，信息系统也越来越接近于生命系统、社会经济系统等复杂系统。因而，在其研究设计中也越来越多地需要应用和借鉴那些曾在生物学、生态学和社会科学等领域的复杂系统研究中获得广泛应用的系统方法和系统理论。

事实上，这方面的研究尝试也越来越多，下一节我们将对系统科学应用于信息系统研究的现状进行基于文献的回顾与分析。

① 这一术语的意义可参考 <http://www.humanware.com>.