

物联网工程研究丛书

事件驱动的物联网服务 理论和方法

章 洋 毛艳芳 著



科学出版社

物联网工程研究丛书

事件驱动的物联网服务 理论和方法

章 洋 毛艳芳 著

科学出版社

北 京

内 容 简 介

随着智慧城市、智能电网和工业 4.0 的发展,感知设备日益多样、感知数据日益丰富、用户需求日益个性化,而物联网服务生成与运行理论和方法的系统性滞后阻碍了物联网应用的发展。

本书首先引入资源模型,将物理系统表现在信息世界中,使物联网资源模型成为连接物理世界和信息世界的桥梁,而事件就是行驶在这些桥梁上的“交通流”。在物联网资源模型基础上,将异构物联网资源通过适配的方式自动接入信息系统。本书采用统一消息空间作为服务总线统一接纳路由服务与感知事件。对基于统一消息空间的单个服务,本书采用事件驱动的方式将其组织在一起,并解决其中集成和扩展的理论问题。在既有事件和服务的基础上,本书阐述基于知识推理和混合学习的方法,从孤立事件中提取复合知识,用其改进服务提供质量,包括自动服务生成和服务可适性调整。同时对物联网服务安全的独特性进行阐述。

本书适合于物联网专业研究生和从事该行业的科研与工程人员阅读。

图书在版编目(CIP)数据

事件驱动的物联网服务理论和方法 / 章洋, 毛艳芳著. —北京: 科学出版社, 2016

(物联网工程研究丛书)

ISBN 978-7-03-050611-5

I. ①事… II. ①章… ②毛… III. ①互联网络—应用 ②智能技术—应用 IV. ①TP393.4 ②TP18

中国版本图书馆 CIP 数据核字(2016)第 271144 号

责任编辑: 陈 静 董素芹 / 责任校对: 郭瑞芝

责任印制: 张 倩 / 封面设计: 迷底书装

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>

新科印刷有限公司 印刷

科学出版社发行 各地新华书店经销

*

2016 年 12 月第 一 版 开本: 720×1 000 1/16

2016 年 12 月第一次印刷 印张: 21 1/4

字数: 418 000

定价: 98.00 元

(如有印装质量问题, 我社负责调换)

前 言

物联网系统最关键的特点与要求是“联”。虽然有很多物联网系统部署在实际环境中，但它们基本都是专门领域的系统，未重点关注“联”的问题。在人类不长的计算机技术发展历史上，跨行业领域的系统互联一体化，还未有成熟的经验可以借鉴，也缺乏完备的理论。本书对物联网系统的“联”中两个关键问题：“集成”与“扩展性”，进行了探讨，阐述了我们的技术尝试与思考，希望能够抛砖引玉。

在本书中，首先引入资源模型，将物理世界表现在信息世界中，使物联网资源模型成为连接物理世界和信息世界的桥梁，而事件就是行驶在这些桥梁上的“交通流”。事件将现实物理世界的真实状况带到信息世界，并将信息世界的控制指令返回到物理世界中，完成对物理世界的监视与控制。为了方便信息世界的事件处理和信息共享，对物联网资源模型进行了计算机可理解的标准化表示，从而构建物联网资源与事件的知识基础。

在此知识基础上，将异构物联网资源通过适配的方式自动接入信息系统，当物联网资源与事件进入信息世界后，需要将它们在合适的时间、合适的地点交付给合适的服务和人员，为此本书采用统一消息空间模型完成此任务。对基于统一消息空间的单个服务，本书采用事件驱动的方式将其组织在一起，并解决其中集成和扩展的理论问题。在既有事件和服务的基础上，本书阐述了基于知识推理和混合学习的方法，从孤立事件中提取复合知识，用其改进服务提供质量，包括自动服务生成和服务可适性调整。同时对物联网服务安全的独特性进行了阐述。

总之，本书试图回答物联网服务构建的技术挑战：异构感知设备和异构感知数据的集成和共享的问题；社会规模感知系统互联集成时的扩展性和实时性的问题；感知数据充分利用满足人们个性化需求的问题；社会关键基础设施开放互联后的安全性和可靠性保障的问题。

虽然市面上已有不少关于物联网技术的书籍，但是缺少专门针对物联网应用的相关理论与技术的归纳。本书首先讨论将物理世界与信息世界联系起来，并由计算机自动完成识别和连接的方法。其次，本书探讨物联网系统事件驱动的本质，物理世界发生的动态变化现象表现为新的监测事件，而信息世界对物理系统的调整则表现为控制事件的发生，为这两类不同事件提供其获取、传输、处理、分析的算法、系统与支撑平台是研究的根本目标。最后，探讨事件驱动的物联网服务生成与运行的理论和方法，同时，对如何保障这类系统安全的问题进行探索。因此，本书的特色，是紧扣物联网服务系统的基本特征，从物联网服务系统构建的关键技术问题出发，理论与实际相结合开展研究与开发。

基于事件的大规模物联网服务系统的构建理论和方法等关键技术的探索,有助于夯实物联网现代服务业的发展基础。由于该方向的研究尚处于起步阶段,尚无体系化的理论方法,本书的研究内容正是对这个全新研究领域进行理论探索和尝试,这对于提高现代服务业的价值生存空间将产生重要的影响,也可进一步推动物联网服务业向更深层次的发展。在社会工程实践方面,智慧城市、智能电网等物联网应用蓬勃发展,在这些应用中会出现大量的感知事件流,并且这些行业应用都是包含大量基于事件进行服务协作的大规模复杂服务系统,它们都是本书的实际领域对象与应用领域。

本书的适用范围既包括单个物联网服务系统的构建领域,也包括多个服务系统互联领域,但是本书的重点是试图解决“联”的问题,即如何让不同的物联网系统自动互联、互通、互操作,并在互联过程中解决扩展性和自治性的问题。本书在物联网的“联”方面进行了探索,并尝试给出我们的答案,希望能够抛砖引玉。本书可作为物联网技术领域的人员与学生的参考书,在每章节中,都对相应技术的相关工作进行了综述,可以作为该领域研究者的文献查找阅读的信息来源。

本书第1章对物联网服务系统的发展背景与技术趋势进行探讨和描述。第2章对事件驱动的物联网服务系统相关技术进行综述性的介绍。第3章着重介绍物联网资源建模、接入、复杂事件处理等方面的内容。第4章介绍统一消息空间,这也是本书的重点内容,即如何为物联网服务提供一个基本的运行与生成环境。第5章介绍如何构建大型的物联网服务系统,并提供系统生成柔性化保证,以适应不同的运行场景。第6章则讲述如何保障物联网服务系统的安全,描述其独特需求、安全设计方法、运行时安全保障的实现路径等内容。

本书是北京邮电大学网络技术研究院网络服务基础理论研究中心整个团队研究开发结果的汇总与理论总结,在此对项目组所有成员的辛勤工作表示感谢。相关研发工作得到了国家自然科学基金面上项目(项目编号:61372115,61132001)的资助。首先要感谢网络服务基础理论研究中心的首席科学家陈俊亮院士,陈院士对项目组的工作给予了极大的关怀与指导,在此表示真挚的谢意。另外,程渤、乔秀全、刘传仓、吴步丹、黄霁葳、赵帅都参与了深入讨论,并为本书提供了宝贵意见,在此一并致谢!

本书第3章的代码实现与测试由吴思齐与左文峰完成;本书第4章的代码实现与测试由王双锦、温鹏和肖丹完成;本书第5章的代码实现与测试由林垚和张丽欣完成;本书第6章的代码实现与测试由周海静、华强和王兴完成。在此一并致谢!

本书不足之处在所难免,欢迎各位专家、读者批评指正。

章 洋 毛艳芳

2016年8月

目 录

前言

第 1 章 绪论	1
1.1 物联网服务系统中事件驱动技术发展现状与趋势	1
1.2 事件驱动的物联网服务提供技术现状与发展趋势	4
1.3 本书的研究内容组成	6
参考文献	7
第 2 章 事件驱动服务相关工作基础	11
2.1 概述	11
2.1.1 电信业务支撑平台	12
2.1.2 智能网业务生成平台	12
2.1.3 下一代网络业务交付平台 SDP	14
2.1.4 下一代服务层叠网协同平台	16
2.2 服务平台实现模型及其体系结构基础	18
2.2.1 SOA 的体系结构基础	19
2.2.2 SOA 服务支撑平台模型	24
2.2.3 服务总线模型	27
2.3 服务支撑平台实现技术	33
2.3.1 服务支撑平台的功能性视图	36
2.3.2 服务支撑平台的性能与服务质量	40
2.3.3 服务安全	45
2.4 物联网相关技术	47
2.4.1 物联网相关项目	47
2.4.2 事件驱动物联网服务	50
2.4.3 一阶逻辑	52
2.4.4 Event-B	54
参考文献	55
第 3 章 物联网资源	58
3.1 引言	58

3.2	物联网资源建模	58
3.3	物联网资源知识表示及其扩展	64
3.4	基于物联网资源的复杂事件处理	68
3.4.1	复杂事件处理相关工作	69
3.4.2	基于物联网资源的复杂事件处理问题概述	70
3.4.3	举例及复杂事件处理问题展示	71
3.4.4	为到达事件选择相关资源实例	74
3.4.5	物联网资源中复杂函数的表示与估计	75
3.4.6	复合理论	79
3.4.7	复杂事件处理服务	80
3.4.8	实验	82
3.5	物联网资源管理平台	82
	参考文献	87
第4章	统一消息空间——物联网服务通信基础设施	91
4.1	引言	91
4.2	事件路由	97
4.2.1	需求概述	97
4.2.2	系统设计	100
4.3	事件转发	109
4.3.1	需求分析	110
4.3.2	整体方案设计	113
4.3.3	系统实现	116
4.4	服务接口	129
4.4.1	服务接口需求分析	129
4.4.2	服务接口设计	132
4.4.3	服务接口实现	142
4.5	管理控制台	155
4.6	服务生成	159
4.6.1	需求分析	159
4.6.2	系统设计	163
4.6.3	服务部署运行	167
4.7	平台中分布式实时数据服务	170
4.7.1	分布式实时数据服务设计	171
4.7.2	分布式实时数据服务实现	176

4.8 相关工作综述	184
参考文献	187
第5章 可扩展物联网服务柔性化生成	192
5.1 引言	192
5.2 物联网服务定义	194
5.2.1 基于事件的物联网服务	194
5.2.2 事件会话	196
5.2.3 事件驱动的物联网服务定义	197
5.3 生成物联网服务	198
5.3.1 物联网服务需求表示	198
5.3.2 物联网服务计算需求的表示	203
5.3.3 物联网服务细化部署	204
5.4 物联网服务属性计算	205
5.4.1 物联网属性计算的理论基础	205
5.4.2 环境建模	207
5.4.3 计算服务属性	209
5.5 物联网服务的可扩展运行	210
5.5.1 物联网服务实例的高并发	210
5.5.2 物联网业务流程的分布式执行	214
5.5.3 分布式业务流程的属性保障	216
5.6 物联网界面服务案例	221
5.6.1 需求分析	221
5.6.2 总体设计	225
5.6.3 人机界面服务生成	228
5.6.4 人机界面服务的部署运行	235
参考文献	241
第6章 物联网服务安全	243
6.1 服务安全基础	243
6.1.1 服务安全基础设施	243
6.1.2 消息安全	246
6.1.3 信任框架与访问控制	249
6.2 物联网服务安全设计	256
6.2.1 基于属性的访问控制策略	256
6.2.2 事件驱动的智能电网服务案例	257

6.2.3	分布式安全框架	260
6.2.4	基础的同态加密方案	261
6.2.5	分布式安全方案	263
6.2.6	身份服务与身份认证	266
6.3	物联网服务运行时安全	274
6.3.1	内存取证技术基础	276
6.3.2	需求分析	280
6.3.3	整体方案设计	282
6.3.4	系统实现	287
6.4	物联网服务 I/O 安全控制方案	294
6.4.1	相关技术介绍	294
6.4.2	需求分析	298
6.4.3	系统设计与实现	302
6.5	物联网服务安全方案中安全管理服务	306
6.5.1	需求分析	309
6.5.2	系统设计实现	312
6.5.3	系统部署应用	320
6.6	物联网服务(工控系统)主动综合安全方案	323
	参考文献	330

第1章 绪 论

1.1 物联网服务系统中事件驱动技术发展现状与趋势

随着未来网络研究的兴起,以及物联网和云计算应用的发展,事件以及“事件驱动”的方法学重新吸引了学界和业界的目光。例如,DONA(Data-Oriented Network Architecture)^[1]、基于内容中心网络(Content-Centric Networking,CCN)的项目(CCNx)^[2]、未来网络体系结构与设计(4WARD)^[3]、命名数据网络(Named Data Networking,NDN)^[4]和发布-订阅网络技术(Publish-Subscribe Internet Technology,PURSUIT)^[5]等研究项目都采用了事件驱动的思想,其中PURSUIT直接采用事件驱动的方法构建新一代网络体系结构。SRT-15^[6]是FP7项目中使用事件驱动方法来解决服务之间灵活和及时交互的问题的一个代表。作为物联网行业应用的例子,美国的GridStat项目(Washington State University)和NASPInet(North American Synchro Phasor Initiative network)项目^[7]采用事件驱动的方法学为智能电网服务构造基础的通信平台。他们认为,伴随电力系统的发展,传统SCADA(Supervisory Control and Data Acquisition)系统较慢的(秒级)数据更新率、不同参与者之间数据不同步已不能适应现代电网的需求,尤其是新能源的加入造成了更多不确定性和不可预测性,以及长距离传输造成的广域影响,要求对电网有更充分的状态感知能力、更充分的数据实时共享、更内谱的智能电网服务协同。GridStat和NASPInet项目总结了智能电网服务通信平台的基本要求。

(1)采用事件驱动方法学,利用其解耦性提高电网系统的扩展性和协同性,并提升智能电网控制与保护服务的实时响应能力。

(2)一对多的服务交互是智能电网系统本质性需求和常态。

(3)每个事件本身是独立有意义的实体,可自解释。

(4)端到端保证存在一个宽的选择范围。

(5)差异性事件接收端、事件接收速率和对事件详略程度的需求都存在差异。

(6)超低时延(5ms),高吞吐量。

“事件驱动”其含义是将事件看成网络服务中的最基本实体,是服务间交互的基本机制。在该机制下,服务消费者订阅满足其意图的事件,服务提供者并不关心哪些消费者在什么时间或什么地点、订阅了哪些事件。服务消费者和提供者之间互不关注对方的时间、位置等状态信息,只关心如何向网络表达真实的兴趣与意图。因此基

于事件机制的面向服务体系结构(Event-Driven Service Oriented Architecture, EDSOA)比传统的面向服务的体系结构(Service-Oriented Architecture, SOA)具有更进一步的松耦合特点,称为EDSOA的解耦性。EDSOA解耦性具有五个层面的含义。

(1)在单个事件层面,在具有足够数据区分能力的名称结构前提下,事件类型和主题通过层次性名称来表达,使事件种类简化和规范化,让其易于理解、分析和互操作。

(2)在服务接口层面,服务接口主要简化为事件发布操作和接收事件通知操作,接口调用和数据返回在时间和空间上解耦,即消费者在不知道服务提供者状态(如地址、是否在线)的情况下发布请求事件,然后进行其他计算和操作(如离线),之后可能于不同地点和时间接收返回事件或其他事件。

(3)在服务行为层面,不同动作间的偏序约束关系通过事件间真正的因果依赖而最小化,服务行为体现为对到达的一对多的离散事件的反应。

(4)在服务系统层面,系统局部结构与行为的改变与演进不影响系统的既有功能和稳定性,或通过全局事件间因果关系保持使动态性影响最小化。

(5)在服务属性层面,服务属性的保证机制可附加在独立有意义的事件实体上,减少对端系统假设的依赖,实现单纯端系统不能保证的系统属性。

虽然事件驱动的DEBS(Distributed Event-Based System)^[8, 9]已经有了十多年的发展历史,但是事件驱动的服务方面研究成果并不丰富。作为工业界的努力,Web服务通知(Web Services Notification, WSN)^[10]和WS-Eventing^[11]规范定义了Web服务如何使用事件作为交互手段。作为学界的努力,文献[12]给出了如何将由中心编制点生成的服务流程转化为依赖于分布式事件的相互协作的一组服务;文献[13]和文献[14]提出了基于分布式事件进行协同的PI演算模型。这些零星的研究不能回答如何基于事件机制构建面向服务的体系结构,以及如何生成大规模的事件驱动的物联网服务系统。对这两个问题,从理论到实践都缺少系统而深入的研究成果。

从目前的计算机科学与技术的发展程度来看,为大型复杂的计算机系统建立精确模型的理论和方法依然不成熟^[15, 16]。既缺乏相应的理论,又缺乏相应的大型计算机系统建模的实践经验。虽然大型系统的建模需要相应的证明未来系统能够满足需求与约束的手段,但是人们依然未准备好使用形式化手段来建模和设计复杂计算机系统,也就是说如何采用形式化方法简单而有效地设计系统仍然处在探索之中。不少学者在这个方面进行了积极而有意义的探索,如B方法,本书的研究也将沿用该方法。因为要研究的基本对象是事件、服务和服务体系结构,其中事件又是最基本的元素,因此本书借鉴Event-B方法^[15, 16]作为研究的起点和出发点。

Event-B^[15, 16]以事件为基础给出了将复杂系统进行分解和逐步细化的形式化方法,它的数学基础是集合论和一阶谓词逻辑。它认为一个系统可以由多个模块构成,每个模块可以定义自己的上下文环境和事件机,每个事件机由变量、事件和不变属性构成。对于事件机,可以将其逐步精化,但是在精化的过程中必须保证不变属性

的稳定性。可以使用推理机对 Event-B 方法构建的模型系统进行分析和属性推理。然而, Event-B 方法对事件机的复合缺乏足够的支持, 而对服务计算来说, 服务组合是其一个最基本的议题, 而且是必须回答的问题。另外, 对于服务自身的独特性问题, Event-B 也未进行深入的研究。同时, 关于如何使用 Event-B 建模 SOA, 尤其是 EDSOA 的研究也相对较少。

支持 Event-B 的组合并不简单, 它涉及分布式系统经典的并发问题, 如何将经典的并发理论与 Event-B 方法结合起来, 需要进一步研究探索。另外两个问题同样没有现成的答案。对这些问题进行研究的总指导思想是: 分离交互和计算、统一结构与移动性。Milner 认为经典的程序存储式计算机解决的是“数值计算”的问题; 而现在的网络环境中, 不论是物与物、人与物、人与人, 还是人造的计算实体之间, 体现更多的是通信和交互, 交互是互联网上“计算”的本质。对交互而言, 第一个概念是离散空间, 它包含计算主体、位置和相互之间的连接; 当空间被重构时, 就涉及第二概念, 即移动性。EDSOA 的理念与总的指导思想恰好相符, 事件的解耦性体现的是交互的分离, 服务对事件的反应则是计算的体现; EDSOA 的体系结构体现的是空间关系, 其局部的动态性体现为移动性, 其行为则表现为一系列具体的交互。因此, 本书探索在事件驱动的情况下, 大规模复杂服务系统构建的理论和方法。

对于服务系统来说, 由于不同的服务是由不同团队在不同时间、地点各自独立开发的, 将它们组合在一起必然存在接口、行为和属性不匹配的问题, 以及服务编排规范的可实现性的问题。传统上, 对于服务可实现性问题的解决思路是增加附加的消息; 对于不匹配问题, 则通过生成适配子来解决。但是, 对于 EDSOA 则存在由解耦性导致的事件独立性问题, 即事件是环境中自解释、独立有意义的实体, 既不依赖于服务提供者也不依赖于服务消费者而解释, 或者说事件是匿名的, 其适配性必然有其不同的特点。

对于服务适配问题^[17], 其适应性解决方案可以分为三类: 受限的适应性、通用适应性和 Ad-hoc 型适应性。受限的适应性方法^[18, 19]通过限制和删除不匹配的行为来解决问题, 从而限制了复合系统的功能。通用适应性方法^[20-22]根据抽象规范(适配契约)使用中介程序进行变量重命名、记录消息、改变消息序来解决服务间的适配问题, 这种方法往往是半自动和复杂的。Ad-hoc 型适应性方法^[23, 24]采用适配模式和适配操作代数解决服务间特定的不匹配问题。服务适配问题的研究, 在问题表达、适配支持和自动化方面已经相对成熟。例如, 文献[25]基于进程代数为服务自适应提供了一套理论框架。文献[26]和文献[27]则使用综合性方法对服务接口和服务行为不匹配问题进行了研究。然而这些方法在服务属性适配方面未能提供理想的答案, 不能给出面向服务的计算中, 服务自适应问题的完整而系统的解决方案。基于 EDSOA 的服务解耦性特征有可能给出服务适配问题不同的解决视角和更深入全面的答案。

因此对服务适配问题, 本书主要的研究目标就是利用 EDSOA 的解耦性从三个

方面：接口匹配、行为匹配和属性匹配，重新进行思考，重点是后两者适配方法的探索和研究。

对于服务特性的验证问题，目前主要基于 Petri 网、进程代数和状态机进行，缺少非常切合服务特点的验证理论。当在服务系统上继续施加安全等属性限制时，问题将变得更加复杂。如何将安全等属性和服务行为一体化，获得合适的服务验证方法也是一个较困难和重要的问题。

文献[28]提出使用模型检测的方法对安全策略约束的服务流程进行自动化分析。它将具有基于角色的访问控制(Role-Based Access Control, RBAC)策略和代理约束的服务流程建模为迁移系统，使用线性时序逻辑表达职责分离属性。它使用 NuSMV 模型检测器^[29]对部分实例进行了验证。文献[30]则使用 SAL 模型检测器^[31]对综合了资源分配和安全约束的服务流程进行验证。文献[32]将工作流模型和 RBAC 模型结合，使用扩展的状态机进行安全属性描述，并基于 Spin^[33]进行验证。其他的工作包括文献[34]~文献[39]，其中比较有代表性的是文献[39]。它提出了基于动作的语言，使服务流程建模和安全建模分离，并可以自然地结合，将复杂的建模问题简化，在验证时采用 cCalc^[40]。这些方法在服务验证上都有所欠缺，往往采用通用的语言，只抓住服务的部分特征，另外也未针对 EDSOA 服务。

基于事件的大规模复杂服务系统的构建理论和方法等关键技术的研究，有助于夯实物联网服务业的发展基础。由于该方向的研究尚处于起步阶段，尚无体系化的理论方法，本书的研究正是对这个全新研究领域进行的理论探索和尝试，这对于提高事件驱动物联网服务系统的价值生存空间将产生重要的影响，并进一步推动事件驱动物联网服务系统向更深层次发展。在社会工程实践方面，智慧城市、智能电网等物联网应用蓬勃发展，在这些应用中会出现大量的感知事件流，并且这些行业应用都是包含大量基于事件进行服务协作的大规模复杂服务系统，它们迫切需要可行与前瞻的理论指导，本书的研究成果不但可以为它们提供理论指导，而且可以直接应用在其中。

1.2 事件驱动的物联网服务提供技术现状与发展趋势

物联网的概念关键体现在“物”和“联”两个字上。“物”，就是将物理世界的任意实体，哪怕是一片尘埃，也赋予其标志，予以识别、感知和控制。“联”，则蕴涵着多层意思。目前工业界和学术界都开发出了很多物联网的项目，这些项目基本都做到了“物”这一点，然而很多项目却忽略了“联”。“联”不仅仅是物体和网络的连接，更重要的是体现一个交互和共享的概念。包括跨网络、跨应用程序、跨领域的服务和能力的交互以及数据和信息的共享及融合，现有的传感网(sensor web)项目往往都是“竖井式”与具体应用相关的“intranet of things”，并没有做到“internet of things”。要实现“联”需要提供两个维度的交互能力：垂直交互和水平交互。垂

直交互是指不同的应用程序可以共用一个或多个感知延伸网络,摆脱每开发一个应用程序部署一个专有的感知延伸网络的“竖井式”现状。水平交互是指:首先,感知延伸网络之间可以相互交互;其次,应用系统之间的服务和能力能够相互交互、共享数据并对服务进行组合、对信息进行融合进而提供更高层次的服务能力和信息。

要实现上述万物互联的目标,需要解决透彻感知、数据与服务汇聚传输、情景语意解释、异构融合可扩展平台等技术需求。透彻感知主要包括传感器、激励器、被动(主动)数字标志、嵌入式系统等物理设备,它们能够贴附或嵌入物理对象中,使得这些物理对象更加“智能”,从而加入物联网中成为其中的一个“物”。在这个层面主要是硬件制作工艺和技术方面的挑战,如电池的电量和体积、传感器的体积、传感器包装材质等。数据与服务汇聚传输,是将传感器和嵌入式系统无缝地整合到 Internet 中,使得固网、移动有线或无线网能够无障碍连接,为不同层次(物理对象、应用程序、服务)提供高效、实时、可靠的数据与服务汇聚传输。情景语意解释,是指将现实世界的数形成语义情景信息,并提供给各种应用程序。异构融合可扩展平台是指,支持综合地监控和管理所有涉及的物理实体、软件组件,保证可伸缩性、高可用性和安全性,以及一些附加的增值服务,如以可接受的代价按需查找资源和数据、分布式大规模数据的高效存储服务。

开放地理空间信息联盟(Open Geospatial Consortium, OGC)组织的 SWE(Sensor Web Enablement)^[41]提出了一个标准的基于服务的(service-based)体系结构来描述传感器资源,包括它们的能力和测量类型,以及其他与传感器运行有关的环境的描述。这些标准用于分类各种传感器资源,理解它们观测到的数据,以及提供基于可扩展标记语言(eXtensible Markup Language, XML)和标准化标签的有限的交互和数据交换能力。SWE 为开发 Web 互联的(Web-connected)传感器和各种传感器系统构建了一个开放标准框架。这个框架称为传感器 Web,具体是指 Web 可以访问的传感器网络和传感器数据可以通过标准的协议和应用程序接口被搜索和访问。SWE 由三组语言和四组服务规范组成,语言包括 SML(Sensor Model language)^[42]、OM(Observation and Measurements)^[43]和 TML(Transducer Markup Language)^[44];服务规范包括 SOS(Sensor Observation Service)^[45]、SPS(Sensor Planning Service)^[46]、SAS(Sensor Alert Service)^[47]和 WNS(Web Notification Service)。

目前,存在很多物联网相关技术的研究项目,如 SENSEI、ASPIRE、IoT-A、PECES、CONET、SPITFIRE、SensorGrid4Env 等^[48]。其中 ASPIRE 体系结构基于 EPG Global,并进行了一些附加的扩展;在基于射频识别(Radio Frequency Identification, RFID)的应用场景中,通过 RFID 读取器以被动通信的方式来读取资源的标志和内容;通过资源目录实现资源的信息服务,该目录存储一个具体资源的 What、Where、When、Why 等几个方面的信息;通过语义查询解析器为应用层提供查询服务。另外,ASPIRE 提出了业务事件生成器来为 RFID 应用程序的语义交互提

供实现逻辑；还提供了三种标准的交互模式：订阅、轮询和即时通信。LLAAL 项目提出了 SOA 和事件驱动架构的整合方案，该方案基于 OSGi (Open Service Gateway Initiative) 进行实现，包括两个主要的子系统：服务平台 openAAL 和复杂事件处理系统 ETALIS。它提出了通用的平台服务，如情景数据管理，用于收集和抽象环境数据；基于工作流的系统行为；基于语义的服务发现等。框架和平台服务基于共享的词汇集以松耦合的方式交互和通信。

在将物理实体引入数字空间后，需要解决基于感知数据的物联网服务提供问题。目前，物联网服务提供技术引起了越来越多的重视和关注，已有大量工作研究了通过移动代理或者中间件技术来实现物理世界和企业服务的集成。然而，在集成过程中面临的一个主要的障碍是大量已部署的传感器网络或者应用系统采用了专有业务平台和专有的技术，这些非标准的接口协议导致应用系统与传感网络的集成非常复杂。为了解决跨业务域的系统集成，最近研究人员将 SOA 应用到物联网服务提供领域。考虑到传统的 SOA 标准和技术的设计初衷主要是解决互联网环境下重量级的企业级服务的集成，研究人员尝试提出了一些适合于资源受限的嵌入式设备的轻量级 Web 服务协议。

总体而言，现有的研究主要集中在使用 SOA 技术或者更轻量级服务来实现不同物理实体和企业应用系统的互联互通问题上。事实上，互联互通是物联网服务面临的一个重要问题，但随之而来的另一个问题就是在解决了设备的互联互通之后，如何在一个分布式、松耦合的环境中实现感知信息在信息提供者和消费者之间的有效分发，并基于物理世界发生的事件而动态地协调相关的企业服务作出快速响应。目前，大多数物联网应用系统是一种紧耦合、封闭式的服务提供模式，也就是传感器网络采集相关的感知信息并存储在一个集中的数据库中，相关特定的应用系统通过访问数据库中的信息从而提供相应服务。这种服务提供模式极大地限制了系统的灵活性和扩展性，不适合大规模分布式、复杂物联网服务系统的构建。为了支持不同的服务系统间的信息交换，传统的以“请求-应答”为主的 SOA 经常用集中式的服务编排机制来通过服务的调用交换数据。然而，这种模式导致了信息的提供方和消费者彼此直接交互。同时，由于业务流程常常通过一些流程编制语言如 BPEL (Business Process Execution Language) 或者工作流语言来进行描述，这样当应用程序需求发生变化时，如新集成一个服务或者一个服务的撤销等，这种紧耦合的系统架构缺乏足够的灵活性来进行适配，导致必须重新修改流程。事件驱动、面向服务的物联网服务提供方法通过集成事件驱动架构和面向服务架构的优势，实现了感知信息的按需分发和事件驱动的服务动态协同。这种“时间-空间-控制”解耦的方法能够很好地适应动态变化的物联网应用环境。

1.3 本书的研究内容组成

本书阐述的内容如图 1-1 所示，我们试图解决事件驱动技术与 SOA 技术集成的

问题,从而实现事件驱动的物联网服务提供。图 1-1 主要包含四大部分五个层面的内容,底层是物联网服务资源描述与接入处理部分,负责将物理实体接入数字空间,将数字化的物理对象标准化为物联网资源,为上层应用与服务提供情景化的物联网感知信息的语意解释。物联网服务生成与运行环境的统一消息空间部分,负责将感知数据、语意解释、服务提供汇聚在一起,进行寻址、路由与传输,并为物联网服务提供基本的运行环境与生成编程环境。可扩展物联网服务柔性生成的事件驱动服务运行(含人机界面案例)部分,试图解决利用跨域的物联网资源、物联网功能组件构建大型的物联网服务系统,并支持其多环境部署、动态适配调节等功能。物联网服务安全部分,试图在设计时与运行时为物联网服务系统提供安全保障,从而提供社会公共基础设施开放互联时运行安全的可能性。



图 1-1 本书内容组织结构

第 2 章对事件驱动的物联网服务系统相关技术进行综述性的介绍。第 3 章着重介绍物联网资源建模、接入、复杂事件处理等方面的内容。第 4 章介绍统一消息空间,这也是本书的重点内容,即如何为物联网服务提供一个基本的运行与生成环境。第 5 章介绍如何构建大型的物联网服务系统,并提供系统生成柔性化保证,以适应不同的运行场景。第 6 章则讲述如何保障物联网服务系统的安全,描述其独特需求、安全设计方法、运行时安全保障的实现路径等内容。

参 考 文 献

- [1] Koponen T, Chawla M, Chun B G, et al. A data-oriented (and beyond) network architecture. SIGCOMM'07: Proceedings of the 2007 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications, ACM, New York, NY, USA, 2007: 181-192.
- [2] Palo Alto Research (PARC). CCNx: Web Site (2010). <http://www.ccnx.org>[2016-6-1].
- [3] Ericsson A B. 4WARD: Web Site (2010). <http://www.4ward-project.eu>[2016-6-1].

- [4] NDN Project Team. Named Data Networking. <http://www.named-data.net>[2016-6-1].
- [5] The PSIRP Project'S Member Institutions. PSIRP Project. <http://www.fp7-pursuit.eu/PursuitWeb/PURSUIT>[2016-6-1].
- [6] SAP A G, Dresden T U, Epsilon S R L, et al. The SRT-15 Research Project. <http://srt-15.unine.ch/home>[2016-6-1].
- [7] Bakken D E, Bose A, Hauser C H, et al. Smart Generation and Transmission with Coherent, Real-time Data. <http://www.gridstat.net/trac/#>.
- [8] Eugster P T, Felber P A, Guerraoui R, et al. The many faces of publish/subscribe. ACM Computing Surveys (CSUR), 2003, 35 (2): 114-131.
- [9] Muhl G, Fiege L, Pietzuch P. Distributed Event-Based Systems. Berlin: Springer-Verlag, 2006.
- [10] OASIS. OASIS Web Services Notification (WSN) TC. https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsn[2016-06-01].
- [11] Box D, Felipe L, Curbera F, et al. Web Service Eventing (WS-Eventing). <http://www.w3.org/Submission/WS-Eventing>.
- [12] Li G L, Muthusamy V, Jacobsen H A. A distributed service-oriented architecture for business process execution. ACM Transactions on the Web, 2010, 4(1): 2.
- [13] Ferrari G L, Guanciale R, Strollo D. Jscl: A middleware for service coordination. Lecture Notes in Computer Science, 2006, 4229: 46-60.
- [14] Ciancia V, Ferrari G, Guanciale R, et al. Event based choreography. Science of Computer Programming, 2010(75): 848-878.
- [15] Abrial J R, Hallerstede S. Refinement, decomposition, and instantiation of discrete models: Application to Event-B. Fundamenta Informaticae, 2007: 1-28.
- [16] Iliasov A, Troubitsyna E, Laibinis L, et al. Supporting reuse in Event B development: Modularisation approach. Abstract State Machines, Alloy, B and Z, Lecture Notes in Computer Science, 2010, 5977: 174-188.
- [17] Poizat P. Formal model-based approaches for the development of composite. Universidad de Málaga Campus de Excelencia Internacional Andalucía Tech, 2014.
- [18] Reussner R H. Automatic component protocol adaptation with the CoConut/J tool suite. Future Generation Computer Systems, 2003, 19(1): 627-639.
- [19] Inverardi P, Tivoli M. Deadlock free software architecture for COM/DCOM applications. Journal of Systems and Software, 2003: 173-183.
- [20] Bracciali A, Brogi A, Canal C. A formal approach to component adaptation. Journal of Systems and Software, 2005: 45-54.
- [21] Brogi A, Canal C, Pimentel E. Component adaptation through flexible subservicing. Science of Computer Programming, 2006: 39-56.