Broadview
www.broadview.com.cn

WILEY

Pentaho Kettle是ETL解决方案的不二之选

Pentaho Kettle Solutions:
Building Open Source ETL Solutions
with Pentaho Data Integration

Pentaho Kettle解决方案：

使用PDI构建开源ETL解决方案

Matt Casters
Roland Bouman
Jos van Dongen
著

初建军
曹雪梅
译

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

Pentaho Kettle Solutions:
Building Open Source ETL Solutions
with Pentaho Data Integration

Pentaho Kettle 解决方案：

使用PDI构建开源ETL解决方案

Matt Casters
Roland Bouman 著
Jos van Dongen

初建军
曹雪梅 译

電子工業出版社
Publishing House of Electronics Industry

内容简介

本书主要介绍如何使用开源ETL工具来完成数据整合工作。

本书介绍的PDI(Kettle)是一种开源的 ETL 解决方案，书中介绍了如何使用PDI来实现数据的剖析、清洗、校验、抽取、转换、加载等各类常见的ETL类工作。

除了ODS/DW类比较大型的应用外，Kettle 实际还可以为中小企业提供灵活的数据抽取和数据处理的功能。Kettle除了支持各种关系型数据库、HBase、MongoDB这样的NoSQL数据源外，它还支持Excel、Access这类小型的数据源。并且通过插件扩展，Kettle 可以支持各类数据源。本书详细介绍了Kettle可以处理的数据源，而且详细介绍了如何使用Kettle抽取增量数据。

Kettle 的数据处理功能也很强大，除了选择、过滤、分组、连接、排序这些常用的功能外，Kettle 里的Java表达式、正则表达式、Java脚本、Java类等功能都非常灵活而强大，都非常适合于各种数据处理功能。本书也使用了一些篇幅介绍Kettle这些灵活的数据处理功能。

本书后面章节介绍了如何在 Kettle 上开发插件，如何使用Kettle处理实时数据流，以及如何在Amazon AWS上运行Kettle 等一些高级主题。

除了介绍PDI(Kettle)工具的使用和功能，本书还结合Kimball博士的数据仓库和ETL子系统的理论，从实践的角度介绍数据仓库的模型设计、数据仓库的构建方法，以及如何使用 PDI实现Kimball博士提出的34种ETL子系统。

图书在版编目（CIP）数据

Pentaho Kettle解决方案：使用PDI构建开源ETL解决方案 /（美）卡斯特 (Casters,M.)，（美）布曼(Bouman,R.)，（美）东恩(Dongen,J.V.) 著；初建军，曹雪梅译. —北京：电子工业出版社，2014.3
书名原文：Pentaho Kettle Solutions:Building Open Source ETL Solutions with Pentaho Data Integration
ISBN 978-7-121-22445-4

Ⅰ. ①P… Ⅱ. ①卡… ②布… ③东… ④初… ⑤曹… Ⅲ. ①数据库—技术 Ⅳ. ①TP311.13

中国版本图书馆CIP数据核字（2014）第021514号

策划编辑：张月萍
责任编辑：贾　莉
印　　刷：北京中新伟业印刷有限公司
装　　订：三河市皇庄路通装订厂
出版发行：电子工业出版社
　　北京市海淀区万寿路173信箱　　邮编：100036
开　　本：787×1092　1/16　　印张：30.25　　字数：832千字
印　　次：2014年3月第1次印刷
定　　价：89.00元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。
质量投诉请发邮件至zlts@phei.com.cn，盗版侵权举报请发邮件至dbqq@phei.com.cn。
服务热线：（010）88258888。

用户体会

报表管理是销售管理的一项重要工作，面对70多个国家合作伙伴的不同种类型的销售报表，如何通过自动化手段进行格式统一、计算和分发，是我2009年在新兴市场集团工作时的“痛点”——是我特别想实现的，而这套方案必须要基于开源的方案来控制开发及未来维护的成本。

Jason给我推荐开源的ETL工具Kettle来完成这个工作，他使用 Kettle 加上开源的报表工具 Jasper，用了两个月的时间就把这套系统实现并上线了：

它可以自动从各个数据源获取数据、自动生成 Excel 报表，并自动投送到相关业务人员的邮箱里，这节省了我三个做报表的人力！而且数据更及时、准确！非常了不起！！

赵海生
客户数据与市场秩序 总监
联想集团

数据是投资的重要基础，但由于数据量大且指标较多，从各种不同格式的报告中摘取我们希望的数据一直是让我们头疼的事情。这一事情的改观发生在2011年Jason为我们带来Kettle工具之后，经过几个月的开发和测试，我们的指标自动抓取系统正式上线并一直沿用至今，它能从各种格式的报告中摘取重要的数据，这些数据形成我们分析的基础。实际上，这只是使用了Kettle工具的一小部分功能而已，相信在数据抓取和处理领域，我们还将有更多的合作机会。

从2011年就听到Jason要翻译本书的计划，很高兴能看到这一目标最终实现，这是Jason本人的一个里程碑，也是让更多的人受益于Kettle工具的一次契机，祝Jason和Kettle的路都越走越宽。

郑伟征
业务总监
北京中能兴业投资咨询有限公司

从企业架构的角度来看，和传统的编写代码相比，ETL工具在开发实施效率（包括代码复用）、可靠性、低出错率、可维护性上绝对都是巨大的进步。我个人相信在企业ETL领域，编程语言的工作未来可能会减少到总任务量的10%，剩下90%均需要借助ETL工具来实现。

本书和市面上林林总总的介绍ETL工具的书籍不同。书籍的原作者Matt是Kettle的核心设计与开发者（Kettle的灵魂所在），而且Jason带领的团队对 Kettle的源代码有深入的了解，并有丰富的实践经验，他们对本书所涉及的主题有切身的体会，这样可以最大限度地避免出现很多计算机译本图书出现的读者“不知所云”的情况。本书亦可以看作是Jason在国内不遗余力推广Kettle ETL解决方案的又一个里程碑。

徐洋

葛兰素史克（中国）投资有限公司 Enterprise Architect

数据对于每个企业来说都是极其重要的，它蕴含的价值不可限量，尤其是对于我服务的电信行业。我们做ETL工作先是自己开发程序或者存储过程，后来慢慢转型使用ETL工具，深刻体会到工具对于生产力发展的重要性。

随着对ETL的认识越来越深入，要求也越来越高，尤其是一些企业特定的需求即使是顶级的商业ETL软件也无法满足。在2012年一次偶然的机会我认识了Jason从而认识了Kettle，在Jason的帮助下仅用了三天时间就开发出一个定制化的组件，解决了长期困扰我们的问题，Kettle本身的灵活性、扩展性再加上Jason的团队对这款软件的驾驭能力，都是我们公司所需要的。

在得知Jason要翻译此书时，真的非常期待并衷心祝愿Jason和Kettle在中国能有更好的发展。

黄磊

上海理想信息产业（集团）有限公司 BI架构师

译者序

对于Kettle这个软件，要从2006年年初说起，当时我所在的公司内有很多数据整合类项目，当时商业的ETL工具太昂贵，不适合在预算有限的项目内使用，而手工写代码或写存储过程又会耗费大量的开发时间，还会增加项目的维护成本，于是我计划开发一款 ETL 工具，来满足公司内部数据整合项目的需求。作为产品开发的第一步，就是要调研当时市场上同类的开源产品，我调查了包括 Kettle、CloverETL、OctopusETL、KETL 等在内的多款开源 ETL 工具。

当时Kettle 2.2 版本刚发布，在阅读完Kettle 2.2的代码后，我觉得这就是我想要的 ETL 工具，它的插件丰富功能强大，并行执行效率高，最重要的是它有灵活可扩展的插件架构，可以通过二次开发来提供更多的功能，并可以作为引擎方便地集成在第三方应用中。

于是我放弃了自己开发一个 ETL 工具的想法，转而成为一个 Kettle 开发人员。当时的Kettle刚刚问世，功能还有很多不完善的地方，于是我经常在MSN上和Kettle的作者Matt Casters先生讨论Kettle的一些技术细节。Matt 是一位非常热心的程序员，对于我的问题或建议都能给予耐心的解答。这里要感谢Matt，正是因为他的无私奉献，才使Kettle成为世界上最流行的开源ETL工具。

2010 年*Pentaho Kettle Solutions*一书出版，我通过朋友李小凡获得了该书，李小凡是www.itisbi.com网站的站长，他也为Pentaho在中国的推广做出了重要贡献。

看过该书后，我觉得这是一本非常好的介绍 Kettle的书籍（尽管之前还有其他两本介绍Kettle的书籍，但那两本书基本是Kettle的入门书籍），*Pentaho Kettle Solutions*一书内容全面，对Kettle的技术细节也有一定程度的阐述，具有一定的理论深度。于是我计划翻译该书，以便让更多的做数据整合工作的开发人员都了解这个开源的ETL工具。

尽管从2010年年底我就开始动手翻译本书，但其间因为各种项目的耽搁，本书的翻译工作断断续续持续了三年。在翻译本书的过程中我还得到了www.pentahochina.com社区里一些朋友的支持，包括阿虎、银杏树、Super-超，他们协助我翻译了本书的一部分内容，在此也表示感谢。在 2013年年中，本书即将翻译完成的时候，遇到中山大学的黄志洪教授，黄教授推荐了电子工业出版社出版该书。这里感谢黄志洪老师、张月萍老师的帮助。

在本书的翻译过程中，尤其要感谢父母和妻子对我的支持，我以翻译本书为由逃避了大部分的家务劳动。我的妻子曹雪梅除了承担大部分的家务劳动外，还承担了本书的一部分翻译工作。

在本书的成书过程中，我还得到了很多朋友和同事的帮助，包括曾浩、刘小朋、郭振未、方进、任潇楠、郑发林、贺警阳、李超、冯海军、孙斌、饶丽、刘赫扬等同事，他们都为本书的出版做出了贡献，在此一并表示感谢。

现在我已经从一个 Kettle 开发人员变成了 Pentaho/Kettle 的粉丝，并且成立了一家商业智能/数据整合类的咨询公司（北京傲飞商智软件有限公司），从事 Pentaho/Kettle 的咨询、培训、开发、实施等工作（公司是 Pentaho 的官方合作伙伴，是 Pentaho公司在中国地区唯一授权培训合作伙伴）。

由于本书是基于Kettle 4.0版本的，而Hadoop等大数据技术在Kettle 4.3版本中才支持，所以本书没有介绍Kettle的大数据功能。关心Kettle如何支持大数据的读者可以到Pentaho的wiki去了解。

虽然译者已经尽力去翻译本书，但出于译者能力所限，如在书中出现错误和疏漏，还请读者不吝指正。

关于作者

Matt Casters是一位具有多年工作经验的独立商业智能顾问。他为许多大公司建立了无数个数据仓库和BI解决方案。在过去的8年里，Matt Casters把自己的时间都用于研发一个ETL工具——Kettle。2005年12月，Kettle成为开源软件。2006年初期，Kettle走进Pentaho。随后，Matt就职于Pentaho，成为数据集成总监。在Pentaho，他继续从事Kettle的研发工作。Matt致力于帮助建设Kettle社区，回答论坛上的提问，有时在世界会议上发表演讲。博客：http://www.ibridge.be。Twitter：@mattcasters。

Roland Bouman目前从事前台页面和商业智能的研发工作。他从1998年开始从事IT行业。多年来一直致力于开源软件的研发，尤其是数据库技术、商业智能以及页面开发框架。同时，Roland Bouman还是MySQL和Pentaho社区的成员。他经常参加MySQL使用者会议、OSCON、Pentaho社区等国际会议。Roland Bouman不仅是*MySQL 5.1 Cluster Certification Guide*和*Pentaho Solutions*两本书的合著者之一，也是MySQL和Pentaho相关书籍的技术评论家。技术博客：http://rpbouman.blogspot.com。Twitter：@rolandbouman。

Jos van Dongen是一位著名的商业智能专家、作家和演说家。他从1991年开始从事软件开发、商业智能以及数据仓库等领域的工作。Jos van Dongen曾先后就职于顶级的系统集成公司和管理咨询公司。1998，他创立了自己的咨询公司，Tholis Consulting。他为许多商业和福利组织构建了BI和数据仓库系统。Jos为丹麦Database Magazine撰写了新的BI研发成果，并且经常在国内和国际会议上发言。Jos van Dongen撰写了一本关于开源BI的书，并且和Roland Bouman合作编写了*Pentaho Solutions*。更多信息参考：http://www.tholis.com。Twitter：@josvandongen。

致谢

按照惯例，本书的封面列出了参与这本书创作的人员名单。但实际上，这本书是许多人共同劳动的成果。正是这许许多多人在幕后辛勤地工作着，才成就了本书的出版。作为作者，我们真心地感谢所有为本书做出贡献的人们。感谢大家帮助我们完成了*Pentaho Kettle Solutions*。

首先感谢为本书直接做出贡献的作者们。

- Ingo Klose提出了在单个转换中生成偏移键的解决方案。（参考第8章“处理维度表”的“基于计算器生成代理键”一节，图8-2。）
- Samatar Hassan提供了Kettle支持 RSS的转换例子。在第21章“Web Services”中，RSS部分几乎都是Samatar Hassan的贡献。
- Mike Hillyer和MySQL团队创建并一直维护着Sakila样例数据库。他们的Sakila样例数据库会在第4章有详尽阐述，本书的其他章节也会引用Sakila样例数据库。
- Kasper de Graaf是本书的第四位作者，虽然书的封面只出现了三位作者。他来自DIKW-Academy, 为本书写了Data Vault一章，阐述了自己对这个领域的专业见解。Johannes van den Bosch审阅了Kasper的工作，让Data Vault这一章更加清晰完整。
- Bernd Aschauer和Robert Wintner 均来自 Aschauer EDV（http://www.aschauer-edv.at/en），两人为第6章SAP部分提供了例子和屏幕截图。
- Daniel Einspanjer来自Mozilla Foundation，为第7章提供了转换例子。

感谢你们的辛勤工作，为本书做出了巨大的贡献。同时感谢所有Pentaho和Kettle的技术专家，他们在繁忙的工作日程中抽出时间审阅此书，使得此书的羽翼丰满。

我们还要感谢Kettle和Pentaho社区。在整个写书前后，社区中的伙伴们对ETL、数据整合和Kettle提出了宝贵的意见和建议。我们希望这本书对于正在使用和计划使用Kettle的朋友们起到实际的作用。读者是判定我们是否成功的最好证明。如果成功了，我们要感谢Kettle和Pentaho社区中的每一位伙伴。

感谢Kettle软件项目的所有贡献者和研发人员。我们大家都是Kettle的热心使用者。我们热爱Kettle, 因为它用直接有效的方法解决了日常中的数据整合问题。使用Kettle工作快乐无比，这也是完成这本书的巨大动力。

最后，感谢Wiley出版社。感谢他给我们提供了写这本书的机会，感谢他完美的管理体系和对我们的巨大支持。我们还要尤其感谢项目主编Sara Shlaer。尽管我们总是延期交稿，但是Sara一直耐心鼓励着我们，她的建议、关心、镇静和无以伦比的幽默感让本书及时地在截止日期前完成。另外，还要感谢执行编辑Robert Elliot，感谢他对我们小小团队的信任以及他对*Pentaho*

*Kettle Solutions*所做的努力。

——作者的话

由于当时正忙于发布Kettle 4，日程安排非常紧，所以单靠一个人的力量写这本书是极其困难的。感谢Jos和Roland，感谢他们的认真和专业，我们一起努力完成了这本书。也同时感谢他们接受了我的邀请。即使写书的过程很痛苦，但是和Jos、Roland一起合作，这一切都是值得的。

在Kettle的源代码还没有公开之前，Kettle还不太为人所知，因为没有太多的人会关注一个不开源的ETL工具。Kettle从当时的无人所知到现在成为世界上最广泛使用的开源ETL工具，这要感谢成千上万的志愿者们，他们帮助解决了很多问题。自从Kettle开源以来，Kettle和Kettle社区都快速发展。Kettle社区的贡献功不可没。正是由于这些研发人员、翻译人员、测试人员、错误报告人员、论坛参与人员、那些带来新想法的人们，甚至那些爱抱怨的人们，造就了今天的Kettle。这里我尤其要感谢我们社区中一个很重要的成员，Pentaho。Pentaho CEO Richard Daley和他的团队自从加入进来，一直都在支持着Kettle 项目。没有他们的支持，Kettle不会取得今天的成绩。对于我来说，能和Pentaho的伙伴们一起工作是我的荣耀。

我们社区的几位成员对本书的部分技术内容进行了审阅。Nicholas Goodman、Daniel Einspanjer、Bryan Senseman、Jens Bleuel、Samatar Hassan以及Mark Hall审阅了我写的章节。他们提出了中肯的意见和建议。这是我第一次写书，难免有不妥之处，感谢他们花费宝贵时间和辛勤劳动。

——Matt Casters

首先我要感谢和我一起写这本书的两位合著者Jos和Matt。和这样经验丰富、知识渊博的专家们合作，是我的荣幸。希望今后有机会我们能够继续合作。我们彼此不同的技术领域帮助完成了这本书的不同方面。

此外，我还要感谢对我写的章节进行审阅的专家们，Benjamin Kallman、Bryan Senseman、Daniel Einspanjer、Sven Boden和Samatar Hassan。他们的评论、建议、直率中肯的批评促使书的写作更加完善。

最后感谢我博客http://rpbouman.blogspot.com/上的读者们。他们的评论给了我极大的鼓舞。当我公布要写*Pentaho Kettle Solutions*后，收到了许多好的反馈。

——Roland Bouman

回顾2009年10月，那时*Pentaho Solutions*才刚刚出版两个月。Roland和我一致同意不会再写另外一本书。但Bob Elliot找到了我们，建议我们再写一本。的确，我们也一直在讨论这个问题，一直认为如果再写一本书的话，那一定是关于Kettle的书。这也正是Bob想让我们做的，写一本用Kettle解决数据整合问题的书。接着我们很快发现Matt Casters也要成为书的作者，并向我们发出了一同写书的邀请。我们欣然接受。回顾过去，我不敢相信我们取得了这么大的成功。真心感谢Roland和Matt一直以来容忍我，接受我。感谢Bob。尤其感谢Sara，感谢她不懈的努力让我们圆满完成任务。

尤其要对Ralph Kimball说一声谢谢。读者会在书中找到他的观点。Ralph允许我们使用Kimball Group的34种ETL子系统作为这本书许多内容的框架。Ralph也审阅了第5章，他为第5章写出了极其丰富的评论，使得第5章为书中第二、三、四部分奠定了坚实的基础。

最后感谢Daniel Einspanjer、Bryan Senseman、Jens Bleuel、Sven Boden、Samatar Hassan以及Benjamin Kallmann, 他们对我写的章节给予了技术的审阅，他们的评论、疑问和建议增添了书的价值。

——Jos van Dongen

介绍

50多年以前，以普通应用为主的计算机首次出现，计算机的应用进而延伸到科学和商业领域里。在过去，大多数公司基本只有一台计算机，连接一台显示器和一台打印机。因此，想要整合存储在不同系统中的信息是不可能的。然而，19世纪70年代后期，关系数据库的出现改变了这一切。80年代，计算机和数据库对公司信息的采集进一步升温。毫无疑问，这一切推动了一个新行业的产生，正如IBM研究者Dr. Barry Devlin 和Paul Murphy的会议论文所说："一个商业和信息系统的建筑"（1988年 IBM Systems Journal首次出版，27卷，1号）。作为"商业报表的单一逻辑数据库"， 商业数据库的概念首次出现。随后不到五年的时间，Bill Inmon 出版了*Building the Data Warehouse*，这本书具有划时代的意义，进一步推动了构建逻辑数据库的概念和科技。

数据仓库领域的一个重要概念就是数据整合。数据整合就是把不同数据库中的数据组合到一起，对外提供统一的数据视图。数据整合最典型的案例就是整合存货数据和订单数据。数据整合的另一个案例就是把各个部门的客户关系管理系统中的客户信息整合到公司客户关系管理系统中。

说明： 阅读本书，你会看到有两个术语"数据整合（data integration）"和ETL（extract, transform, and load）在交叉使用。尽管技术上并不完全正确（ETL 只是数据整合的一种，见第1章），但大多数开发人员都把这两个词视为同义词，很多年来我们也一直如此。

在理想情况下，不应该存在数据整合的需求。企业运营所需要的所有数据都应该保存在一个单一的系统里，所有的主数据都应该100%正确，所有分析和决策使用的外部数据都应该自动链接到我们的数据上。这个系统可以存储所有的历史数据，也可以以秒级的速度查询和分析数据。

遗憾的是，我们并非生活在理想世界中。在现实世界中，大多数的组织为各种目的使用不同的系统。有CRM 系统，有会计系统，有销售和销售支持系统，有物流系统，有库存管理系统，等等，这个列表还在不断增长。更糟糕的是，业务上相同的数据可能会保存在不同的系统中，但内容并不完全相同。

为了能够处理所有这些问题，需要创建一个单一的、合成的、一致并且可靠的数据库来展现分析数据，由此看来，数据整合工具是必不可少的。目前最流行、功能最强大的数据整合工具是Kettle，也被称为Pentaho Data Integration。这正是本书讨论的主题。

Kettle的起源

Kettle起源于十年以前，本世纪初。当时，ETL工具千姿百态，比较流行的工具有50个左右，ETL框架数量比工具还要多些。根据这些工具的各自起源和功能可分为以下4种类型，如图1所示。

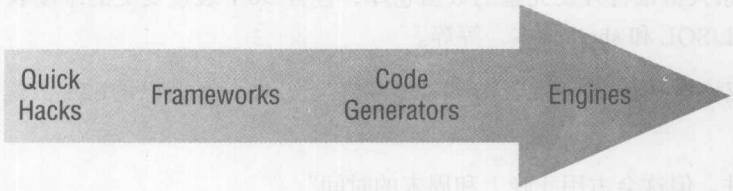

图1 ETL工具

- **快速代码修改（Quick Hacks）**：这类工具主要用于抽取数据和加载文本文件。很多这类工具现在仍在使用。“hacker”和“hacking”这样的词汇现在成了贬义词。商业智能本身比较复杂，在很多情况下quick hacks是项目成功与否的关键，而且能够节省时间和成本。这种quick hacks 的解决方案主要由咨询公司创造，一般是跟随项目的一次性解决方案。
- **框架（Frameworks）**：通常情况下，当一个商业智能顾问同时做几个相似的项目时，代码可以在小范围内调整就可以应用到不同的项目上。这样说来，每个咨询公司都有自己的framework，因为这些frameworks 帮助构建了ETL程序。而且通过改变参数，就可以完成不同项目的抽取数据、加载、日志、信息捕获、数据库连接等工作。
- **代码生成器（Code Generators）**：当 frameworks上再加一个开发界面作为额外的抽象层时，就可以基于元数据为某个平台（C、Java、SQL，等等）生成代码了。这种代码生成器的种类很多，有的是生成简单代码，还需要你手工维护代码，有的功能强大的ETL工具可以生成各种代码。这类 ETL 工具一般也是由一些比较著名的咨询公司开发的。
- **引擎（Engines）**：随着 ETL 技术的不断发展， ETL引擎技术出现了，这样就不必再生成代码。引擎可以执行参数化的可配置的ETL过程，也就是对 ETL 本身的描述。这样就避免了生成代码、编译、部署等困难。

从大概的统计来看，超过一半的项目使用的是快速代码修改和框架的方法。剩下的项目里大部分使用各类基于代码生成器的ETL方法。基于引擎的ETL方法只在少量的ETL项目里使用，主要在非常大的 ETL 项目里使用。

说明： 十年前，开源的ETL项目非常少。Enhydra Octopus 是其中的一个，它是一个基于Java的，代码生成器类型的ETL工具（可以从 www.enhydra.org/tech/octopus 下载）。很多用户从这个开源项目受益，所以这个项目现在仍旧可以使用，它也是开源项目可以持续不衰的一个例子。

正是在这样的背景下，Kettle 软件的作者，本书作者之一，Matt Caster 每天忙于咨询工作，为不同项目不停地修改 ETL 代码和框架，部署各种 ETL 工具的代码生成器。

时间回退到 2000 年，Matt 还是商业智能顾问，他通常的职位是数据仓库架构师或管理

员。在这样的职位上，经常需要把企业数据转换为业务需要的各种信息。通常这些工作是在没有一个很强大的ETL工具的情况下完成的，因为这类的ETL工具都很昂贵。尤其对中小型项目来说，不可能使用这么昂贵的工具。在这种情况下，没有什么选择：一次次面对同样的问题，使用各类框架或代码生成技术来完成自己的工作。他做过相关的工作包括：用C语言和嵌入式SQL（ESQL/C）从Informix里抽取数据；用Microsoft VB开发一个抽取工具，从IBM AS/400 Mainframe系统里抽取数据；为一家大型银行开发完整的数据仓库，包含 90个缓慢变更的维度表和 35个事实表；完全手写Oracle PL/SQL 和 shell 脚本，等等。

但是这些经历，使Matt明白应该做些什么。在 2001 年Matt就有了开发一个自己的ETL工具的想法。

Matt：“我想写一个 ETL 软件。但这会占用我晚上和周末的时间”

Kathleen（Matt 的夫人）：“Oh，太好了！要用多长时间？”

Matt：“如果一切顺利，第一个能运行的版本应该用三年时间，第一个完整的版本要用五年。”

Kettle 的设计

因为十年来一直在同各种 ETL工具做斗争，所以Matt 确定了 Kettle 的一个主要设计目标是尽可能开放。主要就是指：

- 开放，可读的元数据格式（XML）。
- 开放，可读的关系型资源库格式。
- 开放的API。
- 容易安装（少于2分钟）。
- 对各类数据库开放。
- 容易使用的图形用户界面。
- 容易传送数据。
- 容易把数据转换成各种格式。

在最开始的两年，进度比较缓慢，大量的工作都用于研究这个新的ETL工具应该具有哪些功能。开发一个并行化的ETL引擎就是在这段时间确定的。多年的基于代码修改方式、框架方式、代码生成器方式的ETL项目惨痛经验，使Matt 确信，新的 ETL工具一定是应该基于引擎方式的。

因为Matt以前主要使用C语言进行开发，所以开始时，他使用客户端/服务器这样的代码，来测试在不同的处理器和服务器之间传递数据的性能。通过很多不同场景的测试，他明白ETL性能瓶颈应该主要在于数据的编码和解码。所以，Kettle 的一个设计原则就是尽量不做数据的转换。如今这一原则仍在Kettle 中可以体现出来。

说明： Kettle 一词起源于“KDE ETTL Environment”，因为最开始的计划是在K Desktop Environment（www.kde.org）上开发这个软件。在这个计划被取消后，才把它重命名为“Kettle ETTL Environment”。

由于缺少各种关系数据库的驱动，最终还是采用了当时较新的Java开发语言。在2003年，又选择了SWT（Standard Widget Toolkit）来开发界面，因为Matt不喜欢Java AWT（Abstract Window Toolkit）的性能和界面风格。而SWT使用了本地操作系统的组件库，因此性能更好，而且界面也更符合操作系统的风格。

因为是Java新手，又要开发这个复杂的ETL工具，可以想象在开始的第一年，Kettle的代码库就是一团糟。代码没有包；没有结构，命名方式非常可笑（C语言的风格）。没有做异常处理，经常发生崩溃。这样的Kettle版本唯一能做到的事情就是它可以工作了。它可以读取文本文件，读取和写入数据库，而且它有了一个JavaScript 脚本步骤，用来解决各种复杂问题。而且它还非常灵活和易于使用。这毕竟是一个商业智能工具，而不是一个Java项目。

但有一点很清楚，Kettle 需要改进。于是有人在这个时候提供了帮助，朋友 Wim De Clercq，他是ixor（www.ixor.be）的合伙人，也是一位高级Java 架构师。他解释了很多Java 概念，如包、异常处理等。慢慢地时间就在学习Java设计模式（如单例模式等）中流走了。

听朋友的建议就意味着大量的代码重构。所以第一个版本后，Matt 把每个周末的时间都用到了代码重构上，重新写了几万行代码。逐渐的，几个月以后，事情就开始向着好的方向发展。

Kettle 获得机会

Matt和朋友、同事、其他高级商业智能（BI）顾问们分享了Kettle的最初成就，听取了他们对Kettle 的想法，从那时开始 Kettle 开始逐渐被大家了解。后来在2004年，Matt 把 Kettle 部署在了比利时Flemish 交通中心（www.verkeerscentrum.be），Kettle从遍布比利时的几千个数据源整合几亿条数据。当时项目小组没有时间写代码，也没有钱去买那些商业ETL工具，所以项目就使用了Kettle。这个比利时交通中心的项目提升了Kettle 的功能和性能，在这个时期，Kettle 发展非常迅速。因为有了各种数据的测试用例，Kettle可以更好地支持各种数据库。也是在这个时期，世界各地的开发人员都可以免费下载Kettle版本使用（免费下载，还没有开源）。

最开始反馈不多，但大多数的反馈都是积极的。最有趣的一个反馈来自于一个德国的开发人员，Jens Bleuel，他问是否可以把第三方的软件整合进Kettle，他希望把SAP/R3 connector整合到Kettle。Kettle 当时的版本是1.2，还没有插件架构。Jens Bleuel的这个需求是当时开发插件框架的主要原因，后来就形成了 Kettle 2.0。最后直到2004年底，才开发完成。这是一个相对完整的版本，支持缓慢变更维度、杂项维度、28个步骤和13种数据库。直到这个时候，这个工具的真正潜力才发挥出来。然后，Jens Bleuel 创建了Kettle 的第一个插件，ProSAPCON，用来从SAP/R3 服务器读取数据。

Kettle 走向开源

在那个时期有很多令人激动的事情，Matt 和 Jens 打算把Kettle商业化，通过kettle.be 网站和Jens所在的Proratio（www.proratio.de）公司来销售Kettle。

Kettle 还是在取得进展，也有一些试用的请求。但是，做一个完整的ETL工具的开发和销售是一项艰巨的任务，不是靠个人可以完成的。而且Matt还发现，用 Kettle来工作很有乐趣，而销

售Kettle 却没什么乐趣。他不得不找到一种方式，可以把精力聚集到有乐趣的开发工作中。所以最后在2005年的夏天，Matt决定把Kettle 开源。这样Kettle可以自己卖自己，并吸引更多的人参与开发工作。

2005年12月份，Kettle 2.2 的代码第一次发布后，反响非常强烈。JavaForge 上第一周的下载数量就达到了35 000 次，消息迅速在全世界传播。

因为很多开源项目到最后都成了无人管的项目，为了避免这种情况的发生，要尽快为Kettle项目构建一个社区。这就意味着，在随后的几年可能需要回答上千封的电子邮件和论坛帖子。幸运的是，Kettle很快获得了开源商业智能公司Pentaho的帮助（www.pentaho.com），Pentaho获得了源代码的版权，Matt也成为 Pentaho的内部人员，带领Kettle项目的开发，随后Kettle改名为Pentaho Data Integration。

Kettle同样获得了来自世界各地的开发人员、翻译人员、测试人员、文档编写人员的帮助，没有这些人的帮助，Kettle 也不能像今天这样发展迅速。

关于本书

本书起源于2009年8月，此时Roland 和 Jos 的第一本书《Pentaho 解决方案》（即*Pentaho Solutions*）已经出版。正如跑过马拉松的人，他们发誓以后“再也不跑了”。但是当他们看到第一本的热烈反响，他们的态度慢慢地转变过来了。他们想如果还要写一本书的话，就要写Kettle和数据整合。当《Pentaho 解决方案》一书的出版人Bob Elliot问他们能否再写一本Kettle 书的时候，Kettle书的主题和目录其实都已经确定下来了。另外，还让他们受到鼓舞（和吃惊）的是，他们的配偶都鼓励他们继续写作。还有其他的好消息，当他们邀请Matt Casters为本书审核时，Matt Casters也希望参与到本书的写作中，这当然是求之不得的事情了。

当时写《Pentaho 解决方案》一书的动力和主要原因，如今仍然存在：企业开始逐渐了解商业智能能给企业带来什么价值，并逐渐认识了开源和免费商业智能的解决方案。这些商业智能解决方案要求事先把数据集中在一起，然后再进行分析、报表和仪表盘等工作。所以商业智能项目的开始阶段都要先进行数据的整合，本书就可以帮助你进行数据整合的工作。

在过去的十多年里，各种类型的开源软件逐渐被大家认识和接受，来代替那些价格昂贵而不灵活的商业软件。人们常会错误地认为开源软件就是没有成本的，尽管从软件License的角度看这是正确的，但一个商业智能解决方案（永远）不会免费。在一个解决方案里有硬件成本、方案实现的成本、维护的成本、培训和移植的成本，等等，所有这些费用加一起，软件License的费用只占其中一小部分。开源降低了软件本身的成本，而且任何人都可以获得源代码，都可以发现源代码的bug，这样也提高了代码的质量 。开源软件一般都是基于开放的标准和通用的编程语言（大部分是Java），这样使开源软件更灵活、更具扩展性。而且大多数开源软件都不局限在某个操作系统平台上，这样更扩展了软件的灵活性和自由度。

但开源软件缺少文档和手册。很多开源软件提供了高质量的代码，但开发人员因为更多地关注于软件本身而没有太多的时间写文档。尽管你可以找到很多关于Kettle的零散的信息来源，但我们觉得有必要为正在探索Kettle并构建数据整合解决方案的ETL开发人员提供一个完整的信

息来源。这就是本书的目的——帮助你使用Kettle 构建数据整合的解决方案。

谁需要读这本书

本书适合于想知道如何用Kettle来构建 ETL 解决方案的人。例如寻找低成本的ETL方案的IT经理、想扩充自己知识和技能的IT 专家、负责开发ETL方案的BI或数据仓库顾问。可能你是一个有丰富经验的软件开发人员，但对数据整合领域不太了解；也可能你是一个经验丰富的使用某些商业软件的 ETL开发人员，但对于Kettle不太了解。无论哪种情况，因为有了这本参考书，你就可以直接上手了。当然读者如果能有商业智能、数据库等方面的知识更好，但在本书的开始部分会介绍一些这方面的知识。当然本书也会介绍数据整合方面的概念，但重点还是讲解如何把这些概念转换为实际的工作方案。这也是本书被命名为《Pentaho Kettle 解决方案》的原因。

阅读本书的前提

只要能做到下面两件事，你就可以开始阅读本书了：有一台计算机，能连接到互联网。本书使用到的所有软件都可以从互联网下载。对计算机的系统也没有太多的要求，有1GB内存和2GB空闲硬盘的，一般在4年之内的计算机都可以很好地运行Kettle作业。

本书一些章节里有提供软件下载的URL地址。对于Pentaho软件，除了源代码，还提供有4种类型的软件版本：

- GA（General Availability）releases：这是稳定的发布版本，并不是最新的版本，但是最可靠的版本。
- Release candidates：候选版本，可能会成为下一个发布版本的版本，里面可能还会有一些未发现的bug。
- Milestone releases：最新的里程碑版本，里面都会有一些新功能。
- Nightly builds：每天的build版本，最新的版本，也是最不稳定的版本。

在写作本书的时候，我们使用的都是每天的build版本。在写作本书时，Kettle 4.0的GA版本已经发布了。也就是说，在你读本书的时候就可以使用没有多少bug的正式发布的稳定版本。

Kettle社区版的下载地址是：http://wiki.pentaho.com/display/COM/Community+Edition+Downloads。

从本书可以学到什么

本书会教给你：

- 数据整合是什么，数据整合的价值。
- Kettle 解决方案的概念基础。
- 如何在单机和客户/服务器环境下安装和配置Kettle。
- 如何使用MySQL的Sakila演示数据库构建一个完整的端到端的ETL解决方案。

- 34种ETL子系统，如何使用Kettle 实现这34种子系统。
- Kettle 如何完成数据抽取、清洗和确认、处理维度表、加载事实表、操作OLAP 立方体。
- Kettle 的开发生命期。
- 如何在Kettle 环境里使用Pentaho 的敏捷BI 工具。
- 如何调度和监控作业和转换。
- 多个开发人员如何协同工作，如何管理不同版本的ETL方案。
- 什么是数据的血统分析、影响分析、审计。Kettle 如何支持这些概念。
- 如何利用分区、并行化和动态集群技术提高Kettle 的性能。
- 如何使用复杂文件、Web 服务和Web API。
- 如何使用Kettle 加载基于Data Vault 模型的企业数据仓库。
- 如何在其他应用中集成Kettle，如何通过二次开发扩展Kettle。

本书如何组织

本书解释了ETL的概念、技术和解决方案。本书的很多场景都使用了MySQL的Sakila数据库，除此之外，我们还通过其他不同的场景来演示不同的概念。如果例子还依赖于其他数据库，我们都确保这些例子和MySQL数据库（5.1版本）兼容。

这些例子都提供了在实际环境中应用所必需的技术细节。例子范围覆盖了从部门级的数据集市到企业级的数据仓库。

第一部分：开始

本书的第一部分在一个较高的层次上，对Kettle 的架构、功能等做了一个快速的介绍。这一部分包括下面几个章节。

第1章：ETL入门。介绍数据整合项目的主要概念和挑战。我们介绍事务系统和分析系统的主要区别、ETL适用于哪些场合、如何用ETL解决方案的不同部分来解决数据整合问题。

第2章：Kettle基本概念。介绍了Kettle的一些基本设计原则，以及Kettle 的软件组成。我们介绍了Kettle的基本概念，如作业、转换、步骤、步骤之间的连接等，以及它们之间的相互作用。另外本章还介绍了Kettle 如何和数据库交互、设置数据库特定的参数。本章还提供了Kettle 界面的一个快速教程。

第3章：安装和配置。介绍如何获得Kettle 软件，如何安装Kettle。Kettle包含哪些组件，它们之间的关系，如何利用它们构建ETL解决方案。最后，我们解释不同的配置选项、配置文件和配置文件的位置。

第4章：ETL示例解决方案——Sakila。基于MySQL Sakila数据库，介绍一个完整的ETL解决方案的例子。使用本章的星型模型，你可以了解如何处理缓慢变更维度以及如何加载事实表。另外本章的一个重要主题就是如何通过映射步骤重用转换。