



程序天下事

——编程语言栏目 2007-2012 年内容汇总

编程语言 2011 年第 12 期——通用智能语言 UBL.....	3
编程语言 2011 年第 11 期——Google 结构化 Web 编程语言 Dart.....	4
编程语言 2011 年第 10 期——开源的系统级编程利器：D 语言.....	5
编程语言 2011 年第 09 期——函数式编程语言 F#.....	6
编程语言 2011 年第 08 期——年度编程语言锁定 Objective-C.....	7
编程语言 2011 年第 07 期——移动开发新宠：Lua 语言.....	8
编程语言 2011 年第 06 期——云端的非结构化数据存储服务.....	9
编程语言 2011 年第 05 期——JVM 上的新语言：Ceylon.....	10
编程语言 2011 年第 04 期——新闻标识语言 NewsML.....	11
编程语言 2011 年第 03 期——JavaScript 代码生成器 CoffeeScript.....	12
编程语言 2011 年第 02 期——跨越 JVM 与 .NET CLR 的 Fantom.....	13
编程语言 2011 年第 01 期——DOI：国际化的数字对象标识符.....	14
编程语言 2010 年第 12 期——LEGO 机器人控制语言：NXT-G.....	15
编程语言 2010 年第 11 期——超越 GCC 的编译环境：LLVM.....	16
编程语言 2010 年第 10 期——达尔文信息分类架构 DITA.....	17
编程语言 2010 年第 09 期——并发编程好手：Clojure 语言.....	18
编程语言 2010 年第 08 期——Go 语言发展近况.....	19
编程语言 2010 年第 07 期——RailsConf 2010 大会概览.....	20
编程语言 2010 年第 06 期——了解 REBOL 语言.....	21
编程语言 2010 年第 05 期——动态语言实现 Map-Reduce 应用.....	22
编程语言 2010 年第 04 期——Rails 3.0 新版本预览.....	23
编程语言 2010 年第 03 期——搭上 LuaJIT 2.0 的高速列车.....	24
编程语言 2010 年第 02 期——Django 1.2 开发进展与版本新特性.....	25
编程语言 2010 年第 01 期——JVM 上动态语言发展现状.....	26
动态语言 2009 年第 12 期——动态语言的 GIS 开发插件.....	27
动态语言 2009 年第 11 期——云计算基础设施中的动态语言项目.....	28
动态语言 2009 年第 10 期——Python 语言的拓展应用.....	29
动态语言 2009 年第 09 期——Django 及国内 Python 社区进展.....	30
动态语言 2009 年第 08 期——支持动态语言的 Android 平台.....	31
动态语言 2009 年第 07 期——动态语言的代码托管云服务.....	32
动态语言 2009 年第 06 期——你是否关注 Twitter 的性能改进.....	33
动态语言 2009 年第 05 期——JavaFX 的挑战与机遇.....	34
动态语言 2009 年第 04 期——Python 近况全接触.....	35
动态语言 2009 年第 03 期——Django 是否在重复制造轮子？.....	36
动态语言 2009 年第 02 期——动态语言社区回顾.....	37
动态语言 2009 年第 01 期——是谁让动态语言归来.....	38

动态语言 2008 年第 12 期——支持动态语言的 GlassFish v3	39
动态语言 2008 年第 11 期——JRuby on Rails 上的项目宿主平台 Kenai	40
动态语言 2008 年第 10 期——开源世界的节日	41
动态语言 2008 年第 09 期——JVM 之上的动态语言	42
动态语言 2008 年第 08 期——聚焦 Django 1.0	43
动态语言 2008 年第 07 期——Rails 2.1 与 Python 科学计算工具包	44
动态语言 2008 年第 06 期——Python 与 Rails 发布版稳步推进	45
动态语言 2008 年第 05 期——动态语言的虚拟机平台	46
动态语言 2008 年第 04 期——SaaS 模式的 Rails 开发平台	47
动态语言 2008 年第 03 期	48
动态语言 2008 年第 02 期	49
动态语言 2008 年第 01 期	50
动态语言 2007 年第 12 期	51
动态语言 2007 年第 11 期	52

关于作者：高昂

中国标准化研究院，高新技术与信息标准化研究所助理研究员。关注开源社区，也是 OSGeo 中国和 InfoQ 中文站成员。

<http://www.gaoang.com/>



编程语言 2011 年第 12 期——通用智能语言 UBL

近日, OASIS 发布了通用商业语言 UBL (Universal Business Language) 的 2.1 版本征求意见稿, 这个版本在保持与 UBL 2.0 版本良好兼容性的同时, 面向电子商务应用对 UBL 语言进行了多处增补。在新版本中, UBL 还针对财经业务进行了优化和增强, 改进了 UBL 与财务系统基础设施的接口互操作性, 并在标准 UBL 语言基础上添加了拓展, 以实现借助 XAdES 格式对 UBL 文档进行 XML 高级数字签名的功能。

自 2004 年由 OASIS 技术委员会制定第一个版本以来, 通用商业语言 UBL 就一直致力于促进电子数据格式的标准化, 来保证企业间信息与数据交换的有效性和一致性。在 UBL 规范中, 以通用 XML 库的形式给出了电子商务文档格式的定义, 包括产品预售、业务订单、数据交换、发票单据、付款支付等各个业务环节。在设计上, UBL 遵循面向对象理念, 能够在不同业务领域间有效进行条目重用。UBL 具备严格的标签命名方式, 设计者同时借助了 CCTS 语义原子性方法来保证 UBL 规范对信息描述的精确性。此外, 在可拓展性方面, UBL 文档的参考引用、标识符、代码段可以根据具体的业务环境进行定制和修改, 并能够借助 XML 映射将 UBL 文档转换成各类其他文档类型。

对于特定的大型行业信息化系统建设来说, 准确清晰的业务逻辑描述往往比具体功能的技术实现更为关键。UBL 提供了将电子商务交易过程中纸质文件转换为以 XML 格式为载体的非结构化信息标准文本格式的方法和途径, 借助通用的标准规范, 来保证企业在进行电子商务在线交易时, 使用标准通用的格式进行文档和数据交换。

UBL 规范吸纳了业界多个数据标准化组织共同参与, 以便于 UBL 能够直接植入已有的在线交易、业务审计、记录管理等业务系统中使用。此外, UBL 自 2.0 版本以来, 就与电子数据交换格式 EDI(Electronic Data Interchange)及其派生的 XML 标准保持兼容, 以便融合 EDI 中定义的贸易、运输、保险、银行、海关等行业信息交换格式, 在以贸易为中心的电子商务应用系统中进行处理。

创建于 1993 年的结构化信息标准促进组织 OASIS 一直推进着 UBL 规范的发展和更新。OASIS 最初以标准广义标记语言 SGML (Standard Generalized Markup Language) 组织的形式成立, 旨在推动产品互操作性架构的建立以及 SGML 语言的定义。在 1998 年更名为 OASIS 开放组织后, 向将业务拓展到更为广阔的范畴, 并成为推动电子商务标准发展、融合及采纳的非营利性国际组织, 为数据安全、网络服务、接口一致性、电子交易、产品供应链、公众服务, 及企业间互操作提供通用性标准。OASIS 制定的各种规范帮助商业需求被精确描述, 减少标准重复开发并保证标准的延续性, 在协调各行业标准化组织的基础上通过一系列基础性工作推进软件供应商之间的合作。更为重要的是, OASIS 各项标准均以免费且公开的形式向公众开放使用。

目前 UBL 已经在欧洲政府采购联盟 PEPPOL (含 12 个欧盟成员国), 电子货运管理 (EFM) 等多个领域内得到成功应用, 如果您所在的机构感兴趣使用 UBL 定义的低成本、迅捷、安全的电子文档处理方式来提高工作效率, 升级传统业务流程, 不妨访问 UBL 官方网站 (ubl.xml.org), 更进一步了解 UBL 通用智能语言规范的细节内容。

编程语言 2011 年第 11 期——Google 结构化 Web 编程语言 Dart

GOTO 大会 (gotocon.com) 是面向企业级软件开发筹办的全球性技术开发大会, 参会者包括开发团队领袖、架构师、项目管理者、技术媒体等。GOTO 大会组织者同时也是企业级软件的设计者和开发者, 这也使得 GOTO 大会承载了来自技术型企业和开发社区的最新动态。

今年 10 月在丹麦举办的 GOTO 大会上, 最吸引人的内容是 Google 新编程语言的首次亮相。在大会开场的主题演讲中, Google 工程师捷拉德·布拉查(Gilad Bracha)和拉尔斯·巴克(Lars Bak)为开发者揭开了新语言 Dart 的面纱。

Dart 语言面向结构化 Web 编程设计, 是继 Go 语言后 Google 推出的第二门编程语言。Dart 中文释义为“飞镖”, 象征着这门语言灵巧高效的特性。Dart 的设计目标是创建面向 Web 应用开发的结构化语言, 用以创建灵活的网页程序。为便于开发者掌握并使用, Dart 语法在设计上尽可能以开发者熟悉的形式出现。在浏览器兼容性方面, Dart 考虑了各种浏览器差异, 并在各浏览器平台上保持统一的性能与表现。

在门类众多的轻量级编程语言中, Dart 具备不少吸引开发者使用的独特优势。首先 Dart 简化了 Web 应用的开发工作量。其次, Dart 应用借助浏览器展示给用户, 在具备良好平台独立性的同时, 无需用户进行任何程序安装操作, 并且对现代浏览器、移动设备、平板电脑的支持都具有良好表现。

Dart 在定位上并非与 JavaScript 竞争, 而是要填补 Web 编程方面缺少面向对象编程结构、静态类型、难以进行大型应用开发方面的空白, 并为传统 JavaScript 编程带来大幅度的性能改善。

Dart 代码运行阶段, 可以依托浏览器内置的 JavaScript 引擎执行程序, 同时 Google 鼓励各浏览器厂商在应用中绑定 Dart 虚拟机以实现 Dart 的原生支持。当然, Chrome 浏览器已计划在未来版本中内置原生的 Dart 虚拟机运行时。根据性能测试数据, Dart 虚拟机装载 5 万行 Dart 代码仅用时 640 毫秒, 而从快照中加载相同的程序则更加高效, 仅需用时 60 毫秒。

Dart 开发工具支持上, Google 基于 Eclipse 提供了轻量级的开发组件, 用于编写代码并预览 Dart 应用。同时 Google 还正在开发一个名为 Brightly 的云端 IDE 提供给开发者, 值得一提的是, 这个 IDE 也是使用 Dart 语言开发实现的。

目前 Dart 语言站点 (dartlang.org) 上展示了语言规范、实例教程等内容。在 Dart 项目站点 (dart.googlecode.com) 提供了 Dart 类库、样例代码、Dart 虚拟机及源代码、Dart 转换 JavaScript 编译器等内容。

Google 面向现代浏览器设计的 Dart 语言能够给 Web 编程带来良好的开发效率和优异的性能。随着浏览器应用规模化开发时代的到来, 以及移动设备对浏览器应用需求的提升, 我们期待新生的 Dart 语言为 Web 开发注入新的活力。

编程语言 2011 年第 10 期——开源的系统级编程利器：D 语言

在 Digital Mars 公司的 D 语言首页上，放置着这样一段话“计算机编程语言门类众多，其来源大致可分为两种途径。一类新语言来自于极具创新精神的学术界；而一类则源于关注研发结合和互联网的新技术公司。目前正是在实现编译器的实践经验中诞生一门新语言了。”在九月份的 TIOBE 语言流行度排行榜中，正是这门支持内嵌汇编的 D 语言，以其系统级编程的特性而被越来越多的开发者选用而重新入围了排行榜前二十位。

D 语言是由 Digital Mars 公司开发的面向对象、指令式、多范型的系统级编程语言。诞生于 1999 年的 D 语言，在其设计开发十年后的 2009 年开放了 D 编译器 DMD 的源代码，包括 D 1.0 和 D 2.0 的编译器前端和后端源代码，分为 Windows, Mac 和 Linux 三个版本。编译器前端以 GUN GPL 和 Artistic 开源协议授权。

可以说，D 语言是由数十年来实现多种语言编译器的经验积累而形成，源自 C++，并为改进和超越 C++ 而生。D 语言对 C++ 的部分特性进行了重新设计，其模板在 C++ 的基础上做了扩充，并在语法设计上借鉴了 Java、C# 等多种流行语言的优点。D 语言在提供 C++ 这样强大能力的同时，又具备与动态语言相媲美的开发效率。

C++ 应用广泛的一个重要原因是 C++ 进行操作系统底层编程的能力。D 语言同样也提供了底层编程的特性，并加入了完善的内嵌汇编支持。内嵌汇编的特性让系统级开发者能够在标准 D 语言代码中嵌入目标平台的汇编指令，来完成高级语言难以实现的工作。这些处理任务包括底层驱动程序开发、获取任务堆栈信息、获取 SP 或 RA 等寄存器的状态、视频解码等各个层面。除此之外，在某些特定的数值运算中，如带进位加法运算，开发者常嵌入汇编指令来增强程序代码执行的效率。

D 语言的系统级编程特性，成为 D 语言区别于 Java、C# 等其他高级语言的主要特性。如 Java 语言在语法级别上不提供内嵌汇编的支持，如需在代码中嵌入汇编指令，只能借助于 JNI 的 Java 本地调用，通过引用 C/C++ 代码来间接嵌入汇编指令。而 D 语言则可以直接与只接受汇编指令的处理器进行交互，直接执行汇编指令集来保证操作的精确性和原子性。

尽管现代的高级语言编译器已在编译优化上做了足够大的功夫，但高级语言编译器为了保证通用性，往往只编译生成通用指令集，没有针对特定处理器的专用指令集进行优化，且无法生成新型处理器增加的高性能操作指令集。无疑，D 语言将是内嵌汇编的最佳语言，它带来的便利让开发者撰写最高性能代码来充分发挥硬件潜力变得触手可及。

编程语言 2011 年第 09 期——函数式编程语言 F#

源于微软研究院的 F# 语言因其优良的设计和强大的并行编程能力, 正得到越来越多 .Net 开发者的选用。在本月的 TIOBE 语言流行度排行榜中, F# 语言首次进入了 TIOBE 排行榜的前二十位。F# 是微软 .NET 框架环境下的静态类型化函数式编程语言, 支持以面向对象或泛型编程等多种风格来撰写程序代码。

F# 基于函数式编程语言 Objective Caml (OCaml) 设计, 具有 OCaml 常用的核心语言功能, 以及函数式编程语言的其他特性。OCaml 是在 Perl 之外, 又一门以骆驼为吉祥物的编程语言。OCaml 被选中的部分原因是它提供了函数式、命令式和面向对象的混合编程风格支持。在 OCaml 语言中, 函数式能够像变量一样被方便的在程序中传递, 并且 OCaml 是能够自动侦测程序范型的函数式编程语言。

F# 语言最初由微软研究院的首席研究员 Don Syme 设计, 设计者 Don Syme 于 1999 年从剑桥大学计算机实验室获得博士学位, 他还参与了 C# 泛型和 .NET CLR 的设计工作。在语法设计上 F# 具备优雅的结构, 同时 F# 被设计为类型安全且具备良好性能的编译语言。在微软决定将 F# 进一步推广应用之后, F# 被转移到微软专门的开发部门维护和更新, 并在 .NET Framework 和 Visual Studio 2010 开发环境中为 F# 提供了全面支持。此外, 开发者还能够 Visual Studio 2008 中使用 F#, 或借助 Mono 在 Linux 系统上使用 F#。

作为微软支持的第一个函数式语言, F# 在项目中被越来越多的开发者选用, 这与 F# 对程序并发和异步编程的良好支持密不可分。微软为 F# 添加了不少简化开发者处理程序并行和异步编程的便捷特性, 以辅助开发者轻松完成多核并发和 Web 分布式系统的应用开发。在常见的开发模式下, F# 被用于撰写安全并发和异步处理相关的程序组件, 用户界面则使用 Visual Basic 或 C# 来编写, 而最终产品由 F# 实现组件和其他 .Net 开发语言组织而成。

不仅如此, F# 目前还被广泛用于 Xbox 360 游戏平台上的代码开发中, 为开发者提供了在 XNA Game Studio 环境下的 F# 编程支持。微软研究院曾使用 F#、TrueSkill 以及 XNA 开发了一款名为 “The Path of Go” 具备故事情节的 3D 围棋游戏, 以展示 F# 在游戏开发和人工智能领域方面的并发编程能力。

感兴趣的开发者可以阅读 Don Syme 及其他几位 F# 语言设计者共同撰写的《Expert F# 2.0》一书, 来深入学习并使用 F# 语言。也可以阅读游戏开发者 Giuseppe Maggiore 撰写的《FRIENDLY F# with game development and XNA》一书, 来学习 F# 语言在游戏开发中的应用。

编程语言 2011 年第 08 期——年度编程语言锁定 Objective-C

TIOBE 编程语言排行榜每年初都将统计评出上年度的最佳编程语言大奖，获奖的编程语言被认为在该年度市场份额增长及用户认可度方面获得了最为突出的成绩。历年获奖的编程语言包括 2004 年的 PHP、2005 年的 Java、2006 年的 Ruby、2008 年的 C 至 2009 年的 Go 语言，这些编程语言都曾拿到年度最佳的殊荣。成为系统脚本事实标准的 Python 语言，因其简单易用并能成功运用在 Web 开发等各种不同类型的应用中，而与 2010 年和 2007 年两次获得年度编程语言奖项。

2011 年已过去一半，从目前市场占有率趋势来看，Objective-C 已经脱颖而出，相对 2010 年的市场占有率增长了 2.68%，毋庸置疑，Objective-C 将成为年度编程语言奖项的得主。其实在 2010 年，在 iOS 平台应用快速增长的推动下，Objective-C 已成为上年度使用率上升最快（增速超过 100%）的语言，但从绝对数量来看，Python 以拥有最大的市场份额增长率而胜出。

Objective-C 设计的初衷是为在 C 语言基础上添加面向对象拓展，目前 Objective-C 已发展成为苹果各系统平台上应用开发的事实标准。Objective-C 包含一个用 C 编写的小巧的运行库，提供给开发者各种实用的操作。Objective-C 的编译可以在现存 C 编译器基础上实现，而不需专门开发一个全新的编译工具。这让 Objective-C 能重用大量遗留的 C 代码、类库等各种资源。目前 Objective-C 代码的编译在 GCC 和 LLVM/Clang 中已得到完美支持，并且 GCC 的最新 4.6 版本还支持多项 Objective-C 2.0 语言的新特性，比如说快速枚举、dot 语法、可选协议方法、类拓展以及新的 Objective-C 运行时 API 等。

Objective-C 的独特之处在于，它将垃圾回收等一些真正高层次的语言特性与 C 语言低层级的函数功能相结合，通过高层次的语言功能提高程序员的生产力。但即便如此，不少开发者仍旧对语言的发展给予厚望。Mac OS 应用 Illuminate 的作者 Andy Finnell 在他的 Blog 中写道，希望在下一个 Objective-C 3.0 版本发布时，增加闭包的支持，并增加匿名函数以方便在函数中与前端 JavaScript 交互，并提供在匿名函数中访问本地变量的功能。

在公司或个人准备启动一个新的软件开发项目时，TIOBE 编程语言流行度排行常被用作检测编程语言或技术先进性的参照尺度，来辅助技术选型的决策。苹果 iPhone 和 iPad 平台市场领导地位的确立和 Apple 各系统不断推陈出新，极大的推进了 Objective-C 语言的普及和流行，使其成为 TIOBE 本年度最佳编程语言。随着苹果公司 Mac OS X Lion 和 iOS 5.0 发布日期的临近，具备更强劲性能的硬件和更丰富易用的平台将很快与苹果的用户见面，这也呼唤更多的开发者加入到 Objective-C 应用开发的行列中来，让这门本年度最佳编程语言为用户实现更大的价值。

编程语言 2011 年第 07 期——移动开发新宠：Lua 语言

TIOBE 编程语言流行度排行榜的统计数据来源于 Google、Bing、Yahoo! 等主流搜索引擎，以及各大技术网站发布的招聘信息，在很大程度上代表了编程语言的流行趋势。在本月的 TIOBE 编程语言排名中，Lua 语言第一次进入了排行榜的前十名。

轻量级脚本语言 Lua 在开发者中正变得越来越流行，这与移动应用开发的升温密不可分。以 iPhone 为代表的移动平台，为不少开发者提供了新的应用展示空间，也为支持 iOS 系统的应用开发带来了巨大商机。包括大获成功的游戏《愤怒的小鸟》在内，不少 iPhone 和 iPad 流行应用都在使用了 Lua 作为游戏脚本。Lua 语言最大的优势在于其轻量级脚本和高效的运行效率，这一得天独厚的优势让 Lua 在以娱乐和用户体验而著称的 iOS 系统之上大有用武之地。

小巧的 Lua 语言来自于巴西里约热内卢天主教大学于 1993 年启动的实验性项目。Lua 语法简洁优美，解释器由标准 C 编写。以快速小巧著称的 Lua 拥有在目前所有脚本引擎中最快的运行速度，这样的特性让 Lua 成为用作嵌入式脚本的最佳选择。Lua 代码几乎能在所有操作系统和移动终端上编译运行，同时 Lua 提供了 LuaJIT 项目 (<http://luajit.org/>)，针对特定目标平台为开发者提供即时编译功能，进一步增强 Lua 的执行效率。

Lua 的设计初衷并非意图构建完整的应用，而是嵌入在应用程序中为应用提供灵活的扩展和定制功能，所以 Lua 仅提供了基本的数学运算和字符串处理等函数库，而并未涵盖程序设计的方方面面。在语言兼容性上，Lua 一开始就瞄准了具有广泛应用场景的 C 和 C++，让 C/C++ 编写的代码可以很方便的调用 Lua 脚本，同时 C/C++ 函数也能够在 Lua 脚本中调用。与 C/C++ 代码的良好交互性，让 Lua 在应用程序开发中有着广泛的应用基础。

Lua 最流行的应用场景是在游戏中作为处理逻辑的脚本，即用来编写游戏的任务脚本，以实现游戏的可配置性和可扩展性。在游戏中 Lua 脚本的加载不需重启服务器，尽管脚本的运行效率低于编译好的二进制代码，但目前 CPU 速度已完全能够让 Lua 脚本迅速解释执行。在以《魔兽世界》为代表的网络游戏中，几乎所有的插件都是使用解释性的脚本 Lua 编写，核心的 C++ 代码只占全部代码中的很小一部分比重。

目前 Lua 语言的最新版本是 5.2 Beta，如果你也准备在移动应用开发中尝试嵌入 Lua 脚本，在熟悉语法之后，便可移步 Lua 开源项目站点 luaforge.net 或 Lua 开发者站点 lua-users.org 的 Wiki 库，下载感兴趣的 Lua 程序脚本，开启 Lua 脚本的学习之旅。

编程语言 2011 年第 06 期——云端的非结构化数据存储服务

近日, Amazon 与 Mark Logic 公司开展合作, 在 AWS 云服务中加入 XML 格式数据的搜索与处理功能, 以针对特殊用户的需求增强云服务的适用性。Mark Logic 为 Amazon 的用户提供运行在 AMI 服务器映像上的 XML 服务器资源, 以及虚拟化的 XML 数据存储服务。

对大多数国内数据库使用者来说, Mark Logic 的知名度与主流数据库厂商相去较远, 用户并不算多。但在业界, Mark Logic 却是非结构化数据管理技术的领导者, 其主要用户覆盖新闻出版部门、政府机构、财经信息服务等不同的专业领域。

Mark Logic 公司的拳头产品 MarkLogic Server 是以文档为中心的领域专用数据库, 专门针对半结构化和非结构化数据进行设计和优化, 能够实现 TB 级非结构化数据资源的全文检索。MarkLogic Server 支持针对 Web 内容、XML 文档和 JSON 内容的 RESTful 和 HTTP 请求。在数据模型组织方面, MarkLogic Server 采用 XML 树状结构组织, 数据查询和检索使用的 DML 和 DDL 语言为 XQuery, 此外, Mark Logic 还是 XQuery 标准发展和应用的推动者。

较之同类数据库, MarkLogic 在技术上有着独特的优势。MarkLogic 始终保持着远超同类数据库(如 IBM DB2 Viper 2)的 XML 文档处理速度, 并且能够保证数据在事务处理过程中的原子性、一致性、独立性和持久性要求。此外, MarkLogic 对 XML 文档提供多种形式的索引, 索引包含文档实体、父子关系以及要素取值等内容。由于 MarkLogic 可以在不预先建立文档 Schema 的基础上自动索引 XML 包含的所有要素, 所以 MarkLogic 对文档的管理几乎不需借助 DDL 数据库模式定义。

MarkLogic 已广泛应用在信息服务领域, 用户包含世界领先的科技及医学期刊数据库 Elsevier; 为法律和学术领域提供专业信息服务的 LexisNexis 数据集团; 提供金融财经领域高质量信息及工作流程解决方案的威科集团(Wolters Kluwer)及摩根大通银行; 帮助全球性出版、财经、传媒服务集团 McGraw-Hill 对外提供信息服务, 并辅助其旗下的标准普尔为全球资本市场提供信用评级、指数服务、风险评估和数据服务。此外, MarkLogic 还被美国陆军、美国国防部等不少政府机构采用来建立信息管理业务。

如今, 随着大型跨国企业数据量的逐步增加, 越来越多传统的数据库应用已不能满足企业的需求。在存储和服务器成本的不断降低的趋势下, 不少企业开始寻找在云端存储以 XML 形式存在的媒体文件、文档、网页等信息的有效途径。

MarkLogic 已在帮助拥有海量非结构化数据的大型企业摆脱传统数据库组织数据并建立索引的束缚, 快速搭建数据搜索和查询应用。这样的趋势正快步走向云端, 为那些希望使用云服务来增强非结构化信息索引能力的中小型企业带来新的机遇。

编程语言 2011 年第 05 期——JVM 上的新语言：Ceylon

在 2011 年 4 月举办的 QCon 北京大会上，来自 Red Hat 的 Gavin King 首次公开了关于 Ceylon 语言的重量级话题。作为 Hibernate 的创始人，Gavin 广为 Java 开发者熟知。本次技术大会上透露的 Ceylon 语言，是 Gavin 在 Red Hat 近两年一直投身设计并开发的新一代 JVM 语言，计划在企业开发领域赶超 Java 的领导地位。在 QCon 北京，Gavin 以题为“Ceylon 项目——下一代 Java 语言？”的主题演讲为开发者揭开了 Ceylon 语言的面纱。

Ceylon 面向企业级应用开发设计，在 JVM 运行时基础上执行，支持与 Java 的交互操作。Ceylon 语言充分借鉴了 Java 语言的优势并屏蔽其在设计上的不足，使用静态类型并提供高阶函数支持，在保持代码良好可读性的同时进一步降低开发者入门的难度。

Ceylon 语言没有设计特殊的 Primitive 类型或复合类型，所有类型在类型系统内部都是可表达的。Ceylon 的通用类型设计比 Java 的实现更为简便，而且相比其他类 Java 语言具备更高的安全性，Ceylon 的元编程（Reflection）也必须是类型安全的。此外，Ceylon 语言减少了开发者对于配置文件的依赖，屏蔽了 Java 开发者常常需要对各种 XML 进行的配置和修改操作。

Java 语言的确具备众多适宜于企业开发的优点，包括良好的开放性、适合企业级开发的健壮性等特点。但是其生产效率和过重的负荷却一直令人诟病。作为一门深受 Java 语言影响的 JVM 语言，Ceylon 由熟知 Java 语言的专家设计。正是因为 Gavin 多年的 Java 开发经验，促使他着手实现这样一个期望能够超越 Java 的语言。

当然，从头开始创建一门编程语言以及 SDK 工具包无疑是一项将耗费巨大工作量的任务，但 Gavin 在他的 Blog (<http://in.relation.to/>) 上撰文写道：我们在设计开发 Ceylon 语言的过程中，借鉴了大量 Java 开源社区中的成果，如 Open JDK, JBoss 以及 Eclipse 社区，并且这些工作对于 Red Hat 这样规模并汇聚众多开发好手的公司来说并非是不可能的任务。

社区支持对于 Ceylon 语言的发展是必不可少的，然而，要想获得社区的认可与追随，Ceylon 语言本身能够提供一个设计优良的杀手级开发框架将是个巨大的激励因素，就像 Ruby 语言的 Rails 框架或 Scala 语言的 Lift 框架那样在众多项目中得到成功应用。

此外，IDE 环境对于开发语言的普及具有很大的推动作用，如被不少 Java 开发者所推崇的 Scala 语言，则同时拥有基于 Eclipse 的 IDE 工具、基于 IntelliJ IDEA 的开发环境，以及 NetBeans 的 Scala 开发插件。目前，Ceylon 语言也同样需要一个令人满意的 IDE 工具，来增强语言的适用性。

作为 Hibernate 的创始人，EJB3.0 专家委员会成员，JBoss Seam 的开发者，Gavin King 已经取得了令开发者瞩目的成绩，运行在 JVM 之上的新一代编程语言 Ceylon 能否真正赶超 Java，造就另一个辉煌，让我们拭目以待。

编程语言 2011 年第 04 期——新闻标识语言 NewsML

在 XML 文档格式基础上，专为特定领域设计的各种 XML 方言，在各行业领域中一直有着广泛的应用。如描述站点消息来源的 Atom 格式，业务过程执行语言 BPEL，OGC 组织定义的地理标志语言 GML，Google Earth 和 Google Map 使用的要素标记语言 KML，以及用于数字版权描述的开放数字许可语言 ODRL 等。

在新闻传媒领域，发生在世界各地，由不同语言进行传播的新闻事件、体育赛事、财经数据等海量信息汇总在一起，常使得数据交换和展示的工作变得极为复杂。如何让多种来源的新闻信息在交换时具备良好的格式定义，让不同语言的信息提供者能够得到具有统一结构定义的新闻数据，为了解决这样的问题，新闻标准化机构就制定了面向新闻稿件制定的数据格式标识语言 NewsML，以帮助新闻工作者在收到信息的同时即可获得准确的新闻元数据，并且 NewsML 同样是基于 XML 定义的内容规则描述方言。

NewsML 文档包括用来定义 NewsML 文本逻辑的 Schema 结构，定义 NewsML 文档显示格式的 XSL 样式表，以及符合 Schema 格式定义新闻主体和元数据。其中 Schema 定义了新闻信息标记符的语法描述规则，指定 NewsML 文档包含的元素、元素属性以及元素间的关系；可扩展样式语言 XSL 定义新闻内容的显示格式，能够将新闻内容以多种样式展示。

除对新闻信息内容进行组织描述外，NewsML 提供了强大的元数据描述能力，能够将新闻的每个部分附上元数据来描述其特性。在常见的新闻稿件中，一般包含稿件日期、标题、作者、资料来源、分类类别等稿件标识信息，在 NewsML 标识语言中，不仅包含上述新闻标识信息，还包含题注、关键字、出版者、审改人、签发人、语言、分类、版权、受众对象、重要性、原稿条目、相关超文本链接等多项拓展信息，这些新闻元数据大大提高了媒体间信息交换、新闻管理以及检索的效率。同时，由 NewsML 描述的新闻图片，也将作者、版权、背景描述等元数据加入图片附带的信息中。NewsML 通过对新闻信息和元数据的有效标记，为新闻描述提供了更细的粒度，以方便其准确检索使用和自动化处理。

NewsML 最初由英国路透社设计，希望通过创建一种新的描述格式来包装多媒体新闻资源。目前 NewsML 的管理和维护工作由国际出版电信联盟（IPTC）负责。IPTC 联盟的主要职责是制定和维护各类新闻传媒所需的信息和数据标准，包括新闻交换格式标准、新闻编码、图片元数据等内容。

目前 NewsML 的最新版本为 NewsML-G2 2.7 规范，NewsML 用于通用化的多媒体新闻描述，此外，为了更准确的描述特定类型新闻信息，IPTC 还定制了用于新闻事件描述的 EventsML-G2 标准，以及报道体育赛事的 SportsML-G2 标准，为常用的新闻形式提供细节更为丰富的可定制内容。

编程语言 2011 年第 03 期——JavaScript 代码生成器 CoffeeScript

在 Web 前端开发领域，HTML、CSS 和 JavaScript 的组合以其实用性和通用性而被广泛采用。在越来越多网站开始使用 HTML5 技术来展现 Web 应用的同时，各式各样功能强大的 JavaScript 框架也成为开发者不可或缺的重要工具。JavaScript 框架以其丰富的 API 和透明的跨浏览器支持来帮助开发者优化编码过程。除外，为了改进 JavaScript 语言本身所固有的缺点，已有开发者着手创建一种能够通过编译生成 JavaScript 代码的新语言。

CoffeeScript 就是这样一种使用 Ruby 编写的编程语言，旨在通过简单的编码方式让开发者在 Web 应用中生成并使用 JavaScript 代码。CoffeeScript 融合了 Ruby 的简洁和 JavaScript 的灵活，让开发者通过简单易读的语法来撰写逻辑规则。CoffeeScript 编译器负责将代码逐行解释为等效的 JavaScript 程序，并保证生成的 JavaScript 代码与源代码在逻辑结构上一致。

较之 JavaScript 语法，CoffeeScript 在语法设计上更为严谨。为了让代码看起来更加简洁精致，CoffeeScript 使用空格缩进而非大括号来进行代码段分隔。开发者可以在 CoffeeScript 编程中调用任何已有的 JavaScript 类库，并能够与类库实现无缝集成。经编译生成的 JavaScript 代码，具备良好的版式和可读性，相比较开发者手写的 JavaScript 代码具备同等或更优的执行效率。

CoffeeScript 翻译器是使用 CoffeeScript 语言本身编写，能够在编译的时候格式化 JavaScript 代码，这一点在 CoffeeScript 官方网站上也有所体现。站点为开发者提供了交互式的 CoffeeScript 在线编译环境，开发者在一侧敲入 CoffeeScript 代码的同时，代码被实时解释为对应的 JavaScript 程序，并可随时在浏览器上运行验证。

CoffeeScript 自 2009 年 12 月份诞生以来，经历了一年多开发完善，前不久刚发布了 1.0 版本。在新版本中，CoffeeScript 改进了循环体结构，并提供了简便的闭包实现。为了方便 Java 开发者使用，CoffeeScript 还提供有一个被称作 JcoffeeScript 的 Java 类库，让 CoffeeScript 代码能够在 Rhino 环境中编译。Rhino 项目由 Mozilla 基金会赞助，完全使用 Java 语言开发了 JavaScript 引擎，以便嵌入在 Java 应用中为用户提供脚本支持。

目前已有不少 Web 开发项目开始使用 CoffeeScript 生成的 JavaScript 来构建站点。比如提供各种新技术资讯的 arstechnica.com 站点，在其推出的 iPad 应用中，就是使用 HTML 和 CoffeeScript 为读者提供前端展示。37Signals 发布的 iPad 白板应用 Chalk 也是使用 CoffeeScript 编写，并且能够借助 Cache Manifest 实现应用的离线运行。运行在浏览器端的坦克大战游戏 Orona 同样借助 CoffeeScript 实现。此外，站点 thelincolnshirepoacher.com 还使用 CoffeeScript 和 JavaScript 的矢量图形库工具 Raphaël 来演示如何随即生成各式各样有趣的图案。

编程语言 2011 年第 02 期——跨越 JVM 与 .NET CLR 的 Fantom

在 JVM 平台并入 Oracle 产品线后, Oracle 将对 JVM 平台进行多项改进和调整。内容包括增强平台的模块化和集成性等特征、为 Java SE 增加多核处理支持, 在 Java ME 中增加多点触摸支持等。与此同时, Oracle 开发部副总裁 Adam Messinger 在 Qcon 开发大会上透露, Oracle 计划在提供免费版 JVM 基础上新增收费版本, 在收费版 JVM 里整合 HotSpot JVM 与 JRockit JVM, 为高端用户提供功能更为强大的解决方案。

对开发者来说, 无论 Oracle 策略如何, JVM 平台依然有着巨大的吸引力。依托于 JVM 平台的动态语言, 始终保持着蓬勃的生命力。在这些动态语言中, 能够同时兼容 Java 虚拟机和 .NET CLR 通用语言运行时的 Fantom, 是其中极具特色的动态语言。

Fantom 最初被称为 Fan 语言, 由于定名为 Fan 不易检索, 开发者在 09 年 11 月份将当时新发布的版本定名为 Fantom 语言。Fantom 使用与 BSD 类似的 Academic Free License 3.0 开源协议授权。

Fantom 的开发者将其设计为一门实用且有趣的脚本语言, 来解决 Java 和 C# 编程中实际存在的问题, 降低开发者编码的难度。Fantom 语言最大的特点是它的可移植性, Fantom 代码能够无缝的在 Java VM 和 .NET CLR 平台之间进行迁移。为了保证平台间的顺利移植, Fantom 代码首先被编译为 fcode, 即一种字节码表示法, 之后转换成为 Java 字节码或 IL 解释语言, 转换的过程在程序运行时完成, 开发者可以以单个文件的形式在虚拟机中部署 Fantom 代码模块。

对于 Java VM 和 .NET CLR 平台切换时的 API 调用, Fantom 给出了简便易用的解决办法。Fantom 提供了一组与 Java 和 .NET API 配合使用的 API 接口供开发者使用, 来屏蔽不同平台间对于系统 API 调用的差异。

当然, Fantom 语言的可移植特性不仅限于 Java 和 .NET 平台, Fantom 支持将代码编译为 JavaScript 脚本供开发者在浏览器端使用。Fantom 语言从一开始就为跨运行时的编程而设计, 在未来的规划中, Fantom 语言还将支持 iPhone 的 Objective-C 语言, 以及 LLVM 集成编译环境或集合了 Python 与 Perl 语言优势的 Parrot 语言。

在 Fantom 编程支持方面, 开发者可以使用基于 NetBeans Platform 的 IDE 环境 FantomIDE 进行编码和调试, IDE 包含开箱即用的 Fantom 环境, 语法和词法解析使用开源语法分析器 ANTLR 设计, FantomIDE 还提供了 Javascript、CSS、HTML、XML 等脚本支持以及 Subversion 和 Mercurial 等管理工具。

JVM 平台从不缺乏优秀的动态语言, Groovy、JRuby、Jython、Scala 都是其中的佼佼者, 不同语言的风格类型、运行速度和工具支持也各有千秋。Fantom 以其跨运行时的特性和先进的设计理念, 是否将来能脱颖而出得到开发者的认可, 让我们拭目以待。

编程语言 2011 年第 01 期——DOI: 国际化的数字对象标识符

ISO 近期以 100% 的赞成票投票通过了一项新的国际标准“ISO 26324: 数字对象唯一标识符(Digital Object Unique Identifier)”, 简称 DOI 标识。DOI 由统一资源定位符 URL 演变而来, 为网络环境下的信息资源实体确定唯一的标识规则。

ISO 国际标准化组织是成立于 1947 年 2 月的全球性非政府组织, 总部设在瑞士日内瓦。ISO 标准的制修订工作由各技术委员会负责, ISO/TC46 第 46 技术委员会是信息与文献技术委员会, 负责制定和推广与信息 and 文献工作相关的国际标准, DOI 标准由 ISO/TC46 的分技术委员会 SC9 (标识与描述) 归口管理。除 DOI 标准外, 技术委员会还负责国际标准书号 ISBN、国际连续出版物号 ISSN 和国际标准音像号 ISAN 等标准的制修订工作。

在信息化时代, 数据和信息本身的价值往往比作为载体的管理信息系统更重要。在 DOI 出现前, 大量有价值的数据因知识产权得不到保护而不愿共享。此外, 很多具有研究和使用的数据由于缺乏统一标识规范, 常会造成无序的引用和歧义。针对上述问题, 数字对象唯一标识 DOI 通过构建数据资源标识方法, 为数据提供注册、解析和查询途径, 在推动数据资源共享的同时为数据的网络传播提供版权信息。

DOI 标识编码由前缀和后缀两部分组成, 中间用“/”分割。为满足可扩展的数据资源编码需求, DOI 没有限制前缀和后缀的字符长度。DOI 前缀由目录代码和登记机构代码两部分组成, 所有 DOI 代码目录定义为“10.”, 即 DOI 代码均以“10.”开头。登记机构代码通过向国际 DOI 基金会(IDF)申请获得。登记机构代码的分配根据信息资源编码需求制定, 如出版商可为其所有数字出版物申请统一前缀, 也可以为数字图书、音像制品等各类出版物申请不同前缀。DOI 后缀是在特定前缀下的唯一标识, 由登记机构分配并确保其唯一性。后缀可以是任何字母或数字编码, 如机器码、ISBN 或 ISSN 等已有规范码, 具体编码方案由登记机构设定。根据编码规则, DOI 数据资源命名规范通常定义为: “统一前缀/类型标识. 机构名. 内部标识码. 扩展码”的形式。

目前国外 Elsevier、Blackwell、Springer、John Wiley 等大型数字资源提供商已开始使用 DOI 对其出版的电子文献和数字内容进行标引, 帮助基础性数据资源的快捷获取和灵活引用, 同时为了便于数字资源的共享与交换, DOI 登记机构还会在唯一编码基础上提供数据资源的命名、注册、解析、查询和变更等管理功能。DOI 能够实现资源实体的永久性标识, 便于数字资源的归档与检索。目前专业的 DOI 检索站点 CrossRef 已与 Google 搜索技术结合, 在 DOI 检索过程中提供数字文档的全文检索, 为用户建立检索结果到数字文献全文间的获取通道。

由于 DOI 标识的唯一性、持久性、兼容性、互操作性、动态更新等特点, 在正式成为 ISO 标准后, 必将会在信息资源标识引用过程中起到越来越重要的基础性作用。

编程语言 2010 年第 12 期——LEGO 机器人控制语言：NXT-G

在十一月份的 TIOBE 语言排名中，一门特殊的编程语言 NXT-G 首次跻身跃入排行榜的前 20，位列第 17 名。NXT-G 编程语言的特殊之处在于，这并不是提供给专业程序开发者的编程语言，而是由丹麦著名玩具制造商乐高（LEGO）公司为 NXT 机器人玩具设计的可视化编程语言。

头脑风暴（Mindstorms）是乐高出品的可编程机器人系列玩具，NXT 2.0 是目前乐高机器人系列中的主打产品。第一代 Mindstorms 机器人于 1998 年问世，被定名为 RIS（Robotics Invention System），其中搭载了由日本瑞萨电子提供的 8 位微控制器，由专用红外接口向处理器中载入设计好的程序来控制机器人的行为。时隔 8 年后，乐高于 2006 年发布了新一代的机器人 Mindstorms NXT，并在 2009 推出了功能升级后的 NXT 2.0，以满足玩家在组装机器人时对于高级功能如颜色传感、蓝牙设备的需求。

NXT 的核心是一块被称为智能积木的处理器，能够通过 RJ12 规格的 4 芯水晶头线缆接受四路传感信号，并能够控制三个马达协同运转。此外，NXT 智能积木搭载了一块 100x64 像素的单色 LCD 显示器，可以在机器人运行中通过内置的层级菜单接受用户的按键命令或组织简单的处理步骤。

NXT-G 是对应智能积木处理器的可视化编程控制语言，能够帮助机器人实现简单或复杂的控制流程，如马达驱动、传感器输入合并、完成计算任务、实现流程控制等。完成后的程序通过 USB 或蓝牙传输将程序和音频文件装载到 NXT 模块中，以赋予 LEGO 机器人生命，NXT 还支持模块间文件拷贝和通过智能手机的蓝牙连接进行实时控制。

NXT-G 的交互式编程环境基于国家仪器（National Instruments）的 LabVIEW 软件提供，而 LabVIEW 的最大特色就是以图形化编程 G 语言来组织程序，编写程序的展现形式为具备特定逻辑结构的框图。各模块功能实现则由 LabVIEW 中能完成各项编程任务的庞大函数库提供，包括数据采集、数据显示、数据存储、GPIB 通用接口总线、串口控制、数据分析、蓝牙通讯等。此外，LabVIEW 可视化编程环境提供了与传统 IDE 环境相同的程序调试工具，如断点设置、以动画方式显示数据及其子程序的结果、单步执行等功能。

NXT-G 编程环境对 Windows XP、Windows 7 和 Mac OS X 都提供了良好的支持。除了 NI LabVIEW 软件外，还有不少支持 NXT-G 编程的软件程序。包含 NXT-G 零售版和 NXT-G 教育版、LEGO 基于 LabVIEW 简化定制的 RoboLab 工具，乐高机器人专用的汇编指令 NBC，以及基于 NBC 编译器实现且语法与 C 语言相似的高级语言 NXC 等。

目前乐高公司已将 NXT 智能积木的固件设计开源，同时 NBC 和 NXC 编译器也是开源项目。NXT-G 在 TIOBE 语言排名中的提升，从一个侧面反映了可视化编程语言的流行趋势。如果你是喜欢自己动手又对自动控制感兴趣的程序开发者，不妨尝试一下乐高 NXT 积木和 NXT-G 编程环境，LEGO 一定会为你带来丰富的体验和乐趣。

编程语言 2010 年第 11 期——超越 GCC 的编译环境：LLVM

C、Objective-C 和 C++ 的编译工具集 LLVM 在本月发布了最新的 LLVM 2.8 版本。在 2.8 版本中，LLVM 包含了 C++ 标准库实现 LibC++，以增强 LLVM 对于 C++ 规范的支持。新的 LibC++ 库在 C++ 标准符合性以及高效代码生成方面有着很强的优势。同时，在新版本的 LLVM 中，包含了首次正式发布的新一代高性能调试器 LLDB，LLDB 应用与 LLVM 编译器相同的解析器和源代码工具，以替代先前的 GDB 调试器。

LLVM 编译环境是一个集合项目，为面向 C 或基于 C 的语言提供新一代高效易用的虚拟机环境。LLVM 起源于 2000 年伊利诺伊大学发起的开源项目，起先主要在学术研究领域中有所应用。在 2005 年，Apple 为 LLVM 提供了支持以帮助其发展，并保持 LLVM 与 Xcode 良好的兼容性，使其成为一个开发者易于使用的编译器。相比较传统的 GCC，LLVM 的设计更为现代化和模块化。LLVM 为开发者提供了中间代码和编译基础设施，并且 LLVM 能够与 IDE 紧密地交互和集成，作为 IDE 的底层支撑基础，为代码补全、重构等功能提供辅助，同时在程序编译、连接、运行环境执行过程中实现全新的编译优化策略。值得一提的是，LLVM 还提供了很多编译器以外的辅助工具，如对代码进行静态检查并生成 HTML 格式的分析报告等。

在 LLVM 2.7 版本之前，LLVM 支持 C 和 Objective-C 编译，在 LLVM 2.7 版本发布时引入了对于 C++ 的支持。LLVM 的编译器由 Clang 完成，Clang 在编译速度方面非常高效，体现在包括预处理、语法解析、语义分析、抽象语法树生成的各个编译步骤之中。同时 Clang 在内存占用方面的开销非常经济，较之 GCC 有大幅度的缩减。与 GCC 相比，Clang 的设计清晰简单、易于理解，具备良好的扩展性，同时能够保持与 GCC 的兼容性。Clang 为程序诊断信息提供了非常好的版式结构，为程序语法错误提供了源码提示，以及上下文调用的错误提示。

在程序编译的过程中，GCC 完成了从预处理到代码生成的整个过程，其封装式的设计使得很多中间信息无法被其他程序重用。而 Clang 将编译过程分隔成模块化的阶段，以大幅度增强开发者对于代码的操控能力。Clang 的设计目标之一就是要超越 GCC 编译器，在 Clang 官方网站 (<http://clang.llvm.org/>) 的性能比较栏目中，列举了 Clang 与 GCC、Elsa 以及 Portable C 的各项比较优势，感兴趣的开发者可以在官网详细查看。

来自伊利诺伊大学香槟分校的 Vikram S. Adve，同时也是 LLVM 开发者 Chris Lattner 的导师，在代码生成和优化大会 CGO 2009 上的主题演讲《下一代编译器设想》中谈到：下一代编译器、优化器和运行时的基础架构将更注重在静态语言中应用实时编译和动态优化技术，并且编译器将借助更为灵巧的自动调优策略来挖掘程序优化潜能，同时编译器将会采用投机优化的方式来弥补静态分析优化的不足之处。LLVM 项目已经在为下一代编译器的实现方向做出努力，我们没有理由不去尝试使用这个致力于超越 GCC 的新一代编译环境，来为我们的程序编译注入新的动力。

编程语言 2010 年第 10 期——达尔文信息分类架构 DITA

达尔文信息分类架构 DITA (Darwin Information Typing Architecture) 是面向主题的技术文档交付解决方案, 用于定义、编写和交付内容信息规则。DITA 基于 XML 表示并继承了 XML 的文档描述元素, 覆盖内容信息组织、编写、生成和交付的整个出版过程, 并可以面向不同领域的技术出版物进行扩展和定制。

DITA (<http://dita.xml.org/>) 最初由 IBM 开发, 目前交由结构化信息标准组织 OASIS 维护, DITA 通过将文档组织为可重用的主题模块, 并将模块进行组织来表示信息的内容, DITA 提供了面向主题的元数据注释, 用户可以便捷的搜索、过滤和处理内容。

DITA 侧重于交付主题粒度的知识单元, 通过将知识单元进行重组和映射, 生成最终交付的出版文档。在主题粒度划分方面, 作为片段信息载体, 首先应具备自包含特性, 即主题具有信息描述的完备性, 能够准确完整的表达内容含义, 所以划分的粒度不易过细。同时, 作为建立后续关联映射的基础单元, 主题的粒度不易过粗, 应便于通过 DITA 映射进行知识共享和重用。

DITA 的设计理念, 与传统着眼于书籍或文档交付的排版技术如 DocBook 或 LaTeX 等在思路上有很大区别。DITA 和 DocBook 通过定义规范化的文档描述规则, 来解决文档交付过程中遇到的问题。面向不同类型的交付出版物, DITA 和 DocBook 各有见长, 但在实际应用中也有自身的限制因素。DITA 解决了出版物的结构化描述和内容重组问题, 支持多语言版本制作, 适用于对格式有严格限定的技术手册类出版物。但 DITA 不能实现很完美的样式渲染, 且对于内容与格式一体化的复杂出版物, DITA 很难进行主题和界定与划分。所以使用 DITA 进行书籍出版的成本和难度较高。

相比较来说, DocBook 适用于通用出版物, 文档易于组织和排版。但 DocBook 内容以 Section 段落组织, 不具备 DITA 的内容映射机制, 无法做到类似 Topic 这样粒度的内容划分与重组。且对于内容需要频繁修改的文档排版, DocBook 略显力不从心。LaTeX 是富格式文本集, 适用于科技论文作者进行个人创作, LaTeX 文档内容和标签的耦合度非常紧密, 并不适合出版社的排版流程化作业, 以及文档内容的析取和重组, 和多种交付文档的组合生成。

DITA 和 DocBook 都用于交付技术信息, DITA 侧重于交付主题, 而 DocBook 侧重于交付整本书籍。在最终交付物的输出格式方面, DITA 能够生成 PDF、CHM、HTML 等大部分出版交付类型。DocBook 常用的交付格式为 PDF 和 HTML, 其他输出格式需要借助相关的功能插件。

DITA 开放工具箱 (<http://dita-ot.sourceforge.net/>) 是 OASIS 为 DITA 提供的 Java 实现, 能够帮助文档作者将 DITA 的映射和主题转换为最终交付文档。DITA 开放工具箱包含 Ant 构建模板、DITA 词汇表 DTD 定义, DITA 词汇表的 XML Schema 定义以及 Java 文档处理类库等内容。对于 DITA 文档, 源文件和映射文件中的标记项由 DTD 和 Schema 定义, 借助 XSLT 和 CSS 文档, 经处理渲染后得到最终交付文档。

Syntaxt 公司开发的开源的 XML 编辑器 Serna Free 支持 DITA、Docbook、XHTML 等 XML 文件类型, 能够实现可视化的编辑工作。在商业产品方面, 提供软件产品生命周期管理的 PTC 公司开发的 XML 文档处理器 Arbortext, 同样提供 DITA 的编辑与转换支持。

编程语言 2010 年第 09 期——并发编程好手：Clojure 语言

近两个月的 TIOBE 语言排行榜中，在 Lisp/Scheme 语言的栏目项后又出现了一门新的语言——Clojure。诞生于 2007 年的 Clojure 是一门相当年轻的编程语言，Clojure 的发起人 Rich Hickey 也曾是 Lisp 开发者，Lisp 以强大的功能和表达性而著称，但应用范围一直存在着固定的局限，于是 Clojure 的设计初衷便是希望得到一门能够服务于高并发应用场景，较 Lisp 更为先进的函数式编程语言。

Clojure 保持了函数式语言的主要特点，同时避免了不稳定状态、递归和高阶函数。Clojure 在语法设计上非常简单，但对非 Lisp 程序员来说需要一段时间来适应 Clojure 独特的编程思路。Clojure 的语法设计优势在于，让开发者绝大部分工作专注在真正解决问题上，而非耗费在如何模块化设计和如何进行解耦层面上。

与 Lisp 语言相似，Clojure 也是基于 S-表达式（sexp）设计。S-表达式给出了表达半结构化数据的约定，以其在 Lisp 语言中的广泛应用而为人熟知。S-表达式在 Lisp 中既用作代码，也用作数据，其最通用的特性是使用 S-表达式作为括号化的前缀表示。

与 JRuby、Jython、Scala 等依托于 JVM 运行的语言相同，Clojure 是运行在 Java 虚拟机平台上的 Lisp 方言，能够直接调用 Java 类库，借助 JVM 平台广泛的适用性，为 Clojure 语言的应用拓展了空间。此外，Clojure 可以说是社区智慧的结晶，除了 Rich Hickey 等核心开发者外，Clojure 语言的补丁、文档、测试和不少新功能的开发思路都是来自于开源社区参与者的贡献。

Clojure 之所以受到广泛关注，最主要的原因来自于其并发特点，如支持软件事务存储（Software Transactional Memory, STM）等特性。在实际应用方面，站点 FlightCaster 为用户提供了航班延误的预告信息，同时也为 iPhone 和 Blackberry 提供了终端应用。FlightCaster 的 Web 前台用 Rails 实现，部署在 Heroku 上。后台数据分析和处理程序则是由 Clojure 实现，同时在后台使用了 Hadoop、Cascading、Cloudera 等工具。在 FlightCaster 站点中，数据预处理并转换成适当视图用以分析的操作，以及所有统计推断和机器学习代码都使用 Clojure 实现。FlightCaster 站点将 Clojure 数据结构用作通信和存储的中间表示，Clojure 代码分析得到的航班预告模型，以 json 中间格式来表示，然后推送到 Rails 前端，以 json 格式读取数据并展示给用户。

对于 Clojure 开发者来说，开发工具首选是使用 NetBeans IDE 的 Enclojure 插件（<http://www.enclojure.org/>），Enclojure 提供了支撑 Clojure 开发的一组类库，包含独立的 REPL（Read, Evaluate, Print, Loop）支持，同时支持 REPL 和远程 JVM 建立连接。Enclojure 提供的编辑器支持具备语法高亮显示、函数声明定位，S-表达式导航，Lisp 风格缩排，代码结构化导航，内置项目模板，断点设置和单步调试，并能够对代码错误给出超链接提示。

此外 Eclipse 也提供了专门的 Clojure 语言开发插件 Counterclockwise（<http://code.google.com/p/counterclockwise/>），在源代码编辑，代码调试，REPL 支持方面也有独到之处，适合于习惯于 Eclipse 的开发者使用。

编程语言 2010 年第 08 期——Go 语言发展近况

Go 语言是 Google 公司的实验性语言，以 BSD 协议开放源代码。Go 语言在 Inferno 操作系统相关的工作基础之上开展，最初由 Rob Pike 等 Google 工程师在 2007 年 9 月启动设计工作，在 2009 年 11 月 Google 正式对外发布。

开源的 Go 语言特性上与 C 语言类似，在通用的语法结构之外，Go 语言具备类似 C++ 语言的引用参数，还提供类似 Java 语言的对象、实例和数组类型，同时支持反射、垃圾回收以及动态类型。在面向的应用领域上，Go 语言最初定位于网络服务器、存储系统和数据库的程序设计，同时在语言中包含并发构造体，以方便帮助开发者创建并行任务。

Go 语言编写的代码在运行前需经过编译，但 Go 语言将动态语言易于编写的特性和静态语言的高效性相结合，具备良好的易用性和极佳的执行效率。在安全性方面，Go 语言仅允许显式的类型转换，不允许隐式转换，以保证类型安全。Go 语言提供了指针类型，但不提供指针算术运算，以支持保存在栈上的对象实例，同时 Go 语言具备高效的垃圾回收机制，以保证内存安全。

Go 语言具有效率极高的编译器，目前 Go 语言编译器能在 Linux、FreeBSD 以及 Mac OS X 平台下运行，但暂时没有对 Windows 环境提供编译器。Go 语言开发团队曾在博客中表示，编译完成的 Go 程序运行速度接近 C 语言实现的程序。

依据惯例在 TIOBE 每年一月份发布的年度编程语言榜单中，入选 2009 年度编程语言的并非先前呼声最高的 C#，而是在评选前刚发布不久的 GO 语言，Go 语言在年初第一次进入 TIOBE 排行榜就拿到十名左右的位置，Google 对程序开发者的影响力可见一斑。尽管由于应用领域的限制，在今年六月份的 TIOBE 语言排行榜中，Go 语言已无法在前二十名编程语言中站稳脚跟，但 Google 公司却一直不断为 Go 语言添砖加瓦。

在今年五月份举办的 Google I/O 大会上，Go 语言背后的工程师 Rob Pike 和 Russ Cox 两人一同为开发者带来了 Go 语言编程开发的讲座，解释 Go 语言的灵活设计和独到之处。Go 语言的主要设计者 Rob Pike 是 Google 公司的资深工程师，在分布式系统、程序语言设计以及软件开发工具方面有着多年的经验。此前在贝尔实验室工作期间，Rob 曾是网络操作系统 Plan 9 的设计者，同时也是 Inferno 操作系统的设计者，Rob 还曾与 AWK 和 AMPL 编程语言的设计者 Brian Kernighan 一起合著有两本 Unix 环境下编程语言的书籍。

针对 Google I/O 大会上收集到开发者的提问，Go 开发团队通过博客 (<http://blog.golang.org/>) 给出了常见问题的答案。在 Android 平台支持方面，Go 编译器支持面向 ARM 处理器的代码生成，所以在 Android 平台上运行 Go 程序是可行的，Go 语言开发团队也将积极推进 Android 平台上的开发工作。由于 Go 具备简洁的语法和语言本身的一致性，已有在大学执教的老师表示将会在自己教授的编程语言课程中为学生开设 Go 语言选修项目。

目前在 Google 公司内部已有部分应用使用 Go 语言来实现，同时开发团队也表示，Go 相对已经比较成熟，完全可被用于最终系统开发和交付。在后续的工作中，Go 开发团队将不断对语言进行改进和提升，并随时欢迎开发者在社区中反馈 Go 语言开发时遇到的问题。目前在国内使用 Go 语言进行实际开发的公司还为数很少，尝试使用 Go 语言的开发者大多源自个人兴趣或技术预研，但随着 Go 语言类库和成功案例的增多，相信国内将会有越来越多的开发者将 Go 应用在实际开发生产中。

编程语言 2010 年第 07 期——RailsConf 2010 大会概览

本月在美国马里兰州的海滨城市巴尔的摩举办的 RailsConf 2010 开发者大会，吸引了所有 Rails 开发者们的目光。由 O'Reilly 组织的 RailsConf 大会，汇聚了 Rails 开发社区中年度最热门的内容，包含主题讲座、动手教程、专题讨论等 Rails 的方方面面，同时也吸引了 Engine Yard、New Relic、Heroku 等 Rails 主流开发厂商的赞助和参与。

在为期四天的 RailsConf 2010 大会上，主题列表中包含的内容有 OAuth 和 OpenID、NoSQL 与 Rails 的结合应用、Rails 在私有云上的应用、C++实现的 Rubinius 虚拟机、Rails 3 与 JRuby 的结合、插件和 Gems 向 Rails 3 的迁移等众多实用和有趣的主题。

Rails 3 在 RailsConf 会议期间发布了第 4 个 Beta 版本，这也成为 RailsConf 大会上众多开发者关注的亮点之一。在 RailsConf 会议开场时 DHH 的主题演讲中，DHH 谈到了他最看中的一些 Rails 特性，包括 gem 配置和管理的简化、改进后优雅的 Active Record 查询语法、更为简单自然的 Router 语法以及 ActionMailer 的改进等内容。

随着最后一个 Beta 版本的发布，Rails 3 的 RC 发布候选版也将在会议结束后到来，虽然 Rails 3 让开发者们经历了漫长的等待，但根据官方的说法，在 Rails 3 即将到来之时是进行应用升级的最好时机。开发者仅需执行“gem install rails --pre”命令即可获得最新的 Beta 版本，同时 Rails 开发团队将保证发布版本的可靠和稳定。

RailsConf 2010 的一个有趣的环节是，在大会晚间活动中，举行了 Ruby Heros 颁奖仪式。Ruby Heros 奖项颁发给每年对 Ruby 和 Rails 社区贡献最大的开发者。参与开源项目、制作技术文档、开发插件和 Rails Gems、积极组织社区活动，一直默默为社区做出贡献而不收取回报的开发者，将会获得这项社区的荣誉奖项。

本年度 Ruby Heros 的六位获奖者分别是 Plataforma 公司的首席开发者和 Rails Core 的核心成员 José Valim(@josevalim)，Thoughtbot 公司的 Web 开发者 Nick Quaranto (@qrush)，多个开源项目贡献者和 Rails 开发者 Xavier Noria (@fxn)，西雅图华盛顿大学的 Ruby 课程教员 Aaron Patterson (@tenderlove)，AMS 公司的系统架构师和 Rails 开发者 Wayne E. Seguin (@wayneeseguin)，以及《Ruby Best Practices》一书的作者 Gregory Brown (@seacreature)。历年 Ruby Heros 奖项的获奖者参见站点 <http://rubyheroes.com/> 的汇总介绍。

在本次大会上，来自国内的 Rails 开发者 Guo Lei 与其所在 Idapted 公司的创始人 Jonathan Palley 一起做了 Rails 系统重构方面的主题讲座。讲座内容来自于 eqenglish.com 站点随着业务增长，两年来使用 Rails 对网站代码进行重构的经验。感兴趣的读者可以在 Idapted 公司博客 (<http://developer.idapted.com>) 中浏览 Guo Lei 记录的 RailsConf 2010 大会整个行程。

随着国内越来越多 Rails 站点的兴起和成功，锻炼了一批富有经验的国内 Rails 开发者。在国际化的舞台上，来自国内开发者社区的声音将越来越显著，与此同时，国际化参与和交流的增多也将不断推动国内 Rails 社区的发展和壮大。在本月底，第二届 Ruby 技术大会 (RubyConf China) 将在上海举办，Ruby 之父 Matz 和众多国内开发者也将将在上海聚集一堂，让我们共同期待 Ruby 和 Rails 的开发社区在国内不断发展壮大。

编程语言 2010 年第 06 期——了解 REBOL 语言

在今年 QCon 北京大会上，来自支付宝的架构师蔡学镛为参会者分享了 REBOL 编程语言相关的主题讲座。对大多数开发者来说，在实际开发项目中选用 REBOL 语言目前并不算多，但 REBOL 语言却有着与生俱来的优势，能够帮助程序员得心应手地完成特定应用场景的开发工作。

REBOL (<http://www.rebol.org/>) 是 Lisp 语言的替代品，由业界知名的操作系统和编程语言架构师 Carl Sassenrath 设计，是为分布式计算和通信量身打造的轻量级开发语言。REBOL 完全能够胜任 Lisp 语言的应用场景，同时 REBOL 又打破了 Lisp 的局限。REBOL 更类似于一般编程语言的表达方法，允许使用中缀运算符，且不需使用 Lisp 语言那样一对对嵌套的小括号来表示递归。REBOL 语言具有统一的库函数，如 REBOL/Core、REBOL/View、REBOL/Command、REBOL/SDK、REBOL/IOS 等库函数，在图形界面、编程辅助、网络处理等方面提供丰富的支持。

REBOL 的图形界面设计能力十分强大，通过简单的图形界面编程语句就可以完成图形界面的创建，并且图形界面的展示美观漂亮。同时 REBOL 内置了 View Draw Dialect 方言，帮助开发者以最简便的方式完成图形界面中元素的定制，同时可以根据开发者的需求进行方言的拓展，定义开发者需要的子语言。作为网络编程语言，REBOL 在网络处理方面具备很强的能力，对于文件下载、网页抓取、邮件接收等操作，在 REBOL 中仅需几条语句即可完成操作。

对比 REBOL 语言，在动态语言中应用非常广泛的 Python 语言，具有完整的面向对象特性，语言简便易用，编写的程序非常优雅。从加快开发速度的角度考虑，能够代替 C 和 Java 语言完成工作任务，同时与 C 语言的沟通能力非常好，任何为 C 编写的接口能够很容易转化为 Python 接口。而 REBOL 是基于相关表达式的对象语言，REBOL 被称作最接近人类语言的编程语言，有着自身的独特之处，能够方便的表示数据和元数据。REBOL 的设计理念之一就是始终保持小巧，即便 REBOL 已内置了几十种数据类型和数百条函数，以及多种 Internet 协议、错误处理等内容，REBOL 解释器仅有几百 K 大小，REBOL 程序编写方便，在代码编写时程序不需要对库和包含做声明即可使用，帮助开发者以最精炼的语句实现应用。同时 REBOL 代码可以跨越几十种系统平台，在不同平台之间移植代码不需做任何改动。

但需要强调的是，REBOL 是商业公司产品，可以提供给开发者免费使用，但源代码并不开放，使用部分 REBOL 函数库如数据库连接库，需要支付额外的费用。但从另一个角度看，由公司维护的语言能够在最大程度上保证类库的统一，减少开发者额外的负担。

目前 REBOL 语言的解释器最新版本为 R3 Alpha，在 REBOL 语言中文参考文档方面，蔡学镛和蔡民奎先生已经在网络整理了 REBOL 语言的中文参考文档 (<http://sites.google.com/site/cnrebol/>)，其中包含 REBOL 3.0 指南，REBOL 内建函数，REBOL 数据类型等不同的内容，感兴趣的开发者甚至可以加入到其中，一同更新维护 REBOL 语言的中文资料。

编程语言 2010 年第 05 期——动态语言实现 Map-Reduce 应用

并行计算技术的发展和网络带宽的提升,让云存储和云计算一步步走进实际业务应用之中。对于动态语言开发者,如果希望用动态语言借助 Hadoop 平台实现分治规约应用,并不是一件复杂的事情。

Hadoop 框架使用 Java 语言开发,但业务处理不要求必须使用 Java 语言,而可以使用 Python 或 C++等其他语言进行编码。在 Hadoop 安装路径下的/src/examples/文件夹中,给出了 Python 代码实现的分布式应用示例。但对于 Python 程序员,编写应用不必使用 Hadoop 官方示例的模式通过 Jython 运行时转换为 Java 的 jar 文件来部署,而可以借助 Hadoop 发布版中的 Hadoop Streaming 工具,利用可执行程序或脚本来实现 Map-Reduce 过程中的 Mapper 或 Reducer 过程,而不需编写 Java 的实现类。

如何以 Pythonic 的方式完成 Hadoop 经典的 WordCount 词频统计工作,来自德国的开发者 Michael G.Noll 在博客中撰文讲解了实现方法:在 Python 编码实现过程中,通过 STDIN 标准输入和 STDOUT 标准输出在 Map 和 Reduce 代码之间传输数据。Python 的 sys.stdin 方法用于读取输入数据,并通过 sys.stdout 打印输出数据,而数据的交互处理过程交由 Hadoop Streaming 来处理。以同样的方式,可借助 Perl、Ruby 或其他动态语言完成 Map-Reduce 处理过程。

无独有偶,著名音乐站点 Last.fm 也发布了基于 Python 的 Dumbo(小飞象)项目,Dumbo (<http://github.com/klbostee/dumbo>)能够帮助 Python 开发者更方便的编写 Hadoop 应用,并且 Dumbo 为 MapReduce 应用提供了灵活易用的 Python API。Last.fm 的开发者,同时也是 Dumbo 项目发起人 Klaas Bosteels 认为,对于定制 Hadoop 应用,使用 Python 语言代替 Java 会让工作变得更有效率。

在基于 Dumbo 框架的应用实现中,不需模仿 Michael G.Noll 给出的代码将 mapper 和 reducer 存放于不同的文件中,而只需在程序中定义 mapper 和 reducer 函数,并通过语句 `dumbo.run(mapper, reducer)`来启动执行过程。在作者给出的 Dumbo 示例中,能够将 Hadoop 原先 60 行 Java 源代码实现的 WordCount 示例,简化为 Python 实现的 7 行源代码来完成。此外, Dumbo 还提供了一些易用的功能,包括向 `dumbo.run()` 执行操作传递第三方参数。并在 Dumbo 项目文档中,给出了从 Apache 访问日志文件中进行 IP 统计的完整示例,以及四个简短的应用示例。

另外一个相似的项目 Happy (<http://code.google.com/p/happy/>),则为 Jython 开发者使用 Hadoop 框架提供了便利,Happy 框架封装了 Hadoop 的复杂调用过程,让 Map-Reduce 开发变得更为容易。Happy 中的 Map-Reduce 作业过程在子类 `happy.HappyJob` 中定义,当用户创建类实例后,设置作业任务的输入输出参数,然后调用 `run()`方法即可启动分治规约处理,此时,Happy 框架将序列化用户的作业实例,并将任务及相应依赖库拷贝到 Hadoop 集群执行。目前,Happy 框架已被数据集成站点 `freebase.com` 采纳,用于进行站点的数据挖掘与分析工作。

编程语言 2010 年第 04 期——Rails 3.0 新版本预览

Rails 3.0 的 Beta 版本已经发布, 由于 Merb 框架与 Rails 3.0 的融合, 新版本集成了 Merb 框架的最佳特性, 改善了 Rails 框架中晦涩难懂的部分, 让 Rails 拥有轻量级的内部实现, 为开发者提供更为简洁易用的 API。

对于 Rails 3.0 的改进, 负责新版本设计的总架构师 Yehuda Katz (wycats) 曾谈到, Rails 3.0 需要完成的目标之一是将 Merb 模块化与定义良好的内部 API 引入到 Rails 代码基中。此外, 通过内核重构进一步改进 Rails 框架内核耦合紧密的现状, 让新版的 Rails 遵循模块化设计, 允许开发者使用不同的 ORM 及 NoSQL 储存系统构建自己的数据支撑环境, 或方便的使用 RSpec 及其他测试工具来测试 Rails 应用程序。

在 Rails 3.0 的新增特性中, 引入了全新的 Router, 更新了 API 并着重 RESTful 风格的声明。效仿 Action Controller 为 Rails 增添了全新的 Action Mailer API。此外, 基于关系代数构建了全新的 Active Record 链式查询语言, 以及使用 Bundler 进行显式的依赖管理。在 JavaScript 支持方面, 使用 Unobtrusive 助手与 Prototype 或 jQuery 以及其他 Ajax 框架进行集成。

在安全性方面, 新版 Rails 在发布前对已知现存的安全缺陷进行了修补, 同时开发团队仔细检查了 Rails 的内部实现, 以避免跨站脚本缺陷 (XSS) 等安全问题的出现。

在 Rails 3.0 Beta 版本之后, Rails 3.0 最终版本的正式发布也指日可待。以 DHH 为首的 Rails 核心开发团队, 为迎接 Rails 3.0 做好了充分的准备, 核心团队近期又迎来了两位新的开发者 José Valim 和 Carl Lerche 的加盟。José 是 Plataforma Tec 公司的联合创始人和主要开发人员, 公司主要涉及以 Ruby on Rails 的网站开发和咨询服务。José 从 2006 年就进入 Rails 开发领域, 并且在 2009 年参与到 Rails 项目的 Google Summer of Code 活动中, 积极贡献项目代码。另一位新加入的核心成员 Carl 是 Engine Yard 的开发工程师, 在 Web 开发方面有着丰富的经验, 早在 13 岁时, Carl 就开始使用 PHP 开发 Web 应用。目前, 这两位新加入的成员都在致力于 Rails 3 版本的开发工作。

Ruby 开发界最重大的会议 MountainWest RubyConf 2010 大会刚于三月中旬在美国犹他州的盐湖城落下帷幕。在 RubyConf 2010 会议的议题中, 自然不乏对新版本 Rails 的关注, 如 3.0 新版总架构师 Yehuda Katz 的演讲主题“编写模块化的 Ruby 代码: 从 Rails 3 开发过程中积累的知识”。当然, 会议还包含众多 Ruby 和 Rails 的新技术和应用展示, 如 Ruby 与纯函数式语言 Haskell 之间的桥梁 Hubris 介绍, 基于 MongoDB 的开发规则, Ruby 的宏, Lambda 演算细节, 以及 Ruby 之父 Matz 的大会发言等精彩纷呈的演讲主题。

对于希望尽快体验 Rails 3.0 新特性的开发者, 可以试用官方的 Rails 升级辅助插件 rails_upgrade 提供的 Rake 任务, 来尝试升级你的应用。此外, 由 Jeremy McAnally 撰写, 长约 120 页的“Rails 3 升级手册”(Rails 3 Upgrade Handbook) 也会是应用升级遇到问题时的有力辅助文档。

编程语言 2010 年第 03 期——搭上 LuaJIT 2.0 的高速列车

Lua 语言的即时编译器 LuaJIT 2.0 版本即将发布,从已发布的 2.0 Beta 版性能表现来看,执行效率的大幅提升令人惊讶。对比 Lua 语言的基准测试结果显示,提升幅度最大的 md5 算法,在 LuaJIT 2.0 下的执行效率是原生 Lua 执行效率的 152.7 倍。而对比使用 C 语言实现的快速傅氏变换,Lua 的性能是 C 的 1.48%,LuaJIT 1.1.5 的性能是 C 的 9.74%,而在 JIT 2.0 Beta 优化下的执行效率为 C 实现的 58%。LuaJIT 2.0 所带来执行效率的大幅提升,对于 Lua 开发者社区来说,无疑异常让人兴奋。

小巧的脚本语言 Lua,代码简洁优美,几乎能在所有操作系统和平台上编译运行。Lua 原生的解释器由标准 C 编写,解释器不过 200k 大小,在目前所有脚本引擎中,Lua 的速度几乎是最快的。与 Python、Ruby 等动态语言不同的是,Lua 并没有强大的类库支持,并不适合独立应用程序开发,但 Lua 生为嵌入式脚本,具备数学运算和字符串处理等操作,适合于嵌入在应用程序中提供灵活的扩展和定制功能。

较之 Perl、Python、Ruby 等其他动态语言,Lua 的运行效率已非常高,但依然有开发者尝试更进一步提升 Lua 的运行效率,于是诞生了 LuaJIT 项目,为 Lua 脚本提供在特定平台上的即时编译功能。项目利用即时编译技术(Just-In-Time)把 Lua 代码编译成本地机器码,然后交由处理器执行,让 Lua 的运行效率再跨上一个新的台阶。

对于即将到来的 LuaJIT 2.0 版本,来自德国的 LuaJIT 开发者 Mike Pall 表示,LuaJIT 2.0 的虚拟机被重新编写,包含使用汇编语言实现的高速语法解释器,以及全新设计的即时编译器,这些工作为 LuaJIT 2.0 执行效率的提升做出很大贡献。此外,一个创造性的代码跟踪编译器也集成在 LuaJIT 2.0 版本中,以增强编译器的表现。

在实现方面,LuaJIT 编译器支持 Lua 语言所有标准语义和标准库函数,接受相同源代码和预编译字节码。LuaJIT 同时保留了 Lua 轻量级、高效和可扩展的精髓。在代码编译时,所有函数缺省会被 JIT 编译为本地机器码,但开发者可以选择性打开和关闭即时编译函数、子函数甚至整个模块。同时,LuaJIT 优化器被分成几个可选模块,能够在运行时根据需要来加载。

即时编译技术被广泛用于 Java 字节码编译中,通过实时将字节码编译成机器语言来加速程序执行效率。显然,相对 JVM 对字节码的解释执行过程,经 JIT 翻译过的机器码执行效率更为迅速。同样,对于 Python 和 Ruby 等动态语言,也有自己的即时编译项目,Ludicrous (<http://rubystuff.org/ludicrous/>) 为 Ruby 代码提供了即时编译功能。而据称是 Google App Engine 团队成员创建的 unladen-swallow 项目 (<http://code.google.com/p/unladen-swallow/>),则通过将 Python 迁移到 LLVM 虚拟机上运行来提高其执行效率,目标是将执行效率提升至 CPython 实现的 5~7 倍。

编程语言 2010 年第 02 期——Django 1.2 开发进展与版本新特性

Python 的 Web 开发框架 Django 刚刚发布了 1.2 alpha 版本，在这个版本中，为 Django 增加了多数据库支持特性。开发者在 Django 框架中定义不同数据源之后，可以在数据查询中通过 `using()` 方法指定要查询的数据库，同时也可用参数指定特定的数据库来保存数据对象。

目前大型的 Web 应用开发，往往不仅在局限于后台单一的数据库系统，而是根据需要选择使用多种组合的数据存储方式。多数据库支持是评价一款 Web 开发框架可扩展性能的重要标志，而 Django 1.2 提供的多数据库支持特性，无疑为 Django 框架向企业级应用开发又迈进了一步。

此外，在 Django 1.2 alpha 版本中，还增加了可定制的 Message 消息框架，对于匿名和已验证的客户端提供了基于 cookie 和基于 session 的内置消息支持。消息框架替代了已失效的用户消息 API，并允许用户在请求提交中临时性的存储信息，以便在后续请求中使用。

新版本在安全性方面有了极大地改进，对跨站请求伪造 (CSRF) 攻击方式可能存在的安全漏洞进行了修补。并增加了 Django 发送 e-mail 邮件的配置，开发者可以指定 e-mail 发送的后端，而不使用默认的 SMTP 发送方式。此外，新版还添加了页面模板的缓存机制，使模板渲染时不必每次都从服务器磁盘上加载。

对于 Django 开发者，1.2 alpha 版本中提供的新增特性无疑使开发者对 Django 1.2 正式版更为期待。根据 Django 的发布路线图，1 月末发布的 Django 1.2 beta 版本将锁定所有新增特性，正式版本的 Django 将在 2010 年三月初到来。

值得一提的是，Django 可自动生成的 Admin 管理界面，如今也被移植到 Ruby 开发框架 Merb 之中。Django 开发者一直把 Admin 工具作为简化数据管理界面定制工作的有效利器。此次发布的 MerbAdmin (<http://github.com/sferik/merb-admin>)，作为 Merb 的一个插件提供，最新版本适用于 Merb 1.0.x 与 Merb 1.1 版本，并兼容 Ruby 1.9，支持 MySQL、PostgreSQL 及 SQLite 等多种数据库，支持 DataMapper、ActiveRecord 及 Sequel。

在 MerbAdmin 提供的功能与特性中，包括后台数据表展现，在管理界面中实现数据创建、更新与删除操作。同时，MerbAdmin 提供了表单数据自动验证、图片 URL 预览、数据表记录检索、数据过滤、排序、分页显示等功能。其安装和配置过程也十分简便，只需通过 `gem` 命令从 Gemcutter 安装，在 Merb 配置文件中启用并配置路由信息，执行安装命令，即可在启动 Merb 服务后在浏览器中访问到 MerbAdmin 界面。

Admin 管理界面无疑是 Django 框架最方便易用的功能，而优秀的特性常常在同类开发框架之间相互模仿。Merb 的主要开发者 Yehuda Katz——同时也是 Ruby on Rails 及 jQuery 核心团队的成员——也开始与 MerbAdmin 的开发者 Erik 合作，将 MerbAdmin 管理工具移植到 Ruby on Rails 之中。或许在不久之后，Rails 开发者也可以在开发框架中使用可自动生成的 Admin 管理界面，来轻松管理后台的数据表。

编程语言 2010 年第 01 期——JVM 上动态语言发展现状

运行在 Java 虚拟机上的动态语言已不在少数, 渐为开发者所熟知的包括 Ruby (JRuby)、Python (Jython)、Groovy 以及 JavaScript (Rhino) 等。在 JVM 跨平台特性的辅助下, 虚拟机上的动态语言被赋予了更为广阔的应用场景。

运行在 JVM 的动态语言中, JRuby 无疑是应用最为成功的。伴随 JRuby 版本更新, 在性能提升方面取得的进展已为众多开发者所认同。JRuby 开发的活跃, 也带动了 Ruby 在提供 JVM 支持的各种平台上的发展, 已有开发者将 JRuby 代码运行在 Android 手机平台上。

JRuby 新版本中提供哪些特性, JRuby 是否值得在项目实践中选用, 其今后的发展趋势又是如何? 下面是三位专家对 JRuby 的评价。

骆古道: 当前最新 JRuby 1.4 版本积极跟进对 Ruby on Rails 框架的支持, 并实现了对当前流行的 Ruby 1.8.7 版本的支持。这个 Ruby 版本将是 Rails 3.0 新里程碑版本的最低需求。

丁雪丰: JRuby 在 1.4 版本中, 在 Java 集成方面不仅增加了新方法, 还加入了新的 RedBridgE 框架。总的来说, 新版本在兼容性、性能、易用性等诸多方面都有了很大的提升。值得一提的是, JRuby 正在开发中的新中间表示技术, 在成型后将替代目前的 AST, 这也让我们看到了 JRuby 性能飞跃的曙光。

骆古道: 鉴于 JVM, JRuby 实现了 Ruby 语言真正的多线程。大量分析资料显示, JRuby 是 Ruby 1.8 最快运行环境。JRuby 1.4 出现了基于 Windows 平台的安装程序和全新的执行 JRuby 脚本程序, 它能够使得我们可以看到 jruby.exe 进程, 而不是 java.exe。JRuby 未来将会全面支持自 Ruby 语言发行以来最重要的版本 1.9。Java Web 开发者越来越多的正在使用 JRuby。

高昂: NetBeans IDE 在 JRuby 开发方面有着与生俱来的优势, 最新的 NetBeans 6.8 版本在全面支持 Java EE 6 的同时, 集成了 Sun GlassFish Enterprise Server v3, 同时支持 JRuby 1.4、Ruby 代码调试并提供 RSpec 1.2.7 支持。NetBeans 6.8 还改进了在 JRuby 开发中的重命名重构、类型推断以及导航支持。此外, 使用 JRuby on Rails 开发的 Web 应用可以直接部署在 GlassFish v3 中调试并运行。

动态语言 2009 年第 12 期——动态语言的 GIS 开发插件

在以 Google Maps 为代表的众多网络地图应用的推动下，GIS 开发已成为当前热点，与此同时，动态语言在 GIS 开发中的应用也越来越广泛。

对于 Django 的开发者，使用 GeoDjango (<http://geodjango.org/>) 在应用中嵌入地图要素无疑是再合适不过的。GeoDjango 是一款支持地理数据存储和空间查询的 Django 衍生项目。在使用 `django-admin.py` 脚本创建项目后，可在模型定义中借助 GeoDjango 来表达地理数据，并通过 `syncdb` 命令将包含地理几何字段的模型同步到 PostGIS 等支持空间拓展的数据库中。

在功能方面，GeoDjango 可以帮助开发者实现空间兴趣点查找，空间投影转换，地图数据展现等功能，并且 GeoDjango 在 Django Admin 管理界面中提供通过 OpenLayers 构建并修改地理要素的功能。在 Django 1.0 版本后，GeoDjango 已作为默认模块 `django.contrib.gis` 集成在框架中。GeoDjango 的详细配置和使用，Geoinformatics 站长撰写了中文 GeoDjango 教程 (<http://geoinformatics.cn/lab/geodjango/tutorial.html>)，以帮助开发者在 GeoDjango 的辅助下快速实现 Web GIS 应用。

对于 Ruby 和 Rails 开发者，Geokit (<http://geokit.rubyforge.org/>) 提供了非常便捷的 GIS 功能封装。Geokit 代码与 Ruby 1.9.1 保持兼容，其功能包含 geocoding 地址译码、地址查询器、距离计算、基于 IP 的位置查询及其他相关 GIS 功能。

除此之外，`acts_as_geocodable` (<http://graticule.rubyforge.org/>) 和 YM4R 也是非常好用的 Rails 地图插件，`acts_as_geocodable` 可以在 Rails 模型保存的同时自动进行地址译码工作，同时也提供了距离计算和位置查询功能。YM4R (<http://ym4r.rubyforge.org/>) 提供了针对 Google Maps 和 Yahoo! Maps 的 Rails 封装，以及 Mapstraction 插件来提供通用 JavaScript 形式的地图 API 类库，帮助开发者在 Rails 应用中方便地调用和使用地图数据。

另外，针对 Google Maps 的 Mashup 应用，还有专门的 Rails 插件 Cartographer (<http://github.com/joshuamiller/cartographer>) 来辅助开发者生成嵌入地图所需的 JavaScript 代码，如嵌入带有控制栏和标尺的地图，加入带有位置标注的提示窗口等内容，以简化开发者在编写 JavaScript 代码上的工作量。

动态语言 2009 年第 11 期——云计算基础设施中的动态语言项目

云计算平台能够提供可动态访问的主机、网络存储等多种资源，目前 Amazon、Google、Joyent 等厂商推出的云计算服务已为开发者所熟知。与此同时，Eucalyptus、Enomaly、Nimbus、OpenNebula 等开源云计算平台的搭建方案也层出不穷。动态语言编写的代码不仅被部署在云端实现业务应用，在云基础设施的搭建环节，也出现了不少动态语言实现的云项目。

Vertebra 项目是 EngineYard 推出的新一代云计算平台，并在 GitHub 储存库上对开发者提供项目源代码。Vertebra 基于 Erlang 和 Ruby 构建，具备服务广播和发现机制的消息总线，用于编写分布式实时应用。Vertebra 可适应异构云计算平台的差异，进行自动化流程及应用的管理。其中，包含安全机制，服务发现机制，状态机工作流系统和分布式键-值存储系统在内的核心代码由 Erlang 编写，Ruby 语言被用来编写中代理的框架和实际代理。Vertebra 致力实现安全、可拓展且易于管理的云计算组件，不仅能够运行在自有云基础设施中，也可部署在 Amazon EC2 或 VMware VCloud 等第三方云计算厂商搭建的环境中。

此外，开源项目 CloudCrowd 是一个完全使用 Ruby 语言实现的云方案，帮助 Ruby 开发者更为容易的编写分布式处理应用。CloudCrowd 设计目的是处理大规模且消耗资源较多的事务，比如完成大数据集迁移、视频编码、图片生成和处理、从 OCR 或 PDF 文件中进行文本抽取等工作。

相比较而言，用于实现云存储框架中分布式 Key-Value（键值对）存储系统的 CouchDB 项目，有着更为广泛的用户且开发社区非常活跃。CouchDB 项目由 Erlang 语言实现，是 Apache Incubator 开发的分布式、容错、面向文档的数据库，目前最新版本为 0.9。CouchDB 底层是一个 B-tree 存储结构，与传统关系型数据库不同的是，其数据表仅包含文档 ID、文档版本号和内容字段。CouchDB 在应对网站并发性上具有很大优势，在基准测试中，CouchDB 可以应对多达 2 万个并发连接。CouchDB 对外提供了简便易用的 RESTful API 格式的接口，可以使用不同语言的客户端如 Python 等进行调用，并且 CouchDB 中数据的读写都可通过调用其 HTTP 请求实现。

动态语言 2009 年第 10 期——Python 语言的拓展应用

伴随 Python 3.1.1 正式发布,更多新的特性被引入到 Python 语言中。在这个向后兼容的版本里,对开发者来说,最大的改进无疑是以 C 语言实现的 I/O 模块对 3.0 版本中 Python 代码实现的 I/O 模块进行了替换。更新之后的 I/O 模块,在打印输出和二进制文件读写方面的效率有了不同程度的进步,特别是在文本文件读取上,性能获得大幅度提升。除此之外,Python 3.1.1 版本对 int 类型进行了优化,并添加新的单元测试特性。

Python 语言 3.x 与 2.x 版本之间,一直存在着语言的向上兼容性问题,大多数国内开发者依然选择使用稳定成熟的 Python 2.5 版本进行开发,而很少有开发者将已有应用的代码重新改写,迁移到 Python 3.x 平台之上运行。此外,开发者在编码中所使用的很多第三方库,都是针对 python2.4/2.5 实现的,这也为应用向新版本迁移带来了一定程度的障碍。Google 的 App Engine 平台也一直在使用 Python 2.5 版本,来适应覆盖面最为广泛的 Python 开发者。

Python 版本存在的兼容性问题,并不影响 Python 语言在国内开发者之间的推广和使用。哲思社区组织撰写的《可爱的 Python》一书新近上架,ZoomQuiet 作为书籍主创,在互动出版网本月举办的九周年庆生会上,分享了成书过程以及在合著写书时所用的各种自由软件工具。不仅如此,在哲思社区发起的中文开放技术图书项目(OpenBookProject)中,还有多本技术书籍的撰写工作正在进行或计划中,以帮助国内开发者更多了解 Python 和相关的开发技术。

Python 简洁而清晰的语法,适合完成各种高层任务,其用武之地不仅体现在通用程序开发上。Python 拥有大量用于科学计算、图像绘制方面的高品质函数库,以帮助程序员实现基于 Python 语言的拓展应用开发。

在 Python 拓展应用方面,最近开发者 HYRY 正在撰写的《用 Python 做科学计算》(<http://hyry.dip.jp/pydoc/>)系列文章值得一读。在这个系列中,介绍了 Python 语言的数值计算库 NumPy、科学及工程计算库 SciPy、绘图库 Matplotlib、2D 绘图库 Chaco、3D 图形库 Visual 等函数库在应对不同种类开发方面的使用方法。系列文章还介绍了使用 Python 语言处理声音、图像、视频等各种数据的常用算法和处理方法。并介绍了书籍和教程撰写时使用的结构化文本格式 reStructuredText、HTML 转换工具 Sphinx,以及树状结构管理文本、代码的编辑器 Leo 等内容。

动态语言 2009 年第 09 期——Django 及国内 Python 社区进展

在 Django 框架的 1.1 发布版中，对 ORM 进行了增强，支持 `count()`，`max()`，`min()` 等聚合函数和查询表达式。在模型层，添加了非托管模型，允许用户选择使用脱离 django 管理的 model 层，并允许用户定义的 model 不与数据库表一一对应。在 Django 自带的 Admin 管理界面上，新版本添加了不少迎合开发者需求的新功能，包括可编辑并保存的 `changelist` 字段，同时，用户还可以定义 `admin actions` 来对一组记录执行特定操作。

Django 1.1 在 GIS 应用支持方面也得到了加强，GeoDjango (`django.contrib.gis`) 组件为 Django 开发的 Web 应用提供了地理数据存储和空间查询支持，并在新版本中实现了求和、求交等空间分析操作，并为 SQLite 数据库空间拓展 SpatiaLite 增加了支持。

Django 以其简便、快速的特性，吸引着越来越多的国内开发者关注和使用，已有不少开发者在 CPyUG 讨论组上公开了自己使用 Django 开发的 Web 应用及源代码。与 Django 类似，国内开发者 Limodou 的 Uliweb 项目也一直在进展之中，Uliweb 在 App 管理，Views 的方便性，url 映射，模板特性方面较之 Django 有自身的独到之处，感兴趣的开发者可以在 Limodou 发布的《Learning Uliweb》文档中获取更多细节。

除此之外，华麟用户组 (CPyUG) 和上海 Python 社区也在近期展开了线下技术聚会，与开发者分享的主题包含：“Django 开发的基于 Scrum 方法学的项目管理平台”，“使用 mock 进行单元测试”，“Python 几种框架的并发调度方式介绍和对比”，“基于 KDB+ 的大型日志数据分析系统”以及“Python 手机自动化测试应用”等。

对于关注 Ruby 的朋友来说，即将于 9 月 11 日在北京举行的 2009 敏捷中国大会上，敏捷宣言的缔造者之一、《程序员修炼之道》(2003 年 Jolt 效能大奖)、《Programming Ruby》以及《应用 Rails 进行敏捷 Web 开发》等书籍的作者 Dave Thomas，将为动态语言爱好者分享题为“Ruby 对象模型”和“程序员修炼之道，十年之后”的课程。

Dave Thomas 一直活跃在 Ruby 和技术书籍出版领域，在为初学者提供进阶指南的同时，还为有经验的 Ruby 程序员提供帮助。在题为“Ruby 对象模型”的演讲中，Dave Thomas 将为开发者分享自己对 Ruby 工作方式和 Ruby 在对象和类管理方面的理解。

动态语言 2009 年第 08 期——支持动态语言的 Android 平台

得到业界广泛支持的 Google Android 平台，以其开放式设计，使每个开发者都能获取 Android 平台源代码及开发 SDK。随着移动终端设备的普及和性能提升，未来在手持设备上的应用程序或许会超过 PC 上运行的应用。

Android 支持 Java 语法编程，吸引了不少 Java 程序员转向 Android 平台开发，而在最近发布的 Android Scripting Environment (ASE) 项目支持下，为 Android 平台增加了 Python、Lua 及 Beanshell 等脚本语言的支持，并且在不久后还将提供 Ruby 语言和 JavaScript 支持。同时，ASE 项目还计划让更多 JVM 语言运行在 Android 上，包括改进 Android 对于 JRuby 的支持等。

借助 ASE 的支持，开发者可以脱离 Android 开发工具，直接在 Android 平台上编写并执行动态语言代码，来进行快速应用原型搭建，而无需通过原先 PC 端的 IDE 启动、编译、部署等环节。

ASE 使用 Python 及 Lua 等动态语言的本地运行时来执行脚本代码，通过 JSON-RPC 为动态语言代码提供 Android API 的访问方法。而对轻量级 Java 脚本语言 BeanShell 的执行，基于 JVM 的解释器可在 ASE 内部直接加载，而无需间接的 RPC 调用。除此之外，ASE 设计了更为简化的动态语言调用接口，来实现 Android 平台上 Intents 操作、Activities 启动、电话呼叫、短消息发送、条形码扫描、地理位置获取、传感器数据获取等调用操作。

在 ASE 官方站点 (<http://code.google.com/p/android-scripting/>) 给出的示例代码段中，使用脚本语言 Lua，通过获取 Android 手机传感器信号，实现了将手机屏幕向下放置，可以让手机自动调整到静音状态，将屏幕向上放置，就自动打开手机铃声。同时还提供了 Python 语言的代码段，实现收到用户聊天消息时，将发信人和聊天内容播报出来的功能。

对比 iPhone 平台的开发，苹果公司的策略致使 iPhone 平台不向第三方开发者开放，同时平台自身的封闭性不允许应用自带解释器，自然无法实现动态语言在 iPhone 平台上解释执行。与其不同的是，Android 平台开源且开放的特性对开发者则更为友善，动态语言的引入，将大幅简化移动平台的开发工作，帮助开发者快速实验新的想法，把不同语言的开发者融入 Android 平台，为 Android 手机应用的不断丰富增加新的活力。

动态语言 2009 年第 07 期——动态语言的代码托管云服务

Heroku 的 Rails 托管方案自推出之日起，一直倍受 Rails 开发者好评，在为其一年的免费测试阶段之后，详细的商业销售计划于近期推出。

以提供方便使用且稳定 Rails 托管服务而著称的 Heroku，对于 Ruby 和 Rails 开发者来说并不陌生。Heroku 在易用性设计方面值得称道，其提供的 gem 插件能帮助用户在最短时间里，将 Rails 应用直接导入到托管环境中。

站点可扩展性架构方面的讨论，一直是开发者所关注的热点，而 Heroku 公开的站点架构设计方案和使用的组建，同样也适合大多数 Web 2.0 站点参考。Heroku 采用 Git 进行版本管理，其背后的云计算平台，让用户在托管代码之后的性能拓展变得异常方便，使用者只需要通过简单的选项，即可实现已部署应用在架构和性能上的调整。

Heroku 最基础的 5M 数据库空间目前仍保证免费提供，商业用户可以选择购买从 50M 到 2TB 不等的数据存储空间。除此之外，用户可以定制应用所使用的动态网格 (Dyno Grid) 数目，以类似于应用服务器集群的方式来增加应用的负载性能。值得一提的是，Heroku 还为用户提供了多种易用的可选插件，包括域名绑定、https 访问支持、站点备份、动态子域名等，以符合用户托管空间的特定需求。

Heroku 以其良好的架构设计和用户支持度，在去年曾获得了 300 万美金天使投资。与此同时，提供动态语言托管服务提供商一直处于增长势头，除了 Amazon 的老牌的 EC2 服务和 Python 与 Django 开发者热衷使用的 Google App Engine，Engine Yard 也推出了新的 Rails 托管服务 Solo，来作为原有云服务的补充，为用户提供更高性价比的解决方案。

此外，Engine Yard 新生的云服务 Flex，则面向专业用户提供了具备强大可靠性与伸缩性，并且支持集群的 Rails 应用托管环境。Flex 提供了与敏捷开发过程相适应的特性，如将储存库中的代码自动部署在托管环境中，以及应用服务器自动恢复等技术。

以 Ruby on Rails 为代表的动态语言 Web 开发框架，以其强大生产力得到开发者的认可，而与之相应的代码托管服务也将越来越多，并且这些服务无一例外具备云计算特性。无疑，这样的服务正在受到越来越多开发者青睐，也必将成为更多动态语言代码部署的首选平台。

动态语言 2009 年第 06 期——你是否关注 Twitter 的性能改进

对我来说，Twitter 火爆的最好证明，并非是奥巴马竞选总统时借助 Twitter 来推广执政理念，而是经常接触的技术圈好友，无一例外的出现在 Twitter 之上，并且喜欢上了 Twitter 的信息传播方式。

Twitter 用户和消息数量在以惊人速度膨胀的同时，其客户端应用也以各种形式出现在我们的周围。主流操作系统上的 Twitter 应用不必细数，iPhone、Windows Mobile、Symbian 等移动平台上的 Twitter 客户端也足够使用者费一番功夫进行选择。

同时，Twitter 客户端还在不断与我们身边应用进行集成，单在 Firefox 浏览器上，就有 TwitterFox、TwitKit、TwitterBar 等二十余款不同的 Twitter 插件。还有一个名为 Twessenger 的 MSN 插件，可以将 Windows Live Messenger 的自定义状态同步更新到 Twitter 之上。笔者也尝试使用 Java 库 Twitter4J 写了个小的 OpenMapTwitter 应用 host 在开源项目站点 kenai.com 上。

基于 Ruby on Rails 开发的 Twitter，由于其高速增长访问量而带来的性能烦恼，曾在开发社区中引发了众多关于 Rails 可伸缩性问题的争论。前不久的 QCon 伦敦站，来自 Twitter 的开发者和性能优化专家 Evan Weaver 在演讲中（讲义可在 qconlondon.com 站点的 Schedule 栏目中下载）介绍了 Twitter 应对高访问量的优化细节。

Twitter 的优化涉及到 Rails、C 代码、Scala, Java, MySQL 等不同的层面。

Ruby 代码方面，采用写入式缓存替代原先的读取式缓存，并且自己开发了 Cache Money 插件代替 ActiveRecord 的写入式缓存机制，该插件最初由 Twitter 的工程师 Nick Kallen 在自己的博客中宣布将这个插件开源。

在原先的读取式缓存工作模式下，执行查询动作时，先在分布式内存对象缓存系统 Memcached 中查找，之后在数据库中查找，如果缓存中没有查询的对象，则生成新的结果。

而写入式缓存模式下，在对象创建、变更和删除的同时，自动更新高速缓存中的内容，以此提升查询性能。在完全写入状态下，数据在更新高速缓存时，同时也被写到磁盘上。尽管 cache-money 无法覆盖所有的查询，但已支持最为常用的 Active Record 查询操作，并在检索效率上做了优化。

除此之外，Twitter 架构上其他的几处改进包含：采用基于 C 库 libmemcached 的 Memcached Client 方案，使用了灵活的 Scala 语言改写中间层 Ruby 的 GC 实现，并通过 MBARI 和 Railsbench 等 Ruby 补丁来提升应用的性能等。

在 Evan 幻灯片的最后，给出各种提升 Twitter 性能的工具的链接地址。如果你的 Rails 应用在性能拓展方面遇到问题，Twitter 的一系列改进方法无疑具有极大的参考价值。

动态语言 2009 年第 05 期——JavaFX 的挑战与机遇

JSR 223 规范为 JVM 加入了动态语言运行时的支持。当动态语言代码转换为字节码运行在 JVM 之上，可以方便享用 Java 平台在拓展性、移植性和安全性等方面的诸多优势，同时可以引入 Java API 和为数众多的第三方库。

JavaFX 是运行在 JVM 上的脚本语言，自两年前在 JavaOne 大会诞生算起，已经由最初 Chris Oliver 创建的实验性项目 F3 演变成一套完善的 RIA 解决方案。

在语法上，JavaFX 使用专门设计的 DSL 对 GUI 界面描述，融合了 JavaScript、ActionScript 等脚本语言的特征，并且同 Java 代码有一定相似性，但较之 Java 代码抽象层次更高，且直观易读。同时，JavaFX 文件简短，无需其他 RIA 实现冗长的 XML 界面描述，对习惯于编写代码的开发者来说，完全可以脱离设计器手工编写 JavaFX 应用的界面。

较之 Flex、Ajax、Silverlight 等主流 RIA 技术的脚本语言，JavaFX Script 起步较晚，但在技术上却有独到之处。除 JavaFX SDK 提供的通用特效、动画功能之外，代码中可以直接调用 Java API 和第三方 Java 类库。应对桌面开发，可借助 Java 多线程特性来保证应用的健壮性。尽管 JavaFX 并非为取代 Swing 界面而出现，但有开发者已开始尝试在 Java 代码中反向调用 JavaFX 定义的类，以便在应用的 MVC 分层中，表现层使用 JavaFX，逻辑和控制层由 Java 代码实现。

在 JavaFX 目前最新的 1.1.1 版本中，提供 JavaFX Mobile 提供了移动终端支持，借助 JavaME 的广泛应用，JavaFX 在移动终端平台的优势要强于 Flex 和 Silverlight 等 RIA 技术。

但需要注意的是，JavaFX 代码通过自身 Class Loader 加载到 JVM 上运行的效率还需要进一步优化，一般不大的 JavaFX 应用从装载到启动的时间仍然相对较长。在 IDE 支持方面，NetBeans 提供的 JavaFX 模块已集成了不少可拖拽添加的 JavaFX 组件，但在用户界面控件上较之 Flex 和 Silverlight 的开发工具仍然相对薄弱。不过 Sun 已在计划 JavaFX 的 2.0 版本中提供完整的用户界面组件，包含容器和布局控制等不同层面。

较之 Flex 与 Silverlight 庞大的开发者群体，JavaFX 的开发者 and 社区规模仍然薄弱，JavaFX 还需要更多的实践和成功案例来吸引前端开发者加入其中。但如果你是 Java 技术的爱好者，同时又对 RIA 感兴趣，JavaFX 将会是一个不错的开始。

动态语言 2009 年第 04 期——Python 近况全接触

在三月份德国汉诺威举办的 CeBIT 2009 大会上，Python 获得了 Linux New Media 颁发的最佳开源编程语言奖，以肯定 Python 一直在提升语言功能性和易用性方面做出的贡献。

与此同时，Python 3.1 的第一个测试版本 3.1 Alpha 1 已经到来，Python 3.1 保持了与 Python 3.0 的兼容性，采用 C 语言重新实现的 IO 模块极大改善了处理性能，同时这个发布版修订了模块和命令行参数执行时已报告的 Bug。

关于 Python 3.0 和 2.6 版本的选择，Python 之父 Guido 在不久前接受采访时谈到：开发者可以根据喜好自由选择使用 Python 3.0 或 2.6 版本，使用者不需要担心由于使用 2.6 版本而有被抛在后面的风险。3.0 和 2.6 版本都是由同一个核心 Python 开发小组维护，版本更新几乎同步。但对于初学者来说，直接从 Python 3.0 启程将更加容易上手，一些困扰初学者的过时语法在 3.0 中已被去除。当学习 3.0 版本后再去了解 2.x 版本，则更容易掌握 2.6 和 3.0 版本间的区别。

对于打算深入 Python，通过研究 Python 来提高代码性能和功能的爱好者来说，RobertChen 所著并由 CPyUG 和 ZEUUX 社区审校的《Python 源码剖析》一书将会是不错的借鉴，在书中对 CPython 虚拟机及 Python 技术实现和特性进行了分析，并对 Python 简洁语法背后的机理给出完善的解释。

IDE 方面，NetBeans 新版本 6.7 将在六月份 JavaOne 大会的同时发布。目前 Ruby 和 PHP 的开发支持模块已作为可选项集成在 IDE 中，针对于 Python 开发的 NetBeans Early Access 版本也可作为单独运行包下载。这款借助于 NetBeans Platform 的 OpenIDE API 以及插件模块技术开发的 Python IDE，默认集成了 Jython 支持，使用者也可以根据需要切换到 CPython 下使用，其压缩包仅有 28M 大小。

在 Python 的中文社区方面，由 ZoomQuiet 发起的开放图书计划一直在推动着中文 Python 技术图书的策划与写作，一本名为《可爱的 Python》的技术书籍目前正在最后修订中，将在不久后与 Python 爱好者们见面。

Python Web 框架 Uliweb 的开发者 limodou 也正在撰写一本名为《Learning Uliweb》的开放文档，来向开发者介绍如何使用 Uliweb 进行 Web 开发，以及作者在 Web 开发框架设计方面的经验和思考。

动态语言 2009 年第 03 期——Django 是否在重复制造轮子？

Python 的 Web 开发框架 Django 自 1.0 版本发布后，已经逐步趋向稳定，目前最新版本为 1.0.2，功能方面没有大的改动。五月份即将在捷克首都布拉格举办的 Django 大会 EuroDjangoCon 2009 正在展开各项准备工作，来自 Digg、Osmosoft、Six Apart 等 Web 应用开发者将在大会中分享各自的 Django 开发经验。

在 Django 各方面功能趋向稳定之后，在国内 Python 社区引发了一场关于 Django 优缺点的讨论。“Django 到底有什么好？”，对于这个问题，开发者们对比了 Django 与其他 Python Web 框架的特点与异同之处，并给出了各自的看法。

开发者令狐虫对 Django、TurboGears、CherryPy、Pylons 等 Python 开发框架进行了对比，认为 Django 重造了所有轮子，并不符合具有底层控制欲望的开发者和程序极客的习惯。Limodou 就此谈到：框架会面临许许多多的使用者，其层次不同，要求不同，水平不同。而 Django 的特点就是统一、集成、功能完备、自动生成的 Admin 界面也是极为出色的亮点。Henryhuang 认为：Django 做了所有框架相关的工作，可以让开发者的注意力集中在业务逻辑之上。对于喜欢自行定制的开发者的，Django 的 Middleware，ORM 等组件都允许热插拔，在实际开发时可以选择使用其他组件进行替换。而 Django 自带的 MVT，以及大量脚本都极大加快了开发效率。

一款框架到底是否值得选用？尽管这是每个开发新手在面临选择并开始学习时就要考虑的问题，但根据个人情况的不同，很难说有放之四海而皆准的答案存在。任何一个框架都有自己的优缺点，能找到一款合适喜欢的框架进行深入学习和应用，无疑是一件让人兴奋的事情。对于大多数开发者，如果更注重快节奏开发，希望快速建立原型并逐步完善，那么 Django 无疑是全栈式解决方案的不错选择。

由 Limodou 创建的另一款 Python 框架 uliweb 也与 Django 的 MVT 模型类似，项目由若干个相对独立的 app 构成，每个 app 有自己的配置文件，templates 和 static 目录。uliweb 在开发过程中将更注重每个 app 的质量，以便于将代码与开发经验进行模块化重用。与项目对应的 uliweb-tests 项目提供了 uliweb 的不同测试程序，以方便开发者测试 Web 应用。

其实，重复发明轮子并非毫无意义的事情。恰恰相反，在 Python 的世界里，解决领域问题的最佳方式，或许就是在已有代码基础上，制造出合适自己的轮子。

动态语言 2009 年第 02 期——动态语言社区回顾

对动态语言来说，2008 无疑是取得长足进展的一年。Ruby、Python、PHP、Rails 等开发技术不仅自身保持着火热的势头，还带动了众多有趣实用的动态语言如 Erlang、Lua、Perl、JRuby、Jython、Pike、Ioke 等技术的进步或复苏。毋庸置疑，无论是雨后春笋抑或是涅槃重生，这些技术在国内正在受到越来越多开发者的关注，并在实际生产环境中发挥着越来越重要的作用。

技术活动是开发者们近距离接触和交流的最佳方式，回顾国内的 unconference 活动 Open Party 和 Open Source Camp，以动态语言为主题的 Topic 已不再是少数开发者讨论的话题：Grails 应用开发、GAE 之上的 Django 框架、Erlang 程序设计、JRuby 实战、Groovy 和 Grails 等，这些不同的 Topic 让参与其中的开发者们对主流动态语言技术耳熟能详。在 2008 软件开发 2.0 技术大会上，专家们也带来了 Erlang、Scala、ActionScript、DSL 等与动态语言密切相关的主题，这在一定层面上反映了业界对于动态语言的重视。

年关将近，回顾和展望也成为各个技术社区必不可少的主题。Ruby 方面，InfoQ 中文站推出了最受关注的十大 Ruby 新闻，主打新闻包括 Ruby 1.9 新增特性；Groovy 和 Grails 技术进展；JRuby 开发者 Ola Bini 在 JVM 之上打造的强类型、基于原型的动态语言 Ioke；以及 InfoWorld 最佳 Rails 开发 IDE 评选等内容。

JavaEye 也评选出最受关注的 20 篇 Ruby 新闻，包括可以在 JavaScript 和 Flash 虚拟机上运行 Ruby 代码的 HotRuby 项目；Merb 框架将合并到 Rails 3.0 版本中，并预计在 2009 年第二季度发布；GemStone 公司开发的高性能 Ruby 虚拟机 MegLev 项目；DHH 谈 Rails 文化以及 Basecamp 扩展等内容。

IDE 之争依旧精彩纷呈，NetBeans 以其免费开源、版本更新迅速等特点吸引了众多开发者尝试并使用，预计在 09 年内发布的 NetBeans 7，已为 Ruby 和 Python IDE 加入了开发者们期待的代码覆盖测试等功能。在 NetBeans 中，对于 Ruby、Python、PHP、Scala、Groovy、Erlang（ErlyBird 插件）等动态语言都提供了支持，其中 JavaEye 会员 dcaoyuan 则是 Scala 插件的主要贡献者。

除此之外，JetBrains 开发的 RubyMine 建立在 IntelliJ IDEA 平台上，在必备的编辑、调试、代码自动提示及补全等功能外，还提供多种便捷功能来提升开发效率。RubyMine 号称最为智能的 Ruby IDE，如果这款 IDE 工具如同使用 IntelliJ IDEA 编写 Java 代码一般得心应手，无疑将非常值得期待。

相信来年动态语言技术和社区的发展将更加精彩，祝动态语言爱好者们新春快乐！

动态语言 2009 年第 01 期——是谁让动态语言归来

Python 3000 最终版本已在 12 月份发布, 新的 Python 3.0 与 2.x 版本相比, 在字典和字符串等众多内置对象细节方面发生了较大变化, 并且标准库的部分内容也已进行重新组织。

伴随新版本发布, Python 社区关注重点将会逐渐转向到 Python 3000 之上, 对于已有代码, 如果您打算向新的平台迁移, 可以尝试官方 SVN 储存库中提供的代码自动转换工具 Python 2to3。2to3 工具可以根据参数将指定文件或目录代码转换成 Python 3.0, 当然, 2to3 工具需要运行在最新的 Python 2.5 环境下才可以顺利完成转换。

Python 语言以其优雅设计和开发效率而著称, NetBeans IDE 也为 Python 开发者推出仅有 28M 大小的 NetBeans 6.5 Python Early Access 版本, 在 IDE 中提供了高亮显示、代码折叠、智能感知等代码编辑特性, 并同时支持 Jython 和 Python 两种运行时环境。

昔日王者 Smalltalk, 最近由于其 Web 框架 Seaside 的日益火热而受到众多关注。这门早在上世纪七八十年代就已开始流行的语言, 曾对 Ruby 的发展产生过深远影响, 发明人 Alan Kay 也曾是图灵奖得主。

Seaside 构建在 Smalltalk 实现的 Squeak 虚拟机之上, 可以帮助 Smalltalk 开发者进行快速 Web 应用搭建, 目前 Seaside 最新版本是 2.8, 在这个版本中, 提高了程序运行的性能, 并且较之上一个版本的内存消耗降低到四分之一, 此外在国际化方面也提供了更为完善的支持, RoR 之父 DHH 曾在自己博客中高度评价 Seaside 的优异表现。

毫无疑问, IT 领域每年都会有众多新事物诞生, 同时, 已有技术也在新技术的带动促进下发生着不断的演进, 一旦时机成熟, 便会展现出新的活力。在一定程度上, 这些无疑都为整个开发技术的繁荣做出贡献。从 Python 到 Ruby, 从 Smalltalk 到 Erlang, 还有 Lisp、Perl、Groovy 以及轻量级 Java 脚本语言 BeanShell, 不论是老枪或是新兵, 越来越多的动态语言正在走进开发者们的视线。

正是开放源代码社区和动态语言社区中开发者们的坚持和努力, 让一度远离大众视线的动态语言焕发出勃勃生机, 而其中某些技术的成功, 又成为带动相关技术跨越式发展的动力。正如 Rails 的成功, 为 Django、Seaside、CakePHP 等框架的兴起和发展指明了方向。动态语言及其技术便如此这般, 在一个个反复轮回中生生不息。

动态语言 2008 年第 12 期——支持动态语言的 GlassFish v3

本月 Python 3000 的第二个发布候选版本已如期而至，新版本针对原生类库中存在的 Bug 进行了修正，并且修改了在 Cygwin 环境下由于文件名过短而带来的问题，以及在 Windows 中读入非 ASC 码字符时出现的错误。Python 3000 的最终版本计划将在 2008 年 12 月份发布，如果开发团队的进程一切顺利，或许在您读到这段文字的时候，已经可以在 Python 3000 的环境下调试您的 Python 代码了。

Ruby 方面，JRuby 1.1.5 已经正式发布，在新版本中修正了在 REXML 解析 XML 请求时可能引发的 DoS 攻击漏洞，在这个修正之前，开发者一般需要给 REXML 模块中的实体类打上动态补丁，来确保应用不会受到此漏洞所造成的 DoS 攻击。

在新版 JRuby 中，实现的改动还包括对所有类均含有的方法进行缓存，通过常量缓存来提升常量查找效率，降低编译执行时的内存占用，以及引入 1.3.1 版本的 Ruby Gems 管理器等新的改进。

同时，Rails 2.2 的第一个发布候选版本也已经来临，DHH 已经为 Rails 2.2 版本即将到来的正式发布做好了准备，新版 Rails 2.2 将在国际化、线程安全以及 JRuby 和 Ruby 1.9 兼容性等方面有较大的改进。

开源应用服务器 GlassFish v3 Prelude 本月正式发布，v3 Prelude 架构在 Apache Felix 实现的 OSGi 运行时之上，提供了非常强的拓展性，在简化组件的同时注重了容器的轻量化。EJB 容器、Admin 管理控制台及 Metro Web 服务栈等备选组件均可通过 GlassFish v3 自带的 Update 工具定制安装。除此之外，v3 Prelude 提供的功能还包括重新部署时的 Session 保留、NetBeans 和 Eclipse 插件支持、嵌入的 API、符合 REST 风格的 Web 服务以及基于 IPS 的升级中心等等。

值得一提的是，GlassFish v3 为多种动态语言提供了运行平台。在 GlassFish 应用服务器中，目前已对 Groovy 和 JRuby 运行环境提供了良好的支持。同时，GlassFish 水族馆博客中的文章透露，在使用 Python 语言实现 WSGI 规范的 Modjy 项目支持下，已可以让 Jython 和 Django 运行在 GlassFish 之上。在对 PHP 支持方面，借助于 Java 实现的 PHP 5 解释器 Quercus，已可以在 GlassFish 中运行 PHP 应用，并给出 MediaWiki 代码包在 GlassFish 中的部署实例。在开源 Java 集群框架 Terracotta 最新的 2.7 版本里，已提供面向 GlassFish 的集群支持。同时，GlassFish 自身在应用服务器集群方面也有先天的优势。GlassFish 正在为基于动态语言开发的 Web 应用提供越来越完善稳定的支持，或许会有更多的企业开始尝试将应用移植部署到 GlassFish 应用服务器之上。

动态语言 2008 年第 11 期——JRuby on Rails 上的项目宿主平台

Kenai

Python 2.6 和 3.0 的开发周期一直保持同步，并且版本发布时间也始终一致。Python 3.0 的开发进展带动着 Python 2.6 版本的特性更新，Python 2.6 版本部分新的特性是由 Python 3.0 反向移植而来。在经历数个 Beta 版本后，Python 2.6 终于迎来正式版发布。

在 Python 2.6 开发过程中，Bug 管理工作从先前的 SourceForge 平台转移到基于开源项目 Roundup 的 Bug 管理系统之上。同时，新的文档将使用 Python 开源项目 Sphinx 来产生美观的 HTML 输出，以替代之前一直沿用的 LaTeX 格式。除此之外，Python 2.6 在字符串格式化、异常处理、Byte 文本方面都进行了改进。更多细节可以查看 Python 开发者林胜在站点 onlypython.com 给出的 Python 2.6 版本新特性全文翻译。

Ruby 方面，随着基于 Rails 的站点逐步增多，Ruby 和 Rails 开发职位一直保持着足量供应的势头。站点 RubyInside.com 每月都会列出最具吸引力的 Ruby 和 Rails 的职位以备开发者参考。同样，37signals 的工作布告板以及站点 jobs.rubynow.com 也都会列出最新的开发职位介绍。在国内，上海 Rails 用户组也在 10 月份举办了第五次 Shanghai on Rails 开发者活动，内容包括针对初学者和进阶者介绍的 Rails 指南与技巧，旨在为 Rails 开发者搭建一个下线交流与分享的平台，并吸引更多开发者了解并开始使用 Rails。

IDE 方面，为 Visual Studio 提供 Ruby 开发支持的 Ruby In Steel 发布了新的免费版和个人版。Ruby In Steel 个人版本中，在增强了语法感知、代码着色、自动缩进、代码格式化等辅助功能之外，还在 All-in-One 安装包中包含了一份 Visual Studio 的免费拷贝，以方便开发者安装后直接使用。

Sun 公司为开源项目提供的宿主平台 Kenai.com 已正式开放，Kenai 提供了 SVN 和 Mercurial 代码储存库支持，Bug 管理平台，以及 Wiki 和邮件组等项目协作工具。值得一提的是，Kenai 站点采用 Script 和 Java 结合的方式，构建在 JRuby on Rails 之上。其后支撑平台全部来自 Sun 开源软件栈，包括 GlassFish 应用服务器、MySQL 数据库和 Solaris 操作系统等。与 Sun 传统的开源项目宿主平台 java.net 相比，Kenai 的设计更强调社区开发者之间的互动，通过易用的管理平台来带动社区活跃度，促进项目的良性发展。Sun 旗下的开源软件和 Web 应用为数不少，但组织上还欠系统化。如何有机联合起来，作为一个强大的整体为开发者提供服务，还需在探索中做出不断努力。

动态语言 2008 年第 10 期——开源世界的节日

本月初 Django 的 1.0 版本如期发布，在发布声明中这样写道：来自于 Django 社区的热情和凝聚力一直推动着项目的进展，在 Django 全部的代码中，大约有一半由非核心提交者所贡献。从上个稳定版本到 1.0 的发布，我们添加、修改和删除的代码超过了 350000 行。

Django 项目诞生 3 年来，经过漫长而细致的开发，终于发布了 1.0 版本，难掩兴奋之情的 Django 开发团队在名为 Tied House 的酒吧举办了发布庆祝会，邀请 Django 爱好者共同分享他们的喜悦。

在 Django 1.0 版本发布的同时，第一届 DjangoCon 大会也在 Google 总部所在地，位于加州 Santa Clara 山景城的 Googleplex 召开。会议由 Django 软件基金会和 Google 的 Open Source 团队共同组织，来自伦敦用户组的 Django 开发者 Robert Lofthouse 担任本次会议主席。DjangoCon 上众多主题演讲都涉及到 1.0 版本的最新特性，包括性能提升，新的管理界面设计，ORM 内部细节等等。

越来越多开发者看好 Python 和 Django，除了语言和框架本身的吸引力之外，将 Python 作为三种主要开发语言之一的 Google 公司的推动和支持也不可忽视。Python 之父 Guido 曾数次表示，Django 是他目前最喜爱的 Python Web 开发框架。自然，Guido 在 DjangoCon 上带来题为 Django 与 Google App Engine 的演讲也备受开发者青睐。GAE 的开发应用一直是国内开发者，特别是 Python 社区关注的焦点。在八月底举办的北京 Open Party 上，来自啄木鸟社区的清风就分享了自己在 GAE 上进行 Django 开发时积累的实践经验。

值得一提的是，关注地理位置服务应用的开发者也逐步增多。DjangoCon 上介绍的 GeoDjango 框架与 Rails 社区的 Map Rails Kit 和 GeoKit 都为今后逐渐增多的基于地图应用的 Web 开发做好了充分的准备。

Ruby 社区本月也同样热闹，Ruby 类库 NeverBlock 发布了新的版本，使用多线程方式运行 Rails，可在很大程度上提高 MySQL 或 PostgreSQL 的使用率。在 NeverBlock 官方提供的性能测试上，对比常规的 Rails 运行方式在性能上有了大幅度提升，但在 Rails 开发者使用 Thin、Rails 和 PostgreSQL 进行的实际对比测试中，发现 NeverBlock 带来的性能提升还需要更进一步的测试来检验。

九月份自由软件爱好者们再次迎来了 FOSS 软件自由日，这是属于开源和自由软件爱好者的节日。源自于开源社区的动态语言也将随着开发者的增多，将有益成果更多回馈于开发者社区，在来年的软件自由日中扮演更加重要的角色。

动态语言 2008 年第 09 期——JVM 之上的动态语言

JVM 之上可用的动态语言已不在少数，细数有 JRuby、Jython、Groovy、Jacl (Tcl 的 Java 实现) 等多种实现。工作在 JVM 上的开发者对动态语言的积极态度，也使得 Sun 公司越来越重视 JVM 的多语言支持能力，并一直在改进动态语言在 JVM 上的易用性和执行效率。

就面向 Web 开发的动态语言框架来看，JRuby 凭借 Rails 的杀手级应用无疑已经占得先机。伴随 JRuby 1.1.3 版本发布，JRuby on Rails 逐步增多的应用案例将会带给开发者更多的信心。在 1.1.3 版本中，提供了对 RubyGems 1.2 的支持，并且在优化解释器性能的同时极大降低了内存使用率，JRuby 对于调试器的支持以及 IO 内存使用效率也得到了巨大改善。

JRuby 可将代码编译为 Class 字节码在 JVM 上运行，其编码生产力无疑非常高，但函数式的语法却给 Java 开发者带来较大的学习曲线。当然，目前 JRuby 对于 Java 库支持及 JRuby 原生库方面在不断改善。并且供职于伦敦 ThoughtWorks 的 Ola Bini 和道喜技术日记站长骆古道老师的两本 JRuby 开发书籍也先后上架，在引领开发者将 Rails 和 Ruby 应用到 Java 平台方面提供了有益指导，并在 JMX 上也给出详细介绍，书中给出的指南也将吸引更多开发者尝试在项目中采用 JRuby。

值得注意的是，随着 Python Web 开发框架的不断改进，近期 Jython 的活跃程度也在不断上升。即将发布 1.0 版本的 Python 框架 Django 也将能够运行在 Jython 平台之上。Jython 的开发者 Jim Baker 在其 Blog 上展示了 Django 框架向 Jython 平台的迁移成果，同时在 <http://dojstatus.leosoto.com> 上给出了 Django 框架中每个文件向 Jython 迁移的测试统计结果。Django 1.0 的推出以及 Django 向 Jython 的迁移，或许将会成为吸引开发者使用 Jython 的杀手级应用。

对于 Java 开发者来说，在部分环节使用动态语言可以加速整个开发进程。在 JVM 上动态语言不断发展的带动下，Sun 公司官方的 NetBeans IDE 也为动态语言提供了强有力的支持。目前，JRuby 已是 Netbeans RubyIDE 的默认解释器。支持语法高亮、代码补全、断点调试、PyUnit、Python 库管理等特性的 nbPython 插件也为 Jython 开发提供了良好支持。与此同时，Sun 的 Ruby 和 Python 开发者中心也相继上线，吸引更多的开发者在 JVM 之上使用动态语言来提高生产效率。

动态语言 2008 年第 08 期——聚焦 Django 1.0

Python 框架 Django 在 0.96 版停留了相当长的一段时间后，终于公布了 1.0 版本的发布计划。在官方公告中，Django 开发者 Jacob 提醒用户牢记：1.0 版本的正式发布将定于 9 月 2 日。

与 Rails 框架如火如荼的发展势头恰恰相反，Django 的开发者在版本更新上一一直相对保守，也没有过多的进行宣传造势，只是一直在慢条斯理的埋头做着喜欢的事情。在 1.0 版本发布之前，还将会有一个 alpha 版本（7 月 20 号）来包含所有必须的新特性，之后还会有两个 beta 版本（8 月初）来冻结 1.0 版本的最终特性，并且尽量剔除所有的 Bug。当距离庆祝还有一步之遥的时候，将会释出一个 Release Candidate 版本（8 月 19 号）来为最后的发布做好充分的准备工作。

1.0 版本的 Django 将使用 newforms 来对 admin 模块进行重构，并将完全替换原先使用的 oldforms，不会再有功能对其产生依赖。新版的 Django 还将 100% 同用以实现 Python 框架和 Web 服务器之间接口的 WSGI（Web Server Gateway Interface）保持兼容，以保证应用的可移植性。尽管 Django 1.0 在很长一段时间里迟迟未能露面，但不少 Python 的开发者依然看好这款框架，Django 开发的成功案例也在不断增多。在 Django 官方站提供的资料中，收录了近 1500 个 Django 开发的站点示例，虽然在数量上仍无法与 Rails 比拟，但其中不乏优秀的站点应用出现，如好看薄（haokanbu.com）等极具特色的 Web 2.0 应用。

目前 Django 的 SVN 存储库已经有了 Git 的克隆版本，用户可以在 Git 存储库中随时获取 Django 的最新开发代码。与此同时，Django 项目在发展了五年之后，成立了非盈利的开发者基金会，来运作 Django 开发者的讨论、集会及其他社区活动，并且促进 Django 全球开发者社区的活动同时推进 Django 项目的持续发展。

InfoWorld 日前对主流 Rails 开发 IDE 进行了比较，并从功能性、易用性、集成性、性能表现等方面进行综合评分。具备众多优秀特性的 Netbeans 6.1 在评比中得到了最高的 9.0 分。同时参加评比的还有 Aptana RadRails 1.0，CodeGear 3rdRail 1.1 以及 TextMate 1.5.7 等同样具备优秀特性的 Rails 开发 IDE。

动态语言 2008 年第 07 期——Rails 2.1 与 Python 科学计算工具包

Ruby on Rails 2.1 已在六月初正式发布，在波特兰举办的 RailsConf 2008 上，Rails 核心开发团队成员 Jeremy Kemper 在演讲结束时首先公布了这一消息。Rails 2.1 是即六个月前 Rails 2.0 版本发布后的最重要的一次升级更新，发布版包含 1600 多处修补和升级。

在 Rails 2.1 版本提供的重要新特性中，包括有 Geoff Buesing 贡献的时区支持功能，ActiveRecord 的部分更新及脏数据追踪检查，Gem 包依赖描述，由 Rails 插件 has_finder 转换而来的 Rails 2.x 新功能 named_scope。除此之外，Rails 2.1 还提供了基于 UTC 的迁移功能和更好的缓存机制。

关于 Rails 2.1 的更多细节，可以查看站点 ryandaigle.com 上的详细说明，同时站点 railscasts.com 也更新了相应的 WebCast 视频。这些可以帮助我们在更新 Rails 版本之前先对新版本所做的改进有个迅速的认识。

Ruby 方面，Ruby 1.8.7 和 JRuby 1.1.2 也相继发布。在 Ruby 1.8.7 发布版中，部分新特性是源自于 Ruby 1.9 的向下移植，同时在 Ruby 解释器方面也做了性能提升。JRuby 1.1.2 在之前版本 Bug 修复的基础上，着力于提升 JRuby 在 JVM 上的运行性能，比方说将 JRuby.jar 转移到根类路径下，来节省字节码校验时间并提升加载速度，同时新版 JRuby 在函数调用、线程和 YAML 解析方面都进行了相应改进。

同样在 RailsConf 2008 上，EngineYard 创始人 Ezra 展示了面向分布式实时应用开发的下一代云计算平台 Vertebra。Vertebra 基于 Erlang 与 Ruby 构建，系统的核心如安全/发现机制、状态机 workflow 和分布式键值存储系统由 Erlang 编写，代理框架和实际代理由 Ruby 编写。目前 Vertebra 尚未正式发布，这是否会成为一个构建和整合众多独立服务的强大平台，还需等待用户的检验。

面向科学计算的第七届 SciPy 2008 大会即将于 8 月份在加利福尼亚召开，SciPy 是 Python 语言撰写的科学与工程计算工具包，可以处理向量、矩阵、二维图形和表格等，应用面向统计分析，数值积分，常微分方程求解，梯度优化，以及通过 C++ 拓展实现并行编程。Python 语言有众多开源的科学计算辅助工具包，除 SciPy 之外，还有面向基础数据运算的 NumPy 和用于二维绘图的 Matplotlib 等优秀工具。

动态语言 2008 年第 06 期——Python 与 Rails 发布版稳步推进

Python 语言成为 Linux 杂志近期评选的“读者最喜爱的编程语言”，与此同时，Python 的发布版一直以稳健的步伐向前迈进。本月 Python 2.6 的 alpha 3 版本以及 Python 3.0 的 alpha 5 版本发布。在 Python 3.0a5 版本中，修正了执行特定函数时出现的 SystemError 异常，以及命令行指令以 Unicode 格式传递字符串时出现的问题。

尽管目前 Python 3000 的发展状况良好，但 Guido 及 Python 开发团队一直还在致力于 Python 2.6 的开发与完善。在 Python 2.6a3 中，为 Jython、IronPython 以及其他 Python 版本实现在使用时保留了 -J 和 -X 等参数，并且修正了八进制接收时引发的 SyntaxError 异常。

正如 Guido 在 3 月份一次题为“Python 3000 and You”的演讲中提到，Python 2.6 一直是十分稳定的发布版，并且兼容性很好，并且许多 Python 3.0 的特性也被添加进入 2.6 之中。Python 这两个并行的版本将会保持每月一个小发布版的速度持续更新，Python 2.6 以及 3.0 的最终版本将于 08 年 8 月份一同发布。

Rails 框架自诞生以来，一直保持着快速的发展势头，开发者在从其特性更新中享受新增功能的同时，也需要及时适应并学习新的改动。Rails 2.1 的预览版本已经发布，新版本中包含显著的特性改进，如以 named_scope 命名集成了广受好评的 HasFinder 插件，还添加了较之 Hibernate 更为简洁、强大的对象变更检查机制，并在此之上加入仅更新对象变更属性的特性。

有意思的是，Rails 2.1 RC1 版本更新消息最先是 DHH 通过他的 Twitter 地址 (<http://twitter.com/d2h/>) 发布的，这个 Rails 开发的 Web 2.0 应用同时也成为了解 Rails 最新消息的便捷途径。DHH 在 Twitter 中提到“Rails 2.1 RC1 版本的 Gem 包已经提交到服务器上，正式的声明稍后会在官方站点发布，希望开发者现在就去尝试并使用这个新的版本。”在位于 GitHub 上的 Rails 存储库 (<http://github.com/rails/rails/tree/master>) 中，也可以获取 Rails 最新的源代码。

部署方面，旨在从 Apache 服务器上更便捷部署 Rails 项目的 mod_rails (Passenger) 成为近期热点，并且 mod_rails 在 Rails 官方博客上受到推荐，同时知名的主机提供商 DreamHost 也传出准备采用 mod_rails 对 Rails 部署提供支持的消息。mod_rails 可以帮助开发者像部署 PHP 一样去部署 Rails，并且在运行多个 Rails 进程时，mod_rails 对于内存占用过多的问题进行了优化。在站点 modrails.com 上，为开发者提供了一个使用 mod_rails 的快速上手视频教程，感兴趣的开发者不妨花几分钟来尝试一下。

动态语言 2008 年第 05 期——动态语言的虚拟机平台

在芝加哥举办的 PyCon 2008 大会上，Python 之父 Guido 为开发者带来了 Python 3000 及其新特性的主题演讲，并为开发者如何适应语言的新特性，以及迎接 Python 3000 应该做哪些准备提供了建议。

值得注意的是，同样在 PyCon 上，来自微软的动态语言运行时 DLR 首席架构师 Jim Hugunin，为开发者演示了 Django 在 .Net 平台、SQL Server 2005 和最新的 IronPython 2.0 Beta 1 上的运行情况，以展示微软 DLR 团队所取得的进展。为了使 Django 支持 SQL Server 2005 数据库，并且可以正确地处理 Unicode，仅对 Django 源代码做了几处修改就使其顺利运行在 IronPython 之上。

Python 语言实现的快速 Web 开发框架 Django 自带了模板系统、对象关系映射以及动态管理界面，目前已是 Python 领域最著名的 Web 开发框架之一。来自于 IBM 多伦多软件实验室的 Antonio Cangiano 统计了 Django 框架相关的 IRC 讨论组及 Google 用户组数量以及 TIOBE 指数等有趣的数据，将其与 Ruby on Rails 进行了一个对比。Antonio 认为，随着运行在 .NET 和 JVM 上的 Python 实现逐步成熟，Django 框架的开发应用将会迎来一个快速增长的时期。同时，商业推进系统公司 BPS 的首席技术官 Gavin Terrill 也在 InfoQ 站点撰文对 Django 是否将达到引爆点提出了自己的看法。

在 Ruby on Rails 2.1 版本发布前夕，Rails 开发团队将代码存储库从 Subversion 迁移到 Git 之上。DHH 在 Rails 的官方博客中写到：“我们准备将 Rails 开发的代码存储库从 Subversion 迁移到 Git 之上，并且会使用 Github 来帮助开发者进行协作，与此同时，我们的问题跟踪系统也将转移到基于 Rails 开发的 Lighthouse 应用之上，于是到目前为止，我们的存储库和问题跟踪系统都是使用的 Rails 应用，这对我们是一个很好的鼓励。”包括 Rails 在内，目前已有众多开源项目迁移到 Git 上来进行源代码版本管理，比如 Rubinius、Merb、Wine、Beryl 等。

作为 Aptana IDE 的一部分，RadRails 目前发布了 1.0 版本。在 Aptana 官方提供的性能比较中，较之 Netbeans 和 CodeGears 等 Ruby IDE 工具，RadRails 在重构和性能剖析方面更胜一筹。

Sun 的 Ruby 技术中心 (<http://developers.sun.com/ruby/>) 也于近期上线，从 NetBeans 的 Ruby IDE 到 Ruby 技术中心上线，可以看出 Sun 公司加大了对 Ruby 等动态语言的支持力度，同时也使得 JRuby 逐渐为开发者所熟悉和认同。

在 Java 与 JVM 不断受到其他新兴语言和平台挑战的今天，SUN 的虚拟机在动态语言方面的实践如：JRuby、Jython、Groovy、Scala 等，对于 Java 语言、开发社区和 SUN 公司本身的发展都具有着积极的意义，这些努力将推动着 JVM 未来发展成为真正的多语言虚拟机平台，而非仅仅为 Java 语言服务。

动态语言 2008 年第 04 期——SaaS 模式的 Rails 开发平台

如果你还没有找到开发和部署 Rails 应用的最佳环境，不妨尝试一下 Heroku 站点 (<http://heroku.com/>) 为 Rails 开发者提供的在线 RoR 开发与部署环境。在 Heroku 之上，用户仅通过浏览器就可以在线完成 Rails 应用从开发到部署的各个环节。Heroku 为用户提供了集编码与调试于一身的在线 Rails 集成开发环境，以及完善的 Ruby 和 Rails 运行环境，同时开发者还可以把本地开发的 Rails 应用导入并部署在 Heroku 之上，甚至可以使用 Amazon 提供的服务进行弹性拓展。

由于 Heroku 提供 Rails 的开发部署环境无需安装与配置，很适合作为开发初期为客户提供应用展示的工具。有开发者在 Heroku 的官方网站上对其评价道：“Heroku 是一个极有前途的项目，我无法想象 Ajax 打造的 Web IDE 可以运行的如此流畅。”无疑，通过 SaaS 的方式为开发者提供在线的 IDE 开发调试环境，将成为未来 SaaS 应用新的亮点。

苹果公司也在 MacOSForge 上增加了新的开源项目 MacRuby (<http://trac.macosforge.org/projects/ruby/wiki/MacRuby>)，为 Ruby 1.9 提供一个 Objective-C 的移植版，来提升 Mac OS X 上 Ruby 的运行表现，并使开发者可以通过 Ruby 写出非常棒的 Cocoa 应用。目前 MacRuby 项目的参与者还都是苹果公司的开发者，但同时也欢迎对此有兴趣的开发者参与到 MacOSForge 上的项目中来。

Python 方面，本月 Python 2.3.7 和 Python 2.4.5 的最终版本都已发布，但这两个发布版本只提供源代码，并没有释放编译后的二进制版本。Windows 以及 Mac OS X 的用户目前还无法通过本地编译源代码的方式来使用这个发布版本，如果需要二进制的发行版，可以使用先前发布的 Python 2.3.5 和 Python 2.4.4 版本。

由 Matrix、ThoughtWorks、InfoQ 中文站、BJUG 等合作举办，代号为“春归”的第二次 Open-Party 活动也在本月举行。在参与者提交的十三个演讲主题中，开源、创业、敏捷实践和动态语言方面的主题受到了参与者们的广泛关注与好评。为主题演讲准备的技术交流资料和讲义可以在官方站点 <http://www.beijing-open-party.org/> 上下载获取。

根据薪金调查机构 PayScale.com 的统计数据显示，RoR 开发者的年薪与 Java 开发者的年薪在美国已经相当接近，并且当开发者从业时间和经验较少时，RoR 开发者的薪水要比 Java 开发者的薪水高出一个区间，这项调查的结果同时也在侧面反映了两种开发技术在学习曲线上的差异。

动态语言 2008 年第 03 期

根据 TIOBE 给出的编程语言流行度评估报告, Python 语言获得 2007 年的年度编程语言称号。Python 3000 目前的版本是 3.0a2, 根据官方制定的开发计划, Python 3000 将在本年度八月份发布最终的稳定版本。随着 3.0 正式版发布日期的临近, Python 语言的使用比率在 08 年还有很大持续上升空间。

在 TIOBE 给出的 2 月份编程语言排名中, 脚本语言 Lua 目前已经进入排行榜的前 20 位。几乎专为嵌入其他语言使用以及游戏脚本编写而设计的脚本语言 Lua, 逐渐受到越来越多开发者的关注, 也意味着开发者倾向于使用更多可选的途径, 以最为适合的方式来解决为题, 而并非仅把创造力局限在少数几种主流语言当中。来越多开发者的关注, 也意味着开发者倾向于使用更多可选的途径, 以最为适合的方式来解决为题, 而并非仅把创造力局限在少数几种主流语言当中。

JRuby 项目的 1.1RC2 版本于本月发布, 这个版本对先前的 RC1 进行了 200 多处修改, 同时对大规模 IO 的实现进行了重构, 并对 JIT 即时编译的内存占用情况进行了改善。JRuby 1.1 的目标致力于解释效率的增强和更为精巧完善的设计, 其最终版本将指日可待。

开源项目 Rub.NET 为 Ruby 代码在 .NET 平台之上运行提供了实现。但在最近, Ruby.NET 项目邮件组收到项目领导人 Wayne Kelly 博士一封名为“Ruby.NET 项目未来”的信件中, 其中写道: 在经历 3 年多的发展历程之后, Rub.NET 项目将停止后续的更新计划。

由于 Rub.NET 与 IronRuby 本身为 .NET 平台上两个相似的 Ruby 实现, IronRuby 基于动态语言运行时 DLR 工作, 而 Ruby.NET 基于通用语言运行时 CLR 工作。Wayne 认为 DLR 更有发展潜力, 并且 IronRuby 今后将会对 Silverlight 技术进行支持。在 Ruby.NET 项目暂停之后, Wayne 本人将集中精力, 专注于 IronRuby 项目的开发。

首届 RubyFools 大会将于 4 月初在 Rails 的诞生地丹麦首都哥本哈根举行, 大会为期两日, 主题演讲者包括 Ruby 创始人 Matz, Rails 畅销书作者 Dave Thomas 等众多为 Ruby 开发者所熟悉的名字。议题中包含 Rubinius、Ruby 中的 REST、Ruby.NET、AWS 上使用 Rails、Merb 开发框架、RSpec 等众多开发者感兴趣的内容。

动态语言 2008 年第 02 期

伴随 Ruby 1.9 的正式发布，相关的文档与图书支持也在不断跟进，帮助开发者迁移到新的 Ruby 平台之上。著名的“镐头书”《Programming Ruby》宣布了第三版的出版计划，Pragmatic 出版机构预计将在本年度的 5 月份将第三版上市，不过按照惯例，本书已经给出了 Beta 版样例供给读者下载阅读。

Ruby 的版本一直遵循一个个默认规则，就是如果版本号末位是奇数，表示该 Ruby 版本为试验版本，末位为偶数则表示为正式版本。比如 Ruby 1.7 是试验版本，而 Ruby 1.8 则为正式版本。由于 Ruby 开发团队认为 Ruby 1.9 已经较为完善，所以版本号命名为 1.9.0。意味着 Ruby 1.9 作为正式版本发布，虽然进入实际产品应用尚需时日，但 Matz 表示会继续改进 1.9 版本的稳定性，并且包括 Rails 在内的主要 Ruby 项目，已经着手进行 Ruby 1.9 的兼容工作。但仍旧有部分开发者表示，还需要等待其核心代码经过充分测试并且稳定之后，才会考虑完全转移到 Ruby 1.9 之下。

当前的 Rails 2.0.2 版本中，默认的数据库已经由 MySQL 转换为 SQLite。这是由于 SQLite 的使用比 MySQL 更为简便，同时国外大多数开发者在 Mac OS X 系统中进行 Rails 开发，并且 10.5 系统 Leopard 中已经很好的集成了 SQLite 数据库，方便开发者随时使用。如果希望 Rails 应用依然使用 MySQL 数据库，可以在应用创建时使用参数 `rails -d mysql` 来指定。

为 Rails 应用提供模板支持的 Haml 项目本月也发布了 1.8.0 版本。由于 Rails 2.0.1+ 引入了新的模板引擎，所以 Haml 也由不久前发布的 1.7.x 升级到 1.8.0，来适应 Rails 新的变化。Haml 致力于帮助开发者摒弃 Rails 默认 HTML/ERb 格式的代码，而以优雅整洁的缩进代码代替。Haml 的作者认为其并非是一种反叛，而是一种最为自然的 XHTML 构造方式。

Erlang 方面，TSS 在本月发布了来自于 ThoughtWorks 的 Dennis Byrne 撰写的《整合 Java 与 Erlang》一文。文中指出 Java 与 Erlang 并非相互排斥，反而是两者可以优势互补，通过 Jinterface 可以很好的将两者整合在一起，来解决业务逻辑相对复杂的问题。

尽管 Python 框架 Django 依然停留在 0.96 版，但为 Django 提供开发指南的官方出版物《Django Book》已经撰写完成，印刷版将在年初上市，值得一提的是，这本书使用了 GNU 的开放文档协议授权，读者可以在站点 djangobook.com 免费阅读全部内容。

动态语言 2008 年第 01 期

对于 Ruby 语言的开发者来说, 本月的开发者社区里无疑充满着令人兴奋的事件。Ruby on Rails 在近一年的期待之后, 终于迎来了 Rails 2.0 版本的最终发布。对于 2.0 的发布, Rails 创始人 DHH 也在其博客 loudthinking.com 中撰文表达了兴奋和激动之情, 并表示会在近期内把那些曾经鼓舞众多开发者尝试 Rails 的视频教程, 使用新版 Rails 2.0 重新录制。

在 Rails 2.0 之中, 对 URL 路由资源添加了命名空间支持, 并将 Session 存储机制默认定义为基于 Cookie 实现, 同时在 Action Pack 中为应用提供了统一的异常处理机制。Active Record 方面, 除了增加查询缓存来提升性能, 还简化了数据迁移脚本的书写方式, 使其更为清晰条理。值得一提的还有, Rails 2.0 中增加了应用调试器, 支持断点设置和单步跟踪等功能, 同时将各项 plugin 的管理变得更为人性化。

从年初的 Rails 1.2 发布到新版的 Rails 2.0, Rails 本身一直处在不停歇的发展完善之中, 完善和改进体现在框架的各个层面。与此同时, 针对 Rails 的拓展支持也越来越丰富, 替代 Rails 模板的 Haml 项目也发布了 1.7 版本。Haml 可以使得开发者摒弃 Rails 默认的 HTML/ERb 格式的代码, 而以优雅整洁的缩进代码代替。Haml 的作者认为其并非是一种反叛, 而是一种最为自然的 XHTML 构造方式。

Ruby 语言的下一个重大版本 Ruby 1.9 也已经整装待发。尽管 Ruby 1.9 较之上个版本有了大幅度的改进, 但仍旧不能完全代表未来 Ruby 2.0 发布时的模样。在 SVN 存储库的 HEAD 版本中, Ruby 开发团队正在不断尝试一些新奇的改进和完善。当然, Ruby 1.9 在性能方面的提升是开发者最为关心的问题, 一份来自于 Ruby 开发者的测试报告说明, 在对 Ruby 1.9 版本、Ruby 1.86 版本、以及 JRuby、XRuby、Ruby.Net、Rubinus 等 Ruby 实现的性能差异测试上看, Ruby 1.9 在运行时间上要远远少于其他版本。同时也在 Mac OS X 10.5 系统下, 使用斐波那契递归函数对 Ruby 1.9.0 版本、Ruby 1.8.6 版本以及 Python 2.5.1 进行了递归与函数调用测试。相同循环次数中, Ruby 1.8.6 用时 158.869 秒、Python 2.5.1 用时 31.507 秒, 而 Ruby 1.9.0 仅用时 11.934 秒。从以上数据看, 即将到来的新版 Ruby 1.9.0 在程序解释运行效率上有了显著提升。

开发工具方面, 随着 Netbeans 6.0 在本月初正式发布, 在一年多的开发过程中, Netbeans 6.0 在 5.5 版的基础上做出了重大改进。NetBeans 相应的 Ruby IDE 也更新到 6.0 版本, 提供了对 Ruby/JRuby/Ruby on Rails 的默认支持和强有力的 Ruby 调试环境, 并且可以在 Ruby 代码重构的同时进行快速修补。由官方站点 netbeans.org 发布的 NetBeans 杂志, 在 12 月份也推出了 Ruby IDE 专刊, 内容包含使用 NetBeans 和 Rails 创建动态站点, 以及在 NetBeans 环境下创建 REST 方式的 Web Service 等方面的文章和教程。如果您还在为选择 Ruby IDE 而举棋不定, 开放源代码的全新 NetBeans 6.0 无疑是个不错的选择。

Python 3000 在本月发布了第二个 Alpha 版本。根据 Python 官方公布的发布计划, 进入到 08 年, 将会发布一系列 Python 3000 的 Beta 版本, 之后将在 08 年的 8 月份发布最终版本的 Python 3000。对于 Python 的发展方向, Python 之父 Guido 曾谈到, 保持语言的简单性是 Python 一直强调的重点, 未来的 Python 3000 将去除一些华而不实的功能, 取而代之的会是简洁和优雅的新特征。

动态语言 2007 年第 12 期

Ruby on Rails 2.0 的 RC1 版本已经在本月 9 号释出。在 RC1 的官方发布说明中，建议 Rails 使用者先升级到当前的稳定版本 1.2.5，修正运行中出现的警告之后，再考虑升级到目前最新的 Rails 2.0 RC1 版本。即将到来的 Rails 2.0 正式版中，框架特性的优化、REST 风格的 Web 服务、以及改善的数据迁移和 Active Record，将把开发者手中的这把利器打磨的更加锋利。

在国内 Rails 社区中，相关的专业培训也开始慢慢起步。ChinaonRails 组织的免费公开课在本月付诸实施，这是针对 Rails 爱好者提供的较有特色的短期训练项目。在课程中，参与者划分成数个开发小组，并在 Google Code 之上建立小组的开源项目，课程进行的同时也将不断提交完善相应的项目开发代码。在这样高效沟通、互动促进的实战环境下进行 Rails 开发训练，所花费的时间和精力都将低于开发者单独自学的开销。

Ruby 方面，继月初的 RubyConf 2007 之后，Ruby 欧洲开发者大会 Euruko 2007 也在本月中旬召开。与此同时，Ruby 语言的开发组依然活跃，但 Ruby 社区在国际化沟通方面的问题依然存在。本月 InfoQ 上的一篇文章指出，Ruby 创建者 Matz 以及开发团队中的大部分成员都来自日本，所以 Ruby 语言开发相关的讨论，大多都在日文 ruby-dev 邮件列表中展开，这使得日本开发团队与国际化 Ruby 社区间的语言障碍依然存在，并且这样的障碍将阻止熟悉英语的开发者参与到 Ruby 开发团队的讨论沟通之中。

JRuby 1.1 的 Beta1 版本也在本月发布，1.1 版本在 JRuby 性能方面实现了大幅度提升。在 JRuby 性能测试报告中，部分情况下 JRuby 的表现要比原生 Ruby 解析器更为高效，并且 Ruby 代码已经完全可以编译在 JVM 的 AOT(Ahead Of Time)以及 JIT(Just In Time)模式下运行。

Python 语言方面，Web 开发框架 TurboGears 目前最新 Beta 版本是 1.0.4b2。新版 TurboGears 在事务处理能力和分页方面进行改善的同时，实现了对于 SQLAlchemy 0.4 最终版本的支持，借助其来完成数据库实体关系映射。另一款 Web 开发框架 Django 在 0.96 版本之上，又增加了安全性方面的修补，使得攻击者无法通过先前 Django 框架预处理 cache 方面的漏洞对系统造成安全威胁。Python 社区方面，继 PyCon 2007 结束之后，PyCon 2008 将于明年在芝加哥进行，目前会议的讨论提议征集已经向 Python 使用者们开放。

伴随多核计算时代的到来，Erlang 语言在本年度引起了开发者的广泛关注。《设计模式》一书的作者之一 Ralph Johnson，曾在博客中撰文预言 Erlang 将会是多核计算时代的 Java 语言。Erlang/OTP 目前的稳定版本是 R11B，在这个版本中，最为重要的改进是提供对于均衡多处理技术的支持，在双处理器、超线程以及多核技术应用推广的趋势下，均衡多处理技术将帮助 Erlang 进一步的利用多核技术的所带来的性能提升，对于 Erlang 程序编写者来说，这并不会增加额外的负担。

结构化的动态类型编程语言 Erlang 内建了并行计算支持，采用基于函数式的编程方法，适于构建分布式系统。Erlang 中文社区为推进 Erlang 在中国的普及应用，一直做着不懈的努力，在线上交流的同时也组织了 Erlang 爱好者定期的研讨与聚会活动。作为成熟稳定且适合高伸缩性系统开发的语言，在更多成功案例积累和强有力公司支持之下，必将会有更为长远的发展前景。

动态语言 2007 年第 11 期

伴随 Rails 框架与敏捷开发思想的广泛普及, Ruby on Rails 自身的发展与完善也一直没有停息脚步。Rails 1.2.4 版本在本月发布, 新的版本包含安全方面的修补和部分性能优化, 并在官方发布通告中建议之前版本的用户更新升级。与此同时, 十月初 Rails 框架迎来了 2.0 测试版的发布。当然, 在 Rails 2.0 正式版发布前, 必然会经历数个 Beta 版本, 但提前了解一些 Rails 2.0 的新特性将必然对其应用前景信心大增。

在即将推出的新版 Rails 2.0 中, 新增与改进的特性包含简化的数据迁移脚本、更好的 REST 功能支持、新的断点调试支持、更轻松的插件配置、内置的 Rails 缓存、增强的静态资源管理等众多相关改进。Rails 框架一直以其良好的设计和高效的开发效率而为人称道, 新版 Rails 2.0 的推出在易用性和功能方面有了不少的改进, 随着框架与开发者的进一步贴近, 可以预见 Rails 将为基于动态语言的 Web 开发带来新一轮热潮。

在动态语言 IDE 工具方面, CodeGear 新近推出的 Rails 开发工具 3rdRail 吸引了众多开发者的关注。笔者也下载了 3rdRail 的试用版, 在初步试用中体验到 3rdRail 的代码补全、错误检查、Rails 脚本命令都非常实用, CodeGear 在 IDE 开发方面积累的大量经验已体现在这款新的动态语言开发工具之上。应该说 3rdRail 是在 NetBeans Ruby IDE 之后一款不可多得的 Ruby 和 Rails 开发工具, 较之同样基于 Eclipse 的 RadRails 毫不逊色。值得一提的是, CodeGear 为 3rdRail 提供了完善的联机帮助和开发指南, 对于迅速掌握 3rdRail 开发技巧有着极大的帮助作用。

Ruby 到 Java 字节码的编译器 JRuby 也发布了 1.0.1 版本, 在 1.0 稳定版基础之上做了一定修补, 重点放在编译器兼容性的提升之上, 下一个版本 JRuby 1.1 的发布预计会在 11 月初举行的 RubyConf 2007 大会期间。

Python 实现的 Web 开发框架 Django 依然停留在 0.96 版本, 1.0 版本在支持者的期待中仍然没有露面, 但使用 Django 开发的成功案例却在不断增多。

让我们再把目光转向老牌的 PHP 开发, 在 Web 应用开发方面, 尽管群雄并起, 但 PHP 仍然是目前占据最大市场份额的开发语言。新版的 PHP 5.2.4 已经发布, 修正了部分低优先级安全漏洞, 以及 120 个错误, 使得 PHP 5.2.X 版本的稳定性得到有效的增强。在官方声明中, 鼓励所有 PHP 用户都升级到新的版本。

有趣的是, 本月一篇发表在 O'Reilly 站点上, 名为“两年 RoR 开发付之东流, 重返 PHP 的七大理由”的文章引起了众多开发者关注。作者 Derek Sivers 在文中总结了放弃 Rails, 重新转回到 PHP 的七大原因, 如系统间整合、使用习惯等。当然, Derek 并未否定 Rails 框架为开发者带来的益处, 并对 Rails 高效的模板机制、Active Record 数据模型类封装等独具特色的功能加以称赞。

对于敏捷 Web 开发, 速度决定一切。PHP 无疑简单易用, 而 Rails 在保证开发速度的同时提供了优良的设计, 选择永远是仁者见仁、智者见智的事情, 相比较而言, 或许开发技巧的积累和经验的沉淀才是最为关键的。

亲爱的动态语言爱好者, 很高兴你可以耐心翻到文档的最后一页
希望和细心的你保持联系, 我的邮件是: tomgaoang@gmail.com
关于我的更多文字, 请参见这个链接, <http://www.gaoang.com/publication>

祝一切好 ☺