Atitit 编程与软件的 趋势艾提拉总结

目录

[未来编程预测No.2：数据库逐渐可以完成复杂的分析 1](#_Toc30980)

[未来编程预测No.3：JavaScript是一切 2](#_Toc7468)

[关于编程语言未来的预测8：插件将取代完备的程序 3](#_Toc5052)

[未来编程预测No.11：外包和内包将继续陷入僵局 3](#_Toc23978)

# **未来编程预测No.2：数据库逐渐可以完成复杂的分析**

程序员一旦把对数据的控制权让给一个叫“数据库”的东西，那么，他们就不再是“程序员”，而是“数据库用户”。这也不是一件坏事。毕竟，数据库在处理数据上被设计成比一般的程序员更高效的，尤其是在时间紧急的情况下。他们也（通常）非常有效地使用RAM和处理复杂的分级存储器体系。把数据导到数据库，让数据库做自己的事，这往往不是最聪明的处理数据的解决方案，而是要让数据库在时间方面变得更智能才是聪明的做法。

这是因为未来的数据库一定不仅仅是存储数据。许多数据库系统已经内置了成熟的报表引擎（又称“商业智能”），能够运行更加复杂多变的算法，能够分析数据、在数据模式下更有效地搜索数据，以及做更多目前被吹捧的“大数据”相关工作。

这个的能力和复杂度将由移动数据的成本所驱动。单纯地从数据库中提取信息，并将其移交给一个独立的“大数据”包将变得越来越耗时，而且需要更多的编码。把数据库中的数据交给它引擎来处理分析将会快很多，因为它不但会限制通信的开销，而且还减少了程序需要从数据存储中提取数值的总数。

# **未来编程预测No.3：JavaScript是一切**

JavaScript在未来的编程语言世界当然不会是唯一的语言，特别是考虑到Cobol程序员的招聘广告还在热招中，但它必定会这样的苗头。JavaScript是浏览器中必不可少的，现在几乎所有的客户端计算机中都占有主导地位。在服务器端则使用像Node.js的工具。

JavaScript无疑也将会在其他领域上更具有优势。要进军智能手机的领域唯一的方式就是使用制造商所要求的本地语言来编写代码：iPhone的是Objective-C，Android的是Java，微软的是C#。现在大多数的手机程序员可以把HTML5的应用程序通过类似浏览器窗口表现出来。这样的效果或许没有本地代码那么敏捷方便，但是对于Web来说，JavaScript也是不错的，很轻便的语言。

浏览器不只是植入手机内，它是在吞噬整个平台。Chrome OS和Chromebooks会定期对废弃的操作系统做处理。当JavaScript和浏览器能做任何事情的时候，又何必担心那些事情呢？

大型机将主要用Cobol，生物学家们将可能继续做Python，Linux仍用C，但是几乎所有的一切都是一场公平的游戏，如同JavaScript吞噬整个世界。

# **关于编程语言未来的预测8：插件将取代完备的程序**

基础的网络应用将不只是可以嵌入到大型的架构中的那些代码段的那些应用。Phontoshop作为对图片进行再处理的重要的工具，一定程度上是因为丰富的插件。现在类似MagicHour这样新的应用使得插件更加的简便。例如MagicHour的用户可以像过去分享图片一样分享过滤器。绝大多数的平台都提供了良好的插件API，很多的平台都有成千上万的模块、类库和插件等构成的繁荣的生态系统。

这些新增代码的生态系统意味着程序员将写出更多的片段和更小的应用。适量的接口代码可以复用无数次，这要比写一个庞大的、手写的MB量级大小的应用要有用的多。小的片段可以与整个生态系统中的各片段达成一致，而大的应用就需要在内部处理所有的事情。

有悟性的程序员将会学着创建插件，而不是整个项目。他们将要学习宿主平台的API并把各部分穿连起来。他们几乎不再需要从头开始创建应用。 我们都将成为这个新生世界的一部分。

# **未来编程预测No.11：外包和内包将继续陷入僵局**

一个标准的预测，所有的编程工作将流向工资最廉价的国家

这个大胆的经济学定律是由预测反映了低端的工作会由自动化工具所取代，而真正的工作将是为那些理解业务的人提供。

双方将继续为真实。外包队伍将赢得与低投标工作，但通过新的自动化工具被挤压。一组网站，可以很容易地在全球范围内雇佣低工资工人。另一组的网站是使它更容易，更容易从0到60不雇用人的团队。依赖于外包合同小队会发现自己对招投标的谁知道业务，并不需要帮助的人的内部团队

## Smalltalk 正是一种考虑了“与编辑器、调试器、性能分析器等开发工具相互配合”而设计的语言

近年来，编程语言似乎越来越难以脱离IDE（Integrated Development Environment，集成开发环境）而单独拿出来说了。对于Ruby也总有人问“没有IDE吗？”之类的问题，当然，好消息是最近Eclipse 和NetBeans 已经支持Ruby了。

有点跑题了。总之，未来的编程语言可能不会像过去的编程语言那样，让语言本身单独存在，而是和编辑器、调试器、性能分析器等开发工具相互配合，以达到提高整体生产效率的目的。话说，那不就是 Smalltalk 吗8？

8 Smalltalk 正是一种考虑了“与编辑器、调试器、性能分析器等开发工具相互配合”而设计的语言。过去它并不能算是成功的，但随着技术的进步，其理念获得了越来越多的用武之地，或许真能卷土重来也未可知。

转载：关于编程语言未来的 12 个预测 - 只为分享 - CSDN博客.mhtml

未来的编程语言究竟是什么样？-图灵社区.mhtml