

# 未来的应用程序

下文介绍菜鸟教程版本的有关 未来 Web 应用程序。

## 可执行文件将消亡，JavaScript 将存活

编译的可执行文件（如 C 或 Java 语言的编译）不能在不同的硬件上运行。

可执行文件（EXE 文件、ActiveX 和 COM 对象、DLL 文件）是防止运行在互联网上的应用程序发展的组件。

未来的应用程序将无法使用，或依靠，安装在客户端计算机上的组件。

### 我们的建议：

仅使用 HTML、CSS 和 JavaScript 来编写您未来的应用程序。

确保您未来的应用程序可以运行在任何的 Web 浏览器中。

## Web 应用程序将成为互联网服务

过去的应用程序都是大型的、为特定目的创建的应用程序。它们中的大多数很快将会消亡，因为它们不能随着需求的变更提供服务。

应用程序应该是灵活的、通用的、能在不被破坏的情况下很好地适应需求的变更。

应用程序应该能够扩展，从支持每天数以百万计的请求。

应用程序应该能够在不被破坏的情况下，在多个服务器之间进行传播和移动。

应用程序应该能够与其它的应用程序共同使用。

应用程序不应该包含大量的代码。应用程序应该被分解成更小的服务，以便易于创建和易于维护。

应用程序是可返回数据到提交的互联网请求的一系列互联网服务。

应用程序应该在不保持到服务器的永久连接的情况下通过标准的互联网协议请求服务。

### 我们的建议：

使用基于 SOA（Service Oriented Architecture 面向服务架构）的互联网来编写您未来的应用程序。

让您的应用程序服务更通用，更灵活，且能服务于不同类型的请求。

## 未来的应用程序将很容易创建和编辑

客户端和服务端将在一个易于理解的方式进行数据交换。

如果能够避免，应用程序将不被编码。

应用程序将通过编辑模型而不是编辑代码来创建和修改。

应用程序描述将被人类可读。

应用程序描述将具有自我描述性。

应用程序将由用户编写，而不是由程序员编写。

### 我们的建议：

使用人类可读的文本文件来描述服务，并通过执行这些描述来提供服务。

使用文本文件（如 XML 文件）来描述应用程序。

使用文本文件（如 XML 文件）来进行数据交换。

使用 HTML、CSS 和 JavaScript 来执行应用程序。

## 三个小的 Web 开发人员...

很久以前，有三个小的 Web 开发人员，开发一个新的 Web 站点。

1. 第一个 Web 开发人员使用 <AppML>。
2. 第二个 Web 开发人员使用他最喜欢的服务器端编程语言。
3. 第三是使用专业的企业 Web 开发框架。

第一个 Web 开发人员在两天内完成演示。经过与用户的合作，一周内完成了初步原型。经过两周的测试，一个智能的、快速的和易于使用的网站准备发布。

6 个月后，第二个 Web 开发人员准备好了他的网站。但是 WWW 已经改变了需求，所以对他的网站并不满意。这个 Web 开发人员无法对他的项目做出重大的改变，因为它包含了太多的代码。于是，他开始第 2 版的开发。

第三个 Web 开发人员从来没有成功地完成他的工作。那个专业的 Web 开发框架，使用起来很困难，很难理解，几乎是不可能的测试。

[看一看第一个开发人员是如何做到的。](#)

← AppML 案例 Products

 点我分享笔记