**译者序**

最近二十年，互联网、移动互联网、大数据、云计算、智能手机都有了长足的发展，互联网应用已经深入渗透到人们生产生活的各个领域各个方面与各个环节。浏览器因其使用简单方便、表现力強、跨平台而成为网络应用人机交互的主流界面之一。

网页与浏览器发明的时候是用于欧洲核子中心检索浏览研究资料的，早期的网页交互性不强，通过导航切换到下一页来呈现不同的信息，因网络带宽的延时而缺乏桌面程序实时响应的体验，因此并不适合交互式的应用。浏览器因此发展了JavaScript 与文档对象模型(DOM)，提供浏览器端对网页内容的实时处理能力，也就是动态网页，不需要切换网页通过慢速网络从服务器重新下载。另一方面也发展了cookie、session 、用户身份验证等各种方式来在浏览器与服务器两端分别记住网络应用的状态，基于浏览器的交互式网络应用于是出现并迅速发展，这就是B/S应用模式。

大约二十年前，AJAX技术开始出现并流行，浏览器可以在后台通过xmlhttp 协议加载数据，响应用户的操作动态调整网页上展示的数据，像Google 等自此提供了即时搜索建议等服务，基于浏览器的互联网应用，其交互式体验有了质的飞跃。

这二十年间，基于JavaScript 的各种浏览器端的应用框架和UI控件得到了长足的发展，浏览器UI现在具有了传统的C与Java桌面程序一样的可视化效果和交互响应能力。然后网络带宽的迅速提高和WebSocket 协议等的出现，令浏览器与服务器之间在后台快速传输大量数据成为了可能。浏览器也在迭代进化，更好用易用了，现在已经成为了具有良好交互能力与丰富表现力的数据应用客户端。

然而浏览器与网页及JavaScript 的设计之初，是轻量级的瘦客户端，是小数据的应用场景，它们并不具备处理大数据的能力。随着互联网应用的迅速发展和普及，数据出现了爆炸式的增长，大数据、云计算技术应运而生，最近十几年也得到了长足的发展，人工智能在大数据的加持下也迎来了繁荣(本质上基于概率论与统计学，所以须有大数据作基础)。在大数据与人工智能领域，人们要方便的进行数据的批量处理，分布式处理，作集合运算，矢量运算，张量运算等，这些都是算力集中型的重活。所以大数据平台的底座框架，一般用Java和C，稳定高效，跨平台可移植可扩展，适应性好。具体算法的实现，一般用C、Fortran 、Java等，也是要效率、稳定和扩展性与可移植性。

而在数据应用的层面，世界上大部分的数据科学家都选择了R语言与Python，每种语言都有它的设计目标和长处，R语言的长处是统计学与数据批量处理，简洁紧凑。Python也长于此道并且易学易用，在云计算的环境中支持较强。但二者都需要较强的专业技能，比如数据分析与编程能力。

这就产生了这样的一种需求，去连接浏览器的易用通用和大数据云计算AI平台的数据、算法和算力，通过数据科学家们的工作，把云端的数据、算法和算力资源，方便的提供给最终的用户，因此产生了各种各样的数据分析中间件平台。

Shiny就是这样的一个数据分析中间件平台，它的客户端是浏览器，标准的HTML 、CSS 、JavaScript UI界面，所以易用通用，适应面广，各种台式机、笔记本、PAD、智能手机随时随地在线可用。它是一个中间层，数据层可以连接云端充分利用云计算平台的数据、算法和算力。它用R语言编程，客户端的网页UI完全由R语言API动态生成，数据科学家们用熟悉的R或Python连接云端完成数据分析，再把结果推送到浏览器UI呈现给最终用户，非常高效。由于UI完全由R语言(配合各种定制的JavaScript 框架与控件库与WebSocket 通讯协议等)动态生成，它的交互式体验很好，接近本地桌面应用。然后Shiny的独门绝技reactive编程设计了这样一个反应式框架，让服务器只对浏览器输入的变化作出必要的计算，更新两端的数据、状态与UI，就像Excel 表格上一处的更新会自动更新所有引用它的地方一样。是的，Shiny reactive编程大大简化了浏览器与服务器之间数据的同步，你不需要去管底下的过程，就跟在电子表格里写本地程序更新本地数据一样，从而融合了客户端与服务器端两端的编程(程序)。这极大的简化了B/S网络数据应用的开发与调试，大大提高了应用开发、运行与运维的效率。另外一方面，通过R语言的reticulate 包，Shiny可以连接Python，从而连接各种只支持Python API的云端，因此可以应用于几乎所有的云计算环境，是不是很强悍？

作为几十年来信息化建设的成果，各行各业中积累了大量的传统软硬系统和数据资产，Shiny平台设计的一个聪明之处，是把数据分析应用封装为一个个独立的网页APP，这样就可以非常方便的通过网页嵌入APP的方式，把作为数据分析中间件独立部署的Shiny平台集成进基于浏览器的传统WEB应用系统之中。HTML5提供了浏览器中跨文档通信API(Cross-document messaging)来在它们之间交换数据，实现工作流与数据流的集成。这样就可以有效的为这些传统系统扩充大数据分析能力，用较小的代价去盘活大量存量的软硬件和数据资产，性价比非常高。

这就是我要深入去了解和体验Shiny平台的原因，的确是个非常优秀的平台，值得学习。Shiny与Reactive编程模式大概在十年前出现，发展到现在已经比较成熟了，在各种场景中都有广泛的应用。从前面浏览器端B/S应用的发展过程可见，其来有自，它不是凭空发明的，而是代表了一种技术发展的趋势，水到渠成。只是国内对它的了解还比较少，推广应用少，也缺乏中文的书籍和资料。也许类似Shiny但使用Python语言的中间件平台很快就会出现了。

Rstudio首席科学家Hadley Wickham写的《Mastering Shiny》系统而全面的详细介绍了Shiny APP开发与reactive 编程的原理和要点等方方面面，程序例子简单易行，深入浅出，是一本非常好的入门书籍。英文版已经在2021年5月由O’Reilly出版社出版，Github 上有作者免费的在线阅读版，有兴趣的人可以读一读原著。国内目前还没有介绍Shiny的中文书籍，译者这里尝试翻译一下，抛砖引玉，以方便有需要的人。不切之处，还请读者诸君不吝赐教校正。

翻译的工作量较大，所以借助了现在效果已经不错的人工智能机器翻译，然后人工逐句审定，以提高效率。目前国内外大厂的机器翻译准确率都超过90%，有些甚至超过95%了，因百度的机器翻译提供了逐句高亮关联显示的功能，便于逐句审核校对，所以选用了百度的机器翻译帮助完成此项工作，并列为第一译者，在此一并表示感谢。

说明一下，是为序。

2022年8月14日