9.2.4 对响应的处理

很多机器人的兴趣主要在于用简单的 GET 方法来获取所请求的内容，所以，一般不会在处理响应的方式上花费太多时间。但是，使用了某些 HTTP 特性（比如条件请求）的机器人，以及那些想要更好地探索服务器，并与服务器进行交互的机器人则要能够对各种不同类型的 HTTP 响应进行处理。

1. 状态码

总之，机器人至少应该能够处理一些常见的，以及预期的状态码。所有机器人都应该理解 200 OK 和 404 Not Found 这样的状态码。它们还应该能够根据响应的一般类别对它并不十分理解的状态码进行处理。第 3 章的表 3-2 给出了不同状态码的分类及其含义。

有些服务器并不总能返回适当的错误代码，认识到这一点是很重要的。有些服务器甚至会将 HTTP 状态码 200 OK 与描述错误状态的报文主体文本一同返回！很难对此做些什么——只是实现者应该要了解这些情况。

2. 实体

除了 HTTP 首部所嵌的信息之外，机器人也会在实体中查找信息。HTML 元标签，[13] 比如元标签 `http-equiv`，就是内容编写者用于嵌入资源附加信息的一种方式。

服务器可能会为它所处理的内容提供一些首部，标签 `http-equiv` 为内容编写者提供了一种覆盖这些首部的方式：

```
<meta http-equiv="Refresh" content="1;URL=index.html">
```

这个标签会指示接收者处理这个文档时，要当作其 HTTP 响应首部中有一个值为 `1,URL=index.html` 的 `Refresh` HTTP 首部。[14]

有些服务器实际上会在发送 HTML 页面之前先对其内容进行解析，并将 `http-equiv` 指令作为首部包含进去；有些服务器则不会。机器人实现者可能会去扫描 HTML 文
227 档的 HEAD 组件，以查找 `http-equiv` 信息。[15]

注 13：9.4.7 节列出了一些附加的元指令，站点管理员和内容编写者可以通过这些元指令来控制机器人的行为，以及这些机器人对已获取文档所执行的操作。

注 14：有时会将 `Refresh` HTTP 首部作为将用户（或者在这种情况下，就是将机器人）从一个页面重定向到另一个页面的手段。

注 15：根据 HTML 的规范，元标签一定要出现在 HTML 文档的 HEAD 部分。但并不是所有的 HTML 文档都会遵循规范，因此，它们有时也会出现在 HTML 文档的其他区域中。