

## 16.2.5 Content-Type 首部 and Charset 首部以及 META 标志

Web 服务器通过在 Content-Type 首部中使用 charset 参数把 MIME 字符集标记发送给客户端：

```
Content-Type: text/html; charset=iso-2022-jp
```

如果没有显式地列出字符集，接收方可能就要设法从文档内容中推断出字符集。对于 HTML 内容来说，可以在描述 charset 的 <META HTTP-EQUIV="Content-Type"> 标记中找到字符集。

例 16-1 中展示了 HTML META 标记如何把字符集设置为日语编码 iso-2022-jp。如果文档不是 HTML 类型，或其中没有 META Content-Type 标记，软件可以设法扫描实际的文本，看看能否找出语言和编码的常见模式，以此推断字符编码。

例 16-1 可以在 HTML META 标记中规定字符编码

```
<HEAD>
  <META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-2022-jp">
  <META LANG="jp">
  <TITLE>A Japanese Document</TITLE>
</HEAD>
<BODY>
...
```

如果客户端无法推断出字符编码，就假定使用的是 iso-8859-1。

## 16.2.6 Accept-Charset 首部

在过去的几十年间，人们开发了成千上万种字符编解码方法。大多数客户端不可能支持所有这些不同的字符编码和映射系统。

HTTP 客户端可以使用 Accept-Charset 请求首部来明确告知服务器它支持哪些字符系统。Accept-Charset 首部的值列出了客户端支持的字符编码方案。例如，下面的 HTTP 请求首部表明，客户端接受西欧字符系统 iso-8859-1 和 UTF-8 变长的 Unicode 兼容系统。服务器可以随便选择这两种字符编码方案之一来返回内容。

375

```
Accept-Charset: iso-8859-1, utf-8
```

注意，没有 Content-Charset 这样的响应首部和 Accept-Charset 请求首部匹配。为了和 MIME 标准兼容，响应的字符集是由服务器通过 Content-Type 响应首部的 charset 参数带回来的。不对称真是太糟了，不过需要的信息倒是都有了。