

学校的理想装备

电子图书 · 学校专集

校园网上的最佳资源

Visual Interdev 6.0

程序员指南 (三)





[返回总目录](#)

第二篇 创建 Web 项目

第六章	Web 项目概念	5
6.1	项目结构	5
6.2	Web 文件处理	13
6.3	Web 应用开发周期	24
6.4	源 控 件	44
6.5	安 全	51
第七章	本地工作	75
7.1	单独开发应用	75
7.2	规定项目模式	77
7.3	获得本地主副本	78
7.4	预览 Web 应用	80
7.5	更新主 Web 应用	81
7.6	更新本地 Web 应用	82
7.7	离线工作	84

第八章	同多个开发者一起工作	86
8.1	在 Web 服务器上组建源控件	86
8.2	把源控件加到 Web 应用上	89
8.3	在源控件下工作	92
8.4	调试源控件	101
第九章	增加安全措施	104
9.1	添加安全页面	104
9.2	设置 Web 应用权限	105
9.3	连接代理服务器	111
第十章	管理 Web 项目	114
10.1	使主文件与本地文件同步	114
10.2	复制 Web 项目	119
10.3	重新组织项目结构	121

第二篇 创建 Web 项目

在第二篇中将详细讨论 Web 应用和 Web 项目，包括工作模式、源代码控件、安全和开发周期。

第六章 Web 项目概念

第六章讨论 Web 项目的有关基础概念，这些概念构成 Visual InterDev Web 应用的基础，包括“项目结构”、“Web 文件处理”、“Web 应用开发周期”、“源控件”和“安全”。

第七章 本地工作

在 Visual InterDev 中，以在线方式工作的时候有两种选择：可以直接以主模式用主 Web 文件进行工作，也可以开发你自己 Web 应用的工作方式或部分在本地模式工作。本章涉及的内容包括如何以本地模式工作，以及在本地模式工作时如何把主 Web 应用集成在一起进行工作。

第八章 与多个开发者一起工作

当你与工作组一起工作时或你自己独立工作时，用微软的 Visual SourceSave 很容易跟踪并保存文件的改变。这一章介绍如何在 Web 中使用源控件。

第九章 增加安全性

这一章讨论与安全有关的一些题目，例如“增加安全页面”、“设置 Web 应用权限”和“连接代理服务器”。

第十章 管理 Web 项目

你若使用 Web 应用文件进行工作，可能希望改变项目的结构，或者在其它服务器上制作项目副本。这一章讨论“使主文件与本地文件同步”、“复制 Web 项目”和“重新组织项目结构”。

第六章 Web 项目概念

6.1 项目结构

微软 Visual InterDev 中的项目结构为你提供创建坚强 Web 站点和 Web 应用所需的一切。借助于使用项目，你可以把注意力集中到页面应用的专门细节上，而把文件和源两方面的管理问题留给 Visual InterDev。

下面各节对项目、Web 应用和系统部件进行介绍：

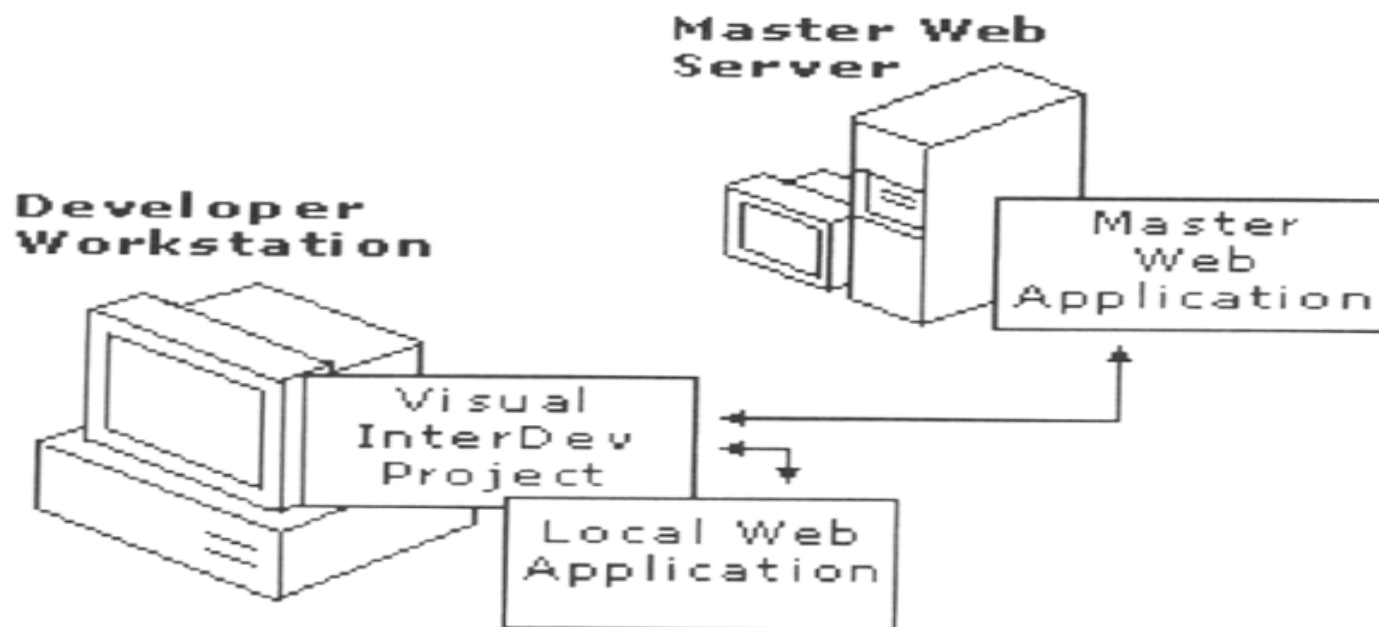
- Web 项目和两种 Web 应用
- 本地模式和主模式
- 项目和 Web 应用。

Web 项目和两种 Web 应用

Web 应用由一些文件组成。这些文件包含你 Web 内容的文件和功能：如 .htm、.asp 和图像文件。这些文件是你 Web 站点的全部或部分内容。借助于 Visual InterDev Web 项目，可以标识并操作 Web 应用文件。项目不是 Web 应用文件组的组成部分，而是由 Visual InterDev 用来指明与 Web 应用有关文件的文件。

每个开发者都创建一个 Web 项目，以便使用 Web 应用进行工作。Web 项目以图形方式在 Project Explorer 中表示 Web 应用文件。Web 项目文件位于开发者本地计算机上，这个文件不是 Web 应用的组成部分。当你部署一个 Web 应用时，项目文件并不获得这个应用的副本，只是那些包含 Web 内容和功能的文件才获得应用的副本。

当你创建一个 Web 项目的时候，实际上用分开的两个 Web 应用进行工作。一个应用是主 Web 服务器上主文件组，另一个应用是本地文件组，存在于你的计算机上。当你改变项目中的 Web 文件时，这些改变是直接在本地图文用户界面文件组上进行的。



Web 项目指向两种 Web 应用

项目结构为你提供几种好处。Visual InterDev 使用 Web 项目的指针指明 Web

应用的本地版本和主版本，并使用这个项目执行你的命令，维护这两种文件组。

Project Explorer 维护 Web 文件。Project Explorer 为你提供一个单一的窗口，对每个 Web 应用的文件进行维护。你可以使用 Project Explorer 选择工作中用到的文件，还完成各种任务。

Project Explorer 提供与文件有关的信息，包括每个文件的状态。文件名称后边的图标传达特定文件的信息。

隔离式开发。这种项目结构还允许单独进行工作，与主文件组隔离开来，这样使用主文件组的其他人都不受你改变文件的影响，直到你完成使用这些文件的工作。为此，项目要同时维护 Web 应用的两种版本。一个版本是 Web 文件的主版本，这个版本对其他 Web 开发者或对最终用户都是随时可以使用的。第二个版本驻于本地，用于在你本地计算机上进行隔离式开发、测试和调试。

例如，在上边的图中，你会看到包含 Web 文件的一个目录位于一个服务器上。这是主 Web 应用。第二组文件位于你的驱动器 C 中。这是本地 Web 应用。借助于 Project Explorer，Web 项目便于管理这些文件。你可以在本地文件组上进行工作，当你准备好的时候，就可以更新主文件组。

由于项目是独立于 Web 应用的一个文件，你可以从本地计算机中删除这个项目，不会影响到主 Web 应用文件。如果你希望的话，还可以使用 Visual InterDev 删除项目和 Web 应用。

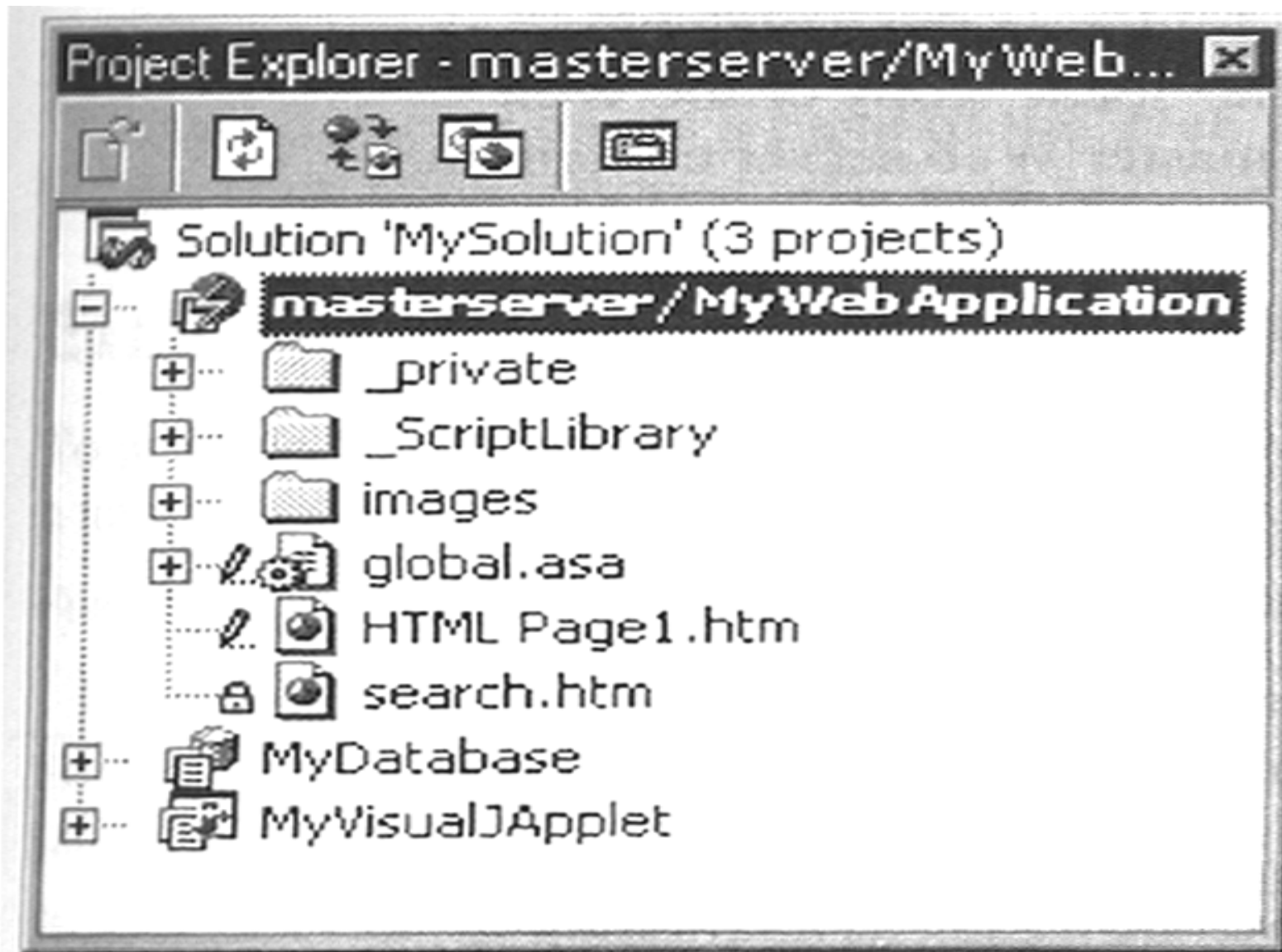
便于多个开发者进行访问。使用项目管理 Web 应用文件，允许许多开发者用 Web 应用同时进行工作。每个开发者都创建一个指向主应用的项目。使用他们自己的项目，开发者可以在本地同时进行工作。

解决方案中的项目

你在 Project Explorer 中建立的每个 Visual InterDev Web 项目都是解决方案的组成部分。一个解决方案可以由几个相同类型项目或几个不同类型项目组成。例如，你可能有一个解决方案包含两个项目：其中一个项目由一个 Web 应用组成，你把这个应用用于一般公用；另一个单独的项目包含一些 Web 页面，这些页面用于一组严格限制的用户。尽管两个项目在同一个解决方案中，但是它们中的每一个可能指向驻于不同服务器中的一个不同 Web 应用。

你的解决方案还可能包含其他类型的项目，例如数据库项目或用 Microsoft Visual J++ 建立的项目。.sln 文件存储指向解决方案中项目的信息。当你打开一个解决方案文件时，这个解决方案中的项目也都在 Project Explorer 中打开。就像项目文件一样，这个解决方案文件不看作是一个 Web 应用，也不在主 Web 应用中出现。

如果你创建一个 Web 项目，而且没有打开的解决方案，那么就为你创建一个解决方案。



一个解决方案包含几个项目

本地模式和主模式

一个项目总包含两种 Web 应用文件的副本：主文件副本和本地文件副本。当你改变文件时，这些改变是在 Web 应用文件的本地副本上产生的。项目的工作方式决定你在你本地文件上作出的改变什么时候传送到主 Web 应用中。

一个项目可以以本地模式或主模式工作。一般情况是，在你创建一个项目时采样一种模式，然后在整个开发周期中一直保持这种模式。

本地模式 你作出的改变只保存在本地文件组版本中。只有在你作出特殊选择时，主 Web 应用才保存你的修改。

主模式 你对本地文件版本作出的改变同时保存在本地版本中和主版本中。两组文件还需要进行维护，但是它们同时更新。

注意：除了两种项目模式之外，如果你不希望连接到某个 Web 服务器上的话，你还可以离线工作。关于更多内容，参见本章后边的“离线工作”一节。

关于使用项目模式的更多内容，参见本章后边的“Web 文件处理”和“Web 应用开发周期”，或第七章“本地工作”。

项目和 Web 应用

你可以找到项目文件夹及其在开发者工作站上的 Web 应用文件夹。开发者工作站通常包含 Web 项目文件的“本地”副本。主 Web 服务器文件夹位于主 Web 服务器上，所包含的文件可随时提供给其他开发者使用，也尽可能提供给最终用户使用。

本地文件夹和文件

在你创建一个新 Web 项目的时候，可以引用主 Web 服务器上的已经存在的 Web 应用，你也可以建立一个新的 Web 应用。当你创建一个新项目的时候，Visual InterDev 在你的工作站上创建以下内容：

- **一个新文件夹：**包含 Web 应用的一个本地副本。这个文件夹包含一个子文件夹。在你预览和调试页面时，这个子文件夹用于保存由工作站上 Web 服务器软件使用的本地 Web 应用。
- **Web 项目定义文件：**存储 Visual InterDev 使用的用于维护和管理项目的信息。例如，你拥有解决方案定义文件（.sln 文件）、项目定义文件（.vip 文件）和 Visual InterDev Cache 文件（.vic 文件）。
- **一个新创建的项目：**出现在 Project Explorer 之中。如果你选择创建一个新 Web 应用，那么在这个项目中的文件反映你在“向导”（wizard）中指定的文件，例如一个搜索页面文件（search.htm）。如果你选择引用一个已经存在的 Web 应用，那么你的新项目就对那个 Web 应用的结构和内容生成镜像。
- **一个新的虚拟根：**规定在本地 Web 服务器上（如果有一个 Web 服务器）。

主 Web 服务器文件

如果你选择在创建一个新项目的同时创建一个新 Web 应用，那么 Visual InterDev 就在主 Web 服务器上创建以下内容：

- **主 Web 应用文件夹：**包含 Web 应用的主副本。典型情况，这个文件夹是 Web 服务器根目录的引导子目录，而且同 Web 应用有相同的名字。这个

文件夹还保存几个隐蔽文件夹，保存着与你文件有关的元数据。如果你正在运行 Internet Information Server（Internet 信息服务器），你还可以得到一个虚拟根。关于虚拟目录的更多内容，参见 Internet Information Server 文档。

- **一个新应用根：**指向 Web 应用文件夹，还有一个权限组，允许读出页面和执行脚本。
- **一个空启动页面（Global.asa 文件）和一个搜索页面（Search.htm 文件）：**可能出现在 Web 应用文件夹的根上，取决于你选择的选项。Global.asa 文件允许你以主功能相似的形式控制 Web 应用，或者控制传统应用中的程序。当访问者打开一个 Web 应用的页面时就执行控制，并且控制 Web 应用的各种不同事件。

文件名字和位置举例

为了在你创建一个新项目之后提供 Visual InterDev 把 Web 项目文件放于何处的例子，下边表格给出典型文件组所使用的样板路径：

文件类型	项目文件夹 ¹	本地 Web 应用子文件夹	Web 服务器文件夹 ²
解决方案定义 (.sln)	\personal\projectname	No	No
Web 项目定义 (.vip)	\personal\projectname	No	No

续表

Web 应用文件 (Global.asa)	\\personal\\project name\\ projectname_local \\ global.asa	\\Default Web Site\\Web application name
Web 应用页面 (.htm,.asp)	\\personal\\project name\\ projectname_local \\Web application pages	\\Default Web Site\\applicati on name

1 对于 Windows NT, Web 项目的默认目录是 \\winnt\\Profiles\\<用户名>\\Personal

2 对于 Internet 信息服务器, 默认根是 Default Web Site. 你的 Web 服务器组建可能不同.

6.2 Web 文件处理

文件如何处理, 什么时间处理, 取决于多种因素, 其中包括你所处于的开发周期阶段、你在工作中使用的项目方式以及你对 Web 应用的需求。

下边几节说明的系统部件, 在你运行期间、设计期间和测试期间都会影响你的文件和对你的文件出现的事情。

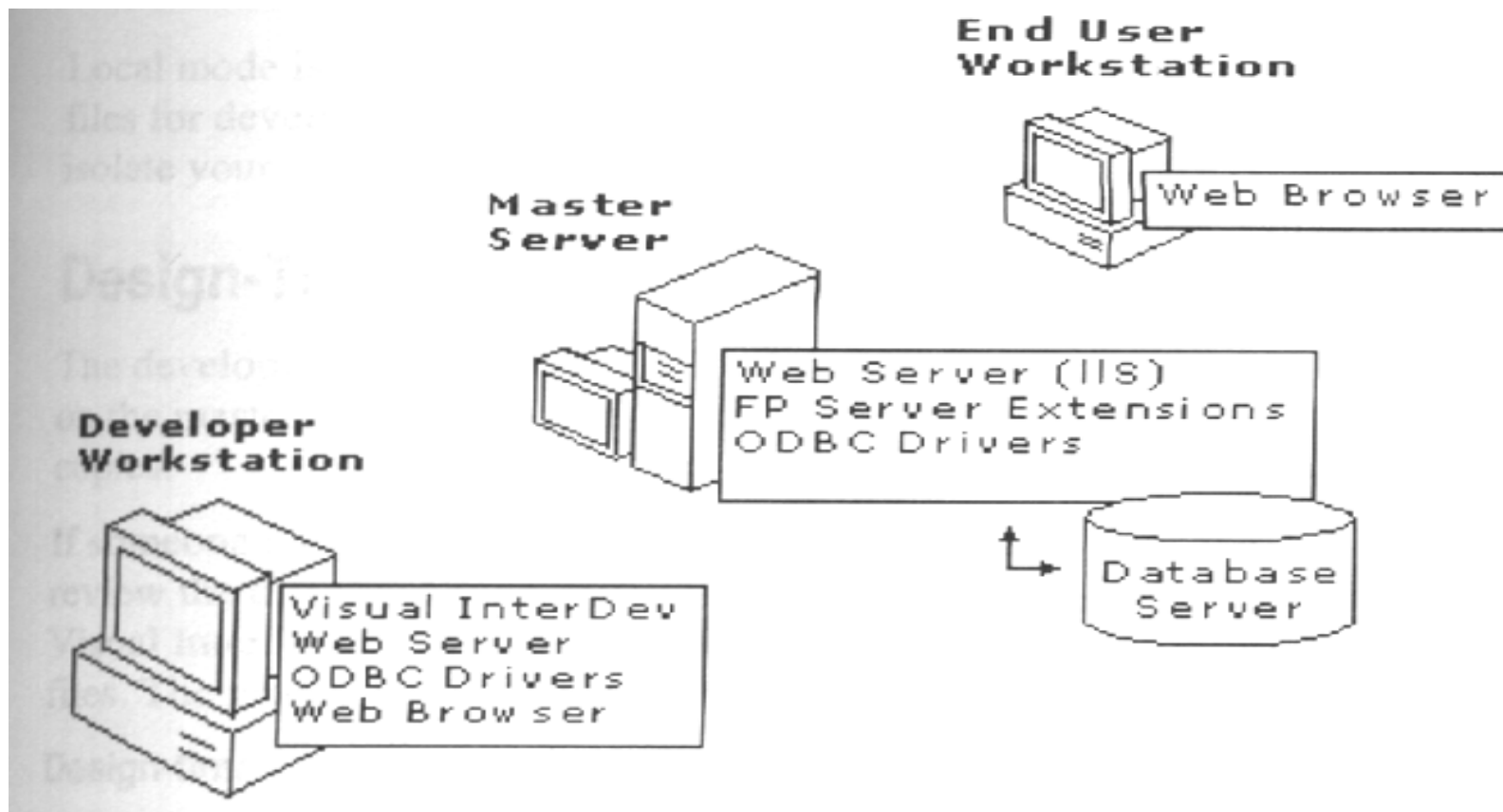
- 影响你 Web 应用的系统部件
- Web 服务器连接
- 运行期间部件的相互作用
- 本地方式交互作用
- 主方式交互作用
- 离线项目

影响 Web 应用的部件

每个 Web 应用都要求把一些硬件和软件结合起来，这些硬件和软件是结合数据库开发的 Web 应用所需要的。下边表中对每种硬件和软件做详细说明。

部 件	功 能
开发者工作站	创建 Web 项目、存储项目信息并连接到其它部件上
数据库服务器	如果需要，存储数据库定义和 Web 应用所使用的数据
带有 FrontPage 扩展版本的 Web 服务器	存储主 Web 应用、把你的 Web 应用页面发送给客户，还要处理服务器脚本和数据请求，其中包含到数据库服务器的连接
Web 浏览器	测试和预览页面，处理客户脚本

这些部件可能常驻于一个计算机中，也可能出现在两个或更多个网络上的计算机中。



典型的 Web 系统部件组

Web 服务器连接

部件之间的详细相互作用取决于你项目所使用的模式。下面说明在本地模式中在设计与测试的时候系统部件的相互作用同在主模式中系统部件相互作用的差别。

典型的系统部件相互作用使用 HTML；例外的是数据库部件，这些部件经常使用局域网 (LAN) 连接或广域网连接 (WAN)。

Visual InterDev 通过 HTTP 同主 Web 服务器进行通信。通过 HTTP 进行通信，允许你在分布环境中开发 Web 应用。在这样分布环境中，主 Web 服务器和本地开发计算机可能只通过 Internet 连接起来。HTTP 连接还允许你通过防火墙和代理服务器连接主 Web 服务器，从而防止未授权者访问你的本地开发站点。关于在代理服务器上配置 Visual InterDev 的信息，参见第九章“增加安全措施”中的“连接代理服务器”一节。

如果你的 Web 应用使用一个数据库服务器，则需要把一条 ODBC 数据库连接连到 Web 项目上，从而可以使用微软 VisualDatabase Tools（Visual 数据库工具）的长处。要得到创建数据连接的信息和用你的 Web 应用使用数据库的信息，参见第十八章“数据库概念”。

在你建立起一条数据库连接之后，可以使用 Visual 数据库工具操纵该数据库。

运行期间的部件相互作用

了解部件如何共同工作的最简单情况出现在运行期间。在运行期间，Web 服务器是在 Web 应用中的中心部件。它从用户的 Web 浏览器接收对 Web 页面的请求，把关于数据或数据库命令的任何请求都发送到数据库服务器；再从数据库服务器接收数据，处理服务器脚本，再把页面和数据送回给用户的浏览器。

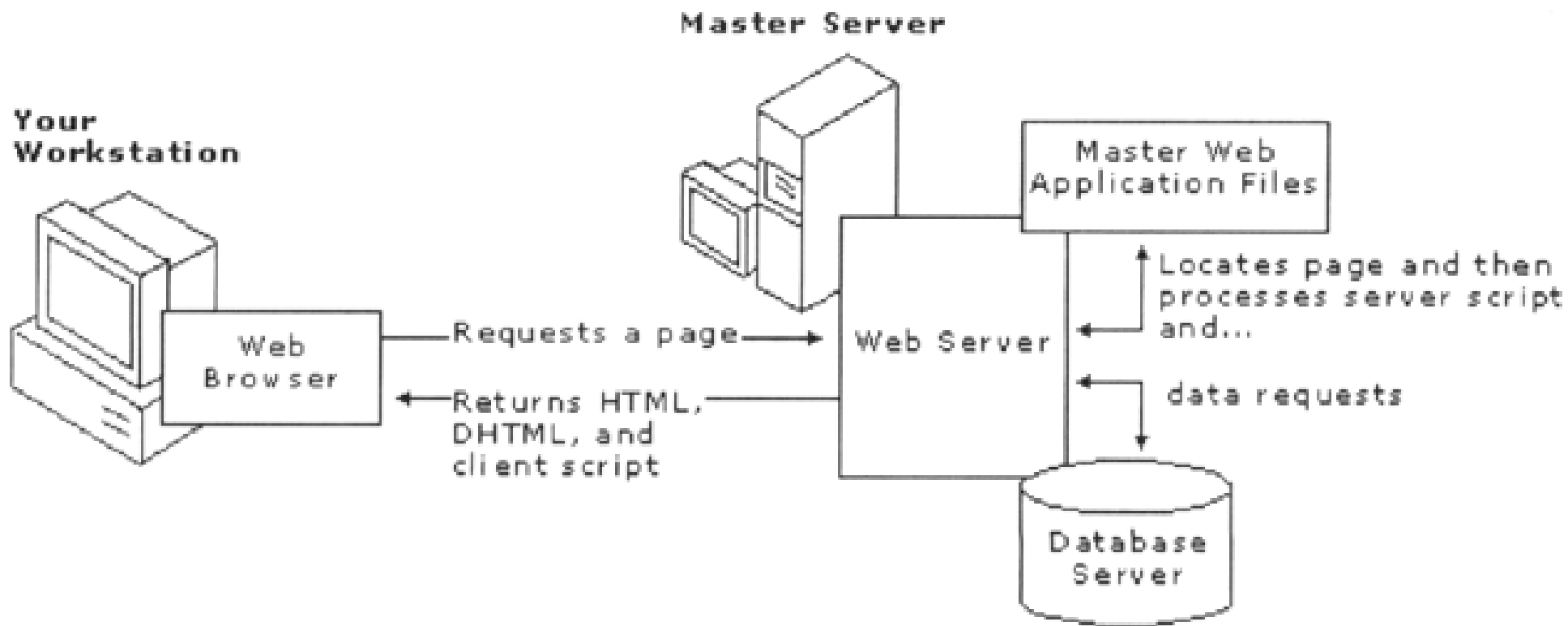
当你把 Web 应用准备给其他人看时，可以使得 Web 页面随时可以使用主 Web 服务器，或者可以把 Web 应用配置成一个产品 Web 服务器。

建议把产品服务器同你的主 Web 服务器分开，因为产品服务器有一个附加的保护层。例如，你有多多个开发者用他们的本地文件更新主服务器，就有可能

出现一个开发者作出的改变同其他开发者作出的改变互相冲突，因而会部分“打破”Web 应用。使用产品服务器，可以测试主服务器，而且把要更新的 Web 应用向公众说明这些 Web 应用的“最后”工作版本什么时候可以使用。

如果你计划使用产品 Web 服务器, Visual InterDev 使得从主 Web 服务器向产品服务器复制 Web 应用变得很容易，最终用户在产品 Web 服务器上可以浏览 Web 应用。要了解更多内容，参见第二十八章“Web 应用部署”。

下图说明在运行的时候系统部件之间的相互作用。注意，在运行的时候不使用项目。项目只在设计的时候和测试与调试期间使用。



运行期间系统部件相互作用

本地模式相互作用

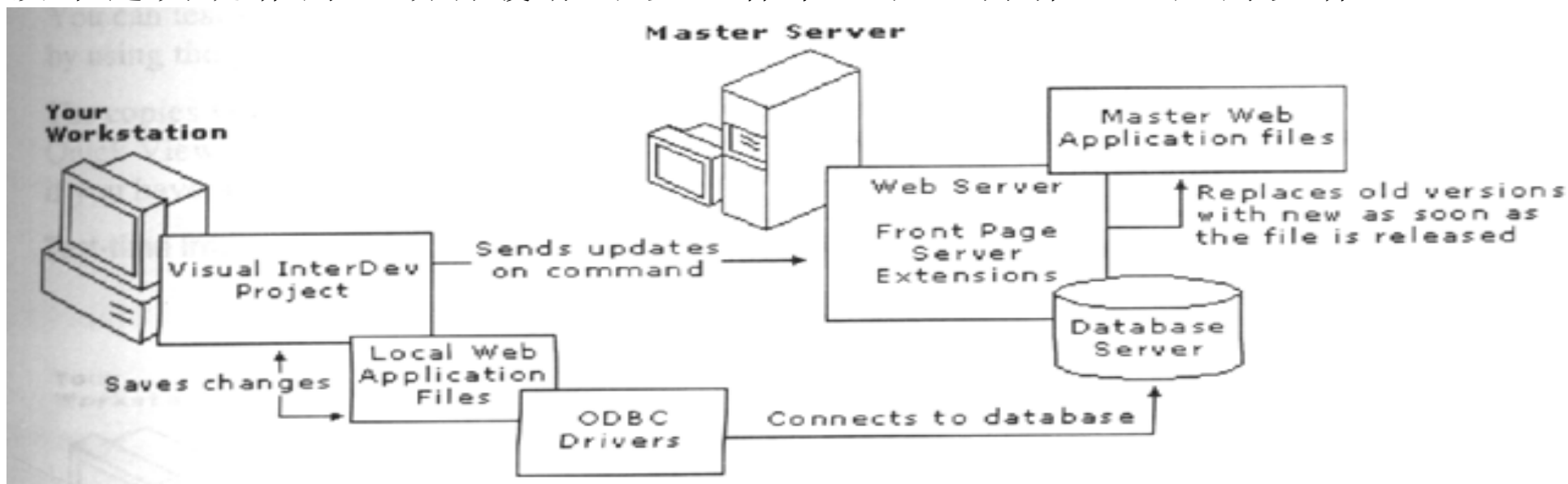
设计的本地模式可以为你提供用于隔离开发的 Web 应用版本。下边几节说明不同部件的工作如何同你的工作隔离开来，并且还迅速地把你的改变按照你的命令发送给主版本。

设计时部件的相互作用

Web 应用文件的开发在本地副本上进行，只有在你明确用本地副本更新主服

务器的时候，这些文件在主服务器上才得到更新。

如果有人对文件作了改变，而你正在本地工作，你可以查看它们的区别，在你更新主服务器的时候归并文件。Visual InterDev 项目在 Web 应用文件的开发中起关键作用。项目使你查看本地和主两种 Web 应用文件。



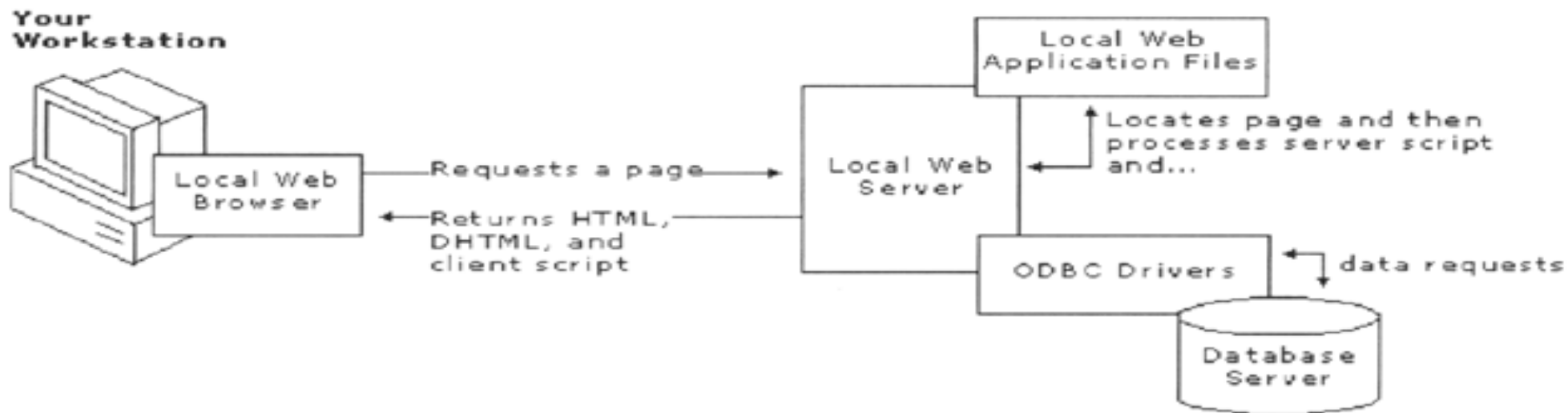
设计期间部件在本地模式中的相互作用

测试期间部件的相互作用

在本模方式中，你测试的 Web 文件是在你计算机上项目文件夹中存储的本地文件副本。你可以使用编辑器的 Quick View 选项卡测试这些文件，或者通过在你的 Web 浏览器中预览 Web 应用的办法进行测试。

如果你的工作站有一个服务器，Quick View 选项卡不使用你的本地 Web 服务器，而且不在你的 Web 应用中处理服务器脚本。

在 Web 浏览器中进行预览，你可以测试你的客户脚本。如果你的工作站有一个 Web 服务器，你还可以在 Web 浏览器中测试你的服务器脚本和 ASP。下图示出当你以本地方式测试你的 Web 应用的时候部件之间的相互作用。



测试期间在本地模式中部件之间的相互作用

要了解测试你的 Web 应用的更多内容，参见本章后边的“Web 应用开发周期”、第 11 章“站点设计”中的“链接检验”和第 23 章“脚本编程概念”中的“脚本调试过程”。

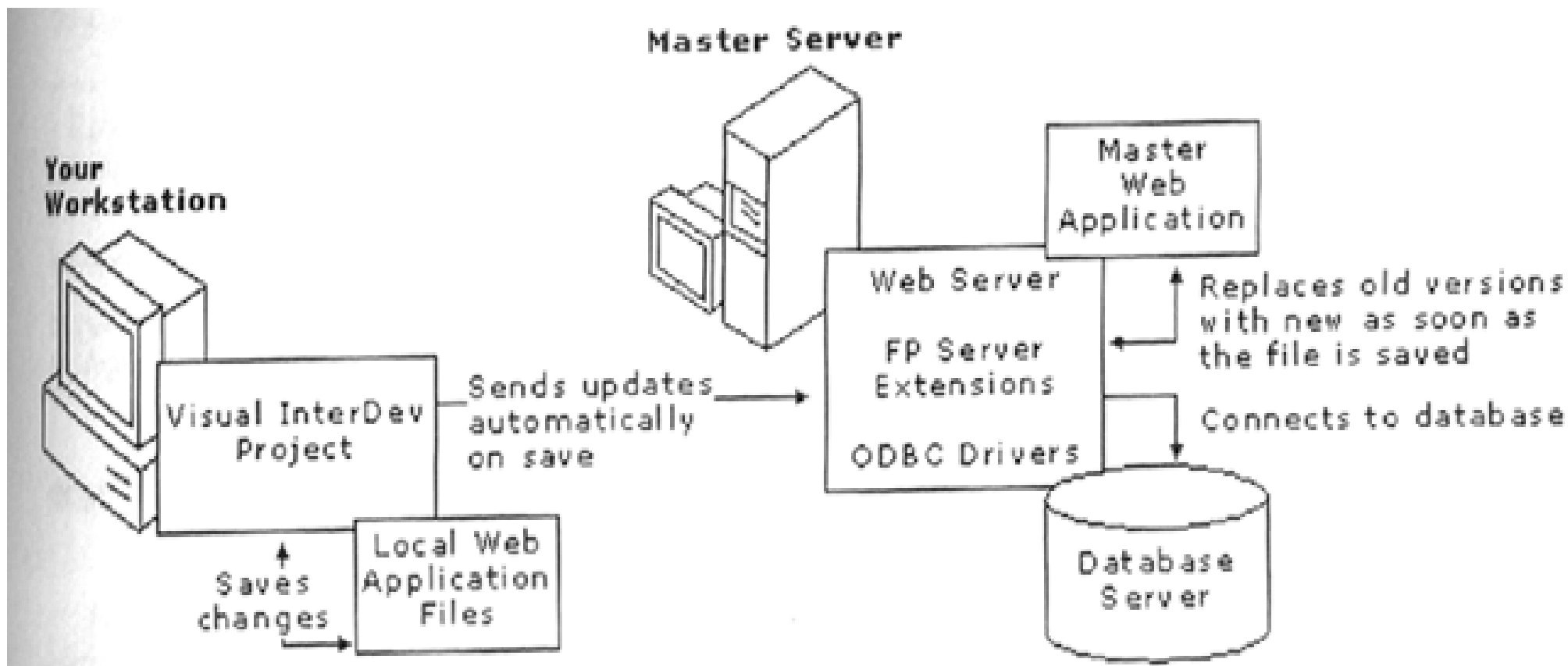
主模式交互作用

主模式把你对文件作出的改变自动传播给主 Web 服务器。这可能影响到在该 Web 应用上进行工作的其他开发者的结果。如果你是唯一修改 Web 文件的人，则主模式非常有用，但是在多个开发者情况中不推荐这种模式，因为对文件的立即改变会影响到在该应用上进行工作的其他人。

设计期间部件之间的相互作用

设计期间系统部件之间的相互作用取决于你正在进行的工作。在主模式中，文件开发在本地副本上进行，但是改变内容却在你保存这些改变到文件中的时候立即在主服务器上得到更新。

如果有人改变了文件，便会提醒你查看其差别，并在你发出工作副本的时候归并这些文件。在 Web 应用文件开发过程中，Visual InterDev 项目起关键作用。



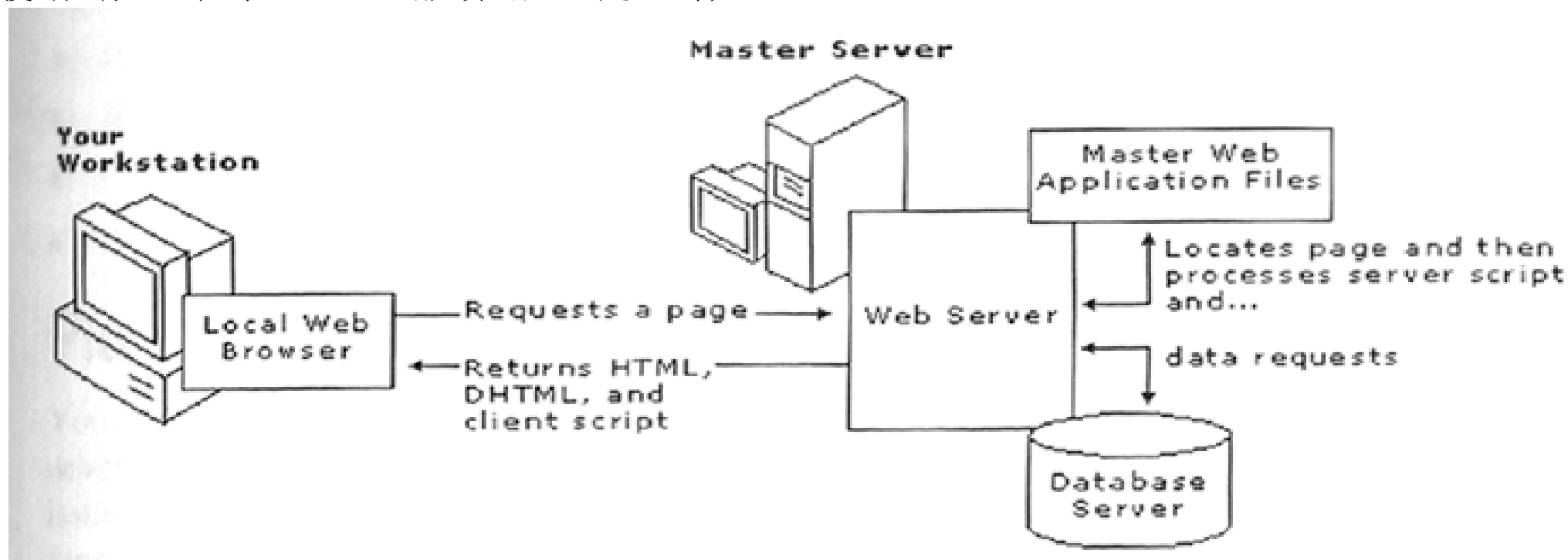
设计期间在主模式中部件之间的相互作用

测试期间部件的相互作用

你可以在主模式中进行测试，可以通过在你的 Web 浏览器中预览 Web 应用的办法，也可以用编辑器中的预览模式，还可以在链接视图中用检验链接的办法。

你进行测试的副本是主 Web 服务器上相应内容的副本。如果你使用编辑器的 Quick View 选项卡，你正查看的是本地文件，服务器脚本也不进行处理，即

便你有一个本地 Web 服务器也是一样。



测试期间在主模式中部件的相互作用

在主模式中进行测试期间，不使用本地 Web 应用，也不使用本地 Web 服务器。

要得到关于测试你 Web 应用的更多内容，参见下一节“Web 应用开发周期”、第 11 章“站点设计”中的“链接检验”和第二十三章“脚本编程概念”中的“脚本调试过程”。

离线项目

你可以离线测试项目，并继续在你的 Web 应用文上进行工作。例如，如果

你使用一个膝上计算机的话，你可能在文件上进行工作的时候从服务器上完全断开。

离线工作同本地模式类似，但是你不能更新 Web 服务器，直到你返回在线。

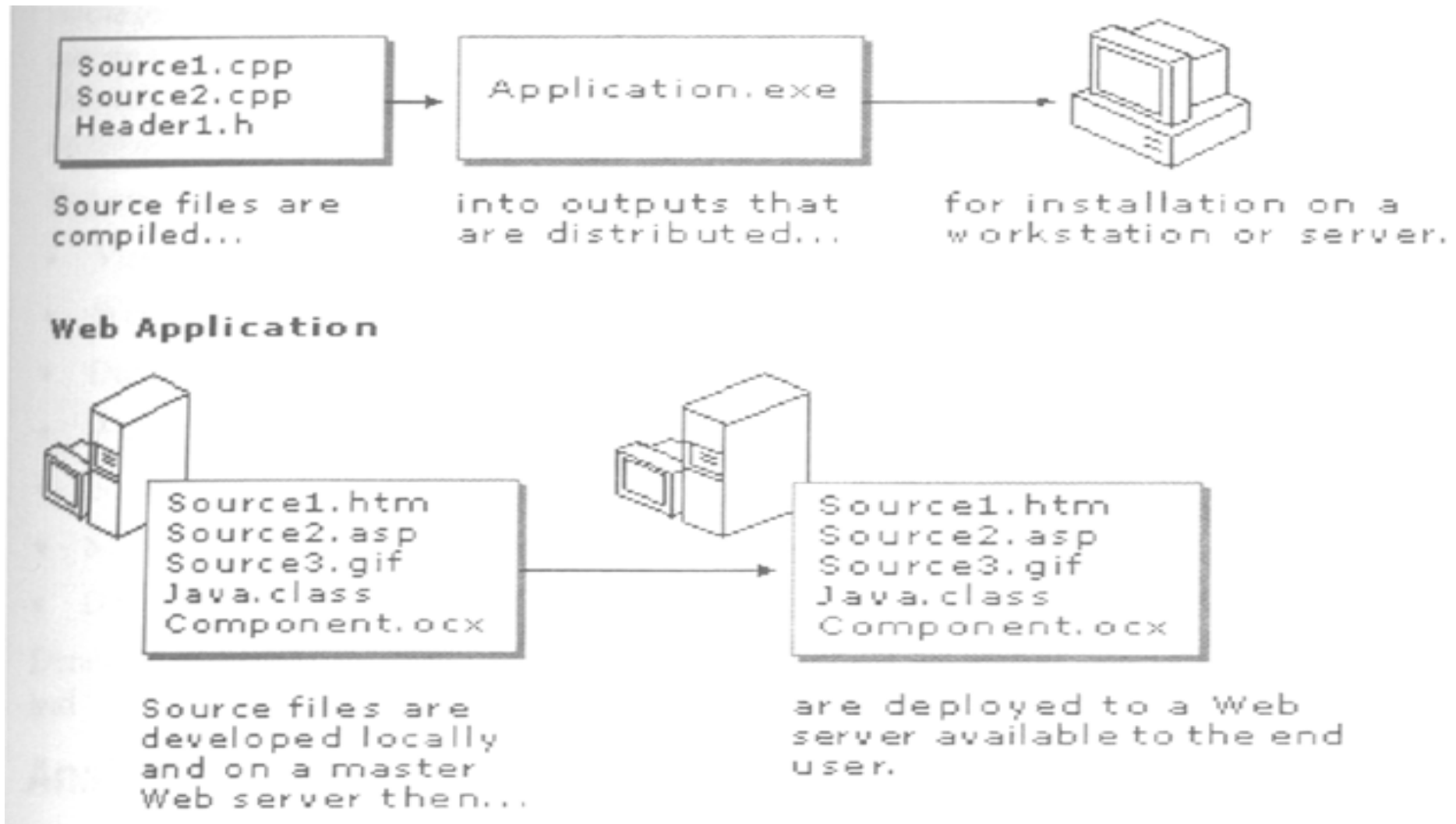
要了解关于项目模式和离线工作的更多内容，参见第七章“本地工作”和本章稍后的“离线工作”一节。

6.3 Web 应用开发周期

Web 应用开发要求同其它应用有相同的衔接阶段，然而 Web 应用本身不与传统应用相同。

传统应用在开发期间需要一组专门的文件，但是分派不同的输出。例如，类别文件、图像文件和用于开发 Java 小程序的源代码文件，都可以常驻在一个单一的微软 Visual J++项目中。然而，为了分派，这些文件都编译成一个单一的可执行文件，这个可执行的文件独立于保存在开发者 Visual J++项目中的文件。执行的结果不要求出现在开发它的源代码文件中。

另一方面，Web 应用在开发期间和开发之后都是用相同文件组构成的。这里不编译产生将变成 Web 应用的可执行文件。例如在你的 Web 文件项目中的 .htm、.asp 和 .exe 文件都是将传播到你产品 Web 服务器中的相同文件。在这些 Web 文件中的原代码或脚本都只在浏览器需要 Web 页面的时候才能在客户机上或在服务器上执行。



传统应用和 Web 应用的原文件和输出文件

在开发 Visual InterDev Web 应用中，下述过程是你要考虑的典型过程：

- 规划 Web 应用
- 组建一套解决方案

- 创建和测试 Web 项
- 采用并维护 Web 应用

规划 Web 应用

你可以使用 Visual InterDev 工具快速启动你的 Web 应用开发。认真规划可以节省更多时间。如果你在开发的早期阶段就能确定一些问题，便能比较好地实现一些有效性能和功能。例如，你在开始时确定的设计依据可能同你如何在 Web 应用中创建项发生冲突。

当你规划 Web 应用的时候，要考虑下述动作，然后创建 Web 解决方案。

- 标识观众和浏览器
- 指明 Web 应用的目标
- 确定内容
- 分析开发资源

标识观众和浏览器

规划的一项基本考虑内容是标识谁将使用 Web 应用，即观众是谁。你可以通过下述办法定义你的 Web 应用观众：标识谁将有权访问 Web 应用和他们所使用的 Web 浏览器类型。访问可来自内部网、国际互联网或来自外部网络。外部网络是 Web 站点上只对一组注册的参观者可以使用的一个区域。

在许多方面，访问 Web 应用有助于你确定适应于你 Web 应用的浏览器类型。例如，如果一个 Web 应用是为某个内部网设计的，你一定要知道每个人使用的是哪种浏览器，而且可以对你的 Web 应用进行编程以利用应用特性的优点。

Web 服务器还有能力在使用客户脚本和服务端脚本方面在你的 Web 应用中安

排你的计划。使用服务器脚本和 Active Server Page (ASP, 激活服务器页面), 你可以很容易地生成独立于服务器的页面。服务器处理服务器脚本, 然后把 HTML 发送给浏览器进行处理。与此相反, 如果你知道用户有什么类型的浏览器, 则可以使用脚本生成页面, 使得服务器上的负荷最小。根据你 Web 应用的需求, 你可以把系统的优点和你目标观众的长处结合起来。

观众也决定着你的 Web 应用运行期间需要什么类型的安全机制。你希望每个人都能够阅读 Web 应用页面吗? 是不是有些页面可对每个人, 而另外一些页面只满足某些人的一定需要? 例如, 你可能希望你的下载页面对完成某种窗体的人随时可用。关于运行期间安全问题的更多内容, 参见本章后边的“安全”一节, 再阅览安全顾问站点 <http://www.microsoft.com/Security/>。

指定 Web 应用目标

开发中的另一个关键因素是 Web 应用的目标。例如, 你的应用可能是一组通知页面, 把当前的事情通知给访问者。你的 Web 应用还可能是一个大型商业 Web 站点的顺序入口组成部分。你的 Web 应用可以是一种游戏, 用来吸引 Web 访问者, 并向他们介绍站点销售的产品。

确定内容

内容包括用于指定你 Web 应用的特性和实现 Web 应用目标的所有 Web 项和设计元素。例如, 通告页面的内容使用简单的 HTML 页面实现, 或者使用与流行的“激活服务器页面”数据库连接来实现。一个顺序入口 Web 应用可以把各种窗体和使用服务器脚本处理与客户脚本处理的数据库交易结合在一起。一种游

戏应用可以用 Java 小程序 (applet) 建立起来，也可以用动态 HTML 和页面对象建立起来，从而利用快速客户站点处理的优点。

在规划详细内容之前，你可能希望查看一下需要考虑的不同内容，以及可以快速实现设计的 Visual InterDev 工具类别。

有些工具，例如向导和样板，可以快速实现内容而不需要拥有详细内容的规划。其它一些工具，例如设计期间控件，需要你操心设计的细节，同时这种工具又特别在乎实现该设计的代码和脚本。

下边列表可帮助你规划你 Web 应用的内容：

- 使用主样板和布局的可视设计和站点导航
- Web 应用介绍和一般信息
- 用于显示、更新、添加和删除记录的数据库集成
- Web 应用控件
- 窗体和相互作用的页面
- 有条件处理的脚本
- 设计期间和运行期间安全

关于使用的工具和实现的构思细节，在本章后边的“建设一个解决方案”和“创建并测试 Web 项”两节中说明。

分析开发资源

你的开发资源会影响如何规划和设计你的 Web 应用。

谁在你的开发组中

你有一个 Web 开发者小组吗？你的开发组中有编著者和图形设计者吗？你

希望使用源控件吗？把源控件设置成允许对一个文件进行多重检查或设置专门检查吗？小组的每个成员需要什么级别的安全？开发者在他们的工作站上都需要什么工具？

对这些问题的回答有助于决定你是否需要使用源控件，以及使用什么类型的设计期间安全机制。要了解更多内容，参见本章后边的“安全”一节。

如果你属于一个开发小组的，你大概希望每个人都使用本地模式在 Web 项目中进行工作。本地模式允许开发者工作时同其他开发者作出的改变隔离开来。关于项目模式的更多内容，参见第七章“本地工作”。

以开发组形式工作，你可能把熟练程度不同的开发者结合在一起，创建那些很难由一个开发者创建或不可能由一个开发者创建的 Web 应用。以小组形式工作，下述概念就变得很重要了。

多个 Web 项目同时工作 Visual InterDev 允许多个开发者在同一个时间和同一个 Web 应用中对文件进行加工。对于多个开发者在同一个应用上进行工作的情况，必须首先使用同一个主服务器。开发者然后可以建立指向这个 Web 应用的本地项目文件。这就允许开发者维护他们自己的选项设置和主文件的本地副本。

如果你的开发组中有使用 Microsoft FrontPage 的作者，他们也可以在项目中引用你的 Web 应用。关于建立 Web 项目的更多内容，参见本章前边的“项目结构”。关于使用 FrontPage 和 Web 应用的更多内容，参见第二十九章“综合任务”中的“使用 FrontPage 和 Visual InterDev 创建 Web 站点”一节。

源控件 因为所有文件都常驻在主服务器上，多个人可以同时 Web 应用进行工作。多个用户可以从主服务器上得到文件，用这些文件进行工作，然后进

行修改，更新服务器副本。

为了防止多个用户在同一时刻改变同一个文件，你可以安装 Visual SourceSafe 5.0 或以后的版本，从而使源控件可用于你的项目。关于使用 Visual SourceSafe 和 Visual InterDev 的更多内容，参见本章后边的“源控件”一节。

多重检查和归并解析 当有多个开发者的时候，你可以让多个人同时在一个文件上进行工作。例如，你可能让一个作者对内容进行工作，让一个开发者改变同一个文件的某些脚本。当文件在主服务器上进行更新的时候，正在保存改变的第二个人有机会使这种差别消失。要了解更多内容，参见第八章“同多个开发者一起工作”。

你的开发系统是什么系统

你的开发者工作站上有开发与测试 Web 应用所需要的服务器部件和客户部件吗？你的服务器部件、数据库开发部件和开发者工具都分别在不同的计算机上吗？你的 Web 应用要求什么级别的设计期间安全措施？

这些部件的位置影响你如何测试你的 Web 项和 Web 应用。例如，Web 应用在本地进行测试，因为本地 Web 服务器没有不能处理的“激活服务器页面”。如果在你的开发者工作站上没有 Web 服务器副本，你可能希望考虑使用主模式中的 Web 项目，从而不必明确更新主服务器并对主服务器进行测试。

你的测试系统是什么系统

你有一个为开发组同时进行开发使用的测试服务器或称登台服务器（staging server）吗？你们开发组每个成员都在本地开发他们的文件，然后把这些文件上移到这个登台服务器吗？你的测试数据库常驻在哪里？

测试位置有助于确定你希望使用的项目方式。推荐的单个页面测试与开发过程是借助于把你的项目设置成使用本地模式同主版本上完成的测试隔离进行的。

你的部署系统是什么系统

你用于测试的 Web 服务器上采用你的 Web 应用吗？你的主服务器也是你的产品服务器吗？你的产品数据库常驻于什么地方？你的 Web 应用要求部署服务器使用什么级别的运行时安全机制？

你测试服务器和产品服务器之间的关系决定你如何完成某些服务器管理和文件管理任务。我们更愿意把开发中的应用和产品版本保持在不同的机器上。要了解详情，参见本章前边的“项目结构”一节。

开发企业的应用

按照大多数人的说法，一个企业的应用规模是可变的、分布式的、以部件为基础的并且是任务关键性的。企业应用趋向于数据集中式的，并且要严格满足在安全、管理和维护方面的要求。

开发扩展的企业级应用使得 Web 应用开发过程增加一个或两个数量级的复杂性。但是把 Visual InterDev 同其它微软 Visual Studio 工具结合起来，你可创建这些复杂的应用。

在 Web 应用的所有这些问题也都出现在企业应用上，但是在开发企业应用的时候还需注意两点：

- 规划扩充。在设计阶段考虑伸缩性和性能。如果你预料到有大量网络信息，规划就要利用微软事务处理服务器技术和 ASP 连接组合的长处。

把业务、数据库和用户接口尽可能分开。

- 尽可能创建部件。把功能包装在 ActiveX 控件中和 COM 对象中，从而允许你有效地分享和维护代码。Visual Basic、Visual FoxPro 和 Visual C++ 便于把你的代码包装到在 Visual InterDev 应用中有影响的部件中。

要了解讨论企业应用开发的详细内容，参见 *Developing for Enterprise* 一文。“什么是企业应用”对企业模型做了很好的概括，也是在规划企业应用时起步的好文章。

组建一个解决方案

一个解决方案可使你在一个时期内有效地在几个项目上进行工作。例如，一个解决方案可以包括一个 Web 项目和一个微软 Visual J++ 项目。这个 Web 项目管理组成 Web 应用页面的所有文件。例如，你可能有几个页面，这些页面构成数据库的前端。或者你可以创建和采用某些页面，把处于开发中的 Web 应用的情况通知你的用户。对于这组页面，你可以创建一个项目，用于发行某些基本 Web 页面，宣布 Web 应用，并在全部 Web 应用可用时保持访问者不断更新。

你可以把第二个 Web 项目加到解决方案上，以便为最后 Web 应用开发一个原型。还有，如果规划建立一个数据库，你还可以增加一个数据库项目。你的 Web 项目还可以包括在 Visual Basic 或 Visual C++ 中建立的 Web 部件。

在组建一个解决方案的时候，你大概希望完成下列一般工作：

- 建立一个 Web 项目，用于创建和管理你 Web 应用中的文件。
- 增加一条数据库连接或创建一个新数据库。

- 通过向你的 Web 项目增加 Web 项的办法开发一个原型。

建立一 Web 项目

为了在创建 Web 应用中使你花费的时间和代价最小，Visual InterDev 组建你所需要的基本部件，并且为你提供选择，使你一开始就在布局和感觉上都实现标准化。

为组建一个 Web 项目，你需要提供下列信息：

- 主 Web 服务器名字
- 本地项目名字和物理目录
- Web 应用名字和虚拟根
- 主服务器更新模式：主模式或本地模式
- 主样板（可选）
- 布局（可选）。

在你创建一个 Web 项目之后，你可以加上一些项，这些项实现在规划阶段你所标识的特性。

独立工作

你的 Web 项目实际支持两组 Web 应用文件。一组文件常驻于主 Web 服务器中，另一组文件在本地你的项目目录中。本地版本允许你独立于主版本进行开发与测试，这里主版本依赖于项目的模式。详细内容，参见第七章“本地工作”。

你的项目可以在下述两种模式之一中运行：本地模式或主模式。在本地模式中，你可以保存和测试你在本地 Web 应用文件中作出的改变，不影响对主副本的改变。对于主 Web 应用而言，你的工作同其他人作出的改变是隔离的，直

到你发出本地副本以前，你的改变都不保存到主 Web 服务器中。关于这两种方式的更多内容，参见第七章“本地工作”中的“规定项目模式”。

当你以本地模式工作的时候，你作出的改变都施加于叫做本地 Web 应用的基于本地磁盘的 Web 应用上。当你最后更新主 Web 应用、刷新项目或同步文件组的时候，这个本地 Web 应用对主 Web 应用结构作出镜像，同主 Web 应用一模一样。详见第十章“管理 Web 项目”中的“使主文件与本地文件同步”一节。

有时候，你的本地 Web 应用可以包括不同于主 Web 应用的一组文件。例如，如果你把一个文件加到本地方式中一段时间，这个文件就立即加到你本地 Web 应用中，但是不加到主 Web 应用中。如果开发组从他们的本地 Web 应用更新主 Web 应用，你的 Web 项目将不显示这些改变。要解决这个问题，就要更新主 Web 应用，刷新你的项目，或者用主 Web 应用同步你的本地 Web 应用。要了解详细内容，参见第七章“本地工作”中的“更新主 Web 应用”和“更新本地 Web 应用”两节。

离线工作

如果你不需要访问服务器或服务器已经停止工作，则可以离线工作。在离线工作的时候，服务器所要求的命令都不能使用了。例如，你可以在本地目录中的文件做出改变，但是你不能发出这些改变去更新主 Web 应用。详见第七章“本地工作”中的“离线工作”。

此外，离线测试项目的功能取决于你的开发者工作站上的系统部件。如果在你的开发者工作站上有一个 Web 服务器软件，该项目就组建一个 Web 应用根，这个根允许你测试你工作站上的 Web 服务器软件和服务器扩展。如果你没有 Web

服务器软件和相应的扩展，文件就直接从本地 Web 目录中装载。关于测试各种 Web 项所需要部件的更详细内容，参见本章前边的“项目结构”一节。

在项目之外工作

如果你添加、创建或删除 Visual InterDev 和 Project Explorer 之外的 Web 应用文件，则需要在返回到 Project Explorer 的时候刷新该项目。例如，如果你添加或删除使用 Windows Explorer 的一个文件，该项目将不对本地目录和本地文件显示这种改变，直到你返回该项目为止。关于刷新 Web 项目的更多内容，参见第十章“管理 Web 项目”中的“使主文件和本地文件同步”。

增添数据库

如果你的 Web 应用规划包括数据库访问，则可以把已经存在的数据库集成起来，或专门为 Web 应用修改数据库。在 Web 项目中，仅仅添加一条到数据库的连接，你就可以在 Web 应用文件中使用数据库中的信息。这条连接还可以显示数据视图窗口，你可以使用这个窗口查看数据库中的对象，并把数据环境添加到你的项目中。

你还可以使用 Visual Database Tools（Visual 数据库工具）从你的 Web 项目中修改数据库对象。在你有了一条数据库连接之后，就可以在该 Web 项目中继续工作，开发一个原型。关于 Visual 数据库工具，参见第十八章。

开发原型

在你创建 Web 项目和数据连接之后，就可以迅速开发 Web 应用的原型。某些开发者等待添加他们 Web 应用视觉和感觉元素，直到他们实现了重要特性和

功能。在你创建项目和选择主样板与布局的时候，你可能已经把这些元素加到 Web 应用上。关于主样板和布局的更多内容，参见第十一章“站点设计”中的“站点一致性”。

创建和组织一组页面

通过内容规划，你可以用创建站点图的办法为你的 Web 应用勾画出 HTML 和 ASP 页面布局。站点图在 Web 项目中提供页面的图形视图。这些页面自动把你组建 Web 项目时选择作为默认的主样板和布局结合在一起。

在你组建起初始原型页面和 Web 应用组织之后，你可以修改它们，以满足 Web 应用的要求。

在站点图中建立起页面之间的层次关系，你就可以容易地为 Web 页面设计导航链接。如果希望修改视觉和感觉视图，你可以借助于改变主样板和布局迅速完成。关于站点图的更多内容，参见第十二章“设计 Web 站点”。

介绍 Web 应用的作者文本

典型的 Web 应用页面有一些很少改变的信息。HTML 就用于实现这样的文本。

你的 Web 应用通常至少在一个页面中包括介绍 Web 应用目的的信息和告诉 Web 访问者如何使用 Web 应用或特定页面的信息。这些信息一般是放到 Web 应用的起始页面上，为 Web 应用提供一个通用入口点。你的 Web 应用还可能包括其他设计元素，如窗体、数据或相互作用目标等。

如果你正在为 Internet 进行设计，这个页面应当是独立于浏览器的，并把正确运行 Web 应用所需的任何特殊软件或浏览器要求通告给用户。

实现数据和脚本

在创建一组初始页面之后，你就可以在编辑器中打开页面，添加能实现 Web 应用目标的文本、图像和脚本。在确定如何实现你的内容时，要考虑的主要问题之一是你是否希望在客户软件或服务服务器上完成处理。其答案取决于观众、他们浏览器的能力和你开发系统的资源。使用 Web 项要综合考虑，详见下一节“创建和测试 Web 项”。

不论你决定使用客户还是服务器脚本编程，Visual InterDev 都提供一个 Web 页面编辑器和数据环境，允许你迅速实现功能。例如，你可以使用数据环境创建用于多个页面的一个记录集对象。这个编辑器还提供数据绑定设计期间控件，你可以迅速地把这些控件添加到你的页面上，以便显示这些记录。详见第二十三章“脚本编程概念”中的“Web 应用中的脚本”和“文档元素”两节。

控制 Web 应用

为按照 Web 应用控制你的页面，你可以使用服务器处理“激活服务器页面”及其特定文件，即 Global.asa 文件。在 Global.asa 文件中，你可以对对话级(session level)和 Web 应用级事件和变量编写脚本，这些事件和变量在 Web 应用中使用服务器脚本的所有页面上随时可用。

创建和测试 Web 项

Visual InterDev 提供各种可视设计软件和 ActiveX 控件以实现你的特性组。

标识项使用

在规划你的 Web 应用功能的时候，你可能希望考虑使用各种项来规定 Web 应用中的特性。因为一个 Web 应用实际上是用一组常驻于服务器中的文件组成的，因而你可以用文件类型把每个项类型标识出来。例如，你可以把下列任何项组合到你的 Web 应用中。

添 加	考 虑
静态页面或客户脚本	.htm 文件
服务器脚本	.asp 文件
数据公布	记录集
设计工具	设计期间控件
可视设计	样式、主样板、布局、图像和多媒体文件
相互作用页面	ActiveX 控件
综合 Web 解决方案	从其它项目，例如从小应用程序，输出文件
可下载文档	文档文件、电子表格

创建简单信息页面

你可能希望创建一些直截了当的 HTML 页面，用来宣布你的新 Web 应用正处于开发之中、什么时候可以得到它们、以及你的用户可以从中得到什么效益。一些少量的静态信息页面可以占据你 Web 应用的位置，产生使用这些信息的兴趣。你可能采用这组页面组建一个 Web，使它在内部网上，同在 Internet 上一样有用。

在布置这些页面的入门版本之后，你可以很容易地形成动态 Web 应用的起始页面。在你的 Web 应用完成之后，你需要做的全部事情就是采用整组文件，并用改进的页面和构成 Web 应用的其它页面取代以前的版本。与原始介绍页面的链接将继续工作，把用户直接带到你的 Web 应用中。

如果你希望创建几个页面或简单创建一个包含宣布内容的 HTML 页面的话，你可以使用站点图。你可能希望添加图像或其它效果，增加页面的视觉感染力。

公布动态数据

Web 应用可以作为任何 ODBC 数据源的前端提供服务。你可以设计一些页面，用于显示数据或允许用户更新、添加或删除记录。要显示数据，你可以使用数据绑定设计期间控件。你可以选择数据是否在服务器上进行处理，并把文本发送给客户，或者在客户机上实现一些动态数据的能力。对于数据连接的更多内容，参见第十八章“数据库概念”和第三章“数据库基础”中的“连接数据库”一节。

为交互式页面编写脚本

典型的 Web 应用包括带有窗体或其它项的页面，其中这些项要求用户动作或对用户动作作出响应。窗体构成 HTML 本征元素。你可以使用设计期间控件快速裁剪窗体的元素，来实现 Web 应用的要求。有了动态 HTML，你就可以在页面上创建动态效果，并且编写脚本对事件或时间通道作出响应。要了解更多内容，参见第五篇“编辑和编写脚本”。

尽管许多 Visual InterDev 和工具都对 Web 应用的许多元素生成基本代码

和脚本，你大概还是希望使用某种条件控制你 Web 页面的结果。例如，你可以希望某些页面出现以告诉用户：他已经完成了窗体制作，或者提醒用户窗体制作已经完成但尚未填写完格式。关于使用编辑器和对页面进行脚本编程的更多内容，参见第二十三章“脚本编程概念”中的“脚本对象模型”或“Web 应用中的脚本”。

增加运行期间安全措施

你可以通过使用实现安全措施的办法防止你的资源被未授权使用。除了在你的服务器中设置安全措施之外，还可以增添口令页，用户通过 Web 应用检验脚本。例如，你可以增添一些页面，这些页面把收集用户标识符和口令的 HTML 窗体同服务器脚本集成在一起，对这些信息同一个数据库进行比较。关于 Web 应用安全方针和增添安全措施的更多内容，参见本章后边的“安全”一节。

设计可视效果和站点导航

可视设计和站点导航决定你的 Web 观众对你的 Web 应用的初始印象。采用彩色、图像和声音的方式可使 Web 应用显得直观，并且容易使用。对你的站点导航进行精心设计可使用户从一个页面到另一个页面周游你的 Web 应用变得很容易。使用一个站点图，你可以在你的页面上布置图形，同时规定导航。

在创建 Web 应用之前，规划站点的总视觉与感觉。或者选择一个 Visual InterDev 主样板和布局，在创建 Web 应用页面的时候就为你的 Web 应用赋予和谐的可视效果。当创建一个项目的时候，就可以指定由 Visual InterDev 提供的多个主样板中的一个主样板。

如果在你的心中有了一个设计主样板，就可以创建自己的主样板、样式和布局，很快在这个 Web 应用上产生和谐效果。

主样板和布局文件

Visual InterDev 为你提供几种主样板(theme)和布局文件，可以使用这些文件为 Web 应用赋予和谐的视觉和感觉。详见第十七章“页面样式定制”。

站点导航

站点设计者为设计你的站点导航提供一种图形方法。创建站点布局图，定义页面之间的层次关系。当你规定这种关系时，站点图便自动规定页面之间的链接，并把这些链接加到你的 Web 页面上。更多内容，参见第十三章“设计站点导航”。

动态 HTML 和 ActiveX 控件

你可以使用动态 HTML 在 HTML 页面上布置文本、图像和多媒体内容。动态 HTML 为你提供一组丰富的控件，用于对多媒体页面编写脚本。这些控件为你提供一组可以动态改变的对象和参数，用于在页面上改变视觉和声音效果。关于使用控件的详细内容，参见第二十四章“用设计期间控件和脚本对象编写脚本”。

测试链接和调试脚本

你可以按照你的进度测试整个 Web 应用。要测试的两个主要方面是链接和脚本。

查找断开的链接和没有链接的页面

在创建 Web 应用中，页面之间链接的检验与测试是一个主要问题。Visual InterDev 为你提供检验与测试你 Web 应用之间链接的工具。

检查单页上的链接 使用链接视图，你可以检验一个页面上的链接是否指向已存在的一个页面。通过链接图，你可以看见到达 Web 页面或从这个页面发出的链接图形表示。链接视图只检验到达一个存在 Web 页面的链接点。它不能确定链接点是否到达正确的页面。关于链接视图的更多内容，参见第十一章“站点设计”中的“链接检验”一节。

检查整个 Web 应用的链接 如果你希望发现整个 Web 应用中断开的链接和没有链接的页面，可以使用“断开链接报告”生成一个列表。要了解更多内容，参见第十六章“维护链接”中的“维修链接”一节。

检验链接终点 要测试每条链接是否到达正确页面，需要在一个浏览器中打开一个页面，并检查每条链接。要对你的项目在本地模式中或在离线情况下测试链接，就需要确保你希望查看的所有文件都在本地目录中。如果目标文件不在本地目录中，浏览器就不能分辨链接。

自动维修链接 Visual InterDev 还提供链接维修特性，在你改变文件时，这个特性可以自动确定链接。例如，如果你改变一个文件的名称，到这个文件的所有链接都用新名称更新。甚至简单改变名称或移动一个文件，都会强迫项目中的其它文件作出响应的改变。

链接维修以你的名义修改所需要的文件，更新链接指向正确文件。链接的改变首先施加到主 Web 应用上，然后施加到本地 Web 应用和所有受影响的需要更新的文件上。如果一个受影响的文件在外部编辑器中打开，或者，如果其他

开发者有一个打开的文件，那么在这些文件中的链接必须手动确定。更多内容，参见第十六章“维护链接”中的“维修链接”一节。

调试脚本

你可以像在传统 Web 应用中一样对脚本进行调试。当预览一个页面时，你可以查找语法错误、运行期间错误以及逻辑错误。

使用编辑器及其调试软件，你可以控制脚本的执行，并建设变量值和特性值，辨别所遇到错误的原因。关于调试脚本的更多内容，参见第二十三章“脚本编程概念”中的“脚本调试过程”或第二十六章“调试页面”。

部署和维护 Web 页面

在你对 Web 应用调试之后，并对应用性能感到满意的时候，你可以把应用部署到你对用户开放的 Web 服务器上。因为一个 Web 应用实际是一组文件，你需要把虚拟根和其文件组复制到产品 Web 服务器上。如果你的页面使用这个虚拟根作为它们链接的根基，那么所有链接都应当继续工作。如果你的项目包括一个依赖项目，你还需要确保服务器部件的输出在服务器上恰当注册。关于部件注册的更多内容，参见第六篇“建设综合解决方案”。

Web 应用的优点是易于维护。要更新一个 Web 应用，你不需要重新编译和重新分布整个可执行的新文件。你需要做的全部事情就是添加新文件并更换以前的版本。

在更新期间，你的用户不被中断而继续工作，因为他们的浏览器正在使用从 Web 服务器得到的原始文件副本。当原始文件被新版本更换时，用户只在页面被刷新以后（而不是刷新时）看见新版本。新文件组的更新是平滑进行的。要

了解更多内容，参见第三十章“部署和维护 Web 应用”。

6.4 源 控 件

你的 Web 应用可以很容易地成长为一个大型文件组。要保持你的 Web 应用反映当前状况，就需要一个工作组创建并维护这些文件。不管你是以工作组还是个人进行工作，你都可以很容易地使用微软 Visual SourceSafe 追踪并保存对文件的改变。为了维护对 Web 文件的改变，你可以使用 Visual SourceSafe，并与微软的 Visual InterDev 保持协调一致。

使用源控件，你同 Web 工作组其他成员可以分享文件，独立修改文件，并在随后归并这些变化。Visual SourceSafe 也在一个数据库中保存这些文件的以前版本，留下变更日期和时间踪迹，并提供一个用于保留注释日志的选项。

最常用的 Visual SourceSafe 命令可以随时直接从 Visual InterDev 得到。Visual SourceSafe 命令组的其余命令随时可从 Visual SourceSafe Explorer 中得到。

要使用 Visual InterDev 所具有的源控件，需要了解：

- 源控件部件
- 你同 Visual SourceSafe 的交互作用
- 在 Visual SourceSafe 同 Visual InterDev 之间的交互作用。

源 控 件 部 件

要在 Web 文件中使用源控件，你需要下述内容：

- **Visual SourceSafe 服务器** 用选择的综合选项安装到你的主 Web 服务器中。详细内容，参见第八章“同多个开发者一起工作”中的“在 Web 服务器上安装 Visual SourceSafe”一节。

注意：在开发者工作站上安装 Visual SourceSafe 客户软件是可选的。

客户软件对检查你的 Web 应用来说不是必需的，只要在主 Web 服务器上安装了 Visual SourceSafe 就可以了。

- **FrontPage Extensions** 安装到主 Web 服务器上。Visual InterDev 使用 FrontPage Extensions 把命令发送给 Visual SourceSafe。
- **Visual InterDev** 安装到你的开发者工作站上。
- **源控件开启** 在 Web 应用上实现源控件开启，并在主 Web 服务器上设置使用 Visual SourceSafe。

注意：要得到源控件的最好结果，就要使用一个服务器上的 Visual SourceSafe，而这个服务器要运行带有 NTFS 系统的 Windows TN。

同 Visual SourceSafe 的相互作用

可以按照你希望完成的任务，在几种级别上同 Visual SourceSafe 相互作用。有些任务只需执行成一次，其它一些任务则需要定期执行。

下表给出你可能要完成的一些典型任务，怎样完成这些任务，以及完成这些任务时要使用的接口。每个任务的更详细信息可通过使用表中的链接得知。

任 务	使用频度	使用的软件
安 装 并 组 建 Visual SourceSafe	每个主 Web 服务器用一次	Windows 和 Visual CourceSafe Setup
为多个用户检查设置选项	偶尔使用	Visual SourceSafe Administrator
把源控件加到 Web 应用上	每个 Web 应用使用一次	Visual InterDev
检查文件出入	经常使用	Visual InterDev
解决归并冲突	偶尔使用	Visual InterDev
设置检查代码选项	偶尔使用	Visual InterDev
查看文件历史	偶尔使用	Visual SourceSafe Explorer*
重存过去版本	偶尔使用	Visual SourceSafe Explorer*

* 如果使用代理服务器用于远程访问，就不能使用 Visual SourceSafe Explorer 查看你的 Visual SourceSafe 数据库，因为该 Explorer 要求局域网连接。要了解更多内容，参见第六章“Web 项目概念”中的“项目结构”。

下边各节介绍文件追踪、同时检出(checkout)与归并、历史细节，以及其它 Visual SourceSafe 选项。

文件追踪

当你安装 Visual SourceSafe 并使它为你的 Web 应用做好准备之后，就可以直接进入 Visual InterDev 开始工作。Visual SourceSafe 和 Visual InterDev 结合在一起平滑地追踪你对文件施加改变和在你 Web 项目中的远程文件。你对 Web 应用文件施加的任何改变都扩展到 Visual SourceSafe 项目中，而不要求打开 Visual SourceSafe。

当你检出 Web 项目中一个文件的时候，你要在本地 Web 中获得一份文件的可写副本。在你检查文件进入时，新副本就复制到主服务器的主 Web 应用中，并发送给 Visual SourceSafe 存储器。你的本地副本变成只读文件。如果你正在离线方式中工作，在你进入在线方式并检查一个文件进入时，新版本同时复制到主 Web 应用和 Visual SourceSafe 存储器中。更多内容，参见第八章“同多个开发者一起工作”中的“在源控件下工作”一节。

同时检出与归并

如果 Visual SourceSafe 设置成允许同时检出 (checkout)，你会担心在对文件进行检出时是否有某个人对该文件作出改变。当出现这种情况时，你可以查看这两个文件之间的差别，选择对它们进行归并，接受指定的变更，或者两个版本中都去掉变更。

例如，你正在一个脚本上工作，而某个人正在改变一个 Web 页面的内容。当你检入 (check in) 这个文件时，你可以把这两个版本归并到一个包含两种变更的文件中。

同时检出的选项是用 Visual SourceSafe Administrator 设置的。关于

检出的更多内容，参见第八章“同多个开发者一起工作”中的“解决归并冲突”一节。

文件历史细节

Visual SourceSafe 提供这个选项，对每次检入（check in）都保存注释，在你检入一个文件时提醒你做一个注释。如果你的规划不保存历史注释，可以使用 Visual SourceSafe Administrator 中的 Source Control（源控件）选项，禁止出现注释对话框。

Visual SourceSafe 选项

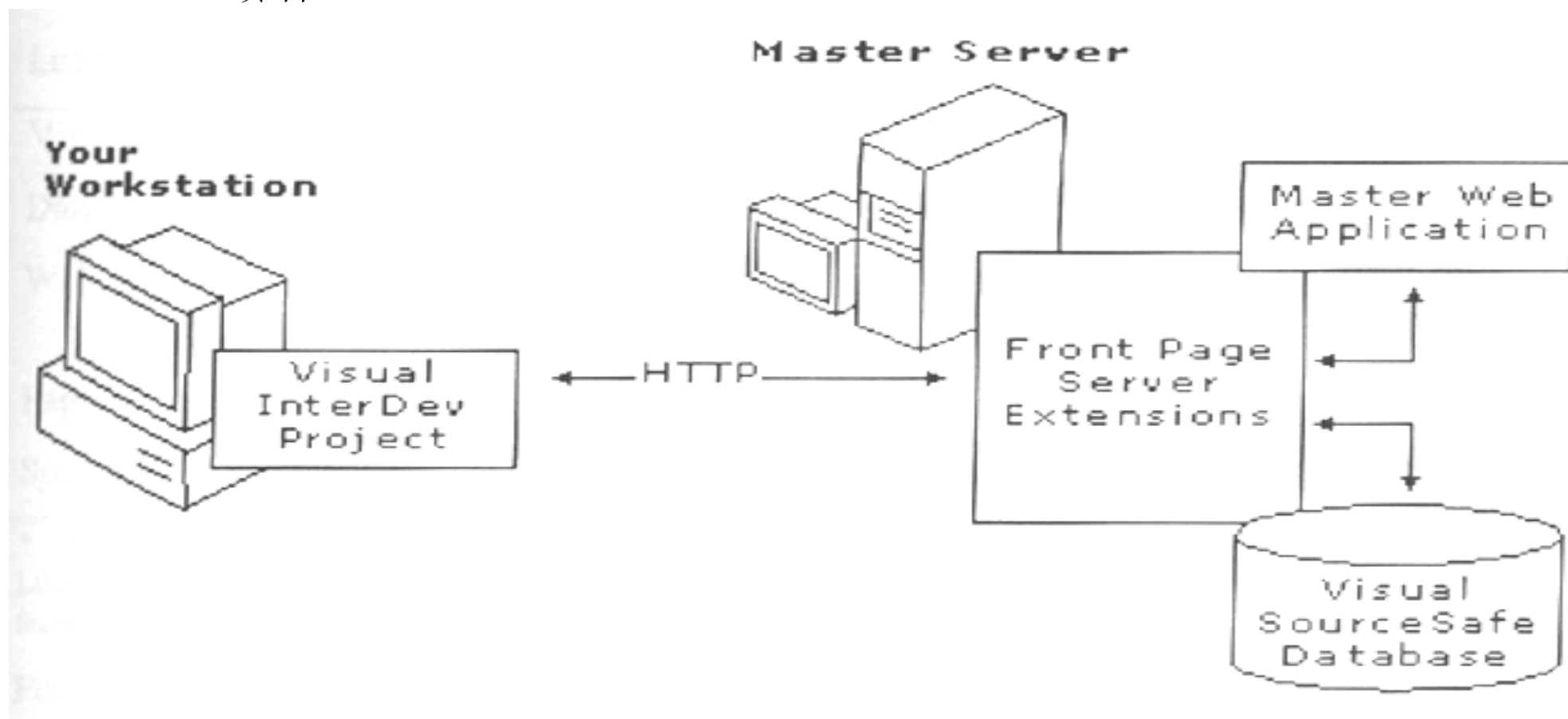
你还可以设置附加选项，确定 Visual SourceSafe 如何提供提醒，并处理对文件的改变。这些设置影响用于 Web 应用的 Visual SourceSafe 选项，因而也影响在 Web 应用工作的每个人。关于设置选项的更多内容，参见 Visual SourceSafe 题目中的“选项命令”。

你还可以设置检出选项，决定是否总是获得允许写副本、只读副本还是提示进行选择。关于设置默认选项的更多内容，参见第八章“同多个开发者一起工作”中的“检出文件”。

Visual InterDev 与 Visual SourceSafe 之间的相互作用

Visual InterDev 和 Visual SourceSafe 通过 FrontPage 服务器扩展版本而相互作用。Visual SourceSafe 和 FrontPage 扩展版本必须常驻于同一台计算机中，并对同一个主 Web 服务器进行工作。如果你有“Internet 信息服务器”

(IIS) 4.0 版本，就可以有多个虚拟主 Web 服务器，但是只能涉及到一个 Visual SourceSafe 项目。



Visual InterDev 和 FrontPage 服务器扩展

FrontPage 服务器扩展版本提供 Visual InterDev 和该 Web 服务器上其它服务之间的连接服务。在你检入 (check in) 一个文件时，该文件从开发者工作站上的 Visual InterDev 移动到 FrontPage 扩展中，后者更新主 Web 应用并把一个副本交给 Visual SourceSafe。Visual SourceSafe 把该副本，连同记有日

期、时间和检入该文件用户的一个记录存入自己的存储器中。如果你选择添加注释，这个注释文本也包括在其中。

下表示出在你完成通用文件命令时出现的动作。

任务	动作
检出并打开一个文件	本地模式和主模式：检查该文件是否已经进行特别检出。如果没检出，则检出该文件并把该文件的允写副本放到你本地 Web 应用中 如果你处于离线模式：在你与服务器断开期间不能检出文件。你要在离线之前检出文件。
保存文件	本地模式：把文件保存在本地 Web 应用中。该文件不保存到主 Web 应用中，也不检入到 Visual SourceSafe 中。 主模式：把文件保存到本地 Web 服务器中，并更新主 Web 应用中的副本。该文件不检入到 Visual SourceSafe 中 如果你处于离线模式：该文件保存到本地 Web 应用中，不保存到主 Web 应用中。该文件不检入到 Visual SourceSafe 中。
保存与检入文件	本地模式与主模式：检入文件并更新 Visual SourceSafe 中和主 Web 应用中的副本。根据你的检入选择，还可能从本地 Web 应用中去掉可写副本。 如果你处于离线模式：在同服务器断开期间你不能检入文件。在你返回到在线方式之后需要检出这些文件。

6.5 安 全

World Wide Web 和 HTTP 为 Web 应用提供越来越多的观众，对安全的需要也成比例增加。此外，Web 开发工作组现在遍布全球。需要严格控制观众和工作组如何在 Web 应用文件上进行工作，并对有安全要求的系统控制访问权。

Web 应用的安全是一个复杂的问题，因为它可以在不同模式中设置几个级别。这些选择既取决于使用的系统和服务器，也取决于 Web 应用的需要。

一些考虑因素如下：

- 设置安全措施位置
- 用户受权
- 允许访问 Web 服务器、文件夹和文件
- 设计期间安全方针
- 运行期间安全方针
- 保证 HTTP 安全传输 (SSL)
- 管理和维护方面的考虑

关于安全方面的更多内容，参见 MSDN Technical Artcles 中的“Authentication and Security for Internet Developers”（MSDN 技术论文集集中的“对 Internet 开发者的授权与安全措施”），或参见在 <http://www.microsoft.com/security/> 的 Microsoft Security Advisor Web site（微软安全顾问 Web 站点）。

设置安全措施位置

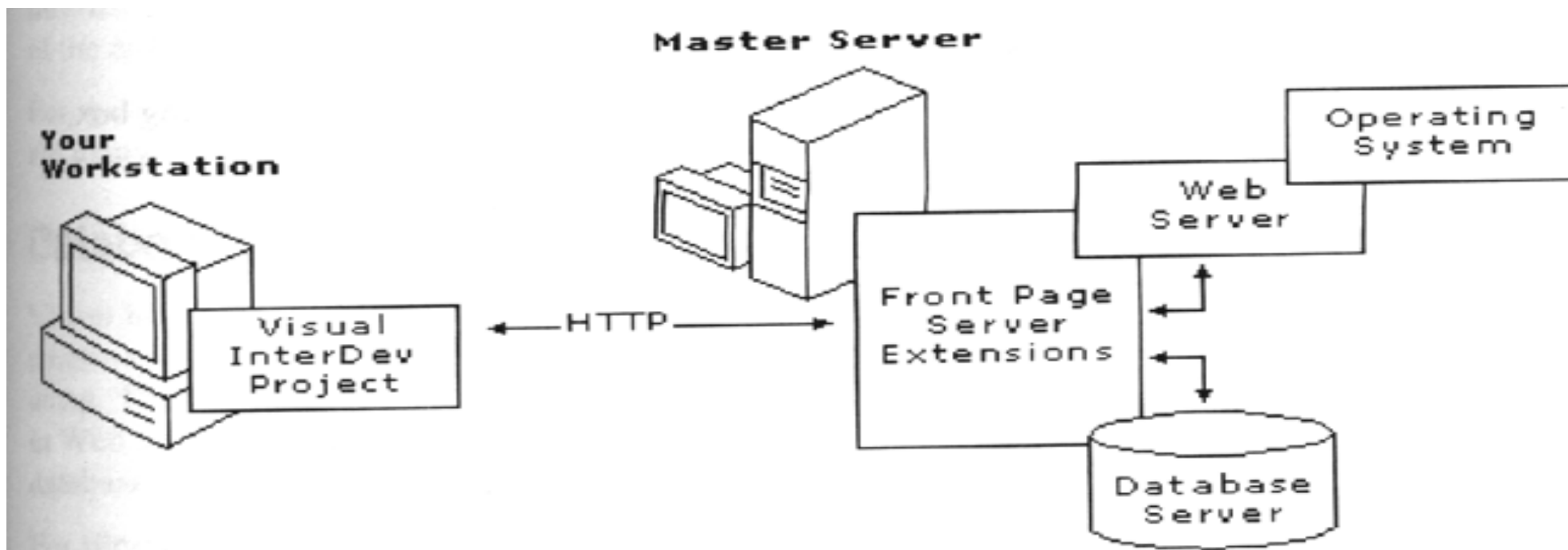
根据你的系统和对访问者制定的前提条件，你可以在几个位置设置安全措施。例如，对于内部网的基本安全可以同网络本身的安全问题同时解决与维护。对于 Internet Web 应用，你可以把安全机制加到页面上，再通过这些页面把安全提供给使用该 Web 应用的 Web 浏览器。

下表列出施加安全机制的许多位置选项。正如你所见到的那样，你可能需要几种接口获得希望的结果。这个表假设你正在使用微软的 Windows NT，并带有 Microsoft Internet 信息服务器作为你的 Web 服务器，还带有 Microsoft SQL 服务器作为你的数据库服务器。

位 置	接 口 举 例
操作系统	在 Windows NT*的 User Manager 中规定的用户帐号
文件夹	在 Windows NT Explorer 中设置的共享特性
Web 服务器	“ Internet 信息服务器管理器 ” 中设置的用户帐号和服务器特性
虚拟根	“ Internet 信息服务器管理器 ”
数据库	在 SQL 服务器，企业管理器中规定的权限
Web 应用	在 Visual InterDev Web Permissions 中设置的并保存在一个对话期间的 Web 应用特性
页面	在 Visual InterDev, Source Code Editor 中编写的服务器脚本
源控件	在 Visual SourceSafe Administrator 中规定的用户帐号和权限

* 只用于 Windows NT 和 FS 文件系统的 Web 服务器上。NTFS 允许你对每个文件规定“访问控制列表”（ACL）。使用 FAT 文件系统，不论是在 Windows 95 上还是在 Windows NT 服务器上，都将大大降低安全特性。

对于你 Web 服务器上的所有交互作用，Visual InterDev 都使用 Visual FrontPage 扩展服务器，而这个扩展服务器可以使用操作系统和 Web 服务器上已经有的安全特性。例如，这个扩展服务器同 Windows NT 和 Internet 信息服务器集成在一起，管理 Web 应用的安全。你可通过 Visual InterDev 用户接口与 FrontPage 扩展服务器进行交互作用。



Visual InterDev 和安全部件

当规划在 Web 应用中使用什么安全选项和希望在什么地方设置这些安全选项的时候，考虑对你的站点有四类访问是有益处的。你可能希望：

- 允许任何 Web 访问者执行 ASP 页面并在运行时阅读 HTML 页面
- 限制在运行时对注册的 Web 访问者进行访问
- 允许 Web 开发者和作者在设计时编写你的文件
- 限制对受权的用户进行 Web 管理。

要全面了解对你的 Web 应用有影响的安全选项，可以阅读 FrontPage 扩展服务器、Internet 信息服务器和 Windows NT 的有关安全特性的内容。如果你在 Web 应用中使用数据库，你就可能希望熟悉 ODBC 数据库的安全模式，例如 Microsoft Access 或 SQL 服务器。要了解对于 Windows NT 和 Internet 信息服

务器设置安全选项的更多内容，参见这些产品的文档。

用户受权

在 Web 安全方面，你特别关心的问题是识别用户和控制用户对 Web 应用及其资源的访问。通过对受权的用户进行核实并规定访问权限的办法达到这个目的。

在选择安全选项的时候，你需要规定用户如何向你的 Web 应用提供一种标识、如何检验这种标识，以及允许用户得到什么级别的访问权或权限。下表提供一些方针，帮助你为你的 Web 应用选择相应的选项。

位置	接口举例
操作系统	域和在 Windows NT 中规定的用户帐号
Web 服务器	在 Internet 信息服务器管理器中设置的用户帐号和服务器特性
数据库	在 SQL 服务器，Enterprise Manager 中规定的权限
Web 应用	在 Visual InterDev Web Permissions 对话框中、在登录页面上，以及在对话期间存储的变量中设置的 Web 应用特性

这些位置的每一种位置都对实现用户标识、检验和权限级别提供不同特性设置。根据你使用的系统和软件，这些特性设置可以同操作系统级别设置的选项一起工作，或者同其它级别上变更相关选项一起工作。例如，如果你在 Visual InterDev Web Permissions 对话框中的一个文件或一个文件夹中设置权限，那么你在 Windows NT 中也设置同样的权限。

操作系统级

在操作系统级别进行安全设置，取决于你的操作系统提供的特性和选项。例如，运行 Windows NT 时，你可以为很多人授权，用不着使用一个特定的用户帐号来明确规定这些用户帐号，这也就是所谓“客人用户”(guest user)。这个帐号是默认设置的。如果你想区分各个用户，并追踪他们，就需要在另一种级别上而不是在操作系统上实现安全选择。

除了授权之外，操作系统使用的文件系统也会影响你赋予用户的权限。例如，Windows NT 可以使用 Windows NT 文件系统 (NTFS)，也可以使用文件位置表 (FAT)。NTFS 允许你对文件或文件夹规定“访问控制列表”(ACL)，因而你可以比较灵活地控件访问。但是，FAT 不提供规定列表能力，提供的控件也很少。

Web 服务器级

操作系统一旦完成对用户标识的检验，Web 服务器就可以评估这个用户的标识。Web 服务器主要是在运行时对访问进行控制。例如，如果你对 Internet 信息服务器使用默认配置，那么 Web 服务器就把匿名请求作为 Internet 信息服务器匿名用户 IUSR_<machinename>来处理。

Internet 信息服务器创建的匿名用户与在 Windows NT User Manager 中定义的 Internet 客人帐号类似。但是，口令变更不能自动处理。对其中之一变更口令，还要在另一个系统中明确变更口令。

如果你设置的匿名帐户允许访问请求的文件，那么 Internet 信息服务器就允许访问这个文件，满足请求。否则，Internet 信息服务器拒绝请求，并给客

户返回一个出错信息，通知客户 Internet 信息服务器支持的授权方法。

详细内容，参见你使用的 Web 服务器文档中有关安全方面内容。

数据库级

Visual InterDev 允许连接广泛的数据资源。你为数据库设置的安全措施取决于使用的数据库管理系统。下边提供一些经验，用于在 Web 应用使用的数据库中设置安全措施。对于数据库设置安全的最新信息，参见数据库管理系统文档中安全题目中的内容。

对于以文件为基础的系统，例如微软 Access 中的 .mdb 文件，你可以通过共享操作系统中文件夹和文件中的权限来控制安全。如果你正使用 SQL 服务器，可以使用这些数据库管理系统中提供的交付与撤销特权的特性。例如，微软 SQL 服务器就有一种 Security Manager（安全管理器），允许你为单个用户或一组用户规定特权。

尽管 SQL 服务器为 Windows NT 提供三种类型登录安全措施，我们还是推荐选择使用标准登录安全措施。标准登录安全措施在访问服务器时要求一个 ID 登录号和一个口令。Web 应用通过数据连接提供这些信息，你的用户不需要提供任何其它标识。

某些 Web 应用可能使用综合安全机制（但这不向所有系统推荐）。如果你选择使用综合安全机制，要记住，SQL 服务器将使用 Windows NT 安全措施对用户授权，因而 Web 应用和数据库必须常驻同一个服务器中。Windows NT 不把安全机制委派给其它服务器。

Web 应用级

在你的 Web 应用中，你可以利用安全特性中一些特性的优点，而这些安全特性是通过 FrontPage 扩展服务器和通过随时可用 Active Server Pages 进行处理的 Global.asa 文件提供的。

FrontPage 扩展服务器

Visual InterDev 通过 FrontPage 扩展服务器进行工作，使用主 Web 服务器上主机操作系统的基础安全模型，为管理设计时 Web 权限提供可能性。

如果你的操作系统是带有 NTFS 文件系统的 Windows NT，FrontPage 扩展版本便使用下表中用于 DLL 的 ACL 文件管理管理员和作者的访问权。这些目录对 Web 服务器来说是隐蔽的，但是文件系统可以随时使用。

功 能	DLL	位置
管理（例如设置 Admin.dll 权限）	Admin.d ll	<Webdir>/_vti_bin/_vti_adm
作者（例如打开一个文件）	Author. dll	<Webdir>/_vti_bin/_vti_aut
浏览（例如查看链接）	Dvwssr. dll	<Webdir>/_vti_bin/_vti_aut

在你执行一个功能的时候，例如改变 Web 应用上的权限，你的请求发送到服务器上的 HTTP，通过路由选择到达这些 ISAPI DLL 中的一个。例如，要完成一项管理功能的请求就是由那个 Web 应用的 Admin.dll 处理的。在这个请求中，

客户提供证据，证实该他是在客户工作站上登录的用户。这个用户必须有 DLL 处理该请求的阅读权限（相当于阅读和执行个人权限）；否则该请求被拒绝。

FrontPage 可以借助于控制 DLLs in_vti_bin 阅读权限而限制谁能够执行请求。不论什么时候通过 Web 权限对话框对 Web 应用权限作出变更，服务器上的 FrontPage 扩展版本便修改那个 Web 应用相应_vti_bin 目录中 DLL 的_vti_adm 子目录和_vti_aut 子目录上的 ACL。

注意：FrontPage 管理设计期间安全机制时不改变内容文件上的 ACL；只改变包含看护文件 admin.dll、author.dll 和 dvwssr.dll 目录上的 ACL。

FrontPage 操纵内容文件 ACL，管理运行期间安全，这是下一节的题目。

Global.asa 文件处理

Visual InterDev 支持一些可以充分使用“激活服务器页面”的 Web 应用。“激活服务器页面”的特性之一便是在同那个 Web 应用建立一个新会话的时候自动处理 Web 应用的 Global.asa 文件，如果这个文件出现的话。你可以使用这个 Web 应用和会话来控制对 Web 应用的访问权和处理权。用户实际上从来看不见 Global.asa 文件和你施加的安全措施。

允许访问 Web 应用、文件夹和文件

在规定识别用户的办法之后，你就可以通过设置权限来控制对系统资源的访问权。一般情况是依据文件和目录设置权限。

在操纵系统级别上，你可以对单个用户设置权限：读、写、执行、删除、改变权限，或者取得一个文件的所有权。

此外，Windows NT 还定义一些标准权限，这些权限是把一个权限或多个单

项权限结合起来的权限。这些权限通常分配给文件或目录。例如，拥有变更权限就相当于拥有读、写、执行和删除各单项权限。拥有全面控制就相当于拥有所有单项权限。拥有标准阅读权限就相当于拥有分别阅读和执行权限。

控制用户使用 ACL 访问资源

把标识和授权结合起来决定一个用户对资源的访问权。把整组权限分配给一项资源，例如分配给一个文件或一个目录，便是所谓“访问控制列表”（ACL，Access Control List）。当一个用户企图访问一项资源的时候，操作系统便检查该资源的 ACL，检验那个用户是否拥有相应的权限完成企图进行的访问。例如，一个用户对一个文件拥有只读权限，但他企图删除该文件，那么这个用户就被拒绝访问，他不能删除该文件。

这里有一个简单例子：一个 Web 站点管理员希望使用安全手段保护该站点上一个文件，比如 Sample.htm 文件，以防止用户通过匿名浏览站点的办法来阅读文件。简单改变 Sample.htm 上的权限，使得匿名用户帐号没有这个文件的阅读权限，于是匿名用户便不能看见 Sample.htm 文件。

注意：你为 Windows NT 用户组设置的权限中“任何人”（Everyone）也包括匿名用户。

Visual InterDev Web 应用

Visual InterDev 把由 FrontPage 定义的安全模式暴露给一些 Web，这些 Web 寄宿在运行 FrontPage 扩展服务器的一个服务器上。这个模式汇总如下，在 FrontPage 文档中的“Using the FrontPage 97 Server Extensions with the Microsoft Internet Information Server”（在微软 Internet 信息服务器上使

用 FrontPage 97 扩展服务器软件) 一文中 有详细描述。把上述文件和目录中讨论的权限模式扩充到 Web 应用中。

Web 访问级别

FrontPag 扩展服务器为 Web 应用定义三个访问级别。对单个用户或整个用户组可以分配这三个级别中的一个级别，或不分配访问权。访问权是以每个 Web 应用为基础定义的。在 Web 上定义访问权的操作便是所谓“设置 Web 权限”。三个级别访问权是管理员、作者和浏览。这些访问权描述如下：

访问级别	授予的权限
管理员	创作与浏览文件 如果在根 Web 上，在根 Web 服务器上创建或删除 Web 应用。 如果在 Web 应用上，在该 Web 应用上改变权限
作者	在 Web 应用上浏览、打开和修改文件，包括添加和删除内容
浏览	通过浏览器阅览 Web 的内容 在默认情况下，在创建一个 Web 应用时，浏览访问权授予所有用户。你可以对 Web 应用限制浏览访问权，规定用户是否拥有浏览访问权

设计期间安全方针

设计期间安全管理是通过控制那些用户或用户组授予对 Web 应用的管理访问权和创作访问权来实现的。

要给一个用户对一个 Web 应用全部设计期间访问权，用户就应得到对那个 Web

应用的管理访问权。如果你希望一个用户能够在一个服务器上创建新 Web 应用，这个用户就需要在那个服务器上根 Web 的管理特权。要给一个用户足够的访问权对已经存在的 Web 应用进行充分修改，但是不能删除它，这个用户就应该得到对 Web 应用的作者访问权。

在设计期间，你可能需要对 Web 应用工作组成员提供不同级别的访问权。不是所有成员都需要管理特权。有些成员可能只需要对某些文件夹或目录的写访问权。作为默认权限，Web 应用继承在该 Web 应用第一次创建根时所具有的同样权利。关于在设计期间规定特权的更多内容，参见第九章“增加安全措施”中“设置 Web 应用权限”一节。

控制开发者对开发 Web 应用的访问权

如果你希望控制工作组对改变你的 Web 应用文件所拥有的特权，你可以使用 Visual InterDev 和在其它地方的安全设置。下表可帮助你选择同 Web 应用相适应的安全设置。

设置位置	典型内部网设置
操作系统	Windows NT Challenge/Response (NTLM)
Windows Explorer 文件夹 (OS)	把用户组或个人加到权限列表中
Web 服务器	取决于具体服务器
数据库	在“企业管理器”中把安全类型设置成标准类型
Visual InterDev Web 项目	把用户加到 Visual InterDev Web 权限列表中
Visual SourceSafe	把用户加到 Visual SourceSafe Administrator 上

下表给出在 Internet 上的典型安全设置。

设置位置	典型 Internet 设置
操作系统	基本清除文本
Windows Explorer 文件夹 (OS)	把用户组或单个个人加到权限列表中
Web 服务器	与操作系统相同
数据库	在“企业管理器”中把安全类型设置成标准类型
Visual InterDev Web 项目	把用户加到 Visual InterDev Web 权限列表中
Visual SourceSafe	把用户加到 Visual SourceSafe Administrator 上

规定 Web 管理员

在你第一次安装的 Web 服务器时，将提示你提供一个 Web 管理员口令。你可以规定其他用户对你的 Web 应用具有同样的管理员特权。例如，你可能希望某个人也能添加新用户或者改变 Web 权限。

你可以使用下表选择同你的 Web 应用相适应的安全选项。

设置位置	典型内部网设置
操作系统 (Windows NT)	Windows NT Challenge/Response (NTLM)
Windows Explorer 文件夹 (OS)	把用户组或单个个人加到权限列表中
Web 服务器	用户组和口令设置必须同操作系统一致
数据库	在“企业管理器”中提供相应的数据库权限
Web 应用	Global.asa 文件
Visual InterDev Web 项目	把用户加到 Visual InterDev Web 权限列表上

下表给出对 Internet 的典型安全设置

设置位置	典型 Internet 设置
操作系统 (Windows NT)	基本清除文本
Windows Explorer 文件夹 (OS)	把用户组或个人加到权限列表中
Web 服务器	允许匿名
数据库	提供 TCP/IP 访问权和专用 .dll; 在“企业管理器”中提供相应的数据库权限
Web 应用	Global.asa 文件
Visual InterDev Web 项目	把用户加到 Visual InterDev Web 权限列表上

使用源控件

在开发期间对你的 Web 应用采用源控件是在设计期间增添安全措施的另一途径。在你的 Web 服务器上加上 Visual SourceSafe 有助于控制谁可以在 Web 应用中检出文件和修改文件。

运行期间安全方针

在缺省安装微软 Internet 信息服务器和 FrontPage 服务器扩展版本之后，FrontPage 服务器上的 Web 应用可能被可以对服务器产生 HTTP 请求的任何浏览器匿名阅读。为了保证 Web 应用不被这样不受限制的访问，Visual InterDev 作者使用运行期间安全机制。

作为开发过程的一部分，作者选用阅读他们 Web 页面的浏览器定期测试他

们的 Web 应用。由于这个原因，作者甚至在创作的时候也必须考虑运行期间的安全问题。

注意：如果你用于产品的服务器同用于开发和测试的服务器是同一个服务器，那么运行的安全就变得同产品安全一样了。然而，如果一个 Web 应用放到一个不同服务器的产品中，那么在这个产品服务器上的安全设置就值得研究，在需要的时候要进行调整。

例如，如果该产品服务器有 FrontPage 扩展版本，并且 Web 应用使用 Copy Web 命令被传递到该产品服务器上，安全设置就需要通过 Visual InterDev 重新建立。如果产品服务器是一个 Internet 信息服务器，而没有 FrontPage 扩展版本，安全机制就应通过使用管理 Web 应用内容的文件和目录 ACD 的 Windows Explorer 来建立。

运行期间的安全管理，首先决定是控制所有用户还是只允许注册用户浏览 Web 应用，第二是控制对哪些用户授权，以便去访问只允许注册的浏览器才能访问的 Web 应用。

浏览许可

FrontPage 安全模式便于最终用户使用两种访问级别进行访问：无限制浏览与有限制浏览。

无限制浏览

无限制浏览是新 Web 应用的默认设置。所有用户都可以阅读（浏览）不受限制浏览的 Web 应用。请求阅读 Web 页面的每个最终用户都可以作为匿名用户；也就是说，最终用户在得到页面之前不需要对他们检验。

注意：因为从不要求最终用户标识自己，因此他们对 Web 应用是否有浏览访问权，不加区别。对无限制浏览的方式进行解释是用 Internet 信息服务器的“允许匿名”设置实现的，参见本章后边的“运行期间安全考虑”一节。

有限制浏览

使用 Web 权限对话框，任何根 Web 或子 Web 应用都可以设置成只对注册用户的有限制浏览。FrontPage 扩展把对匿名用户的阅读访问权从 Web 应用的所有内容文件中去掉。于是，匿名最终用户对阅读 Web 应用页面的要求被 Internet 信息服务器所拒绝。最终用户必须在 Windows NT 中对他们的标识得到授权，授权的用户帐号必须在请求得到满足之前拥有对 Web 应用的浏览访问权。当一个用户得到对 Web 应用的浏览访问权的时候，FrontPage 扩展版本把用户帐号阅读访问权赋予该 Web 应用的所有内容文件（参见本章后边的“运行期间安全考虑”）。

为了对可以浏览 Web 应用的人进行完全控制，要限制 Web 应用，以便只允许注册的用户才能浏览，然后再授予特定用户或用户组对 Web 应用的浏览访问权。

运行期间安全实现细节

FrontPage 扩展版本使用 WWW 服务器软件的基础安全模式来管理运行期间的 Web 权限。这里我们只考虑一种情况，就是在带有 NTFS 文件系统的 Windows NT 上运行的“Internet 信息服务器”3.0。在非“Internet 信息服务器”上的安全超出了本题目范围。在非 NTFS 文件上的安全性质要严格限制。

FrontPage 运行期间安全

FrontPage 对运行期间安全的管理同设计期间安全的管理不同。这是因为运行时 HTTP 要求阅读的页面不指向 ISAPI DLL，而设计时要求指向这里。Internet 信息服务器直接到达被请求的文件，并且企图阅读它，如下述那样。由于这个原因，FrontPage 直接修改 Web 应用中内容文件上的 ACL，以便控制运行期间的安全。

运行期间安全考虑

运行期间，你可能希望允许每个人都阅读 Web 应用的页面，或者局限于某些受权用户进行访问。

任何 Web 访问者都允许访问

你可以把你的系统设置成允许任何访问者进行访问。下表列出一些典型设置，你可以使用这些设置控制 Web 应用的安全。其它设置和设置结合对某些 Web 访问者可能产生运行期间错误。

设置位置	内部网设置	Internet 设置
操作系统 (Windows NT) / Web 服务器 文件夹	允许匿名 通过 Web 服务器处理权限	允许匿名 把匿名 ID 加到相应用户组上，通过 Web 服务器处理权限
数据库 Web 应用 页面	SQL 服务器：标准的 无规定 无规定	SQL 服务器：标准的 无规定 无规定

无限制浏览

当一个 Web 应用设置成允许无限制浏览时，FrontPage 就把匿名用户帐号标准阅读访问权赋予那个 Web 应用中的所有有关内容文件。

限制 Web 访问者的访问权

你可以通过在一个方面设置安全选项的办法限制对你的 Web 应用的访问权。有些开发者组建一个站点，这个站点有一个 Web 应用，这个应用对介绍和链接到另一个 Web 应用上的一般内容不限制访问，但是这另一个 Web 应用却是只对受权的访问者开发的。这第二种 Web 应用一般要求访问者进行登录，以便对其 ID 和口令同注册用户数据库进行比较。

你还可以参考下表选择安全选项，使其适应于你的应用。

设置位置	内部网设置	Internet 设置
操作系统 (Windows NT) / Web 服务器 (IIS)	Windows NT Challenge/Response (NTLM) 为用户组或个人加权限； 用户组和口令设置必须同 操作系统一致	把匿名用户 ID 加到 相应用户组中
Windows Explorer 文件 夹 (OS) 数据库 Web 应用	把用户组或个人加到权限 表中 标准 Global.asa 文件	把用户组或个人加到 权限表中 标准 Global.asa 文件：标 识授权用户并为用户 保存会话方式
页面	使用 .asp 文件和服务器 脚本	使用 .asp 文件和服 务器脚本

注册用户

当一个 Web 应用设置成只允许注册的用户浏览它的时候，FrontPage 就把所有匿名用户帐号权限从该 Web 应用内容文件中去掉。因为 Internet 信息与服务器与个人无关，从而在所有请求期间匿名用户也就不具有用户资格（在允许检查匿名的时候），这就引起标识失败问题，并开始如上所述的授权过程。

HTTP 传输安全 (SSL)

SSL (Secure Sockets Layer, 安全软插口层) 是一种协议, 用于保护 HTTP 传输不被意外窃听。在设计的时候和完成 Web 复制操作的时候, Visual InterDev 支持在客户/服务器链接上使用 SSL 选项。你还可以在运行的时候使用 SSL。在你创建一个新 Web 项目或启动一个 Web 复制操作的时候, 选择使用一条 SSL 连接。

若一条 SSL 连接正在使用中, 在传输之前, 所有 HTTP 请求和回答都是加密的, 在收到之后解密。这个过程不要求额外时间而降低性能。

除了这个加密/解密步骤之外, 所有 Visual InterDev 运作都严格同在非 SSL 连接上一样进行。你必须在可以启动一条 SSL 连接之前在你的 Web 服务器上获得并安装 SSL 验证。关于在 Internet 信息服务器上获得并安装 SSL 签证的更多内容, 参见 Internet 信息服务器文档中的 “Securing Your Site Against Intruders” (保护你的节点不被入侵)。

管理与维护 Web 应用的安全

对你的 Web 应用和系统选择相应设置的时候, 你还可能考虑到在维护安全规划方面的管理需求问题。

例如, 如果你选择使用 Windows NT Challenge Response, Web 权限就要以在那个服务器上定义的个人和用户组为基础。这项设置对内部网也能很好运行, 因为用户情况就像一般情况那样组建和维护, 而且用户数量也是知道的。Web 权限可以与网络权限同时提供。

然而, 如果你的 Web 应用正在 Internet 上运行着, 而且你使用 Windows NT

Challenge Response，用户情况的大量潜在问题使得对用户的管理成为系统管理员的全天候工作。还有，操作系统一般对你可以指定的用户配置文件数目有限制。

简化运行期间安全措施

像在设计期间的安全问题情况一样，你可以简化运行期间安全管理任务，办法是在 Web 服务器上建立一个本地组，比如 Web_Readers，给该组对根 Web 应用的浏览访问权。一定要确实限制根 Web 应用，只让注册的用户可以浏览它。在创建新的子 Web 应用的时候，保持它们的缺省设置，从该根 Web 应用继承权限。

现在把批准一个新用户在服务器上浏览 Web 应用的任务简化成使用 Windows NT User Manager 把这个新用户名字加到 Web_Readers 组中就可以了。关于添加用户组的更多内容，参见第九章“增添安全措施”中的“设置 Web 应用权限”一节。

管理多 Web 应用的安全

以一个个 Web 为基础，设置并施加权限设置。你可以创建子 Web 应用，像前边说的那样，从根 Web 中继承权限，从而节省设置时间。

根 Web 权限的任何改变都施加于从该根 Web 继承权限的所有子 Web 应用。管理员可以不直接改变子 Web 应用上的权限，直到它被设置成拥有自己的独有权限为止。

有效管理权限

为了方便，Windows NT 允许定义用户集合名称，叫做“组”（groups）。一个组一旦定义出来，而且用户已经加入其中，组中的每个用户就可以通过改变该组对一项资源权限的办法，允许或拒绝对该资源的访问。

第七章 本地工作

在 Visual InterDev 中，当你以在线模式工作时有两种选项。你可以用主模式，直接用主 Web 文件进行工作；也可以对 Web 应用开发自己的工作模式，即属于本地模式的一部分。

在本地模式中，你可以作出改变，获得 Web 应用的允写副本，而不影响主 Web 应用。你可以使用本地模式功能，制作 Web 应用新版本原型，而不改变你的主 Web 文件，或不干扰其他开发者所做的工作。

除了本地模式和主模式之外，如果因为某种原因你的工作不需要同主 Web 服务器连接，你可以选择离线方式工作。例如，你可以把你 Web 应用的本地副本放到膝上计算机中，然后离线对文件工作。更多内容，参见本章后边的“离线工作”一节。

7.1 单独开发应用

你可能希望在你的 Web 应用上工作的时候处于隔离方式，不改变你的主 Web 应用，也不受其他 Web 开发者改变主 Web 文件的影响。

在 Visual InterDev 中，你可以使用本地模式改变 Web 页面，然后测试你的页面如何同几个其它页面连带工作，而不影响主文件或其它开发者。如果你

的主 Web 服务器是你的产品服务器，你对本地 Web 文件作出的任何改变都不会影响你的 Web 访问者。你可以打开多个文件，改变这些文件，作为一个本地 Web 应用测试它们，然后发送你的本地副本去更新主 Web 应用。其他开发者或你的访问者只在这些文件发送之后才能看到改变。

你可以使用下述主要步骤进行本地工作，标识你可能希望做的其它任务。每个主要步骤的详细过程在有关题目中叙述。

本地工作

1. 为 Web 应用打开或创建一个项目。更多内容，参见第一章“Web 项目管理”中的“创建一个 Web 项目”一节。
2. 保证你的项目设置为本地模式。有关选择一种模式的更多内容，参见下一节“规定项目模式”。
3. 保证对你希望使用的文件有一个本地副本。关于得到最新版本或文件本地副本的更多内容，参见本章后边“获得本地主副本”一节。
4. 打开文件，编辑这些文件，再保存你的改变。因为你的项目在本地模式中，这些改变只保存在你的本地 Web 应用中；主 Web 应用不予改变。更多内容，参见第二章“Web 基础”中的编辑，并保存你改变的过程。
5. 如果你移动了文件或手动添加了链接，你可以使用 Link View 迅速查看检查这些链接。关于使用 Link View 的更多内容，参见第十一章“站点设计”中的“链接检验”一节。
6. 在 Web 浏览器中预览 Web 应用的工作版本。如果在你的开发者工作站上没有 Web 服务器软件，这个文件就用文件 URL 装载，而且服务器脚本也不

工作。如果你有一个本地 Web 服务器，你的 .asp 文件就使用这个本地 Web 服务器处理该服务器脚本。一个虚拟根自动建立起来，页面装载到 Web 浏览器中。关于浏览的更多内容，参见第二章“Web 基础”中的“预览页面”一节。

7. 更新主 Web 应用。当你对工作版本满意的时候，可以发送你的本地副本去更新主 Web 应用。更多内容，参见本章后边的“更新主 Web 应用”一节。

7.2 规定项目模式

你可以在本地模式和主模式之间转换你的项目。本地模式是推荐模式，因为它允许你同主服务器“隔离”开来工作。本地模式保证你文件的本地副本不受其他开发者对主 Web 应用作出更新的影响。这个模式也保护你的变更不受其他人使用主 Web 应用的影响。

为你的项目规定本地模式

1. 在 Project Explorer 中，选择你希望使用本地模式的项目。
2. 在 Project 菜单中选择 Web Project，然后选择 Working Mode（工作模式）。
3. 在子菜单中选择 Local（本地）。

如果你单独工作，不需要使用主服务器作为你的产品服务服务器，就可以直接在主模式中工作，不会同使用该应用的其他人发生冲突。

转换成主模式

1. 在 Project Explorer 中，选择你希望转换到主模式的项目。
2. 从 Project 菜单中选择 Web Project，然后选择 Working Mode。
3. 在子菜单中选择 Master。

7.3 获得本地主副本

要进行本地工作和测试，你需要把你的 Web 文件复制到本地 Web 应用中。你可以从主 Web 服务器上的主版本得到副本。

你有一种选择，可以得到文件的允写副本或只读副本。如果你希望编辑这些文件，就要获得文件的允写副本。要测试你的 Web 应用，还需要得到在测试期间你可能希望使用的有关文件副本。这些有关文件可能是允写的或者是只读的。

获得文件的允写副本

1. 在 Project Explorer 中，选择你希望编辑的文件。
2. 从 Project 菜单中点 Web Files，然后选择 Get Working Copy。

注意：如果你使用源控件，你可能需要在 Project 菜单中的子菜单 Source Control 上选择 Checkout（检出）。

文件的允写工作副本位于你本地 Web 应用中。有了本地 Web 应用中文件的允写副本，你就可以开始编辑，把文件保存到应用中。关于这些操作的更多内

容，参见第一章“Web 项目管理”；第二章“Web 基础”和第四章“编辑基础”。

要使用相关路径规定的链接测试页面，你就需要把相关文件从主 Web 应用中复制到本地目录中。例如，你希望测试 MyFile.htm 文件中的链接，这个链接指向 TargetFile.htm 文件，你就需要在你的本地 Web 应用中有 TargetFile.htm 的副本。如果一个 Web 应用被复制到另一个 Web 服务器中，同一个应用中多个文件之间链接的相关路径是推荐路径，因为它们是简便路径。

获得相关文件

1. 在 Project Explorer 中选择你希望的相关文件。
2. 从 Project 菜单中点 Web Files，然后选择 Get Latest Version。

注意：如果使用源控件，你可能需要在 Project 菜单中的子菜单 Source Control 上选择 Get Latest Version。

这个最新主版本复制到你的本地 Web 应用中。如果本地文件是只读的，便只在你的本地 Web 应用中阅读。如果你有一个允写副本，而且主版本是较新的，你便得到提示，把这两个版本归并到一起，用主版本取代本地版本，或者跳过更新，保持本地版本。

你可能希望把整个 Web 应用的副本放到你的本地计算机中。根据你的应用大小，这个操作可能需要一定时间和计算机的相当空间。

把整个 Web 应用复制到本地计算机上

1. 在 Project Explorer 中选择你希望得到的项目。
2. 在 Project 菜单中，点 Web Project，然后选择 Refresh。
3. 在 Project 菜单中，点 Web Project，然后选择 Get Latest Version。

注意:如果使用源控件,你可能需要在 Project 菜单中的子菜单 Source Control 上选择 Get Latest Version。

如果你有任何允写副本,而且主版本是较新的,就会得到提示,选择归并、取代或跳过。

7.4 预览 Web 应用

当准备测试你的本地 Web 应用时,你可以在 Web 浏览器中查看它的文件。预览你的本地 Web 应用,将允许你在发出副本给主服务器之前测试其功能。你还可以在一个 Web 浏览器中运行该应用之前使用 Link View 直观测试你的链接。关于测试链接的更多内容,参见第十一章“站点设计”中的“链接检验”和第十六章“维护链接”。

在你的本地 Web 应用原型中测试功能的可用性取决于你本地计算机中的系统部件。你总可以使用本地 Web 应用测试你的 .htm 文件和你的客户脚本。

然而,为了使用服务器脚本充分测试 .asp 文件,你需要有一个支持 .asp 文件的 Web 服务器本地副本。关于获得本地副本的更多内容,参见本章前边的“获得本地主副本”一节。关于可以测试什么的更多内容,参见第六章“Web 项目概念”中的“测试期间部件的相互作用”一节。

只要你拥有到主 Web 服务器的连接,就可以检验到其他文件的链接,而不必得到这些文件的本地副本。甚至在本地模式中,Visual InterDev 也可以接触主 Web 服务器,检验链接。

然而为了在离线工作的时候测试链接,你需要得到在本地 Web 应用中使用

的文件副本，因为到该主 Web 服务器的连接已经不能使用了。

预览本地 Web 应用

- 在 View 菜单中选择 View in Browser

如果 Web 服务器同本地 Web 应用常驻于同一个计算机中，就可以测试你的服务器脚本以及你的客户脚本和 HTML。如果在同一台计算机上没有 Web 服务器，你只能测试你的客户脚本和 HTML。服务器脚本要求 Web 服务器对它进行处理。

你可能希望回顾一下主 Web 应用的当前版本是如何运行的。

预览主 Web 应用

- 如果你的项目是处于本地方式中，则需要打开 Web 浏览器，为主 Web 服务器上的 Web 应用指定 URL。

或者

如果你的项目处于主方式中，从 View 菜单中选择 View in Browser。

7.5 更新主 Web 应用

在你已经把变更保存到本地 Web 应用之后，并做好准备把变更送到主 Web 应用的时候，可以发出工作副本。

更新主 Web 应用

1. 在 Project Explorer 中，选择你希望保存到主 Web 应用中的文件。

2. 在 Project 菜单中，点 Web Files，再选择 Release Working Copy。

你的文件本地副本被复制到主 Web 服务器中，并变成主 Web 应用上的当前版本。如果你得到变更的副本之后，还有人对主版本作出变更，则 Merge（归并）对话框就出现，允许你审视一下差别，决定接受还是拒绝变更。详见第八章“同多个开发者一起工作”中的“解决归并冲突”。

你可能不希望变更传播到主 Web 服务器上。例如，你可能已经决定认为当前版本处理脚本变成策略形式比刚刚对本地文件作出的变更要好一些。你可以抛弃当前的允写副本，机智地留下主版本。

抛弃对本地副本的变更

1. 在 Project Explorer 中，选择你不希望发给主 Web 应用的文件。

2. 在 Project 菜单中，点 Web Files，然后选择 Discard Changes（抛弃变更）。

你选择的每个文件在本地的副本都被该文件的主版本所取代。

7.6 更新本地 Web 应用

在本地方式工作期间，你的 Web 应用文件的本地副本都被保护起来，避免工作组其他成员对主 Web 应用作出变更的影响。你可能希望知道其他 Web 开发者作出的变更。例如，其他人可能变更了已经存在的文件，或者已经把新文件添加到主 Web 应用中。如果你希望从主 Web 服务器上得到最新文件列表，可以刷新项目。如果你希望得到在本地已经有的文件的最新版本，可以把本地 Web

应用与主 Web 应用同步起来。还有，如果你最近打开一个 Web 项目之后，又有人把源控件用到该项目上，就需要刷新项目，使用源控件特性的优点。

获得项目结构的变更

1. 在 Project Explorer 中，选择你希望刷新的项目
2. 在 Project 菜单中，点 Web Project，然后选择 Refresh Project View。
在 Project Explorer 中的文件列表和本地只读文件都被更新，反映着主服务器上应用的当前方式。尽管 Project Explorer 视图已经改变，甚至那个文件的主版本都改变了，本地文件的允写副本并没有改变。
例如，你有一个允写文件的本地副本，而那个允写文件已经从主应用中删除了，你的本地副本依然同以前一样保留着。然而，本地文件的只读副本被更新以反映主服务器上作出的任何变更。例如，下述变更：新版本、删除文件和文件重命名，都是在本地完成的。

注意：刷新项目视图不取代你计算机上允写文件的新本地副本。关于得到最新文件版本的信息，参见第十章“管理 Web 项目”中的“主文件与本地文件同步”一节。

你可能希望从主服务器得到更新的文件，但是不希望得到整个 Web 应用。你可以使特定文件的本地版本与主版本同步。

获得变更的文件

1. 在 Project Explorer 中，选择带有你希望得到的文件的项目。
2. 在 Project 菜单中，点 Web Project，然后再选择 Synchronize Files。

7.7 离 线 工 作

你可能希望把膝上计算机带到会议上，演示你的应用的当前版本。你可以在没有连接到主 Web 服务器的情况下，把建议和变更权利在会议上结合起来。在项目处于离线状况下，你可以打开项目，编辑文件的本地（工作）副本，浏览变更等，却没使用到主服务器的连接。

离线工作

1. 打开一个项目或为 Web 应用创建一个项目。更多内容，参见第一章“Web 项目管理”中的“创建 Web 项目”。
2. 保证你希望工作的文件有一份本地副本。关于得到文件的最新版本或本地副本的信息，参见本章前边的“获得本地主副本”或“更新本地 Web 应用”。
3. 把你的项目置为离线工作模式。更多内容，参见下边步骤。当你的项目处于离线模式时，你在项目上的工作就像在本地模式一样，所不同的是更新、重构或需要主 Web 应用中信息的各种命令都不能使用。例如，你不能得到最新版本、发送工作副本、或在项目中移动文件。
4. 打开文件、编辑文件并保存变更。因为你是在离线模式中工作，这些变更只能保存在本地 Web 应用中。要了解更多内容，参见第二章“Web 基础”中编辑，并保存你变更的步骤。
5. 在 Web 浏览器中预览 Web 应用的本地版本。如果在你的计算机中没有 Web 服务器软件，该文件就要用文件 URL 装载，服务器脚本也不能工作。如果

你有 Web 服务器软件，你的 .asp 文件就使用本地 Web 服务器软件处理这个服务器脚本。一个虚拟根自动建立起来，页面也装入这个 Web 浏览器中。关于预览的更多内容，参见第二章“Web 基础”中的“预览页面”一节。

6. 把你的项目设置为离线工作模式。如果在希望转换到离线模式时，你的项目处于本地模式中，你可能希望在你返回在线模式以前更新主 Web 应用。如果项目处于主模式中，在你返回在线模式时，主 Web 便自动更新。要得到更多内容，参见本章前边的“更新主 Web 应用”一节。

把你的项目设置为离线模式

1. 在 Project Explorer 中，选择你希望离线工作的项目。
2. 保证你在本地拥有希望工作文件的副本以及在本地 Web 应用中进行测试所需要的相关文件。关于获得文件的最新版本或允写副本的更多内容，参见本章前边“获得本地主副本”的步骤。
3. 在 Project 菜单中选择 Web Project, 选择 Working Mode, 再选择 Work Offline。

注意：如果你希望对站点结构、重命名文件作出变更，或移动文件，就需要使用在线模式。

第八章 同多个开发者一起工作

不论你是在工作组中工作还是单独工作，都很容易地使用 Visual SourceSafe 追踪并保存你作出的变更。当在你的 Web 应用上使用源控件时，你可能每天要执行一些任务，有些任务需要偶尔执行，有的任务可能只需执行一次。

8.1 在 Web 服务器上组建源控件

要安装 Microsoft Visual SourceSafe，把它设置成同 Visual InterDev 一同工作，你就需要使用一对不同接口完成不同的过程。

第一步涉及到使用 Microsoft Windows 和 Visual SourceSafe Setup 在主 Web 服务器上安装 Visual SourceSafe。

在你的 Web 服务器上组建源控件

1. 在主 Web 服务器上安装 Visual SourceSafe。更多内容，参见下一节“在 Web 服务器上安装 Visual SourceSafe”。
2. 在 Visual SourceSafe 中，把 Visual SourceSafe 权限授予用户。更多内容，参见下边“把权限授予用户”一节。
3. 对于 Windows NT 上的 Web 服务器，把权限加到匿名用户帐号上。更多内

容，参见下边“把权限加到匿名用户帐号上”一节。

4. 在各个 Web 项目之间共享源控件。

在 Web 服务器上安装 Visual SourceSafe

在安装 Visual SourceSafe 的时候，要保证把 Visual SourceSafe 服务器软件和主 Web 服务器软件安装到同一个计算机上。要做到选择的选项同其它可视产品综合的选项。没有这个选项的设置，Visual SourceSafe 便独立运行，不能辨别由 Visual InterDev 使用的命令。如果你的规划要对文件进程 Visual InterDev 进行检查，那么你就不需要在你本地计算机上安装 Visual SourceSafe。

下边的过程假设你已经把 Microsoft Internet 信息服务器和 FrontPage 服务器扩展版本安装到你的主 Web 服务器上。

在主 Web 服务器上安装 Visual SourceSafe

1. 运行 Visual SourceSafe 安装程序。

2. 选择 Custom，要保证至少安装下列部件：

- Create SourceSafe Database
- Administrative Programs
- Enable SourceSafe Integration

3. 如果你运行的是 Windows 95 和微软“个人 Web 服务器”，就下载并安装“Distributed Component Object Model” (DCOM，分布式部件对象模型)。

如果你需要下载“分布式部件对象模型” (DCOM)，就到微软公司 Web 站点 <http://www.microsoft.com/search/default.asp> 上，并且查找“DCOM”。

当你在 Web 服务器上安装了 Visual SourceSafe 之后，必须把读/写权限赋予你希望能够使用 Visual InterDev 或微软 FrontPage 作者文件的所有用户。

把权限赋予用户

1. 运行 Visual SourceSafe Administrator 应用。这个应用应当出现在 Start 菜单的 Programs 部分。如果不出现，检查你是否安装了上边列出的所有部件。
2. 选择 Users 菜单，然后选择 Add User。
3. 键入用户名字，然后，如果不要求，就空着 Password 框。要保证没选择 Read-Only 选项。

注意：如果你分配了口令，用户在使用项目时就多了一个登录步骤。

如果在 Windows NT 服务器上安装了 Visual SourceSafe，则除了对用户授予读/写权限，还必须对匿名用户帐号授权。

把权限加到匿名用户帐号上

1. 在服务器上打开 Internet Information Manager。
2. 在微软 Management Console 上选择 Internet Information Manager，然后选择运行 Visual Source Safe Administrator 的计算机。
3. 选择 Default Web Site (默认 Web 站点)，并打开其相应的对话框。
4. 在 Directory Security 选项卡上，选择 Edit 来改变 Authenticaion Methods。
5. 在 Anonymous User Account 对话框中，复制用户名。

注意：作为默认值，这个特性设置是 IUSR_*machinename*，这里的 *machinename*

是服务器名字，你在这里安装了 Visual SourceSafe Administrator。

6. 在 Visual SourceSafe Administrator 应用中，通过选择 Users 菜单中的 Add User，并作为 UserName 特性的一部分，为 Anonymous 用户创建用户帐号。让 Password 框空着，并保证不选择 Read-Only 选项。

在 Visual SourceSafe 踪迹变更的时候，操作系统进行辨别，并记录是谁作出的变更。

有的操作系统只能辨别并记录作出变更的匿名用户名字。如果你的 Web 服务器正在运行 Windows 95, 或使用 File Allocation Table (FAT) 文件系统的 Windows NT, 那么通过 Visual SourceSafe 检查的所有文件都要对相同用户帐号进行检查。这个用户帐号可能不代表执行该操作的用户。

在 Windows NT FAT 系统中，匿名用户帐号执行服务器上所有源控件操作。在 Windows 95 系统中，用户帐号指出 Windows 95 在什么时候开始在服务器上执行所有源控件操作的。

8.2 把源控件加到 Web 应用上

你一旦安装了微软的 Visual SourceSafe，你就可以对某个 Web 应用开启源控件，使用任何可以引用那个应用的 Web 项目就可以做到这点。只是一个开发者需要对那个应用开启源控件。

Visual SourceSafe 提供两种方式控制源文件。

- **特殊检查**。这是一种默认选项，这种选项只允许一个用户一次检查一个文件。
- **多重检查**。一种选项，这种选项允许多个人一次检查同一个文件。关于解决由于多重检查引起的冲突的有关信息，参见本章后边“解决归并冲突”一节。

在某个开发者对一个 Web 应用开启源控件以后，打开项目引用该 Web 应用的其他开发者必须刷新或重新打开他们的 Web 项目，以便使源控件对这些项目产生影响。

对 Web 应用开启源控件

1. 在 Visual InterDev 中，打开或创建一个 Web 项目，这个项目引用你希望置于源控件之下的 Web 应用。
2. 在 Project Explorer 中，选择你希望被源控件所使用的项目。
3. 从 Project 菜单，选择 Source Control（源控件），并选择 Add to Source Control。
4. 在 Enable Source Control 对话框中，检验项目名称是你所希望的对应于 Source Control Project 的名称，然后单击 OK。
对于 Visual SourceSafe 项目，你可以键入不同的名字（如果需要的话）。如果你用了不同名字，名字前边必须加一个美元符号和一个正斜杠 (/)。例如：\$/MyWebApplication。

如果你希望开启多重检查，就需要对主 Web 服务器的 Visual SourceSafe Administrator 有管理员特权。

对一个 Visual SourceSafe 项目进行多重检查

1. 在 Visual SourceSafe Administrator 中，在 Tools 菜单中选择 Options。
2. 在 General 选项卡中，选择 Allow multiple checkouts。
3. 单击 OK。

如果你不知道一个 Web 应用是否拥有已开启的源控件，可以对引用那个 Web 应用项目的特性表进行检查。

对某个 Web 应用确定是否已经开启源控件

- 在 Project Explorer 中，检查在项目名字后边是否出现锁图标。
如果项目在源控件之下，锁图标将出现在项目名字后边。
如果项目不在源控件之下，将不出现锁图标。

你还可以从引用 Web 应用的任何 Web 项目关闭对那个 Web 应用的源控件。

对 Web 应用开启源控件

1. 打开或创建一个新 Web 项目，这个项目引用的 Web 应用是你希望从源控件中去掉的那个应用。
2. 在 Project Explorer 中，选择你希望从源控件中去掉的项目。
3. 从 Project 菜单选择 Source Control, 然后选择 Disconnect Web Project。

当你关闭对某个 Web 应用的源控件的时候，在你第一次开启 Web 应用源控件时创建的 Visual SourceSafe 项目仍然保留在 Visual SourceSafe 服务器上。这就意味着你可以使源控件重新开启，并为你的 Web 应用选择那个 Visual SourceSafe 项目。

使源控件关闭之后再使其开启，可能会引起一些预料不到的问题。例如，如果你关闭源控件，然后从 Web 应用中删除文件，然后重新使源控件开启，你删除的文件将会再次出现在 Web 项目中。出现这种情况是因为被删除的文件在源控件之外，因而原来的 Visual SourceSafe 项目仍然把它们看作项目的组成部分。你可以删除 Visual SourceSafe 项目，办法是从 Visual SourceSafe Explorer 中删除它。

8.3 在源控件下工作

一个 Web 应用一旦有了开启的源控件，你就检出检入文件，而不是像以前做的那样获取和发送工作副本。在微软 Visual InterDev 中通过 Web 项目检入检出 (check in and out) 文件，当主 Web 应用和 Visual SourceSafe 存储器由本地 Web 应用的较新版本进行更新的时候，工作组成员便拥有更多的控件。

注意：如果你使用 Visual SourceSafe 客户软件而不是用 Visual InterDev 特性去完成源控件任务，要保证在你的工作继续之前刷新你在 Visual InterDev 中的 Web 项目。

向源控件开启的 Web 应用添加文件

一旦使某个 Web 应用的源控件开启，你使用 Visual InterDev 向 Web 应用创建或添加的任何文件被自动加到 Visual SourceSafe 项目上。这包括在任何 Visual InterDev 编辑器中创建的文件或在其它应用中创建的并加到 Web 项目中的文件。

注意:如果你通过文件程序把文件直接加到服务器的项目目录中, 这些文件便不能自动加到 Visual SourceSafe 项目中。为了把这些文件加到 Visual SourceSafe 项目中, 你需要把本地 Web 应用同主 Web 应用同步起来, 刷新项目视图。

使本地应用和主 Web 应用同步

1. 在 Project Explorer 中选择 Web 项目。
2. 在 Project 菜单中选择 Web Project, 然后选择 Synchronize Files。
你的 Project Explorer 视图被刷新, 包含在该项目目录中的所有文件都出现在 Project Explorer 中。

如果你看见带有红色“X”的图标, 相关文件就没在 Visual SourceSafe 项目中。

如果在你的 Web 项目中有一个文件, 而由于某种原因这个文件不在源控件之下, 你可以手工加上源控件。

把一个文件手工添加到源控件上

1. 在 Visual InerDev 中, 选择你希望加到源控件上的 Project Explorer 中的文件。
2. 从 Project 菜单中, 选择 Source Control, 然后选择 Add to Source Control。

从源控件中去掉文件

1. 在 Visual InterDev 中, 选择你希望加到源控件上的 Project Explorer

中的文件。

2. 从 Project 菜单中，选择 Source Control，然后选择 Remove From Source Control。

关于使用 Visual SourceSafe 的更多内容，参见本章后边“调试源控件”一节。

检出文件

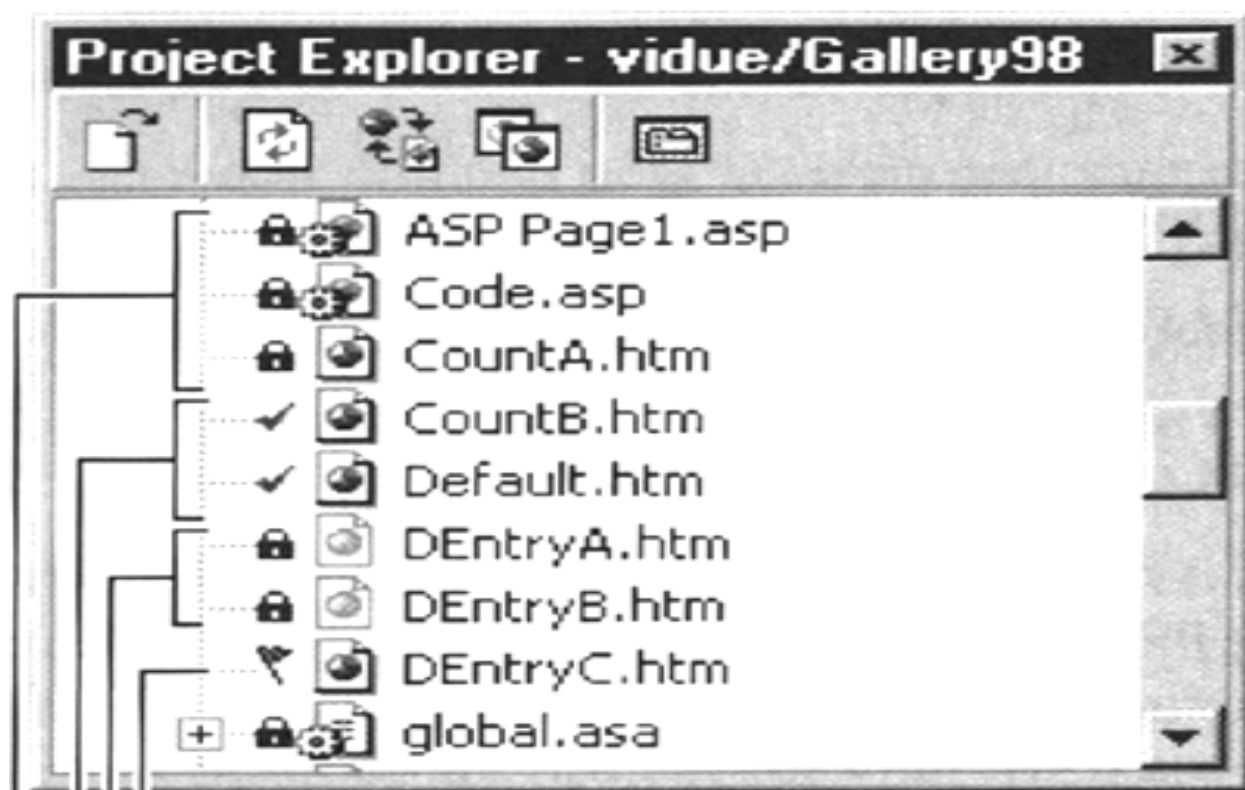
你可以不运行 Visual SourceSafe 客户文件而直接在 Visual InterDev Project Explorer 中检出 (check out) 一个文件。作为默认情况，在你检出一个文件的时候不会有人获得这个文件的允写副本。如果你请求一个文件的副本，而这个文件是另一个人已经检出的，Visual SourceSafe 便显示一条消息，指出另一个人已经检出了这个文件，你可以得到这个文件的只读版本。

检出一个文件

1. 在 Project Explorer 中，选择你希望检出的文。

2. 从 Project 菜单中，选择 Source Control, 然后选择 Check Out。

在 Project Explorer 中在本地目录中允写的文件用彩色表示出来，而只读副本和没有任何本地副本的文件都带有灰色阴影。



在 Project Explorer 中的文件状态

当一个文件被检出时，你可以得到这个文件的允写副本，而 Visual SourceSafe 在 Project Explorer 中对这个文件加上被检出的标志。当你检入

它的本地副本时，这个文件加上检入标志，对该文件作出的所有变更都保存在主 Web 服务器中。

按照默认设置，将提醒你在该文件的允写副本或只读副本之间进行选择。如果你总是希望这种副本或另一种副本，则可以把 Visual SourceSafe 设置为总提供这种或另一种副本。

规定文件如何检出

1. 从 Tools 菜单中选择 Options。
2. 在 Options 对话框中选择 Project，然后选择 Web Projects。
3. 从 When opening a file 对话框中选择你希望的源控件选项。
4. 单击 OK。

允许多重检出

当你的工作组成员希望能够在同一个文件上同时工作时，你的 Visual SourceSafe 管理器通过允许多重检出的办法允许多个人检出并编辑某一个文件。这个选项允许多个人在一个文件上同时工作。

在允许多重检出时，检出该文件的每个用户都得到该文件的允写副本。如果你是检入这个文件的第二个人，你可以查阅差异，并把两个版本归并在一起。

允许多重检出

1. 在你的主 Web 服务器上打开 Visual SourceSafe Administrator。
2. 在 Tools 菜单中选择 Options。
3. 在 General 选项卡中选择 Allow Multiple Checkouts。

4. 单击 OK。

获得主版本文件

如果你希望查阅一个文件的最新版本，又不改变该文件，你可以取得一个主版本文件的副本，而无需检出该文件。当你取得一个文件的最新版本时，该主文件的只读副本被放置到本地 Web 服务器上。例如，你可能希望得到最新版本的文件是链接到你正在开发的文件上的文件。

获得主版本文件的副本

- 从 Project 菜单中选择 Source Control，然后选择 Get Latest Version。

如果你已经有了该文件的本地副本，Visual InterDev 就显示一个对话框，在这个对话框中你仍然选择这个已经有的本地文件，或选择用其主副本取代它。于是所请求的文件副本就放到你本地 Web 应用上。

注意：如果本地目录不与主 Web 目录保持同步，这个命令就失败。如果出现这种情况，使用 View 菜单中的 Refresh 命令同步 Web 目录中的项目，然后再试一试刚才的命令。

检入文件

当你检入 (check in) 一个文件时，你用本地 Web 目录中的已经修正的文件版本取代 Visual SourceSafe 中当前文件版本。Visual InterDev 自动同 Visual SourceSafe 相互作用来检入这个文件；你不必执行任何其它步骤。

检入一个文件

- 从 Project 菜单选择 Source Control，然后选择 Check In。

注意：Check In（检入）命令不同于 Save（保存）命令。Save 命令在本地或主 Web 应用中更新文件，但是不更新 Visual SourceSafe 中的源控件版本。

在文件检入之后，你原来检出的版本放到那个文件的 Visual SourceSafe 的历史中。

如果你改变了这个文件，但是不希望把它保存在 Visual SourceSafe 中作为最新版本，则可以使用 Discard Changes（放弃变更）命令，放弃得到该本地副本以来作出的所有改变。

放弃对文件作出的改变

- 从 Project 菜单中选择 Source Control，再选择 Undo Check Out。

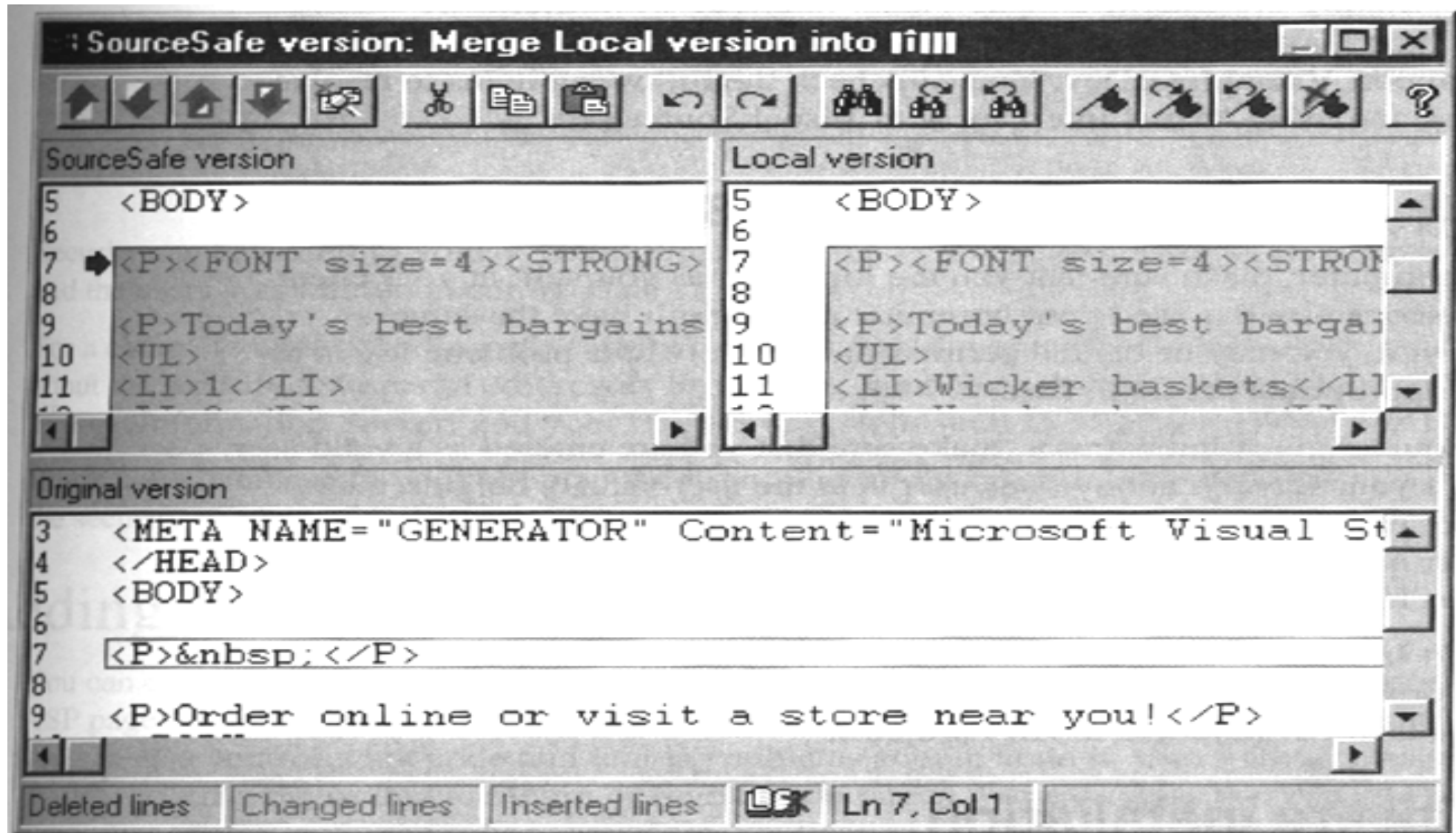
解决归并冲突

在允许多重检出的时候，如果工作组其他成员改变了你要检入的文件，你可能需要解决归并冲突。

当你检入一个文件的时候，而这个文件已经被其他人改变了，则 Visual InterDev 便检查归并冲突，并显示 Merge（归并）对话框，让你查看两个不同的版本。例如，你检出一个文件，以便改变某些客户脚本，而一个 Web 作者检出同一个文件，以便重写某些内容。如果你检出之前，这个 Web 作者检入这个被修改的文件，Web 作者的该文件版本便变成新主版本。当你检入你的该文

件副本时，便出现 Merge（归并）对话框，允许你把两个版本归并到一起，并保存该归并的文件作为最新版本。

如果有人在你得到你改变的副本之后对该主版本作出变更，那么在你发出你的本地副本时出现 Merge 对话框。在 Merge 对话框中，你可以查看这两个文件的差别，可以接受或者拒绝这些变更。



例如，在 Visual SourceSafe 中的版本是某个人保存在服务器中的主版本。Local version 反映最新本地副本。Original 版本栏中允许你编辑该文件，显示你已经作出的归并决策。

归并文件

1. 在底栏中，定位并选择你希望解决的冲突范围。
2. 单击右键，从菜单中选择你希望归并的变更。
3. 关闭对话框，选择 Save（保存）。

主副本和本地副本都反映你保存的变更。

你可以比较 Visual SourceSafe 中的版本与本地版本，查看它们的差别。

手工查看差别

- 从 Project 菜单中选择 Source Control，然后选择 Compare Versions。关于比较文件的更多内容，参见 File Difference Options 对话框。

8.4 调试源控件

如果你用 Visual SourceSafe 体验使用 Visual InterDev 中的困难，你可能需要在支持 Visual InterDev 的应用之内，在 Visual InterDev 之外进行一些改变。你可以检查下列项，解决以 Visual SourceSafe 打开并使用 Visual InterDev 的问题。

适当登录与授权

- 在本地计算机中，你要确保就像自己在客户机上一样进行登录。如果还有别人用过你的计算机，但是没有同你一样的权限，你可能被拒绝授权。要解决这个问题，就要像你自己做的一样进行登录，获得对文件的适当

访问权。

- 在 Visual SourceSafe Administrator 中，你要作为一个合法用户进入。还要保证你的服务器匿名用户在用户列表中。用默认名称，则匿名用户名字是 IUSR_<machinename>。如果你不知道你的计算机匿名用户名字，则可以在“Internet 信息服务器”配置中，在 WWW 服务属性内找到。
- 在“Internet 信息服务器管理器”中，要保证 Basic 授权或 NT Challenge/Response 是开启的。如果 No Authentication 选择被选中，则访问权不能用于修改项目。

安装适当软件

- 在你的 Web 服务器上，检查你已经安装的 FrontPage Server Extension、Active Server Page 和 Visual SourceSafe。如果 Visual SourceSafe 是用 Typical 或 Client 安装过程进行安装的，你就要重新使用“Custom”或“Server”进行安装，选择的选项是“Enable Source Safe Integration”。

Visual SourceSafe 连接

如果除了 Web 服务器之外，在你的本地计算机上还安装了 Visual SourceSafe，就要保证你的计算机到服务器之间的 LAN 连接在正常工作，并且设置了适当的权限。

在你本地计算机上不需要 Visual SourceSafe，你只需要 Visual InterDev。如果你希望使用 Visual SourceSafe Explorer，直接在 VSS 数据库中对你的 Web

文件进行工作，你就需要在你的本地计算机上安装 Visual SourceSafe 客户软件。

第九章 增加安全措施

可以在不同级别上把安全措施加到你的 Web 应用中。到底在哪个级别上增加安全措施，取决于你的系统和你希望接纳的用户。

要全面了解影响你 Web 应用的安全选项，请阅读下述各方面与安全特性相关的内容：FrontPage、你的 Web 服务器（例如 Internet 信息服务器）和你的操作系统（例如 Windows NT）。

如果你在 Web 应用中使用数据库，你可能希望熟悉 ODBC 数据库的安全模型，例如 Microsoft Access 或 SQL Server。

9.1 添加安全页面

你可以把 HTML 格式同客户脚本和服务端脚本结合起来，保证站点的安全。因为 ASP 页面先产生你的访问者可以看见的动态内容，然后才把这些内容发送给浏览器，因而你还可以控制在你的访问者浏览器中按照原代码可以访问的代码和文章。典型的安全页面包括登录和注册页面。例如，你可以添加登录页面，用于控制到应用其余部分的入口。

你为建立安全所添加的实际页面和脚本取决于你的设计目标。下边的步骤给出一种方式，用于在你的站点实现安全页面。

添加安全页面

1. 把 HTML 格式加到你主站点页面上，这个页面上有用户的标识和口令信息。详细内容，参见第二十五章“用 HTML 元素编写脚本”中的“用 HTML 格式收集信息”一节。
2. 在主站点页面上，添加客户脚本，用于检验在格式中键入的值。详细内容，参见第四章“编辑基础”中的“添加脚本”一节。
3. 在数据库中添加一个表，用于存放用户标识和口令信息。
4. 在项目中，添加到数据库和表的连接。详细内容，参见第三章“数据库基础”中的“连接数据库”一节。
5. 创建一个 ASP 页面，在页面上根据用户的输入接收一个记录，并把输入值同数据库中的值进行比较。
6. 添加一个脚本，用于处理链接该站点的失败登录和成功登录。

9.2 设置 Web 应用权限

Web 上的应用都有权利设置以标识用户并指定特权。在 Visual InterDev 中，你可以为你的应用设置设计期间权限。按照默认，一个新 Web 应用继承根目录同样的权限。你可以按照客户要求对这些权限进行增删，然后按照不同用户和用户组对这些权限进行控制。

设置 Web 权限涉及到以下步骤：

- 按 Web 对权限进行增删

- 对个人设置权限
- 对用户组设置权限
- 对计算机设置权限
- 简化设计期间安全机制
- 撤销权限

按 Web 应用增删权限

你可以对当前激活 Web 项目中的 Web 应用设置独有的权限。为 Web 应用设置独有的权限必须在你可能修改它的用户和用户组之前进行。

为 Web 应用设置独有权限

1. 在 Project Explorer 中，选择你希望设置权限的项目。
2. 在 Project 菜单中，选择 Web Project, 然后选择 Web applications。
3. 在 Settings 选项卡中，选择 Use unique permissions for this Web application (对这个 Web 应用使用独有权限)。这就规定当前的 Web 应用不从根 Web 应用中继承权限设置。
4. 选择 Apply。

注意：要变更这个选项卡中的独有设置，就须在其它选项卡可以变更之前进行。

对个人设计权限

在你为 Web 应用设置客户权限之后，就可以添加一个个用户，并控制他们的权限。

把用户加到 Web 应用上

1. 从 Projects 菜单上选择 Web Project, 然后选择 Web Permissions。
 2. 在 Setting 选项卡中选择 Use unique permissions for this Web application, 然后选择 Apply。
 3. 在 Users 选项卡中, 选择 Add。
 4. 在 Add Names 框中, 为新用户键入域名和用户名, 格式是: 域名/用户名。
或
在 Obtain list from 框中, 选择域名并从 Names 段中添加用户。
 5. 在 New users can 框中, 为新用户选择权限级别。
注意: 选择的权限级别施加到 Add Names 段中所有新用户上。
 6. 选择 Apply。
- 你可以为每个用户进行权限编辑。

改变用户权限

1. 从 Project 菜单中选择 Web Project, 然后选择 Web Permissions。
2. 在 Users 选项卡中, 选择用户编辑, 并选中 Edit。
3. 在 Edit Users 框中, 为用户选择新权限级别。
4. 选择 Apply。

为用户组设置权限

你可以控制对用户授予的读写权限。

注意:如果你的主 Web 服务器的操作系统使用 File Allocation Table 文件

系统 (FAT), 例如 Windows 95, 而不是 NT File System(NTFS), 你就不能为用户组改变权限。

把用户组添加到 Web 应用上

1. 从 Project 菜单中选择 Web Project, 然后选择 Web Permissions。
2. 在 Settings 选项卡中, 选择 User unique permissions for this Web application, 然后选择 Apply。
3. 在 Groups 选项卡中, 选择 Add。
4. 在 Add Names 框中, 键入新用户组的域名和组名, 格式是: *domain\group name*。

或者

在 Obtain list from 框中, 选择域名, 并从 Name 字段中填加组名。

5. 在 New users can 框中, 选择新用户组的权限级别。注意, 所选的权限级别要施加于 Add Names 字段中所有新用户组上。
6. 选用 Apply。

一个组的权限一旦给定, 就可以改变这个组的权限。

改变组的权限

1. 从 Projects 菜单中先选择 Web Project, 然后选择 Web Permissions。
2. 在 Groups 选项卡上, 选择要进行编辑的组, 并选定 Edit。
3. 在 Edit Groups 对话框中, 为该组选择新的权限级别。
4. 选定 Apply。

设置计算机权限

可以依据 IP 地址，控制分配给计算机权限的读出与写入。

注意：如果主 Web 服务器所在的操作系统使用 File Allocation Table(FAT)文件系统，如 Windows95，而不是使用 NT File System(NTFS)，则不能改变这种计算机权限。

把计算机加到 Web 应用上

1. 从 Projects 菜单先选用 Web Project，后选用 Web Permissions。
2. 在 Setting 选项卡上，选择 Use unique permissions for this Web application，然后选择 Apply。
3. 在 Computer 选项卡上选用 Add。
4. 在 IP Address 框中，键入计算机地址。

注意：你可以使用星号*规定一组计算机具有同一个开始地址。

5. 在 Computer permissions 框中，为该计算机选择权限级别。
6. 选择 Apply。

为计算机授予权限之后，你就可以改变这个计算机的权限。

改变计算机权限

1. 在 Project 菜单中选择 Web Project，然后选择 Web Permissions。
2. 在 Computer 选项卡中选择计算机或计算机掩码来编辑，选择 Edit。
3. 在 Edit Computer 对话框中，为用户组选择新权限级别。
4. 选择 Apply。

简化设计期间安全机制

你可以很容易地简化设计期间安全机制，办法是先修改 Web 权限，然后使用 Windows NT User Manager 实用程序顺序进行安全修改。

注意：如果你的主 Web 服务器所依存的操作系统使用 File Allocation Table 文件系统（FAT），例如 Windows 95，你就不能为用户组改变权限。

简化设计期间安全机制

1. 在你的 Web 服务器上的 Windows NT User Manager 中，创建两个新本地用户组。

例如，你可以为管理人员创建一个用户组，叫做 Web_Admns，再为开发者创建一个用户组，叫做 Web_Devs。

2. 在 Visual InterDev 中，为 Web 权限添加新用户组。

当你在这个服务器上创建新 Web 应用时，你可以选择，使这个新 Web 应用继承根目录的权限。

你可以把这个服务器中 Web 应用的管理员访问权或作者访问权赋予新用户，办法是使用 Windows NT User Manager 把他们的名字加到相应用户组中，而不用重新调用 Web Permissions 对话框。

然而，如果你想用不同权限维持服务器上单个子 Web 应用，而不是多个其它应用，你必须为它设置独有的权限，然后分别管理它的用户或用户组。

撤销权限

你可以通过删除用户列表或用户组列表中权限的办法撤销单个用户或用户

组的权限。

删除 Web 应用上用户或用户组的权限

1. 从 Project 菜单中，选择 Web Project，然后再选择 Web Permissions。
2. 选择要从 Users 选项卡中取消的用户，或选择要从 Group 选项卡中要取消的用户组。

注意：如果你的主 Web 服务器所依存的操作系统使用 File Allocation Table 文件系统（FAT），例如 Windows 95，而不是 NT File System(NTFS)，你就不能改变用户组的权限。

3. 选择 Remove。
4. 选择 Apply。

9.3 连接代理服务器

如果你的服务器允许你连接代理服务器，你可以用 Visual InterDev 进行连接。你可能希望在内部网之外的某个服务器上对 Internet Web 应用进行工作。例如，你可能需要改变你的公司防火墙之外某个 Web 服务器上的 Web 文件。

使用 Visual InterDev，你可以连接到代理服务器上，以便访问 Internet。你可使用的连接和能力取决于你所工作的系统和你希望连接的系统。

你可以规定项目所使用的代理设置与操作系统的 Internet 设置是同一套设置，你也可以规定一个不同的服务器。

规定代理服务器与操作系统是同一套装置

1. 从 Tools 菜单中，选择 Options。
2. 在 Options 对话框中，选择 Project，然后选择 Web Proxy Settings。
3. 在 Web Project Proxy 选项卡中，选择 Use system settings。

系统设置出现在该选项卡上，但是这是只读型的。要改变实际设置，你需要改变操作系统的 Internet 设置。例如，如果你已经使用了 Windows NT 工作站，那么你就要到控制面板上，在 Internet Properties 对话框中改变代理服务器的设置。

如果你不希望代理服务器与操作系统使用同一个服务器，你可以规定一个的服务器供 Web 项目使用。

只为 Web 项目规定一个代理服务器

1. 在 Tools 菜单中，选择 Options。
2. 在 Options 对话框中，选择 Project，再选择 Web Proxy Settings。
3. 在 Web Project Proxy 选项卡中，清除 Use system settings。
4. 键入你希望在 HTTP Proxy 框中使用的代理服务器名称。
5. 在 List of hosts without proxy 框中，键入你希望不经过代理服务器进行访问的任何代理服务器 IP 地址。
6. 如果你不希望使用代理服务器处理本地地址的话，选择 Do not use proxy server for local (intranet) address 框。

注意：在创建或复制一个 Web 文件的时候，你可以使用服务器名字，或明确键入它的 IP 地址来规定对象 Web 服务器。如果使用 IP 地址取代名字，

而且它是一个本地或 Internet 地址，你就需要在没有代理框的主机列表中键入 IP 地址。尽管一个 IP 地址可能标识一个本地地址，IP 地址有时候仍可能经过代理服务器进行处理，并且可能产生一个授权错误。

Visual InterDev 使用你所规定的 Web 项目代理服务器，只对 Web 项目中的文件进行操作。你可以另外规定一个代理服务器，供 Web 浏览器使用。如果你希望对 Internet Explorer 规定一个代理服务器，则可以在 Internet Explorer Internet Options 对话框中的 Connection 选项卡上实现。

第十章 管理 Web 项目

当你在你的 Web 应用文件上工作的时候，你可能希望改变你项目的结构，或者制作其它服务器的副本。

10.1 使主文件与本地文件同步

在你工作的时候，由于 Visual InterDev 6.0 保持应用的两个副本，你可以发现某些文件在本地，而不在主服务器上，另一些文件的情况正好相反。例如，如果你正同多个开发者以本地方式或离线方式进行工作，你可能需要确认你的本地文件列表与主应用一致。

你希望执行的任务是：

- **主副本同本地版本进行比较** 查看这两个文件的差别。
- **更新主应用 (master application)** 以反映你在本地应用中所做的变更。如果你在主模式中，这种更新会自动执行。
- **本地文件与主文件实现同步** 如果你正同多个开发者进行工作，你希望刷新你的本地版本(别人正使用着该版本的文件)，你可以得到当前本地应用中只读文件的最新版本。
- **刷新 Project Explorer** 以反映主应用和本地应用中的当前文件列表。

尽管你可能不需要包括主应用中所有文件的本地应用，你还是可能希望看看产生两种应用的整个文件设置。

不在你本地目录中的主服务器中的文件叫做主文件。没复制到主目录中的本地文件叫做个人文件。

比较主副本与本地副本

你可以看一看文件主副本与本地副本原代码的差别。

比较一个文件的主副本和本地副本

1. 在 Project Explorer 中，选择要比较的文件。

注意：如果你正在主方式中工作，在比较之前不保存文件。如果你保存，变更便自动保存到主服务器中，因而文件将是相同的。

2. 在 Project 菜单中，选择 Web Files, 然后选择 Compare to Master Web。出现 File Difference 对话框，其中的差别用亮灰色表示。如果文件没有差别，则出现一条信息，说明相同。

关于文件的更多内容，参见 File Difference（文件差别）对话框。如果你的项目使用原代码，参见第八章“同多个开发者一起工作”中的“解决归并冲突”一节。

更新主应用

一旦你已经把作出的变更保存到你的本地 Web 应用中，而且已做好准备把变更加到主 Web 应用中，你便可以发出本地副本。

更新主 Web 应用

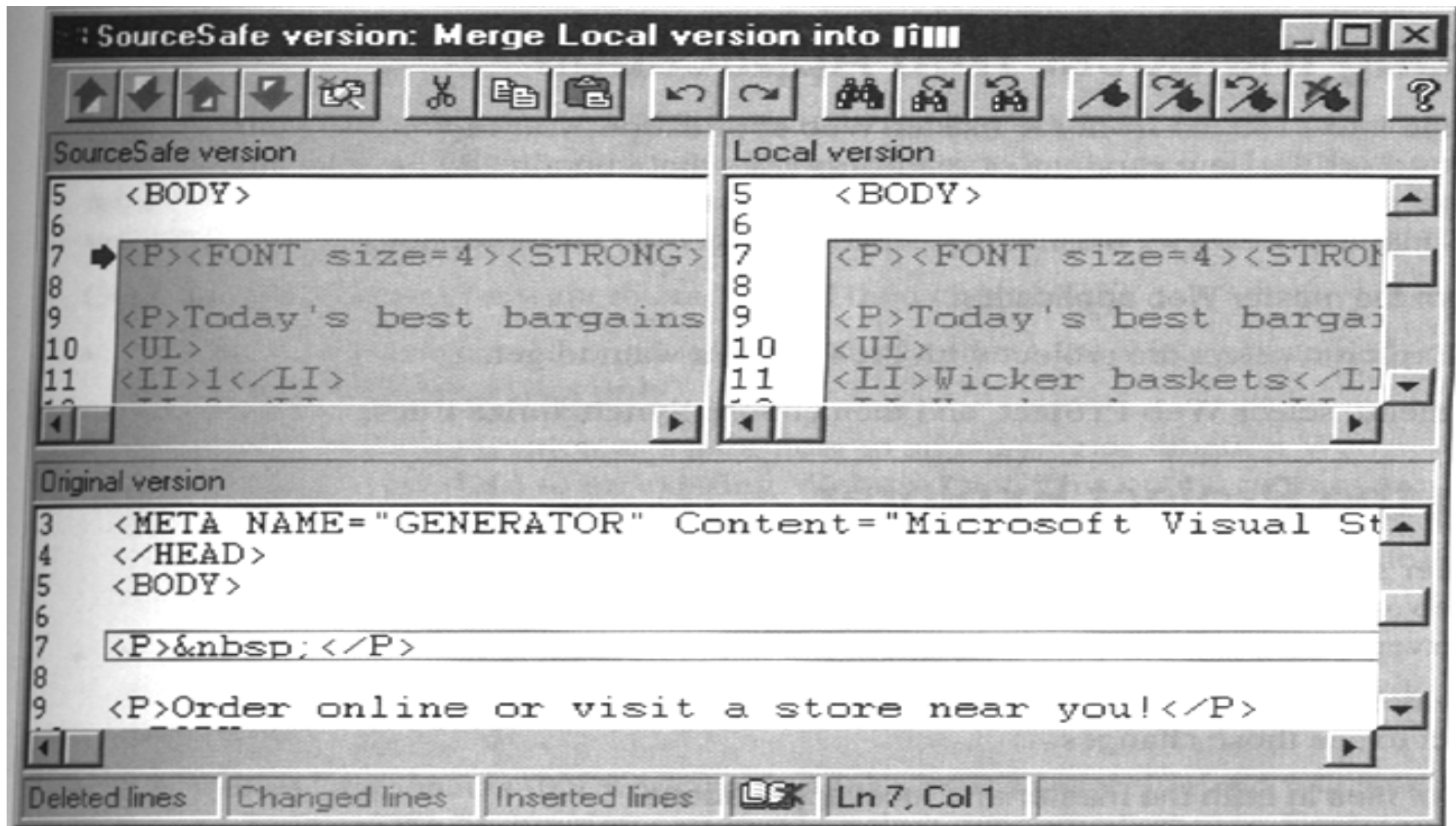
1. 在 Project Explorer 中，选择你希望保存到主 Web 应用中的文件。
2. 在 Project 菜单中选择 Web Files，然后选择 Release Working Copy。

注意：如果你的项目在源控件之下，在 Project 菜单中选择子菜单中的 Check In。

你的文件本地版本复制到主 Web 服务器中，并变成主 Web 文件的当前版本。

在你变更副本之后，如果有人又改变了主版本，就会在你发送本地副本的时候出现 Merge（归并）对话框。在 Merge 对话框中，你可以查看文件之间的差别，然后接受或拒绝改变。

例如，在下图中，SourceSafe 版本是某个人保存到服务器中的主版本。本地版本反映最新本地副本。原始版本窗格允许你编辑文件，并显示你作出的归并决定。



归并文件

1. 在底窗格中，定位并选择你希望解决的冲突区域。

2. 右键单击，在该菜单中选择你希望归并的变更。
3. 关闭对话框，选择 Save。

主副本和本地副本都反映你保存的变更。

你可能不希望你的变更扩散到主 Web 服务器中。例如，你可能觉得当前版本中的脚本比起你刚刚作出变更的文件更好一些。你可以丢弃你当前本地副本，留下未受触动的主版本。

丢弃对本地副本的变更

1. 在 Project Explorer 中，选择你不希望保存到主 Web 应用中的文件。
2. 在 Project 菜单中，选择 Web Files，然后选择 Discard Changes。

注意：如果你的项目在源控件之下，选择 Project 菜单中子菜单 Source Control 中的 Undo Checkout。

你选中的每个文件的本地副本都被该文件的主版本所取代。

同步本地 Web 应用

如果你想从主 Web 应用得到最新文件列表，则可以刷新你的项目。如果你想得到你已经在本地已有的只读文件的新版本，你可以使本地 Web 应用与主 Web 应用同步。要想用主副本取代你的允写文件，你可以获取最新版本。

从主 Web 应用中获得变更的文件

1. 在 Project Explorer 中，选择你希望获得文件的项目。
2. 从 Project 菜单中，选择 Web Project，然后再选择 Synchronize Files。

刷新 Project Explorer

Project Explorer 显示 Web 应用的文件列表，主应用或本地 Web 应用都在这个列表中。如果有人对主服务器的文件完成了添加、删除、移动或重命名，你对主应用也随之出现了相应变更。

还有，如果有人对主项目的源控件完成了开启或关闭，你就需要刷新这些项目以查看这些变更。

查看主应用和本地应用两处的当前文件列表

1. 在 Project Explorer 中，选择你希望刷新的项目。
2. 从 Project 菜单中，选择 Web Project，然后再选择 Refresh Project View。

10.2 复制 Web 项目

你可以把整个 Web 应用复制到另一个服务器中，或者在新 Web 应用名字下在同一个服务器中复制这个应用。要复制到另一个目录中或另一个服务器中，你将需要这个服务器上管理员级的特权，以便创建一个新目录。如果目标服务器没有 FrontPage 扩展版本，你就需要使用 Posting Acceptor 2.0。Posting Acceptor 2.0 在 Visual Studio 的企业版中是现成的。

复制一个 Web 应用

1. 在 Project Explorer 中，选择的项目是指向你希望复制 Web 应用的项目。
2. 在 Project 菜单中，选择 Web Project，然后选择 Copy Web Application。

3. 在 Copy Project 对话框中，选择你希望复制的应用副本。典型的情况是你选择主版本。

4. 在 Server name 框中，键入目标 Web 服务器的名字。

5. 在 Web project 框中，键入你希望的 URL 名字。

6. 清除 Copy changed files only 复选框。

注意：如果这个目标服务器有一个 Secured Socket Layer(SSL) 签证，你又希望在复制操作期间使用安全的 HTTP 传输，就选择 Connect using Secured Socket Layer 框。有关 Secured Socket Layer 更多内容，参见第六章“Web 项目概念”中的“安全 HTTP 传输”一节。

7. 检查你希望开启的选项，然后选择 OK。

- 要只复制最近一次复制 Web 以来所更新的文件或添加的文件，便选择 Copy changed files only。
- 要把 Web 应用作为一个子文件夹复制到一个已经有的目标 Web 应用中，就选择 Add to an existing Web project。这个选项把你希望复制的 Web 应用中的文件和文件夹归并到目标 Web 应用中。这个目标 Web 应用必须已经存在；如果没有，则出现错误信息。
- 如果 Web 应用是根应用，而且你希望把原服务器上的所有 Web 应用都复制到目标服务器上，就选择 Copy child Webs。
- 如果你有一些部件带有服务器注册标记，而且你希望在复制的时候在目标服务器上注册，就选择 Resister server components。参见第三十章“部署与维护 Web 应用”中的“部署综合 Web 解决方案”一节。

注意：如果你有一些部件，在其属性页面上标志着进行服务器注册，而且你

清除了这个选项，那么这些部件在操作完成之后将不在目标服务器上注册。

在 Web 服务器上建立一个新应用根，这些 Web 应用中的文件也复制到这个新文件夹中。你在 Copy Project 对话框中规定的名字变成应用 URL 的组成部分。

你现在可以完成应用的最后检验。整个 Web 应用，除了其安全设置之外，都在新名字下复制到目标服务器中。新复制的 Web 应用继承目标根项目的安全设置。关于安全设置的更多内容，参见第九章“增加安全措施”。

完成复制之后，原项目仍保持打开着，也没在当前解决方案中改变。要访问新复制的 Web 应用，你需要创建并打开一个项目，这个项目引用目标服务器上的副本。

10.3 重新组织项目结构

你可以对 Project Explorer 添加、删除、重命名或移动文件或文件夹。当你重命名或移动一些内容时，变更便立即加到主应用和本地应用中，不管你的项目是在主方式中还是在本地方式中。打开该项目的其他开发者在他们刷新 Project Explorer 之后可以看见施加于主应用的变更。

告诫：如果本地应用与主版本没同步，对项目的重组织操作就会失败。在重组织项目之前你可能希望刷新或同步你的项目。

重新组织一个项目也会影响文件中的链接。你要重命名、删除或移动的文件都叫做目标文件 (target files)。链接到目标文件上的文件叫做引用文件 (referring files)。

你可以选择是否更新引用文件来反映你的目标文件作出的变更。作为默认选择，提示你更新这些引用文件。关于设置链接维修选项的更多内容，参见第十六章“维护链接”中的“维修链接”一节。

删除文件或文件夹

1. 在 Project Explorer 中，选择你希望删除的文件或文件夹。
 2. 在 Edit 菜单中，选择 Delete。
- 你可能需要改变文件或文件夹的名字。

改变文件或文件夹的名字

1. 在 Project Explorer 中，选择你希望改变名字的文件或文件夹。
2. 在 File 菜单中，选择 Rename，然后键入新名字。

注意：如果你有文件的工作副本，就需要在允许你改变它的名字之前发出该副本。

你可能需要改变文件或文件夹在 Web 应用中的位置。

移动文件或文件夹

1. 确信文件的工作副本已经发出。
 2. 在 Project Explorer 中选择文件或文件夹，然后把它们拖到 Project Explorer 中的目的地。
- 你可能希望复制你的 Web 应用中的文件或文件夹。

复制文件或文件夹

1. 在 Project Explorer 中选择你希望复制的文件或文件夹。
2. 按 CTRL 按键，然后把它们拖到 Project Explorer 中的目的地。

