
Dashcode 使用手册

[Tools](#) > [Scripting & Automation](#)



2008-07-25



Apple Inc.
© 2008 Apple Inc.
保留一切权利。

事先未经 Apple Inc. 书面许可，此出版物的任何部分均不得以任何形式或通过任何方式（包括机械、电子、影印、记录或其他方式）复制、储存在检索系统中或传播，但以下情况例外：任何人在此被授权将文稿存储在单台电脑上以仅供个人使用并打印文稿的副本以用于个人用途，只要文稿包括 Apple 的版权声明。

Apple 标志是 Apple Inc. 的商标。

事先未经 Apple 书面同意，将“键盘”Apple 标志 (Option-Shift-K) 用于商业用途可能会违反美国联邦和州法律，并可能被指控侵犯商标权和进行不公平竞争。

本文稿不对所描述的任何技术授予明示或暗示的许可。Apple 保留与本文稿中所描述的技术相关的所有知识产权。本文稿只可用来帮助应用程序开发者仅为贴有 Apple 标签的电脑开发应用程序。

我们已尽力确保本文稿中的信息准确。Apple 对印刷错误概不负责。

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, Dashcode, iCal, iPhoto, iPod, Mac, Mac OS, Monaco, Objective-C, Quartz, QuickTime, Safari, and Xcode are trademarks of Apple Inc., registered in the United States and other countries.

Finder and iPhone are trademarks of Apple Inc.

Adobe, Acrobat, and PostScript are trademarks or registered trademarks of Adobe Systems Incorporated in the U.S. and/or other countries.

Helvetica and Times are registered trademarks of Heidelberger Druckmaschinen AG, available from Linotype Library GmbH.

Java and all Java-based trademarks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

同时在美国和加拿大出版。

虽然 Apple 已检查了本文稿，但 Apple 对本文稿及其质量、准确度、适销性、特定用途的适用性不作任何明示或暗示的保证或表示。因此，本文稿按“原样”提供，而读者要承担有关其质量和准确度的整个风险。

在任何情况下，Apple 将不对因本文稿中的任何瑕疵或错误而造成的直接、间接、特殊、偶然或必然

损失承担责任，即使 Apple 已告知这类损失的可能性。

上述保证和补救措施是排他性的，替代所有其他口头或书面以及明示或暗示的保证和补救措施。未经授权，任何 Apple 经销商、代理商或雇员都不得对此保证进行任何修改、扩充或添加。

某些州不允许排除或限制对某些偶然或必然损失的暗示保证或责任，因此上述限制或排除可能不适用于您。此保证给予您特定的法定权利，因州而异，您可能还拥有其他权利。

目录

介绍

Dashcode 使用手册的简介 9

- 哪些人应当阅读本文稿? 9
- 本文稿的结构 9
- 获得和运行 Dashcode 10
- 报告错误 10
- 另请参阅 10

第 1 章

Widget 教程 13

- 开始之前 13
- 选取一个模板 13
- 设定目标日期 14
- 测试“倒计时器” 15
- 自定 Widget 的外观 15
- 使用部件来添加功能性 16
- 编写代码以显示 Apple Store 17
- 部署 Widget 17

第 2 章

Web 应用程序教程 19

- 开始之前 19
- 选取一个模板 19
- 了解有关模板的信息 20
- 测试应用程序 21
- 探索堆栈布局中的视图 22
- 使用部件来添加功能性 23
- 编写代码以执行搜索 24
- 部署 Web 应用程序 25

第 3 章

开始一个项目 27

- 使用模板来创建项目 27
- 使用现有 Widget 来创建项目 27
- 提供 Widget 属性 28
- 为 Web 应用程序提供属性 28
- 打开现有的 Widget 29

第 4 章

设计 Widget 或 Web 应用程序 31

- 布置界面 31
- 显示 Widget 的两面 31

- 将部件添加到界面 31
- 使用来自 iPhoto 的照片 32
- 更改元素的属性 32
- 排列和锁定元素 33
- 绝对定位与文稿流定位 33
- 搜索元素 33
- 标尺和参考线 34
- 放置关闭方框 34
- 停用画布 34
- 预览 Widget 的默认图像 34
- 设计 Widget 图标 35
- 为 Web 应用程序设计 Web Clip 图标 35

第 5 章 添加源代码 37

- 查看项目的源代码 37
- 使用 HTML、CSS 和 JavaScript 编程界面 38
- 为事件添加处理程序 38
- 为自定控制器添加代码 38
- 使用“代码预见”来完成代码 39
- 代码片段 39
- 添加影片文件和文件夹 39
- 资源访问 40
- 代码编辑偏好设置 40

第 6 章 测试和共享 43

- 测试 Widget 或 Web 应用程序 43
- 运行日志和跟踪执行 44
- 暂停和逐步执行 44
- 检查内存中的值 45
- 断点 45
- 代码评估者 45
- 共享 Widget 46
- 部署 Web 应用程序 46

第 7 章 有关 Widget 的高级主题 47

- 本地化 47
- Widget 插件 47

附录 A Dashcode 模板 49

- Widget 模板 49
 - “自定”Widget 模板 49
 - “倒计时器”模板 49

“地图”模板	49
RSS Widget 模板	50
Podcast 模板	50
Photocast 模板	50
Quartz Composer 模板	51
“视频 Podcast”模板	51
“量表”模板	51
“每日提要”模板	52
Web 应用程序模板	52
“自定”Web 应用程序模板	52
“浏览器”模板	52
Podcast Web 应用程序模板	53
RSS Web 应用程序模板	53
“实用工具”Web 应用程序模板	53

附录 B Dashcode 部件 55

活动指示器	55
返回按钮	55
浏览器	55
画布	56
呼叫按钮	56
栏布局	56
边缘到边缘列表	56
前进按钮	57
量表	57
指示器	57
层次指示器	58
邮寄按钮	58
地图按钮	58
Quartz Composer	58
QuickTime	59
圆角矩形列表	59
滚动区域	59
堆栈布局	60
堆栈布局方法	60
堆栈布局过渡	60

修订历史 文稿修订历史 63

图形

第 1 章

Widget 教程 13

- 图 1-1 生日 Widget 13
- 图 1-2 显示一个新 Widget 的项目窗口 14
- 图 1-3 “倒计时器”模板的属性 15
- 图 1-4 使用“填充和线型”检查器来调整 front 图像 16
- 图 1-5 源代码编辑器中的一个函数 17

第 2 章

Web 应用程序教程 19

- 图 2-1 基于“浏览器”模板的新 Web 应用程序项目 20
- 图 2-2 基于浏览器的默认 Web 应用程序包含一个顶级页面（左）和一个详细页面（右） 22
- 图 2-3 给 Web 应用程序的用户界面添加部件 24
- 图 2-4 添加代码以处理按钮的点按事件 25

Dashcode 使用手册的简介

本文档提供了对 Dashcode 开发环境的概述。它描述了如何使用 Dashcode 来创建两种类型的项目：

- **Dashboard Widget** 十分简单，是在 Dashboard 环境中执行单个任务的轻量级应用程序。Widget 实际上是由标准 Web 技术〔例如超文本标识语言 (HTML)、层叠样式表 (CSS) 和 JavaScript〕提供支持的打包网页。
- **Web 应用程序**是为用于 iPhone 上的 Safari 而进行了优化的网页，能够给用户提供分离的功能性。Web 应用程序也使用 HTML、CSS 和 JavaScript 等 Web 技术。

Dashcode 的集成环境允许您布置、编码甚至测试 Widget 和 Web 应用程序，而无须打开任何其他应用程序。它的布局工具、设计器和编辑器简化了创建这些项目所需的所有资源的流程。Dashcode 还包括一些便于使用的编码和调试工具，这些工具能够帮助您管理和测试您编写的任何代码。

哪些人应当阅读本文档？

请阅读《Dashcode 使用手册》以了解如何使用 Dashcode 来创建 Web 应用程序和 Dashboard Widget。对 Widget 或 Web 应用程序创作并不熟悉的开发人员可以了解如何构建简单的项目，以及了解有关 Dashcode 功能的更多信息。有经验的开发人员可以了解如何加快使用 Dashcode 进行开发的速度。

本文档的结构

本文档包含以下章节：

- **“Widget 教程”**（第 13 页）会引导您使用 Dashcode 来创建您的第一个 Dashboard Widget。
- **“Web 应用程序教程”**（第 19 页）向您示范了如何使用 Dashcode 来创建简单的 Web 应用程序。
- **“开始一个项目”**（第 27 页）对使用 Dashcode 时的不同起点进行了论述。
- **“设计 Widget 或 Web 应用程序”**（第 31 页）向您展示了 Dashcode 提供用于设计 Widget 或 Web 应用程序的用户界面的工具。
- **“添加源代码”**（第 37 页）详细描述了 Dashcode 附带的源代码编辑工具。
- **“测试和共享”**（第 43 页）包括有关测试、调试和分发 Widget 或 Web 应用程序的信息。
- **“有关 Widget 的高级主题”**（第 47 页）谈论有关使用 Dashcode 将 Widget 本地化和包括 Widget 插件的信息。

《Dashcode 使用手册》还包括以下附录：

- **“Dashcode 模板”**（第 49 页）描述了 Dashcode 附带的项目模板。

- “Dashcode 部件”（第 55 页）包括有关通过 JavaScript 来控制源于 Dashcode 的自定元素（称为部件）的信息。

获得和运行 Dashcode

Apple 提供了用于创建 Mac OS X 软件的全套开发人员工具（包括 Dashcode）。Xcode Tools 包括了一些能够帮助您设计、创建、调试和优化软件的应用程序。此工具套件还包括用于 Apple 技术的标头文件、样本代码和文稿。您可以从 Apple 开发人员联盟 (ADC) 网站 (<http://developer.apple.com.cn>) 下载 Xcode Tools。要求注册，但是免费的。

下载了 Xcode Tools 之后，请使用 Xcode Tools 安装器来安装 Dashcode。安装之后，您将在 /Developer/Applications/ 中找到 Dashcode。

报告错误

如果您在 Apple 软件或文稿中遇到错误，我们鼓励您将它们报告到 Apple。您也可以提交功能增强请求，以描述您想在产品或文稿的将来版本中看到的功能。要提交错误或功能增强请求，请前往 ADC 网站的“错误报告”页面，该页面的地址为以下 URL：

<http://developer.apple.com/bugreporter/>

您必须有有效的 ADC 登录名称和密码才能提交错误。您可以按照“错误报告”页面上的说明，免费获取登录名称。要提交 Dashcode 的错误，请使用 Dashcode 组件（版本 X）。

另请参阅

有关如何创建能够在 iPhone 和 iPod touch 上很好地工作的 Web 应用程序的深入信息，请参阅《Safari Web Content Guide for iPhone》（用于 iPhone 的 Safari Web 内容指南）。有关如何设计此类应用程序的用户界面的指南，请参阅《iPhone Human Interface Guidelines》（iPhone 人性化界面指南）。

有关如何在不使用 Dashcode 的情况下创建 Dashboard Widget 的说明，请参阅《Dashboard Tutorial》（Dashboard 教程）。

有关使用 Dashcode 来创建 Widget 时可用的技术的信息，请阅读《Dashboard Programming Topics》（Dashboard 编程主题）。本文稿中所讨论的所有特定于 Dashboard 的信息，在《Dashboard Reference》（Dashboard 参考）中进行了更深入的描述。附加的 Dashboard 文稿和样本代码可以在 [Dashcode 的 ADC 主题页面](#) 上找到。

Safari 的 [ADC 主题页面](#) 包含有关 WebKit 的有用信息，WebKit 技术为 Dashboard Widget 提供了支持。有关 WebKit 中的 HTML、CSS 和 JavaScript 功能的更多信息，请查阅：

- 《Safari HTML Reference》（Safari HTML 参考）
- 《Safari CSS Reference》（Safari CSS 参考）
- 《Safari JavaScript Reference》（Safari JavaScript 参考）

XMLHttpRequest 对象允许您在 JavaScript 中解析 XML，并使用解析结果。有关更多信息，请阅读 ADC 文章[“Dynamic HTML and XML: The XMLHttpRequest Object”](#)。

Widget 教程

此教程会引导您使用 Dashcode 创建 Dashboard Widget。遵循这些步骤操作时，您将逐步学会如何选择 Widget 模板、自定 Widget 的外观和代码以及与其他人共享您的 Widget。完成此教程是开始在 Dashcode 中构建 Widget 的一种快速而简便的方法。

此文稿包括另一个教程（“[Web 应用程序教程](#)”（第 19 页）），它跟在此教程后。文稿的剩余部分会更深入地研究 Dashcode 开发环境，描述它如何支持 Widget 和 Web 应用程序开发。如果不想学习如何创建 Web 应用程序，您可以跳过“[Web 应用程序教程](#)”（第 19 页），并通过阅读“[使用模板来创建项目](#)”（第 27 页）来继续学习有关 Dashcode 的更多知识。

开始之前

在本教程中，您可以构建一个 Widget 来为自己的生日进行倒计时，它与图 1-1 中所示的 Widget 相似。

图 1-1 生日 Widget



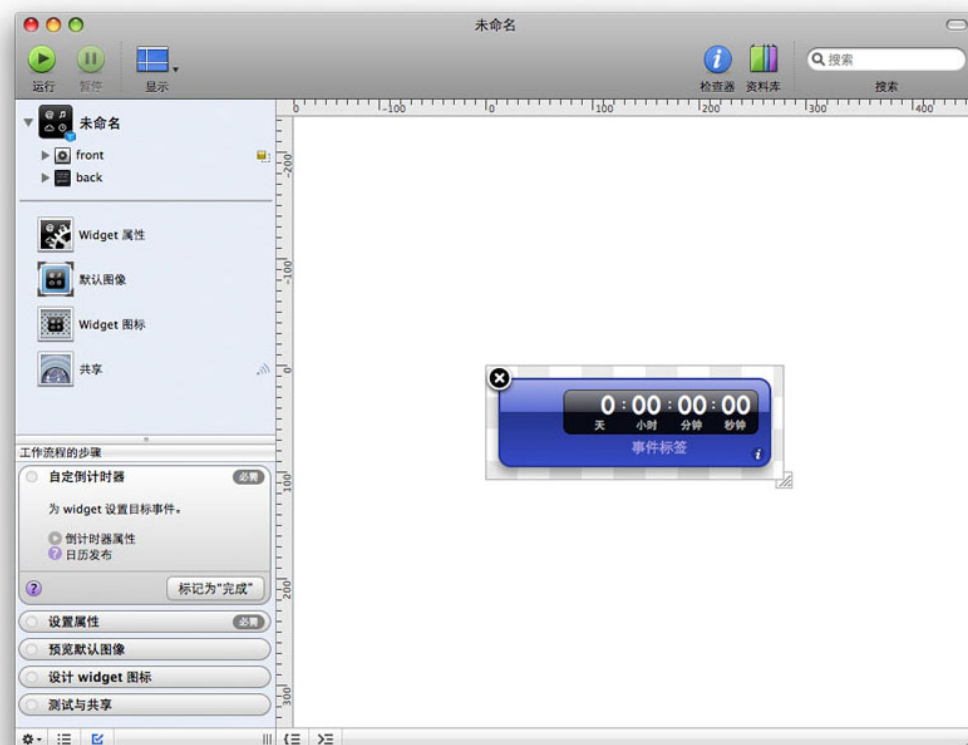
请确定您已经在 Mac 上安装了 Dashcode（位置是 /Developer/Applications/），然后再继续。如果您没有安装 Dashcode，请阅读“[获得和运行 Dashcode](#)”（第 10 页）以了解如何获得和安装 Dashcode。

选取一个模板

要开始，请连按 Dashcode 图标以打开它。新项目窗口会打开并显示一个对话框，在此对话框中，您先选择感兴趣的项目类型（在此例子中为 Dashboard Widget），然后从模板分类中选择想要创建的 Widget 类型。模板是便于使用的起点，用于创建常见类型的 Widget。选择一个模板的图标，以显示对该模板的用途的简短描述。

要制作“生日 Widget”，本教程使用“倒计时器”模板。请选择它的图标并按“选取”。一个项目窗口会随基于“倒计时器”模板的新 Widget 一起打开，如图 1-2 中所示。

图 1-2 显示一个新 Widget 的项目窗口



项目窗口的左侧上是导航器，用于在设计 Web 应用程序时的各种可用工具之间切换。窗口的主要部分是画布，用于设计 Widget 的界面。

在图 1-2 中，您可以在导航器的底部看到“工作流程的步骤”列表，它将引导您完成 Widget 开发过程。每个步骤都是创建一个 Widget 中的里程碑，告诉您执行哪些操作以及在何处执行。完成一个步骤之后，请将它标记为“完成”，然后继续前进到下一个步骤。

注：如果您不想看到“工作流程的步骤”列表，您可以通过选取“显示”>“步骤”或通过点按项目窗口底部边缘看起来类似于注册格的按钮（此按钮在图 1-2 中被高亮显示），将该列表隐藏。

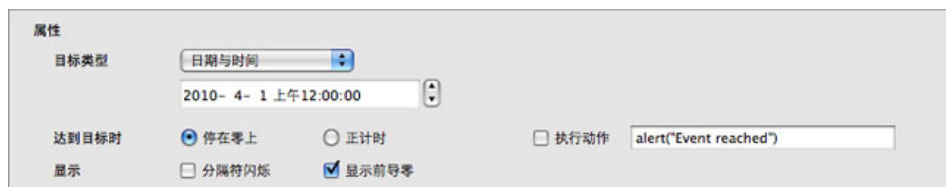
或者，您也可以显示 widget 文件的列表以代替“工作流程的步骤”列表。如果您想要查看“文件”列表，请选取“显示”>“文件”，或点按项目窗口底部边缘的列表按钮（该按钮在注册格按钮的左边）。

设定目标日期

“倒计时器”模板能够为您提供一个 Widget，这个 Widget 带有为某个事件进行倒计时所需的所有元素和代码。您只需将目标日期告诉 Widget 即可。要设定目标日期，请在导航器中选择“Widget 属性”。画布会被“Widget 属性”面板取代，在该面板中，您可以指定您的 Widget 需要的重要值。

在“Widget 属性”面板的“属性”部分，请在“目标类型”弹出式菜单中选取“日期与时间”，然后输入您下一次生日的日期，如图 1-3 中所示。

图 1-3 “倒计时器”模板的属性



测试“倒计时器”

您的新 Widget 已经具有全部功能。要证明这一点，请选取“调试”>“运行”，以运行该 Widget。Dashcode 可以运行一个 Widget 而无须在 Dashboard 中打开它，从而使 Dashcode 成为一个方便位置，您可以在那里测试您的 Widget 以及修正您遇到的任何问题。载入 Widget 之后，它会朝着您的下一次生日开始倒计时。

如果您对 Widget 能够按您预期的工作而感到满意，请选取“调试”>“停止”，以停止它。

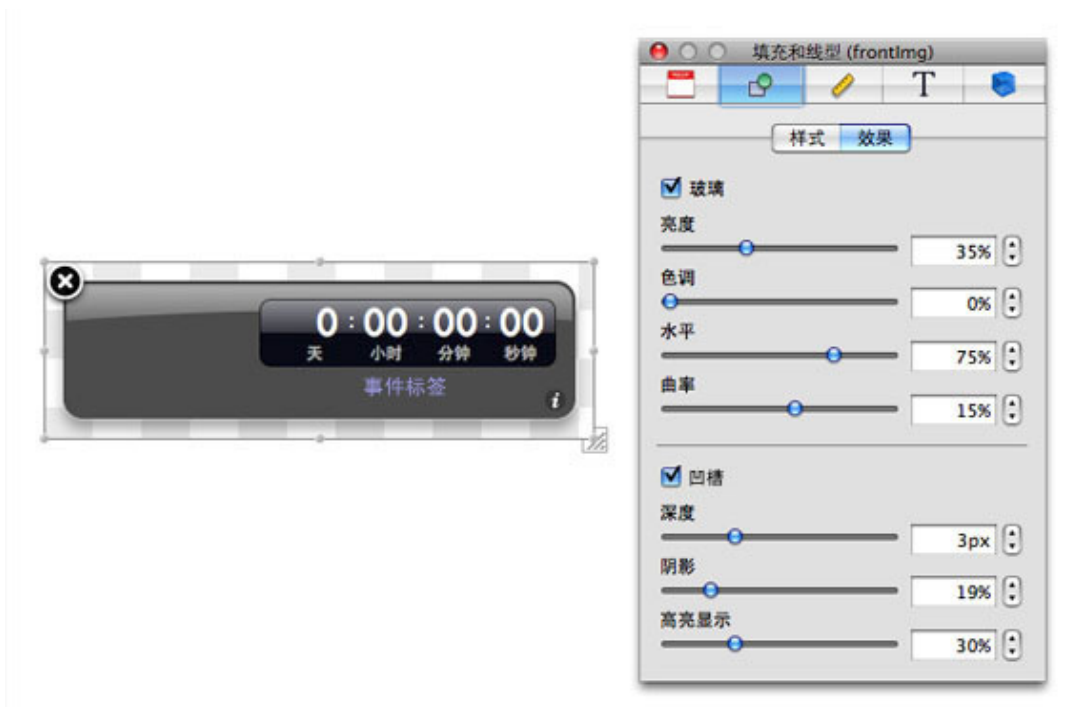
现在是存储 Widget 项目的很好时机。选取“文件”>“存储”，以存储该项目。给您的项目命名，然后选择一个位置来存储它。您的 Widget 将存储在一个 Widget 项目中，该项目装入了 Dashcode 为您创建 Widget 所需的 Widget 和信息。

自定 Widget 的外观

虽然现在您有一个功能完整并且已准备好用于共享的 Widget，但您不妨将它个性化，以使它显得比较独特。Dashcode 的设计工具使自定 Widget 的界面变得容易。在导航器中选择 Widget 对象（它应当显示您在存储项目时给它命名的名称）。“Widget 属性”面板会被替换成画布。

首先，请更改您的 Widget 的主体颜色。在画布上选择 Widget 主体（也称为 front 图像或 frontImg），然后选取“窗口”>“显示检查器”。“检查器”窗口可让您修改所需元素的属性，例如它的外观和行为。点按“检查器”窗口顶部的“填充和线型”按钮（左起第二个）。如果它不是已被选定，请点按“样式”标签以显示填充、线型、边角圆度和不透明度值。在出现的“颜色”窗口中，点按颜色池并选取新颜色。请尝试不同的填充样式，直至找到您喜欢的组合。如果您想要尝试更改其他效果，比如玻璃外观，请点按“效果”标签以显示这些值，如图 1-4 中所示。

图 1-4 使用“填充和线型”检查器来调整 front 图像



完成自定 Widget 的主体之后，您可以从 iPhoto 将自己的照片添加到 Widget。您可以从“资料库”窗口访问 iPhoto 图库。要显示 iPhoto 图库，请选取“窗口”>“显示资料库”；然后点按“照片”按钮。找到一张照片，然后将它拖移到画布上的 Widget。调整它的大小，方法是拖移照片上的任意一个调整大小手柄。

最后，将“事件标签”文本更改为类似于“距离我的生日还有...天”之类的文字。执行此步骤的方法是：选择“事件标签”文本，点按“检查器”窗口中的“属性”按钮（最左边的按钮），然后在“值”栏中输入文本；或者连接 Widget 主体本身中的文本，然后输入新文本。

使用部件来添加功能性

既然您有了为自己的生日进行倒计时的个性化 Widget，您就可以添加一个点按时会显示 Apple Store（从而您的朋友和家人可以购买生日礼物给您！）的按钮。要给您的 Widget 添加按钮，请使用“按钮”部件。部件是在 Widget 界面上使用的控制和显示。

要查找按钮部件，请选取“窗口”>“显示资料库”，然后点按“部件”。您可以使用窗口底部的搜索栏来辅助查找特定部件或者部件类型。从部件列表中，将“菱形按钮”部件从“资料库”窗口拖移到 Widget 的主体上。连按该按钮以选择它的标签文本，然后输入文本“给我购买礼物”并按下 Return。您将可能需要调整按钮的大小以适合新标签。

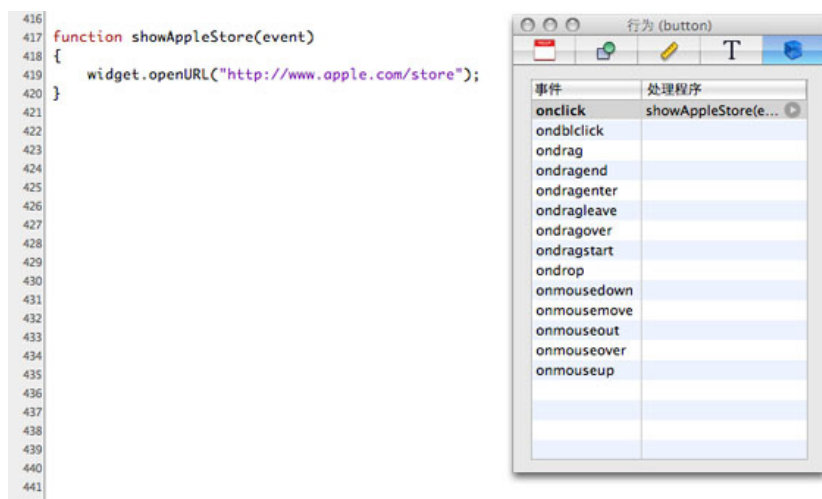
编写代码以显示 Apple Store

要使用户点按该按钮时登录到 **Apple Store**，您需要给该按钮添加一种行为。在“检查器”窗口中，点按“行为”按钮（最右边的按钮）。这样会显示“行为”面板，在该面板中，您可以给对象上的事件分配处理程序函数。在画布上选择该按钮，然后连接 onclick 事件名称旁边的“处理程序”栏。输入新函数的名称，例如 showAppleStore，然后按下 **Return**。按您输入的函数名称旁边的箭头以在画布下方显示源代码编辑器。您可以在此处编写代码，以给您的 **Widget** 添加功能性。点按箭头会显示 **Dashcode** 插入到 **Widget** 的代码中的 showAppleStore 函数。在大括号 ({ .. }) 之间，请输入以下代码行：

```
widget.openURL("http://store.apple.com/");
```

源代码编辑器中的代码应该看起来像图 1-5 中的那样。

图 1-5 源代码编辑器中的一个函数



通过选取“调试”>“运行”，再次测试您的 **Widget**。点按您添加的按钮，并确定随显示的 **Apple Store** 网站打开一个新的 **Safari** 窗口。请确定您经常存储项目以保留所做的更改。

部署 Widget

恭喜您！您已经使用 **Dashcode** 创建了第一个完整的 **Widget**。

要在 **Dashboard** 中打开您的 **Widget**，请选取“文件”>“部署 **Widget**”。点按对话框中出现的“安装”以在 **Dashboard** 中查看 **Widget**。

要与周围的人共享您的 **Widget**，请在导航器中选择“共享”。出现的共享面板会显示您在“测试”[倒计时器](#)（第 15 页）中选取的 **Widget** 项目名称，但是如果您想要的话，可以用不同的名称替换此项目名称。您还可以设定运行 **Widget** 的最低 **Mac OS X** 版本。点按“存储到磁盘”以存储您的 **Widget**。现在您可以将它以电子邮件的形式发送给朋友，或者张贴在网页上。您可以使用 **Finder** 中的“文件”>“压缩”命令来归档它。

Web 应用程序教程

此教程将引导您完成使用 **Dashcode** 来创建 **Web** 应用程序。遵循这些步骤的同时，您就能够了解如何选取模板、自定应用程序的代码以及与其他人共享您的应用程序。完成此教程是开始在 **Dashcode** 中构建 **Web** 应用程序的一种快速而简便的方法。

这是本文稿中两个教程的第二个。如果您有兴趣了解如何开始开发 **Widget**，请务必阅读[“Widget 教程”](#)（第 13 页）。文稿从[“开始一个项目”](#)（第 27 页）起的剩余部分更深入地研究 **Dashcode** 开发环境，描述它如何支持 **Widget** 和 **Web** 应用程序开发。

开始之前

在此教程中，您将构建一个 **Web** 应用程序来显示国家公园的有关信息。此 **Web** 应用程序基于 **Web** 应用程序的浏览器类型，并且支持浏览多层次的内容。

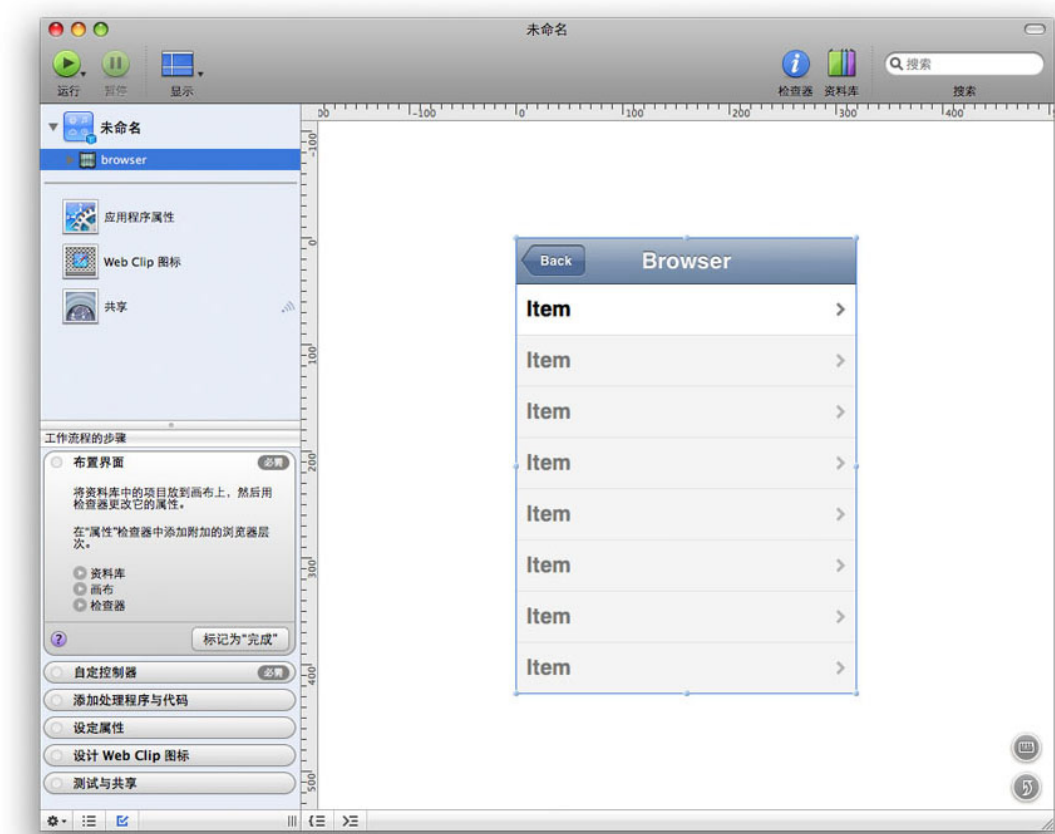
请先确定您已经在 **Mac** 上安装了 **Dashcode 2.0** 或更高版本（安装位置是 `/Developer/Applications/`），然后再继续。如果没有安装 **Dashcode**，请阅读[“获得和运行 Dashcode”](#)（第 10 页）以了解如何获得并安装 **Dashcode**。

选取一个模板

要开始，请连接 **Dashcode** 图标以打开它。这时会打开一个新的项目窗口并显示一个对话框，该对话框会显示两种类型的项目（**Dashboard Widget** 和 **Web 应用程序**）以及用于每种项目类型的各类模板。模板是便于使用的起点，用于创建常见类型的 **Web** 应用程序和 **Widget**。（要了解模板的用途，请选择它以显示对该模板的功能的简短描述。）

对于此教程，请选择“**Web 应用程序**”项目类型。然后，选择“**浏览器**”模板并点按“**选取**”。这时会打开一个项目窗口，显示基于“**浏览器**”模板的新 **Web** 应用程序的第一页，如图 2-1 中所示。

图 2-1 基于“浏览器”模板的新 Web 应用程序项目



项目窗口的左侧上是导航器，用于在设计 Web 应用程序时的各种可用工具之间切换。窗口的主要部分是画布，用于设计 Web 应用程序的界面。

在图 2-1 中的导航器底部，您可以看到“工作流程的步骤”列表，它会指导您完成 Web 应用程序的开发过程。每个步骤都是创建一个 Web 应用程序中的里程碑，告诉您执行哪些操作以及在何处执行。完成一个步骤之后，请将它标记为“完成”，然后继续前进到下一个步骤。

注：如果您不想看到“工作流程的步骤”列表，您可以通过选取“显示”>“步骤”或通过点按项目窗口底部边缘看起来类似于记事格的按钮（此按钮在图 2-1 中被高亮显示），将该列表隐藏。

或者，您也可以显示 widget 文件的列表以代替“工作流程的步骤”列表。如果您想要查看“文件”列表，请选取“显示”>“文件”，或点按项目查看底部边缘的列表按钮（该按钮位于记事格按钮的左侧）。

了解有关模板的信息

“浏览器”模板可以为您提供一个 Web 应用程序，该 Web 应用程序会显示一些内建信息并支持从一个页面浏览到下一个页面。您根本不需要自定应用程序便可以查看它的工作情况，但您应当指定要显示的标题。

在画布中，连接“浏览器”并键入一个新的标题，例如“Parks”。您将在第一页的标头中看到此标题。

您还应当给网页本身命名。当您开始一个项目时，此名称是未命名的。要指定一个名称，请连接导航器顶部的“未命名”并键入您想要的名称，比如说“National Parks”。

现在也是存储您的项目的最佳时机。选取“文件”>“存储”，以存储该项目。给您的项目命名，然后选择一个位置来存储它。项目装入了 Dashcode 为您创建应用程序所需的 Web 应用程序和信息。

测试应用程序

虽然您的新 Widget 只显示占位符数据，但它已经具有一定功能了。要证明这一点，请点按 Dashcode 工具栏中的绿色“运行”按钮，以运行该 Web 应用程序。（或者，您也可以选取“调试”>“运行”以测试您的应用程序。）

Dashcode 会在模拟程序中打开您的 Web 应用程序。虽然 Web 应用程序的外观与在 iPhone 或 iPod touch 上不完全一致，但它会运行，从而您可以测试它并修复遇到的任何问题。

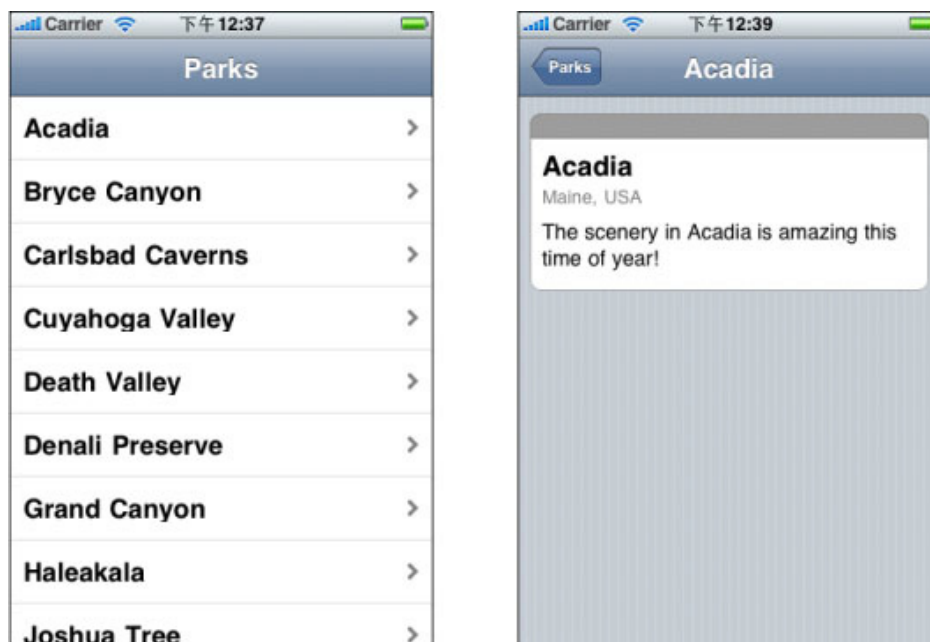
请花点时间使用该应用程序。您应当能够看到“Parks”（或您键入的其他标题）显示在第一页的标头中，并且应当在标题栏区域（如果该区域可见）中看到“National Parks”（或您给网页命名的其他标题）。

请注意当您点按第一页中的一个行项目时，会出现一个新的页面，提供有关该项目的详细信息。您还应当注意有关第二页的若干其他信息：

- 第二页的标题与您在第一页中所选的行项目的名称相同。
- 返回按钮出现在标题的左侧，它的标签是“Parks”（第一页的标题）。您可以点按此按钮以返回到第一页。
- 第二页中间的文本包括您所选的行项目的名称。

Web 应用程序的两个页面应当与图 2-2 中所示的相似。

图 2-2 基于浏览器的默认 Web 应用程序包含一个顶级页面（左）和一个详细页面（右）



虽然有关国家公园的信息只是由“浏览器”模板提供的占位符信息，但它能够帮助您查看页面之间的一些连接。您将在下一个步骤（“探索堆栈布局中的视图”（第 22 页））中了解有关这些连接的更多信息。

请先通过从模拟应用程序菜单中选取“退出”或按下 **Command-Q** 来退出模拟进程，然后再继续。或者，您可以点按 **Dashcode** 中的红色“停止”按钮。

探索堆栈布局中的视图

一个浏览器样式的 Web 应用程序允许用户从一个页面浏览到下一个页面。测试您的 Web 应用程序时，您会发现它已经支持以下功能：选择顶级列表中的项目以及在详细视图中显示有关它的信息。此结构的其中一个关键要素是堆栈布局。

如果您还不能在导航器中看到它，请通过点按“National Parks”以及浏览器（位于“National Parks”的下方）旁边的显示三角形来显示堆栈布局。当您点按 `stackLayout` 旁边的显示三角形时，您可以看到两个项目：`listLevel` 和 `detailLevel`。在此应用程序中，列表级视图包含公园的列表；详细级视图包含描述所选公园的文本。

您可以更改在顶层列表中选择一行时详细信息页面的显示方式。要执行此操作，请遵循以下步骤：

1. 请确定检查器已打开。如果未打开，请点按 **Dashcode** 工具栏中的蓝色“简介”按钮。
2. 在导航器中选择 `stackLayout`。
3. 点按检查器（位于左侧的那个）中的“属性”工具栏项目。这会将检查器的标题更改为“属性 (`stackLayout`)”。

检查器的底部是标题为“子视图过渡”的部分。此部分中的值控制页面之间的过渡。您可以选取让页面从右到左滑入（此为默认的方式），也可以让页面按其他方式滑入，例如从上到下。请注意您可能无法在模拟程序中查看更改过渡类型的效果，但当您在实际设备上运行应用程序时，您将能够看到效果。

注：此步骤中所描述的过渡基于 CSS 过渡和动画。在堆栈布局“属性”检查器中，Dashcode 为您提供了一种方法，用于指定您想要使用的过渡的属性，但不提供用于直接控制它们的用户界面。完成此教程之后，您不妨访问这些过渡的代码，以进行细致的调整。在执行此操作之前，您应当先阅读《*Safari CSS Animation Guide for iPhone OS*》（iPhone OS 的 Safari CSS 动画指南）和《*Safari CSS Transform Guide for iPhone OS*》（iPhone OS 的 Safari CSS 变换指南）以获得更多信息。

使用部件来添加功能性

正如您所看到的，虽然您使用“浏览器”模板所创建的 Web 应用程序的所有数据都是静态的，但它已经具有一定功能了。要使它变得更加有用，您可以添加从 Web 获取当前信息的功能性。在本步骤中，您会使用 Dashcode 的设计工具给详细信息页面添加一个按钮，该按钮用于执行有关重要公园的 Google 搜索（您将在下一个步骤中添加代码以支持此操作）。

要给您的 Web 应用程序添加按钮，请使用“按钮”部件。部件是在 Web 应用程序的用户界面中使用的控制和显示。Dashcode 在“资料库”中列出了可用的部件。

要查找合适的按钮部件，请选取“窗口”>“显示资料库”，然后点按“部件”。在部件列表中滚动，直至您找到按钮部件，然后将此部件拖移到画布上的详细页面中。

连按详细页面上的按钮并给它添加标签（您可以通过在该按钮的“属性”检查器的“标签”栏中键入标签来执行此操作）。您不妨调整按钮的大小以适合您选取的标签。在接收到新的标签之前，图 2-3 会在详细视图中显示新按钮。

图 2-3 给 Web 应用程序的用户界面添加部件



编写代码以执行搜索

要使按钮执行有关重要公园的搜索，您需要给该按钮添加行为。在“检查器”中，点按“行为”（最右边的按钮）。这样会显示“行为”面板，在该面板中，您可以给对象上的事件分配处理程序函数。

在画布上，选择您添加到详细页面的按钮。在“行为”检查器中，连接 onclick 事件名称旁边的“处理程序”列。输入新函数的名称，例如 detailButtonHandler，然后按下 **Return**。点按您输入的函数名称旁边的箭头，以在画布下面显示源代码编辑器。点按箭头会显示 Dashcode 插入到 Web 应用程序的代码中的 detailButtonHandler 函数。

在大括号 ({...}) 之间，请输入以下代码行：

```
document.location = ("http://www.google.com/search?client=googlet&q=" +  
detailController._representedObject);
```

此行代码指示 Web 应用程序显示重要公园的搜索结果（detailController._representedObject 包含详细信息页面上的公园名称）。添加此功能之后，源代码编辑器中的代码应当类似于图 2-4。

图 2-4 添加代码以处理按钮的点按事件

```
101  
102 function detailButtonHandler(event)  
103 {  
104     document.location = ("http://www.google.com/search?client=googlet&q=" +  
105     detailController_representedObject);  
106 }  
107  
108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122
```



通过点按“运行”（或通过选取“调试”>“运行”），再次测试您的 Web 应用程序。在公园的详细信息页面中，点按您添加的按钮并确定它会显示有关重要公园的 Google 搜索结果（尝试执行此操作之前，请先确定您的电脑已连接到 Internet）。

部署 Web 应用程序

恭喜您！您已经使用 Dashcode 创建了第一个 Web 应用程序。

要在 iPhone 或 iPod touch 上使用您的应用程序，您需要部署该应用程序并使它在 Web 服务器上可用。要了解如何执行此操作，请参阅[部署 Web 应用程序](#)（第 46 页）。

开始一个项目

当您开始新项目时，您将它基于 Dashcode 模板，或者对于 Widget 的情况您还可以将它基于现有的 Widget。Widget 项目装入了 Dashcode 用来为您构建一个 Widget 的所有资源。类似地，Web 应用程序项目包含 Dashcode 建立 Web 应用程序所需的所有资源。

本章讨论了您开始项目时拥有的选项。它还对在 Dashcode 中打开一个 Widget（一种可让您弃用 Dashcode 的设计工具而对现有 Widget 进行编码、测试和调试的功能）进行了讨论。

使用模板来创建项目

Dashcode 会基于模板创建新项目。模板是预配置的 Widget 或 Web 应用程序，它们包括用于执行常见任务的代码和图形。当您打开 Dashcode 时或者在打开它之后选取“文件”>“新项目”，一个对话框会出现，供您选取 Widget 或 Web 应用程序项目类型。当您选取其中之一时，对话框会让您为项目类型选取合适的模板。点按一个模板的图标，以查看对该模板的功能的简短描述。如果模板符合您要尝试执行的任务，请选择它的图标并点按“选取”。有关 Dashcode 附带的模板的更多信息，请阅读[“Dashcode 模板”](#)（第 49 页）。

选取模板之后，您可能需要提供必需值以使 Widget 或 Web 应用程序正常工作。特别地对于 Widget，请确定提供“身份”和“属性”值，如[“提供 Widget 属性”](#)（第 28 页）中所述。

使用现有 Widget 来创建项目

Dashcode 可以使用现有 Widget 来创建新项目。如果您想要继续在 Dashcode 中处理某个现有 Widget，您应当导入它。导入一个 Widget 会将现有 Widget 拷贝到一个新的 Widget 项目，并启用代码生成器。当代码生成器处于活跃状态时，Dashcode 的所有设计和管理工具会被启用。

要导入一个 Widget，请选取“文件”>“导入 Widget”，或从模板选取器对话框中选取“导入”。

注：如果您的 Widget 使用在运行时创建的对象（例如那些从“Apple 按钮”或“Apple 滚动区域”创建的对象，这两种 Apple 类都是 Apple 提供给 Widget 开发人员），它们不会出现在画布上，因为显示 Widget 时没有执行它们的构造函数。

一旦您已将 Widget 导入到一个 Dashcode 项目，您可以使用“Widget 属性”面板（如[“提供 Widget 属性”](#)（第 28 页）中所讨论的）来编辑 Widget 的“身份”值。

提供 Widget 属性

每个 **Widget** 都含有您所需的值，以使 **Widget** 能够正常工作。您可以在“**Widget 属性**”面板中设定这些值，您可以通过在导航器中选择“**Widget 属性**”来访问该面板。请确定在编辑器的以下部分中提供值（如果可用的话）：

身份

此部分中的值用于标识一个 **Widget**。

您需要提供一个 **Widget** 标识符，供 **Dashboard** 用来将您的 **Widget** 与其他的区别开来。**Widget** 标识符通常以反向域名批注格式化，开头为一个顶级域名（例如 com），接着是公司或创建者名称（例如 apple），最后是一个唯一的产品名称（例如 my-fabulous-widget，从而生成一个诸如 com.apple.my-fabulous-widget 的名称）。

此外，您需要提供唯一的版本号。此号码供 **Dashboard** 用来区别 **Widget** 的版本，从而使它始终运行最新的版本。

注：Widget 标识符和版本字段与 CFBundleIdentifier 和 CFBundleVersion 信息属性列表值相对应，如 **Dashboard Info.plist Keys** 中所讨论的。

属性

此部分讲述了模板专有的选项。如果您使用的是已导入（或已打开）的 **Widget** 或“自定”模板，则此部分不存在。有关每个模板的特定属性，请阅读“**Dashcode 模板**”（第 49 页）。

为 Web 应用程序提供属性

每个 **Web** 应用程序项目都含有一些值，这些值是您需要提供以使应用程序正常工作的。您可以在“**Web 应用程序属性**”面板中设定这些值，您可以通过在导航器中选择“**应用程序属性**”来访问该面板。尽管仅需页面标题，您最好设定其他值以使 **Web** 应用程序如预期般工作。在编辑器的以下部分中提供值（如果可用的话）：

通用

此部分中的值用于标识 **Web** 应用程序。

在“页面标题”栏中，用合适、易读的名称来给 **Web** 应用程序命名。您提供的标题是网页中的 `<title>` 元素并且将在浏览器的标题栏中显示。

视口

此部分中的值会控制用户当他们在 **iPhone** 和 **iPod touch** 上使用您的 **Web** 应用程序时可以如何查看它。当用户将设备方向从纵向更改为横向时，网页可以更改比例以适合新屏幕方向。

“方向”值的两个选项是：

- “调整页面宽度以适合”。当您选取此选项后，当设备方向更改时 **Web** 应用程序会调整大小（即增加或减少宽度）。由于此选项会增强用户将 **Web** 应用程序看作独立的应用程序而不是网页的感觉，通常情况下建议 **Web** 应用程序选定此选项。

- “缩放页面以适合”。当您选取此选项后，当设备方向更改时，Web 应用程序的宽度不会更改，但比例会更改。换句话说，Web 应用程序的内容将看起来稍为大些。如果 Web 应用程序的布局比较复杂，而您不想在设备方向更改时更改宽度，您不妨选取此选项。

此部分中的“页面缩放”值会控制当用户在 iPhone 和 iPod touch 上查看 Web 应用程序时是否可以缩放它。通常情况下，由于您想要 iPhone 和 iPod touch 用户将 Web 应用程序当作独立的应用程序而不是网页来查看，建议您关闭“页面缩放”（即取消选择“允许用户调整页面缩放”）。

Web Clip

此部分中的值控制可以在 iPhone 和 iPod touch 上如何显示 Web 应用程序以及如何创建和显示 Web Clip 图标。

在 iPhone OS 2.0 和更高版本中，您可以选取使 Web 应用程序在全屏幕模式下可用，全屏幕模式会隐藏 Safari 工具栏和导航栏。如果您想要执行此操作，您还必须提供 Web Clip 图标，因为轻按 Web Clip 图标是用户可以在全屏幕模式下打开一个 Web 应用程序的唯一方法（在 Safari 中导航到应用程序不会在全屏幕模式下打开它）。如果您选取不支持全屏幕模式，您仍应当考虑提供 Web Clip 图标，以使用户可以便捷地打开 Web 应用程序。

要使 Web 应用程序在全屏幕模式下可用，请选择“显示为全屏幕应用程序（隐藏 Safari 工具栏和导航栏）”笔记格。这允许用户轻按 Web Clip 图标时在全屏幕模式下打开 Web 应用程序。请注意用户按下主屏幕按钮会离开以全屏幕模式运行的 Web 应用程序。

如果您选定了“显示为全屏幕应用程序（隐藏 Safari 工具栏和导航栏）”笔记格，则也可以指定以下三种样式的其中一种用于状态栏：

- 灰色
- 黑色
- 黑色（半透明）

使用弹出式菜单来选取与 Web 应用程序的外观最协调的状态栏样式。只有在选定了“显示为全屏幕应用程序（隐藏 Safari 工具栏和导航栏）”笔记格时才可以指定状态栏样式。这是因为当 Safari 状态栏可见时，状态栏外观会自动匹配它的外观。

“图标”值的两个选项是：

- “使用自定图标”。如果您想要设计自己的 Web Clip 图标来表示您的 Web 应用程序，请选取此选项。有关如何执行此操作的更多信息，请参阅[“为 Web 应用程序设计 Web Clip 图标”](#)（第 35 页）。

如果您选取此选项，则您也可以指定是否想要通过选择“添加玻璃视觉效果”笔记格来自动添加玻璃光泽叠层效果。请注意您无法控制 Web Clip 图标的圆角和投影；这些效果总是自动添加的。

- “使用 Safari 生成的图标”。如果您想要允许用户创建自己的 Web Clip 图标来表示您的 Web 应用程序，请选取此选项。

打开现有的 Widget

如果您想要弃用 Dashcode 的项目管理和设计工具，您可以在 Dashcode 中打开现有的 Widget，而不必将它导入到一个新项目中。当 Dashcode 打开一个 Widget 时，您可以访问它的代码和 Dashcode 的所有调试工具，但不能访问它的设计工具。

要在 Dashcode 中打开一个 Widget，请选取“文件”>“打开”，然后在“打开”对话框中选择一个 Widget，或选取“文件”>“新项目”，然后点按模板选取器对话框底部的“打开现有的”按钮。当您使用 Dashcode 打开一个 Widget 时，画布是锁定的。这意味着 Dashcode 的设计工具会被关闭，这些工具负责为 Widget 生成 HTML、CSS 和图像。当您打开一个现有 Widget 而不将它导入到一个新项目中时，Dashcode 会在画布的底部上显示一个锁。

设计 Widget 或 Web 应用程序

当已经开始一个项目之后（如[“开始一个项目”](#)（第 27 页）中所述），请使用 **Dashcode** 的设计工具来设计 **Widget** 或 **Web** 应用程序的界面。本章对画布、**Dashcode** 中的界面设计区域以及如何开始布置界面，进行了描述。本章还对如何将 **Dashcode** 部件添加到您的项目以及如何使用“检查器”窗口来调整部件的外观或行为，进行了论述。

请阅读本章，以了解如何使用 **Dashcode** 的设计工具来创建、修改和预览 **Widget** 或 **Web** 应用程序的外观。

布置界面

Dashcode 包括一个工作区，您可以在那里布置组成了 **Web** 应用程序 **Widget** 的元素。此工作区称为画布，当您在导航器中选择您的 **Widget** 或 **Web** 应用程序时，它在项目窗口的主体部分中可见。当画布可见时，您可以随意移动界面中的任何元素并调整其大小。要移动一个元素，请将它拖移到您想要的位置。要调整一个元素的大小，请拖移它的调整大小手柄之一。如果您在调整一个元素的大小时按住 **Shift** 键，则会保持原始比例。

Dashcode 还提供了一些命令，您可以用来排列和对齐 **Widget** 或 **Web** 应用程序界面中的元素。有关如何执行此操作的更多信息，请参阅[“排列和锁定元素”](#)（第 33 页）。

您可以从 **Finder** 将任何对象拖移到画布，以将它添加到 **Widget** 或 **Web** 应用程序。这意味着您可以在应用程序（如 **Adobe Illustrator** 或 **Photoshop**）中设计用户界面，存储图像，将它们从 **Finder** 拖移到您的 **Dashcode** 项目。尽管如此，**Dashcode** 还是提供了其自己的设计元素。请在[“将部件添加到界面”](#)（第 31 页）中了解有关这些元素的信息。

显示 Widget 的两面

默认情况下，所有从 **Dashcode** 模板创建的新 **Widget** 都具有两面：“front”和“back”。画布一次只显示这两面的其中一面。要在这两面之间切换，请在导航器中点按“front”或“back”。（如果在导航器中看不到“front”和“back”，请点按 **Widget** 的名称旁边的显示三角形。）

当一个面在画布上可见时，您可以在它上面添加和排列元素。要在另一面上工作，请在导航器中点按相应的名称。

将部件添加到界面

Dashcode 包括一组控制、形状和称为部件的显示。选取“窗口”>“显示资料库”，然后点按“部件”按钮以查看 **Dashcode** 附带的部件。有些部件（例如返回和前进按钮）专门供在 **Web** 应用程序中使用。

要给 Widget 或 Web 应用程序的界面添加一个部件，请将它从“部件资料库”拖移到画布上的 Widget 主体或 Web 应用程序页面。只要部件位于画布上，您就可以更改它的属性（如[“更改元素的属性”](#)（第 32 页）中所论述的）。如果您想要了解有关在项目的用户界面中布置部件的更多信息，请阅读[“排列和锁定元素”](#)（第 33 页）。

许多 Dashcode 部件可以有规划地进行控制。要了解有关使用这些部件的更多信息，请阅读[“Dashcode 部件”](#)（第 55 页）。

使用来自 iPhoto 的照片

您可以将 iPhoto 图库中的任何照片拖移到画布，以将它包括在 Widget 或 Web 应用程序中。要显示 iPhoto 图库，请选取“窗口”>“显示资料库”，然后点按“照片”按钮。只要照片位于画布上，您就可以排列它（如[“排列和锁定元素”](#)（第 33 页）中所述）以及调整它的属性（如[“更改元素的属性”](#)（第 32 页）中所述）。

注：iPhoto 图库要求 iPhoto 6 或更高版本。

更改元素的属性

“检查器”窗口会在画布上显示所选元素的属性。在此窗口中，您可以根据选取的检查器类型来查看和编辑元素的属性。通过点按其按钮（位于窗口的顶部）来选取检查器。“检查器”窗口包括以下五个检查器，从左到右排列：

- “属性”检查器允许您修改元素的 ID（在 JavaScript 用来引用元素）、它的 CSS 类（无论它是否显示在 Widget 或 Web 应用程序和默认图像中）以及属于元素的任何参数。
- 对所选元素来说，如果适合的话，“填充和线型”检查器将允许您调整一个形状或控制的填充样式、不透明度、边角圆度和线型样式。
- “版式”检查器允许您修改元素的大小和位置、以及它的行为（如果 Widget 或网页已被调整大小），控制方式如“自动调整大小”和“约束”部分中所述。“自动调整大小”设置会影响在画布上调整 Widget 或网页的大小时元素的行为。
- 如果所选元素显示了文本，则您可以在“文本”检查器中调整文本的字体、样式、颜色、大小、阴影、对齐和间距。您还可以设定文本是否绕排以及如何处理文本溢出。将文本溢出设定为“裁剪”会根据所选元素的边界来剪切其中的任何字符串，从而使文本不会溢出；反之，将文本溢出设定为“省略号”则会当文本达到所选元素的边界时，在文本的后面加上省略号字符 (...)。

注：如果您打算共享一个 Widget 或部署一个 Web 应用程序，请务必使用符合 Mac OS X 标准的字体，例如 Helvetica Neue、Times 和 Monaco。

- “行为”检查器允许您为各个事件分配一个元素 JavaScript 处理程序。对于每个事件，您可以分配现有的 JavaScript 函数作为其处理程序，或者创建一个空函数（自动添加到项目的 JavaScript 代码）。一旦您为事件分配了处理程序，请点按处理程序名称旁边的箭头以在源代码编辑器中显示函数。

[“添加源代码”](#)（第 37 页）中对处理代码进行了论述。

排列和锁定元素

Dashcode 提供了一些很有帮助的选项，用于在画布上排列元素。如果一个元素被其他元素所遮掩，请选择它并选取“排列”>“前移”或“排列”>“调到前面”，以将它移到当前遮掩它的所有元素的前面。类似地，如果您想要将一个元素移到另一个的后面，请选择它并选取“排列”>“后移”或“排列”>“送到后面”。

Dashcode 提供了对齐和分布选项，用于相对于彼此来排列多个元素。如果您想要将若干个对象向它们的左边缘对齐，请在画布上选择它们，然后选取“排列”>“对齐”>“左侧”。还有一些类似的选项可用于按居中、靠右、靠上、中间和靠下，将元素对齐。如果您想要让多个元素之间的垂直间距相等，请选取“排列”>“分布”>“垂直”。也存在将元素水平分布的类似选项。

如果您想要锁定一个元素，以便不再能够移动它，请选择该元素，然后选取“排列”>“锁定”。如果您改变了主意，想要再次移动它，请选择它，然后选取“排列”>“解锁”。请注意锁定一个元素只会锁定它在画布上的位置，您依旧可以在“检查器”中调整它的属性。

绝对定位与文稿流定位

Dashcode 允许您使用绝对定位或文稿流（即相对）定位来定位 Widget 或 Web 应用程序中的元素。您可以通过在“绝对的”和“文稿流”（在“版式”检查器的“布局”部分中）之间进行切换来更改所选元素的定位。

简而言之，绝对定位意味着元素的位置在它所包含的元素中始终都是相同的，而无论其同属元素的位置如何。换句话说，绝对定位的元素在包含的元素中会保持它的位置，即使它周围的同属元素扩展或收缩亦如此。绝对定位使您能够对复杂的布局进行精确的控制，但它对动态调整大小的支持则不怎么样。

与此相反，文稿流定位意味着元素的位置是根据其文稿流同属元素的位置进行定义的。在文稿流定位中，同属元素按顺序垂直显示，因此单个元素的位置取决于它在一系列元素中的位置。如果一个元素向下扩展，它会将其后面的同属元素向下推移相同的量，而不更改它们的大小或相对位置。虽然文稿流定位提供了巨大的灵活性以支持动态调整大小，但它无法使您对布局进行细致的控制。

由于 Widget 倾向于保持相同大小（或在少数几个预定义的大小之间切换），因此默认情况下 Dashcode Widget 项目会使用绝对定位。相反地，由于 Web 应用程序在大小方面需要更大的灵活性，因此默认情况下 Dashcode Web 应用程序项目会对大多数元素使用文稿流定位。

但是，项目中使用的默认定位样式并不意味着没有不同情况。例如，在浏览器模板 Web 应用程序项目中，列表行中的元素（行标题和行箭头）会被绝对定位，以确保布局一致。

自定一个项目时，您应当注意要更改的元素的定位样式。这是因为将一个部件拖入到画布上的现有容器元素，会导致该部件采用它新的同属元素的主要定位样式。如果您想让新的部件采用备选定位样式，请按住 Shift 键并将该部件拖移到画布上。

搜索元素

使用工具栏中的搜索栏来查找项目的界面和代码中的元素。当您在设计和编码一个 Widget 或 Web 应用程序时，这种特殊类型的搜索才能起作用。当 Dashcode 正在运行一个 Widget 时，工具栏的搜索栏会从运行日志或“栈框架与变量”表格缩窄结果，取决于哪一种可见。稍后会在[“运行日志和跟踪执行”](#)（第 44 页）和[“检查内存中的值”](#)（第 45 页）中对此进行论述。

要搜索一个元素，请在搜索栏中键入元素名称的一部分。当搜索结果返回时，导航器会被替换成搜索结果。选择搜索结果中的一个对象会在画布上将它高亮显示，或者显示包含元素的执行的源代码文件。

标尺和参考线

默认情况下，画布会在顶部和左侧显示标尺。与参考线配合使用时，标尺可以帮助您相对于另一个元素将元素对齐。要添加参考线，请将指针放在标尺中的任意位置，然后朝着画布的方向拖移。当您拖移出标尺时，会出现一条参考线，它会停留在您停止拖移的任何位置。要去掉参考线，请将它拖移回标尺。要隐藏标尺，请选取“显示”>“隐藏标尺”。

放置关闭方框

用户可以点按 **Widget** 的关闭方框以关闭它（**Web** 应用程序没有关闭方框）。默认情况下，**Widget** 的关闭方框出现在 **Widget** 主体的左上侧。如果有必要的话，请移动关闭方框，以便它覆盖 **Widget** 主体的左上角。要隐藏画布上的关闭方框，请选取“显示”>“隐藏不可视项目”。

停用画布

由于画布能够自动为您生成 **HTML** 和 **CSS**，如果您要手动调整元素，则不妨将它的代码生成关掉。要关闭自动代码生成器，请选取“显示”>“停止代码生成器”。当您完成手动调整值以后，您可以再次打开代码生成器，方法是选取“显示”>“起动代码生成器”。

预览 Widget 的默认图像

默认图像预览会向您显示 **Widget** 的默认图像，在 **Dashboard** 中载入 **Widget** 时也会显示该图像。要查看 **Widget** 的默认图像，请在导航器中选择“默认图像”。

默认情况下，除了文本部分（文本、文本区域和文本栏），**Widget**“front”上的所有元素都包括在 **Widget** 的默认图像中。其他显示文本的部分（例如弹出式菜单）也包括在内。因为当 **Widget** 被本地化时，界面上的任何文本都可能会更改，所以您无法将文本保留在 **Widget** 的默认图像中。因此，您应当从 **Widget** 的默认图像去掉显示文本的部分。要从 **Widget** 的默认图像去掉任何元素，请遵循以下步骤：

1. 在导航器中选择 **Widget** 对象以显示画布。
2. 在画布上，选择您想要将其从默认图像去掉的元素。
3. 显示“属性”检查器（如“[更改元素的属性](#)”（第 32 页）中所论述）。
4. 取消选择“在默认图像中显示”。

默认图像预览下面的工具栏为默认图像提供了附加的配置选项：

开始同步/停止同步

默认情况下，只要 Widget 的界面更改了，Dashcode 就会更新 Widget 的默认图像。要关闭此行为，请点按“停止同步”。要启用此行为（如果该行为已被关闭），请点按“开始同步”。

导入

如果您已经有一张图像，并且想要将其用作 Widget 的默认图像，请点按“导入”按钮，然后在出现的对话框中选择该图像。

在外部编辑器中打开

如果您想要在一个应用程序而非 Dashcode 中调整您的默认图像，请点按“在外部编辑器中打开”按钮。当您点按该按钮时，如 Dashcode 中所示的默认图像会在您的默认 PNG 处理应用程序中打开。

设计 Widget 图标

您可以使用 Widget 图标编辑器来设计 Widget 的图标。在 Dashboard Widget 栏和 Widget 管理器中，Widget 图标代表一个 Widget。要显示 Widget 图标编辑器，请在导航器中选择“Widget 图标”。

要修改 Widget 图标的主体的外观，请显示“填充和线型”检查器，然后更改填充样式、边角圆度、不透明度和线型样式。Widget 图标编辑器底部的工具栏为 Widget 图标提供了附加的配置选项：

开始同步/停止同步

默认情况下，Dashcode 会将 Widget 的 front 图像的样式与它的 Widget 图标同步。要关闭此行为，请点按“停止同步”。要启用此行为（如果该行为已被关闭），请点按“开始同步”。

放置

要在 Widget 的图标上放置图像（例如标志），请点按“放置”按钮并选择该图像文件。一旦放置好图像之后，您就可以移动它以及调整它的大小。

导入

如果您已经有一张图像，并且想要将其用作 Widget 的图标，请点按“导入”按钮，然后在出现的对话框中选择该图像。

在外部编辑器中打开

如果您想要在一个应用程序而非 Dashcode 中自定 Widget 图标，请点按“在外部编辑器中打开”按钮。当您点按该按钮时，图标会在您的默认 PNG 处理应用程序中打开。

为 Web 应用程序设计 Web Clip 图标

Web 应用程序可以为用户提供 Web Clip 图标以放置在他们的主屏幕上，并用作应用程序的书签类型。当用户轻按 Web Clip 图标时，Web 应用程序会自动打开，而无需用户浏览到它。Web Clip 图标应当简单、富有吸引力且容易被用户识别。

最好为大多数 Web 应用程序提供 Web Clip 图标，但对可以在全屏幕模式下运行的 Web 应用程序而言，Web Clip 图标则是必要的。这是因为用户可以体验 Web 应用程序的全屏幕模式的唯一方式便是通过轻按 Web Clip 图标来打开它。有关指定 Web 应用程序可以运行的模式的更多信息，请参阅[“为 Web 应用程序提供属性”](#)（第 28 页）。

注：由于 iPhone 和 iPod touch 给您所提供的图标添加了合适的边角半径和投影，因此您不应自己添加这些效果。然而，您可以通过在“应用程序属性”面板中指定一个值来控制是否自动添加玻璃光泽叠层效果（有关此操作的更多信息，请参阅[“为 Web 应用程序提供属性”](#)（第 28 页））。

您可以使用 Web Clip 图标编辑器来设计应用程序的图标。要显示 Web Clip 图标编辑器，请选择导航器中的 Web Clip 图标。

要修改 Web Clip 图标的主体的外观，请显示“填充和线型”检查器，然后更改填充样式、不透明度和线型样式。Web Clip 图标编辑器底部的工具栏为以下图标提供了附加的配置选项：

放置

要在 Web 应用程序的图标上放置图像（例如标志），请点按“放置”按钮并选择该图像文件。一旦放置好图像之后，您就可以移动它以及调整它的大小。

导入

如果您已经有一张图像，并且想要将其用作 Web 应用程序的图标，请点按“导入”按钮，然后在出现的对话框中选择该图像。

在外部编辑器中打开

如果您想要在一个应用程序而非 Dashcode 中自定 Web 应用程序图标，请点按“在外部编辑器中打开”按钮。当您点按该按钮时，图标会在您的默认 PNG 处理应用程序中打开。

添加源代码

开始一个项目并设计好用户界面之后，您可能需要给您的 **Widget** 或 **Web** 应用程序添加代码。本章介绍了如何查看组成您的项目的文件，以及如何给它们添加代码以执行不会由模板自动提供的任务。本章还向您示范了如何打开对资源（例如项目和网络外部的文件）的访问。最后，本章涵盖了能够影响在 **Dashcode** 中的代码编辑体验的应用程序偏好设置。

查看项目的源代码

Dashcode 包括了组成 **Widget** 或 **Web** 应用程序的所有 **HTML**、**CSS** 和 **JavaScript** 文件。当您想要添加模板不提供的功能性时，您需要查看并编辑这些实现文件。要显示项目的实现文件，请选取“显示”>“文件”。这会在导航器的下面显示“文件”列表。在“文件”列表中，您可以添加、复制文件和文件夹以及给其重命名，如“[添加影片文件和文件夹](#)”（第 39 页）中所述。

注：当您编辑文件的名称时，请确定在该文件的所有引用中进行了更改。文件引用通常位于 **Info.plist** 和 **HTML** 文件中。请务必检查这些文件，如果含有对旧文件名称的引用，请将它们更新为新名称。

当您在“文件”列表中选择 **HTML**、**CSS**、**JavaScript** 或属性列表文件时，源代码编辑器会出现在画布的下面，画布显示了所选文件的内容。在代码编辑器中，您可以编辑执行您的 **Widget** 或 **Web** 应用程序的实际 **HTML**、**CSS** 和 **JavaScript** 文件。有关使用 **HTML**、**CSS** 和 **JavaScript** 进行编码的更多信息，请阅读“[使用 HTML、CSS 和 JavaScript 编程界面](#)”（第 38 页）。

源代码编辑器的文本显示部分上面是两个弹出式菜单，一个文件弹出式菜单位于左侧，一个函数弹出式菜单位于该菜单的右侧。文件弹出式菜单列出了 **Widget** 或 **Web** 应用程序中的所有文本文件，您可以使用源代码编辑器来编辑这些文件。当您在文件弹出式菜单中选择一个文件时，它会显示在源代码编辑器中。函数弹出式菜单列出了 **JavaScript** 文件中所有函数的名称或 **CSS** 文件中所有类的名称。查看 **HTML** 和属性列表文件时函数弹出式菜单不可用。

添加代码的一种常见方法是给用户界面中的元素添加事件处理程序。您可以使用行为检查器来执行此操作，如“[为事件添加处理程序](#)”（第 38 页）中所述。

处理来自 **Widget** 或 **Web** 应用程序外部的资源（**XMLHttpRequest**、命令行工具、**Java applet** 等等）时，您需要明确打开对这些对象的访问。要了解有关资源访问以及什么元素需要激活的更多信息，请阅读“[资源访问](#)”（第 40 页）。

您还可以修改源代码编辑器的外观和行为，如“[代码编辑偏好设置](#)”（第 40 页）中所述。

使用 HTML、CSS 和 JavaScript 编程界面

当您显示一个项目的源代码之后，您可以修改它的 HTML、CSS 和 JavaScript 文件，如[“查看项目的源代码”](#)（第 37 页）中所述。任何能够在 Safari 及其 WebKit 引擎中工作的 HTML、CSS 或 JavaScript 代码都可以在 Widget 或 Web 应用程序中使用。要了解有关 HTML、CSS 和 JavaScript 的更多信息，请阅读以下文稿：

- [《Safari FAQ》](#)（Safari 常见问题解答）
- [《Web Kit DOM Programming Topics》](#)（Web Kit DOM 编程主题）
- [《Safari HTML Reference》](#)（Safari HTML 参考）
- [《Safari Web Content Guide for iPhone》](#)（用于 iPhone 的 Safari Web 内容指南）
- [《Web Kit DOM Reference》](#)（Web Kit DOM 参考）

要了解有关创建网页或 Web 应用程序以在 iPhone 或 iPod touch 中运行的更多信息，请参阅[《Safari Web Content Guide for iPhone》](#)（用于 iPhone 的 Safari Web 内容指南）。

除了 WebKit 的 HTML、CSS 和 JavaScript 编程界面，Apple 还提供了专用于 Dashboard 的编程界面，以供在 Widget 中使用。有关这些编程界面的更多信息，请阅读[《Dashboard Programming Topics》](#)（Dashboard 编程主题）和[《Dashboard Reference》](#)（Dashboard 参考）。

为事件添加处理程序

当您使用“行为”检查器来添加处理程序时（如[“更改元素的属性”](#)（第 32 页）），新的处理程序函数会被插入到项目的 JavaScript 文件中。当您在行为检查器中点按处理程序函数名称旁边的箭头时，源代码编辑器会与所选函数一起出现在画布的下面。然后您就可以添加您的自定义代码以处理该事件。

有关使用源代码编辑器的更多信息，请阅读[“查看项目的源代码”](#)（第 37 页）。要了解有关 JavaScript 和 Dashboard 编程界面的信息，请阅读[“使用 HTML、CSS 和 JavaScript 编程界面”](#)（第 38 页）。

为自定控制器添加代码

如果您想要通过更改或添加视图（例如列表视图或详细视图）来修改 Web 应用程序，则您也需要添加代码来控制它们。Web 应用程序中的每种视图都是由 JavaScript 控制器对象控制的。对于包含列表的视图，控制器对象充当了列表的数据来源。较常见的情况是，控制器对象负责配置一种视图范围内的界面，以便界面正常显示，并且视图能够处理在它里面发生的所有活动。

有关您可以添加到 Web 应用程序的一些部件的更多信息，请参阅[“Dashcode 部件”](#)（第 55 页）。要了解有关 JavaScript 和 Dashboard 编程界面的信息，请阅读[“使用 HTML、CSS 和 JavaScript 编程界面”](#)（第 38 页）。

使用“代码预见”来完成代码

Dashcode 的源代码编辑器包括一种代码完成功能，该功能可以基于部分已输入的文本来建议可能的变量和函数名称。当“代码预见”有建议时，它会给您正在键入的文本加下划线。要查看建议的列表，请按下 **Option** 和 **Escape** 键。使用上箭头和下箭头键来选择您想要的符号。要添加符号，请按下 **Tab** 键。要离开建议列表，请按下 **Escape** 键。

有关如何设定“代码预见”偏好设置的更多信息，请参阅[“代码编辑偏好设置”](#)（第 40 页）。

代码片段

Dashcode 包括一组可重复使用的代码片段。每个代码片段都提供了一些功能性，您可以用它们来增强您的 **Widget** 或 **Web** 应用程序。要显示代码片段，请选取“窗口”>“显示资料库”，然后点按“代码”按钮。要将代码片段添加到您的项目，请将它从代码库拖移到源代码编辑器。

除了 Dashcode 附带的代码片段，您还可以存储自己的片段以供以后使用，方法是将它们从源代码拖移到您在代码库中创建的一个组别。要创建一个新组别，请从代码库的列表下面的操作菜单中选取“新建组别”。

添加影片文件和文件夹

您可以给 Dashcode 中的 **Widget** 或 **Web** 应用程序添加附加文件和文件夹。虽然很少有必要这样做，但在整理大量文件或功能性的区域时，这是非常有帮助的。要添加新文件，请显示文件列表，选取“文件”>“新建”，然后选取“JavaScript 文件”、“CSS 文件”或“文件”。创建新文件之后，您可以查看并编辑其内容，如[“查看项目的源代码”](#)（第 37 页）中所述。

要给项目添加文件夹，请选取“文件”>“新建”>“文件夹”。要给文件夹添加对象，请将它们拖移到它的图标上。

在“文件”列表中，您可以复制、移动和删除文件和文件夹以及给其重命名。要给对象重命名，请在“文件”列表中选择该对象，然后从操作菜单（“文件”列表下面的齿轮图标）中选取“重命名”。要复制一个对象，请从操作菜单中选取“复制”。要移走一个对象，请从操作菜单中选取“移到废纸篓”。要移动一个对象，请将其图标拖移到另一个文件夹的图标。

注：Dashcode 不会管理对 Widget 或 Web 应用程序中的文件和文件夹的引用。如果您要移走、重命名或移动一个文件或文件夹，则需要移走、重命名或更新对项目主 HTML 文件中的文件或文件夹的引用。无法更新对已移走、已重命名或已被移动的文件或文件夹的引用，可能会导致发生意外的行为。

当您添加新的 JavaScript 或 CSS 文件时，您需要包括对文件的引用，以便它可以被您的 Widget 或 Web 应用程序使用。给项目主 HTML 文件顶部的新文件添加引用。在一个 Widget 中，主 HTML 文件是与 Widget 的 Info.plist 文件中的 MainHTML 键配对的文件。在一个 Web 应用程序中，主 HTML 文件是 index.html 文件。

在项目的主 HTML 文件中，您可以使用 <style> 标记，给使用 @import 指令的新 CSS 文件添加引用。使用 <script> 标记给新 JavaScript 文件添加引用。

资源访问

如果您打算在您的 Widget 中使用任何以下资源，则需要先打开对它们的访问：

- 网络资源（包括 XMLHttpRequest）
- 外部文件
- Internet 插件，例如 QuickTime 和 Quartz Composer
- Java applet
- 命令行应用程序

重要事项：通常情况下，iPhone 上的 Safari 不支持要求访问文件系统的任何第三方插件或功能，因此大部分这些资源不适合设计用于 iPhone 或 iPod touch 的 Web 应用程序。

然而，iPhone 上的 Safari 的确支持 XMLHttpRequest。要了解有关 iPhone Web 应用程序中不支持的技术的更多信息，请参阅《Safari Web Content Guide for iPhone》（用于 iPhone 的 Safari Web 内容指南）。

要打开对这些资源的访问，请在导航器中选择“Widget 属性”，然后为您正在尝试使用的资源选择选项。选项包括：

网络/磁盘访问

如果您的 Widget 要求访问网络资源或磁盘上的文件，请选择合适的对象。除非您的模板已经需要访问网络资源（例如 XMLHttpRequest）和 Widget 包外部的磁盘上的文件，否则这些资源是关闭的。

扩展名

如果您的 Widget 使用通过 Internet 插件或 Java applet 提供的内容，或者通过 widget.system 使用命令行实用工具，请选择合适的选项。除非您的模板已经需要访问插件（例如 QuickTime）、Java 和命令行实用工具，否则这些资源是关闭的。

代码编辑偏好设置

源代码编辑器具有许多偏好设置，您可以设定这些偏好设置来更改它的行为，以便更好地适合您的喜好。要显示这些偏好设置，请选取“Dashcode”>“偏好设置”。“偏好设置”窗口包括以下对象：

- “通用”偏好设置包括一个用于在 **Dashcode** 或帮助应用程序中打开源代码文件的选项。
- 源代码编辑器具有多种可见性和行为偏好设置，包括间隔和行号（在源代码的旁边）的可见性、源代码编辑器的自动换行和 **Tab** 键的缩进行为。您可以在“编辑”偏好设置中设定这些偏好设置。
- 源代码可以基于代码的语法进行着色。您可以在“格式化”偏好设置中调整文本字体、大小和颜色。
- **Dashcode** 提供了一种称为“代码预见”（如[“使用“代码预见”来完成代码”](#)（第 39 页）中所述）。在源代码编辑器的顶部，一个弹出式菜单显示了文件中的符号。“代码预见”偏好设置包括一项用于如何组织这些符号的设置。此面板还包括用于代码完成的设置。

测试和共享

为 Widget 或 Web 应用程序编写代码之后，请进行测试以确保它能够像预期的一样工作。Dashcode 包括自己的 Dashboard 环境，因此您可以在 Dashcode 中运行和测试 Widget，而不必在 Dashboard 中打开它。类似地，Dashcode 包括一个能够模拟 iPhone 和 iPod touch 的某些功能的模拟程序，因此您可以轻松地测试 Web 应用程序。

除了运行 Widget 或 Web 应用程序，Dashcode 还包括一些工具，这些工具可以帮助您调试并修复可能发生的错误。本章包括有关在执行某个 Widget 或 Web 应用程序时如何跟踪事件和检查变量值的信息，还包括有关共享 Widget 和部署 Web 应用程序的信息。

测试 Widget 或 Web 应用程序

要在 Dashcode 中运行一个 Widget 或 Web 应用程序，请选取“调试”>“运行”，或在工具栏中点按“运行”。取决于您的项目类型，这能够起到不同的效果：

- 对于 Widget，这会在 Dashcode 中运行 Widget，而不必在 Dashboard 中打开它。Widget 在此处的行为方式与在 Dashboard 中相同。
- 对于 Web 应用程序，这会在模拟程序中运行 Web 应用程序，而不必在设备中部署并运行它。模拟程序允许 Web 应用程序在其中运行，如同应用程序在设备中运行时一样。

此外，当在 Dashcode 2.0.1 或更高版本中测试 Web 应用程序时，有两个选项供您选择：

- 在 Safari 界面可见的状态下运行 Web 应用程序（也就是让 Safari 的工具栏和导航栏都可见）。这会模仿用户在使用 Safari 浏览到您的 Web 应用程序时所看到的情景。
- 在隐藏 Safari 界面的状态下运行 Web 应用程序（也就是在全屏幕模式下）。这会模仿用户轻按一个 Web Clip 图标以打开可以在全屏模式下运行的 Web 应用程序时所看到的情景。

您可以通过在 Dashcode 项目窗口中按住“运行”按钮来访问这些选项。从出现的菜单中选取“运行”或“在全屏幕模式下运行”。选择之后，点按“运行”或选取“调试”>“运行”会让您的 Web 应用程序在该模式下运行，直到您更改运行模式。

载入 Widget 或 Web 应用程序之后，请进行完整的测试以查看它是否能够像预期的一样运行。发生错误时，程序会抛出异常，而且在默认情况下，执行会暂停。发生这种情况时，有许多工具可供您任意使用，这些工具可以帮助您查看问题所在以及检查变量值：

- 运行日志列出了错误及其他有用信息。如果您启用跟踪，则运行日志还会记录函数的调用。有关详细信息，请参阅[“运行日志和跟踪执行”](#)（第 44 页）。
- “栈框架与变量”表格向您展示了 Widget 或 Web 应用程序中使用的任何变量的值。有关详细信息，请参阅[“检查内存中的值”](#)（第 45 页）。
- 逐步执行允许您逐行完成执行，从而可以查看代码的效果。有关详细信息，请参阅[“暂停和逐步执行”](#)（第 44 页）。

- 评估者是一个控制台，在那里您可以输入单行代码以立即执行。有关详细信息，请参阅[“代码评估者”](#)（第 45 页）。

要继续已暂停的 **Widget** 或 **Web** 应用程序，请选取“调试”>“继续”，或在工具栏中点按“继续”。发生异常时，要关闭暂停，请取消选择“调试”>“在异常处断开”。

要随时暂停一个 **Widget** 或 **Web** 应用程序，请选取“调试”>“暂停”，或在工具栏中点按“暂停”按钮。当 **Widget** 或 **Web** 应用程序被暂停时，您可以逐行执行它的代码（如[“暂停和逐步执行”](#)（第 44 页）中所述）。当执行到达代码的某个特定行时，您还可以使用断点来暂停 **Widget** 或 **Web** 应用程序，如[“断点”](#)（第 45 页）中所述。

要停止在其测试环境中运行一个 **Widget** 或 **Web** 应用程序，请选取“调试”>“停止”，或在工具栏中点按“停止”。

运行日志和跟踪执行

当您在 **Dashcode** 中运行一个 **Widget** 或 **Web** 应用程序时，画布会被运行日志取代。运行 **Widget** 或 **Web** 应用程序时，**Dashcode** 遇到的任何错误都会在此处报告。例如，对不存在的函数的调用，在运行日志中出现为“不允许调用的对象”。

如果运行日志被另一种显示取代，请选取“显示”>“运行日志”以显示它。如果可能的话，选择运行日志中的一行会在源代码编辑器中显示关联的代码行。

除了报告错误，运行日志还可以用于跟踪 **Widget** 或 **Web** 应用程序的执行。跟踪意味着当某个函数开始和完成时都会在运行日志中创建一个条目。当您在观察某个 **Widget** 或 **Web** 应用程序中的执行的流程时，这非常有用。

当运行日志可见时，您可以过滤运行日志的内容，方法是在工具栏的搜索栏中键入一个术语。

暂停和逐步执行

任何时候，您都可以暂停一个 **Widget** 或 **Web** 应用程序的执行。如果您想要检查变量的值，则暂停执行非常有用。要暂停，请选取“调试”>“暂停”，或在工具栏中点按“暂停”。当 **Widget** 或 **Web** 应用程序已暂停时，运行日志会被替换成“栈框架与变量”表格。

当 **Widget** 或 **Web** 应用程序的执行被暂停时，您可以在当前函数的范围和全局范围内检查变量的值。此外，被暂停的代码行在源代码编辑器中会被高亮显示。在源代码编辑器下面的快捷菜单中，您可以看到组成当前执行函数的函数的层次。如果您点按快捷菜单中的另一个函数名称，则函数的变量会显示在“栈框架与变量”表格中，而其代码会显示在代码编辑器中。您还可以搜索某个特定变量，方法是在工具栏的搜索栏中键入它的名称。

当您完成检查变量之后，您可以用以下几种不同方法来继续执行 **Widget** 或 **Web** 应用程序：

继续

选取“调试”>“继续”，或在工具栏中点按“继续”，以便继续执行而不中断。

单步进入

选取“调试”>“单步进入”，或在工具栏中点按“单步进入”，以便执行下一行代码并单步进入函数调用，从而您可以看到该行代码所具有的效果。

单步跳过

选取“调试”>“单步跳过”，或在工具栏中点按“单步跳过”，以便执行下一行代码，从而您可以看到该行代码所具有的效果。

单步结束

选取“调试”>“单步结束”，或在工具栏中点按“单步结束”，以便执行当前函数的剩下部分；当函数完成之后，执行会暂停，从而您可以在撤回变量之前先检查函数的变量。

此外，只要发生异常，执行就会暂停。发生异常时，**Web** 应用程序或 **Widget** 的执行会暂停，运行日志中的一个条目解释了问题是什么。如果您点按该条目，发生异常的那行代码会被高亮显示。默认情况下，此选项已启用。您可以通过选取“调试”>“在异常处断开”来控制此选项。

检查内存中的值

“栈框架与变量”表格显示了 **Widget** 或 **Web** 应用程序中使用的变量的值。当 **Widget** 或 **Web** 应用程序正在运行时，您可以检查它的全局变量。当暂停一个 **Widget** 或 **Web** 应用程序时（如“[暂停和逐步执行](#)”（第 44 页）中所述），当前函数会获得它自己在表格中的标题，并且在下面列出它的变量。如果函数被另一个函数调用，则第二个函数的变量会在第一个的下面列出，依此类推。

连接表格中的一个变量会将变量名称添加到代码评估者中，如“[代码评估者](#)”（第 45 页）中所述。

断点

除了使用“暂停”选项来暂停一个 **Widget** 或 **Web** 应用程序的执行，您还可以在代码中设定用于暂停执行的位置（称为断点）。您可以用两种方法来添加断点。一种方法是在源代码编辑器的间隔处点按。如果没有显示间隔，请前往“编辑”偏好设置并选择“显示间隔”选项。

间隔中的蓝色箭头表示在执行该行代码之前，执行将被中断。要临时停用断点，请点按它；这时断点会由蓝色变成灰色，表示它已被停用。要去掉断点，请将它拖移出间隔。

设定断点的另一种方法是使用“断点”窗口。要显示“断点”窗口，请选取“调试”>“显示断点窗口”，或在间隔处点按。在“断点”窗口中，点按加号按钮 (+)，然后指定要在该处断开的文件名称、行号或函数名称。例如，如果您想要在第 42 行的“Untitled.js”处断开，您可以输入“Untitled.js:42”。或者，您也可以提供想要在该处断开的函数的名称（如 **widget** 中的 `showFront`）。

除了断点，您还可以为断点设定一个条件。该条件是一个 **JavaScript** 语句，它的值为 `true` 或 `false`。当执行通过断点时，如果条件的值为 `true`，则执行会暂停。

在“断点”窗口中，您可以去掉某个断点，方法是从表格中选择一个断点并点按减号按钮 (-)；或者您也可以停用它，方法是取消选择表格中断点的对象旁边的记号格。

代码评估者

当您运行一个 **Widget** 或 **Web** 应用程序时，它对于只执行一行代码来说是非常有用的。代码评估者可让您执行此操作。要显示代码评估者，请选取“显示”>“评估者”，或从工具栏的“显示”菜单中选取“评估者”。在评估者中，您可以输入任意代码并按下 **Return** 键，以立即执行该代码。

如果您连接“栈框架与变量”表格中的一个值，它的名称会自动输入到代码评估者中。此外，如果光标位于提示符所在的位置，并且您按下上箭头键，则可以在代码评估者中的历史条目之间循环。

共享 Widget

当您完成开发一个 Widget 之后，您可以与其他人共享它。有两个命令可用于部署 Widget，而且两者都可以将 Widget 从项目导出为 .wdgt 包，以准备好在 Dashboard 中运行。当您在导航器中点按“共享”时，共享面板会让您选择是将 Widget 部署在 Mac OS X v10.4.3 或更高版本中运行，还是将其部署在所有版本的 Mac OS X v10.4 中运行。大多数情况下，建议您将 Widget 部署用于 Mac OS X v10.4.3 或更高版本，因为这会生成一个较小的 Widget。

如果想要共享您的 Widget（例如通过电子邮件将其发送），请在共享面板中点按“存储到磁盘”并选取一个方便的存储位置。如果想要确定部署前先将 Widget 项目自动存储到磁盘，请选择“部署前先将项目存储到磁盘”注册格。请注意，在您第一次存储 Widget 项目后，此注册格才可用。

如果 Widget 仅供您使用，请选取“文件”>“部署 Widget”，以将它安装在您的电脑上。

部署 Web 应用程序

要部署您的 Web 应用程序，请选取“文件”>“部署 Web 应用程序”。Dashcode 会要求您给该应用程序命名，并选取要存储它的位置。点按“存储”之后，Finder 窗口会打开到您所选取的位置，并显示以应用程序的名称命名的文件夹。如果您打开此文件夹，将看到若干个项目，这些项目包括：

- 一个用于表示您的 Web Clip 图标的 PNG 文件（有关设计此图标的指南，请参阅[“为 Web 应用程序设计 Web Clip 图标”](#)（第 35 页））
- 一个包含用于创建 Web 应用程序中的视觉元素（例如“返回”按钮）的图像文件夹
- index.html、main.css 和 main.js 代码文件
- 一个包含 JavaScript 文件的部件文件夹，这些 JavaScript 文件供 Web 应用程序中的部件（例如浏览器或标头）使用

部署 Web 应用程序之后，您需要在 Web 服务器的根文稿中安装它。如果 Web 服务器正在您用来在 Dashcode 中进行开发的电脑上运行，则只需将已部署的 Web 应用程序文件夹拷贝到合适的位置即可，例如“/资料库/Sites”。如果 Web 服务器在另一台电脑上，则您可以使用网络传输协议（例如 FTP、SSH 或 WebDAV）来拷贝已部署的 Web 应用程序文件夹。请联系 Web 服务器的管理员以了解如何执行此操作。



警告： Web 浏览器（包括 iPhone 上的 Safari）执行称为“相同来源”的安全模式。这意味着不允许网页从它们所来自的 Internet 域之外的域请求信息。如果您已经构建了一个能够从其他网站获取信息进行显示的 Web 应用程序，当您在 Dashcode 的模拟程序中对它进行测试时，它将工作；但是当您在 Web 服务器中安装它以及在 iPhone 或 iPod touch 中运行它时，它将不工作。

有几种方法可以配置 Web 服务器和修改 Web 应用程序，从而使从其他网站获取信息变成可能，但这些技巧不在本文稿的讨论范围内。请联系 Web 服务器的管理员以获取建议。

有关 Widget 的高级主题

除了设计、编码和测试 Widget 等核心任务之外，Dashcode 还提供了附加的高级功能，这些功能可以帮助您开发符合用户需求的 Widget。本章涵盖了本地化技巧以及如何包含以 Objective-C 编写的插件。

本地化

您可以在“Widget 属性”面板的“本地化”部分为您的 Widget 提供本地化的字符串。要显示“Widget 属性”面板，请在导航器中点按“Widget 属性”。

要将您的 Widget 本地化为其他语言，请点按加号按钮 (+) 并选取语言。在第一个表格中的语言名称的旁边，您可以为您的 Widget 提供本地化的名称。此名称供 Finder 以及 Dashboard 的 Widget 栏中的 Widget 使用。

右侧的表格显示了您的 Widget 中使用的所有字符串。在此表格中，您可以连接“值”列中的一个术语，然后提供本地化版本的 Widget 字符串。您可以添加键值组合用于本地化，方法是点按加号按钮，提供唯一的键名称，然后在它的键的旁边输入本地化版本的字符串。

当您添加一种语言用于本地化时，会将一个语言项目文件夹添加到该 Widget。除了字符串，您还可以在语言项目文件夹中放置任何本地化的资源，例如样式表或图像。要显示语言项目文件夹，请选取“显示”>“文件”，然后查找以 .lproj 扩展名结尾的文件夹。有关 Widget 本地化的更多信息，请阅读《Dashboard Programming Topics》（Dashboard 编程主题）中的“Localizing Widgets”（将 Widget 本地化）。

要使测试您的 Widget 的本地化变得简单，请选取“显示”>“自定工具栏”。在对话框中，将“语言”弹出式菜单拖至项目窗口的工具栏中，然后点按“完成”。现在您便可以从“语言”弹出式菜单中选取一种语言，以便看到您的 Widget 采用该语言。如果您点按工具栏中的“运行”，则您的 Widget 会以所选语言运行。

Widget 插件

Dashcode 提供了一个将 Widget 插件与 Widget 联合在一起的位置。如果您已经使用 Xcode 构建了一个自定的 Widget 插件，并且想要将它包括到您的 Widget，请从导航器中选择“Widget 属性”，然后在“插件名称”部分中，点按“选取”按钮并选择一个已构建的 Widget 插件。

有关 Widget 插件的更多信息，请阅读《Dashboard Programming Topics》（Dashboard 编程主题）中的“Creating a Widget Plug-in”（创建 Widget 插件）。

Dashcode 模板

Apple 随 Dashcode 提供了许多模板。请阅读此附录以了解有关每个 Dashcode 模板的特色的更多信息。

Widget 模板

“自定”Widget 模板

Widget 的“自定”模板会创建一个空白 Widget，含有制作一个 Widget 所需的基本文件和图像。使用此模板制作而成的 Widget 不含有其他预先配置的功能。

“倒计时器”模板

“倒计时器”模板会创建一个含预配置的、数码风格的倒计时器的 Widget。要自定此模板，请添加一张图片或其他插图，然后设定倒计时的目标事件。要设定倒计时的目标事件，请在导航器中点按“Widget 属性”，然后修改可从“属性”下面访问的选项。

您可以设定的选项包括：

目标类型

将特定的日期与时间或共享的 iCal 日历设定为 Widget 的目标时间。如果您将 Widget 与共享的 iCal 日历联合在一起，则它会朝着日历上的下一个事件进行倒计时。在该事件之后，Widget 会倒计时直到日历上的下一个事件，依此类推。有关 iCal 和日历发布的更多信息，请阅读[“Mac 101, Lesson 10: iCal”](#)。

达到目标时

当达到目标事件时，您的 Widget 可以停止计数，或者从该事件开始向上计数。此外，如果您选取“执行动作”并提供一个处理程序，则它可以触发 JavaScript 函数，从而使您的 Widget 能够有规划地对事件作出响应。

显示

选择时间单位之间的分隔符冒号是否闪动，以及是否显示每个单位数时间单位的前导零字符。

“地图”模板

“地图”模板会创建一个能够在地图上显示位置和位置专用信息的 Widget。要自定此模板，请在画布上更改它的外观，并提供您发布的地图的 URL。此外，您还可以更改 Widget 显示位置信息的方式。

“地图”模板能够与 KML 文件和 GeoRSS 提要配合使用。要设定 Widget 的目标地图，请在导航器中点按“Widget 属性”，然后修改可从“属性”下面访问的选项。

您可以设定的选项包括：

地图 API 密钥

请在此处提供您的地图 API 密钥。有关注册地图 API 的更多信息，请参阅 Google 开发人员网站上的 [Google Maps API](#)。

初始地址

请在此处提供您想要在 Widget 中显示的开始地址。

Mashup URL

将您的地图提要的 URL 粘贴到此处。有关发布个性化地图的更多信息，请参阅 Google 开发人员网站上的“[How to Create a Map](#)”（如何创建地图）。

注：您可能发现当您把 KML 文件中的点数限制为约 60 个并且将文件的未压缩的大小限制为约 1 MB 时，获得的结果最佳。

RSS Widget 模板

RSS 模板会创建一个显示从 RSS 源来的标题的 Widget。它适合经常更新的 RSS 提要，因为它能够同时显示许多标题。要自定此模板，请在画布上更改它的外观，并提供一个 RSS 提要 URL 以供 Widget 显示。要设定 Widget 的目标 RSS 提要，请在导航器中点按“Widget 属性”，然后修改可从“属性”下面访问的选项。

您可以设定的选项包括：

提要 URL

将 RSS 提要的 URL 粘贴到此处。

显示文章

调整您的 Widget 所显示的文章的篇数，以及文章生成的时间的范围。

显示

选择是否随文章显示日期与时间，以及是否显示新内容标记。

Podcast 模板

Podcast 模板会创建一个显示和播放 podcast 中专题节目的 Widget。要自定此模板，请在画布上更改它的外观，并提供一个 podcast URL 以供 Widget 回放。要设定 Widget 的目标 podcast 提要，请在导航器中点按“Widget 属性”，然后修改可从“属性”下面访问的选项。

您可以设定的选项包括：

Podcast URL

将 podcast 的 URL 粘贴到此处。

核查更新

设定 Widget 核查新专题节目的频率。

Photocast 模板

Photocast 模板会创建一个 Widget，该 Widget 能够将从 iPhoto Photocast 获取的图片显示成幻灯片显示。要自定此模板，请在画布上更改它的外观，并提供一个 Photocast URL 以供 Widget 显示。要设定 Widget 的目标 Photocast，请在导航器中点按“Widget 属性”，然后修改可从“属性”下面访问的选项。

您可以设定的选项包括：

Photocast URL

将 Photocast 的 URL 粘贴到此处。

更改图片

调整更改图片的频率。

过渡和方向

设定更改图片时使用的过渡，以及过渡出现的方向（如果可用的话）。

显示标题

设定显示 Photocast 标题需满足的条件。

Quartz Composer 模板

Quartz Composer 模板会创建一个显示 Quartz Composer 合成的 Widget。Quartz Composer 合成效果在 Widget 中可用于处理和显示图形数据。要自定此模板，请显示“文件”列表，然后在 Quartz Composer 中打开 Default.qtz。Quartz Composer 已作为 Developer Tools（Mac OS X v10.5 版本）的一部分安装了。

有关使用 Quartz Composer 的更多信息，请阅读《*Quartz Composer Programming Guide*》（Quartz Composer 编程指南）。有关 Quartz Composer WebKit 插件（驱动 Quartz Composer 模板的插件）的更多信息，请阅读《*Quartz Composer Web Kit Plug-in JavaScript Reference*》（Quartz Composer Web Kit 插件 JavaScript 参考）。

注：使用 Quartz Composer 模板创建的 Widget 与 Mac OS X v10.4.7 和更高版本兼容。

“视频 Podcast”模板

“视频 Podcast”模板会创建一个显示和播放视频 podcast 中的专题节目的 Widget。要自定此模板，请在画布上更改它的外观，并提供一个视频 podcast URL 以供 Widget 回放。要设定 Widget 的目标视频 podcast 提要，请在导航器中点按“Widget 属性”，然后修改可从“属性”下面访问的选项。

您可以设定的选项包括：

Podcast URL

将视频 podcast 的 URL 粘贴到此处。

核查更新

设定 Widget 核查新专题节目的频率。

“量表”模板

“量表”模板会创建一个带有量表和指示器的 Widget，量表和指示器对监视活动非常有用。要更改量表和指示器的数据来源，请从“文件”列表中选择“SystemMonitor.js”，然后编写代码以解释您的源数据，以及给量表和指示器提供结果。

“每日提要”模板

“每日提要”模板会创建一个 **Widget**，该 **Widget** 会显示来自 **RSS** 源的文章或图像。它适合不经常更新的 **RSS** 提要，因为它一次只能显示一篇文章。要自定此模板，请在画布上更改它的外观，并提供一个 **RSS** 提要 **URL** 以供 **Widget** 显示。要设定 **Widget** 的目标 **RSS** 提要，请在导航器中点按“**Widget** 属性”，然后修改可从“属性”下面访问的选项。

您可以设定的选项包括：

提要 **URL**

将 **RSS** 提要的 **URL** 粘贴到此处。

提要类型

指定提要提供的信息类型。如果它提供的是 **HTML** 内容，请选取“**HTML**”。如果您的提要提供的是图像（例如连环画或每日照片），请选取“**图像**”。

Web 应用程序模板

“自定”Web 应用程序模板

Web 应用程序的“自定”模板会创建一个空白 **Web** 应用程序，内含制作 **Web** 应用程序所需的基本文件和图像。使用“自定”模板开发的 **Web** 应用程序适合在 **iPhone** 和 **iPod touch** 上的 **Safari** 中运行。从此模板制作而成的 **Web** 应用程序不含有其他预配置的功能。

“浏览器”模板

“浏览器”模板会创建支持在多层内容中浏览的 **Web** 应用程序。使用“浏览器”模板开发的 **Web** 应用程序非常适合用于显示层级信息，用户可以在信息中往下浏览。

“浏览器”模板提供的默认 **Web** 应用程序会包含内容顶层的列表视图，以及内容的第二层的详细视图。在顶层列表中选择项目会自动显示与该项目关联的详细视图。您可以添加附加的列表层以及给详细视图添加内容。

要自定此模板，您可以调整用户如何在设备中体验应用程序，并且您可以在详细层视图使用不同部件，比如栏布局。要设定用户如何在 **iPhone** 和 **iPod touch** 中查看 **Web** 应用程序，请点按“应用程序属性”并修改“视口”下方的可用选项。

您可以设定的选项包括：

当设备方向更改时

设定 **Web** 应用程序是否应该更改它的宽度或缩放。

页面缩放

设定用户是否应该能够缩放 **Web** 应用程序，以及如果能够的话缩放系数的最大值为多少。

Podcast Web 应用程序模板

Podcast 模板会创建一个 Web 应用程序，该应用程序会显示和播放您发布的 podcast 中的专题节目。模板设置为显示一系列专题节目，这些专题节目当用户选择它们时会播放。

要自定此模板，请提供您发布的 podcast 的 URL。要这样做，请点按导航器中的“应用程序属性”并修改“属性”下方的选项。（您还可以设定值用于定义用户可以如何查看 Web 应用程序，如[“浏览器”模板](#)（第 52 页）中所述。）

您可以设定的选项为：

Podcast URL

将您发布的 podcast 的 URL 粘贴到此处。

RSS Web 应用程序模板

RSS 模板会创建一个显示来自 RSS 源的标题和文章的 Web 应用程序。由于此模板允许您立即显示许多标题，使用它开发的 Web 应用程序可以处理那些更新非常频繁的 RSS 提要。

要自定此模板，请提供您发布的 RSS 提要 URL。您还可以调整显示多少文章和显示多久。要自定这些属性，请在导航器中点按“应用程序属性”，然后修改“属性”下方的选项。（您还可以设定值用于定义用户可以如何查看 Web 应用程序，如[“浏览器”模板](#)（第 52 页）中所述。）

您可以设定的选项包括：

提要 URL

将您发布的提要的 URL 粘贴到此处。

显示文章

调整在 Web 应用程序中显示多少文章、在什么时间段内文章会发布以及显示多少“热门报道”。

“实用工具”Web 应用程序模板

“实用工具”模板会创建一个 Web 应用程序，此应用程序会在前端显示主要信息并且在后端显示多种视图或配置选项。使用此模板开发的 Web 应用程序非常适用于以一种格式显示少量的目标信息，此格式会使人联想起 Dashboard Widget。

要自定此模板，请将内容放置在前视图，并且将您想要提供的视图或配置选项放置在设置视图。由于用户知道他们可以轻按此控制来查看应用程序的后视图，请务必将信息按钮放置在适当的位置（模板会在前视图的右下角提供此按钮）。点按导航器中的“应用程序属性”以设定一些值（定义用户查看 Web 应用程序的方式），如[“浏览器”模板](#)（第 52 页）中所述。

在它的 main.js JavaScript 文件中，“实用工具”模板还演示了如何使用客户端方的数据库储存来储存内容，这些内容不管应用程序是否在线均可使用。模板包括设置简单键-值表格的代码，表格中保留有在前方显示的信息的值以及信息的字体、大小和颜色。

Dashcode 部件

Dashcode 包括许多独特的元素，您可以在您的 **Widget** 界面或 **Web** 应用程序界面上使用这些元素来显示信息。这些元素的其中一些（称为 **Dashcode 部件**）可以有规划地进行修改。

此附录中列出的某些部件专为配合 **Web** 应用程序使用而设计，不能用于 **Widget**（对它们的描述中有相关说明）。

如果某个部件没有在此附录中列出，则 **Dashcode** 专用的任何编程界面都不需要使用该部件。您可以使用“属性”检查器来更改它的属性，以及使用“行为”检查器来给它分配事件处理程序。有关“检查器”窗口的更多信息，请阅读[“更改元素的属性”](#)（第 32 页）。

活动指示器

“活动指示器”部件为配合 **Web** 应用程序使用而设计。活动指示器会显示出任务或进程正在处理，供用户查看，但不会指示出它将何时完成。

“活动指示器”部件包括 **JavaScript** 代码，此代码会启动和停止旋转动作。您可以使用 `startAnimation` 和 `stopAnimation` 方法来根据任务或进程的进度而调整活动指示器的旋转。

返回按钮

“返回按钮”部件为配合 **Web** 应用程序使用而设计。如果 **Web** 应用程序会显示多层信息，返回按钮提供给用户在应用程序内回溯他们的步骤的便捷方法。这使得应用程序可以在单个 **URL** 中表达它自己的浏览层次。

“返回按钮”部件自动包括在基于“浏览器”模板的项目代码中。它设置为在所有层次都出现（首层除外）。

浏览器

“浏览器”部件为配合 **Web** 应用程序使用而设计。浏览器提供了将元素成组以及向后和向前浏览的区域。当您选择“浏览器”**Web** 应用程序模板时，“浏览器”部件自动包括在内。

“浏览器”部件包括 **JavaScript** 代码以执行某些标准动作。特别地，浏览器支持一种方法，您可以使用此方法来显示信息层次中的下一层。典型情况为，您可以列表部件的控制器代码中调用 `goForward` 方法，将与前进箭头关联的 `onclick` 事件分配给它处理。“浏览器”部件会自动处理相似的“`go back`”函数。

画布

“画布”部件是一个自定的绘制区域，您可以将它添加到您的 **Widget**。Using the Canvas 讨论了如何使用 **JavaScript** 在画布上绘制。

呼叫按钮

“呼叫按钮”部件为配合 **Web** 应用程序使用而设计。呼叫按钮给用户提供了拨打电话的便捷方法。在“属性”检查器中，您可以指定当用户轻按按钮时拨出的默认电话号码。

“呼叫按钮”部件包括了 **JavaScript** 代码以执行某些标准动作，比如获得和设定电话号码的方法。如果按钮已启用，当用户轻按它时的默认动作为向当前设定的电话号码拨打电话。您可以使用 `setPhoneNumber` 方法来允许用户动态地更改电话号码。

栏布局

“栏布局”部件是您可以用于并排放置内容的区域。此部件对 **Web** 应用程序特别有用，因为每栏均可以设定为以绝对方式或相对方式放置内容。

例如，如果您想要提供一栏以显示固定大小的图像，以及另一栏以显示有关图像的可变大小的描述，您可以指定第一栏使用绝对方式放置，而第二栏使用相对方式放置。“栏布局”部件为您处理这种情况提供了极大的帮助。

边缘到边缘列表

“边缘到边缘列表”部件提供了列表格式，其中每行均从它的容器的一边伸展至另一边。边缘到边缘列表包含了可自定的行模板，用于在列表中创建新行。

您可以指定静态数据以在每行显示，或者您可以自定列表的控制器代码以在运行时动态地提供数据。您可以在“属性”检查器中设定这个。

动态接收数据的边缘到边缘列表需要控制器对象（静态的边缘到边缘列表不需要控制器对象）。列表控制器对象必须实现两个必需的数据源方法：`numberOfRows` 和 `prepareRow`。`numberOfRows` 回调方法会返回列表中的总行数。`prepareRow` 方法在每行被准备时均调用一次，并且它会使用传入值来填充行。特别地，`prepareRow` 方法要求以下三个参数：

- `rowElement`。这是整行的嵌入元素。您可以使用此元素为列表中的每行安装点按处理程序。
- `rowIndex`。这是行的基于零的索引。您可以使用此值在应用程序专用的数据结构中查找合适的行数据。
- `templateElements`。这是一本字典，包含模板行中拥有 **ID** 的每个元素的条目。字典中的每个键都是一个 **ID**，关联值则是 `rowElement` 内与原始模板元素相对应的元素，比如 `rowTitle`。您可以使用与这些元素关联的值填充至行中适当的位置。

边缘到边缘列表还包含了公共的 `rows` 属性，使您可以访问列表中的每个行元素。如果列表是动态的，您可以在列表填充同时使用 `rows` 属性查看 `prepareRow` 回调方法中当前已填充的行，但您将看不到所有行，直到列表被完全填充。对于静态和动态列表两者，列表被填充后，`rows` 属性对获得特定行的访问最有用。

如果边缘到边缘列表动态地接收数据，请务必调用 `reloadData` 函数以强制列表重新载入并显示更新的内容。（请注意，`reloadData` 在列表部件初始化时被自动调用。）要这样做，请将这行代码添加到您的列表控制器：

```
document.getElementById("myList").object.reloadData();
```

前进按钮

“前进按钮”部件为配合 **Web** 应用程序使用而设计。前进按钮提供给用户在 **Web** 应用程序的多层中向前移动的便捷方法。

量表

“量表”部件是一个带有指针的刻度盘，可以指示某个范围内的值。您可以使用“属性”检查器来编辑量表的外观和值的范围。如果您要将量表用作控制，请在“属性”检查器中选择“追踪鼠标”选项，然后在“行为”检查器中为 `onchange` 事件提供您自己的处理程序函数。只要量表的值一更改，就会调用您的处理程序。

如果您要使用量表以图形方式显示数据，则需要使用 **JavaScript** 来更新它的值。使用 `setValue` 方法更新量表的值，如此处所示：

```
document.getElementById("gauge").object.setValue(50);
```

请注意提供给 `setValue` 的值应当处于在“属性”检查器中为量表指定的范围之内。

指示器

“指示器”部件是一个指示灯，对不同的值会用不同的颜色表示。您可以使用“属性”检查器来编辑指示器的外观和值的范围。

使用指示器时，您需要使用 **JavaScript** 更新它的值，如此处所示：

```
document.getElementById("indicator").object.setValue(10);
```

请注意提供给 `setValue` 的值应当处于在“属性”检查器中为指示器指定的范围之内。

层次指示器

“水平层次指示器”和“垂直层次指示器”部件是线性指示器，可以显示某个范围内的值。您可以使用“属性”检查器来编辑层次指示器的外观和值的范围。如果您要将层次指示器用作控制，请在“属性”检查器中选择“追踪鼠标”选项，然后在“行为”检查器中为 `onchange` 事件提供您自己的处理程序函数。只要层次指示器的值一更改，就会调用您的处理程序。

如果您要使用层次指示器以图形方式显示数据，则需要使用 **JavaScript** 来更新它的值。使用 `setValue` 方法更新层次指示器的值，如此处所示：

```
document.getElementById("horizontalLevelIndicator").object.setValue(10)
```

请注意提供给 `setValue` 的值应当处于在“属性”检查器中为层次指示器指定的范围之内。

邮寄按钮

“邮寄按钮”部件为配合 **Web** 应用程序使用而设计。邮寄按钮给用户提供了发送电子邮件信息的便捷方法。在“属性”检查器中，您可以指定当用户轻按按钮时使用的默认收件人（电子邮件地址）和主题。

“邮寄按钮”部件包括了 **JavaScript** 代码以执行某些标准动作，比如获得和设定电子邮件地址以及获得和设定主题。如果按钮已启用，默认动作为打开邮件编写视图，其中“收件人”和“主题”栏已填充为当前设定的地址和主题。您可以使用 `setEmailAddress` 和 `setSubject` 方法来允许用户动态地更改电子邮件地址和主题。

地图按钮

“地图按钮”部件为配合 **Web** 应用程序使用而设计。地图按钮给用户提供了查看某个位置的地图的便捷方法。在“属性”检查器中，您可以指定当用户轻按按钮时显示的默认位置。

“地图按钮”部件包括了 **JavaScript** 代码以执行某些标准动作，比如获得和设定要显示的地址的方法。如果按钮已启用，默认动作为显示当前设定位置的地图。您可以使用 `setAddress` 方法来允许用户动态地更改地图位置。

Quartz Composer

Quartz Composer 部件是一个含有 Quartz Composer 合成效果的区域。当您的 **Widget** 正在运行时，如果要控制合成，请使用 Quartz Composer WebKit 插件的 **JavaScript** API。通过阅读《*Quartz Composer Web Kit Plug-in JavaScript Reference*》（Quartz Composer Web Kit 插件 **JavaScript** 参考），您可以了解有关 Quartz Composer WebKit 插件的 **JavaScript** API 的更多信息。

QuickTime

QuickTime 部件是一个用于回放 QuickTime 媒体的区域。使用 QuickTime JavaScript 方法来控制影片的回放，或更改其属性。要了解有关 QuickTime 插件的 JavaScript 方法的更多信息，请阅读《*JavaScript Scripting Guide for QuickTime*》（用于 QuickTime 的 JavaScript 脚本指南）。

圆角矩形列表

“圆角矩形列表”提供了由一群行组成的列表，这些行从容器的边缘插入，并且用圆角矩形作边界。圆角矩形列表部件包含了可自定的行模板，您可以使用它在列表中创建新行。

您可以指定静态数据以在每行显示，或者您可以自定列表的控制器代码以在运行时动态地提供数据。您可以在“属性”检查器中设定这个。

动态接收数据的圆角矩形列表需要控制器对象（静态的圆角矩形列表不需要控制器对象）。列表控制器对象必须实现两个必需的数据源方法：numberOfRows 和 prepareRow。numberOfRows 回调方法会返回列表中的总行数。prepareRow 方法在每行被准备时均调用一次，并且它会使用传入值来填充行。这些值通过以下三个参数传递给 prepareRow 方法：

- rowElement。这是整行的嵌入元素。您可以使用此元素为列表中的每行安装点按处理程序。
- rowIndex。这是行的基于零的索引。您可以使用此值在应用程序专用的数据结构中查找合适的行数据。
- templateElements。这是一本字典，包含模板行中拥有 ID 的每个元素的条目。字典中的每个键都是一个 ID，关联值则是 rowElement 内与原始模板元素相对应的元素，比如 rowTitle。您可以使用与这些元素关联的值填充至行中适当的位置。

圆角矩形列表还包含了公共的 rows 属性，使您可以访问列表中的每个行元素。如果列表是动态的，您可以在列表填充同时使用 rows 属性查看 prepareRow 回调方法中当前已填充的行，但您将看不到所有行，直到列表被完全填充。对于静态和动态列表两者，列表被填充后，rows 属性对获得特定行的访问最有用。

如果圆角矩形列表动态地接收数据，请务必调用 reloadData 函数以强制列表重新载入并显示更新的内容。（请注意，reloadData 在列表部件初始化时被自动调用。）要这样做，请将这行代码添加到您的列表控制器：

```
document.getElementById("myList").object.reloadData();
```

滚动区域

“滚动区域”部件是一个由滚动栏包围的区域，用于显示大于 Widget 界面或 Web 应用程序界面的内容。您可以使用“属性”检查器来自定滚动区域的外观。选项包括当滚动栏的内容适合它的边界时滚动栏是否自动隐藏，以及边界和边缘的尺寸。

要更改滚动区域的内容，请使用来自滚动区域的对象的内容属性，如下所示：

```
var content = document.getElementById("scrollArea").object.content;
```

```
content.innerText = someText;
```

一旦您获得了滚动区域的内容属性，则可以访问<div>滚动区域里面的元素。从那里，您可以使用 `innerText` 或 `innerHTML` 属性来更改滚动区域的内容。

堆栈布局

“堆栈布局”部件为配合 Web 应用程序使用而设计。堆栈布局是包含可以交换的视图的区域。

您可以使用“属性”检查器来添加或删除堆栈布局中的视图，并且您可以指定交换时使用的过渡样式。比如，您可以将视图间的过渡从“推移”（在此样式下新视图会从边缘推移旧视图）更改为“渐变”（在此样式下新视图会淡入至旧视图的顶部之上）。

堆栈布局方法

堆栈布局部件包括了许多方法，您可以使用这些方法来访问它的子视图并设定视图间的过渡。特别地，您可以使用：

- `getCurrentView`。此方法会返回当前活跃的视图。
- `getAllViews`。此方法会返回堆栈布局中视图的 **JavaScript** 数组。您可能想要使用数组索引获得“下一个”视图。
- `setCurrentView`。此方法含视图和布尔值，用于指定过渡是否应该以倒序方式执行（比如，推移过渡定义了倒序过渡）。此方法会将当前视图设定为传入视图，并且适当地设定过渡。
- `setCurrentViewWithTransition`。此方法与 `setCurrentView` 相似，不同之处在于它允许您给新视图额外传入一个专用过渡，而布尔值用于指定过渡是否为倒序。（请参阅“堆栈布局过渡”以了解有关过渡对象的更多信息。）

堆栈布局过渡

过渡对象会处理堆栈布局容器中视图间的过渡。默认情况下，过渡对象会处理少数过渡类型和许多过渡属性。过渡对象由 `Transition` 构造函数创建，如此处所示：

```
Transition (type, duration, timing)
```

`Transition` 函数的参数指定了过渡类型、过渡所花时长和过渡的加速曲线。请注意，当创建过渡对象后，您还可以在检查器中设定这些属性。

可用过渡类型有：

- `Transition.NONE_TYPE`。新视图简单地在旧视图的位置出现。
- `Transition.PUSH_TYPE`。二维过渡，其中新视图会将旧视图推出屏幕的一边。
- `Transition.DISSOLVE_TYPE`。二维过渡，其中旧视图会渐隐到新视图中。
- `Transition.SLIDE_TYPE`。二维过渡，其中新视图会从屏幕的一边滑入至旧视图之上。
- `Transition.FADE_TYPE`。二维过渡，其中新视图会淡入到旧视图之上。

- `Transition.FLIP_TYPE`。三维过渡，其中旧视图会翻转以显示新视图。
- `Transition.CUBE_TYPE`。三维过渡，其中旧视图和新视图看起来就像立方体相邻的两面，旧视图会旋转以显示新视图。
- `Transition.SWAP_TYPE`。三维视图，其中两个视图水平地互相远离，然后垂直地交换位置，使新视图留在顶部。
- `Transition.REVOLVE_TYPE`。三维过渡，其中旧视图和新视图共享一边上的一个轴，围绕该轴旧视图转出，同时新视图转入。

`duration` 参数应该包含过渡从开始到结束所花的秒钟数。最后，对于 `timing` 参数，您可以指定控制过渡的速度和加速度的定时函数（定时函数在 **WebKit CSS 动画规格** 中定义）。可用的过渡定时函数有：

- `Transition.EASE_TIMING`
- `Transition.LINEAR_TIMING`
- `Transition.EASE_IN_TIMING`
- `Transition.EASE_OUT_TIMING`
- `Transition.EASE_IN_OUT_TIMING`

定时函数决定了过渡的加速曲线。线性曲线意味着速度恒定（即无加速度），而“**ease**”意味着速度逐渐增加或减少。比如，`Transition.EASE_IN_TIMING` 函数意味着过渡开始时慢并且逐渐增快，而 `Transition.EASE_OUT_TIMING` 函数意味着过渡在接近结束时逐渐变慢。

您还可以为 `Transition.PUSH_TYPE` 和 `Transition.SLIDE_TYPE` 过渡指定方向。可用方向有：

- `Transition.RIGHT_TO_LEFT_DIRECTION`
- `Transition.LEFT_TO_RIGHT_DIRECTION`
- `Transition.TOP_TO_BOTTOM_DIRECTION`
- `Transition.BOTTOM_TO_TOP_DIRECTION`

最后，您可以为四个三维过渡指定方向。这些过渡的可用方向有：

- `Transition.RIGHT_TO_LEFT_DIRECTION`
- `Transition.LEFT_TO_RIGHT_DIRECTION`

过渡对象包括 `perform` 方法，用于执行过渡。您可以将以下参数传递给 `perform` 方法：

- `newView`。过渡后将可见的视图。
- `oldView`。现在（在过渡之前）可见的视图。
- `isReverse`。布尔值标志，用于指定过渡是否应该以倒序方式执行。（比如，推移过渡可以以倒序方式执行；交叉渐变过渡则以正序或倒序方式执行都一样。）

`perform` 方法会确保传入的视图共享共同的父容器元素，由于过渡在尺寸上受父容器限制，这点很重要。特别地，容器必须限制过渡以避免内容溢出。

请注意，您可以使用过渡对象为堆栈布局 `setCurrentViewWithTransition` 方法传入过渡参数。有关堆栈布局方法的更多信息，请参阅[“堆栈布局方法”](#)（第 60 页）。

文稿修订历史

此表描述对《*Dashcode 使用手册*》所做的修订。

日期	备注
Dashcode 2.0.1	已更新用于 Dashcode 2.0.1
Dashcode 2.0	添加了一个用于 Web 应用程序的新教程以及有关 Dashcode 2.0 中推出的新模板和部件的描述。
	新文稿描述了如何使用 Dashcode 创建 Dashboard Widget。

