

在许多情况下,应用程序设计为在分布式的服务器上执行,但是数据则保持集中,这通常是为了保护数据的完整性。不过,在某些情况下,数据也需要是分布式的,特别是当应用程序部署到一个具有分散或者昂贵的(或者同时是两者)网络基础设施因而带宽很小的地区时。有时,数据分布是通过分布式数据缓存实现的,但是这通常有破坏数据完整性的危险,并可能是成本很高和难于管理的。Lotus Domino 和 LEI 提供了在网络上分布所有类型的数据的理想解决方案,因为它们保证了数据的完整性、安全性和所有数据集实例的同步。用 LEI 分布非结构化的和关系数据提供了将 Lotus Domino 和 WebSphere Application Server 的优点结合在一起以满足应用程序需求的一个例子。

充分利用对 Domino 的投资

为 Domino 平台开发应用程序的一个最大好处是为 Notes Version 1 编写的应用程序可以不加改变地运行在 Domino 6 服务器上。这意味着您的公司仍然可以从超过 13 年以前的投资中实现价值,现在可以用 Web 浏览器、Java API 和 Web 服务访问这些应用程序。随着 Domino 的发展并成为 Lotus Workplace 的服务器平台之一,IBM 继续保护其客户的投资就变得很重要了。

上面一节(“应用程序平台考虑”)可能得出的结论是应用程序需要由 Lotus Notes 客户提供的丰富客户体验。IBM 计划继续支持和增强 Domino 作为 Lotus Workplace 的一个应用程序服务器。以下几节提供了关于在 WebSphere Application Server 或者 WebSphere Portal 上运行的应用程序中利用您的 Domino 投资的信息。

IBM Domino 与 WebSphere 集成

在 IBM 内部,Domino 和 WebSphere Application Server 小组彼此协作以开发和提供客户所要求的关键集成点。因为 WebSphere Portal 是基于 WebSphere Application Server 的,所以它也可从同样的集成点中

受益。所有这些元素都是根据完美的用户体验、容易管理和整体基础设施集成的需要发展而来的。同时 Java 和 J2EE 标准使得集成各种完全不同的应用程序成为可能, Lotus 和 WebSphere 都开发出了强大的、只存在于 Domino 和 WebSphere Application Server 之间的集成点。例如:

- 对于准备进行单一用户目录标准化的客户, Lotus Domino 可以为其他使用 LDAP 的应用程序提供 Domino Directory。这种协议正在迅速成为目录服务的事实标准,并且 WebSphere Application Server 也使用它进行用户身份验证。因为 Lotus Domino 可以作为一个 LDAP 服务器,所以 WebSphere 可以使用这种目录协议,使客户可以在一个位置上管理 Lotus Domino 和 WebSphere 的用户标识,使客户环境中的管理技能要求变得容易了。
- 当 WebSphere Application Server 和 Lotus Domino 使用同样的 LDAP 目录(不局限于 Domino LDAP 能力)时,客户可以配置 Lotus Domino 和 WebSphere 之间的单点登录(Single Sign-On SSO)。SSO 允许浏览器用户只输入他或者她的验证信息一次,并将这些信息储存在一个 WebSphere 和 Lotus Domino 共享的安全上下文中。在启用了 SSO 后,在 WebSphere 和 Domino 之间来回转换的用户就不需要像没有启用 SSO 时那样多次进行验证了。
- 此外, Domino Object for Java(Domino Java API)通过使用 SSO 令牌在 Lotus Domino 中创建会话使得在 Lotus Domino 中可以进行安全数据处理。作为当前 WebSphere Application Server 的 Web 应用程序用户,一位开发人员可以用 SSO 令牌在 Lotus Domino 中创建一个安全会话。这样在访问 Lotus Domino 时应用程序就不需要知道特定用户名和密码,同时将由任何给定应用程序所处理的用户密码的安全风险降到最低。

利用您的 Domino 数据

如果要在使用 WebSphere Application Server 还是 WebSphere Portal 处理表示及应用程序逻辑之间做出抉择，客户可以利用他们在 Domino 应用程序中的数据。根据 J2EE 应用程序的需要，上述工作可以使用 RAD 开发工具或 XML 及 Java 来完成，但后者可以提供更加精确的粒度控制。

WebSphere Portal 工具

WebSphere Portal 提供大量预装的 Lotus portlet，允许您将现有的 Domino 应用程序集成到 WebSphere Portal 环境中。这些 portlet 使得具备基本 WebSphere Portal 管理技能和由少到无 (little-to-no) 编程技能的人员能够快速而高效地配置 portlet。这些 portlet 可以显示 Mail、Calendar、To Do 视图以及 Domino 应用程序的视图中的数据。通过提供这些相互协作的 portlet，Domino 客户可以很容易地把最常见的 Domino 应用程序扩展到 WebSphere Portal 环境中。

通过在现有的 WebSphere Portal Application Integrator 框架中加入 Domino 功能，Portlet Builder for Domino 使得业务线 (line of business, LOB) 用户能够在 IBM Lotus Domino 的基础之上快速且毫不费力地构建出 portlet。这种向导风格的 portlet 使程序员不再需要书写底层函数，同时允许开发人员很容易地把主要 Domino 应用程序和数据引入到 WebSphere Portal 中。这种 portlet builder 还包括了 IBM Lotus Sametime 感知和 Click2Action 等用于在 portlet 间共享信息的功能，从而使门户用户可以即时获得生产力方面的收益。

WebSphere Business Integration Adapter for Domino

WebSphere Business Integration 应用程序提供支持依赖于异类系统数据集成复杂业务模型的能力。现在 WebSphere Business Integration Adapter for Domino 也提供连接到 Domino 的能力。Domino 适配器支持 Lotus Domino 和许多其他企业系统之间的双向、实时

的集成，前提是这些企业系统为包括诸如 Siebel 和 PeopleSoft 这样的系统在内的适配器组所支持。

Domino 特性使应用程序与 J2EE 集成变为可能

Domino 还具有使 Domino 数据与 J2EE 应用程序间的集成变得容易的特性。通过将 Domino 与 J2EE 应用程序，更重要的是与加入到在 J2EE 上构建的可扩展及事务性的应用程序中的人工交互结合使用，在开发应用程序时您可以受益良多。在集成 Lotus Domino 与基于 J2EE 的应用程序时，有两种主要途径可以访问 Lotus Domino:

- *Domino Object for Java*
- *Domino 6 客户 JSP 标签*

Domino Object for Java 实质上就是 Domino Java API。这些对象可用于对面向关键数据库过程的数据存储进行基于对象的访问，即 CRUD 行为。这种 API 不仅允许访问 Domino 数据，而且允许访问某些关键服务，比如注册用户、运行代理、发送邮件等等。

Domino 6 客户 JSP 标签仅用于 JSP 中。这些标签是内嵌在提供数据访问、数据输入和进程控制的 JSP 中的 XML 标签。JSP 标签对 Domino Objects for Java 进行了抽象，并为构建使用 Domino 数据及服务的 J2EE 应用程序提供了一个快速的开发运转周期。

面向 WebSphere Studio 的 Lotus Domino Toolkit

从 Domino 6.0.2 开始，面向 WebSphere Studio 的 Lotus Domino Toolkit 就为开发人员提供了一个直观工具，用于在他们的 WebSphere 应用程序中实现 Domino 客户 JSP 标签。该工具包是一组插件，用于增强 WebSphere Studio Application Developer 用户界面，并且使得 Domino 开发人员可以在不必了解 Java 的复杂性的情况下构建 J2EE 应用程序。

Studio UI 中加入了 Domino 视图，从而使您可以容易地使用诸如表单、字段、视图及代理这样的常见 Domino

对象。这些对象可以被拖放到正在开发的JSP上，然后等效的Domino客户JSP标签代码会被插入。该工具包的另一个好处在于它隐藏了使用Java把您的J2EE应用程序连接到Domino上的复杂性。

XML

Lotus Domino还拥有广泛的XML能力，您可以在J2EE应用程序中使用这些能力。一个J2EE应用程序可以使用到Domino的XML接口访问非机构化的Domino文档，而不需要了解与存储在Lotus Domino中数据有关的细节。然后，该J2EE程序可以对XML形式的结果进行处理，把它转换为应用程序所需要的特定形式。当应用程序准备更新Lotus Domino时，上述同一个XML接口可以被重用，目的是通过XML更新数据库，从而在与Lotus Domino之间来回传递数据。

Web Services

新兴的Web Services标准提供了一个标准化的访问协议，称之为简单对象访问协议(Simple Object Access Protocol, SOAP)，目的是简化应用程序集成。开发人员可以使用XML来表示应用程序接口。Lotus Domino可以作为使用LotusScript公开Domino数据及功能的Web服务的宿主。将诸如WebSphere这样的J2EE服务器、合适的SOAP类及Domino Java Object三者结合使用，开发人员可以把他们当前Domino应用程序中所期望的部分公开为Web服务。这样做要求精通一些Java编程技能，并对Domino对象模型有一定了解。IBM WebSphere Application Server提供了所需的SOAP类，而且WebSphere Studio开发环境为创建、使用、管理及部署Web服务提供了向导。这有助于提高开发人员的生产力，让开发人员可以集中精力把Domino应用程序的基本组成部分公开为Web服务。

DB2 上的 Domino

Domino 7 将支持DB2，将其作为与Notes Storage Facility并列的一种数据存储手段。这将把Domino的协

作应用程序能力和DB2关系数据库体系结构、可伸缩性及性能这二者的力量集合在一起。通过提供通过关系数据库接口对Domino数据的访问，利用您在Domino数据方面的投资将变得前所未有的容易。

通过 portlet 设计 Domino 应用程序

portlet是用于Lotus Workplace应用程序集合的组件模型。Domino应用程序可以针对通过portlet访问来进行开发。从开发人员的角度来讲，这类似于启用一个基于客户端的、通过Web浏览器访问的Notes应用程序。因为Lotus Workplace平台构建在WebSphere Portal框架之上，所以Lotus Workplace平台可以很容易集成原有Domino应用。

Domino Web应用程序有各种各样的设计元素，它们和portlet之间的匹配不是很充分，或者说它们没有遵从门户框架指导方针。其中包括帧和帧标签、JavaScript命名空间问题以及URL重定向的目标是Domino而不是Portal服务器。这些问题中的一些可以使用iFrame或Web裁剪portlet来解决。

通过增强Domino本身和针对使用Domino的特定情况而改进的portlet目标，IBM计划提高在门户环境中设计Domino Web应用程序的能力。2004年初，一个反向代理portlet计划被发布，该portlet允许URL只通过门户服务器进行重定向。这个portlet将解释Domino服务器发出的通用HTML结构，然后把它们转换为符合门户体系结构的结构。也可以添加用于处理不常见的HTML结构的补充规则。

因为面向门户环境开发Domino应用程序与开发Domino Web应用程序非常类似，所以增强用于开发和服务于Web应用程序的工具也在计划之中。计划增强的工具包括允许Domino应用程序开发适合门户框架的Domino Designer和Domino Web引擎。参见表1应用程序开发目标及可用选项。

表 1. 应用程序开发目标及可用选项

应用程序开发目标	<ul style="list-style-type: none"> • 可用选择
通过 Notes 客户端访问应用程序	<ul style="list-style-type: none"> • 使用 Domino Designer
通过 Web 浏览器访问应用程序	<ul style="list-style-type: none"> • 使用 Domino Designer 为 Domino Web 服务器开发应用程序 • 使用 Domino Designer 和前述 J2EE 部分中提到过的选择之一
通过 WebSphere Portal 访问应用程序	<ul style="list-style-type: none"> • 使用 Lotus portlet • 使用 Portlet Builder for Domino 开发 portlet • 使用即将到来的 portlet 表面化 WebSphere Portal 中的 Domino Web 应用程序
访问应用程序数据	<ul style="list-style-type: none"> • 使用 Domino XML 支持 • 使用 Domino Objects for java • 使用 WebSphere Business Integration adapter for Domino
通过 Web 服务访问应用程序	<ul style="list-style-type: none"> • 使用 Domino 作为使用 LotusScript 或 Java 书写的 Web 服务的宿主 • 使用 WebSphere 作为 Web 服务的宿主，并使用前述 J2EE 部分中提到过的选择之一
移动数据到关系数据存储中	<ul style="list-style-type: none"> • 使用 Domino 7 中即将出现的新特性，即允许 DB2 成为与 Notes Storage Facility 并列的一种数据存储手段 • 使用 Lotus Enterprise Integrator for Domino 把数据转换为关系数据存储

快速应用程序开发的未来

随着 Lotus Workplace 及其对按需应变电子商务(e-business on demand)与生俱来的支持，快速开发应用程序的能力将继续作为应用程序开发工具战略的焦点而存在。因为 Lotus Workplace 支持 WebSphere Portal、WebSphere Application Server 及 Lotus Domino 作为应用程序服务器，所以上述各个基本平台都需要提供 RAD 工具来支持 Lotus Workplace。

Domino Designer 得到证实的 RAD 能力使它可以很好地支持 Lotus Workplace 中的 Domino。Domino

Designer 是用于构建 Lotus Notes 及 Domino 应用程序的应用程序开发工具。承前所述，Domino Designer 是在未来将不断得到增强的工具之一，增强的功能特别集中在允许开发用于 Lotus Workplace 中的集成应用程序方面。这些计划包括在 Web 服务、利用 DB2 作为数据存储的特性以及为通过门户环境进行访问而定制的应用程序等各个方面的增强。没有计划将 Domino Designer 引入到 WebSphere Studio 或 Eclipse 中——它将作为 Lotus Notes 和 Domino 的专门开发工具而存在。

随着 IBM 把工作重点放在 J2EE 体系结构上，有一种需求越来越明显，那就是为应用程序开发团体(除 Java 技