

大 连 理 工 大 学 本 科 外 文 翻 译

跨媒体文档链接和导航

Cross-Media Document Linking and Navigation

学 部（院）： _____ 软件学院

专 业： _____ 软件工程（日语强化）

学 生 姓 名： _____ 吕宇

学 号： _____ 201593057

指 导 教 师： _____ 徐睿

完 成 日 期： _____ 2019 年 3 月 23 日

大连理工大学

Dalian University of Technology

跨媒体文档链接和导航

Ahmed A.O. Tayeh, Payam Ebrahimi, Beat Signer

网络与信息系统工程实验室, Vrije Universiteit Brussel, 布鲁塞尔, 比利时
摘要: (原文中如果无摘要, 此内容不写)

文档通常不是孤立存在的, 而是隐式或明确地链接到其他文档的各个部分。然而, 因为众多专有文档格式都有比较简单的连接模型, 所以在不同的文档格式的信息片段之间创建超链接的可能性是有限的。在之前的工作中, 我们提出了一种动态可扩展的跨文档链接服务, 克服了大多数现有文档格式支持的简单链接模型的局限性。基于插件机制, 我们的链接服务支持跨不同文档类型。在本文中, 我们通过集成一些文档格式以及第三方文档查看器来评估链接服务的可扩展性。我们说明了在这些文档格式和查看器之间创建高级超链接的灵活性, 这些超链接无法通过现有链接解决方案或现有文档格式的链接模型实现。我们进一步调查用户在创建和使用跨文档超链接时的体验。

CCS 概念

- 以人为本的计算→用户研究; 用户中心设计;
- 应用计算→超文本/超媒体创作; 文件管理;

关键词: 跨文档链接; 信息链接; 链接导航; 用户链接行为

1 介绍

HCI 研究调查了在阅读电子或者纸质文档时的用户行为, 证实了文档通常不是孤立使用的。根据 Adler 等人的观点, 跨文档引用任务构成了整个阅读和写作活动的重要组成部分。此外, Marshal 声称用户在阅读时会使用高亮或下划线作为锚点, 以此将文档中信息明确地关联(链接)起来。个人信息管理(PIM)领域的其他研究发现, 用户利用电子或者实体文件夹来组织和关联(关联)跨文档的信息。

带有关联性的超链接被认为是创建和管理文档以及其他信息对象之间的关系的基础。它们最初由 Bush 引入, Nelson 以它实现了非顺序阅读和写作。超链接对于成功的数字超媒体模型和系统(如 oN-Line System (NLS))具有开创性作用, 并通过启用内容的引用和注释, 进一步在万维网取得成功。

最新的数字文档格式支持简单形式的链接(即嵌入式和单向关联超链接), 允许用户跨不同文档关联信息。虽然许多文档格式提供了链接到第三方文档的可能性, 但通常无法定位到这些文档的哪一部分。例如, 开发人员可能会在 HTML 文档中创建针对整

个 PDF 或文本文档的超链接，但无法定位到哪一部分。许多解决方案，包括开放超媒体系统（例如，Sun 的链接服务或微观世界），注释系统（例如，MADCOW）或 XLink 标准 1，可以解决现有文档格式的链接模型的缺点。然而，如一些研究所述，大多数现有的链接解决方案有两个主要缺点。首先，它们仅支持跨预定义文档格式的链接，但这些文档格式必须在链接服务本身内查看和编辑。因此，用户必须离开其首选的第三方文档查看器和编辑器（例如 Microsoft Word），才能从链接服务提供的功能中获取到用户想要的信息。其次，如何扩展现有链接解决方案的体系结构以支持其他文档格式，这一问题的解决方案并不明朗。

图 1：动态可扩展的跨文档链接服务的概念模式（图略）

为了克服现有文档格式中链接模型的缺点，也使用户能够跨不同文档类型以链接到更具体的位置，我们提出了动态可扩展的跨文档链接服务。该链接服务使用户能够在文档内和文档之间创建外部双向和多向超链接。它可以通过插件进一步动态扩展，以支持现有和未来的文档格式。在本文中，我们通过集成各种文档格式和查看器（包括纯文本，PDF，XML，Microsoft Office 和 Google Chrome）来评估链接服务的可扩展性。此外，我们还讨论了集成其他文档格式和查看器的可能性。我们提供用户研究，调查用户体验，包括在创建和导航跨文档格式和查看器的双向和多向超链接的使用感受、以及扩展我们的链接服务以支持其他文档格式或查看器时的体验。

我们从第 2 节开始，简要介绍我们的链接服务。在第 3 节中阐述该链接服务提供的高级版超链接的详细信息。我们从第 2 节开始，简要介绍我们的链接服务。在第 3 节中阐述该链接服务提供的高级版超链接的详细信息。

2 跨文档链接服务

大多数现有文档格式提供的简单链接功能仅允许具有对源文档（例如文档的所有者）的编辑许可的用户创建新的超链接。例如，HTML 网页中的超链接始终必须由其开发人员编写。因此，没有写权限且想要跨文档关联信息的用户不能简单得基于文档的特征来创建超链接。大多数文档格式提供单向超链接。这也意味着链接文档（目标）及其读者不知道指向它（目标）的超链接来自于哪些文档（源）。正如前面提到的那样，该解决方案克服了其他服务在整合其他文档格式或查看器时，在不同文档格式的链接功能方面暴露出的缺点。例如，若想使用 Sun 的链接服务支持实现跨文档（在他们自身的第三方文档查看器可查看的文档）的链接，考虑到 Sun 的链接服务库，那么必须重写第三方文档查看器，以促进不同的文档查看器的通信。在早期的工作中，我们提出了一

种动态可扩展的跨文档链接服务，解决了现有文档链接模型和链接解决方案的缺点。图 1 说明了我们的链接服务的概念架构，它提供了一个插件架构来集成不同的文档格式以及第三方文档查看器。正如早期工作中所讨论的，在大多数设计决策中，我们考虑到我们的链接服务应该可由第三方开发人员扩展，以支持现有和新兴的文档格式。

我们的链接服务的链接模型基于 RSL 超媒体元模型[18]及其链接任意实体的想法，其中实体可以是资源，选择器或链接。资源是给定媒体类型的基本单元，例如图像，视频或完整文档。选择器始终与资源相关，并用于定位资源的具体部分。最后，链接可以是任意实体之间的一对一，一对多或多对多双向关联。我们的链接服务提供一种新的文档。它实现扩展 RSL 资源（文档）的数据插件。给定文档格式选择器也可添加到这种新文档中。例如，在纯文本文档格式的数据插件中，文本资源（文档）可以表示为指向文档的 URI，其选择器可以通过开始和结束索引表示引用范围。

我们的链接服务的可视化组件包含一个链接浏览器，用于可视化支持的文档格式。用户界面还提供必要的 GUI 动作以对超链接执行基本的创建，读取，更新和删除（CRUD）操作。在此可视化组件中，我们定义了可视化和与链接服务中的任何文档交互所需的必要功能列表，例如文档的打开，文档选择器的突出显示，选择器的创建和到超链接目标的导航。对于要在链接浏览器中显示的每种文档格式，必须实现可视化插件。给定文档格式的可视插件需要可视化文档及其选择器。此外，它必须提供创建，删除和更新选择器所必需的功能。

图 2：PDF 文档之间的双向超链接在链接浏览器中可视化，并在 Google Chrome 网络浏览器中显示可视化的 HTML 文档（图略）

所呈现的链接服务进一步解决了无缝集成第三方文档查看器的挑战，以使用户能够在能够链接不同文档的同时继续使用他们喜欢的第三方文档查看器。例如，图 2 显示了在我们的链接浏览器中可视化的 PDF 文档和在外部的 Web 浏览器中可视化的 HTML 文档之间创建的双向超链接。我们利用第三方文档查看器（即其 SDK 或 API）提供的可扩展性功能，以便将它们集成到我们的链接服务中。对于要与我们的链接服务集成的每个第三方文档查看器，都需要第三方文档查看器的加载项。该加载项应该提供一个用户界面，使用户能够创建，更新和删除选择器。该加载项应该提供一个用户界面，使用户能够创建，更新和删除选择器。此外，加载项需要通过我们的链接服务的可扩展通信模块提供的任一通信通道（例如，TCP 套接字，WebSockets 或 RESTful API）与我们的链接服务进行通信。加载项和我们的链接服务之间传递的消息以 JSON 格式表示。第三方文档查

看器的文档格式具有不同的逻辑结构以及选择器这一事实意味着链接服务和加载项之间交换的消息与链接服务和另一个加载项之间交换的消息不同。

为了应对这一挑战，我们决定设计调解器组件应构成链接服务的整合模块，以抽象化链接服务和第三方文档查看器之间的消息交换。这允许链接服务理解从任何加载项中得到的消息的类型和结构，并执行所需的操作。加载项也应够理解链接服务发送的任何消息的类型和结构，以便执行必要的任务。我们提出了网关组件以便于第三方文档查看器的集成。在与第三方文档查看器加载项交换的消息转换中所需的所有功能都已在网关组件中抽象化。对于我们的链接服务中支持的每个第三方文档查看器，必须引入扩展网关组件的网关插件，以便在其相应的文档查看器加载项和链接服务之间传递的消息转换。值得一提的是，在链接服务和加载项之间传递的每条消息必须包含一个命令键。不同预定义的请求值传达不同类型的交互（例如，创建选择器，导航到超链接目标或打开文档）。有关消息结构和请求值的更多详细信息，请参阅[22]。

由于 OSGi [7] 的动态可扩展性以及针对不同插件的管理和跟踪，我们的链接服务利用了 OS 的动态模块系统[7]。每个数据，可视或网关插件必须包含特定的键/值元数据，才能成为我们链接服务的有效扩展。跟踪组件利用插件元数据来识别它们，并在运行时正确地在链接服务中注入插件。此外，插件跟踪组件避免使用无效元数据的插件。最后但并非最不重要的是，我们建议使用在线存储库存储链接服务的不同插件和插件。最终用户应该能够通过从在线存储库安装所需的插件来按需支持不同的文档格式或第三方文档查看器。链接服务使用户能够通过在线存储库通信并显示要下载的可用插件来达到此目的。

图 3：用于编辑超链接源和目标的超链接概述菜单（图略）

根据用户请求，跟踪模块安装不同的插件从在线存储库下载它们。这个功能类似于 Eclipse2 IDE 的可扩展性。在下一节中，我们将详细说明最终用户如何使用我们的链接服务在文档中创建高级超链接，以便在链接浏览器或第三方文档查看器中可视化。

3 跨文档链接服务

我们通过使用链接服务提供了三种不同的方案来创建双向和多向超链接。我们通过使用链接服务提供了三种不同的方案来创建双向和多向超链接。在第一个场景中，我们解释了在链接浏览器中可视的文档之间创建超链接，例如如图 4 所示的超链接。在第二个场景中，我们解释了图 2 中描述的超链接的创建，其中一个文档在我们的链接浏览

器中可视，另一份文件在某第三方文档查看器中可视。在最后一个场景中，我们解释了两个文档（在不同的第三方文档查看器中可视）之间的超链接的创建。

图 4：左侧 PDF 文档与右侧 JPEG 图像之间的双向超链接（图略）

用户可以在我们的链接浏览器中可视的文档之间轻松创建双向和多向超链接。在链接浏览器中打开文档后，他们可以选择文档的某一部分并选择“添加选择器”选项，这是为了之后通过此服务支持的 CRUD 操作设置链接的目标区域。然后，链接浏览器允许用户选择（打开）另一个文档以创建目标选择器。用户可以通过按下“创建链接”按钮（即在链接浏览器中提供）来确认创建新的双向超链接，或者打开其他文档以创建其他选择器，从而创建多向超链接。用户始终可以在确认其创建之前，选择编辑超链接的源和目标。链接浏览器为用户提供有关创建的选择器的概述，使他们能够灵活地编辑超链接源和目标。例如，图 3 显示了用户定义的三种不同选择器：文本文档中的选择器（`text1.txt`），Word 文档中的选择器（`wordDoc.doc`）和 HTML 文档中的选择器（`webpage.html`）。用户可以轻松地将 Word 文档选择器从超链接源移动到超链接目标，从而产生具有一个源和两个目标的多向超链接。

与上一个场景类似，用户可以使用链接浏览器提供的“创建链接”按钮确认超链接的创建，或使用图 3 中所示的超链接概述菜单进行编辑和确认超链接的创建。可以通过几个简单的步骤创建图 2 中所示的超链接。假设用户同时使用链接浏览器中显示的 PDF 文档和 Google Chrome 网络浏览器中显示的 HTML 文档。用户从 PDF 文档中选择文本片段“ETH Zurich”，并从链接浏览器支持的 CRUD 操作中选择“创建选择器”选项。选定的 PDF 选择器将列在超链接概述菜单中的超链接源下。如图 5 所示，用户然后从 HTML 文档中选择 ETH Zurich，并从 Google Chrome 加载项支持的 GUI 操作中选择“创建选择器”的选项。加载项向链接服务发送 JSON 请求，其中包含有关文档和选择器的信息。链接服务要求相应的 HTML 网关翻译接收的消息。换句话说，HTML 网关应该将有关 HTML 文档的信息转换为表示 HTML 文档的 RSL 资源。此外，HTML 网关还必须将关于 HTML 选择器的信息转换为表示 HTML 选择器的 RSL 选择器。这意味着 HTML 网关与 HTML 数据插件进行通信，以便将接收到的消息的一部分转换为 HTML 资源或选择器。然后，链接服务将 HTML 选择器添加到正在构建的超链接中。HTML 选择器自动列在超链接概述菜单中的“超链接目标”下。保存超链接后，链接服务会要求 PDF 可视插件和 Google Chrome 加载项更新其文档，以便查看新的超链接。

与前一场景中描述的类似方式，用户还可以在不同文档查看器中可视化的文档之间创建超链接。假设除谷歌浏览器外，链接服务还支持 Adobe 的 Acrobat Reader。用户

想要在 Acrobat Reader 中可视的 PDF 文档和在 Google Chrome 中呈现的 HTML 文档之间创建超链接。他们选择 PDF 文档的一部分，然后从 Acrobat Reader 加载项支持的 GUI 操作中选择“创建选择器”的选项。然后，Acrobat Reader 加载项向链接服务发送请求，其中包含有关文档和选择器的信息，并要求 PDF 网关返回消息指定的 PDF 资源和选择器。因此，链接服务在超链接概述菜单中的“超链接源”下列出返回的选择器。然后，用户选择 HTML 文档的某些部分，并从 Google Chrome 加载项支持的 GUI 操作中选择“创建选择器”的选项。以类似的方式，HTML 网关从根据其相应加载项的消息返回 HTML 资源和选择器，HTML 选择器列在超链接概述菜单中的超链接目标下。最后，在两个文档之间创建超链接。

.....

致谢：致谢已略(见原文)

参考文献：参考文献已略(见原文)