

上海交通大学

工程硕士生开题报告登记表

学 号:	1130379118
姓 名:	胡济豪
导 师 I:	姚建国
单 位:	上海交通大学
导 师 II:	王圣浩
单 位:	中国科学院上海光学精密机械研究所
工程领域:	软件工程
专业方向:	嵌入式系统
班 级:	Z1303793
学 院:	电子信息与电气工程学院(软件学院)
开题日期:	2015 年 7 月 5 日

上海交通大学研究生院制

填表日期: 2015 年 7 月 5 日

学位论文（设计）题目	基于 Gstreamer 的 Chromium 音视频播放系统的设计与实现
课题来源及类型	自选

课题的意义及国内外现状分析:

车载多媒体终端作为一款嵌入式移动设备,被广泛搭在应用消费型轿车上。随着移动互联的大力发展,消费者已经不再满足于传统本地和外部存储介质(例如 U 盘)音视频文件播放,等等,更希望在车载机上随时随地享受移动互联网络音视频播放服务,例如打开网页直接播放音视频,或者打开音乐电台在线聆听美妙的乐曲。因而,随着移动互联网络的高速发展和 WIFI 的逐渐大面积覆盖,在车载多媒体终端上搭载浏览器就显得很有必要,这为客户打开了访问不计其数的网络资源的窗口。开源的 Chromium 浏览器(chrome 是 Chromium 的稳定版,不开源)以其精简的界面、快速的响应、流畅的体验和开源特性获得众多厂商的青睐。

当前,已经有很多浏览器成功扩展或移植到嵌入式平台的案例。较多的表现在互联网电视领域,例如基于 Widget 的 Yahoo ConnectedTV,增加了 Web 页面的扩展功能,能够在 Linux 下的电视一体机终端或者机顶盒上运行,同时传统电视以及互联网功能也均可以正常应用。基于 Android 的 GoogleTV 以 Android 系统和谷歌 Chrome 网络浏览器为基础,融合传统电视服务模式及丰富 Web 服务于身的综合娱乐体验系统。GoogleTV 相比 Yahoo ConnectedTV 的方案会略显复杂,除了涵盖软件系统外,还将外围设备与硬件整合在一起,以此丰富生态系统的完整性。

然而,鉴于多媒体系统功能复杂程度的提高,且不同的多媒体系统硬件平台和软件平台也存在着差异,这使得开发者必须为这些平台重新编写多媒体处理的代码。在开发应用程序时,开发人员也要花费大量的时间和精力来对这些底层的多媒体处理进行整合和使用。Gstreamer 框架是一种管道的设计思想,基于该架构可以任意开发需要增添的媒体格式、传输协议的流媒体数据处理应用。它提供的功能性元件可以有选择性的被安装到应用管道程序中是该架构最显著的优点,若采用 GStreamer 这个强大的通用多媒体框架,应用程序开发人员就可以将底层多媒体处理的整合和使用交给 GStreamer,从而将精力集中于现有的媒体上。除此之外,被封装成共享库的插件所提供的元件可以被多个应用程序共享使用,省去了一些播放器代码的重复编写。基于 GStreamer 架构的播放器不仅能给用户提供更多、更好的音视频享受,简化了集成和复用,快速响应市场需求,提高产品竞争力。

综上,在嵌入式车载多媒体设备上实现基于 GStreamer 框架的 Chromium 音视频播放在当今移动互联时代的工程应用上很有现实意义。

课题计划进度和预期成果：

2015 年 7 月-2015 年 8 月：Chromium 代码分析

2015 年 9 月-2015 年 10 月：Chromium 音视频相关接口抽象

2015 年 11 月-2016 年 2 月：Gstreamer 的封装和解码移植

2016 年 3 月-2016 年 4 月：Chromium 音视频接口与 Gstreamer 封装的接口适配和调试

2016 年 5 月-2016 年 7 月：论文整理

课题研究目标、研究内容和拟解决的关键问题：

课题研究目标：

基于 Gstreamer 架构的 Chromium 浏览器音视频文件播放的设计与实现。

研究内容：

- 1、Chromium 浏览器多媒体播放相关的架构设计与分析；
- 2、Gstreamer 架构的原理和使用方法；
- 3、Gstreamer 音视频播放的实现与接口封装；
- 4、Chromium 浏览器多媒体播放接口与 Gstreamer 封装接口的适配。

拟解决的关键问题：

- 1、Chromium 音视频播放相关架构分析与代码重构；
- 2、Gstreamer 音视频播放的实现；
- 3、Chromium 与 Gstreamer 的接口适配。

上海交通大学软件学院硕士论文开题报告

基于 Gstreamer 的 Chromium 音视频播放系统的设计与实现

The Design and Implementation of Chromium Audio and Video Player Based on Gstreamer

系 别	电子信息与电气工程学院(软件学院)
学科专业	软件工程
研究方向	嵌入式系统
姓 名	胡济豪
导 师	姚建国

上海交通大学软件学院

2015 年 7 月

课 题 来 源	自选
<p>一、研究背景</p> <p>如今网络已经应用于人们日常生活中的各个环节，成为了其重要的组成部分。然而随着计算机网络通信技术、多媒体技术等不断发展，静态的网页和简单的文字已经满足不了人们对网络信息的需求，人们更倾向于通过网络获取包含丰富多彩的音频和视频信息。为此，多媒体网络技术作为一种计算机综合处理动态图文信息的技术，凭借其很好的集成性、友好的交互性以及准确的实时性，将音频和视频的特性相结合^[1]，将人们带到了一个丰富多彩的网络世界，如今这种技术已经得到广泛的应用，大大的便利了人们的日常生活。</p> <p>当下，HTML5 技术、音视频技术都被主流浏览器支持，大多数的视频网站也都同时提供 Flash 和 HTML5 两种页面，特别是 Chrome 浏览器对于 HTML5 的支持^[2]，基于 Chrome 和 HTML5 的视频播放系统，可带来流畅的播控体验，并且浏览器性能消耗更低。这样的方法在嵌入式系统、移动设备中的互联网视频应用方案更能发挥其特有的优势。同时满足跨平台，高性能，体验友好的网站需求。</p> <p>车载多媒体终端作为一款嵌入式移动设备，被广泛搭在应用消费型轿车上^[3]。随着移动互联的大力发展，消费者已经不再满足于传统本地和外部存储介质（例如 U 盘）音视频文件播放，等等，更希望在车载机上随时随地样享受网络音视频播放服务。</p> <p>在车载多媒体终端上搭载浏览器就显得很有必要，开源的 Chromium 浏览器（chrome 是 Chromium 的稳定版，不开源）以其精简的界面、快速的响应、流畅的体验和开源特性获得众多厂商的青睐。</p> <p>然而，鉴于多媒体系统功能复杂程度的提高，且不同的多媒体系统硬件平台和软件平台也存在着差异，这使得开发者必须为这些平台重新编写多媒体处理的代码。在开发应用程序时，开发人员也要花费大量的时间和精力来对这些底层的多媒体处理进行整合和使用^[4]。</p> <p>Gstreamer 框架是一种管道的设计思想^[5]，基于该架构可以任意开发需要增添的媒体格式、传输协议的流媒体数据处理应用^[6]。它提供的功能性元件可以有选择性的被安装到应用管道程序中是该架构最显著的优点，若采用 GStreamer 这个强大的通用多媒体框架，应用程序开发人员就可以将底层多媒体处理的整合和使用交给 GStreamer，从而将精力集中于现有的媒体上，避免在如何处理后端、插件机制等问题上花费过多的时间。除此之外，被封装成共享库的插件所提供的元件可以被多个应用程序共享使用^[7]，省去了一些播放器代码的重复编写。基于 GStreamer 架构的播放器不仅能给用户提供更多、更好的音视频享受，简化了集成和复用，快速响应市场需求，提高产品竞争力。</p> <p>综上，在嵌入式车载多媒体设备上实现基于 GStreamer 框架的 Chromium 音视频播放在当今移动互联时代的工程应用上很有现实意义。</p>	

二、研究目标

本课题通过对嵌入式设备多媒体数据应用的现状和趋势的了解,结合车载多媒体终端软硬件开发平台技术,对常见构建流媒体应用的开源多媒体框架及 Chromium 浏览器音视频播放体系结构的相关技术进行学习研究。根据开发需求和底层操作的分析,设计并实现基于 Gstreamer 架构的 Chromium 音视频播放。

其主要内容分为以下几个部分:

1、Chromium 浏览器多媒体播放相关的架构分析与重构设计;

Chromium 是谷歌开源网页浏览器计划的代号。文中提及的 Chrome 就是在这个基础上剥离出来的简化版本。其核心思想基于高效、稳定、安全等理念,采用苹果公司的 Webkit 解析排版引擎,附上自主研发 V8 Javascript 解析引擎,并捆绑了有名的沙盒机制来实现安全访问。Chromium 最初只支持 Windows、Mac OSX 系统,从 2009 年开始支持 Linux 系统,功能愈加完善。由于开源项目的版本更迭频繁,截止目前,Chromium 的主版本号已经很多,副版本号更是每时每刻都在改变。

对于多媒体播放相关的模块,Chromium 本身提供了一个简单的实现方式,但是非常具有局限性,不能共通使用。在资源十分有限的嵌入式车载多媒体终端产品中,不可能为浏览器单独写相应的多媒体应用模块,一般来说,是多个应用共享一个音视频播放动态库,Chromium 也调用此共享库进行音视频的播放。因此需要对 Chromium 音视频播放相关的部分代码进行重构,添加中间层并抽象出比较通用的接口。这一部分的主要工作是:(1)分析 Chromium 音视频播放相关的代码结构及实现流程;(2)根据需求,分析和抽象出中间层用于共同接口的适配;(3)在原来的基础上重构优化现有的结构和代码。

2、Gstreamer 架构的原理和使用方法;

Gstreamer 是一个创建流媒体应用程序的框架,在 Linux 系统平台上,通过其管道的设计工作的理念和面向物件的思想,开发者可以设计一个简洁而高效的可扩展型的播放器。这种架构最显著的优点是其提供的功能元件可以有机的接入到各种应用管道程序中,从而降低了设计开发工作的复杂性。此外,这种架构是基于插件的,Gstreamer 将各种元件和一些基本的功能模块都以插件的形式封装成箱柜,开发者进行开发时可以引用这些功能模块设计特定的处理程序。几个比较重要的基础结构是:元件(Elements)、衬垫(Pads)、管道(pipelines)、箱柜(bin)。这一部分的主要内容是:研究并设计基于 Gstreamer 开发框架播放器的后台处理模块架构方案,阐述后台播放通道的创建过程,通道总线的使用,Gstreamer 架构对于事件的处理机制和方法,具体了解码箱柜的设计与缓冲区的处理、通道的处理流程等等。

3、Gstreamer 音视频播放的实现与接口封装;

Gstreamer 架构主要包括核心库和插件库,核心库提供了一套开发接口(API),可以利用这套接口构造特定功能的架构,这种特定功能架构可以利用插件模板封装成 Gstreamer 的插件。Gstreamer 架构就好像是一个“组装拼图”,核心库是架构的核心,而基于核心库扩展的功能就好像是一块块的积木,因此,我们可以一用这种模块结构“组装”出播放器的后台处理框架。这一部分的主要内容是:基于 Gstreamer 开发框架后台处理模块架构方案,编程完成后台模块的应用程式;将 Gstreamer 的实现用接口封装起来,做成一个共享库,可以被整个车载系统的任一应用程序加载调用,例如本地播放 USB 存储器中的音乐、视频。这些接口主要包括:播放,暂停,

停止，快进，快退，选时播放等等。

4、Chromium 浏览器多媒体播放接口与 Gstreamer 封装接口的适配。

在 1 和 3 中已经分别对接口进行抽象和封装，这一部分的主要内容就是二者接口的适配与调试，这里考虑采用适当的设计模式，使在使用上更加灵活。还要考虑线程与进程的问题，因为 Chromium 使用了较多的进程与线程。

三、关键技术

1、浏览器模块构成与工作原理

浏览器的模块主要分为网络、存储、2D&3D 图形、音视频、图片解码、HTML/CSS/JavaScript 解释器、渲染等等，其主要功能是将用户选择的 web 资源呈现出来。它需要从服务器请求资源，并将其显示在浏览器窗口中，资源的格式通常是 HTML，也包括 PDF、image 及其他格式。用户用 URI (Uniform Resource identifier 统一资源标识符) 来指定所请求资源的位置。首先是网页内容，输入到 HTML 解释器。HTML 解释器在解释后构成一棵 DOM 树，这期间如果遇到 JavaScript 代码则交给 JavaScript 引擎去处理；如果网页中包含 CSS，则交给 CSS 解释器去解释。当 DOM 建立的时候，渲染引擎接受来自 CSS 解释器的样式信息，构建一个新的内部绘图模型。该模型由布局模块计算模型内部各个元素的位置和大小信息，最后由绘图模块完成从该模型到图像的绘制^[8]，最终呈现在显示界面上。

2、Chromium 音视频播放实现相关的技术分析与代码重构

Chromiumd 的内核 WebKit 提供了支持多媒体规范的基础框架，如音视频元素、JavaScript 接口和视频播放等。根据 WebKit 的一般设计思想，它主要是提供标准的实现框架，而具体的实现由各个移植来完成，因为音视频需要平台的支持，不同的移植有不同的实现^[9]。

3、Gstreamer 的封装与解码器的运用

Gstreamer 采用了基于插件和管道的体系结构，框架中的所有的功能模块都被实现成可以插拔的组件，并且在需要的时候能够很方便地安装到任意一个管道上，由于所有插件都通过管道机制进行统一的数据交换，因此很容易利用已有的各种插件“组装”出一个功能完善的多媒体应用程序。GStreamer 为插件提供的框架是非常灵活，它足以满足大多数插件的需求。由于 GStreamer 本身已经提供了足够多的 demux 与 parse，他们足以解析目前市面上大部分码流，但是面对众多的音视频文件格式，如果想要有更多的支持，就需要开发者自己移植或编写解码模块。

接口的封装可以隐藏掉 Gstreamer 的具体实现，调用者无需考虑其内部的实现逻辑。在接口封装时要考虑共同性及特殊性，不仅满足 Chromium 接口适配的需要，也需要满足其他可能模块的需求。

4、性能改善

嵌入式设备硬件环境远不如 PC 强大，其性能也收到很大的限制。比如在 PC 上可以同时打开多个网页同时播放多个音视频，但是在资源有限的移动终端上往往不能满足上述需求。在嵌入式车载多媒体终端中同样需要对资源的使用加以管控，为了更好的客户体验，一般每次只打开一个网络音视频文件，当打开另一个时会把前一个视频资源释放掉，表现在界面上是暂停状态。其二，出于内存空间及 CPU 占用率的考虑，在播放视频的时候一般不采用视频帧 buffer 的重复拷贝，多采用内存共享。

为更好的提升性能，本课题中将采用在网页上挖洞的方法，将视频的描画界面至于网页 **Layer** 的下方，在网页 **Layer** 上挖取一个透明的窗口，透视到下方的视频播放。另外还有通过设置代理的方法加快音视频流的网络加载速率，等等。

四、可行性分析

1. 当今浏览器的发展已经比较成熟，更有开源的 **Chromium** 可以探索内部机理和实现方式。相关论坛社区也比较成熟，有大量的爱好者致力于浏览器的开发。
2. **Gstreamer** 技术在很多嵌入式设备中被采用，传统本地音视频文件的播放也多有采用，将其嫁接到 **Chromium** 下没有技术鸿沟
3. 在浏览器中播放网络音视频和传统的嵌入式设备本地播放器播放音视频对硬件（网卡除外）的需求差异不大，在性能上二者没有巨大差异。

五、课题创新

将 Chromium 移植到嵌入式设备的方案比较常见，某些文献中也提及到 Chromium 在嵌入式设备上的性能提升方案，但对 Chromium 音视频播放模块的改写及和其他模块共用 Gstreamer 共享库的方案尚未形成系统完整的成果。本课题计划尝试实现 Chromium 框架下的音视频播放的 Gstreamer 改写。

六、研究进度

2015 年 7 月-2015 年 8 月：Chromium 代码分析

2015 年 9 月-2015 年 10 月：Chromium 音视频相关接口抽象

2015 年 11 月-2016 年 2 月：Gstreamer 的封装和解码移植

2016 年 3 月-2016 年 4 月：Chromium 音视频接口与 Gstreamer 封装的接口对接和调试

2016 年 5 月-2016 年 7 月：论文整理

七、参考文献

[1] 阮胜才. 视频点播系统的设计与实现[M]. 安徽：中国宇航出版社，2008：634-635.

[2] 高睿鹏，刘佳玲. 基于 FFMPEG 的通用视频插件[D]. 内蒙古：内蒙古工业大学，2010，27(1)：2474-2476.

[3] 赛迪顾问半导体咨询事业部. 2005 年中国汽车电子市场规模达 624 亿元. 世界电子元器件，2006 年第 3 期.

[4] 李淑荣. 基于 Gstreamer 的网络媒体播放系统的开发[D]. 中国海洋大学：中国海洋大学，2012

[5] wikipedia. <http://en.wikipedia.org/wiki/GStreamer>[J/OL]. last modified on 15 May 2015

[6] Richard John Boulton, Erik Walthinsen, Steve Baker, Leif Johnson, Ronald S. Bultje, Ronald S. Bultje, Tim-Philipp Müller, Wim Taymans. GStreamer PluginWriter's Guide (1.5.0.1) [J/OL]. <http://gstreamer.freedesktop.org/data/doc/gstreamer/head/pwg/html/index.html>. 2015 年 5 月 21

[7] wikipedia. http://en.wikipedia.org/wiki/Dynamic-link_library[J/OL]. last modified on 16 May 2015

[8] Tali Garsiel. <http://kb.cnblogs.com/page/129756/>[J/OL]. 9 Feb 2012

[9] <http://www.chromium.org/developers/design-documents>[J/OL]. 21 May 2015

上海交通大学软件学院硕士论文开题综述

基于 Gstreamer 的 Chromium 音视频播放系统的设计与实现

The Design and Implementation of Chromium Audio and Video Player Based on Gstreamer

系 别	电子信息与电气工程学院(软件学院)
学科专业	软件工程
研究方向	嵌入式系统
姓 名	胡济豪
导 师	姚建国

上海交通大学软件学院
2015 年 7 月 5 日

浏览器是指可以显示网页服务器或者本地文件系统的 HTML 文件内容，并让用户与这些文件交互的一种软件^[1]。互联网的革命浪潮带动了众多技术的快速发展，网络浏览器作为互联网的最重要的终端接入口之一在短短的二十多年时间里日新月异。经过近二十年的发展，现在的浏览器已经有几十种之多。市场占有率在前面几位的主要有 IE、Firefox、Google Chrome、Safari、Opera 等^[2]。特别是在进入 21 世纪后，越来越多的功能被加入到浏览器中来。在 W3C 等标准组织的积极推动下逐步形成 HTML5 技术，更成为了浏览器发展的火箭推进器。

一般说来，浏览器都是基于浏览器排版引擎开发的，很少有人或公司从头开始编写浏览器。目前常用的浏览器引擎有如下几种^[3,4]：

Trident，是 Microsoft Windows 操作系统的浏览器 IE（Internet Explorer）的排版引擎的名称。

Gecko，一套开源的、以 C++编写的网页排版引擎。Mozilla Firefox 浏览器就是以它作为浏览器引擎的。Gecko 是跨平台的，可以运行于 Microsoft Windows、Linux 和 Mac OS X 等操作系统之上。

KHTML，Konqueror 浏览器所使用的引擎。该浏览器引擎用 C++语言编写，它的优点是速度快，缺点是对错误语法的容忍度低。

WebKit，开源浏览器引擎，由 KDE 的 KHTML 修改而来。现在使用它作为引擎的浏览器有 Google Chrome、Midori、Safari、Maxthon3 等，还有其他一些浏览器也计划转到 WebKit 引擎上来。

Presto，由 Opera Software 开发的浏览器排版引擎。使用它的浏览器有 Opera、任天堂 DS、Wii 浏览器等。

Tasmam，微软推出的 IE for Mac 所使用的排版引擎。

2008 年，Google 公司以苹果开源项目 WebKit 作为内核，创建了一个新的项目 Chromium，Chromium 是谷歌开源网页浏览器计划的代号^[5]。该项目的目标是创建一个快速的、支持众多操作系统的浏览器，包括对桌面系统和移动操作系统的支持。Chromium 使用了同 Safari 一样的浏览器内核。在 Chromium 的基础上，Google 发布了自己的浏览器产品 Chrome。不同于 WebKit 之于 Safari 浏览器，Chromium 本身就是一个浏览器，而不是 Chrome 浏览器的内核，Chrome 浏览器一般会选择 Chromium 的稳定版本作为它的基础。Chromium 是开源试验场，它会尝试很多创新的并且大胆的技术，当这些技术稳定之后，Chrome 才会把它们集成进来。也就是说 Chrome 的版本会落后于 Chromium；其次，Chrome 还会加入一些私有的编码解码器以支持音视频等；再次，Chrome 还会整合 Google 众多的网络服务^[6]。自推出以，以其清爽精炼的用户界面(User Interface, UI)、快速的渲染和脚本执行效率迅速壮大成为最流行的浏览器^[7]。

目前，嵌入式系统面临一个巨大的发展机遇，后 PC 时代对嵌入式系统技术也提出了更广泛的需求^[8]。嵌入式设备也凭着精湛的开发技术具备了许多原来所没有的功能，PC 机上的许多应用也在嵌入式设备上得到了成功的移植和充分的应用^[9]。在嵌入式系统技术快速发展的同时无线

网络技术也日益成熟，以往单调的媒体服务也不再满足人们的生活需求。在线直播、网络视频等网络媒体服务随之发展起来，这也对嵌入式设备对多媒体信息处理的能力上也提出了更高的要求。

随着中国三大运营商不断推出各种移动网络业务，在一定程度促进了移动网络的大规模普及，使得移动多媒体业务有了更广阔的市场和前景^[10]。在现阶段，作为一个移动智能设备，多媒体应用是一个必不可少的功能之一。多媒体的播放，包括本地文件和网络的流媒体，都需要强大的内置多媒体播放引擎的支持，这是多媒体功能满足用户体验的重要保证。

各种嵌入式设备上都有有一些不同的浏览器和播放器等多媒体处理软件。现在媒体文件的格式种类非常多，每种格式的编解码都有不同的实现代码，从而使得嵌入式设备上的解码库也变得非常复杂^[11]。并且这些解码库的安装也需要许多的技术支持，同时也会受到一些限制。因此，在嵌入式设备上开发播放器就会变得比较繁琐，如果针对每一种音视频文件都编写相应的解码程序就会变得非常困难也完全没有必要。

而 Gstreamer 就为简化 Linux 下多媒体的开发提出了一种比较简洁的框架结构^[12]。开发者可以利用这种框架结构设计一系列媒体处理模块来进行音频、视频处理。虽然 Gstreamer 的开发人员和开发资料比较少，具有一定的开发难度，但是相信这种多媒体框架以其简洁的开发机制和模块化的开发结构一定会成为 Linux 环境下主要的多媒体开发工具。

Gstreamer 是流行于 Linux 系统下的多媒体应用程序开发框架，其采用了基于元件和管道的体系结构，框架中的所有的功能模块都被实现成可以插拔的组件，并且在需要的时候能够很方便地安装到任意一个管道上，由于所有插件都通过管道机制进行统一的数据交换，因此很容易利用各种插件组装出一个功能完善的多媒体应用程序^[13]。

Gstreamer 框架是一种管道的设计思想，基于该架构可以任意开发需要增添的媒体格式、传输协议的流媒体数据处理应用。它提供的功能性元件可以有选择性的被安装到应用管道程序中是该架构最显著的优点，若采用 GStreamer 这个强大的通用多媒体框架，应用程序开发人员就可以将底层多媒体处理的整合和使用交给 GStreamer，从而将精力集中于现有的媒体上，避免在如何处理后端、插件机制等问题上花费过多的时间。除此之外，被封装成共享库的插件所提供的元件可以被多个应用程序共享使用，省去了一些播放器代码的重复编写。基于 GStreamer 架构的播放器不仅能给用户提供更多、更好的音视频享受，简化了集成和复用，快速响应市场需求，提高产品竞争力。

因此，在将 Chromium 浏览器移植到车载多媒体终端的过程中，应用 Gstreamer 来实现浏览器音视频的播放是一个可行，而且节省成本的做法。而且在技术层面，可以做到二者巧妙的融合。

大体上讲，浏览器的功能包括网络、资源管理、网页浏览、多页面管理、插件和扩展、书签管理、历史记录管理、设置管理、下载管理、账户和同步、安全机制、隐私管理、外观主题、开发工具等^[14]。下面对他们之中的一些重要功能做简要介绍^[15-19]。

网络：它是第一步，浏览器通过网络模块来下载各种各样的资源，例如 HTML 文本、JavaScript

代码、样式表、图片、音视频文件等。网络部分其实非常重要，因为它耗时比较长而且需要安全访问互联网上的资源。

资源管理：从网络下载或者本地获取资源，并将他们管理起来，这需要高效的管理机制。例如如何避免重复下载资源、缓存资源等，都是他们需要解决的问题。

网页浏览：这是浏览器的核心也是最基本、最重要的功能。它通过网络下载资源并从资源管理器获得资源，将他们转变成为可视化的结果。

多页面管理：很多浏览器支持多页面浏览，所以需要支持多个网页同时加载，这让浏览器变得更为复杂。同时，如何解决多页面的相互影响和安全等问题也非常重要。为此，一些浏览器做了大量的工作，例如可能使用多线程或是进程来绘制网页。

插件和扩展：这是现代浏览器的一个重要的特征，他们不仅让你能过显示网页，而且能支持各种形式的插件和扩展。插件是用来显示网页特定内容的，而扩展这是增加浏览器新功能的软件或压缩包。

账户和同步：将浏览器的相关信息，例如历史浏览记录、书签等信息同步到服务器，给用户一个多系统下的统一体验，这对用户非常友好，是浏览器易用性的一个显著标识。

安全机制：本质是提供一个安全的浏览器环境，避免用户信息被各种非法工具窃取和破坏。这可能包括显示用户访问的网络是否安全、为网站设置安全级别、防止浏览器被恶意代码攻破等。

开发者工具：这对普通用户来说用处不大，但对网页开发者来说意义却非比寻常。一个优秀的开发者工具可以帮助审查 HTML 元素、调试 JavaScript 代码、改善网页性能等。

说到浏览器对多媒体的支持，不得不提的就是 Flash 插件和 HTML5 之争。Flash 对 Web 的发展起了非常重要的作用，它能够支持视频、音频、动画等多媒体功能，虽然现在大家都在讨论 Web 前端领域是否应该丢弃 Flash 插件转而支持 HTML5^[20,21]。

对 Chromium 来说，一个完整的多媒体解决方案，通常需要 WebKit 和 Chromium 两个项目共同努力，一起解决相关问题，从总体上来说，大致有四个部分^[22]。

第一部分当然是 WebKit 的基础部分，包括对规范的支持，这其中有 DOM 树、RenderObject 树和 RenderLayer 树等多媒体方面的支持。

第二部分是 Chromium 的桥接部分，也就是将 WebKit 的接口桥接到 Chromium 项目中来，包括接口的桥接、渲染方面的桥接等。

第三部分是依赖的其他多媒体库，包括 ffmpeg、libjingle 等第三方项目，使用它们来做多媒体的处理。

第四部分是 Chromium 对多媒体资源获取和使用多媒体库来帮助解码等管线化过程的具体实现。Chromium 重新实现了整个媒体播放流程，并对桌面系统和移动系统采用了一些特殊的技术和解决方法。

上面介绍了浏览器架构的主要组成部分,并对主要部分的实现功能做了简单介绍。浏览器的各个模块的协同工作,使得浏览器完成了网页数据的请求、响应信息的解析、页面的布局以及显

示。在浏览器工作时存在几个主要的进程,即 HTTP 连接认证进程,网络交互进程,html 解析和显示进程,几个进程相互协作完成浏览器的整体功能。浏览器经初始化后启动,创建一个主窗口进程,对键入的 URL 信息进行解析,解析正确则创建网络进程。网络进程通过 HTTP 访问模型连接到 WEB 服务器,浏览器对响应文件解析,根据缓存预取原理对 cache 进行判断。需要从服务器请求页面则建立与服务器的 socket 连接,获取网页数据,并将页面缓存进 cache 中。浏览器对请求的页面进行数据类型判断,不同类型的数据调用不同命令进行解析,解析后的数据送入窗口布局及显示模块,最终在浏览器窗口中显示请求页面。这些主要的数据操作被封装成浏览器的主要进程,整合成浏览器的工作结构^[23,24]。

Gstreamer 是一个创建流媒体应用程序的框架,其基本的设计思想来源于俄勒冈(Oregon)研究生学院有关视频管道的创意,同时也借鉴了 Directshow 的设计思想^[25]。Gstreamer 最典型的应用就是视频播放器的构建,Gstreamer 的这种管道的设计思想和架构是开发任意类型的流媒体处理程序成为可能。这种架构最显著的优点是其提供的功能元件可以有机的接入到各种应用管道程序中,从而降低了设计开发工作的复杂性。此外,这种架构是基于插件的,Gstreamer 将各种元件和一些基本的功能模块都以插件的形式封装成箱柜,开发者进行开发时可以引用这些功能模块设计特定的处理程序。在这种开发架构中,主要有以下几个重要的基础结构:

1. 元件(Elements)

在 Gstreamer 开发框架结构中,元件是最基本的结构,也是 Gstreamer 中最重要的概念。元件类型主要可以分为以下三种^[26,27]:

- (1) 数据源元件(source elements):数据源元件只有输出端,主要是产生供管道消费的数据,比如音频捕获单元负责从声卡上读取音频数据而不对数据做任何处理,只是把捕获的音频数据传递给其他的单元。
- (2) 过滤器元件(Filter elements):过滤器元件既有输入端又有输出端,数据从输入端进入,经过某种处理后从输出端输出并传递给其他元件。过滤器元件的形式非常灵活,可以有多个输入输出端。在 Gstreamer 处理数据时,一般是将音/视频数据流分开进行处理,这样的数据分离器元件就有一个输入端两个输出端。
- (3) 接收器元件(Sink elements):接收器元件只有输入端,一般是 Gstreamer 处理的终端。典型的接收器元件是音/视频回放单元,元件负责将接收到的数据写到声卡/显卡上来实现音/视频的显示播放。

Gstreamer 框架是由基本的元件、元件的连接和组装、元件间的数据传输来实现各种复杂的多媒体处理架构的。不同的元件具有不同的功能,并且具有不同的创建模板,Gstreamer 核心库默认包含了具有媒体数据基本处理功能的元件,我们可以利用这些处理元件编写应用程序来实现设计所需的功能。当然,开发者也可以动手编写的新的元件来扩展 Gstreamer 核心库。

2. 衬垫(Pads)

上面提到了元件的概念,要实现元件的结构化就需要将元件进行连接,这种连接正是通过衬垫

(Pads)来实现的。元件之间存在这种连接,数据才会在元件之间进行流动和传递。不同的元件具有不同的衬垫,并不是任意两个元件的衬垫都可以连接。衬垫就好像是元件的插头,只有当两个元件的衬垫所处理的数据类型相同的时候才可以实现连接。衬垫也会有源衬垫(sourcepads)和接收衬垫(sinkpads)。数据源元件可以有一个或多个源衬垫,数据源元件通过这些衬垫来向外发送数据。同样,元件接收数据也可以通过一个或是多个接收衬垫(sinkpads)来完成,每一对数据源元件和接收元件之间只有一对衬垫连接。在 Gstreamer 框架中衬垫都是依附于元件存在的,衬垫从某种程度上可以描述元件所具有的能力^[28]。

3. 管道(pipelines)和箱柜(bins)^[29-31]

在我们用 Gstreamer 开发多媒体处理框架时,往往结构会比较复杂,只用元件和衬垫进行一些连接是非常麻烦和难以组织完全的,这样 Gstreamer 提出了箱柜(bins)的概念。简单的说箱柜就像是容纳元件的容器,可以操作被封装在内所有元件来实现特定功能。同时,箱柜也可以看作是一个元件,可以对箱柜进行像元件那样简单的连接操作,因此箱柜也可以被另一个更大的箱柜所容纳。这样,开发者可以通过箱柜连接来开发较为复杂和功能更强大的 Gstreamer 处理架构。箱柜是否存在输入和输出是由箱柜设置的元件类型所决定,如果箱柜是一个过滤器元件,这个箱柜会同时存在输入端和输出端。在利用箱柜开发应用程序时,开发者可以对箱柜进行整体调用,把箱柜设置为普通元件与其它处理结构连接。开发者也可以将箱柜结构进行改进,根据需要插入或是删除处理元件。箱柜通过 Gstreamer 总线消息(bus)实现对其内部元件的控制和监听。管道(pipelines)是箱柜的一个特殊子类型,是一种高级的箱柜。箱柜给设计者提供了一种整体化的处理思想,开发者可以只关注箱柜的输入、输出和实现的功能,而不去关心箱柜内部是如何连接的,从而简化了应用程序架构的设计开发。

结合以上关于浏览器的简介、Chromium 多媒体解决方案的描述、Gstreamer 的概述、以及 Gstreamer 基本的设计理念和原理,可以看出使用 Gstreamer 架构实现 Chromium 浏览器音视频的播放具有比较完善的理论基础。

参考文献

- [1] Wikipedia, Web Brwoser[EB/OL]. http://en.wikipedia.org/wiki/Web_browser
- [2] Tim Berners-Lee. WorldWideWeb, the first Web client[EB/OL/].<http://www.w3.org/People/Berners-Lee/WorldWideWeb.html>, 2015-6-28.
- [3] 赵经纬,周余,王自强等.基于WebKit的嵌入式浏览器的研究与实现.电子测量技术,200934(3),135-137
- [4] 叶卿.嵌入式微浏览器的设计与实现:[D],北京:北京邮电大学.2005,22-25
- [5] 西村贤.Chromium开发者[EB/OL/].<http://www.atmarkit.co.jp/news/200903/30/chrome.html>, 2009-3-30.
- [6] 朱永盛.WebKit技术内幕[M].北京:电子工业出版社,2014:2-4
- [7] 肖梦华.面向智能电视的嵌入式浏览器平台的研究与设计[D].上海:复旦大学.2013
- [8] 许晔.发展嵌入式系统是我国后PC时代的战略选择[J].中国科技投资,2010,2:2-3
- [9] 梁融凌,余昌嵩. 基于嵌入式Linux电子书的设计与实现[J].牡丹江师范学院学报(自然科学版), 2014,4:13-14
- [10] 冯礼哲,高声荣,罗怀琴,李逸.发展中的4G网络[J].科技视界,2015,5:23
- [11] 罗锐.基于Blackfin平台的嵌入式播放器的研究与实现[D].成都:电子科技大学.2009
- [12] 仇昊.基于GStreamer的嵌入式多媒体系统的研究与实现[D].北京:北京邮电大学.2011
- [13] Taymans W, Baker S, Wingo A, etal.Gstreamer Application Development Manual(0.10.35.1)[EB/OL].[2011-07-15].<http://gstreamer.freedesktop.org/data/doc/gstreamer/head/manual/manual.pdf>
- [14] 张爱娟.基于Linux的嵌入式浏览器的设计与实现[D].福建: 福州大学.2010
- [15] Yamakami,T.A micro—component architecture approach for next generation embedded browsers. IEEE embedded Software and Systems,2007,23(12):233-240
- [16] Yao Qiong, Sun peng. Effective Iframe—Based Strategy for Processing Dynamic Data in Embedded Browser. IEEE Intelligent Control and Automation,2008,24(8):538—542
- [17] Lei Zongyan LingXiang.Architecture and design of the Parser used in the embedded browser based on DFA.IEEE Intelligent Control and Automation, 2008,23(11):6627-663
- [18] Alan Grosskurth. Concrete Architecture of Mozilla. Ali Echihabi.2004.10
- [19] 于渤.基于linux的嵌入式浏览器研究[D].重庆: 重庆大学计算机学院, 2006.
- [20] 邓晓,何中灵.浅析HTML5与Flash的发展趋势[J].电子制作,2015,1:47
- [21] 韩舶.简述HTML5与FLASH的关系与发展[J].中国广播,2011,6:16-18
- [22] 蔡智聪.对Chrome的源码简易剖析[J]. 电脑编程技巧与维护,2011,11:43-51
- [23] 吴锐强. 嵌入式浏览器网页解析器的研究与实现[D]. 成都:电子科技大学.2011
- [24] L Millett, B Friedman, E Felten.Cookies and Web browser Design.Design&Test of Computers.2006
- [25] 肖文鹏.用Gstreamer简化Linux多媒体开发[EB/OL]. <http://www.ibm.com/developerworks/cn/li>

nux/l-gstreamer/,2004.6.01

[26] Y.H.Li. Gstreamer based home media center implementation on intel oloriver. 2007.

[27] Gstreamer插件开发指南(0.10.9.1)[DB/OL]. <http://wenku.baidu.com/view/7baed6d5b9f3f90f76c61ba9.html>

[28] 王蕊, 刘卫东, 王金童. 基于GStreamer的媒体播放研究[J]. 电子设计工程, 2012, 20(3): 28-30

[29] Prateek Bansal, Brian Jeff, Isara Indra. Building Gstreamer. TEXAS INSTRUMENTS Application Report SPRAAQ9 . 2008

[30] Richard John Boulton, Erik Walthinsen. GStreamer Plugin Writer's Guide (0.10.32). 2000

[31] 廖勇[等]编著. 流媒体技术入门与提高[M]. 国防工业出版社, 2006