

ICS 33.160

M 70/79

团 体 标 准

T/CVIA-30-2014 代替

CVIA/ZNDSO2-2014

智能电视操作系统技术规范

2014-01-10 发布

2014-01-10 实施

中国电子视像行业协会

发布

目 次

前言	III
1 概述	1
2 定义和缩略语	1
2.1 定义	1
2.2 词汇, 缩略语	1
3 系统描述	2
3.1 系统结构	2
3.2 框架层	3
3.3 应用层	3
3.4 硬件需求	4
3.5 扩展	5
4 内核	5
5 硬件适配层 (HAL) 和 板级支持包 (BSP)	5
5.1 输入设备	5
5.2 传感器、摄像头和音频模块	5
5.3 显示	5
5.4 以太网	6
5.5 WIFI	6
5.6 VOLD	6
6 多媒体	6
6.1 多媒体播放	6
6.2 多媒体扫描	6
6.3 字幕显示	6
7 TV Services	7
7.1 硬件组件需求	8
7.2 资源 URI	10
7.3 TV Manager 接口	10
8 操作系统要求	10
8.1 硬件配置需求	10
8.2 系统框架要求	10
8.3 系统接口要求	11
9 图形	11
9.1 2D 图形	11

9.2	3D 图形.....	11
10	Connectivity.....	12
10.1	Ethernet.....	12
10.2	PPPoE.....	12
10.3	Wifi.....	12
10.4	Bluetooth.....	12
10.5	DLNA /UPnP.....	12
10.6	Samba.....	12
10.7	闪联.....	13
10.8	DiiVA.....	13
11	浏览器（可选）.....	13
11.1	浏览器功能要求.....	13
11.2	浏览器协议要求.....	13
11.3	浏览器界面要求[可选].....	14
11.4	WebCore 扩展.....	14
12	DRM 和安全.....	15
12.1	DRM.....	15
12.2	安全和权限.....	15
13	系统升级与维护.....	16
13.1	系统架构.....	16
13.2	系统升级操作方式.....	17
13.3	系统升级范畴.....	17
13.4	NAND 分区要求.....	17
13.5	其它功能需求.....	18

前 言

本规范为推进智能电视产业健康发展，降低各厂商的重复建设，增强智能电视产品的兼容性，根据智能电视的产业化现状而制定，是智能电视标准体系的一部分。

本规范提出一个可实现中国智能电视终端的系统环境，规定了智能电视操作系统要求，用于指导智能电视操作系统的研发及部署。

智能电视操作系统提供一个完整的，基于开放标准的软件框架。它可以优化智能终端设备，如智能电视，是专为实施互联网连接智能多媒体终端而设计的。它允许用户在看电视的同时，运行多个应用程序和服务。

通过标准的智能终端操作系统，应用程序开发者可以创建和提供具有创新性、创造性的应用，并能将这些应用快速地投入市场，降低了开发成本。

本规范由中国电子视像行业协会提出并归口。

本规范起草单位：中国电子视像行业协会、TCL集团股份有限公司、四川长虹电器股份有限公司、青岛海信电器股份有限公司、深圳创维-RGB电子有限公司、康佳集团股份有限公司、青岛海尔电子有限公司、广州欢网科技有限责任公司、深圳市中彩联科技有限公司等单位共同起草。（排名不分先后）

本规范主要起草人：白为民、郝亚斌、冯晓曦、刘秀云、余潘虎、纪萍、严帅领、王志国、李晓榕、李微、郭建伟、李大勤、姜钧凯、杨海波、董波、邓文科、张新法、田卫平、吴伟、张曼华（排名不分先后）

智能电视操作系统技术规范

1 概述

智能电视操作系统阐述了一个为智能电视软件、硬件提供有效资源分配的系统环境。智能电视操作系统允许芯片供应商、厂商、独立软件开发和服务提供商实施和提供自己的解决方案来满足该文档中所描述的要求。

本规范描述了该操作系统各模块的需求和设计规范，包括：内核，硬件适配层（HAL）和板级支持包（BSP）组件，多媒体框架和扩展，电视（TV）服务框架，图形设施，连接组件，网络浏览器和扩展，系统应用，TV门户服务接口，DRM和安全，配置文件以及工具等。

本规范定义和描述智能电视操作系统的要求，作为智能电视操作系统的通用描述，包括功能描述以及接口调用。

智能电视操作系统的描述会根据未来的新需求的增加而不断地进行更新。

2 定义和缩略语

2.1 定义

智能电视操作系统是智能电视机的关键组成部分，管理智能电视机硬件资源，控制其他程序运行并为用户提供交互操作界面的系统软件集合，主要任务是负责管理与配置内存、决定系统资源供需的优先次序、控制输入与输出设备、操作网络与管理文件系统等基本任务。

2.2 词汇，缩略语

ALSA	Advanced Linux Sound Architecture	高级 Linux 声音架构
A/V	Audio/Video	音频/视频
BSP	Board Support Package	板级支持包
CA	Conditional Access	条件存取
CAM	Conditional Access Module	条件存取模块
DRM	Digital Rights Management	数字版权管理
DSM-CC	Digital Storage Media Command and Control	数字存储媒体命令和控制

DVB	Digital Video Broadcasting	数字电视广播
EPG	Electronic Program Guide	电子节目菜单
HAL	Hardware Abstraction Layer	硬件抽象层
HDCP	High-bandwidth Digital Content Protection	高带宽数码内容保护
HDMI	High Definition Multimedia Interface	高清晰多媒体接口
HDTV	High Definition Television	高分辨率电视
HTTP	Hyper Text Transfer Protocol	超文本传输协议
IP	Internet Protocol	网络互联协议
MPEG	Moving Pictures Expert Group	动态图像专家组
OpenGL ES	Open Graphics Library for Embedded System	嵌入式3D图像加速子集
OpenSL ES	Open Sound Library for Embedded System	嵌入式音频加速标准
OpenVG	Open Vector Graphics.	矢量图像算法标准
PID	Packet Identifier	标志码传输包
PPP	Point-to-Point Protocol	点对点协议
PPPoE	Point-to-Point Protocol over Ethernet	以太网的点对点协议
PSI	Program Specific Information	节目特定信息
RF	Radio Frequency	射频
SI	Service Information	信息服务
TVOS	Smart TV Operating System	智能电视操作系统
TCP	Transmission Control Protocol	传输控制协议
URI	Universal Resource Identifier	统一资源标志
URL	Universal Resource Locator	统一资源定位
VM	Virtual Machine	虚拟机

3 系统描述

本规范描述了智能电视操作系统结构，包含电视模块，智能电视管理模块，系统框架，厂家扩展，Linux内核和硬件层。其中系统框架结构包含智能电视应用层，智能电视框架层，智能电视适配层，具体功能做如下描述。

3.1 系统结构

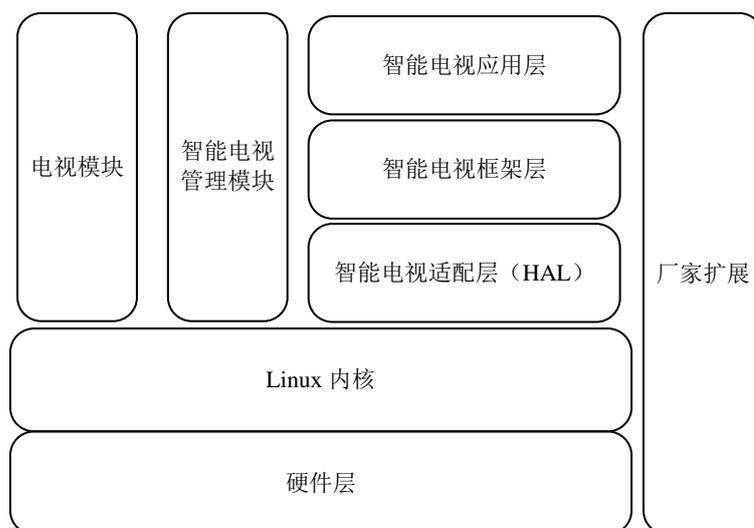


图1： 智能电视操作系统的系统结构

图1显示了系统的体系结构。硬件位于底层，Linux内核和硬件抽象层置于硬件的上层。在硬件抽象层的顶部是智能电视操作系统框架，它为应用层提供相应的接口。应用程序开发者可以使用智能电视操作系统框架提供的API来开发应用。在图的左侧，电视模块和智能电视管理模块位于Linux内核的上层。

智能电视操作系统框架为运行在应用层框架的应用程序提供了系统库。TV框架的细节将会在3.3节进行详细描述。当用户需要重启应用或者重启系统时，智能电视管理模块提供了几种不同的重启方式来重启系统的不同层次。为了能在智能电视操作系统中以不同层次重启系统，智能电视管理模块脱离于TV框架层之外。智能电视管理模块的内部细节将在3.2中详细说明。

为了保持系统的稳定，智能电视管理模块应位于TV框架层的外面，智能电视管理模块也可以独立地重启智能电视操作系统框架。当重启TV框架或者应用程序时，电视模块程序将会被启动并展示在屏幕上，ATV程序是通过电视模块以及它自己的UI系统来实现的，该UI系统使用了linux的frame buffer。同样的原因，ATV模块也应该独立于TV框架层之外。当TV框架重启时，ATV UI是否被启动是可选的。

智能电视操作系统框架需要被移植各种不同的硬件平台，这需要硬件抽象层（HAL）完全支持所有的硬件设备。

为了保证各个制造商不同硬件接口的差异性，图1展示了制造商的扩展层。

3.2 框架层

图2展示了TV框架层高层结构图。TV框架层是由应用程序框架层和各个库组成的。图2显示了应用程序框架层部分和库层部分；其中，应用层仅是示例。每个部分的详细信息将会在下面的章节中详细描述。

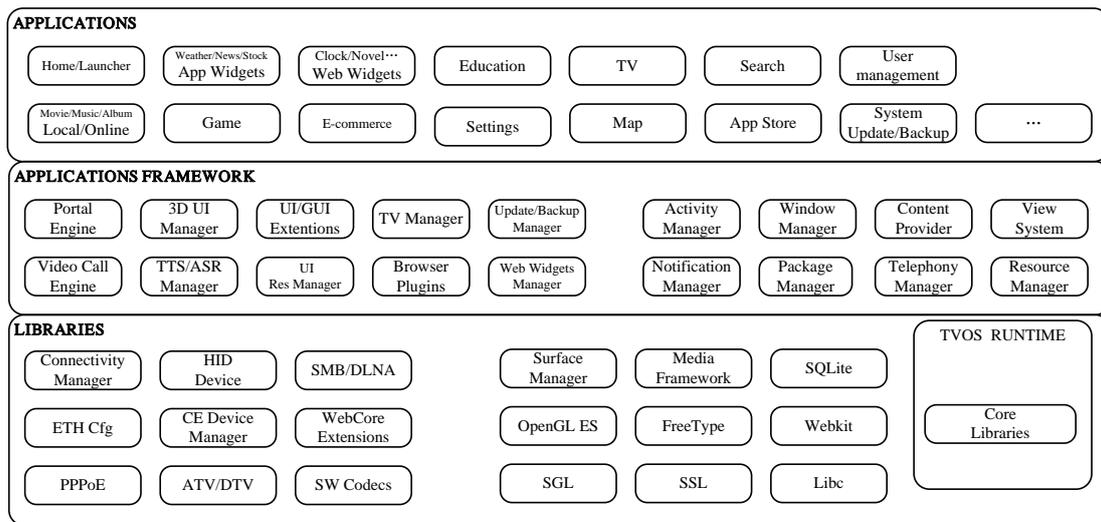


图2： TV框架的结构

3.3 应用层

应用程序层位于 TV 框架的最顶端，如图 3 所示。

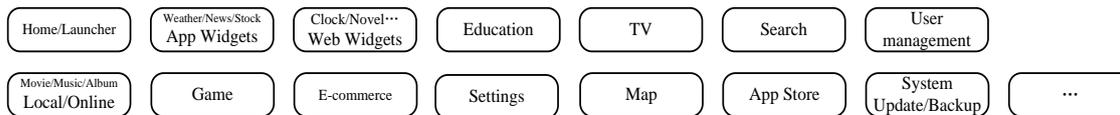


图 3： TV 框架应用层

3.4 硬件需求

下面的列表显示了智能电视操作系统所依赖的硬件需求。基本需求显示了运行智能电视操作系统所需要的最基本的硬件部分，而可选需求显示了智能电视操作系统的可选组件。

3.4.1 基本需求：

3.4.1.1 CPU：主频不低于 1GHz

3.4.1.2 内存：不低于 512MB

3.4.1.3 闪存：不低于 512MB

3.4.1.4 显示: 720P 及以上

3.4.1.5 音视频解码：采用国家数字电视相关标准

3.4.1.6 OpenGL ES 1.x and 2.0

3.4.1.7 硬件接口：

——Di iVA 输入

——调谐器

——以太网

——USB Host (EHCI, OHCI and/or OTG)

——遥控器：IR, or RF

3.4.2 可选需求：

3.4.2.1 OpenVG

3.4.2.2 显示：3D

3.4.2.3 音视频解码：RMVB, AVS, H264, MPEG2, MPEG4, VC1, AC3, MP3 等。

3.4.2.4 外设：

——触摸板或其他输入设备

——蓝牙

——热拔插 USB 存储设备

——热拔插 USB 摄像头

——SD 卡

——智能卡、CAM (PCMCIA CA Module)

——麦克风

——感应器

——WiFi (B/G/N)

——HDMI 输入

3.5 扩展

为了保证不同制造商硬件接口的多样性，图1在图的右侧显示了对各个层的扩展。为了通过厂商维护扩展接口的一致性，每一层的扩展接口都会被定义。

4 内核

智能电视操作系统的核心采用适用于TV Framework的Linux kernel，智能电视操作系统的核心系统服务如内存管理、进程管理、网络协议框架、系统安全、文件系统等基础设施均来自于Linux kernel，相关设备驱动程序也来自Linux kernel。

在智能电视操作系统的开发过程中，当新版本智能电视操作系统需要新的kernel特性支持时，内核必须相应升级到较新版本以保证智能电视操作系统 Framework的功能正常。另一方面也可以进行kernel的单独升级(不升级TV framework)，以支持更多的设备、或用于改善子系统性能等目的。

5 硬件适配层 (HAL) 和 板级支持包 (BSP)

5.1 输入设备

智能电视操作系统支持的输入设备包括键盘，鼠标，触摸屏等。智能电视操作系统使用标准Linux输入设备及驱动模型。

5.2 传感器、摄像头和音频模块

传感器、摄像头和音频模块的硬件抽象层实现都需要通过继承和实现智能电视操作系统中的HardwareInterface类来完成。

5.3 显示

DisplayHardware提供了显示部分的硬件抽象层接口，接口的具体实现可以使用gralloc、copybit或者overlay。对于2D图形加速，可以通过copybit模块实现，OpenGL硬件加速实现的3D图像处理作为一个可选项。

5.4 以太网

在智能电视操作系统中，以太网的HAL层在libnetutils中实现，并且由它来提供上层需要的接口。

5.5 WIFI

智能电视操作系统使用wpa_supplicant作为WiFi设备的平台接口。WiFi驱动兼容标准Wireless-Extensions。Wireless-Extensions提供了Wpa_supplicant与kernel空间wifi协议栈的接口，wpa_supplicant通过socket与wifi适配层通信。Wifi适配层作为一个桥梁连接wifi上层与wpa_suplicant。

5.6 VOLD

VOLD在智能电视操作系统中作为卷管理守护进程，它被用来侦测，挂载和监视外部存储器的状态。

6 多媒体

6.1 多媒体播放

智能电视操作系统多媒体框架支持各种常见类型的媒体播放，多媒体应用开发者可以利用该框架轻松的整合视频、音频和图像到应用中。智能电视操作系统多媒体框架提供了一个可以播放更多媒体格式的多媒体库，可以支持本地媒体播放和流媒体播放。智能电视操作系统多媒体播放器应具备支持解码升级功能。

6.2 多媒体扫描

对外部存储设备可以进行扫描，搜集系统显示所需的图片、媒体文件、文档等。当设备是第一次扫描时，全部扫描该设备内容并建立对应的媒体数据库，下次挂载该媒体存储设备时，多媒体扫描程序会查找以前建立的多媒体数据库。能够根据智能电视系统支持的多媒体格式，把扫描到且系统支持的多媒体文件列表显示到界面供操作者选择播放。

6.3 字幕显示

在播放多媒体文件的过程中，多媒体播放系统会解析多媒体文件字幕的格式，并存储在指定的内存

中，根据字幕控制命令，优先显示内嵌字幕，其次是外挂字幕，并且能够根据所处区域，自动选择字幕语言。

7 TV Services

本节定义了一个能同时支持模拟电视和数字电视平台，并能适用于任何供应商的智能电视操作系统标准。TV Services涵盖包括地面、有线和卫星等各种传输媒介在内的多种数字视频广播标准。



图 4： 智能电视操作系统 TV service 架构图

TV service架构如图4所示。电视管理API为电视服务应用开发接口。电视制造商与芯片供应商可以根据自己的内部实现细节定义电视服务库并通过其适配层来实现电视管理API接口。允许第三方开发者获取的功能，通过API提供；平台或设备的具体应用由厂商自定义扩展。

TV service架构组件描述如下：

a) 硬件适配层（HAL）

HAL 是一组与硬件驱动交互的 API 接口。这部分提供了一个底层的与硬件相关的软件部分的抽象，并便于智能电视操作系统中的电视服务库使用。TV service 的 HAL 层应该能够支持所有中国区电视服务所要求的硬件组件。

b) 电视服务库（TV Service Library）

电视服务库应该包含TV Manager接口所需的数字电视及模拟电视功能的所有实现细节。

c) 适配层（Adaptation layer）

适配层是一个能使通用的JNI函数连接到不同供应商的TV Service Libraries的抽象接口。

d) JAVA 本地调用

JAVA本地调用的API与TV Manager的API一一对应匹配。

e) 电视管理模块 (TV Manager)

TV Manager定义电视应用层所需的API。第三方服务供应商或应用程序开发人员可以被允许使用TV Manager接口去开发他们的应用。因此，TV Manager接口应该能够满足不同应用开发者的所有需求。

f) 应用层 (Application)

任何使用智能电视操作系统 TV service的应用，包括电视的应用或者其他任何应用都应该在这一层运用TV Manager API进行实现。

7.1 硬件组件需求

TV service应该为平台的各种硬件设备提供相关的功能函数。相关设备必须在HAL层得到支持并且在电视服务库为各种设备定义阐明需求。为了确保执行的独立性和灵活性，只有TV Service中必须执行的功能需要在这里描述。

TV service应该在系统中为所有的硬件设备提供相应的功能函数，但硬件平台的功能将通过硬件配置进行限制。

7.1.1 输入信源

TV service应该能够支持多种A/V输入，电视应用层应该能够通过TV Manager管理这些输入源。为了能够支持这些输入源，这些功能应该在硬件抽象层和电视服务库中得以实现，且接口应提供给TV manager。

7.1.1.1 为 TV service 媒体播放器选择 A/V 输入。

7.1.1.2 对 A/V 输入信源格式信息的动态侦测。

7.1.1.3 从输入源直接输出到输出端的路径控制。

7.1.2 输出

TV service 应该能够支持多种 A/V 输出，电视应用层应该能够通过 TV manager 的 API 管理这些输出。为了能够支持这些输出，这些功能应该在硬件抽象层和电视服务库中得以实现，且接口应提供给 TV manager。

7.1.2.1 为 TV service 媒体播放器选择输出。

7.1.2.2 对 A/V 输出的格式信息的动态侦测。

7.1.2.3 输出端的盗版保护处理。

7.1.3 Tuner(调谐器)

TV service 应该能够支持某种格式的广播调谐器，且电视应用层应该能够通过 TV manager 管理这些调谐器。为了支持 tuner 的输入，这些功能应该在硬件适配层和电视服务库中被实现。

7.1.3.1 为 TV service 媒体播放器提供数据输入通道。

7.1.3.2 为频道管理提供支持(调谐)频点切换功能。

7.1.3.3 支持信号强度，信噪比等参数的获取。

7.1.4 频道管理

TV service 应支持对输入射频广播信号的频道管理机制，且电视应用能够通过频道管理管理频道列表和进行频道选择。

7.1.4.1 扫描给定的调谐器输入通道，并设置回调函数通知扫描结果。

7.1.4.2 选择频道或节目进行播放。

7.1.4.3 检索节目信息。

7.1.5 Demux

TV Service 应支持解复用模块，通过 PID 过滤和时钟控制等提供传输流复用和解码。频道管理将与解复用相互作用来获取 PSI 和 SI 列表信息。电视应用层将与频道管理相互作用来控制 Demux 的设定，使得 Demux 与应用层没有直接的接口。

下列功能应在硬件抽象层和电视服务库中实现。

7.1.5.1 DEMUX 需支持多种数据源接入。

7.1.5.2 支持 Filter 的 PID 和掩码设置。

7.1.5.3 过滤器支持 SECTION,PES,RawTS 等多种过滤模式。

7.1.5.4 支持 SI/PSI Section Filter 的数据回调、事件。

7.1.5.5 支持解扰用 ECM 和 EMM Filter 的 PID 设置。

7.1.6 数据通道

TV service 应支持反向通道通信，且电视应用层能控制这些数据通道。

下列功能在硬件抽象层和电视服务库中实现。

7.1.6.1 通过 Tuner 和 Demux 提供基于 DSM-CC 的数据通道。

7.1.6.2 提供原始 VBI 数据通道。

7.1.7 显示控制和图像质量控制

显示和图像质量控制是指管理平台的显示设备的基本控制和增强功能。电视应用层应对这些设置项进行控制。

下列功能在硬件抽象层和电视服务库中实现。

7.1.7.1 设置显示控制，如对比度、亮度、锐利度等。

7.1.7.2 设置图像质量控制，如 MEMC,FRC,颜色增强等。

7.1.7.3 设置显示设备设置项，如位置、大小、相位、背光等。

7.1.7.4 如果支持 3D 播放，设置显示设备的 3D 设置项。

7.1.8 声音控制

声音控制是指电视设备的声音设置和功能管理,且电视应用层通过硬件抽象层和电视服务库的实现来控制这些功能。

7.1.8.1 设置声音控制,如音量、静音、平衡、EQ等。

7.1.8.2 设置声音相关特性。

7.2 资源 URI

智能电视操作系统使用 URI 命名规则来识别和操作资源。

7.3 TV Manager 接口

本版本对 TV Manager 接口不做统一要求。

8 操作系统要求

8.1 硬件配置需求

见 3.4 硬件需求。

8.2 系统框架要求

8.2.1 基于 Linux 内核的操作系统,初始为 Linux kernel 2.6,随着系统的升级要求 Linux Kernel

也不断升级

8.2.2 独立于智能电视操作系统框架的智能电视操作系统 Manager 来实现不同方式的重置

8.2.3 独立于智能电视操作系统框架的 ATV 子系统

8.2.4 能提供丰富的库和应用程序接口的 STA 框架

8.2.5 支持各种常见媒体类型的播放,有自己的播放器并支持厂商播放器

8.2.6 能同时支持模拟电视和数字电视，并能适用于任何供应商的 TV Service

8.2.7 提供开发工具 SDK，NDK 以及其他工具

8.2.8 支持多种可选的网络连接和访问方式，如：Samba,Ethernet,PPPoE,Wifi,Bluetooth,
DLNA / UPnP 以及闪联等

8.2.9 支持智能电视操作系统新功能的 Web 浏览器

8.2.10 提供显示系统信息和管理系统资源功能的系统应用

8.2.11 具备访问应用程序服务器接口，方便应用程序的升级和维护

8.2.12 提供可扩展的，支持多种方案的 DRM 框架，来使应用程序可根据内容的相关许可来管理受保护的内容

8.2.13 拥有统一的的安全的应用程序管理机制，包括权限、签名和认证等

8.2.14 提供系统升级、系统备份、系统恢复及恢复出厂设置等，内核与系统可同时升级也可单独升级

8.2.15 提供选择配置功能

8.3 系统接口要求

8.3.1 开发者可以使用 Java 作为编程语言来开发应用程序

8.3.2 可以通过 NDK 使用 C/C++作为编程语言来开发应用程序

8.3.3 HAL(硬件抽象层)让 framework 的开发能在不考虑驱动程序实现的前提下进行发展

8.3.4 灵活的可以实现所有硬件特性的 API 接口

8.3.5 兼容 Android API

8.3.6 扩展功能提供相应的 API 接口操作

8.3.7 相应的 API 开发文档

9 图形

智能电视操作系统图像采用了自定义 2D 图像库,同时智能电视操作系统架构为高性能 3D 图形提供支持。

9.1 2D 图形

提供两种绘制 2D 图形的方法:

9.1.1 用 layout 绘制图形或是动画到一个 View 对象。

9.1.2 直接绘制图形到一个 Canvas。

9.2 3D 图形

智能电视操作系统使用 OpenGL 特别是 OpenGL ES API 来支持高性能 2D 和 3D 图形。在 Framework 层的两个基本类, GLSurfaceView 和 GLSurfaceView.Renderer 可以使用 OpenGL ES API 来创建和操作图形。

视频图形可以使用纹理映射显示在 3D 图形引擎上。智能电视操作系统采用 OpenGL ES 2.0 在 3D 图形平面显示所有尺寸的视频。可以使用 push and pull 模型传送视频帧到 3D 图形引擎。智能电视操作系统中需要实现 pull 模型来支持在 3D 图形平面显示视频。

10 Connectivity

本章的连接标准是可选的,但至少需要一种以上的协议来和外部设备通讯。

本章主要介绍智能电视操作系统框架如何通过 API 提供连接功能，包括 Connectivity Manager 和 Connectivity Service. Connectivity 子系统提供以下功能：

- a) 监测网络连接状况
- b) 当网络状态变化时发送广播通知
- c) 当前链接断开时, 连接其他网络
- d) 提供应用程序接口, 允许应用程序查询设置网络状态

10.1 Ethernet

Ethernet 子系统在框架层实现了 Ethernet 功能，并且对上层应用提供 API 接口。

10.2 PPPoE

PPPoE 服务向应用层提供 API 接口使其能够通过 PPP 协议连接互联网。

10.3 Wifi

WiFi 服务经由框架层提供接口，驱动及 HAL 需要支持标准 wpa_supplicant。

10.4 Bluetooth

Bluetooth 服务经由框架层提供接口，框架层对于原实现无改动。

10.5 DLNA /UPnP

DLNA 是数字生活网络联盟 (Digital Living Net Alliance)。为了实现家中的数字设备之间的互操作性，以现有的技术标准为基础建立了一套通用的 DLNA 家庭联网设备互操作性指南。该指南规定了可互操作的构建模块、构建平台和软件基础设施。

10.6 Samba

10.6.1 Samba 客户端

Samba 客户端使 TV 作为一个客户端可以去访问局域网上的共享资源。

10.6.2 Samba 服务端

Samba 服务端可以使电视作为局域网上的服务器去共享它的资源。

10.7 闪联

支持闪联协议，能够有效提高现有设备间的互操作性和易用性，充分利用不同设备的功能特点、为用户创造新的应用模式和丰富应用。

10.8 DiiVA

支持 DiiVA 技术，包括传送层和网络层，让视频、音频和数据封包，可以在 DiiVA 家庭网络中，安全地改变从任一信号源传送到任一显示器的路径。

11 浏览器 (可选)

11.1 浏览器功能要求

11.1.1 收藏夹相关功能 [必选]

11.1.2 浏览器设置相关功能 [必选]

11.1.3 页面焦点相关功能 [必选]

11.1.4 页面控制相关功能 [必选]

11.1.5 历史记录相关功能 [必选]

11.1.6 Cache 相关功能 [必选]

11.1.7 Cookies 相关功能 [必选]

11.1.8 公网浏览相关功能 [必选]

11.1.9 视频播放功能 [可选]

11.1.10 多标签相关功能 [可选]

11.1.11 页面搜索和查找相关功能[可选]

11.1.12 JAVA 游戏相关功能[可选]

11.2 浏览器协议要求

11.2.1 支持以下协议 [必选] :

11.2.1.1 HTML4.0.1

11.2.1.2 CSS1、CSS2、CSS3

11.2.1.3 JavaScript 1.5

11.2.1.4 XML

11.2.1.5 DOM1、DOM2、DOM3

11.2.1.6 SSL

11.2.2 支持以下插件接口 [推荐] :

11.2.2.1 NPAPI

11.2.3 支持以下图片格式 [必选] :

11.2.3.1 BMP

11.2.3.2 JPEG

11.2.3.3 PNG

11.2.3.4 GIF (87a/89a)

11.2.3.5 GIF_ANIMATION

11.2.4 支持以下文本编码格式：

11.2.4.1 utf-8 [必选]

11.2.4.2 Gb2312 [必选]

11.2.4.3 GBK [必选]

11.2.4.4 Gb18030 [可选]

11.3 浏览器界面要求[可选]

11.3.1 地址栏

11.3.2 菜单栏

11.3.3 标签栏

11.3.4 状态栏

11.4 WebCore 扩展

为了支持 HTML5 和智能电视操作系统 Web 浏览器中的新功能，WebKit 需要更新与智能电视操作系统扩展。这些扩展提供了良好的接口支持 Web 应用程序和它的发展。

智能电视操作系统 WebKit 由框架层的 java 部分和 webkit 共享库组成。共享库中的 webCore 部分是智能电视操作系统 webkit 扩展实现的主要部分。对于 javascript 引擎部分，目前有两个引擎存在，将统一使用 V8 引擎。

WebCore 扩展将加强引擎对 Web 应用的支持，主要包括以下几个方面：

11.4.1 HTML5 支持：

11.4.1.1 支持音视频，网页 3D。

11.4.1.2 计划添加 section,header,nav 等语义化元素，本地存储的支持。

11.4.2 插件支持：

11.4.2.1 浏览器支持 flash 等 NPAPI 插件。

12 DRM 和安全

本章描述了智能电视操作系统中 DRM 和安全子系统。

12.1 DRM

智能电视操作系统提供了一个可扩展的 DRM 框架。这个 DRM 框架使应用程序可根据内容的相关许可来管理受保护的内容，且支持多种 DRM 方案。

智能电视操作系统的 DRM 框架由两部分组成, DRM 管理层和 DRM 框架接口层。DRM 管理层实现 DRM 框架并公开 DRM 插件接口来处理下层的管理和解密，同时 DRM 框架接口提供应用程序框架给上层的应用。

对于应用程序开发者，DRM 框架提供了一个抽象的统一的 API，从而简化了对那些保护内容权限的管理。智能电视操作系统的 DRM 框架以插件的形式来实施网络内容提供商提供的 DRM 方案。

12.2 安全和权限

智能电视操作系统是一个具有单独特性的操作系统，运行在它上面的每个 application 都有一个唯一的系统 ID。因此 linux 也会使应用程序从其他应用和系统中分离开来。

所有的智能电视操作系统应用程序必须要用开发者自己的密钥来签订一个证书。这个证书用来确定谁是这个应用程序的作者。这允许系统授予应用程序一个 ID 也可以拒绝应用去请求一个其他应用程序已有的 ID。

12.2.1 应用程序签名和打包

12.2.1.1 应用程序签名

智能电视操作系统要求所有安装的程序都必须有一个开发者密钥签署的数字证书。标准的应用程序包也必须把公钥、公钥证书以及数字签名放到META-INFO文件夹。应用程序包中的META-INFO文件夹包括文件有：1) MAINFEST.MF 2) CERT.RSA 3) CERT.SF

智能电视操作系统用数字证书来确定一个应用程序的作者并且用来建立应用程序之间的信任关系。基于这种信任关系，应用程序可以安全的共享代码和数据。但是不能根据数字证书来决定哪些应用程序用户是可以安装的。

12.2.1.2 二次打包

为了确保安装在电视上应用程序的安全性，智能电视操作系统应用提供一种管理应用程序安装的方式。只有通过供应商审核和签署的应用程序才允许安装到系统中。因为应用程序会被用户破解，所以开发者公布的标准应用程序包必须使用新的文件格式进行二次打包。二次打包使用供应商的证书来验证和压缩。验证成功后应用程序可以按标准过程进行安装。

12.2.1.3 API 权限

用户在安装应用程序时可以授权或者不授权那些应用程序通过manifest文件请求的权限。智能电视操作系统定义了一长串的manifest权限来保护系统的各方面或其他应用程序。

在Manifest.permission [7]中可以找到智能电视操作系统提供的所有权限。但这不是一个所有可能权限的综合列表，任何应用程序也可以定义和实施自己的权限，如7.7.1节涉及的TV Service相关的权限定义。

12.2.1.4 越狱和防篡改

越狱是指用户利用智能电视操作系统的漏洞可以执行管理者权限的任务，包括更新系统，安装非官方应用程序商店的应用，访问系统分区，修改受保护的系统文件，并改变他们的权限等。

为了防止发生这种越狱情况，应尽量解决智能电视操作系统存在的隐患，避免应用程序或恶意软件获得超级用户的特权。用户获取root权限的系统服务也应该关闭。

防篡改是指用户用来保护关键内容的一种机制，保护对象可以是核心系统库，服务和那些系统运行时不能被修改或损坏的关键数据。

13 系统升级与维护

本章描述了系统升级与维护相关的需求规范，主要包括系统升级、系统备份、系统恢复及恢复出厂设置几个部分。系统升级与维护子系统是基于智能电视操作系统的 recovery 模块来实现的。

13.1 系统架构

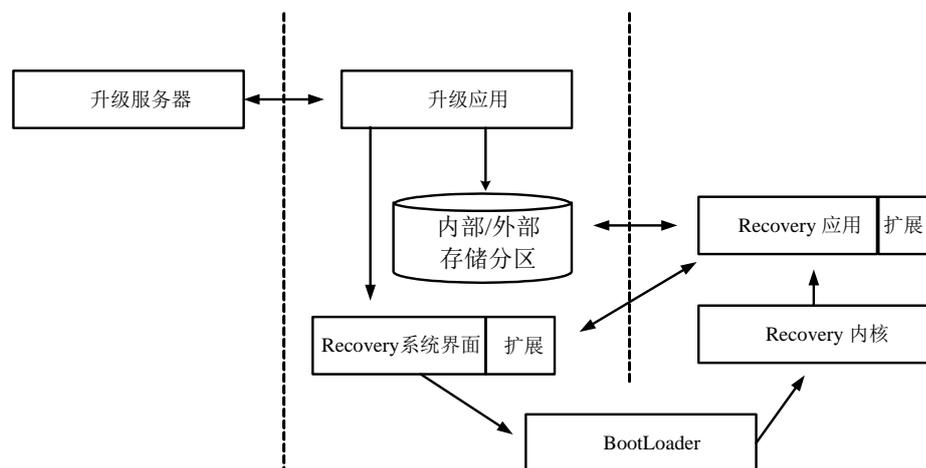


图5： 系统升级模块交互

图 5 描述了升级系统间主要模块的交互过程。智能电视操作系统升级应用与升级服务器交互获取升级包，升级包下载到设备内部 NAND 分区或外部存储设备如 SD 卡或 U 盘，然后 boot loader 重启设备并引导进入 recovery 系统，在 recovery 系统进行升级包的处理和安装工作。

13.2 系统升级操作方式

系统升级有两种启动方式：开机时自动启动和用户通过系统设置手动启动。系统升级包可以通过网络下载也可以从外部存储器直接加载。系统提供一种备份当前系统镜像到内部 NAND 分区或外部存储设备的方式，以便于用户对系统进行恢复或还原。在恢复或还原时，可以检测预设属性字段值，根据

所述预设属性字段值判断当前操作系统是否需要升级，如果需要升级则重启当前系统进入恢复模式。

13.3 系统升级范畴

系统升级至少应支持内核部分，用户数据和系统三个部分的升级，同时应支持的升级文件有镜像文件、库、升级应用程序、JAVA 包等，升级系统也要支持增量或补丁式升级。

13.4 NAND 分区要求

为了支持 Recovery 系统，NAND 分区需要划分单独的分区用于烧写 Recovery 镜像。同时有足够的 data 分区临时存放系统升级包或长时存放系统备份文件，单独的 cache 分区保存智能电视操作系统与 recovery 系统的交互信息。

分区名	描述
Recovery	存放 recovery.img
Data	存放升级包和系统备份文件
Cache	存放智能电视操作系统与 recovery 系统的交互信息

13.5 其它功能需求

13.5.1 安全和保护

网络下载升级包后要进行完整性校验，升级过程中如果出现断电、重启或其它未知原因导致升级失败，系统可能无法正常启动，但 recovery 系统是正常的。

13.5.2 备份和恢复

用户数据及系统应该能够备份，当需要恢复和还原时可以通过预先的备份实现。

13.5.3 恢复出厂设置

通过在 Recovery 模式下格式化数据区数据和缓存数据，实现恢复出厂设置。

13.5.4 Bootloader 和 Recovery 的更新

智能电视操作系统应该支持从外部存储器对 boot-loader 和 recovery 分区进行更新。

13.5.5 Recovery 模式下进行升级包下载 (可选)

Recovery 系统中增加网络功能，Recovery 应用可以和升级服务器直接交互，从升级服务器直接下载升级包进行系统安装更新。

本规范由中国电子视像行业协会组织制定和发布，并负责对所有规范的修改、解释。
