

Standards Propel DTV Forward
How will the TVLA help digital TV deployment?
Chris Dinallo
Pace Micro Technology Americas



Abstract

With the latest announcement of the TV Linux Alliance (TVLA) specification release, what does this mean for the digital home gateway market and how does it help promote the push of Digital TV? This article addresses these questions and explains the goals of TVLA..

SOFTWARE EVOLUTION AND RE-INVENTION

Anyone associated in the technology industries knows the saying that “change is a constant,” but how many of us really ask why? The immediate answer that comes to mind is because technical innovation happens – it’s the very reason why us engineers always try to build a better mousetrap. Software engineering is no different. From the early days of writing monolithic code to progressing to structured application programming interfaces (APIs) and further advancing to object-oriented technologies mixed with message protocol techniques, software engineering has indeed proven that “change is a constant.” Yet with all this change, is the same mousetrap being re-invented? Just ask any software engineer if they have ever written any code that creates and manages a linked-list. The answer is a surprising, almost worrying, - “Yes”. Thousands of software engineers have re-invented the linked-list. What a waste of time, skill, and dollars.

” Software is the function in the form.”

DIGITAL TV EVOLUTION

Fast forward to today and the impending rollout of Digital TV (and home gateways). DTV is as demanding and complex as any of the other technical industries in the world

today. For starters, the silicon hardware is state-of-the-art with tight integration and high performance CPUs. Further, it’s the function the silicon performs that drives the very primal senses man has – sight, sound, and touch. And I wouldn’t bet against future innovations in smell and taste either! Yet, this advanced silicon is merely silicon without well-written software to drive it. For it’s the software logic that is the function in the form. Herein lies the problem: if software is the logic and the logic is constantly getting re-invented, then how will the DTV evolution ever really move forward?

ENTER TV LINUX ALLIANCE (TVLA)

TVLA is a consortium that was formed in June 2001 by Sun Microsystems, Motorola, Lineo, and Liberate. Each of these four ‘A’ Founders are industry leaders in the cable, satellite, terrestrial, and telecommunication sectors. The alliance of four immediately opened membership for a limited time to other industry leaders to join as ‘B’ Founders to start the process of developing a standard for digital TV products. The mission was straight forth with no hidden agendas: to promote DTV products with improved time-to-market and robust feature sets by creating Linux standards and a standardized programming interface (API). Let’s step back for a moment and ask why this was needed. Industry leaders realized that too much effort was being spent developing the same software over and over again each time the silicon hardware changed. There was proof that for DTV products to become ubiquitous, focus and effort must be applied in the proper logical areas and thus re-invention would only slow progress and confuse the market.

Now, 15 months since formation, the Alliance has released its initial specification – version 0.8. It was voted by the founding members to release the 0.8 specification in order to move forward on the agreed interfaces.. While there

is still much work to do to achieve a 1.0 specification, the motivation behind release 0.8 is two-fold: (1) make visible the direction and interfaces specified, thus early adopters can start developing and provide feedback to the Alliance; and, (2) open up a third tier of membership into the Alliance where new members can directly participate in evolving the specification.

FIRST STEPS TO DTV STANDARDIZATION – WHY LINUX?

Linux was chosen as the Operating System (OS) for DTV home gateways because of its advantages over other OS'ses. The most publicized is Linux's royalty- free open-sourced environment based on the GNU General Public License (GPL). GPL mandates that programs (under GPL license) must be distributed with source code and such code is allowed to be freely modified. Furthermore, such modifications will be subject to GPL terms & conditions. With Linux falling under GPL, it is in prime position to meet the thrust of the TVLA's mission and prevent software re-invention.

Linux also brings other benefits such as a large tool chain to tap into - - from code development tools such as editors, compilers, and linkers, to code analysis tools to aid in debugging and performance tuning. Most of these tools are available from the Free Software Foundation. While these benefits reduce development costs and time-to-market, can Linux technically meet the DTV needs? In a word, "Yes". Linux is both robust and efficient. It's architecture is based on a protected memory model with multitasking capabilities –meaning it can execute numerous programs independently by time slicing the processor without the risk of one program invading another's memory space causing ill effects and systems crashes. Linux is also a 'deterministic' OS which is ideal for streaming data. And, although not needed for DTV applications today, Linux can be modified to be a 'real-time' OS via

programming techniques and kernel extensions. Another technical boon for Linux is the vast number of engineers around the world who are skilled in Linux and developing Linux programs. This provides another leg up for the DTV industry to tap into that large engineering community without the usual training costs and time normally associated with proprietary OS'ses and their environments.

TV Linux Alliance, Linux4.TV, LinuxTV:

Due to the ever growing popularity of Linux based systems there has been more organizations and alliances formed to promote their respected industries. Within the DTV/ Set-top market, there are three such initiatives. This accompanying article has focused on TVLA. As of this past August, a new website was launched entitled Linux4.tv. The sponsors of such being National Semiconductor and Century Embedded Technologies. Their goals are focused on supporting Linux developers and promoting open-sourced projects within the same markets as TVLA. The two initiatives are actually complementary to each other where TVLA's focus is on promoting hardware-independent open-standards and Linux4.TV's main focus benefits those projects that are based on National Semi hardware. Furthermore, Linux4.TV has expressed its support and interest in TVLA and similarly TVLA is encouraged to incorporate Linux4.TV's technologies as appropriate.

The third initiative is LinuxTV which happens to be the oldest of the trio that was formed back in 2000 to promote open-source projects based on Europe's DVB standards. What was happening in Europe as MHP implementations were evolving was much of the low layer software APIs were proprietary preventing the Linux community from contributing in an open fashion. Companies such as Nokia and Convergence (in Berlin) have contributed heavily to LinuxTV with their contribution of the DVB APIs. Convergence is also a founding member of TVLA and has contributed to it as well. As of recently, the DVB APIs have been adopted into the official Linux kernel 2.5.44-ac1.

So as what might appear to be confusion and contention in the Linux DTV/Set-top development markets, there is actually a lot of synergy happening worldwide.

THE DTV APPLICATION PROGRAMMING INTERFACE

One can now see why TVLA chose Linux as the first step towards standardization. But, how does the TVLA ensure that the Linux development community doesn't just do the same re-inventions for specific silicon hardware DTV implementations that have been taking place on proprietary platforms? The answer is to create standardized DTV programming interfaces that prevent such re-invention. Let's look at the software architecture that defines the TVLA APIs. Referring to figure 1, the software stack starting at the bottom on top of the Linux kernel is the device driver layer. This layer provides the logic that drives the silicon hardware while interfacing with the upper layers in the software stack. The drivers perform their duties by utilizing services from the kernel. Although the drivers usually need to be developed for specific silicon hardware, the interface to the upper software layers does not, hence TVLA has defined a standardized set of device driver APIs. Also in figure 1, you can see that TVLA defined the interfaces into the Linux OS. This was needed to reduce the number of variants that exist today. For example, in the area of the OS Process API, a decision had to be made for consistency reasons to be POSIX (Portable Operating System Interface) compliant or specify the Linux variant. TVLA chose the Linux variant because it was more consistent with the Linux development community and in some cases offers less overhead. It should be noted that many of the basic Linux APIs are in fact POSIX-compliant.

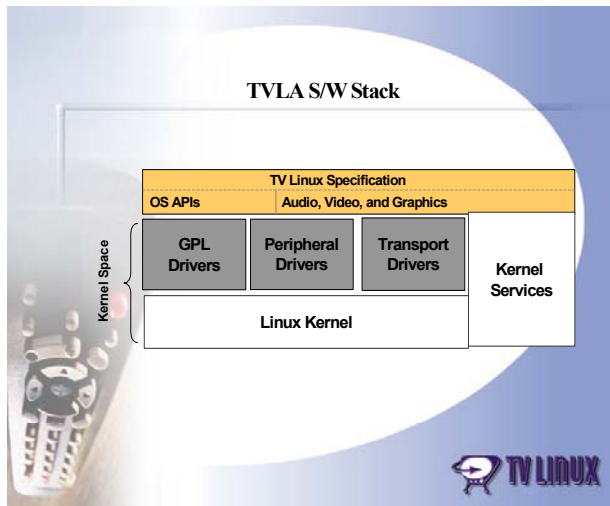


Figure 1

Another area of standardization specified are the services such as networking, audio, video, and graphics. For networking services, TVLA chose to leverage the well-known BSD (Berkeley Software Distribution, the de facto TCP/IP implementation) socket interface and its services. BSD sockets are specified as mandatory by TVLA if networking services are to be implemented. For the graphics and video subsystem, TVLA states that the General Graphics Interface (GGI) is the preferred subsystem, but not required at this initial release of the 0.8 specification. It is anticipated that a future specification release will include GGI as mandatory. On the other hand, the audio subsystem is specified as

being based on the definition of the ALSA (Advanced Linux Sound Architecture) Library API version 0.9.

Lastly, yet the focal core to DTV devices, are the Transport Stream Device Interfaces APIs as defined by TVLA. It's these core interfaces that perform the main duties of a DTV device, hence are common across all DTV devices and can be leveraged accordingly. Let's explore those next.

THE DTV CORE

I refer to the Transport Stream Device Interface as the DTV core because it is the software module(s) that differentiate it from its computer-centric ancestors (i.e. Computers). As the name suggests, these modules drive the MPEG-2 Transport Streams through their silicon processing functions in a DTV home gateway. The TVLA recognized the commonality and standardized the interfaces, thus preventing software re-invention. The services specified in 0.8 specification are the Front-End Tuner (FE) and the Transport Stream Demultiplexer (Demux). As shown in figure 2, the FE function list is described. Note that these interfaces require mandatory compliance.

Routine	Description
open()	Open a named frontend device.
close()	Closed an open frontend device.
OST_SELFTEST	Initiates an automatic self-test of the front-end hardware.
OST_SET_POWER_STATE	Controls the state of the OST device power (on, off, standby, or suspend).
OST_GET_POWER_STATE	Return the power state of the OST device.
FE_READ_STATUS	Return status of the Front-end.
FE_READ_BER	Return the bit error rate for the signal currently recieved by the Front-end.
FE_READ_SNR	Return the current signal-to-noise ratio.
FE_READ_SIGNAL_STRENGTH	Returns the current signal strength.
FE_READ_UNCORRECTED_BLOCKS	Returns the number of uncorrected blocks detected.
FE_GET_NEXT_FREQUENCY	Scan to the next appropriate frequency.
FE_GET_NEXT_SYMBOL_RATE	Increment to the next desirable symbol rate.
FE_SET_FRONTEND	Start a tuning operation using specified parameters.
FE_GET_EVENT	Returns an event of type qpskEvent if available.
FE_GET_INFO	Return information about the Front-end.

Figure 2 – Front End Tuner Functions

Similarly, the Demux interface and functions shown in figure 3 are also listed as mandatory. Together, these functions allow a DTV home gateway to acquire and process a broadcast transport stream. The output of these functions feeds into the audio, video, and graphics subsystems for presentation to the viewer.

Figure 3 – Transport Demultiplexer Functions

Notice that TVLA only specified the basic services leaving other services like Conditional Access (CA – mechanism for authorization of pay TV services) and Personal Video Recorder (PVR) for future standardization. This was done to not stifle technological growth in an emerging industry.

LEAVING ROOM FOR INVENTION

With all this standardization and software reuse, how will companies differentiate themselves and retain that intellectual property? Well, fortunately both Linux and

Routine	Description
open()	Open the named demux filter device.
close()	Close an open demux filter.
read()	Read filtered section or PES data.
write()	Used to replay a recorded transport stream. (available only on selected devices.)
DMX_START	Start the filtering operation.
DMX_STOP	Stop the filtering operation.
DMX_SET_FILTER	Set up a filter with the provided filter and mask.
DMX_SET_PES_FILTER	Set up a PES filter.
DMX_SET_BUFFER_SIZE	Set size of circular buffer used for filtered data.
DMX_GET_EVENT	Return an event if available.

TVLA have a mechanism for dealing with proprietary software modules that companies may not wish to disclose under GPL. Going back to figure 1, the software module labeled peripheral drivers can be any proprietary driver the developer wishes to create. Such examples are numerous devices like front panel functions, infrared and remote control devices, I/O ports, and conditional access as previously mentioned. So it's not difficult to have your cake and eat it too in this environment.

ADHERENCE TO COMPLEMENTARY STANDARDS

A final contention point that may arise is whether TVLA is in line with other DTV open standard initiatives. As shown in figure 4, Open Cable's Application Platform

(OCAP) and DVB's Multimedia Home

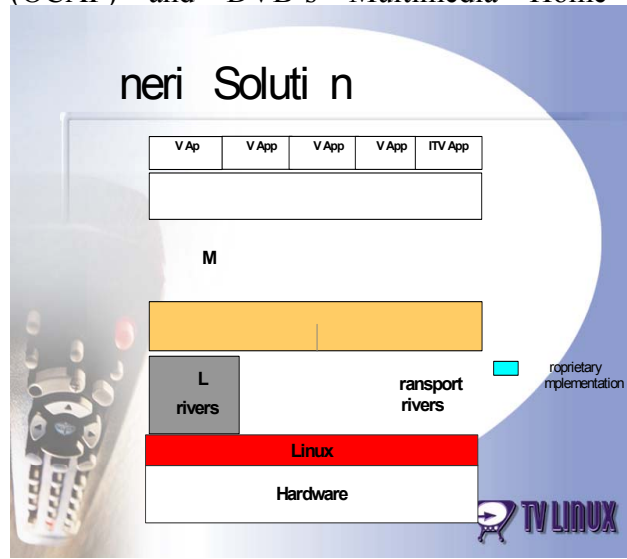


Figure 4 – Coexisting Standards

TV LINUX LAUNCH

Critics may question the viability and level of the current specification 0.8; however, all standards have a starting point and this one is no different. The specification will allow industry leaders and the large Linux community to begin to develop to and enhance the TVLA standard. Its benefits are:

- Being cost-effective to both developers and Operators as a means to reduce royalty license fees from proprietary operating systems.
- Spring boarding more competition in creating DTV applications and enhanced applications.
- Standardizing DTV interfaces.
- Begin in sync with other open standard movements such as Open Cable and MHP.
- Leveraging existing software re-use common to DTV platforms.
- Leveraging a large skilled Linux development community.
- Leveraging a vast Linux tool chain for software development.

Hence, one will find it hard to argue such benefits for the good of the industry. It's time to move forward "enabling digital."

Chris Dinallo, Chief Technologist

Mr. Dinallo's responsibilities for Pace focus on US Cable and include digital home gateway development, future technological directions, participation in standard bodies such as Cable Labs, SCTE, and TV Linux Alliance. Chris brings 18 years' experience developing innovative solutions in software and firmware. From 1984-89 his focus was developing firmware for peripherals in the personal computer space. Since 1989, his expertise working for IBM has been in the discipline of operating systems (DOS Kernel, OS/2, and MACH Kernel), multimedia subsystems (lead audio architect), and object oriented technologies (Taligent OS) where he holds patents, white papers, and trade publishings. More recently and prior to joining Pace, Dinallo has held engineering director positions at DVD and Voice over IP companies. In total, these career disciplines complement each other in the realm of digital home gateway development.

References

TVLA

<http://www.tvlinuxalliance.org/>

Linux4.TV

<http://www.linux4.tv/>

LinuxTV

<http://www.linuxtv.org>

Free Software Foundation

<http://www.fsf.org>

Linux Stand Base

<http://www.linuxbase.org/>

GGI

<http://www.ggi-project.org/>

ALSA

<http://www.alsa-project.org/>