TWO YEARS OF DEPLOYING ITV/EBIF APPLICATIONS – COMCAST'S LESSONS LEARNED

Robert Dandrea, PhD
Distinguished Engineer
National Engineering & Technical Operations
Comcast Cable

Abstract

Comcast has deployed several EBIF-based Interactive TV applications very broadly to its footprint in the past two years. The goal of this paper is to share the technical challenges and lessons learned from this broad deployment experience. Besides reviewing some video platform design considerations that have evolved from this experience we also describe some work done collaboratively with Canoe and Cable Labs that also has aided in creating a scalable EBIF delivery ecosystem.

One challenge has been in extending Comcast's operational support system to cover these new applications. The paper will therefore discuss both platform and process advances that will enable us to support iTV broadly across our footprint. The paper also discusses how Comcast manages scalability as the number of applications and the number of enhanced networks increases.

Comcast's iTV Experience

Over the past two years Comcast has deployed several EBIF-based interactive TV (iTV) applications very broadly throughout its video network. Our currently deployed EBIF user agent – the set top box (STB) client that executes the applications – is compatible with the I05 EBIF specs ([1], [2]) and is deployed on 21 million STBs. Our initial wave of EBIF applications are available in 12 million subscriber homes. The most broadly deployed applications include

1) Caller ID – Comcast digital voice and digital video customers can receive a brief banner popping over any currently viewed video on their TV identifying incoming phone calls.
2) Home Shopping Network's "Shop By Remote" – a bound iTV application carried 24x7 in the HSN video feed which allows viewers to purchase directly through their TV.
3) EBIF enhancements bound to 30 second advertisements locally inserted (via SCTE 30/35) by Comcast's Ad sales team, Spotlight. These include the "RFI – Request for Information" application that offers viewers a direct mail or phone call response from the advertiser and the "Ready – Remind/Record" app that allows a user to automatically set DVR recordings or guide reminders during commercial breaks that advertise upcoming programming.

Currently we are readying more applications for customer deployments, including several bound applications carried in national programming (including some done in partnership with Canoe Ventures, and several unbound applications available over any programming, including enhancements to our Guide and Instant-Info, which is a "News-Weather-Sports" widget-like app.

In the remainder of this paper we shall review some of the major design aspects of our video delivery platform that help to enable a reliable full-footprint deployment of these applications. One of our major lessons

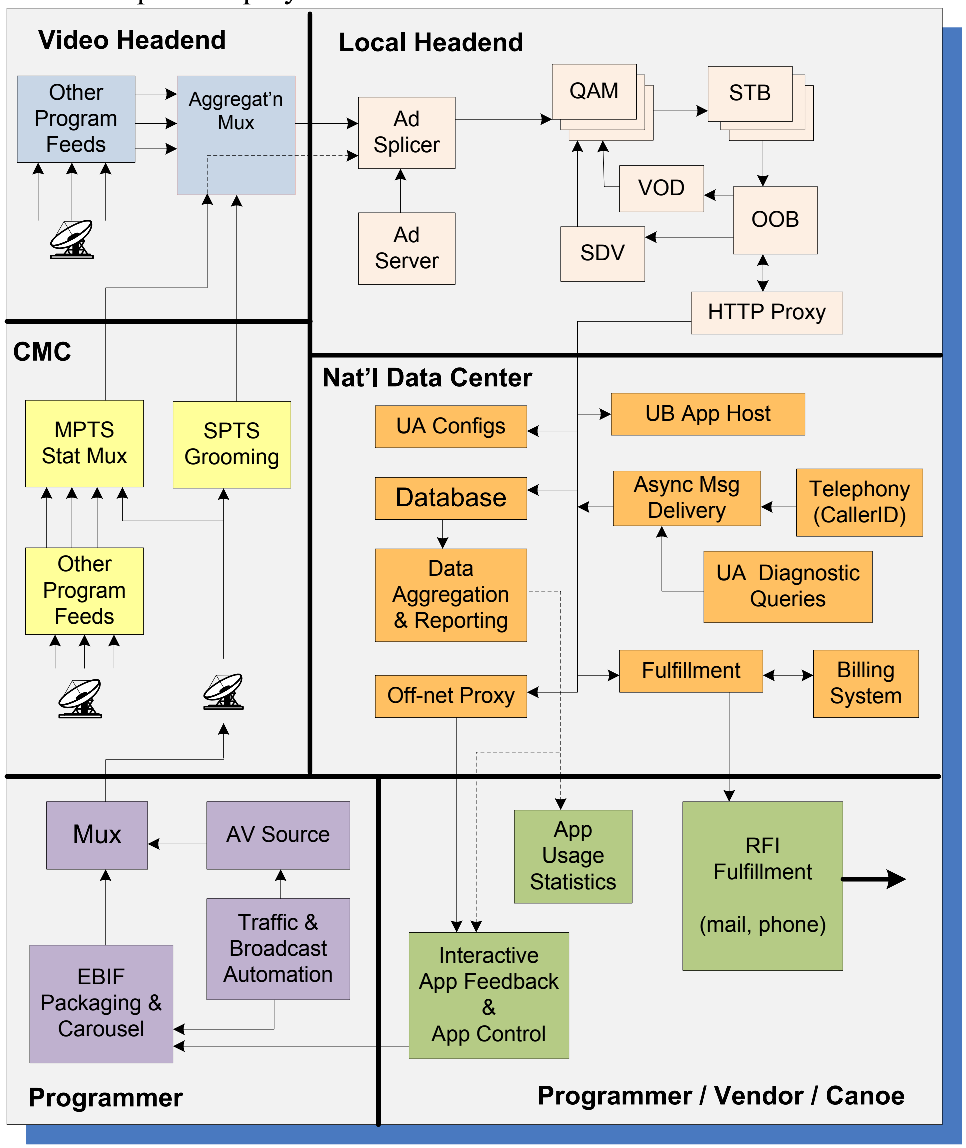

Figure 1: Comcast EBIF Platform - Simplified Functional Architecture

learned has been the need to create an up-front design that meets the operational requirements of reliability, verifiability and problem isolation, we stress platform aspects that contribute to those goals. We also summarize some work done collaboratively with other MSOs, Canoe Ventures and Cable Labs that also has aided in enabling a scalable deployment of many apps across many enhanced networks.

A PLATFORM FOR RELIABLE AND SCALABLE EBIF DELIVERY

Figure 1 shows a simplified functional overview of the Comcast video platform for delivering both bound and unbound EBIF-based iTV applications. In the following discussion we shall look at this architecture assurance and will enable us to scale to deliver dozens of different applications over dozens of different networks in the near future.

<u>Programmer</u>

The "Programmer" block in Figure 1 shows a typical architecture at a programmer's broadcast center for delivering a bound EBIF application over a nationally distributed video network. That platform consists of

1) an EBIF packaging system (to translate graphic app directives described in proprietary schemas used by application developers into normative EBIF resources).
2) a data carousel to encapsulate those resources into MPEG packets and spin those packets for inclusion in the full transport stream. Generally we have found 100-200 kb/s is sufficient for simple EBIF apps.
3) integration with a broadcast automation system so that iTV applications can be constrained into or out of commercial breaks.
4) the standard AV source and a mux to combine it with the EBIF data carousel.

It has taken a significant effort over the past few years to develop an EBIF broadcasting system simple and reliable enough to be integrated with a programmer's existing AV broadcast gear. Syncing nationally broadcast apps with commercial breaks (to remain in the breaks for enhanced advertisements and to remain out of the breaks in the case of enhanced programming) has been a particular challenge. We have used automated signaling such as GPI or SNMP emanating from the programmer's traffic automation systems to keep EBIF apps in or out of commercial breaks. Initially we have

block-by-block, and analyze some of the platform elements that aid in EBIF service

used Comcast-developed subsystems for (1) and (2) above, but in the past year we have worked more collaboratively with vendors and with Canoe to standardize this part of the platform.

Figure 2 shows a sample system design we have used for managing the synchronization of applications across commercial breaks. In this example we've used our own internally developed servers for application packaging and carousel streaming. The design shows the thoroughness at which redundancy is implemented and the manner in which a GPI (general program interconnect) signal from a video broadcast automation system is used to disable and enable an EBIF application. The design shows an app server communicating with a primary and backup app packaging system, and both packagers sending XML descriptions of EBIF page and data resources to both a primary and backup carousel server, where the XML directives are transformed into an MPEG carousel of EBIF and EISS data pids. Both SD-East and SD-West (and potentially HD) versions of the service are simultaneously enhanced. The GPI signals are then used by a redundant pair of ETV filter servers to filter off the EISS signaling passing through them during an ad break. Both primary and backup filtered EBIF carousels are relayed to a mux, where only the primary is used until failover. The GPI triggers need to occur 5 seconds prior to the start of an advertising break due to the EBIF spec requirement that user agents suspend applications (remove them from the TV's on-screen display) within 4 seconds of loss of EISS signaling.

As mentioned above, we have more recently used vendor's EBIF packaging and carousel systems at programmer broadcast systems, and have in this case employed a design for synchronizing the applications to commercial breaks which relies on the ad break signaling altering the app packaging in a manner to temporarily suspend the app.

Another engineering problem within the programmer's broadcast system that needed to be solved was the determination of a

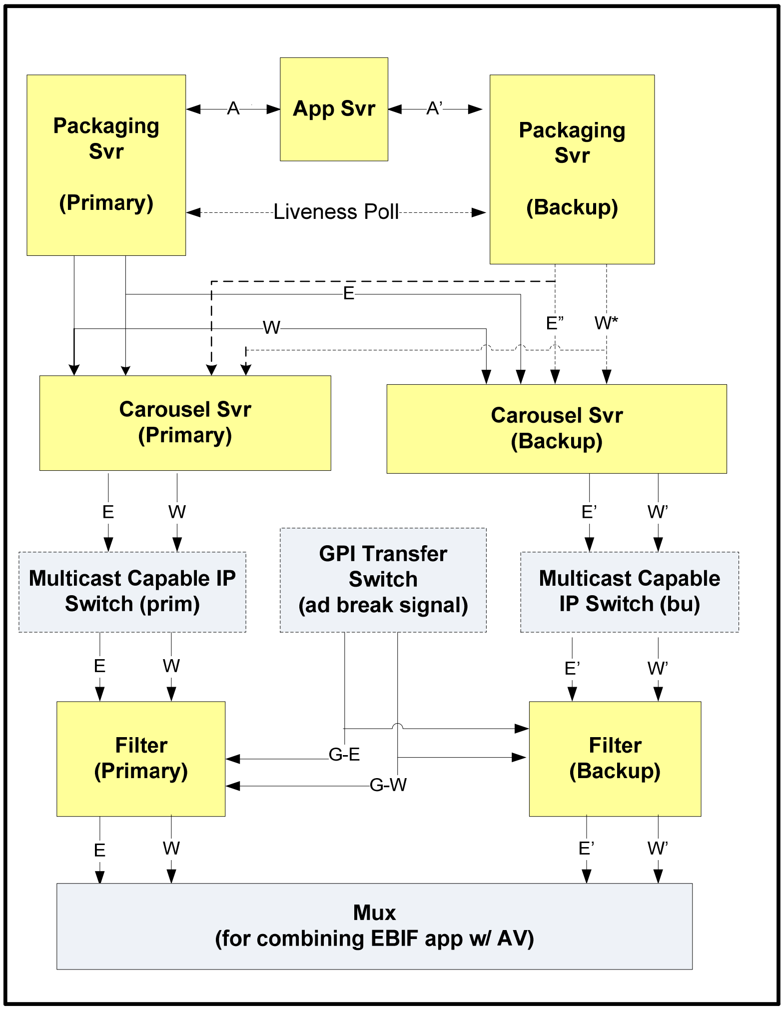

Figure 2: Sample design for EBIF broadcast system synced with ad breaks

mechanism for a programmer to enhance their broadcast feed for some markets or some MSOs without impacting other markets and other MSOs that do not receive the app. We have seen two different general solutions to this: (i) If the programmer has spare satellite bandwidth, they can replicate the program feed on a whole new channel. (ii) More generally the programmer has simply added the new iTV data pids to their single production feed, and added a secondary PMT that includes those new pids (but shares same audio and video pids as primary PMT). Markets / MSOs unauthorized for the app simply continue reading the old PMT without the iTV data pids listed. Other satellite receivers can be separately authorized for the new PMT.

Comcast Media Center (CMC) and Video Headends

The CMC is Comcast's distribution hub for much of its video content. The CMC delivers to local markets nearly all our VOD, HD and SDV content, and a good fraction of our digital SD content too. One major platform change that we are making, partly for the operational advantages it offers for bound application delivery is requiring all local markets to use CMC re-broadcast feeds for EBIF-enhanced programming rather than direct-from-programmer feeds. This simplifies and empowers our operational capabilities, in that a single monitoring point can verify the iTV application at that seminal point in its delivery path, and a single control point offers national on/off control of applications (via pid dropping) at a moment's notice. Furthermore, the use of a standard delivery path through the CMC shortens our application deployment timelines, as local markets needn't be concerned with some EBIF-centric broadcast issues described below.

Use of CMC re-broadcasts also helps manage the in-band bandwidth issues that may arise when multiple programs in a multiplex are enhanced with EBIF applications. The CMC generally offers two types of programmer re-broadcasts: SPTS and MPTS (single and multiple program MPEG transport streams). The SPTS are constant bit rate encoded streams, with the bitrate fixed and common across all streams for bandwidth management reasons (e.g. for Switched Digital Video, SDV). Thus any EBIF app data will simply consume part of the total available bandwidth allocated to that service. A more efficient scheme, however, is to use stat-muxed MPTS transport streams that have been encoded using state-of-the-art multipass MPEG-2 encoding gear that can more optimally find bandwidth space for the EBIF apps in a shared manner across the full multiplex. For our more popular (non-SDV) content, that is our path forward.

Figure 1 shows four different manners that these CMC re-broadcast feeds are treated at the aggregation mux in the market video headend:

1) The SPTS feed can be muxed with other programs into an MPTS feed without any further grooming, thus maintaining its CBR nature. This is the delivery path for services on that are switched onto the RF plant by SDV.
2) The SPTS feed can be re-groomed with other programs into an MPTS stat mux feed.
3) The local aggregation mux can cherry-pick one or several programs out of the CMC MPTS feed, and re-groom them into another MPTS stat mux.
4) The full MPTS from the CMC can be used by the local market without any further grooming.

As described above method (1) above is used for SDV feeds while method (4) above is our path forward for non-SDV services, as it most efficiently uses the available 38 Mb/s of a standard QAM-256 channel. Delivery methods (2) and (3) are only interim steps while moving fully to CMC rebroadcasted programming (and, in the process, dealing with technical challenges like local blackout requirements in some programming).

One technical challenge has been the creation of an encoder/mux system for grooming enhanced programmer feeds that does not substantially de-sync the EBIF app from the underlying audio-video when the video is groomed. Since the EBIF spec uses no timing synchronization to an inherent transport stream timeline like a PCR, proper synchronization of the EBIF app with the underlying audio/video must be maintained carefully when the video is regroomed after the app is embedded. By requiring a maximum of 1-2 seconds of downstream de-

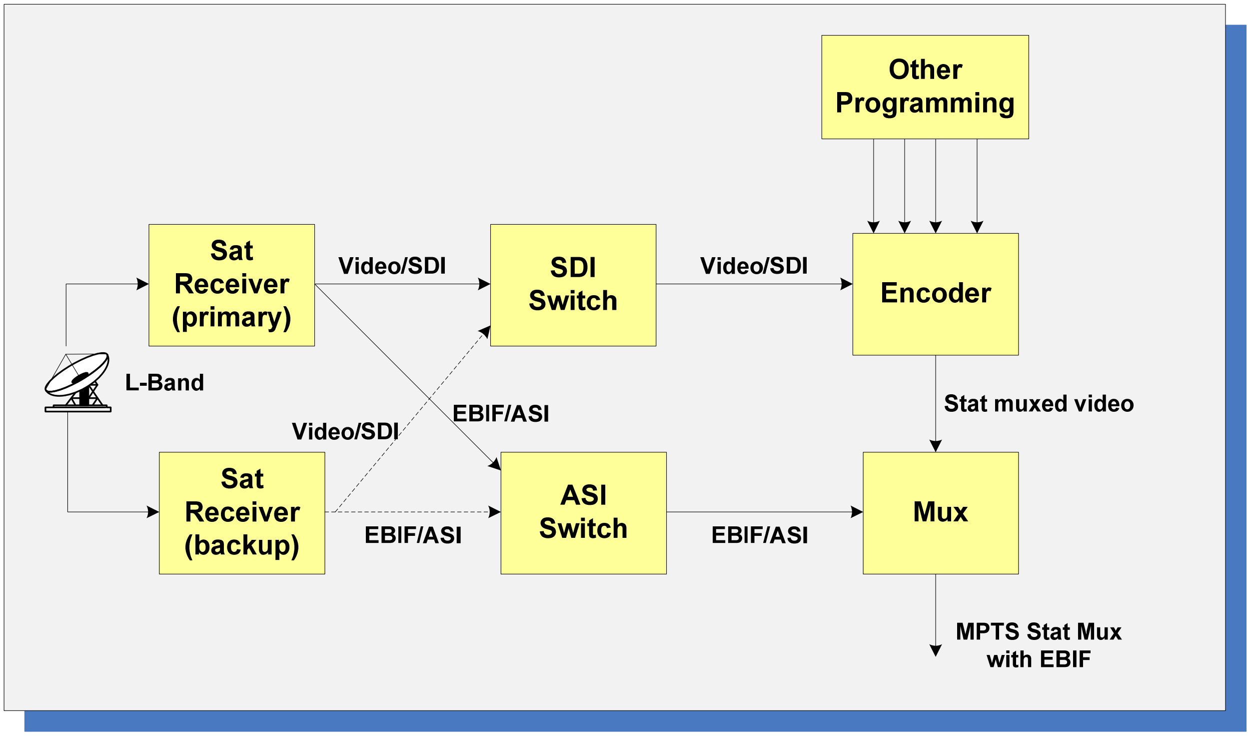

Figure 3: Sample system design for re-grooming EBIF enhanced programming

sync due to further grooming, and requiring that the app is designed so that it can suffer a 1-2 second de-sync without any adverse consequences (e.g. the design for synchronizing the app with commercial breaks has built into it these 1-2 seconds of potential de-sync), we have been able to manage this problem.

A sample design for video grooming of an enhanced service is shown in Figure 3. That design shows an L-band satellite signal as input that feeds two redundant satellite receivers that each have an SDI output for video and an ASI output for the EBIF and EISS app data. SDI and ASI switches then support failover between the two sources. The video is groomed into a stat mux with several other programs, and the EBIF/EISS data is subsequently muxed in. The particular encoder used showed a worst case delay of about 1 second, which would de-sync the EBIF app and audio-video in a manner that would push the app forward up to 1 second in the programming as compared to its original position.

The complexity of the system in Figure 3 demonstrates another advantage to using CMC feeds rather than direct-from-programmer feeds in our local markets: If the market can use the full CMC feed without further grooming, the difficult operational work of building, verifying and maintaining a regrooming system that supports EBIF in each local market is not needed. Centralized distribution of already groomed signals offers a huge operational leverage.

In order to support the broad operational goals of verifying bound application delivery fully from app origin to STB and of enabling traceability of any problems, we are currently developing an MPEG monitoring system to snoop EISS and EBIF data pids at critical junctures in our video delivery platform. The goal of the system is to improve application uptime by supporting both automated alarming when an application fails to be delivered past some network node, and by enabling post-flight analysis to help troubleshoot any problems that arise. Besides occurring at the centralized CMC re-broadcast point and at video headends, this monitoring can also occur downstream of local headend edge muxes / ad splicers.

A final aspect of aspect our headend-centric platform enhancements has been in standardizing mux configurations (for various mux vendors) to meet new requirements on these muxes stemming from iTV, such as the need to prevent un-authorized EBIF content from passing and the need to insert 24x7 EBIF-enabled placeholder PMTs in programming that will support local insertion of enhanced advertisements if the national feed is not already enhanced.

Local Headends

Comcast's current platform design for inserting local advertisements enhanced with EBIF interactive applications is one that minimizes the operational impact of those enhancements: we simply pre-embed the app into the MPEG asset for the ad before installing the ad on the local SCTE 30/35-based ad server. The impacts are then mux configurations as summarized above and verifying the ad server is updated with now-standard software to support the additional two iTV data pids. Before considering last-minute dynamic insertion of EBIF into local ads at the flight time of the ad, we have felt it wise to first perfect the EBIF package-carousel-mux systems operationally at national broadcasts centers before deploying hundreds of such systems at our edge ad zones.

One aspect of iTV application scalability has been in managing any impact the applications might have on our legacy Aloha (non-DOCSIS) out-of-band (OOB) bandwidth. One platform element that has contributed to that ability has been the buildup of monitoring gear on our OOB paths to measure the impact of these applications. Since we can only estimate concurrent application usage statistics, these measurements alarm us if iTV applications use an unexpectedly high fraction of our OOB bandwidth. We have not seen this occur thus far in our deployment history, but this mechanism should inform us of any need for OOB node size adjustments.

Although our tru2Way user agent can send HTTP direct from the STB over a DOCSIS channel, our legacy user agent does not and so we have developed a proxy that sits in each local headend where the interactive messaging from an application is changed to/from HTTP on the upstream/downstream. The communication between the STB and the HTTP proxy is via a proprietary protocol which adds reliable message acks/retries to the core UDP layer, and minimizes OOB bandwidth by allowing encoding of lengthy URLs and STB MAC addresses. As this HTTP proxy is the conduit through which all interactive messaging flows from legacy STBs at each headend, we have also built into our operational support system the ability to snoop and decode application messaging at this network element, filtering by application or by STB ID.

National Data Centers

Figure 1 displays several server systems deployed nationally to support iTV.

1) All unbound (UB) apps are served from the same web server ("UB App Host" in Figure 1), and this server supports

logging to track UB app usage and server scaling behind a load balancer.

2) Our user agent extracts a STB-specific configuration file from the "UA Configs" server at STB boot-time and daily afterwards. This supports many operational purposes, including enabling market-by-market (and even STB-by-STB within a market) level control over such aspects as which apps are authorized to run and application authorizations for STB resources (such as persistent memory).
Within our current footprint of STBs with an I05 compatible user agent, we have developed a scheme to use the test-flag described in the I05 EISS (EBIF-AM) spec to manage our needs to limit app display within a market for trial purposes (e.g. limitation to only headend or employee STBs). That scheme has the 8 bits in the test-flag partitioned so that one bit is used for headend STBs, one bit for employee STBs, 3 bits are used for customer-phase rollouts of Comcast-internal apps and the remaining 3 bits used for customer-phase rollout of cross-MSO apps (e.g with Canoe Ventures). The fact that a single national broadcast of a bound app contains a single test-flag value that goes to all markets and to all MSOs limits the ability of this test-flag mechanism to support multiple trials of multiple apps on multiple networks. For that reason we describe future work to improve this platform functionality in the sections below.

3) We have built an asynchronous message delivery system that is shared across several interactive applications (such as Caller ID) and by an OSS server that performs user agent queries to help diagnose and isolate problems. We have also built a system to continuously inject synthetic transactions into this messaging platform throughout our full footprint, to proactively monitor and alarm for any connectivity problems.

4) Fulfillment for both Comcast-internal and Canoe-originated RFI applications occurs through a national server system that optionally may extract billing system customer identification info if the customer so grants.

5) All messaging from an app (including both user-initiated interactions and app-level usage logging that occurs regardless if users interact) flows through a proxy-point that serves several purposes operationally. For HTTP posts that flow to an external 3rd party host, the "Off-Net Proxy" performs the following functions:
   a) It relays the posts to 3rd parties, generally by adding encryption and doing obfuscation of private data (such as STB MAC addresses).
   b) It acts as a firewall to ensure communication is with an authorized host and that HTTP responses from such authorized hosts follow the technical requirements agreed upon (e.g. response size is within allowed range)

   For all posts (kept internal or relayed to 3rd parties) the national proxy point allows for entry of the Post data contents into our internal database. Queries to that database then allows
   a) aggregated app usage reporting for business purposes
   b) handoff of aggregated data for easy feedback into the app (e.g. vote/poll info), either to the programmer, to Canoe or to a vendor (whomever is managing the application broadcast)
   c) aid in validating app delivery and service assurance at an ad zone level (by tracking the app display logging

message at a zip code and ad zone level)
d) the ability to troubleshoot problems by querying the database by STB or application ID
e) The ability to track OOB bandwidth usage by correlating STBs with OOB upstream and downstream nodes

COLLABORATIVE WORK WITH CANOE AND CABLE LABS

Canoe Ventures, founded by Brighthouse Networks, Cablevison, Charter, Comcast, Cox and Time-Warner was created to enhance the MSO's ability to collectively deliver iTV and advanced advertising to a national footprint. Both Canoe and Cable Labs have substantially aided our ability to deploy a scalable ecosystem for iTV.

Canoe has allowed Comcast to process more new applications by offloading some of our business development and application design / development / QA work for national bound applications onto their shoulders, where it is leveraged across many MSOs. The applications initially being developed by Canoe are templated, so they can be re-used across different networks with only minor changes to the app that do nto require new test and trial cycles. This has aided in minimizing onboading times as different networks strive to enhance their programming.

Canoe Ventures has also contributed substantially (along with the vendor community) to creating a more open, reliable and scalable EBIF broadcasting system for integration with a programmer's video broadcast systems. The Canoe architecture for national bound apps minimizes the MSO role in most operational aspects interfacing with the programmer and so streamlines the deployment process. With their MSO partners, Canoe has also collected a wealth of information regarding practical guidelines for application development across MSO user agents, and this will greatly speed development of next generation applications.

Not only has Cable Labs sponsored the forum through which the EBIF specs have been created, they have also recently sponsored the forum that has generated a set of "SaFI" specs [3-6] (Stewardship and Fulfillment Interfaces) for standard messaging between parties in an inter-MSO iTV application ecosystem. Those specs are at the heart of the Canoe inter-MSO application architecture. Cable Labs is also drafting an "ETV Operational Guidelines" specification [7] which aims to standardize some of the iTV platform aspects (such as those discussed in this paper) to make them more scalable and affordable industry-wide.

Some spec enhancements proposed in the most recent (I06) version of the EBIF specs stem from lessons learned from the MSO community during our past two years of deployment experience. One example of an I06 spec enhancements with direct relevance to our ability to operationally support a wide spectrum of EBIF apps is a method to authorize particular applications with permissions to use particular STB resources. As we deploy more apps on more networks, it is important to manage their impact on both STB and network resources, ensuring, for example, that iTV apps do not negatively impact each other or other non-EBIF applications like VOD, SDV and Guide. The "application permissions" proposal to the I06 EBIF spec would enable our platform to manage these resources more intelligently by providing a mechanism for controlled access to STB resources like VOD, DVR, return path, persistent storage, in-band and out-of-band bandwidth and app lifecycle execution privileges.

The test-flag implementation for supporting limiting app display onto a subset of STBs on plant is contrained due to only having 8 bits in the test-flag and the need for those 8 bits to be shared across all applications and all MSOs. Comcast is also implementing a whitelist/blacklist mechanism as a complementary mechanism to the test-flag, rendering test-flag as a boolean indicator for application test status, similar to how it is treated in OCAP. We expect that other MSOs will implement similar mechanisms that integrate with their OSS applications and processes.

Cable Labs has also recently supported vetting user agents submitted by MSOs against a broad array of test applications (including some they have developed solely for this purpose). We have found that resource to be useful as a research tool as new application functionality and new user agent revisions are developed.

SUMMARY

Comcast has deployed several EBIF-based iTV applications broadly throughout its footprint during the past two years. This paper has reviewed some of the fundamental aspects of our video delivery platform that we have enhanced to enable us to support these and many more future applications in a manner that will ensure an optimal customer experience.

One of our biggest "lessons learned" has been the need to robustly insert proactive monitoring tools in every advantageous network location, including direct application and user agent monitoring and non-realtime operational mining of data already being supplied for purely non-operational purposes.

Another design paradigm that we have learned must pervade our architecture and process is the need to leverage commonality across apps in order to scale them most efficiently. Rather than allow each app to splinter into its own design, we have tried to constrain them to common platform flows with common formatted messaging and common operational support tools and procedures.

Finally, one more key to scalability has been working with other MSO's collaboratively through Cable Labs and Canoe Ventures. The templated apps offered through Canoe Ventures is an ideal example of the above paradigm of leveraging commonality to optimize scalability.

REFERENCES

[1] OpenCable Enhanced TV Binary Interchange Format 1.0. (OC-SP-ETV-BIF1.0-I05-091125)

[2] Enhanced TV Application Messaging Protocol 1.0. (OC-SP-ETV-AM1.0-I04-070921.pdf)

[3] CableLabs Interactive Application Messaging Specification (IAM). AA-IAM-D03.doc.

[4] CableLabs Interactive Application Fulfillment Specification (IAF). AA-IAF-DO3.doc

[5] CableLabs Service Measurement Summary Information (SMSI). AA-SMSI-D03.doc.

[6] CableLabs Campaign Information Package (CIP). AA-CIP-IF-D03.doc.

[7] OpenCable Enhanced TV Operational Guidelines. OC-GL-ETV-OG-V01-060714.pdf.