

UPnP Forum AV:4 Overview



**Keith Miller/ Seung Yang
/Russell Berkoff
AV WC Co-Chairs/ Vice-Chair**

UPnP AV:4 New Features

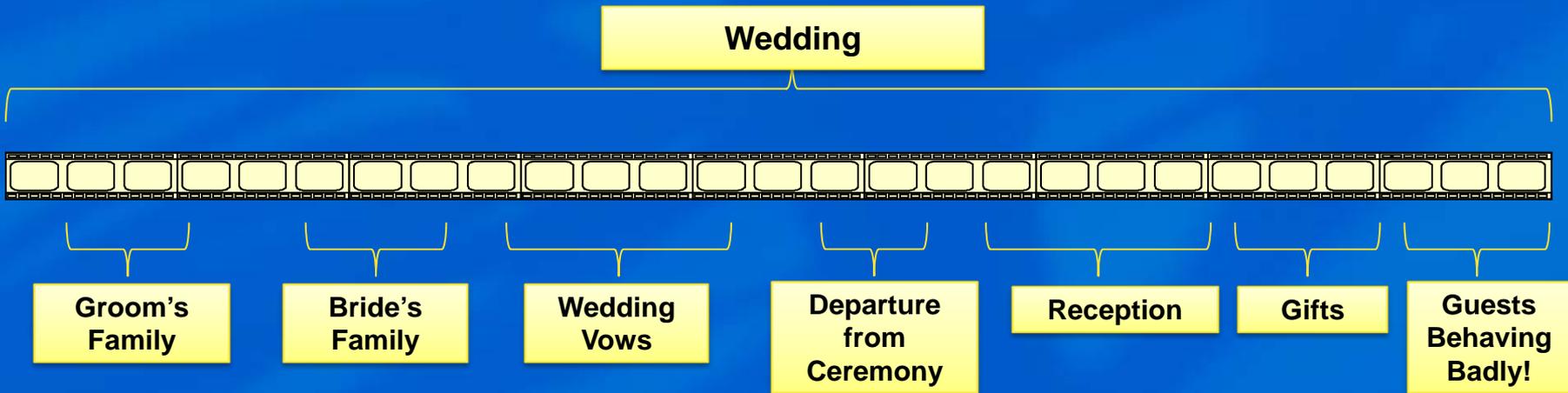
- “DVD like” Navigation of Content
- Richer Content Description (Multi-stream)
- Playback Synchronization (Multi-device)
- Device Resource Control
- Content Privacy
- Enhanced Playlist Support
- Instant Replay/Time Shift Support
- Renderer Content Matching (with DRM)
- Complex Metadata Filtering

Navigation - Content Segmentation

Navigation - Content Segmentation

- A segment item is a CDS item which references a portion of another CDS item's content.
 - Segments behave (mostly) like other CDS items.
 - No explicit segment classes
 - Lifetime is controlled by base content binary.
 - May be implemented as a separate media object or subset of the base content binary.
 - May be pre-authored, or CDS may support runtime creation of segment items.
 - Similar to a DVD chapter (except)
 - Segment items can be played in any container-based order.
 - Segment items can be explicitly ordered via CR-107.
 - Segment items can overlap.
 - Segment items can support multiple binary formats.

Navigation - Content Segmentation



Control Point Display



Navigation - Content Segmentation

Use Case:

- Carrie wants to organize her videos from her wedding. She has 10 hours of video footage. She wants to end-up with about 30 minutes of highlights.
- Carrie uses a video editing control point to create items containing the highlights from the wedding.
- Carrie determines the start and end-points for each highlight and chooses a title and description for each segment.
- The video editing control point requests the media server create a new segment item for each highlight using the information provided associated with the unedited video content.
- Carrie views and updates the segment items as required.

Navigation – Object Linking

Navigation - Object Linking

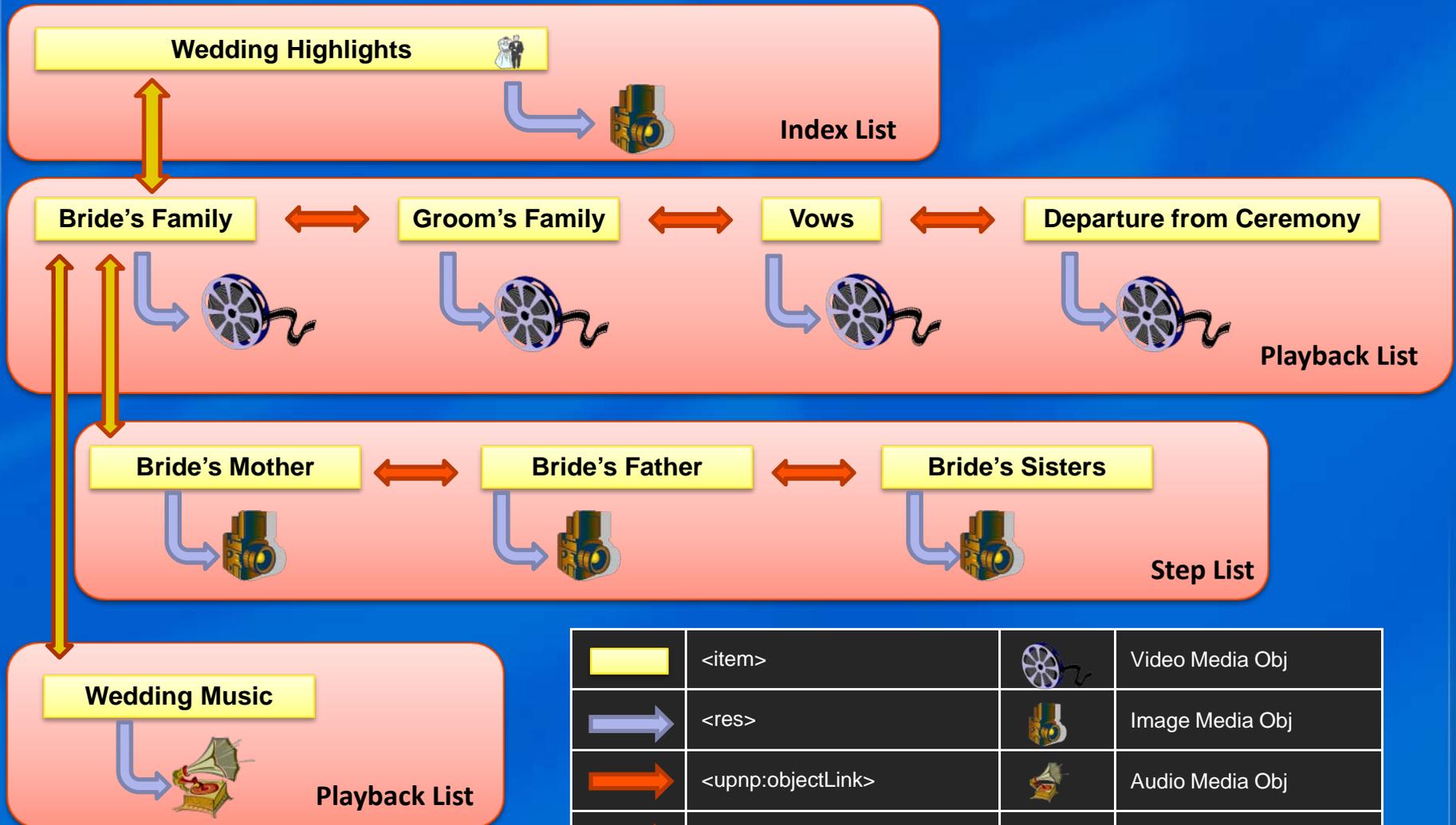
Allows content providers to:

- organize presentations consisting of many distinct CDS items.
- define lists of content items for playback. Lists may cross container boundaries and may have “mixed” classes of content. For example video and still images.
- define branch points within lists of content items. A branch point allows content in a different list to be played. The control point end-user can return to the point where a branch was taken.

Allows control points to:

- display lists of content description items (content monikers). A control point may be able to display text, audio, video, and image content moniker items based on the capabilities of the control point.
- navigate within a hierarchy of linked items. Once any item is located within a group of object linked items, the “top-level” (starting items) can be located.

Navigation - Object Linking



	<item>		Video Media Obj
	<res>		Image Media Obj
	<upnp:objectLink>		Audio Media Obj
	<upnp:objectLinkRef>		

Navigation - Object Linking

Use Case - Authoring

- Prof. John Q. Tenured wants to make his lectures available on the college's campus UPnP server.
- He has various materials including video of his lectures, still images and audio. Some of the material is supplementary and he doesn't need it presented automatically.
- Prof. Tenured tells his underpaid assistant Mr PostDoc to create an object linked presentation.
- The initial list is an index of Prof. Tenured's presentations by topic. In some cases a topic spans multiple lectures so Mr. PostDoc creates lists containing the related videos.
- In some cases the lectures have supplementary materials. Mr. PostDoc links lists containing this material to the lecture video lists as appropriate.
- Mr. PostDoc designates the index of the lectures as the starting item.

Navigation - Object Linking

Use Case – Rendering

- Joe Binge is a student of Prof. Tenured. He has lived up to his name and missed many of Prof. Tenured's classes and needs to prepare for his final exam. Joe is not very computer-literate.
- Joe stumbles to the campus library and sits down at a terminal allocated to Prof Tenured's course. He clicks on any item in Prof. Tenured's course.
- The control point detects an object linked item and asks if it should start from the beginning of the course. Joe is sober enough today to push the <Yes> button.
- The CP locates the beginning of the presentation. The CP determines the presentation starts with an index list and displays the content monikers contained in the list. Joe clicks on the course topics he has missed.
- Joe also uses the supplementary materials within each lecture. The CP indicates these are available by displaying a hand-icon at the bottom of the screen. Joe clicks on this and the CP switches to a list containing still photos of the whiteboard.
- Joe presses return and the control point returns to the topic index. Joe then selects a different topic.

Richer Content Description

Richer Content Description

- Currently a ContentDirectory Service can only describe items that are flavors of the same content. There is no descriptive mechanism available to expose augmented content (selectable components) that can be used simultaneously to enrich the user experience.
- The new feature allows description of multiple, selectable components (multi-streams) complementary to the main item. The complementary items may be in a composite or separate streams.
- A complementing MediaRenderer feature allows customized consumption of the enriched content .

Richer Content Description

Use Cases:

- **Indicating in a CDS**
 - a list of subtitles in an mpeg-TS stream
 - a list of Audio tracks in an mpeg-TS stream
 - a list of alternative videos (Camera Angles) in an mpeg-TS stream
 - a separate subtitle file for a video (SRT file)
- **Selection on a Renderer**
 - a subtitle (close captioning)
 - an audio (language track)
 - A camera angle

Richer Content Description



DTV
Media Renderer



Including subtitling,
alternate audio tracks



Home Media Server

Select Item for playback

Play()

Control Point lists
choices exposed
by renderer for
multi-stream
content



Subtitle Language:	English
Audio Language:	Dutch
Camera Angle:	Main
Zoom:	0%
Playspeed:	Normal

Precision Time-Synchronization

Precision Time-Synchronization

- In certain situations, end-users may desire precision time-synchronized rendering of AV content by several rendering devices.
- Link-layer protocols such as IEEE 802.1 AS are available to provide such capabilities for device network interfaces to operate in Precision Time-Synchronized (PTS) manner.
- This new AV PTS API provides the necessary UPnP-level mechanisms for control points to take advantage of devices' PTS capabilities for simultaneous rendering.

Precision Time-Synchronization

The UPnP AV PTS API provides capability for control points to:

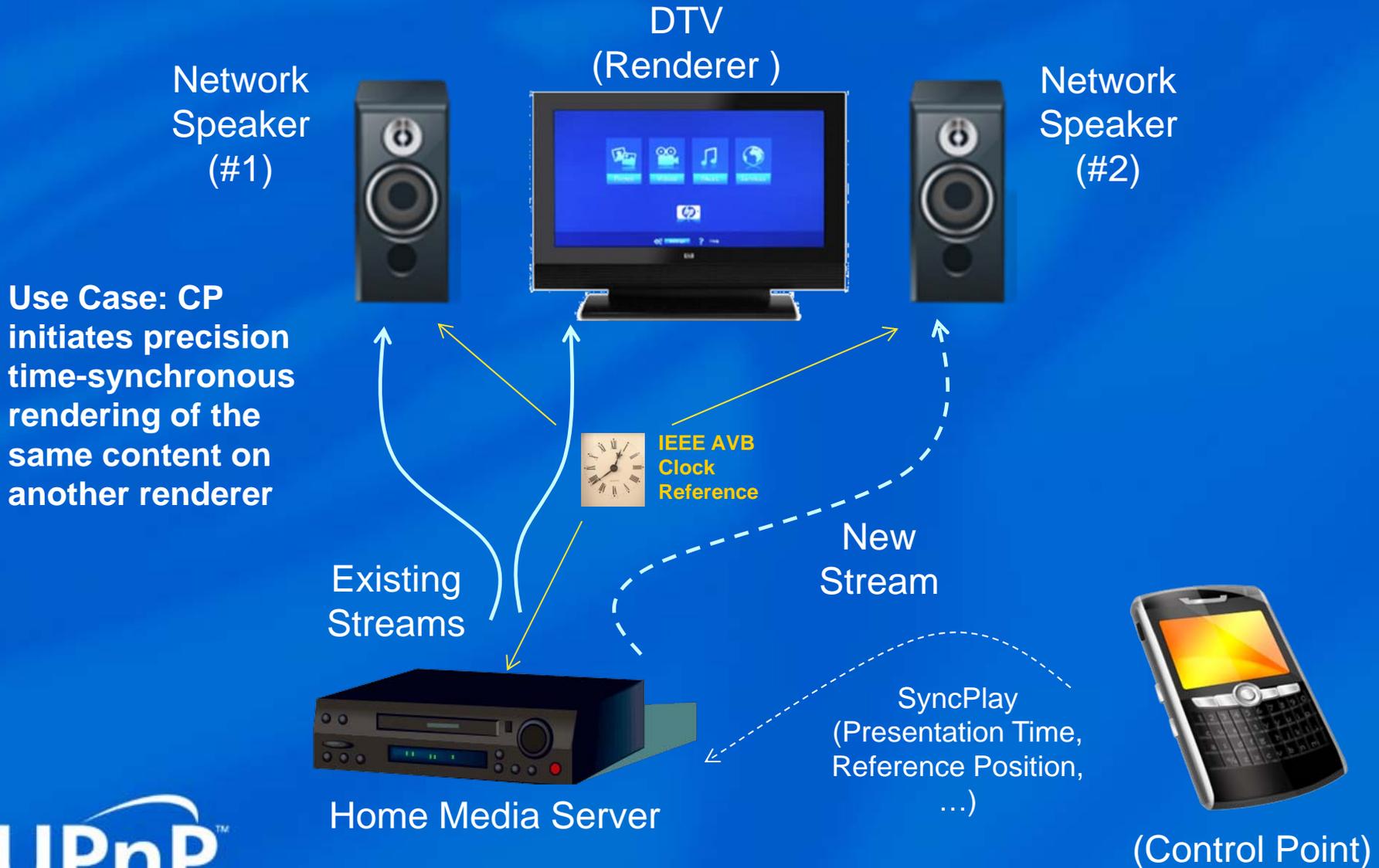
- detect whether a UPnP AV device participates in a network that supports PTS.
- get, set, and adjust the time offset for the PTS capable device,
- defines new actions that can initiate PTS play, pause, and stop of AV transport (including HTTP, RTP, etc).

Precision Time-Synchronization

Use Cases:

- Initiating a second PTS rendering of the same content that is currently playing on another renderer
- Whole-home audio playback wherein the same audio content is rendered (in a PTS manner) by several rendering devices
- The video and audio portions of the AV content are rendered by separate rendering devices in a PTS manner

Precision Time-Synchronization



Precision Time-Synchronization

Network
Speaker
(Bedroom #1)



Network
Speaker
(Bedroom #2)



Network
Speaker
(Kitchen #1)



Network
Speaker
(Kitchen #2)



IEEE AVB
Clock
Reference

Use Case: CP initiates
whole-home audio
playback (same audio
content across several
speakers)

4X
SyncPlay(Presentation Time,
Reference Position, ...)



AV Receiver



(Control Point)

Precision Time-Synchronization

DTV

(Renderer)

Network
Speaker
(#1)



Network
Speaker
(#2)



IEEE AVB
Clock
Reference



Use Case: CP
initiates time-
synchronous video
and audio playback

(Audio)
SyncPlay(Presentation
Time; Reference Position,
...)



Home Media Server

(Video)
SyncPlay(Presentation
Time, Reference
Position, ...)



(Control Point)

Device Resource Control

Device Resource Control

Provides an API for configuring a CDS (and possibly other devices or services) for:

- Managing intense sequential Actions “ActionBurst”
- Reserving particular resources “resourceID” such as a tuner or complete device.

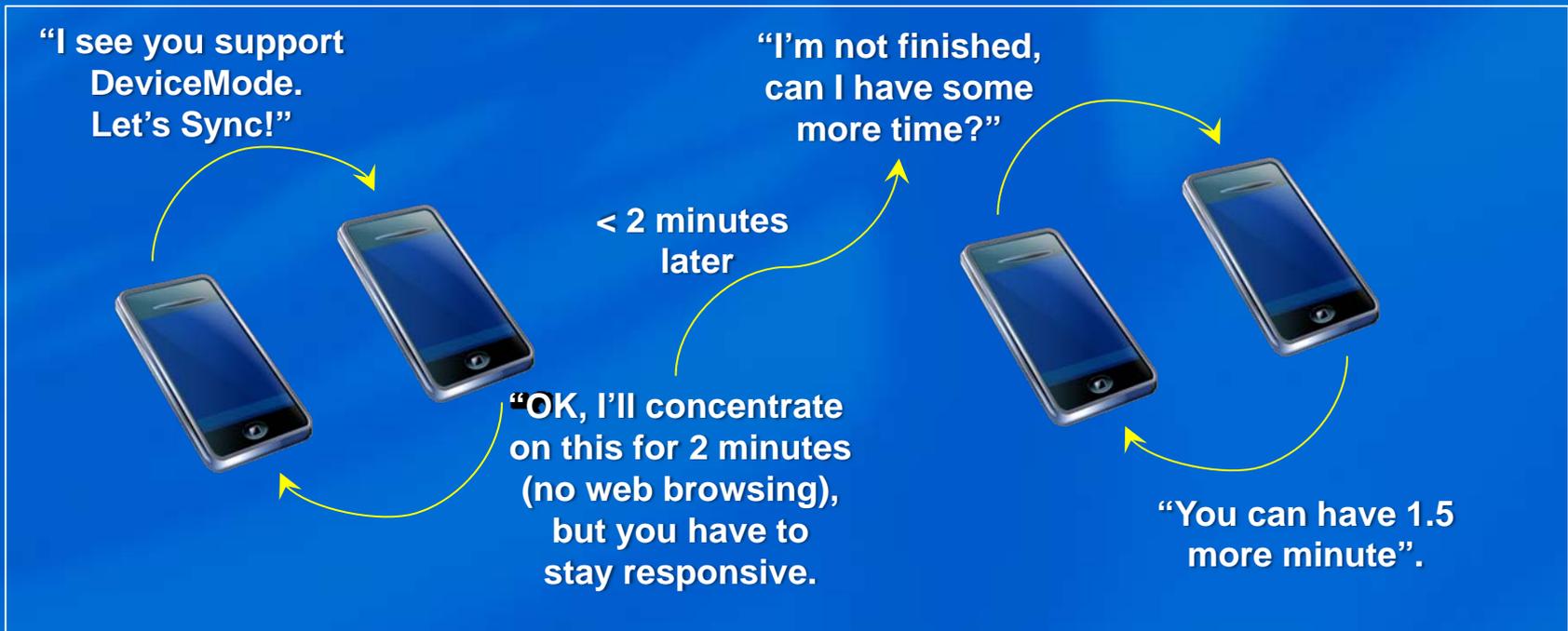
Current device resource model assumes:

- “friendly” control point behavior
- token passing to/from control point controlling the device “Mode”
- Time outs for “sluggish” control points

With DeviceProtection this feature can be extended to exclude “non-friendly” control point behavior.

Device Resource Control

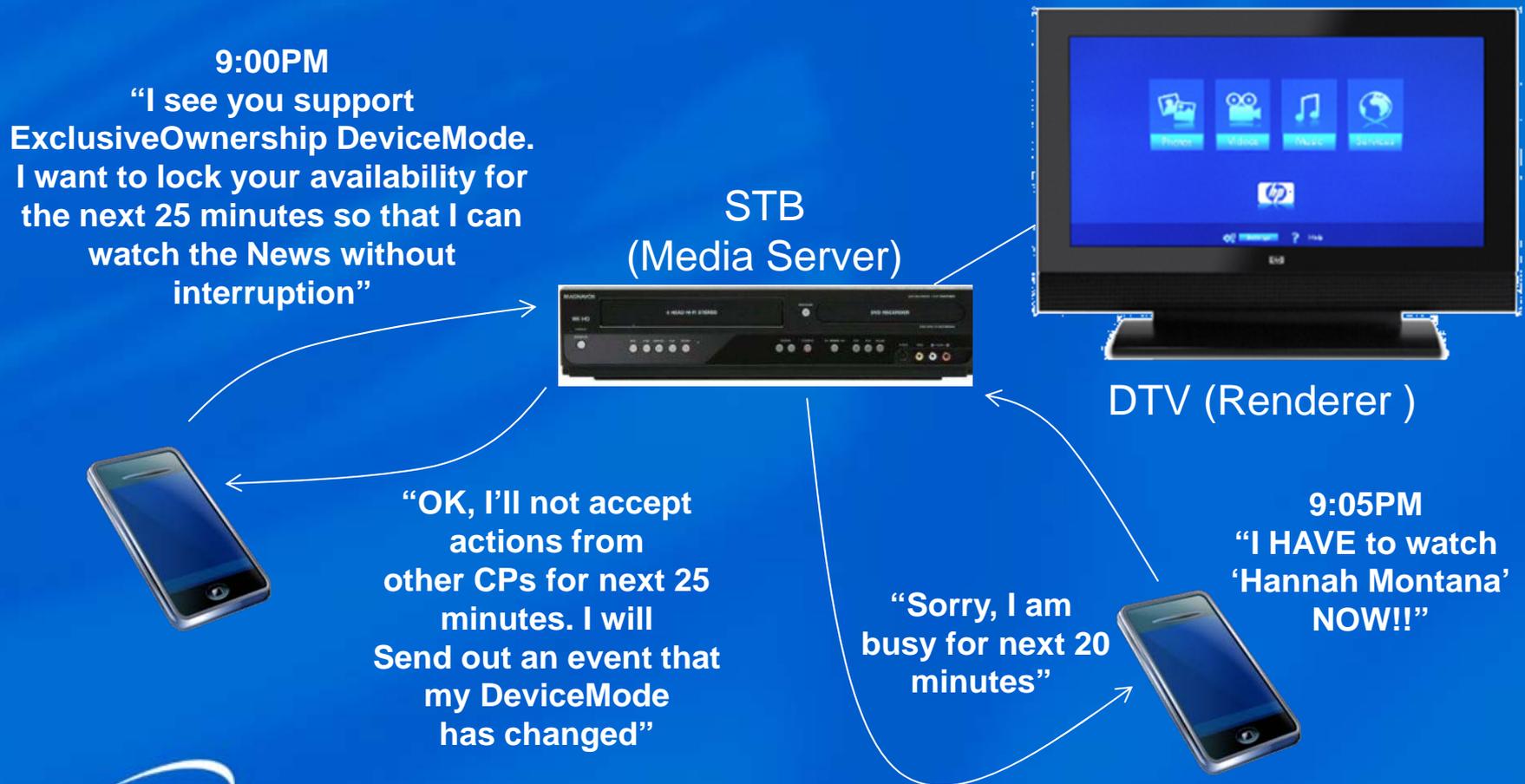
“ActionBurst” Paraphrased



Option to support verbose description of actions

Device Resource Control

“Exclusive Ownership” Paraphrased



Content Privacy

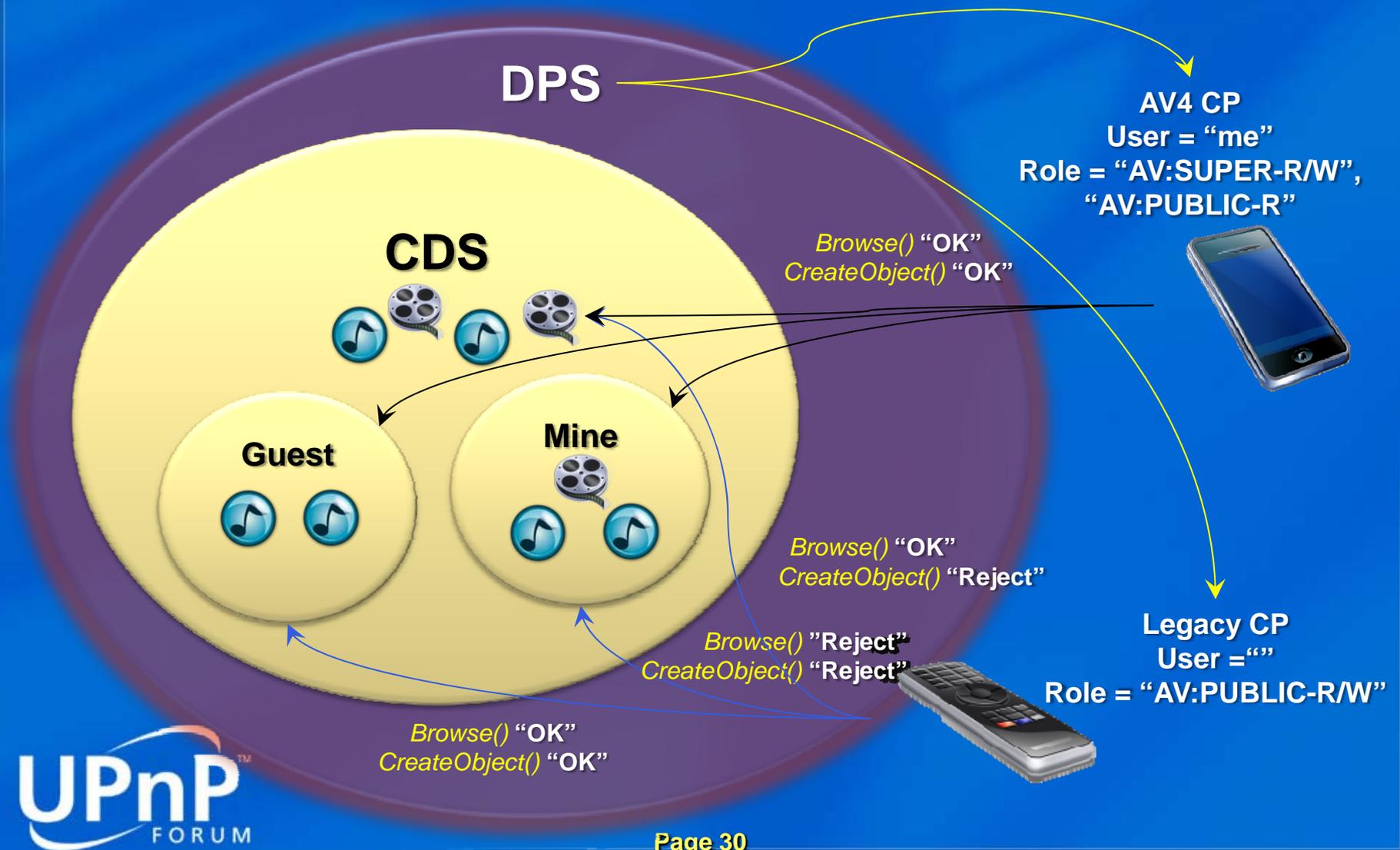
Content Privacy

- Currently ContentDirectory Service (CDS) content (items and containers) are accessible by “All” control points (CPs) with “All” actions.
- Complementary with the new DeviceProtection service (DPS), AV4 extends the access control granularity to the action level and the object level (items and containers):
- Object level access is controlled through two new properties: *inclusionControl* and *objectOwner* .

Content Privacy

- DPS controls access through *Roles*. - *CPIDs* (hardware) and *Users* (logged on) - can have default *Roles* and acquire additional *Roles*.
- AV:4 defines 6 fixed *AV:Roles* partitioned along the concepts of:
 - *Read / Write* access, i.e., *Browse()*, *Search()* / *CreateObject()*, *UpdateObject()*
 - Guest (Legacy), Normal, and Super Users
- AV specific vendor roles can also be defined, such as: *vendor.com:Child*, *vendor.com:Parent*
- Additional *Roles* are defined and managed out-side of DPS and CDS, however, assignment of *Roles* to *CPIDs* and *Users* are part of DPS.

Content Privacy

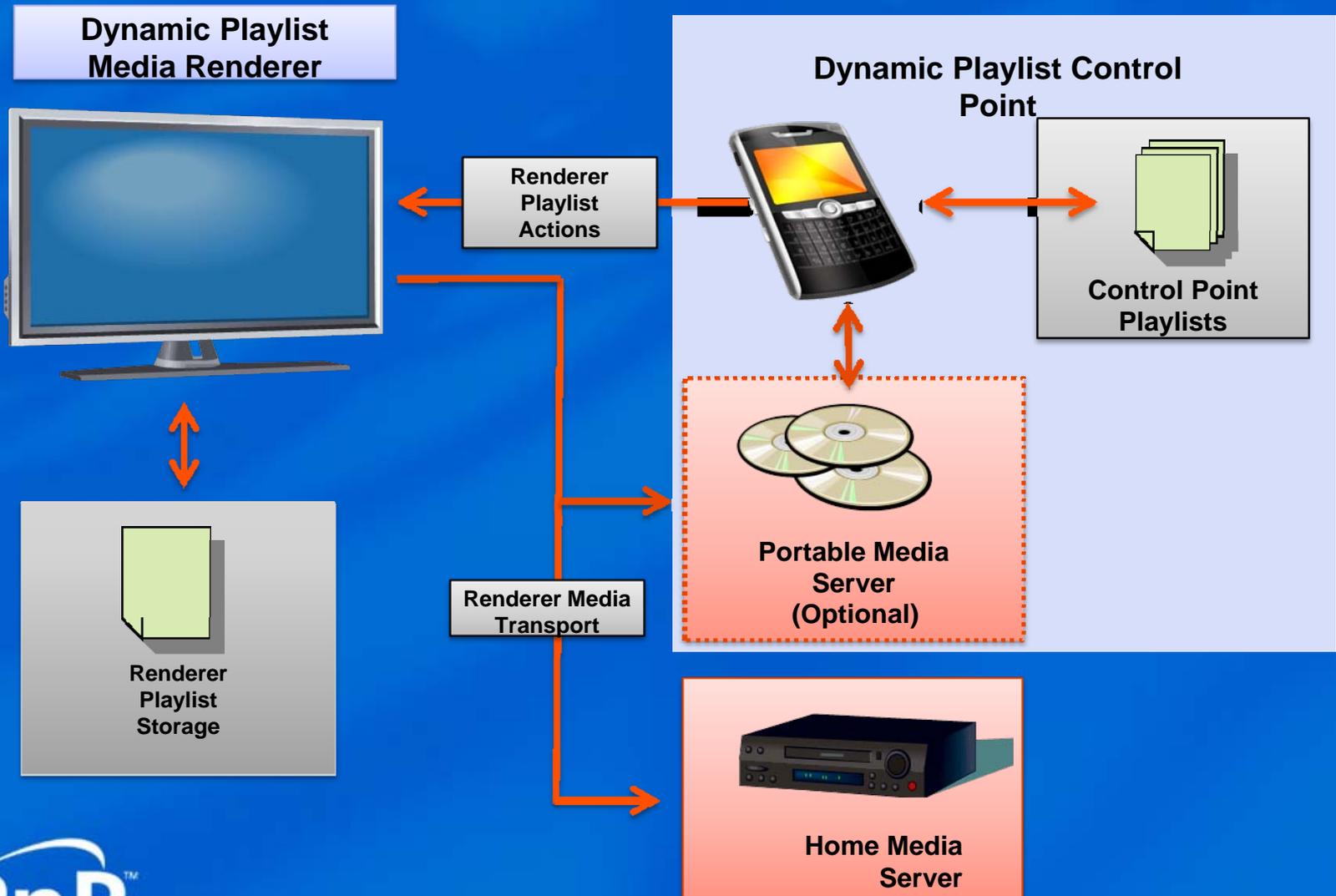


Enhanced Playlist Support

Enhanced Playlist Support

- Allows Control Points to create “private” playlists without first having to store playlist(s) on a ContentDirectory Service.
- Supports submission of “infinite” length playlists (m3u).
- Supports submission of DIDL_S, DIDL_V (XML) playlists and *ObjectLinked* items.

Enhanced Playlist Support



Enhanced Playlist Support

● Use Cases

- A Golfer has videos from his trip(s) to Las Vegas.
- He's in a hotel and wants to watch the videos. He does not want the contents of his Cell-Phone/DMS shared with other guests at the hotel.
- Our golfer can have his Cell-Phone “push” a dynamic playlist to the network DTV in his room.
- The DTV processes the playlist rendering the content stored on his Cell-Phone.

- Joe has a 1TB NAS network drive at home filled with music.
- Joe frequently creates and edits music selection lists on his Cell-Phone/CP. He does not want to take the time or effort to manage permanent playlists on his network drive.
- Joe can have his Cell-Phone/CP “push” a dynamic playlist directly to his networked home audio system.

Instant Time-Shift/Playback Support

Instant Time-Shift/Playback Support

- Currently a recording device can have a time shift buffer which is not described for UPnP.
- This new feature can record programs, wholly or partially stored in the time shift buffer.
- By leveraging existing CDS/EPG and SRS mechanisms with additional time shift buffer information.

Instant Time-Shift/Playback Support



**MediaRenderer
+
Control Point**

**Watch Tuner from
Media server**



Record program



**Time Shift Buffer
for Tuner**

Home MediaServer

- CDS/EPG
- SRS
- Time Shift Buffer
- TSB Support

- Contains start = Y/N
- In progress = Y/N
- Complete = Y/N

Renderer Content Matching (w/DRM)

Renderer Content Matching (with DRM)

- Allows control points to submit one or more metadata items to a Media Renderer for analysis. Media Renderer responds with detailed information about each item.
- Analysis may include:
 - Playback information (required)
 - Is media format playable?
 - License information (optional)
 - License status (valid, expired)
 - License counters (playback time/counts remaining)
 - License rights (stream, copy)
 - Conversion information (optional)
 - Input/Output resolutions
 - Audio conversions
 - Component information (optional)
 - Selectable components in multi-stream items

Renderer Content Matching (with DRM)

Use Case - 1

- An end-user wants to determine how much time is remaining on his network video rental.
- The end-user selects the item. The control point sends the item metadata to the renderer.
- The renderer determines that the item is DRM protected. The renderer requests its DRM agent to provide license information about the item.
- The renderer returns the license information to the requesting control point.
- The renderer indicates whether the user can play the item and how soon their rental will expire.

Renderer Content Matching (with DRM)

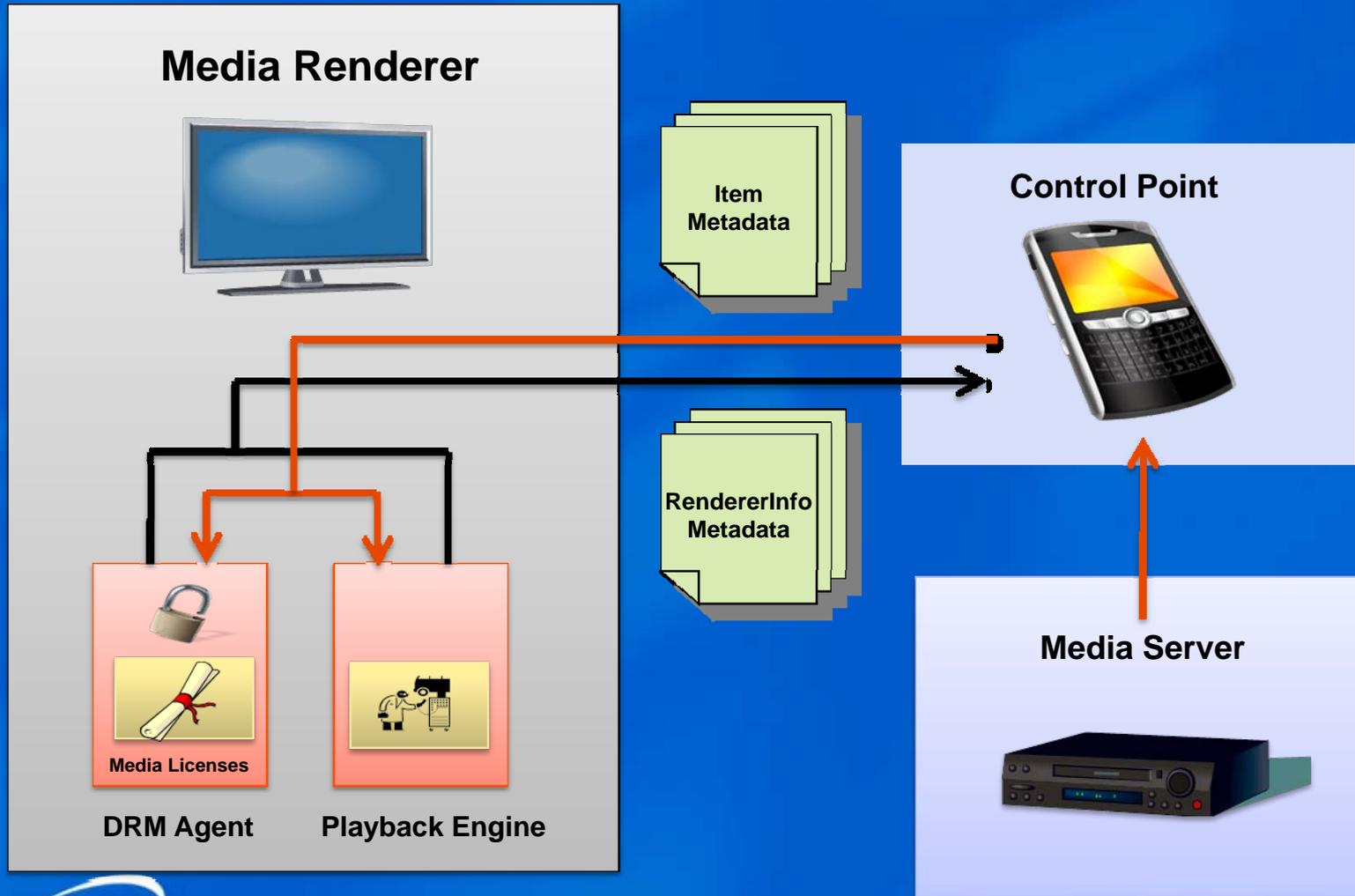
Use Case - 2

- An end-user wants only display items from his library that will provide a “high-definition” playback experience.
- The control point sends metadata for the item(s) to the renderer.
- The renderer returns resolution information for the content binaries in each playback item.
- The control point only displays items that will provide the required playback experience.

Use Case – 3

- An end-user wants to determine the playback options for a multi-language item.
- The control point sends the item metadata to the renderer.
- The renderer returns the selectable playback options for the multi-language item.

Renderer Content Matching (with DRM)



Metadata Filter Enhancement

Metadata Filter Enhancement

Metadata Filter Enhancement defines:

- New Metadata Filter operator (#) to include all supported dependent properties associated with a property.
- Clarifies usage of “::” notation in Metadata Filter strings.

Metadata Filter Enhancement allows:

- Less complex Metadata Filter arguments.
- Ability for control point to request all dependent properties without fully specifying each property name in the Metadata Filter argument.



Foundation of the Connected Home