EBU

OPERATING EUROVISION AND EURORADIO

TECH 3369

REQUIREMENTS FOR NETWORKED DEVICE MANAGEMENT (NDM)

SOURCE: FNS

Geneva
September 2015

[*] Page intentionally left blank. This document is paginated for two sided printing

Contents

Scope 5

1. Introduction 7
   1.1 Scenarios 7

2. Definitions 7

3. Functional requirements 9
   3.1 Identification 9
   3.2 Registration 9
   3.3 Query 9
   3.4 Configuration and Control 9
   3.5 Connection Management 10
   3.6 Monitoring 10
   3.7 Access 11

4. Structural requirements 11
   4.1 Fixed, dynamic, physical, virtual Devices 11
   4.2 Temporality 11
   4.3 Scale 11
   4.4 Separation of traffic 12
   4.5 Registration Topology 12
   4.6 Zero-configuration networking 14
   4.7 Reliability 14

5. Performance requirements 14
   5.1 Set up, configuration, tear-down 14
   5.2 Control operations 14
   5.3 Monitoring and alerting 14
   5.4 Bandwidth constraints 15

6. Security requirements 15
   6.1 Authentication 15
   6.2 Encryption 15
   6.3 Access control 15

7. Interoperability requirements 16
   7.1 Identification and versioning of Device Management Interfaces 16
   7.2 Standards 16
   7.3 Documentation 17
   7.4 Legacy control protocols 17
   7.5 Platforms 17
   7.6 Protocols 17
   7.7 Adoption 17

Annex 1: Scenario 1 - Networked live production with remote elements (BBC) 19
Type of production 19
Production locations 19
Acquisition 19
Live production operations 19
Time-shifted operations and editing 20
"Factory" operations 20
Content monitoring 20
Production communications 20
Networked Devices 20
Locations 21
Set-up 21
Monitoring & diagnostics 21

Annex 2: Scenario 2 - Local Sport (BBC) 23
Background 23
Remote Site 23
Studio Facilities 23
Online production 23

Annex 3: Scenario 3 – Large, full IP indoor audio production (R-F) 25
Overview 25
Current configuration 25
Changes for outdoor production 25
Devices 25
Possible changes for IP architecture 25
Still to be defined 25
Diagram 26

Annex 4: Scenario 4 – Radio shows (SR) 27
Scenario 4a: Typical show 27
Scenario 4b: Morning show 27
Scenario 4c: News show 27
Scenario 4d: Sports show 27

Annex 5: Examples of Devices 29

Scope

This work concerns the identification of functional and non-functional requirements for the management of network-connected Devices.

Examples of Devices include: encoders/decoders/transcoders, cameras, microphones, stream transmitters/receivers, media storage devices, network-based vision/audio mixers, processors, foreign system interfaces, clock generators and monitoring systems.

As well as physical Devices, "virtual" Devices running on general-purpose computer platforms (for example transcoders running within a cluster or a cloud) are in scope.

No particular network architectures are assumed, so for example, management of AVB Devices and IP-based Devices are both in scope, as is management of both local Devices on a small network, and management across multiple connected (sub-) networks.

Management of the connections (Flows) between devices is in scope but management of the network itself, including bandwidth reservation etc., is out of scope. Network switches and routers are not considered as Devices within the scope of this work.

Acknowledgement

This EBU Tech Doc was prepared by a task group within the Strategic Programme on Future Networked Systems (FNS). The group was chaired by Peter Brightwell (BBC) with key contributions from: Madeleine Keltsch (IRT), Fredrik Bergholtz (SR), Francois Ragenard (RF), Simon Hyde (BBC) and Martin Jacober (SRG). Comments were also received from several members of the broadcast industry, and these were greatly appreciated.

EBU staff: Félix Poulin (Project Manager) and Roger Miles (proofing and layout).

[*] Page intentionally left blank. This document is paginated for two sided printing

Requirements for Networked Device Management (NDM)

EBU Committee	*First Issued*	*Revised*	*Re-issued*
FNS	2015		

Keywords: Networked Device management, NDM, AVB, IP.

1. Introduction

The transition to IT/network-based infrastructures for production and contribution promises much for EBU members, but realising the potential benefits will depend on how easily the infrastructure can be managed. To prevent proliferation of multiple incompatible (and possibly proprietary) approaches, which can lead to overheads in system integration and lack of flexibility, it is important that end-users are clear on their requirements. This will help FNS identify suitable architectural approach(es) and technologies.

In early 2015 the EBU's FNS group created a small task group to identify the requirements for Networked Device Management. This document, which was originally created as a wiki page, summarises these requirements.

1.1 Scenarios

The following scenarios were contributed to the group as input to the work:

- BBC: Networked live production with remote elements, Local Sport
- Radio-France: Big full IP indoor audio production
- Swedish Radio: Radio shows: Typical, Morning, News, Sports

Each of these scenarios is provided in an Annex to this document

2. Definitions

To avoid confusion, the following (capitalised) terms are defined within the context of these requirements:

Bundle: A grouping of related Entities for management purposes. For example multiple Devices streaming tiles of a panoramic scene may be bundled together, as may their corresponding Sources, Senders and Flows.

Capability: Provides information about “usefully minimal” item of functionality provided by a Device, e.g. the ability to decode a particular video format, the ability to receive streams up to a certain bit rate, or the ability to apply an audio level adjustment.

Client: An application that accesses a Device Management Interface to manage Devices and other Entities. Might offer a user interface or might be purely automated.

Device: A network-managed element that provides a useful function as part of a signal/media workflow. A Device could be a dedicated item of hardware, but could be provided by more generic compute/network/storage facilities. Also see § 4.4 on real, virtual or composite Devices. Annex 5 gives some examples of Devices.

Device Management: Collective term for registration, configuration, control and monitoring of Devices.

Device Management Interface: Exposes some aspect of Device Management to a Client via a network.

Entity: A "thing" that needs to be identified, discovered and accessed for Device Management. Capabilities, Devices, Sources, Flows, Senders and Receivers are Entities. Other Entities may also be required, e.g. SIP accounts.

Flow: A logical movement of video, audio, or other essence or data into and/or out of a Device. Flows can be realised through live network streams, file movement, or legacy transports such as SDI.

Identification: An unambiguous way of referring to an Entity.

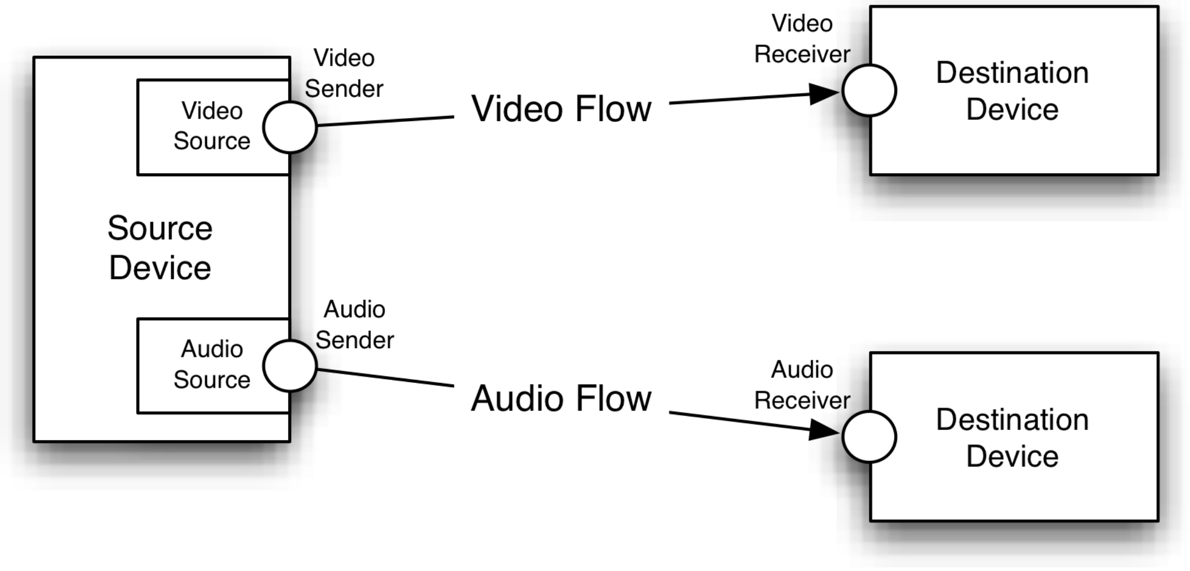
Query: Finding information about a Device or other Entity based on its Identification or other properties.

Receiver: An interface on a Device that consumes a Flow.

Registration: Making a Device or other Entity available to be found.

Sender: An interface on a Device that produces a Flow.

Source: The logical origin of a Flow or set of Flows. A Device may have Sources, possibly more than one; for example a video camera may have separate video and audio Sources (see figure below).



3. Functional requirements

3.1 Identification

A networked management solution should support unique Identification of any Entity:

- within a deployment
- globally, to allow for wide-area scalability.

Both long-lived (e.g. the permanent id used for a camera Device) and short-lived identification (e.g. for a network Flow during a short-term production) may be required.

3.2 Registration

It should be possible to register Devices and other Entities so that they can be found and accessed by Clients.

Manual Registration and de-registration should be possible.

Automatic Registration and de-registration, e.g. using a Discovery mechanism, should be possible, so that a user can start working immediately without having to configure routing. See also "Registration Topology" below.

In some cases, both manual and automatic Registration will be used; for example a set of Devices in an apparatus room that has a fairly static arrangement might be manually configured and registered locally, but could appear dynamically as a set of Sources in a remote gallery

When a Device is powered down or disconnected, its Entities should be (optionally) de-registered. When the same Device is powered back on or re-connected it should (optionally) be possible to register it as before.

3.3 Query

It should be possible to find a particular Entity (Device etc.) based on:

- its unique Identification
- human-readable identifiers, which may be hierarchical, for example allowing a Device to be identified on a geographical basis, or on its function e.g. a Source might be referred to as "Presenter mic 1".

It should be possible to make a Query based on the Capabilities supported by Devices, such as: "*Find all Devices that support live capture, AVC-Intra encoding at up to 100 Mb/s and RTP streams.*"

It should be possible to filter the Entities that are presented to end-users or Clients, for example to present only local Devices, or only audio Devices.

3.4 Configuration and Control

Device Management Interfaces should provide a consistent approach to configuration and control across different types of Device. Examples include:

- Connection management between Devices (see below)

- Setting and getting operational parameter values (e.g. encoding bit rate)
- Starting, stopping and resetting operation of Devices
- Setting and getting Device management parameters (e.g. network connection parameters, Device Identification & version information, software/firmware/hardware revision numbers)
- Managing and changing Device firmware

It should be possible to support "templated" groups of controllable features, for example to set up an agreed coding profile, or mixer "snapshot".

Because time at a studio or a venue is often limited, It should be possible to prepare Device configurations in advance of a production, and to be possible to load it onto the Devices both: (a) beforehand, on a different network, and: (b) on the actual production network.

Some use cases will require control operations to occur at a particular time, for example a video switch on a particular frame, and/or over a particular time range, e.g. for a cross-fade.

3.5 Connection Management

Device Management Interfaces should allow connection of Flows between Senders and Receivers on the Devices.

Connection management using non-networked transports (e.g. SDI) should be supported, as well as Devices using network-based transports.

Solutions should support complex connection scenarios that are common in broadcasting. For example an operator in a Master Control Room may need to make a temporary connection of a Sender in an OB vehicle to a Receiver in a studio.

Often multiple similar operations will be required on each element of a Bundle. For example a Bundle of Receivers might be connected to a Bundle of Senders coming from a remote studio.

3.6 Monitoring

Devices should provide notification of:

- Discrete events such as:
  - Power on/shutdown
  - Failures
  - Configuration changes
  - Operating parameter changes
  - Flow set-up
  - Flow tear-down
  - SIP account Registration
  - AV signal conditions, e.g. silence or black

- Continuous changes such as:
  - Level meters
  - Flow throughput
  - Error rates

It should be possible configure which events and changes Devices notify, and which Clients monitor.

It should be possible to add new events and changes according to use case.

Clients should be able to detect when a Device cannot be contacted e.g. due to power or connectivity loss

Notification should be out-of-band, i.e.:

- asynchronous of control operations (no polling required), and
- not transported within Flows

3.7 Access

Identification, Registration, Query, control and monitoring operations may be invoked by:

- users via a web browser
- users via a dedicated application, possibly with customised hardware interfaces e.g. push buttons
- an external automation system

User interfaces should update automatically to show the availability of newly registered Entities, including where manual Registration is used.

User interfaces should update automatically to show that a de-registered Entity is no longer available, including where manual de-registration is used. Depending on the Entity and use case, this may also require the Entity to be removed from the user interface.

4. Structural requirements

4.1 Fixed, dynamic, physical, virtual Devices

It will be necessary to manage Devices that have either a **fixed** function (e.g. a camera), or a **dynamic** function that is configured as required, typically as processes on a computer node (e.g. a transcoder).

Devices may be **physical**, or be **virtual**, i.e. they are created on demand using virtualisation or container technology.

4.2 Temporality

It should be possible for Entities to be:

- Permanent: such as for a broadcast facility.
- Temporary: quick and easy to set up, configure and tear down for temporary production sites, or "factory" (on-demand) operations.

4.3 Scale

Device Management Interfaces should be suitable for use with both small-scale deployment (a few Devices in a single LAN) and large-scale deployment (hundreds of Devices and Streams in a wide area, routed, network). It should not be required to reconfigure the Device purely on the scale of deployment.

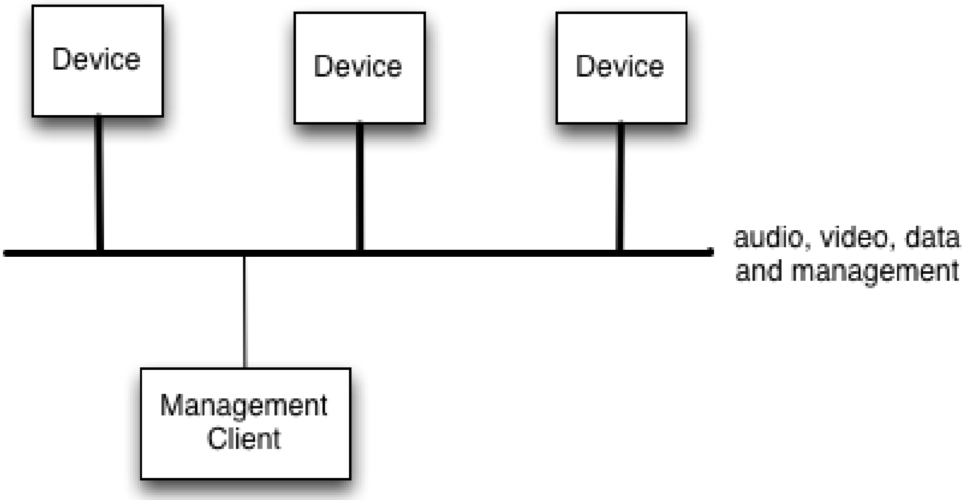
Device Management messages must be routable across different network segments (whether or not the AV Streams themselves are routed)

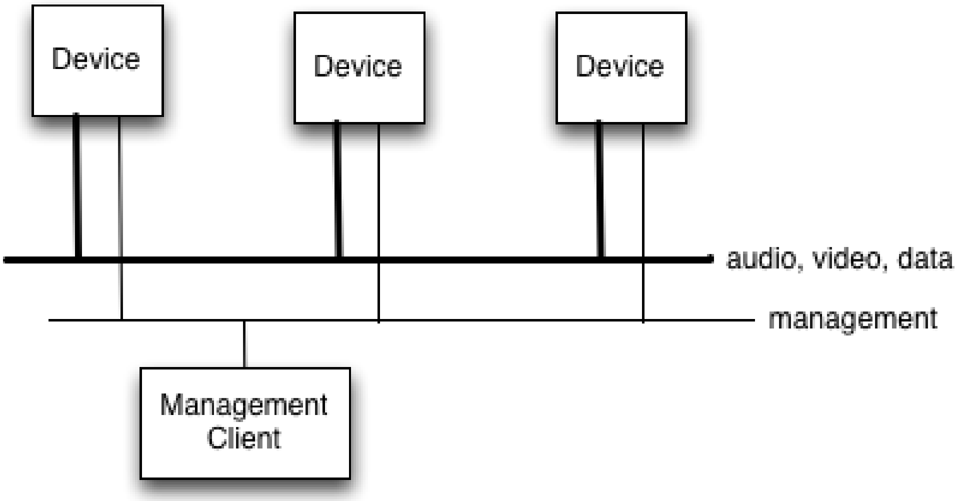
It should be possible for small and large deployments to communicate with traffic separated so that the small deployment is not "swamped" by Registration and control information from the large deployment.

Clients should only receive information about relevant Devices (see also filtering of Queries)

4.4 Separation of traffic

Device Management Interfaces should support operation on (a) a network that is shared with AV and data traffic, (b) a separate network:





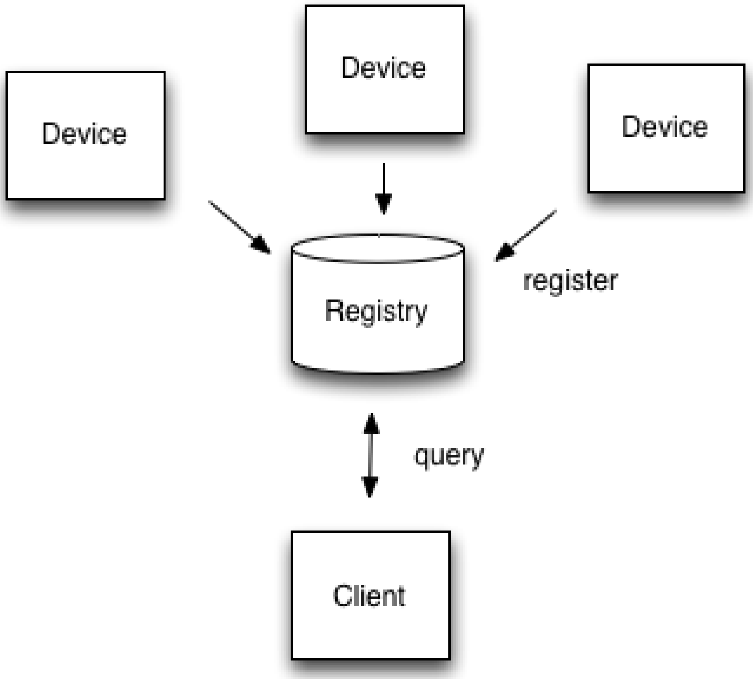
4.5 Registration Topology

To support a wide range of scenarios, several approaches to routing of Registration/discovery information are possible:

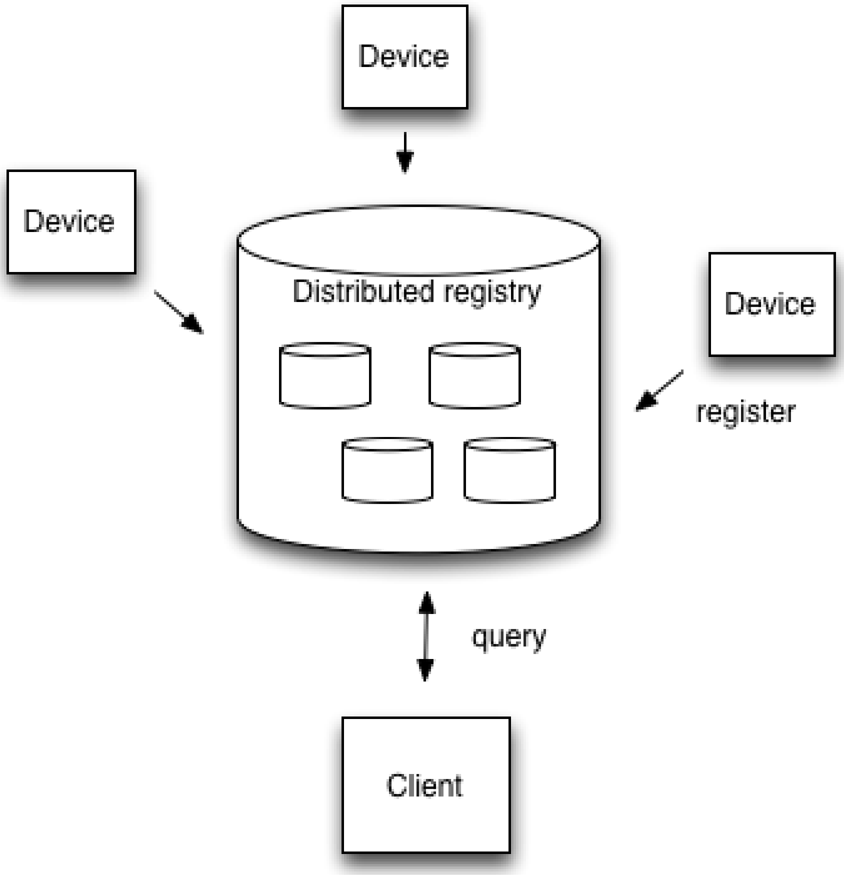
- Manually configured routes.
- Peer-to-peer discovery between Devices. Suitable for small installations via a single cable or

networking switch.

- Each Device registers its Entities with a registry that is queried by Clients (see figure).



It should be possible for the registry to be distributed according to operational needs.



This also allows an architecture where the Devices themselves host the distributed registry.

4.6 Zero-configuration networking

A solution should be able to operate effectively in environments using "zero-configuration" networking. Examples include the use of link-local IP address ranges and ".local" DNS domains to allow Devices to access the network without any manual intervention. Peer-to-peer discovery is likely to be most suitable in such cases.

4.7 Reliability

A solution should allow typical mechanisms to aid reliability to be used, for example multiple redundant management servers.

5. Performance requirements

Different scenarios have different performance requirements so specific delays, etc. are not specified here. In general, management solutions should be fast enough not to:

- disrupt automated operation
- give users the impression of lag

A solution should allow typical mechanisms to aid performance to be used, for example load balancing of management servers.

5.1 Set up, configuration, tear-down

A Device should be available for control from the point at which is operational.

In a practical deployment, many Devices may start up, each at its own pace. It is important that each Device and its Entities can be managed without waiting for everything else to be available.

5.2 Control operations

Many scenarios will demand sub-second response time for a management operation to take effect. For example an audio level change should happen quickly.

Even where such response time is not necessary, it should be possible to provide rapid operator feedback that an operation has been requested.

5.3 Monitoring and alerting

It should be possible to configure response times for presentation of information and alerts, and taking automated action. The actual times are likely to vary significantly between scenarios, so the following are just examples:

- An operator may want to receive an update about the state of a Device within a few seconds
- Users will probably need to be informed of a missing Source within a few hundred milliseconds
- A remote web service (e.g. database Query) may be accessed less frequently and a longer delay may be acceptable

It should be possible to invoke an automated action at a configurable time after an event, for example:

- disconnect a missing source
- de-register a missing Device or other Entity

5.4 Bandwidth constraints

Networked Device management should not add unreasonable bandwidth demands of the network, and in cases where it shares a network with media traffic should not impact the latter.

Solutions should not be too "chatty"; for example notifications should not be sent more frequently than is necessary for business needs, or for changes that a Client is not concerned with.

6. Security requirements

In general, security should be considered as a requirement when a control protocol is designed.

6.1 Authentication

It must be possible to securely verify the authenticity of:

- any registry from a registering Device or Client
- a Device or Client from any registry
- a Device from a Client
- a Client from a Device
- Sources, Senders, Receivers, content Flows, and control Flows should also be considered.

Authentication should be scalable from small ad-hoc scenarios to large enterprise operations. Simple configuration is desirable, defaulting to a secure mode that allows simple setup, but in such as a way that security is not compromised. It must not be possible to automatically disable authentication to ease initial configuration.

6.2 Encryption

Industry standard, provably secure encryption should be available as an option on all Devices and Clients.

6.3 Access control

A system must offer basic authentication to prevent unauthorised access (see Authentication).

A system may choose to offer granulated access control to Device control, to monitoring, to content Flows

- rules-based (e.g. according to location, role)
- restrict discovery.

7. Interoperability requirements

One approach to Networked Device Management is to "hide" multiple vendors' Interfaces from the customer through specific integrations. This adds to the overall cost of development and deployment however, so a common approach to Device Management Interfaces is desirable. This would document mechanisms for:

- Registering and discovering Entities
- Accessing interface endpoints
- Representing Capabilities
- Representing configuration and control parameters
- Representing monitoring information
- Transport protocol(s)
- Status and error reporting
- "Patterns" for synchronous and asynchronous operations
- Identifying the Device Management Interface itself

However, the FNS-NDM group recognises that a real-world deployment is likely to make use of multiple different technologies, and that users may need to consider the implication of managing a heterogeneous environment. For example:

- A production might have different registries for its video infrastructure and its SIP-based communication infrastructure.

- It may be necessary to connect external Senders from a facility using a different type registry and management system

7.1 Identification and versioning of Device Management Interfaces

Further to the above, Device Management Interfaces should use unique identifiers to distinguish between common/standard and vendor-specific parts. In particular, vendors are likely to define their own Capabilities to represent "unique selling points" of their Devices. A namespace-based approach may be suitable.

It should be possible to use multiple versions of a Device Management Interface on the same network, and the unique identifiers should include version numbers. This includes both "major" version changes (which break interoperability) and "minor" changes (which e.g. add new features, but retain interoperability with previous versions).

It should be possible for old Clients to continue working with new Devices without modification. (So for a non-backwards-compatible major Device Management Interface change, a Device would also need to support the old Device Management Interface.)

7.2 Standards

Device Management Interfaces should make use of standardised underlying technologies (wire protocols, data payloads, etc).

Device Management Interfaces should themselves be standardised. This includes specification of how the parts of the interfaces are unique identified (see above). A public registry for these identifiers may be desirable.

7.3 Documentation

Documentation of Device Management Interfaces shall be publicly available.

Freely available (i.e. non-charged) specifications are desirable to aid adoption by smaller developers.

7.4 Legacy control protocols

It should be possible to construct gateways to Device Management Interfaces with legacy control mechanisms, both in terms of control architecture and transport protocols.

7.5 Platforms

Use of a solution should not require adoption of specific compute/network technology; implementations on multiple platforms are desirable.

7.6 Protocols

Solutions shall provide Device Management Interfaces that can support access:

- via an IP-based transport layer
- over NAT, for example for remote control
- on both IPv4 and IPv6 network

Devices that do not implement IP can be supported through gateways to other transports. However, this should be for legacy support and it is reasonable to expect even low-end new Devices to support IP.

Solutions should provide Device Management Interfaces that can be accessed using web-friendly protocols such as HTTP and WebSockets. In some cases an alternative IP-based transport may be appropriate; ideally this should be in addition to support of a web-friendly transport. Where this is not possible, an external gateway can be used.

7.7 Adoption

It is important that equipment from multiple manufacturers can be supported.

[*] Page intentionally left blank. This document is paginated for two sided printing

Annex 1: Scenario 1 - Networked live production with remote elements (BBC)

Type of production

High-quality production of a live event, for example live stadium sports coverage, time-shifted coverage (“as live”), highlights packages, interviews and other features.

Audiences can view the event on televisions, web browsers and mobile devices, and can choose to have a personalised version of what they view.

Production locations

Several locations may be used, with some operations happening remotely, possibly even in a different country. Layer 3 routing will typically be required between operational areas.

Acquisition

Multiple cameras will be used in a production, possibly including UHD as well as HD cameras, and possibly with higher bit rates and dynamic range.

In the short term, cameras will still typically output SDI, take genlock in, and a separate Device will convert this to network streams. Longer term, a network interface will replace these connections.

Camera movement, zoom and focus may be controlled by:

- a human operator at the camera location
- a human operator at a remote location (for example a track camera at an athletics event)
- automation, e.g. using image analysis
- fixed (locked-off camera)

Other camera controls such as iris, gain, colour balance are often remote operated in the "racks" area.

Many microphones and audio sources will be used.

Other live video/audio feeds e.g. from an outside reporter may be used.

As well as live acquisition, productions may also include other ingested and archive content.

Live production operations

Live operations on the content include: vision mixing (mostly cuts, with occasional other effects, especially dissolves), graphics overlay (captions, scores, etc.), action replays (including slow motion), audio mixing (surround), addition of commentary and music.

Production personnel perform live logging to identify events that happen in the stadium, including names of competitors and their location. This may be assisted by automatic analysis and/or use of live captured data.

Time-shifted operations and editing

The production can time-shift parts of the event and present them “as-live” within the main coverage, or as action replays of what was already shown.

Edited packages of the event are created for different purposes, and editing staff may start working on packages while the event is being recorded.

"Factory" operations

As well as the main full-quality output of the production for live distribution to conventional TV platforms, live streams will be created for various other platforms such as mobile and web browsers.

In the longer term an "object-based" approach to distribution will be used, in which the individual live and pre-recorded components of video, audio and data are assembled on demand to create a personalised version of a programme (for example because a viewer is interested in particular athletes or clubs). To provide the flexibility required for the range of different versions, on-demand creation of content processing resources (such as transcoders) may be required.

Content monitoring

Monitoring of live and recorded content for (a) content (b) technical quality. This includes "consumer grade" monitoring, driven by PCs with graphics cards plugged in to domestic TVs, and "professional grade" monitoring, initially via conversion to SDI but later with direct network interfaces on the monitors.

Production communications

Production personnel use talkback, tally and instant messaging for communications. These can be routed automatically to match the set-up of the production.

Presenters may have earphones (“in-ear monitors”) also providing programme audio. In such cases the round-trip delay when programme audio and production communications are combined needs to be low enough that presenter doesn’t here too much delay on his/her own voice, and the director can cue the presenter accurately. This will in turn impose latency requirements on both the production system and the communications system.

Personnel have access to real-time production information, such as running orders, regardless of where they are working.

Networked Devices

This scenario may make use of:

- dedicated hardware Devices, for example camera interfaces
- specialist "PC-based" Devices, for example graphics systems
- Devices deployed on general purpose compute/storage platforms, either physical or using virtualisation/cloud technology

A uniform approach to configuration, control and monitoring of all the above is desired, although it is recognised that specialised Devices may offer a lower degree of flexibility compared to those deployed on general-purpose platforms.

Locations

Some Devices will be permanently installed e.g. in a broadcaster's studio or apparatus room. Others will be deployed for the duration of a production, or part of a production at a particular location. Others will be created on demand (e.g. in a data centre, or "in the cloud") for "factory" operations, so they can scale according to demand.

Set-up

Technical operators can view lists of the Devices available to the production, and can create new devices as required from general-purpose compute/store resources.

They can view Capabilities of the Devices such as:

- types of technical operation that are available, e.g. capture, transcode, storage, display
- the network Flows they can produce
- the formats that can support

In the case of general-purpose Devices, the operators can set these capabilities. This might use a template-based approach.

Devices list their available parameters (e.g. bit-rate) and whether they are fixed or writable. The operator sets them as required.

Monitoring & diagnostics

"Dashboards" provide an overview of the status of the Devices (as well as the networks), and provide alerts of significant problems. Technical operators will be able to examine the status of individual systems, content Flows and network components

[*] Page intentionally left blank. This document is paginated for two sided printing

Annex 2: Scenario 2 - Local Sport (BBC)

Background

FTTC broadband technology has allowed BBC Northern Ireland to expand the reach of its local sport production to new football grounds. At the moment this is limited to IP contributions, which has already demonstrated opportunities and difficulties.

This scenario includes both the above, and an extension to full use of IP for the production.

Remote Site

- Live feeds from all grounds.
- Remote sources are available for selection on the production switcher.
- All sources are viewable in the production gallery at all times on a multiviewer display.
- Feeds are ingested directly into the Television Content Production System (TVCPS). Editing is possible immediately.
- Real-Time proxy and low bitrate streams generated.
- On-Air and Online teams can work in parallel.
- Remote sites should have access to interruptable foldback, tally, production talkback and low bandwidth access to the corporate data network.
- The remote site should have access to production information.
- Reverse vision would be nice to have and would provide a means to deliver Autocue to the remote site.
- 4G connectivity for sites where FTTC circuits are not available.
- Effective error correction for imperfect circuits.
- Store and forward for uploading high quality clips.

Studio Facilities

- Standard TV Production switcher effects.
- Graphics and logos including tickers and lower thirds.
- TV Studio with one or two presenters and commentators and guests.
- Play inserts prepared before and during the programme itself.

Online production

- Full access to user generated content feeds.
- Publishing of goals to social media.
- Multiviewer display of all remote sites.
- Delaying of Multiviewer feed so that games events are not missed.

[*] Page intentionally left blank. This document is paginated for two sided printing

Annex 3: Scenario 3 – Large, full IP indoor audio production (R-F)

Overview

This scenario covers a large indoor audio production, for example a symphonic orchestra recording, based on current practice but also takes into account the possibility of a full IP architecture in the future.

Current configuration

At present the configuration in Radio France is as follows:

- **Mixers:** Normally, each mixer is linked to a dedicated studio, but it can be linked to a side studio if the usual mixer is still used (for post-production for example) or to another mixer if the usual mixer has not enough entries.
- **Mixing surfaces:** operators do not have a direct view of the studio; instead they view via monitors
- **Stage-boxes:** linked to the mixers either directly or through our contribution system. Currently using MADI fibre links
- **Digital Audio Workstations (DAWs):** these can be used for:
  - recording audio tracks out of the main mixer
  - recording the direct out.

Changes for outdoor production

The direct link to the control room is replaced by an IP codec.

Devices

- Mixers 1 and 2: Lawo Mc66
- DAWs: Merging Pyramix.
- Monitoring: Genelec when available
- Microphones: Neumann, Schoeps, etc.

Possible changes for IP architecture

Mixers (Lawo Mc66) and the DAWs (Merging Pyramix) support Ravenna.

Indoor contribution system (Lawo Dallis) can have a Ravenna card, for example, to go to the control room.

Reuse existing MADI fibre links (stage-boxes, sonorisation mixer) for IP, perhaps using MADI/IP converters.

Possibility of digital microphones

Still to be defined

Microphones could be Neumann, Schoeps or other…

Monitoring: one possibility is Genelec when available.

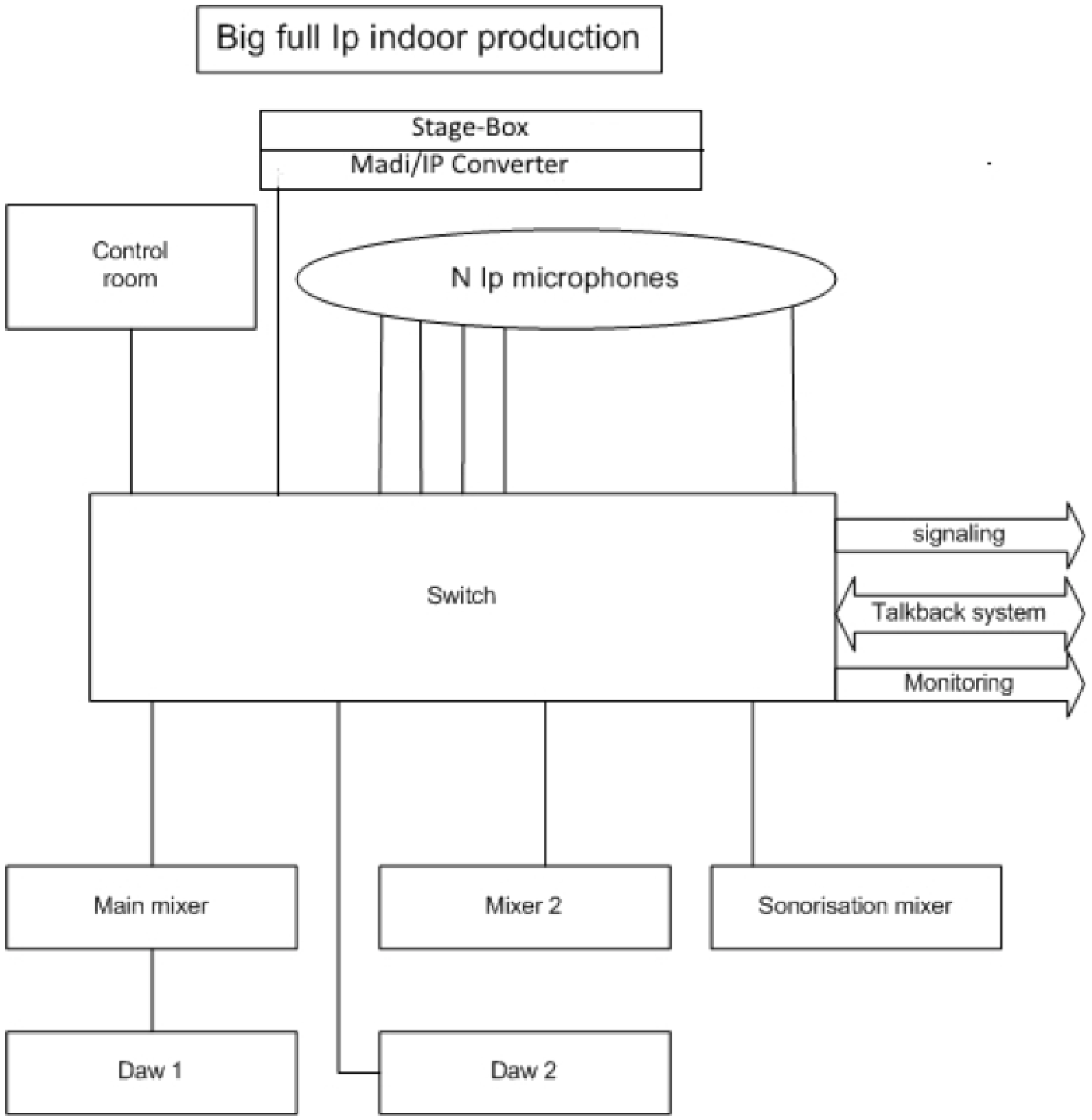
Sonorisation mixer if MADI/IP converter not possible or not suited.

Signalling and talkback are still completely to define.

Diagram

The synopsis below shows a full IP configuration, idealised with only one switch. In practice, management of several switches may be required because:

- the distance is too long
- the number of ports is not enough (many IP microphone links and a large number of monitoring and talkback connections).



Annex 4: Scenario 4 – Radio shows (SR)

Scenario 4a: Typical show

A single presenter in a studio using a mix of the following sources:

- Pre-recorded music
- Pre-recorded items
- Phone-in
- Guest in studio

The presenter logs into the mixer surface in the studio and is presented with a list of snapshots. The first option is the snapshot that is currently loaded for easy access, then comes snapshots related to shows scheduled for this studio or at least the station the studio is connected to. After the recommended snapshots there is a list of all other snapshots available to this presenter.

When the snapshot is loaded, mixing capacity is allocated in a regional mixing engine. Several AES67 audio streams are also established; one for the main mix output and one for each audio input that is both physically connected in the studio in question and also present in the snapshot. The snapshot typically only refers to virtual sources that are matched against the physical sources that are currently available. For example the microphones might be tagged as "Presenter mic 1", "Guest mic 1", etc and they are matched to the relevant slot of the snapshot regardless of what the actual brand, type and physical connection point of the microphones are.

The AES67 streams from the sources are added to the mix engine and the faders on the control surface are linked to the relevant in- and outputs of the mixing engine.

Scenario 4b: Morning show

Similar to the "typical show" scenario with the addition of:

- Reporter in the field
- Interview in the lobby

Scenario 4c: News show

The news show typically consists of a presenter and pre-recorded items, but also adds "reporters abroad". These reporters are ideally connected via ACIP on leased lines, but many other connectivity options must be possible. This might, to some extent, be comparable with the "reporter in the field" concept as described in the "Morning show" scenario.

Scenario 4d: Sports show

Show covering multiple games at once, jumping from arena to arena. Includes multiple ACIP connections

[*] Page intentionally left blank. This document is paginated for two sided printing

Annex 5: Examples of Devices

The following list gives a non-exhaustive list of Devices that may be required, so based on the scenarios of the previous Annexes.

- **Acquisition**
  - camera (iris, gain, white balance, pan, tilt, zoom, focus...)
  - microphone, including IP microphone
  - ACIP/AES67 audio codecs
- **Live production operations**
  - vision mixer
  - audio mixer
  - graphics generator (basic)
  - digital audio workstation
  - keyer
  - record/play
- **Content monitoring**
  - Video monitoring (multiviewers)
  - Audio monitoring
  - Technical parameter monitoring (including QC checks)
- **Communications**
  - talkback
  - chat
  - information systems (for running orders, etc.)
  - tallies (may be part of other Devices)
- **System monitoring**
  - dashboards
  - alerts
- **On-demand "factory" operations**
  - transcoding
  - content packaging
  - streaming
- **Traditional infrastructure**
  - routers
  - converters (aspect ratio, standards)
  - encode, decode, transcode