# 6 Conclusion

This document presents an overall architecture for a broadcast studio control system. It is important for SMPTE to move quickly towards consensus as to the final architecture of such a system. Organizations within the industry are currently developing solutions to meet their own corporate requirements. It is imperative that the necessary SMPTE activities are started such that we may synchronize these industry efforts.

The following summarizes the recommended SMPTE activities as outlined in this overview.

## *6.1 Summary of SMPTE Activities*

#### *6.1.1.1 Content Management Plane*

- The Content and Container data model
- UMIDs and UPIDs (already defined or being defined)
- Identification of any structures that can be used to represent Intellectual Property rights
- Synchronization with the ongoing work of W25

#### *6.1.1.2 Service Plane*

SMPTE should standardize the interfaces between the content-service planes and the service-path planes. Engineering guidelines and/or recommended practices may prove useful in explaining how these interfaces will behave in real installations. Whatever the vehicle, the industry will require direction on how content, services, and paths are modeled to allow the open exchange of such information across multiple vendors' equipment.

The following two items affect the support of managed resources:

- Vendor equipment with capabilities that exceed the standardized model require special considerations in order to utilize those capabilities. Such special considerations are beyond the scope of the standardization activities.
- Vendor equipment that lacks certain capabilities defined in the standard model require special considerations in order to properly manage those limitations. Such special considerations are also beyond the scope of the standardization activity.

Having stated these two items, any standardization activity that precludes the use of any significant percentage of vendor equipment of interest in a user facility runs the risk of being perceived as irrelevant to the user community.

#### *6.1.1.3 Path Plane*

- Agreement on the functionality of studio resource management
- Standardize the interfaces for the Path Construction, the Device Resource Management, and the Network Resource Management

#### *6.1.1.4 Device Plane*

- Standardized MIBS
- Standard list of device types with associated naming hierarchy
- Standard Interfaces for studio equipment classes
- Standard studio data types (i.e. time struct , video control , audio control , file formats, configuration)
- Migration of existing standardization (i.e. SMPTE virtual machines)

- Standard method of describing these interfaces (i.e. IDL, XML, etc.)

#### 6.1.1.4.1 System Layer - Studio Time

- Presentation of a time format including date, time
- Synchronization protocol between objects
- Identification of which legacy formats should be supported natively and which should be proxied to
- Recommendation of new transport mechanisms as well as interaction with existing mechanisms

#### 6.1.1.4.2 System Layer - Fault Recovery

- Standard data format for fault logging
- Standard interface for diagnostic and fault correction
- Modifications to existing status monitoring and diagnostic standards (SMDP)

#### 6.1.1.4.3 System Layer - Identification/ Naming

- Agreement on studio naming convention and device categorization
- Standards on Service Gateway interface
- Standards on device descriptive information format (device metadata)

#### 6.1.1.4.4 System Layer – Security

It is unlikely that SMPTE would need to undertake the development of new security technology in order to meet the needs of the Control System. However, SMPTE will need to choose among existing programming interfaces to security services, or develop programming interfaces to security services, so that applications can interoperate with each other and be portable among Control System implementations.

SMPTE will also need to develop environment classifications based on the level of security required. This will enable applications to interoperate and be portable among different environments, as well as, different Control System implementations. For each of these environments, appropriate profiles of existing security specifications and technologies are developed. These profiles may include:

- Developing or choosing additional programming interfaces
- Choosing authentication and access control mechanisms
- Choosing cryptographic algorithms

For example, one classification, representing the lowest level of security, would include the environment consisting of devices contained within the same room owned by a single administrative domain. The profile for this lowest level of security services would be "null," i.e., no security services. Even though there are no security services required by this environment, requests for security services could be made by applications. Such requests from applications in this environment would result in access always being allowed and/or no action being taken.

Another classification example would include the environment consisting of devices which are geographically distributed but are connected by a private network and controlled by the same enterprise. The profile for this classification might include only authentication services. All other services would be "null."

#### 6.1.1.4.5 System Layer – Configuration

- Describe syntax and facilities to implement configuration services
- Develop a set of common reference templates for each class of equipment (e.g. VTRs, routers, encoders, etc.)

#### 6.1.1.4.6 System Layer - Status/ Alarms

- Definition of standardized MIB extensions and reusable trigger objects
- Describe syntax and facilities to implement status and alarm services
- Develop a set of common reference status and alarm messages for each class of equipment (e.g. VTRs, routers, encoders, etc.)

### *6.1.1.5 System Layer - Communications Layer*

- Acceptance criteria for off-the-shelf technologies
- Recommended protocols and interfaces
- Classifications for the adaptation of existing studio protocols
- Transport interface definition

### *6.1.1.6 System Layer - Transport Layer*

- Acceptance criteria for off the shelf technologies (real-time characteristics)
- Recommended protocols and interfaces
- Classifications for the adaptation of existing studio protocols
- Transport interface definition
- Minimum or mandatory sets of functionality to be available in all systems