

Joachim Schwarz

Europäische Anforderungen an das AES-24 Protokoll

November 1996

Inhaltsverzeichnis

1 Einleitung	1
2 Die Grundidee	1
3 Das Konzept	3
4 Anforderungen	7
5 Schlußbemerkung	11

1 Einleitung

Der AES-24 Standard geht mit großen Schritten seiner Vollendung entgegen. Trotz vieler Mühen und auszuräumender Schwierigkeiten erscheint das Konzept, eine system- und herstellerübergreifende Kompatibilität zu erreichen, doch mehr Sympathien als Ablehnung zu ernten.

Nachdem sich der Standard, wie er in den Vorträgen auf der Tonmeistertagung 1994 dargestellt wurde, in seinen Grundzügen unwesentlich geändert hat, erübrigt sich eine genaue und detaillierte Erklärung, die aber zum besseren Verständnis trotzdem als Précis wiedergegeben werden soll, bevor der eigentliche Kern des Vortrages dargelegt werden kann. Vielleicht, um es gleich vorne weg zu sagen, ist die Behauptung, es gäbe spezielle europäische Anforderungen, ein wenig hochgegriffen. Allerdings betrachten die Mitglieder der verantwortlichen Arbeitsgemeinschaft, SC-10 gewisse Anforderungen, die gerade seitens der europäischen Mitglieder vorgetragen wurden, eben als „speziell europäisch“. Man sollte sich dieser Betrachtungsweise nicht unbedingt anschließen, da wahrscheinlich die gebotenen Möglichkeiten nur konsequent weiter hinterfragt wurden.

Doch nun zur Zusammenfassung:

2 Die Grundidee

AES-24 soll die Verwendung kostengünstiger Komponenten ermöglichen, die dann über ein einfach zu handhabendes und klar strukturiertes Protokoll miteinander kommunizieren und die benötigten Steuerungen ausführen, bzw. Überwachungsfunktionen zur Verfügung stellen. Die Steuerung beruht - konträr

zu den häufig üblichen Master/Slave Konstellationen - auf verteilter Intelligenz, das bedeutet, daß jedes Gerät über einen eigenen Mikrocontroller verfügt, der mit geringem Aufwand angesprochen werden kann und dann komplexe Regelvorgänge steuert. Wie diese lokalen Vorgänge aussehen, ist dem Controller bekannt - daher werden sie mit hoher Geschwindigkeit ausgeführt. Die Betriebssicherheit des gesamten Systems ist hoch, da auch der Totalausfall eines Teilnehmers (Knoten) die anderen Geräte unbeeinflusst läßt.

Die verteilte Intelligenz des Systems gestattet es, mit einer Software viele verschiedene Geräte zu steuern. Hierzu erfolgt ein Datenaustausch zwischen Steuerrechner und Gerät, wobei das Gerät dem Rechner mitteilt, welchen Funktionen es genügt und welche Steuerelemente zur Verfügung stehen. Die Steuersoftware stellt dann, zusätzlich zu den herstellereigenen graphischen Bedienmodulen, die entsprechenden Regler und Schalter auf dem Bildschirm dar.

Aus Redundanzgründen und einer angestrebten hohen Servicefreundlichkeit sollen die verwendeten Netzwerkkomponenten aus dem Standardregal eines jeden Computerhändlers entnommen werden können. Am besten wird daher ein bereits bestehendes Netzwerk genutzt, welches für die Steuerung eventuell nur erweitert werden muß. Die nachfolgende Graphik zeigt, worauf es im Wesentlichen ankommt:

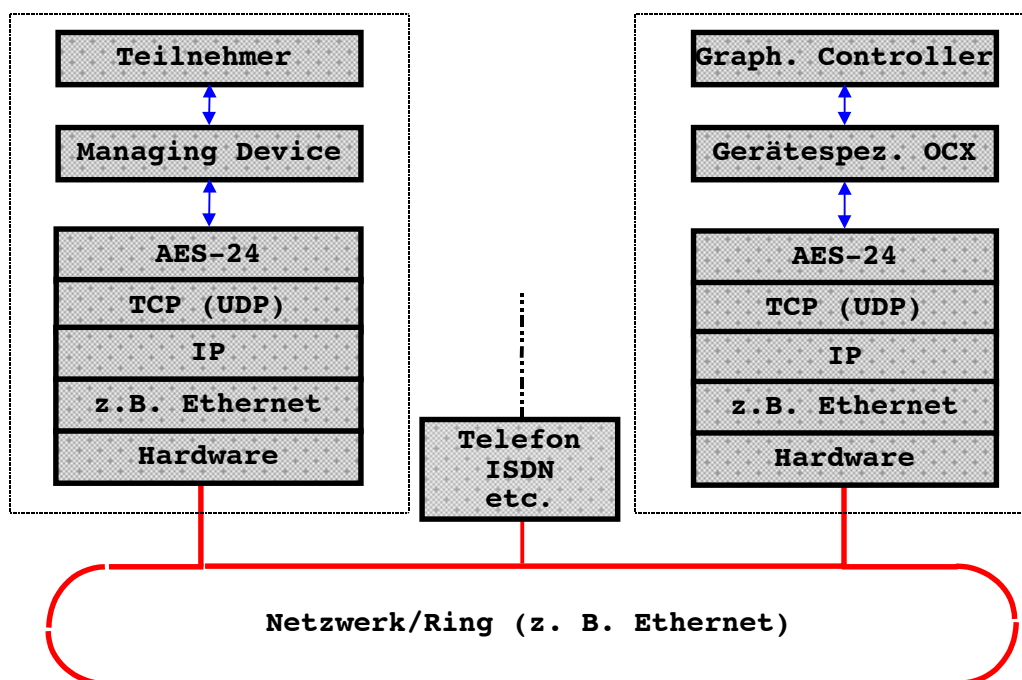


Abbildung 1: Komponenten eines AES-24 Netzwerkes

Dem Bild läßt sich entnehmen, daß die Steuerung einfach zu realisieren ist. Auf der Rechnerseite finden wir ein GUI (Graphical User Interface) mit angegliederter Software, welche die gewünschten Funktionen/Informationen protokollgemäß in AES-24 Steuercode umsetzt. Über das standardmäßige TCP/IP

Protokoll wird alles für die Übergabe an das Netzwerk vorbereitet. Standardkomponenten realisieren dann die Anbindung und Verteilung.

Auf der Teilnehmerseite erfolgt alles in umgekehrter Reihenfolge. Das Managing Device ist ein Mikrocontroller, der die erhaltenen Anweisungen für das Gerät umsetzt, bzw. Meldungen desselben wiederum an das Netz zurückgibt. Die Bidirektionalität erlaubt Steuerung, Rückmeldung und Zustandsanzeigen, sowie aus dem MIDI-Bereich bekannte Kaskadierungen von Befehlssequenzen.

Die Hardware, welche die zu steuernden Geräte an das Netzwerk anbindet, besteht also aus Standardkomponenten. Zur Zeit bietet sich aus Kostengründen die Vernetzung über Ethernet an. Ethernet ist mittlerweile ein so stark verbreitetes Netzwerk, daß die Komponenten jederzeit zur Verfügung stehen und sehr preisgünstig ausfallen. Umsetzer auf optische Verbindungen, Sternverbindungen etc. stellen ebenfalls kein Problem dar. Und Ethernet ist jedem jederzeit zugänglich. Der Einsatz anderer Netzwerke ist im AES-24 Protokoll selbstverständlich gleichberechtigt enthalten.

3 Das Konzept

Es ergeben sich folgende wichtige Punkte:

- a) **Erweiterbarkeit.** AES-24 muß in einer klaren Art und Weise erweiterbar sein, damit zukünftige neue Geräte und Konzepte unterstützt werden können.
- b) **Flexibilität.** AES-24 muß geordnete Verfahren anbieten, um nicht standardmäßige Geräte mit einschließen zu können und eine systemübergreifende Bedienbarkeit zu erreichen.
- c) **Kompatibilität.** AES-24 muß die folgenden drei Kompatibilitätstypen unterstützen:
 1. *Aufwärtskompatibilität*, bei der neuere Versionen von AES-24 auch Geräte unterstützen, welche ältere Versionen benutzen. Die volle Funktionalität der neueren Version wird dabei aber in den meisten Fällen nicht erreichbar sein.
 2. *Abwärtskompatibilität*, bei der ältere Versionen von AES-24 auch Geräte unterstützen, welche bereits eine neuere Version enthalten. Auch hier ist die volle Funktionalität in den meisten Fällen nicht erreichbar.
 3. *Laterale Kompatibilität*, wobei kundenspezifische Verbesserungen oder Veränderungen trotzdem eine übergreifende Bedienbarkeit gewährleisten. Abhängig von der jeweiligen Version der Bediensoftware ist hierbei die volle Bandbreite der spezifischen Lösung verfügbar oder auch nicht.
- d) Flexible Anforderungen an die Datenübertragung. AES-24 muß mit möglichst vielen Datenübertragungsmedien harmonieren, wobei natürlich die Effizienz der Steuerung mit dem Netzwerk steigt oder fällt.
- e) Peer-to-Peer Betrieb (verteilte Intelligenz). AES-24 erlaubt hierdurch, daß von jedem beteiligten Gerät entsprechende Steuer- oder Überwachungsbefehle ausgesendet oder akzeptiert werden können.

f) Sicherheit. AES-24 muß in den jeweiligen Steuernetzwerken verschiedene und ausreichende Sicherheitsabstufungen unterstützen, um unberechtigte Zugriffe zu vermeiden.

AES-24 beruht auf der Grundüberlegung, daß jedem Gerät eine Anzahl von diskreten Funktionselementen innewohnen. Einige dieser Elemente betreffen das Audiosignal direkt, während andere der Verwaltung dienen bzw. den Betrieb unterstützen.

Zusätzlich zu den normalen Audiogeräten wird ein AES-24 Netzwerk normalerweise einen oder mehrere reine Steuergeräte enthalten (sog. System Controller), die alle notwendigen Steuer- und Überwachungsfunktionen zur Verfügung stellen. Diese System Controller werden von AES-24 - entsprechend den Audiogeräten - als eine Anzahl von diskreten Funktionselementen betrachtet.

Im Allgemeinen erscheint es sinnvoll, die System Controller als eigenständige Geräte zu verstehen. AES-24 ist jedoch so aufgebaut, daß es als völlig normal anzusehen ist, wenn ein Gerät beiden Funktionen genügt - also der Tonbearbeitung und der Steuerung des Netzes. Sogenannte Multimedia PCs könnten hier verschiedene Funktionen (wie z.B. Harddisk Recording u.v.m.), sowie die Überwachung und Steuerung einer ganzen Kaskade von nachfolgenden Geräten übernehmen.

AES-24 bezeichnet die verschiedenen diskreten Funktionselemente als **Objekte**. Ein Objekt repräsentiert hierbei Daten und assoziierte Handlungen, die entsprechend den Daten ausgeführt werden. Für AES-24 ist also ein Objekt eine Abstraktion, welche die kontrollierbaren Audio- und anderen Funktionen bildet.

Der Datenaustausch zwischen Objekten erfolgt ausschließlich mit sogenannten **Messages**. Genauer: Bei einer Message handelt es sich um einen genau abgegrenzten Datenblock, der zwischen Objekten ausgetauscht wird.

Um AES-24 und seine Struktur besser verstehen zu können sowie die verhältnismäßig universelle Struktur besser auszuleuchten, schnell noch einige Begriffe mehr. Und zwar die Begriffe **Klasse** und **Instance**, sowie **Klassenkennung**, **Klassenstruktur** und **AES-24 Interface**. Diese Begriffe sind anschaulich und lassen sich graphisch leicht darstellen.

Die nachfolgende Zeichnung erklärt das Prinzip bereits relativ gut. Wir haben es in diesem angenommenen Fall mit einem Verteilverstärker zu tun, wie er in verschiedenen Formen sowohl in Beschallungsanlagen wie auch als möglicher Verteiler usw. im Studio- und Rundfunkbereich vorkommt.

Der Verstärker enthält ein Schalterobjekt für EIN/AUS, vier Verstärkungs-Steuerobjekte und jeweils ein zugehöriges Pegelmeßobjekt, welches die Pegelinformationen und den System Controller zurückmelden kann.

Der System Controller, der in unserem Fall von einem PC dargestellt werden soll, enthält ein graphisches Programm. Dieses GUI (Graphical User Interface) stellt vier bewegliche „graphische“ Fader zur PegelEinstellung, vier Pegelanzeigen und ein Taster/Schalter Objekt zur Ferneinschaltung. Das AES-24 Modell betrachtet die auf dem Bildschirm enthaltenen Elemente als Objekte. Steuerung und Überwachung werden dadurch realisiert, daß Messages zwischen den Objekten der Steuerung und dem Verstärker hin- und hergesandt werden. AES-24 spezifiziert also diese Prozesse, durch welche diese

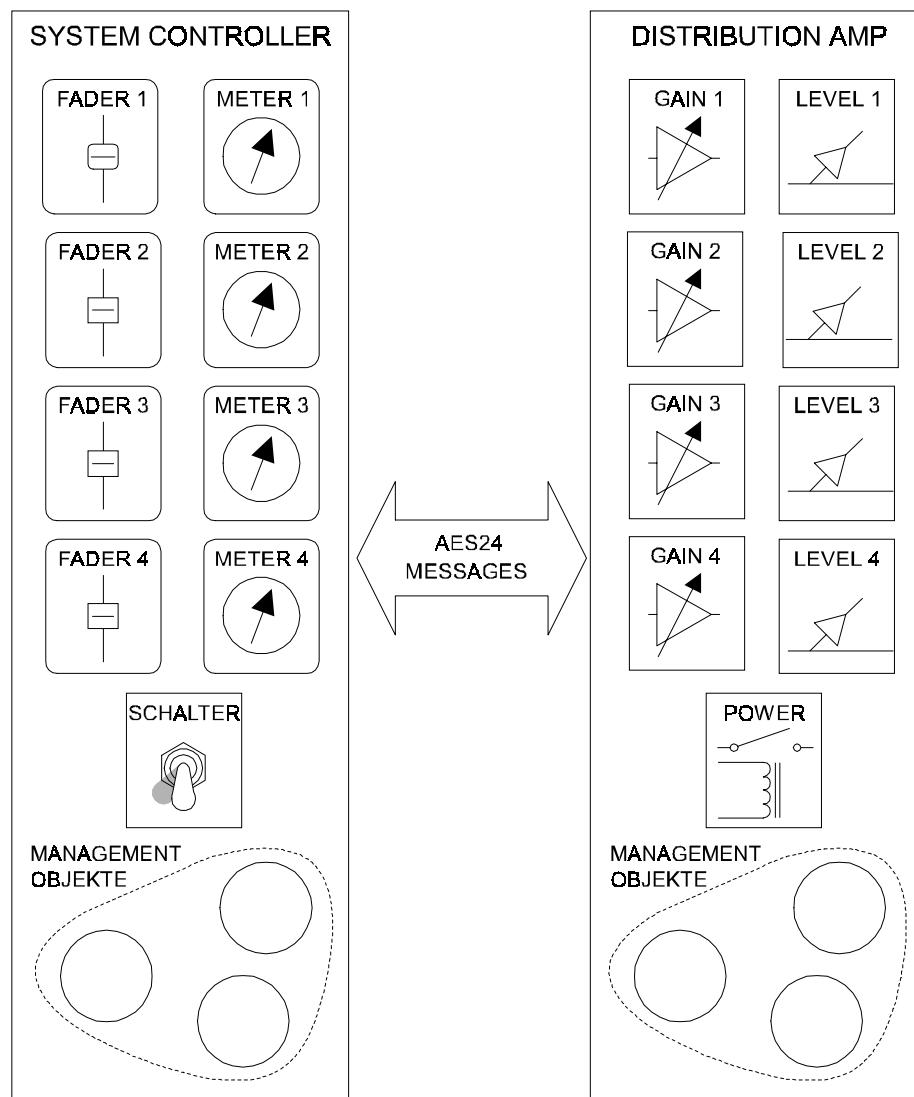


Abbildung 2: System Controller mit gesteuertem Gerät

Objekte sich gegenseitig identifizieren und übereinander informieren, sowie das Format, die Datensequenzen und Auswirkungen der ausgetauschten Daten.

Zusätzlich zu den GUI- und Geräteobjekten zeigt das obenstehende Bild sogenannte „Management Objekte“, die sowohl im Gerät wie auch dem Controller enthalten sind. Diese Objekte realisieren das AES–24 Protokoll, tragen aber ursächlich nicht direkt etwas zur Steuerelektronik oder den GUI-Funktionen bei. Sie dienen der Initialisierung und Konfiguration.

Schön ist vielleicht auch der Umstand, daß AES–24 eine reine Abstraktion der vorzunehmenden Prozesse darstellt. Das heißt, daß es das AES–24 überhaupt nicht kümmert, wie die im Gerät enthaltenen Objekte ihre Regelvorgänge verwirklichen. In unserem Beispiel kann also das gleiche Protokoll sowohl einen analogen Verstärker mit VCAs als auch die Prozesse innerhalb von DSPs steuern. Weiterhin gilt auch noch folgendes: Es ist für das AES–24 Protokoll irrelevant, ob die Steuerbefehle von einem PC

kommen oder von mit dem Netzwerk verbundener Hardware (z.B. einem kleinen Controller mit „echten“ Knöpfen).

Die oben erwähnten Begriffe seien nachfolgend noch einmal kurz umrissen:

Klasse: Die Vorlage, welche einen bestimmten Objekttyp definiert. Die Objektklasse legt die Befehle und Botschaften fest, die ein Objekt empfangen und aussenden kann, und definiert deren Bedeutung.

Instance: Wird als Synonym für ein Objekt benutzt, um dadurch dessen Eigenschaften als Gesamtheit seiner Klassen hervorzuheben.

Klassenstruktur: Ein Anzahl von Klassen innerhalb von AES-24 und deren hierarchische Beziehung.

Klassenkennung: Eine einmalige und unverwechselbare Identifikation einer AES-24 Klasse.

AES-24 Interface: Eine Anzahl von Objekten, die ein Gerät dem AES-24 Netzwerk zu Kontroll- und Monitorzwecken zur Verfügung stellt.

Des weiteren sind zum Verständnis des Protokolls die nachfolgenden, immer wiederkehrende Begriffe von Bedeutung. Diese Begriffe sind daher besonders zu beachten:

Methoden: Eine Methode kann weitgehend als eine Handlungsschnittstelle verstanden werden, welche bestimmte Parameter erfordert, Handlungen ausführt und Statusinformationen zur Verfügung stellt, nachdem eine Funktion ausgeführt wurde. Eine Methode wird vollständig definiert durch:

- a) die Parameter, die sie akzeptiert
- b) die Handlungen, welche sie ausführt
- c) die von ihr generierten Antwortmessages (Datenblöcke)

Jede AES-24 Message ist entweder eine Anfrage einer bestimmten Methode oder die Antwort auf eine vorhergehende Message. Die Anzahl der Methoden innerhalb eines Objektes stellt dessen vollständige AES-24 Funktion dar.

Übertragungsobjekt: Ein Objekt, welches ausschließlich der Datenübertragung und dem Austausch von Messages zwischen Objekten dient.

Handle: Ein dynamisch zugewiesener 32 Bit String, welcher zu einem bestimmten Objekt gehört und es unverwechselbar innerhalb des Netzwerkes identifiziert. Eine genaue Definition des Handles und seine hexadezimalen Bedeutungen muß leider wegen der Kürze des Vortrages entfallen. Die Bedeutung einer eindeutigen Identifizierung ist aber ohnehin jederzeit einleuchtend und offensichtlich.

Registratur: Ein AES-24 Netzwerk kann über ein Register verfügen, welches die im Netzwerk enthaltenen Handles verwaltet. Um Verwechslungen zu vermeiden, kann jedes Netzwerk höchstens ein aktives Register enthalten. Eine Registratur ist also ein „Server“ bzw. Dienstleister, innerhalb eines AES-24 Netzwerkes, welcher die Objektpfade (= strukturierte Namen, welche jedes Objekt im Netz kennzeichnet) an Handles zuweist und umgekehrt. Dabei wird auch eine zusätzliche Namens- und Adressverwaltung zur Verfügung gestellt.

Ein Register stellt also die folgenden Dienstleistungen zur Verfügung:

a) Verwaltung von Handle-Werten. Das Register verwaltet die Handles und Objektpfade, die durch sie repräsentiert werden. Auf Anfrage werden Handles registriert und deregistriert.

b) Informationen über die im gesamten Netzwerk oder Teilen hiervon enthaltenen Objekte (Objektinventur). Die Netzwerkinventur definiert die enthaltenen Objekte, deren Handles und ihre Klassenkennung.

Das Register stellt eine zentralisierte Ausgabestelle für Konfigurationsinformationen dar. System Controller und andere Geräte können im Bedarfsfalle diese Informationen beim Register abfragen. Der Abfrage der Netzwerkinventur wird Inventory Gathering genannt.

c) Übermittlung von Veränderungen der Inventur. Wenn ein Netzwerk initialisiert wird oder sich die Konfiguration eines aktiven Netzwerkes ändert, verteilt das Register entsprechende Hinweise und Nachrichten an seine „Abonnenten“. Jedes Objekt innerhalb eines Netzwerkes kann zum Abonnenten werden, indem eine entsprechende Anfrage an das Register gesendet wird.

Die Graphiken (Abb. 3 und 4) erleichtern hoffentlich das Verständnis. Leider ist bei einem offenen und intelligenten Protokoll ein gewisser Wissens-Wasserkopf notwendig. Vielleicht sei die folgende Nebenbemerkung erlaubt: Die Anzahl der Konventionen, Definitionen und Grundüberlegungen steigt proportional mit der Bedienfreundlichkeit. Denn schließlich soll AES-24 dem Anwender das Leben erleichtern, was natürlich eine wohl durchdachte Universalität voraussetzt.

Bild 3 und 4 stellen im Wesentlichen das gleiche dar, wobei die im Bild 4 enthaltenen Informationen das Prinzip auf ein gesamtes Netzwerk (wenn auch von kleinem Umfang) ausdehnen.

Wesentlich ist, und soviel sei hier zusammenfassend gesagt, daß es sich bei AES-24 um ein objektbezogenes Protokoll handelt. Der Vorteil ist hierbei, einzelne Objekte ändern und zusätzlich aufnehmen zu können und dabei auch zukünftigen Entwicklungen Raum zu schaffen.

Programmiersprachen wie Visual Basic, bzw. Visual C++ arbeiten vollständig objektorientiert. Die Programmierung einer Bedienoberfläche in einer dieser beiden Sprachen ist daher wie geschaffen für die Aufnahme des AES-24 Protokolls. Natürlich sind andere Sprachen nicht davon ausgeschlossen, ebenfalls zur Erzeugung eines GUI herangezogen zu werden.

4 Anforderungen

Lassen Sie uns nun zu den Anforderungen kommen, die seitens verschiedener europäischer Hersteller an die SC-10-2 Arbeitsgruppe des AES-24 Ausschusses herangetragen werden.

Wie schon Eingangs erwähnt, muß man nicht unbedingt von einer speziellen Anforderung sprechen, obwohl bei der Durchsicht des Protokolls und der gebotenen Möglichkeiten sich besonders eine wünschenswerte Eigenschaft förmlich aufdrängt: Die automatische Erkennung von Geräten durch einen

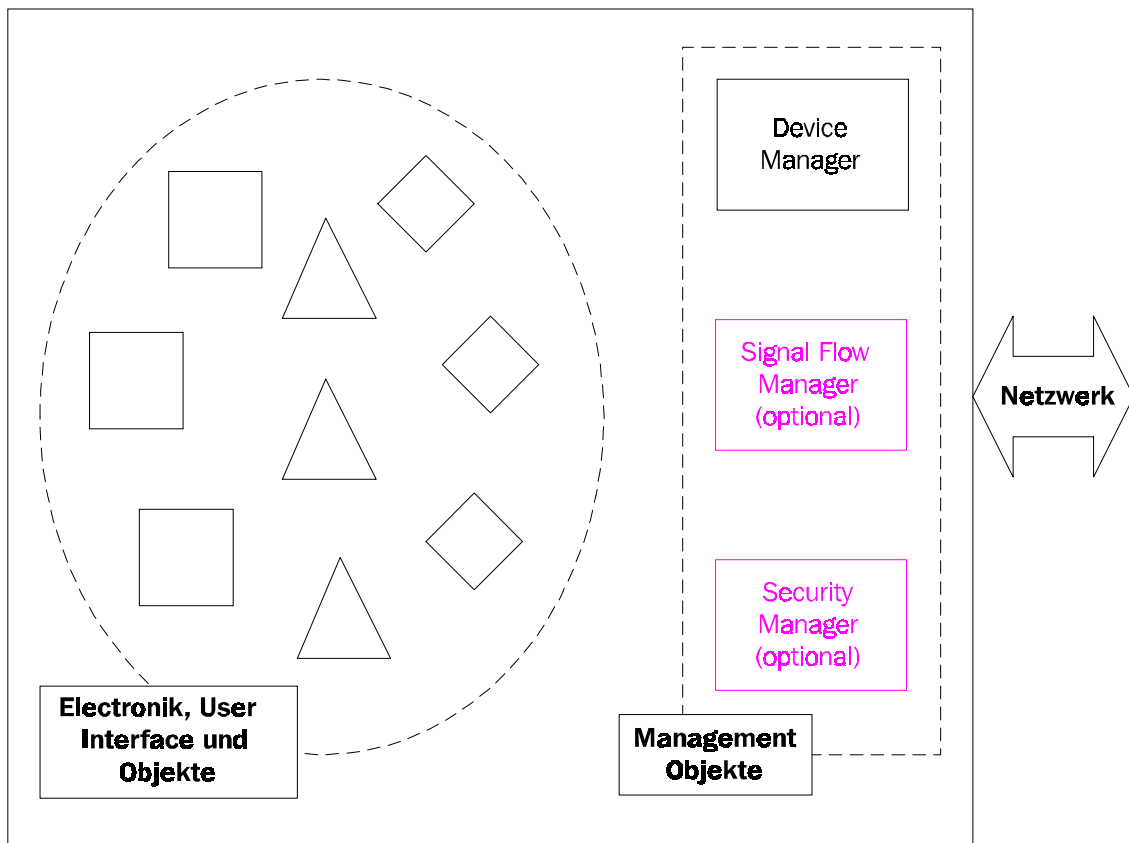


Abbildung 3: AES-24 Device Modell

„generischen“, also offenen und allgemeinen Controller. Dieser erkennt die im Gerät enthaltenen Steuerobjekte und stellt entsprechende graphische Elemente auf dem Bildschirm zur Verfügung. Hiermit wird erreicht, daß auch Geräte, deren Steuersoftware nicht zur Verfügung steht, geregelt und überwacht werden können. Diese generische Steuersoftware ist zwar nicht so schön wie vom Originalhersteller, erfüllt aber ihren Zweck.

Einfacher gesagt als getan! Wie könnte diese nützliche Eigenschaft nun verwirklicht werden? Das Fragezeichen ist bitte wörtlich zu nehmen, da man sich im SC-10-2 Ausschuß noch nicht im Klaren über die Realisation ist. Anregungen in diese Richtung sind daher willkommen.

AES-24 ergreift verschiedene Maßnahmen, um über ein Netzwerk sinnvolle Auskunft zu erhalten bzw. den System Controllern solche geben zu können. Hierbei ist es notwendig, mit Hilfe eines sog. System Builders alle notwendigen Steuerpfade zu errichten. Hierzu sind natürlich ebenfalls verschiedene Schritte vorzunehmen.

Der System Builder (=System Architekt) bekommt vom Register Auskunft, wieviele verschiedene Handles im System enthalten sind und muß daraus die Steuerpfade erstellen. Es ist außerdem notwendig, daß der System Builder hierzu die zum Netz gehörigen Objekte anspricht und Informationen über sie bzw. die richtigen Verbindungen ablegt (registriert). Zusätzlich unterrichtet das Register den System Builder über die entsprechenden Steuerbereiche der enthaltenen Handles.

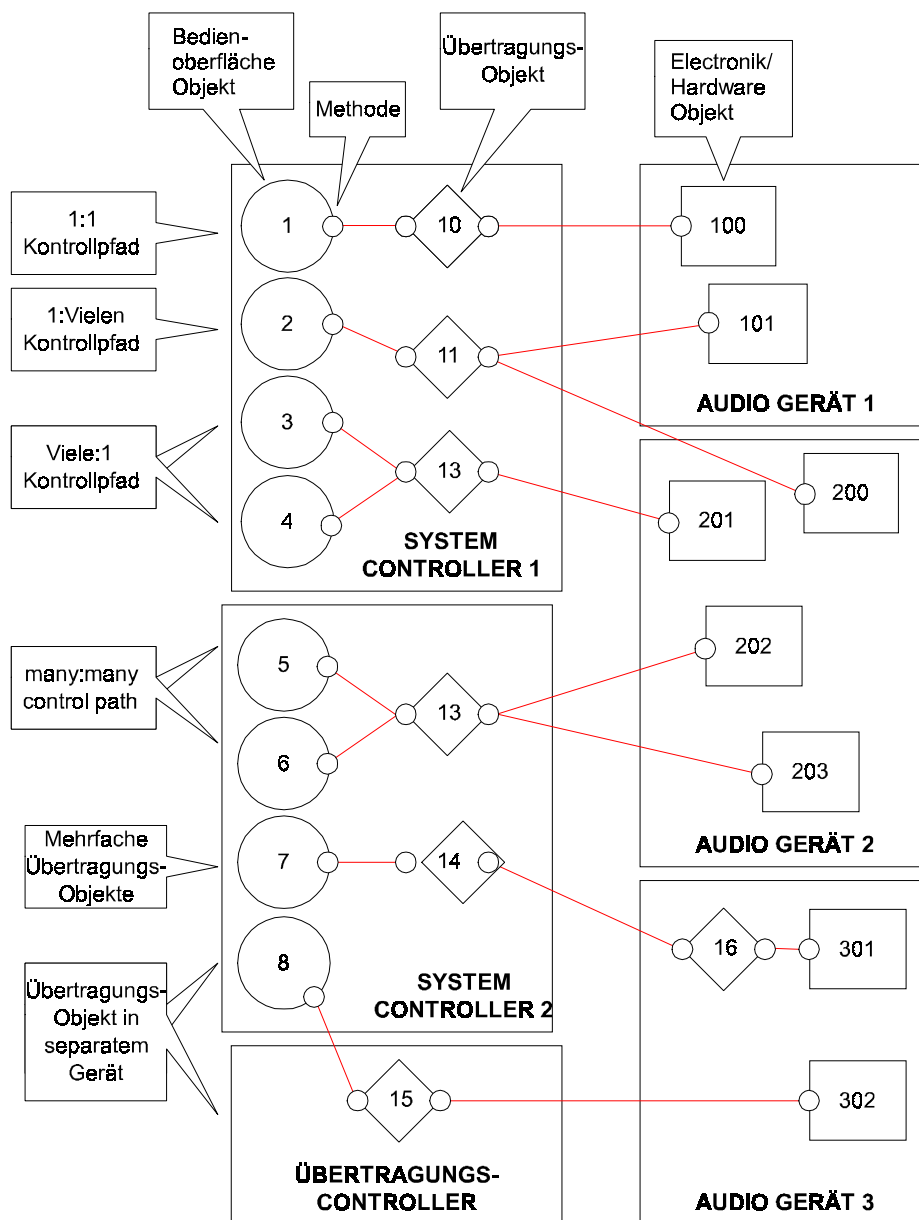


Abbildung 4: Verbindungen, Steuerpfade und Objekte

Über die Informationen hinaus, die ein System Builder vom Register erhält, kann er über einen Vorgang, den AES-24 als „Discovery“ definiert, genaue und detaillierte Auskünfte von den beteiligten Geräten erhalten. Es ist durchaus möglich, daß der Discoveryprozess Informationen liefert, die nicht im Register abgelegt sind, da sie evtl. im Konflikt mit bestimmten Handles anderer Geräte stehen oder bislang nicht notwendig waren. Ein generischer System Controller ist nun gezwungen, mehrere Funktionen ausführen zu können, damit er die gewünschten Steuer- und Überwachungselemente zur Verfügung stellen kann. Die Erfassung der im Gerät enthaltenen Daten kann hierbei entweder automatisch oder halbautomatisch erfolgen, wobei letztere Möglichkeit nicht immer vermeidbar sein wird. Der Vorgang könnte oberflächlich betrachtet, relativ einfach aussehen. Der System Controller entdeckt ein unbekanntes

Gerät:

Frage: Was bist Du?

Antwort: Ein Entzerrer.

Frage: Wieviele Steuerelemente sind vorhanden?

Antwort: 30 Faderobjekte

Frage: Regelbereiche?

Antwort: Fader1: 20 Hz; +/- 15 dB; 0,5 dB Abstufung//Fader 2: u.s.w.

Anweisung an graphischen System Controller: Generiere mit GUI 30 Faderobjekte. Erstelle entsprechende Steuerpfade. Stimme mit dem Register die Klassenkennung und die Identifikation der entsprechenden Handles ab. Erteile dem Equalizer entsprechende Identifikationsnummer.

Natürlich im Zusammenspiel mit dem Systembuilder, ohne den das nicht möglich wäre. Zur Verdeutlichung noch einmal das bereits eingangs gezeigte Bild. Die eben beschriebenen Vorgänge durchlaufen alle Ebenen im Controller wie auch im zu steuernden Gerät!

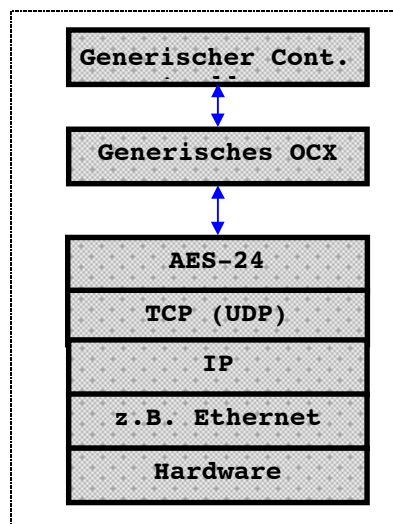


Abbildung 5: Möglicher generischer System Controller

So ähnlich könnte eine generische Erkennung aussehen. Die Schwierigkeiten beginnen eigentlich beim Einholen der Informationen, dem Festlegen der Regelbereiche, dem Erzeugen der Handles, Objektklassen, etc. Diese müssen in jedem Falle unverwechselbar zu dem zu steuernden Gerät gehören. Für einen intelligent programmierten Controller ist das aber keine unlösbare Aufgabe. Trotzdem ergeben sich immer einige Schwierigkeiten. Beispiel: Ein Kombi-Gerät, bestehend aus Delay, Hall, EQ, Kompressor und automatischer Pegelsteuerung soll vom generischen System Controller gesteuert werden. Das heißt, er benötigt hierzu Informationen und Bezeichnungen über die verschiedenen Module, die zugehörigen Steuerelemente, deren Bereiche und die Auflösung. Die Information hierzu - mangels einer entsprechenden Hersteller-Software, muß teilweise erzeugt und teilweise aus den Speichern des Gerätes abgezogen werden. Sicherlich kann der System Controller (weil innerhalb eines PCs ansässig) über bestimmte Informationen verfügen, welche grundsätzlich zur Steuerung der verschiedenen

Geräteklassen notwendig sind. Trotzdem muß im zu steuernden Gerätes alles Notwendige abgelegt sein. Die im Gerät enthaltene Intelligenz wird dabei im Durchschnitt erheblich über einer einfachen Steuerung anzusiedeln sein. Wie fast immer: Alles nur eine Frage der Kosten und Stückzahlen.

Zusätzlich muß die Abfrage der Daten natürlich auch noch schnell realisierbar sein. Hohe Datenmengen erzeugen aber innerhalb eines Netzwerk einen Engpaß. Daher sind sicherlich einige Konventionen zu beachten, die innerhalb des generischen Controllers mit geringem Aufwand für die entsprechenden Möglichkeiten sorgen. Denkbar ist in diesem Falle eine Art Bibliothek, die über Steuerobjekte, deren Bereiche und Auflösung verfügt und im Controller grundsätzlich enthalten ist. Eine einzige Kennzahl wäre dann notwendig, um ein Steuerelement aufzurufen. Nachteil: Keine Zukunftssicherheit. Abhilfe: Software Updates - mit allen Vor- und Nachteilen.

Es ist auch als sicher anzunehmen, daß der generische Controller Bestandteil eines normalen System Controllers sein muß. Daher sind alle Firmen, die einen System Controller anbieten, gezwungen, eine allgemeine Anwendung mit vorzusehen.

5 Schlußbemerkung

Die SC-10-2 Arbeitsgemeinschaft, die den AES-24 Standard während der letzten Jahre kontinuierlich entwickelte, nähert sich der Vollendung. Die hier kurz umrissene Version existiert bereits als vollständiges Protokoll, dessen geringfügige Ergänzungen bis zum Ende Januar 1997 abgeschlossen sind. Nach der Veröffentlichung und der Abwicklung der notwendigen Formalitäten, tritt dieser Standard höchstwahrscheinlich mit der 103. AES Convention 1997 in New York in Kraft.

Die Definition dieses Standards hat sich über mehrere Jahre hingezogen. Der Gründe für immer wieder geäußerte Änderungswünsche waren bessere und umfassendere Erfahrungen und Erkenntnisse mit Steuerungen, sowie einer sich endgültig etablierenden Fixierung des PC-Marktes auf objektbezogene Bedienprogramme und Programmiersprachen. Hierdurch entstand seitens vieler Hersteller ein besseres Verständnis für die eigentliche Bedürfnisse der Anwender. Das Erscheinen immer weiterer proprietärer GUIs mit unterschiedlichsten Anforderungen an die Steuerrechner bzw. Hardware, machten es unabdingbar, mit großem Ernst und Einsatz an die Verwirklichung eines solchen Standards zu gehen. Der Erfolg von MIDI und anderen Standards rechtfertigt hier den gezeigten Einsatz auch ökonomisch.