

We're developing the media network control and monitoring communication standard of tomorrow...today.

Join Us

**OPEN CONTROL
ARCHITECTURE
ALLIANCE**

ABOUT Open Control Architecture

The Open Control Architecture is a system control and monitoring interoperability architecture. The motivation behind OCA is to allow robust control and monitoring interoperability across a wide range of devices from different manufacturers. The creation of a common control architecture will allow new levels of complex system design and integration while simultaneously reducing time, cost, and effort.

OCA's intended application is professional media networks.

The new **AES** standards project **X210** is starting work now to make OCA into an open public standard.

ABOUT The OCA Alliance

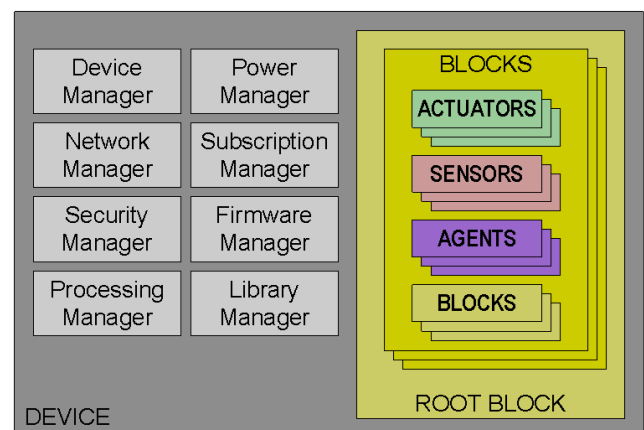
The OCA Alliance was formed by a group of nine professional audio manufacturers who seek to develop and promote the Open Control Architecture (OCA) become an open public communications standard for control and monitoring of devices in professional media networks.

The OCA Alliance is a non-profit trade association and is seeking new members.

Full Members

Bosch
d&b audiotechnik GmbH
Duran Audio
LOUD Technologies
PreSonus
Salzbrenner Stagetec Mediagroup
TC Group
Yamaha Commercial Audio

Device Model Diagram

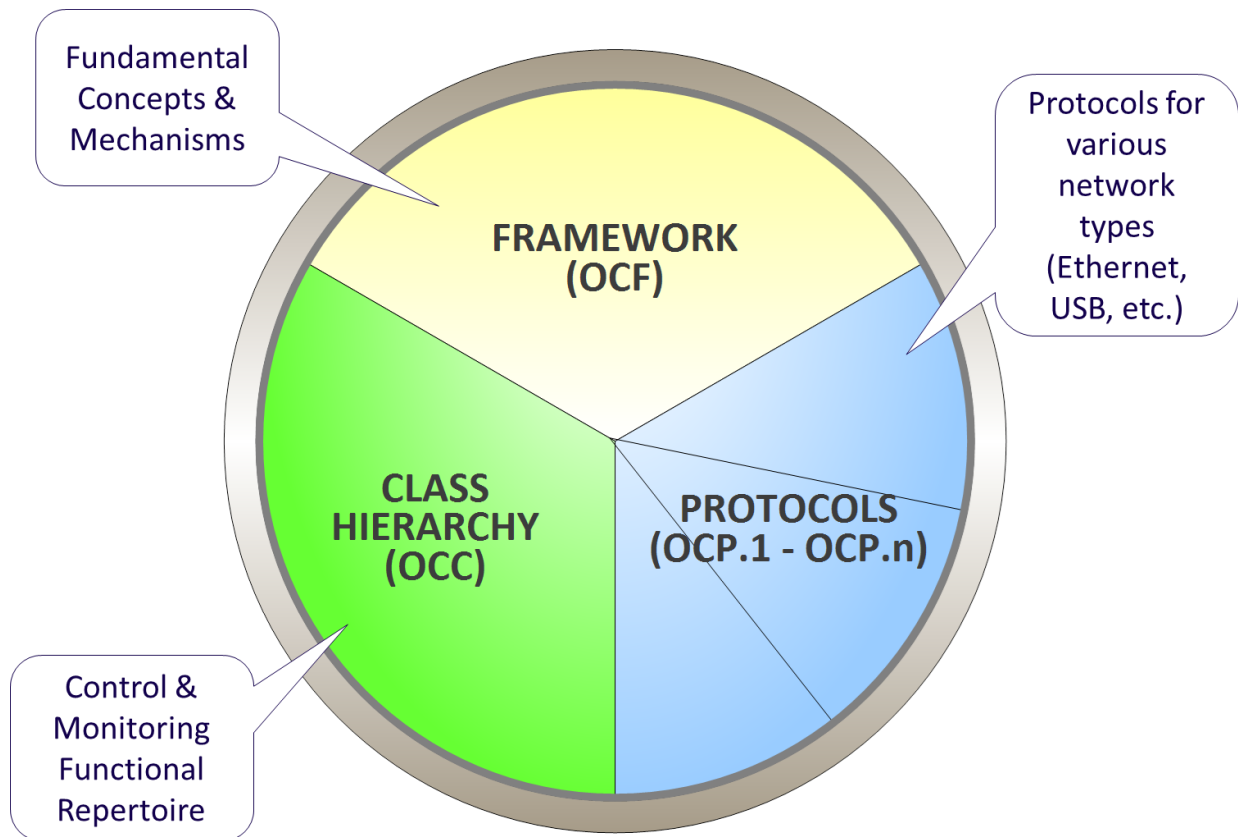


<http://www.oca-alliance.org>

We're developing the media network control and monitoring communication standard of tomorrow...today.

Join Us

OPEN CONTROL ARCHITECTURE ALLIANCE



Framework (OCF)

- Device model
- Functional mechanisms

Class Tree (OCC)

Object-oriented definition of control & monitoring functional repertoire.

- Manager Classes *Device housekeeping & global functions*
- Worker Classes *Signal processors & monitors*
- Agent Classes *Control & monitoring modifiers/aggregators*

Protocol Implementations (OCP.1 ... OCP.n)

OCA will be a family of protocols for different contexts.

- OCP.1 : for TCP/IP networks
- OCP.2-n : TBD, may include USB, XML, ...

Worker Examples

•Actuators

- OcaGain = { Value }
- OcaFilterParametric = { Frequency, Boost, Shape }
- OcaMute = { Setting [Muted | Unmuted] }

•Sensors

- OcaLevelMeter = { Reading, MeterLaw }
- OcaTemperatureSensor = { Reading }

•Blocks

Agent Examples

- Grouper = { Sets of Workers }
- Library = { Sets of stored parameter values }

