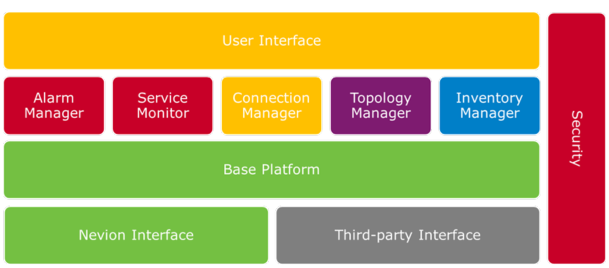
Ref. Ares(2020)627860 - 31/01/2020

Orchestration and SDN Control Platform

The VideoIPath platform addresses the fulfilment, assurance and inventory workflows required to operate modern and rapidly changing media networks. The platform supports both Nevion and third-party devices. New third-party devices are integrated on a per project basis using a software development kit maintained by Nevion.

VideoIPath offers Web based user interfaces and is based on a distributed computing platform with built-in redundancy mechanisms. The platform is designed for carrier-class environments offering a role-based security model. The platform provides published open APIs for integration with third-party control or management systems, which also facilitates development of custom user interfaces.

The platform consists of different modules to support various orchestration and controller functionalities. These modules may be licensed separately and combined to form a deployment for a specific purpose. The figure below illustrates the structure.



At a high-level the system is divided into the following modules:

- User interface: Web based user interface designed around apps for specific purposes. Supports customizable security roles to allow access for end-users to operations and engineering staff.
- Connection manager: Dynamically provision end-to-end services (occasional use or semi-permanent). Ability to provision both media edge and network devices.
- Alarm manager: Monitors alarms received from devices and system generated events. The architecture is fully event-driven and relies on notifications received from devices in combination with regular alarm synchronization.
- Service monitor: Responsible for correlation of alarms towards services. The correlation is automatic for services provisioned by the system, but the behaviour may be customized using templates.

User Interface

The system provides a modern UI that runs directly from a standard Web browser without use of any plugins. This enables the operator to easily access the UI from any location without depending on any specific client-side installation.

The UI is divided into several applications with a clearly defined purpose. These can be filtered in security roles to allow an operator to only see relevant apps for his/her workflow. The screenshot below shows the apps that are currently part of the platform. As can be seen these are divided into three different areas:

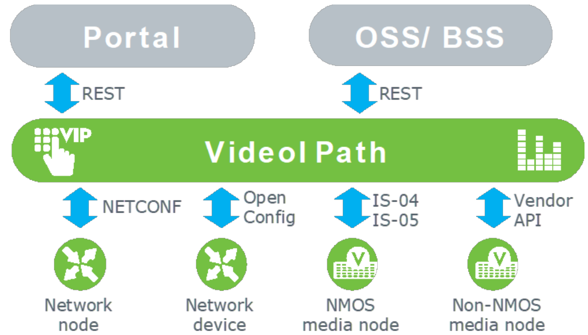
- Service Fulfilment: The process of planning and provisioning services across media nodes and the network fabric.
- Service Assurance: The process of monitoring devices and assessing potential service level impact, in order to allow the operator to focus mainly on issues that affect services.
- System Configuration: These are administration and engineering interfaces to assist with the setup, maintenance and adaptation of the system.

Integration

The VideoIPath system is per-design an open system that can interface with any external system. This is facilitated using an industry standard REST interface that exposes all data within the system and allows external systems to access functionality and modify data provided by the system. The REST interface is already used for integration with existing 3rd party systems like e.g. DataMiner.

The system also supports various other protocols used within the broadcast market for integration with external system (e.g. Ember+). This may allow customers to offer direct integration with broadcast control systems that the client may be using internally, which will enable increased usage of the platform and in turn increased revenue.

The figure below illustrates how the orchestrator is interfacing with other systems and devices.



The platform also includes an SDK for development of third-party drivers, and Nevion has extensive experience with such integrations. This includes a process for defining functional and technical requirements, in addition to the development and verification of the purpose-built driver. The intention is to make the SDK available to customers during 2019.

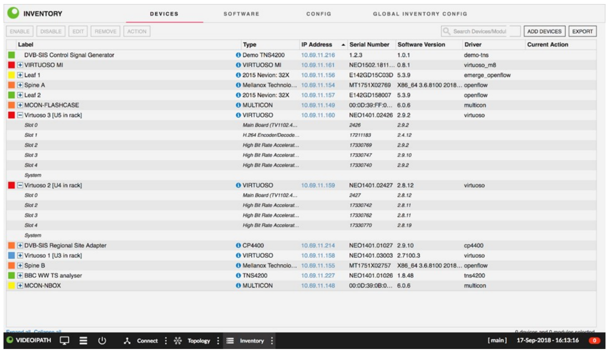
The SDK has extensive support for a variety of different protocols towards media edge and network devices. This includes NETCONF, OpenConfig, Openflow, HTTP(S) with JSON or XML data, SNMP, CLI and socket-based communication. There is also the flexibility to add new protocol support on demand. Note that drivers for Nevion devices are included with the platform.

The system also supports NMOS IS-04 and IS-05 APIs for discovery, registration and connection management of media nodes. This may be particularly relevant when addressing the market for SMPTE 2110 interconnect of IP based broadcast facilities.

The VideoIPath system includes an inventory database that holds all media nodes and network devices that are managed by the system. The inventory enables operators to browse or search for particular devices. In addition, the inventory functionality includes configuration backup, restore and software upgrade capabilities for efficient management of devices subject to driver support.

The system currently supports auto-discovery of known equipment using SNMP. This is achieved by scanning of defined IP ranges using SNMP, and if a known sysObjectId is found then the device is added to the system inventory. The scanning process may be configured as a background task or executed on demand. In addition, devices can be added manually to the system or imported from external data if required.

Nevion has been part of the NMOS standardization and interop testing organized by AMWA. The VideoIPath system includes an NMOS IS-04 discovery and registration service, which enables the system to discover media nodes in a standardized way if they support the NMOS architecture and principles. It is recommended that NMOS IS-04 will find the address of the NMOS registry service using unicast DNS-SD.



Typology

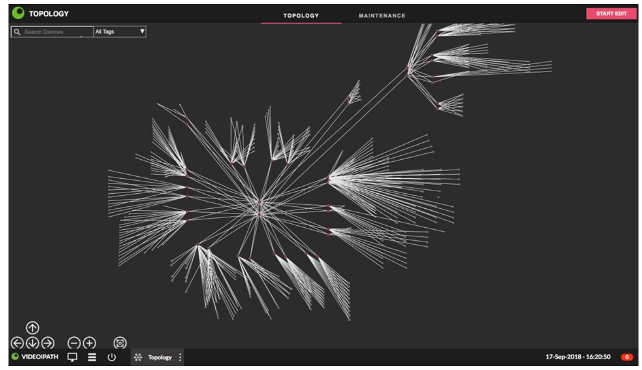
The VideoIPath system has full topology awareness and can perform end-to-end routing based on this information. The topology includes all media nodes and network devices and their connectivity. The topology includes all external ports for a particular device, which may be logical ports (e.g. SMPTE 2110 sender) or physical ports (e.g. ASI port or Ethernet interface).

The topology is used to make intelligent routing decisions and perform bandwidth management to distribute load on available links and prevent over-provisioning of the infrastructure. The topology model is also important to perform diverse path routing for SMPTE 2022-7 protected services.

The system can synchronize information about media nodes and network devices from device drivers. The topology information can be maintained using the Topology app within the system or alternatively using import/export of topology via API.

The topology is also concerned with the connectivity between devices and this information is used for routing purposes. For instance, the topology includes the links between physical ports on a media node and network devices. It is possible to set several properties to influence the routing of services, such as link cost, load balancing, or routing constraints. The connectivity may also be synchronized using information provided by LLDP and/or BGP-LS protocols.

The screenshot below shows an example topology from the Topology app provided by the system.

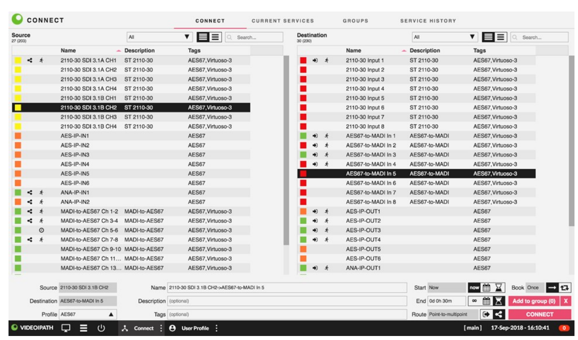


Connection Management

Managing connectivity between senders and receivers at first seems like a simple task, but the complexity quickly grows beyond what is possible to comprehend manually or even with traditional network management systems when you consider a scenario with thousands of endpoints where connections are dynamically changing.

VideoIPath allows the operator to setup connections by selecting the appropriate endpoints and a connection profile, where the profile controls properties such as bandwidth, codec settings, redundancy requirements, etc. The system automates all the complexity associated with routing and configuration of the service both in the media nodes and network devices.

The screenshot below shows the Connect app which is commonly used to establish point-to-multipoint connectivity between senders and receivers.



VideoIPath utilizes a shortest path first algorithm to provision the least cost path from source to one or more destinations. Different routing policies including diverse path, dual path (1+1), passive path (1+0) and partner protection may be configured depending on the redundancy requirements for the service. The system keeps track of bandwidth allocation on each link within the core network to avoid any over-provisioning, delivering predictable performance for IP based services.

The system calculates a path between the sender and receiver based on the topology. Any resources required to establish the connection are reserved for the duration of the connection. This includes intermediate processing devices and bandwidth required to fulfil the request. The operator is notified if for some reason the request cannot be fulfilled due to resource or bandwidth limitations.

Resources along the path of the connection, which again includes sender/receiver, network infrastructure and intermediate processing devices, are configured by the system to fulfil the request. Note that the system has the flexibility to use and configure address translation (network infrastructure dependent) for devices that are statically configured along the path of the service.

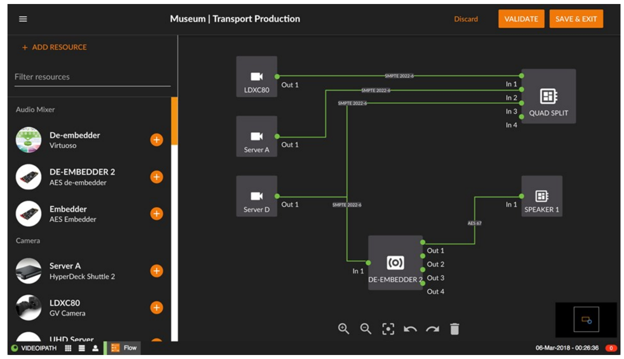
The system will always configure source/destination addresses to be used either from a pool or based on a static assignment. For every sender within the topology it is possible to define if it shall use dynamic addresses from a pool or static addresses. Address conflicts are highlighted in the topology application provided with the system.

Service Grouping

The VideoIPath orchestrator supports service groups that can contain multiple individual connections. This may be used to combine routing of video, audio and data as required or chain together connections in multiple domains or regions. When creating a service group in VideoIPath this is represented as a service definition consisting of multiple individual connections, which is the smallest entity for provisioning within VideoIPath.

Service groups are constructed in the Flow app, which provides a graphical interface to design logical flows between resources without considering how they are physically connected to the network infrastructure. However, the REST API also provides the possibility to define groups externally and instantiate.

An example of a chained service definition from the Flow app is shown below (graphical representation):



Within the Flow app, several pre-planned service definitions may be stored as templates and instantiated on demand (either immediately or scheduled). Activation of groups is available over the REST interface, which means that an external system can activate groups in a single operation.

The VideoIPath orchestrator supports scheduling of groups that is required to effectively share resources. The planned end time can be extended if necessary, but in this case, there may be resource conflicts that may have to be resolved manually. The system does provide detailed view of potential resource conflicts and supports a validation (dry run) capability, which allows the user to check for any conflicts while designing the service definition.

Service Assurance

The VideoIPath system also offers a comprehensive set of monitoring capabilities. This ranges from device level to full end-to-end service monitoring and is supported by the following applications. The functionality also ties in with the system's redundancy controller, which can take actions based on service affecting alarms. This includes the ability to reroute services along an alternative path.

• **Alarms:** The Alarms app provides a UI for managing current and historic alarms. The alarms are presented in a tabular format with one row per alarm and columns showing different attributes. Alarms may be searched, filtered and grouped from the user interface

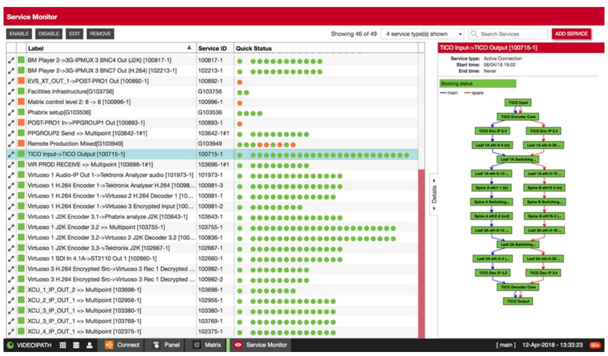
• **Service Monitor:** The Service Monitor is a highly useful tool that allows the operator to view end-to-end service status at a glance. Any issues related to a service is highlighted along with the most probable cause.

• **Dashboard:** The Dashboard app provides a customizable view of various monitoring data. The UI is based on widgets that may be added and customized according to preference. The interface may be used to show service and device status, performance, bandwidth utilization, etc.

• **Analytics:** The Analytics app allows the operator to define metrics (counters) that should be collected from the media nodes or network fabric. The analytics capabilities will be significantly improved as part of the roadmap commitment that is proposed for this project.

The system automatically correlates all alarms received or generated by the system against running services, to detect and display the service impact. The alarming behaviour is also customizable using service templates that may be configured to fit the monitoring needs. This functionality is used extensively in existing deployments for presenting SLA impact (loss of service, redundancy or protection), suppression and hysteresis of alarms.

Screenshot of the Service Monitor app:



All monitoring information is also available on the REST API for integration with other systems. There are several examples of such integrations e.g. with the DataMiner monitoring system. In addition, the alarm information may be forwarded on SNMP to other northbound systems. Alarms are forwarded on a normalized format and enriched with service impact information.

Software Defined Networking (SDN)

The VideoIPath system includes Software Defined Networking capabilities, which is generic and agnostic to the protocols used to control the underlying network infrastructure. The system can work with a variety of different SDN control protocols such as NETCONF, OpenConfig, Openflow or vendor specific APIs.

There are many advantages to opting for an SDN routing approach. First off, it guarantees a much higher level of performance when compared to automatic routing. As the management and orchestration software has all the required information about how endpoints are interconnected and the processing power to make the switching decisions fast. The software is also in control of every media flow, which means it is much more aware of, and much better at dealing with, existing and even planned (scheduled productions) bandwidth requirements.

It is even beneficial from a protection and security perspective. The management and orchestration software know all about the network topology and how to control the routing, which means it can easily create path diversity and dual paths to protect against failures, and can also fully control which destination is allowed to receive a particular service, thereby reducing the security risk substantially.

Deployment

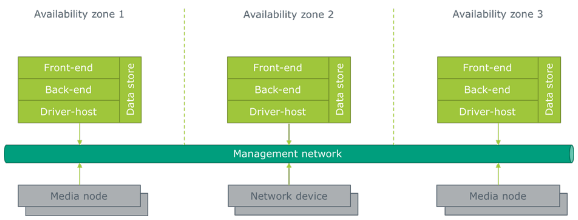
Existing deployments of VideoIPath are using a master/slave high-availability configuration across two different data centres. The most relevant scaling parameter for the VideoIPath system is the number of services, and it is currently supported to have 10000 services per server. In addition, each server can support up to 1000 devices directly.

The latest release 7 of VideoIPath introduces a new cluster architecture where the system may be scaled linearly by adding additional servers. A cluster should always include 2n+1 number of servers for redundancy and can be geographically distributed. The minimum system would consist of 3 servers distributed across 3 different data centres (or availability zones).

To protect against hardware hosting failures, the existing system is based on the RedHat high-availability add-on, which constantly monitors the state of the local server and communication with other servers in the cluster. If a failure is detected the corrective action is based on rules applied by the so-called pacemaker component of the RedHat high-availability add-on.

VideoIPath 8 takes a different approach where all servers in the cluster are active at all times, and not activated/deactivated on demand. The software is already designed to operate in such an environment using the Akka distributed processing framework. This system requires a quorum (majority vote) to perform its work, but will increase resilience to tolerate failures, since there is no need to

activate/deactivate components in such a cluster environment.



Each server in the cluster runs the same services, which is divided into front-end, back-end, driver-host and datastore. The frond-end is responsible for all UI integration, while the backend performs all processing within the server. The driver-host runs a number of driver instances for controlling media nodes or network devices

The datastore is replicated within the system to ensure that there are always two copies of all data. A media node or network device is always connected to one of the servers at any point in time, and if a failure occurs the node or device is moved to another server.

Security

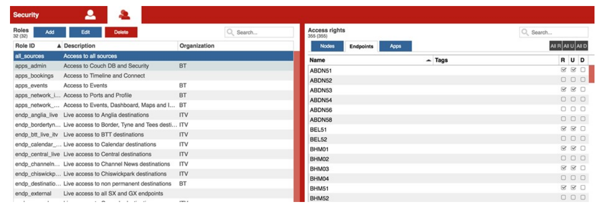
The VideoIPath system includes a role-based security system, where individual users are assigned to roles that control their access rights. Authentication must be provided before login to the system is allowed. The authentication may be done either locally against users stored in the system database (all passwords are securely encrypted) or against central LDAP/Active Directory servers.

The security features in VideoIPath provide a range of possibilities for defining users, roles and rights. This provides the administrator with detailed control over which users can access which devices, endpoints and applications. In addition, VideoIPath includes a feature where services can be locked on an individual basis, to prevent anyone from accidently removing the service.

The role-based security model may be used to grant different levels of access to individual users depending on their assigned roles. The roles may be defined as resource and function roles, relating to which resources the user can access and what functions. The roles may be created and defined by the operator according to the security needs for the orchestrator application. The role-based security model is flexible and can be customized for internal and external access to the system, including machine-to-machine integration using the API.

The system has a full audit log of user activity related to provisioning of services. This is currently being expanded to cover all user-initiated actions within the system including successful and failed login attempts.

The figure below shows how access rights may be assigned to roles in the Security app:



This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 866656. This document reflects only the author's view and the European Commission is not responsible for any use that may be made of the information it contains.