NDI 5.6 White Paper

A guide to getting the most
out of NDI and your network.

Sep 2023

Table of Contents

NDI White Paper

A guide to getting the most out of NDI and your network.

This paper is intended to deliver the essential facts with best practices for professionals familiar with networking devices and concepts.

The wonderful thing about NDI (Network Device Interface) is that it can be utilized on almost any Gigabit network. However, as production needs grow, additional considerations will be required, which this paper will cover.

Overview

NDI is a connectivity technology that enables sharing of video, audio, and metadata across a local area network (LAN). It is used by millions worldwide and has been adopted by more media organizations than any other IP standard, creating the industry's largest ecosystem of products connecting video over IP.

We believe **there is no video without connection**. The future of video is one in which content is transferred easily and efficiently via the Internet Protocol (IP), and this global network will largely supplant current industry-specific connection formats like HDMI, SDI, etc., in any type of video workflow or production pipeline.

NDI is removing the limits to video connectivity by allowing multimedia systems to identify and communicate with one another over IP and to encode, transmit, and receive many streams of high-quality, low latency, frame-accurate video and audio, and exchange metadata in real-time.

NDI can benefit any network-connected product, including video mixers, graphics systems, video cameras, capture cards, multimedia players, and many other devices and software.

NDI operates bi-directionally over a standard GigE network with many streams on a shared connection. Its encoding algorithm is resolution and frame-rate-independent, supporting 4K resolutions and beyond, along with unlimited floating-point audio channels and custom metadata.

NDI enables transitioning to an incredibly versatile IP video pipeline without negating existing investments in SDI and HDMI cameras and infrastructure or costly new high-speed network infrastructures.

1. NDI Discovery and Registration

Zero configuration in AV signal distribution

One of the biggest issues in AV distribution in the IP world is that equipment is not identifiable by its physical connection. In networking, every connected device needs to have a unique address so another device, hardware, and applications can reach it.

But the network physical connection is dynamic and not related at all to the equipment address. For that reason, in a large network with hundreds (or thousands) of devices with addresses, it becomes difficult to find and interconnect equipment. NDI offers two different options for a zero-configuration discovery and registration: **mDNS** and **Discovery Service**.

1.1. mDNS

Sending and receiving video streams across an IP network requires applications that support video and can discover receiving applications that are looking for video.

NDI resolves host names to IP addresses over the LAN and does so automatically. When you start an application that sends NDI, the devices that can receive NDI become aware instantaneously. While this is a typical function on almost all networks, there are some cases where it is important to know how this works to properly configure networks utilizing managed data flows protocols.

As default, NDI utilizes mDNS (multicast Domain Name System)[1] to create the zero-configuration environment for discovery. This service sends an IP multicast message that asks the host to identify itself. The target machine then multicasts a message that includes its own IP address. This multicast is seen by all NDI-receiving machines on the subnet, which then use the information in that message to update their own caches.

These multicast queries are sent to a multicast address, and thus, no single device is required to have global knowledge.

When a service or device sees a query for any service it recognizes, it provides a DNS response with the information from its cache. The primary benefit of using mDNS is that it requires little or no administration to set up. Unless the network is specifically configured not to allow mDNS, NDI sources will be discovered. This format works when no infrastructure is present and can span infrastructure failures.

The mDNS Ethernet frame is a multicast UDP packet that broadcasts to[2]:

[1] Apple's mDNS is published as a standards track proposal (RFC 6762) https://tools.ietf.org/html/rfc6762
[2] https://en.wikipedia.org/wiki/Multicast_DNS

MAC Adress	01:00:5E:00:00:FB (for IPv4)
Ipv4 Address	224.0.0.251
UDP Port	5353

Choosing the network location type on Windows devices is critical for the successful discovery and registration of NDI. Typically, the first time a Windows machine is connected to a network, a dialog window appears that allows the user to choose the network location type: Private or Public. By default, Windows sets a new network location to Public.

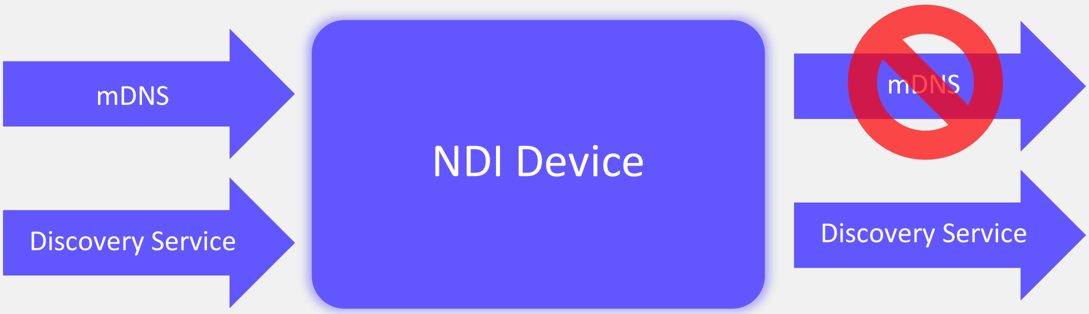
This location is designed to keep machines from being visible and responding to broadcast pings. This location type also affects mDNS responses and keeps NDI video streams from being discovered and registered on the network.

Network locations should be set to Private for successful discovery and registration of NDI. The Domain network location is used for domain networks, such as those at enterprise workplaces. The network administrator controls this type of network location, and it cannot be selected or changed. In this type of configuration, mDNS discovery must be allowed at the domain level. Because mDNS uses a link-local multicast address, its capacity is limited to a single physical or logical LAN.

1.2. Discovery Service

NDI Discovery server is a command line application available for Windows, MacOS, and Linux. The NDI Discovery service is designed to allow you to replace the automatic discovery NDI uses with a server that operates as a centralized registry of NDI sources. This can be very helpful for installations where you wish to avoid having significant mDNS traffic for a large number of sources. It can also be useful when multicast is not possible or desirable; it is very common for cloud computing services not to allow multicast traffic. When using the Discovery service, NDI can operate entirely in unicast mode and thus in almost any installation. The Discovery server supports all NDI functionality, including NDI groups. Clients should be configured to connect with the Discovery service instead of using mDNS to locate sources.

When there is a Discovery server, NDI applications will use both mDNS and the Discovery server to find and receive sources on the local network that are not on machines configured to use discovery.



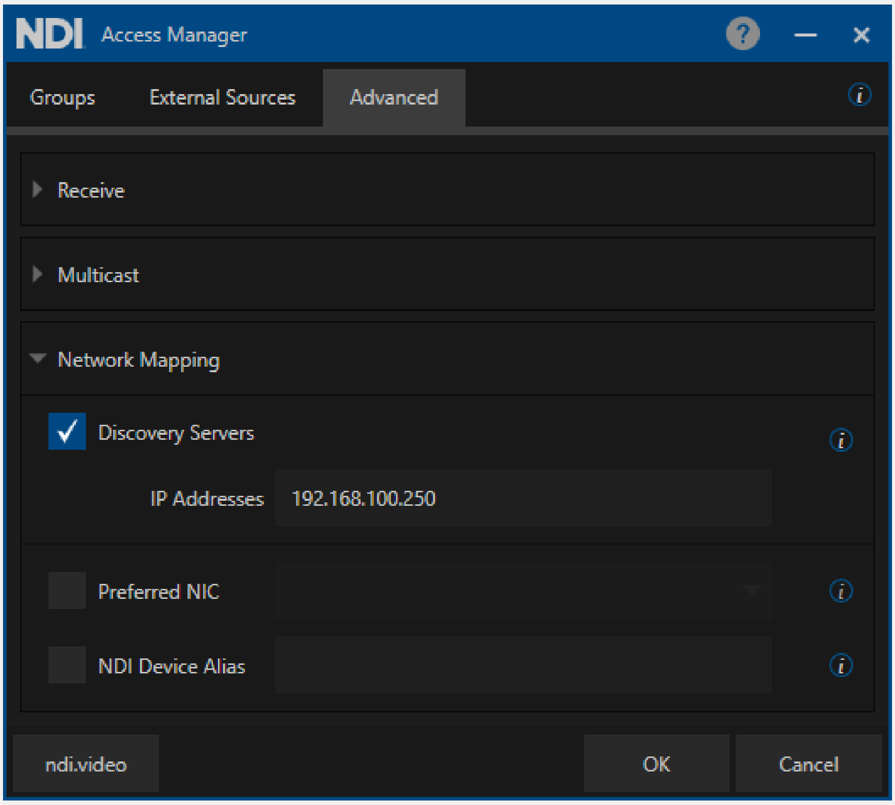
For senders, if a Discovery service is specified, then mDNS will not be used; these sources will only be visible to other finders and receivers configured to use the Discovery server.

To configure the Discovery service for NDI clients, **you may use Access Manager (included in the NDI Tools bundle)** to enter the IP address of the Discovery server machine.

Within NDI version 5, there is full support for redundant NDI Discovery servers. When configuring a Discovery server, it is possible to specify a comma-delimited list of servers (e.g., “192.168.10.10, 192.168.10.12”), and then they will all be used simultaneously. If one of these servers then goes down, as long as one remains active, then all sources will always remain visible, no matter what the others do then all sources can be seen.

This multiple-server capability can also be used to ensure entirely separate servers to allow sources to be broken into separate groups, which can serve many workflows or security needs.

Once two NDI devices have discovered each other on the network, video can be passed from the sending device to the receiving device. After the compression of the video, the NDI sending device opens a session to the receiving NDI device. At this point, we have two endpoints that consist of an IP address and a port number.



☑ Discovery Servers

IP Addresses 192.168.100.250

□ Preferred NIC

□ NDI Device Alias

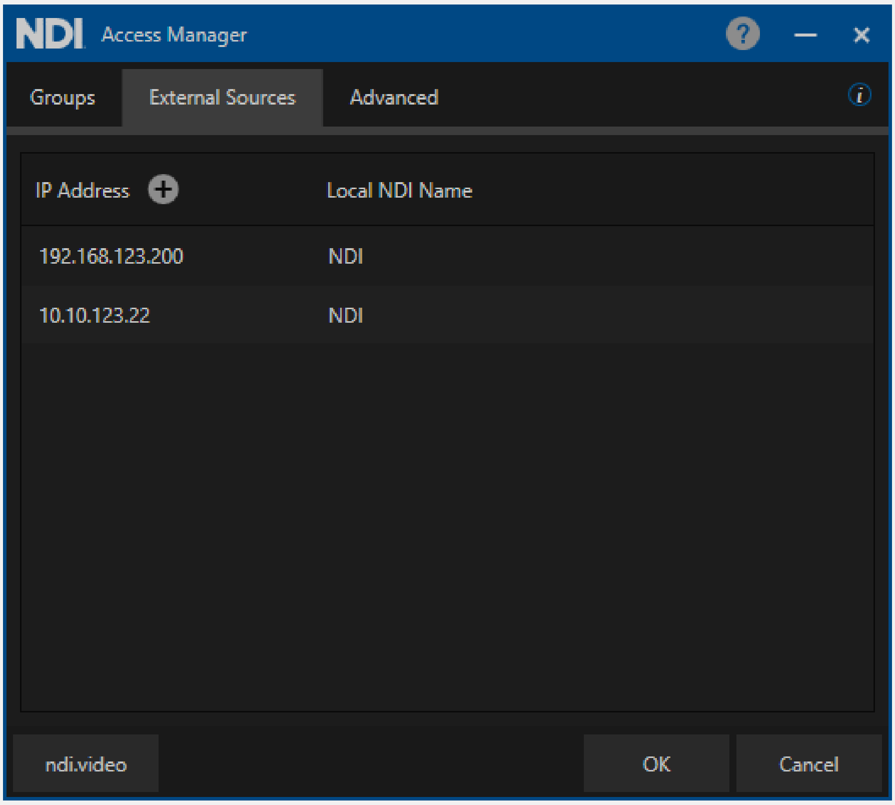
In Windows and MacOS the Discovery Server addresses are configured in the **Advanced Feature Tab.** In Linux the addresses of the NDI Discovery Servers can be manually added in the NDI configuration file located in the home directory of the effective user: "ndi-config.v1.json"

Here is the way to manually set up the **Discovery Service** in the configuration file:

```
"networks": {
    "ips": "",
    "discovery": "192.168.10.10,192.168.10.12",
```

1.3. Manual connection

One approach to manually interconnect NDI devices is to specify the IP address of the transmitter in the receiver.



IP Address	Local NDI Name
192.168.123.200	NDI
10.10.123.22	NDI

In Windows and MacOS, this can be achieved using the NDI Access Manager in the External Sources feature. Several NDI hardware decoders also support this functionality.

For Linux, the IP addresses of NDI senders can be added manually in the NDI configuration file called "ndi-config.v1.json." This file is in the home directory of the user currently logged in.

Specifying the IP address of an NDI source allows the receiver to receive NDI sources that are in a different subnet and may not be discoverable by mDNS (Multicast DNS). This method enables the reception of NDI sources that might be otherwise inaccessible due to network configurations or limitations.

In Linux manual connections can be added in the NDI configuration file located in the home directory of the effective user: "ndi-config.v1.json"

Here is the way to manually set up NDI sources in the configuration file:

```
"networks": {
    "ips": "192.168.123.200,10.10.123.22,",
    "discovery": "",
```

1.4. NDI Groups

NDI groups enhance the efficiency and management of NDI-based workflows by providing a structured way to organize and control the visibility and access of NDI sources and destinations within a network.

NDI devices support two different kinds of Groups: **Send** and **Receive**.

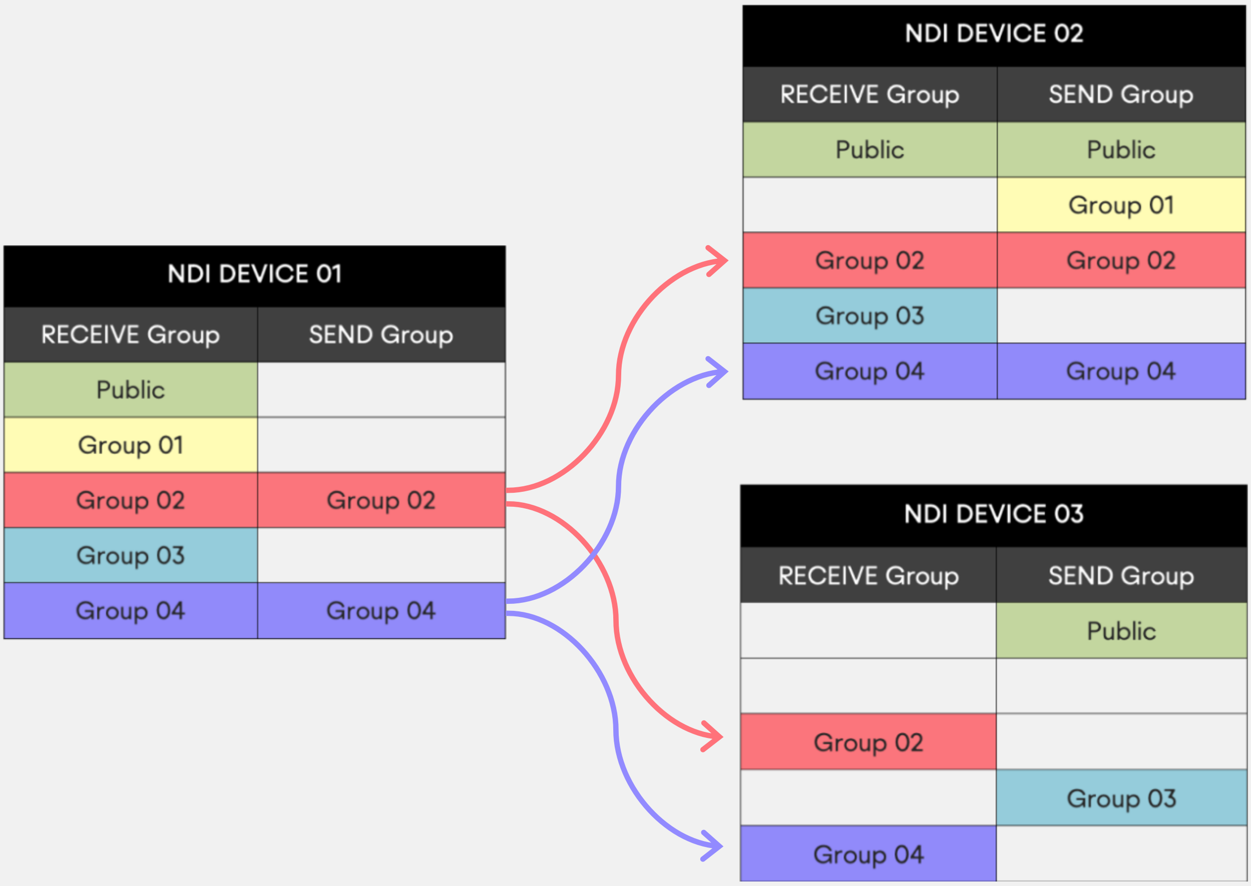
A device can be part of different Groups, some Groups only in the Send or Receive mode:

NDI DEVICE 01	
RECEIVE Group	SEND Group
Public	
Group 01	
Group 02	Group 02
Group 03	
Group 04	Group 04

NDI DEVICE 02	
RECEIVE Group	SEND Group
Public	Public
	Group 01
Group 02	Group 02
Group 03	
Group 04	Group 04

NDI DEVICE 03	
RECEIVE Group	SEND Group
	Public
Group 02	
	Group 03
Group 04	

In this scenario, **NDI Device 01** is sending discovery information in Groups 02 and 04. Devices part of the **Receive Groups 02** and **04** can discover and receive NDI streams from **Device 01.**

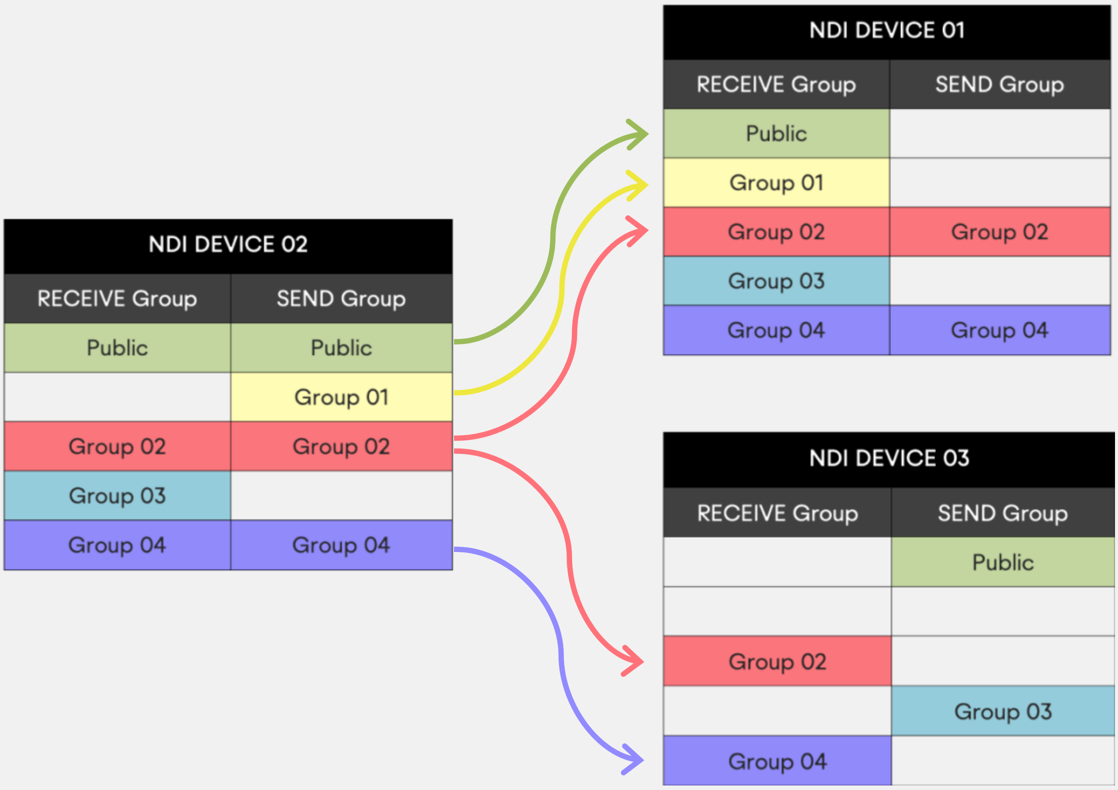


NDI DEVICE 01	
RECEIVE Group	SEND Group
Public	
Group 01	
Group 02	Group 02
Group 03	
Group 04	Group 04

NDI DEVICE 02	
RECEIVE Group	SEND Group
Public	Public
	Group 01
Group 02	Group 02
Group 03	
Group 04	Group 04

NDI DEVICE 03	
RECEIVE Group	SEND Group
	Public
Group 02	
	Group 03
Group 04	

NDI Device 02 is sending discovery information in Groups Public, 01, 02 and 04.

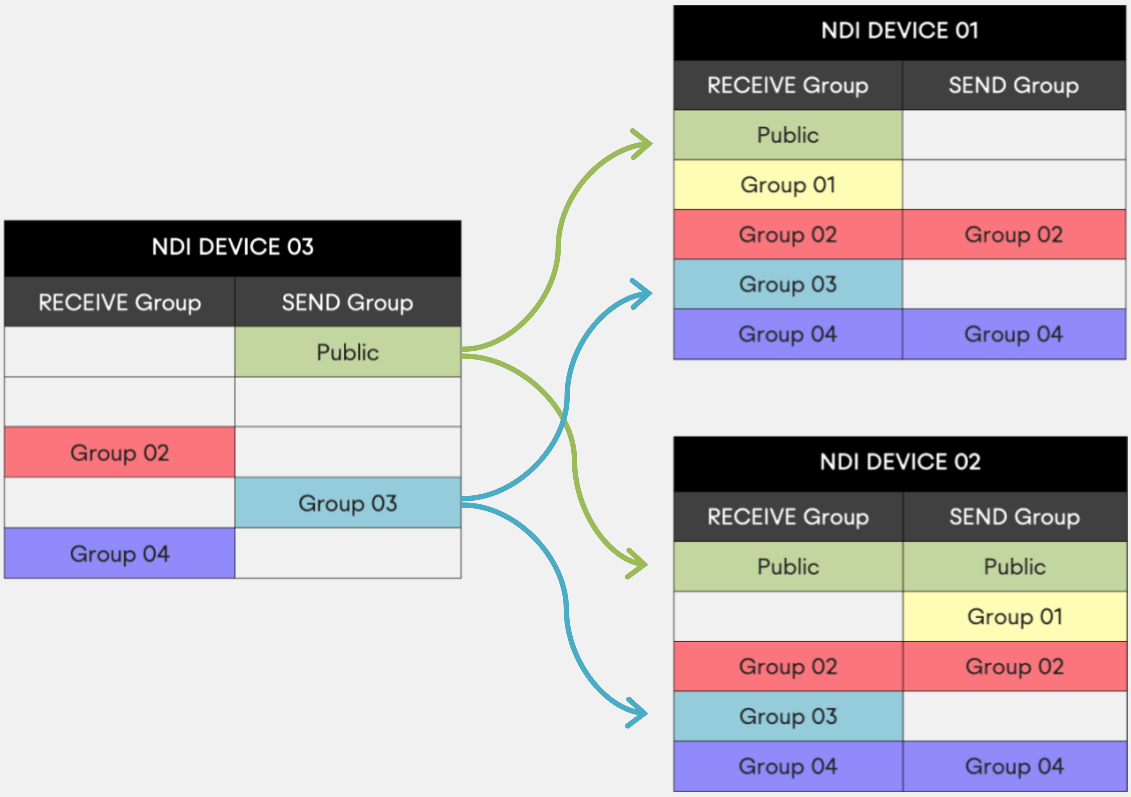


NDI DEVICE 02	
RECEIVE Group	SEND Group
Public	Public
	Group 01
Group 02	Group 02
Group 03	
Group 04	Group 04

NDI DEVICE 01	
RECEIVE Group	SEND Group
Public	
Group 01	
Group 02	Group 02
Group 03	
Group 04	Group 04

NDI DEVICE 03	
RECEIVE Group	SEND Group
	Public
Group 02	
	Group 03
Group 04	

NDI Device 03 is sending discovery information in Groups Public, and 03.

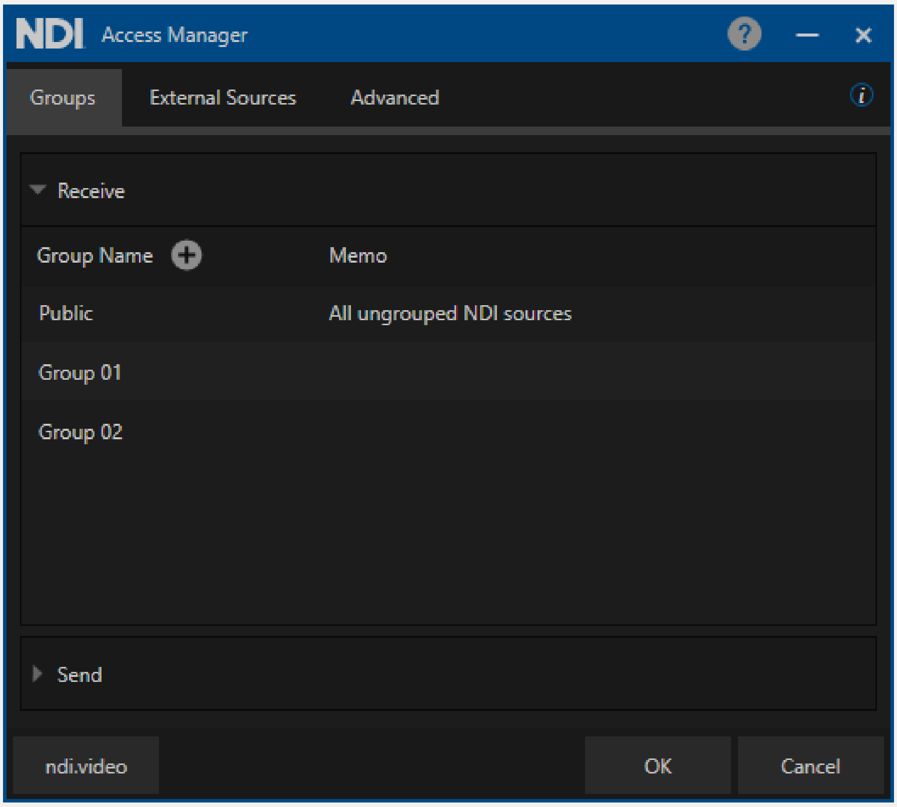


NDI DEVICE 03	
RECEIVE Group	SEND Group
	Public
Group 02	
	Group 03
Group 04	

NDI DEVICE 01	
RECEIVE Group	SEND Group
Public	
Group 01	
Group 02	Group 02
Group 03	
Group 04	Group 04

NDI DEVICE 02	
RECEIVE Group	SEND Group
Public	Public
	Group 01
Group 02	Group 02
Group 03	
Group 04	Group 04

There are different ways to define Groups in NDI Devices:



Group Name	Memo
Public	All ungrouped NDI sources
Group 01	
Group 02	

In MS Windows and MacOS, Groups are defined in the NDI Access Manager, which is part of the free NDI Tools:

The Groups string for each Send and Receive operation must not exceed 248 bytes in length, which means that the total length of the combined Group names should not exceed 248 characters.

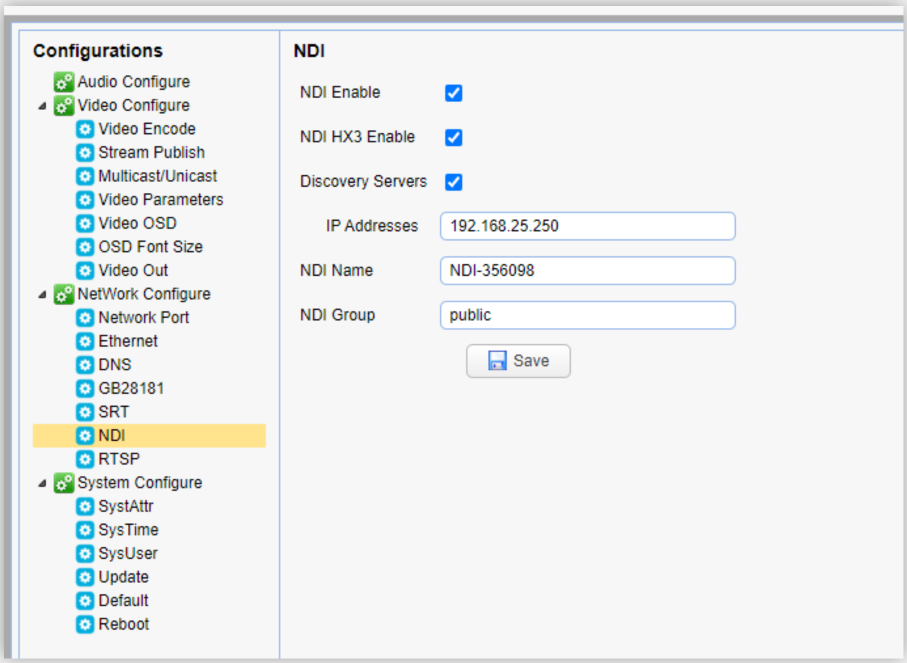
In Linux NDI Groups can be defined in the NDI configuration file located in the home directory of the effective user: "ndi-config.v1.json"

Here is the way to configure Groups in the configuration file:

```
},
  "groups": {
    "send": "Public",
    "recv": "Public,Group 01,Group 02"
  },
```

Hardware NDI Devices must support NDI Groups to be compliant with the NDI standard specifications.

Here are some examples of hardware devices with NDI Groups support:



NDI

NDI Enable ☑

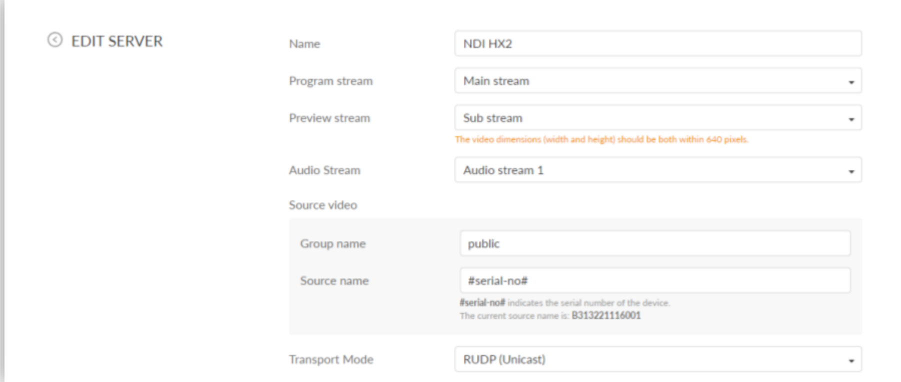
NDI HX3 Enable ☑

Discovery Servers ☑

IP Addresses: 192.168.25.250

NDI Name: NDI-356098

NDI Group: public



Name: NDI HX2

Program stream: Main stream

Preview stream: Sub stream

The video dimensions (width and height) should be both within 640 pixels.

Audio Stream: Audio stream 1

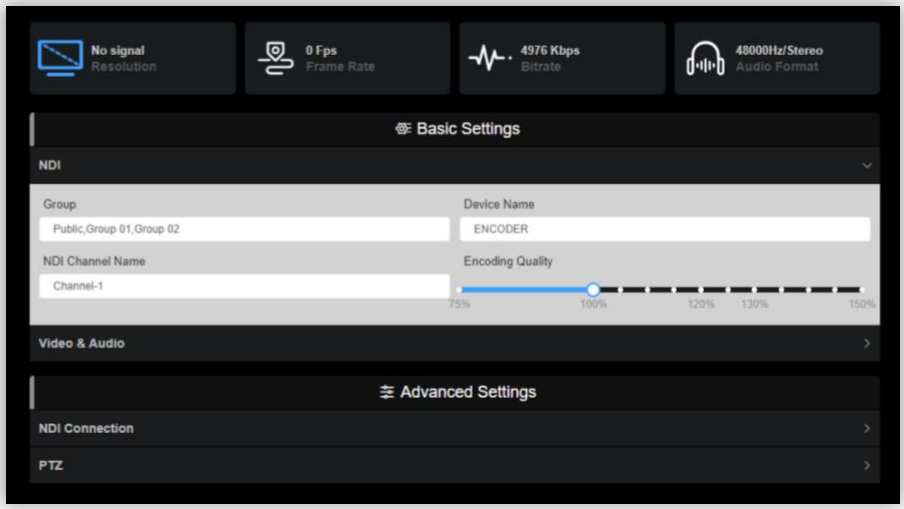
Source video

Group name: public

Source name: #serial-no#

#serial-no# indicates the serial number of the device.
The current source name is: B313221116001

Transport Mode: RUDP (Unicast)



Group: Public,Group 01,Group 02

Device Name: ENCODER

NDI Channel Name: Channel-1

Encoding Quality: 100%

2. NDI Protocols

2.1. Reliable UDP – NDI v5

In NDI version 5 the default communication mechanism is a Reliable UDP protocol that represents the state-of-the-art communication protocol that is implemented by building upon all the experience we have seen in the real world with NDI across a massive variety of different installations

Reliable UDP, also known as RUDP, is a transport protocol that combines the advantages of UDP's low latency and simplicity with the reliability of TCP (Transmission Control Protocol). It is designed specifically for real-time multimedia applications, where maintaining the timeliness of data is crucial.

In the context of NDI, Reliable UDP is employed to ensure that video and audio streams are delivered reliably and with minimal delay. It achieves this by implementing several mechanisms:

Sequencing: Reliable UDP assigns a sequence number to each packet it sends. This allows the receiving end to detect missing or out-of-order packets and request retransmissions if necessary.

Retransmissions: If a packet is lost or arrives out of order, the receiving end can request a retransmission of the missing packet(s) using the sequence number information.

Flow control: Reliable UDP incorporates flow control mechanisms to manage the rate of data transmission. This prevents overwhelming the network or the receiving device with more data than it can handle, ensuring a smoother streaming experience.

Congestion control: RUDP also includes congestion control algorithms to prevent network congestion and avoid unnecessary packet loss. It dynamically adjusts the transmission rate based on network conditions, maintaining optimal throughput without overwhelming the network.

2.2. Multipath TCP – NDI v4

This protocol permits transport across multiple NICs and all network paths, it is intended to use hardware-accelerated network adapters with adaptive bandwidth sharing across NICs.

Multipath TCP is a transmission protocol that offers advantages such as maximizing throughput, optimizing resource usage, and enhancing network redundancy. It can seamlessly integrate multiple network pathways, including wireless and mobile networks. It is especially efficient when used with NDI (equipment that utilizes multiple Gigabit connections to exchange a large number of NDI streams.

However, in scenarios where 10Gbit interfaces are connected with 1Gbit interfaces, Multipath TCP's efficiency is compromised. This is primarily due to network switches being unable to effectively manage network congestion in such situations. As a result, the protocol may not perform optimally in these specific network configurations.

2.3. UDP with Forward Error Correction – NDI v3

This alternative protocol to TCP is used when reliable delivery of data packets is not required. UDP is typically used for applications where timeliness is of higher priority than accuracy, such as streaming media, teleconferencing, and voice-over-IP (VoIP). Forward error correction (FEC) is a method of obtaining error control in data transmission in which the source (transmitter) sends redundant data and the destination (receiver).

UDP (User Datagram Protocol) with Forward Error Correction (FEC) is a beneficial approach when the network is prone to errors or not entirely reliable. It provides a solution for error correction when data packets get lost or corrupted during transmission.

However, it's important to note that using UDP with FEC requires additional computational processing on the receiver side. The receiver needs to implement algorithms and mechanisms to manage the error correction process. This involves decoding the received data and applying error correction techniques to recover any lost or corrupted packets.

2.4. Single TCP – NDI v1

This network communications protocol enables two host systems to establish a connection, exchange data packets, and ensure data is delivered intact to the correct destination. TCP is typically grouped with IP (Internet Protocol) and is collectively known as TCP/IP.

Single-TCP is supported on all NDI versions. While the other transmission modes are likely to perform better, this mode does offer baseline compatibility for all NDI clients.

3. NDI Related Network Ports

Port	Type	Use
5353	UDP	This is the standard port used for mDNS communication and is always used for multicast sending of the current sources onto the network.
5959	TCP	NDI Discovery Server is an optional method to have NDI devices perform discovery. This can be beneficial in large configurations when you need to connect NDI devices between subnets or if mDNS is blocked.
5960	TCP	This is a TCP port used for remote sources to query this machine and discover all the sources running on it. This is used, for instance, when a machine is added by an IP address in the access manager so that from an IP address alone, all the sources currently running on that machine can be discovered automatically.
5961 and up	TCP	These are the base TCP connections used for each NDI stream. For each current connection, at least one port number will be used in this range.
5960 and up	UDP	In version 5 and above, when using Reliable UDP connections, it will use a very small number of ports in the range of 5960 for UDP. These port numbers are shared with the TCP connections. Because connection sharing is used in this mode, the number of ports required is very limited and only one port is needed per NDI process running and not one port per NDI connection.
6960 and up	TCP/UDP	When using multi-TCP or UDP receiving, at least one port number in this range will be used for each connection.
7960 and up	TCP/UDP	When using multi-TCP, unicast UDP, or multicast UDP sending, at least one port number in this range will be used for each connection.
Ephemeral	TCP	Legacy to NDI v1 - The current versions (4.6 and later) no longer use any ports in the ephemeral port range.

4. Getting video across the network

Video, just like voice data in VoIP systems, is a very demanding data stream and will immediately expose a weakness in a network. The network must support multiple video, audio, and data streams in a reliable, synchronized manner without disruption. When delay, packet loss, and jitter reach thresholds where the video is impacted visually, the usefulness of that video drops to zero. It is important to understand the complexities of video in IP data networks to mitigate these factors.

Networks that are designed to move NDI video streams should be thought of as being primarily utilized for video. IP networks are, by their very nature, “best effort delivery” systems and were originally developed for the transport of data. By contrast to video, data services can function happily with packet retransmissions, lost packets, and even packets arriving out of order.

Video streams, while still data are much more rigid in their requirements. With the use of modern networking equipment and proper configuration, video can move across networks whilst still obtaining low latency, frame accuracy, and high-quality requirements necessary for live video production.

5. Network Layout

NDI is designed for use with standard consumer off-the-shelf (COTS) network infrastructure devices. Looking closely at the network topology and configuration will help to ensure that the maximum possible bandwidth is available.

When selecting a network switch, it is important to check the throughput speeds. Ensure that each port is full duplex (i.e., bi-directional communication) and that each port's upstream and downstream data speeds are at least 1 Gigabit per second (Gbps). In some cases, it is best to force the ports on managed switches to utilize 1 Gbps in contrast with Auto-Negotiation. The use of Auto-Negotiation can sometimes (mostly because non fully compatible Network Interfaces) result in 100Mb connections or even lower, which does not renegotiate until the port is flooded with traffic for some time. Also, poor termination of RJ-45 connectors can impact Auto-Negotiation.

The same suggestion applies when considering network switches that include 10 Gigabit per second ports. Many switches manufactured at the time of writing may share bandwidth across the backplane of multiple ports. Since these ports are generally reserved for linking to other switches, the specification for throughput may be listed differently than the Gigabit port section in the product documentation.

It is best to use switches from the same manufacturer, or ideally, the same model of switch, throughout a single subnet. This will simplify configuration and lessen the chances of compatibility and configuration issues.

6. Bandwidth

NDI operates most efficiently in a dedicated network with high bandwidth and high availability. This contrasts with unmanaged environments such as the public Internet or networks where video rides along with data without priority.

Gigabit (1000 Mbps) networks are essential in production workflows. A typical NDI stream consisting of 1080 60P video yields a data rate up to 150 Mbps per stream. This extremely efficient stream is designed to have very low latency and allows multiple streams to be stacked together on a single Gigabit network. Even so, a production environment may require more capacity based on a simultaneous number of NDI streams required.

The following tables are intended to calculate bandwidth needs based on video resolutions and frame rates. It should be noted, however, that NDI is not deterministic. The bandwidth needed for NDI should be based on the determination of the average utilization required[3].

[3] Bandwidth numbers are given as reference and are subject to change

6.1. NDI High Bandwidth based on SpeedHQ2 (8bit 4:2:2)

Resolution Framerate	Maximum Bandwidth Mbps	Proxy Resolution Framerate	Maximum Bandwidth Mbps
720 50p	96.94	640x360 60p	65.83
720 60p	105.83	640x360 60p	65.83
1080 50i	102.50	640x360 30p	18.75
1080 60i	112.50	640x360 30p	18.75
1080 50p	125.59	640x360 60p	65.83
1080 60p	132.14	640x360 60p	65.83
3840x2160 50i	158.33	640x360 30p	18.75
3840x2160 60i	171.42	640x360 30p	18.75
3840x2160 50p	223.80	640x360 60p	65.83
3840x2160 60p	249.99	640x360 60p	65.83

6.2. NDI High Bandwidth based on SpeedHQ7 (8bit 4:2:2:4)

Includes Alpha Channel

Resolution Framerate	Maximum Bandwidth Mbps	Proxy Resolution Framerate	Maximum Bandwidth Mbps
720 50p	121.18	640x360 60p	82.29
720 60p	132.29	640x360 60p	82.29
1080 50i	128.12	640x360 30p	23.43
1080 60i	140.62	640x360 30p	23.43
1080 50p	156.99	640x360 60p	82.29
1080 60p	165.17	640x360 60p	82.29
3840x2160 50i	197.91	640x360 30p	23.43
3840x2160 60i	214.28	640x360 30p	23.43
3840x2160 50p	279.76	640x360 60p	82.29
3840x2160 60p	312.49	640x360 60p	82.29

NDI High Bandwidth is I-frame only.

NDI encoders offer a proxy stream with lower resolution and higher compression, and applications can use the proxy stream for preview, reducing the network usage and processing requirements.

NDI HX h.264 (8bit 4:2:0)

Resolution Framerate	Maximum Bandwidth Mbps	Proxy Resolution Framerate	Maximum Bandwidth Mbps
720 50p	9.11	640x360 60p	3.99
720 60p	9.99	640x360 60p	6.00
1080 50i	9.66	640x360 30p	3.99
1080 60i	10.59	640x360 30p	3.99
1080 50p	14.20	640x360 60p	6.00
1080 60p	15.99	640x360 60p	6.00
3840x2160 50i	19.11	640x360 30p	3.99
3840x2160 60i	20.66	640x360 30p	3.99
3840x2160 50p	26.88	640x360 60p	6.00
3840x2160 60p	30.00	640x360 60p	6.00

6.3. NDI HX h.265 (8bit 4:2:0)

Resolution Framerate	Maximum Bandwidth Mbps	Proxy Resolution Framerate	Maximum Bandwidth Mbps
720 50p	6.33	640x360 60p	3.00
720 60p	6.99	640x360 60p	3.99
1080 50i	6.75	640x360 30p	3.00
1080 60i	7.39	640x360 30p	3.00
1080 50p	9.80	640x360 60p	3.99
1080 60p	10.99	640x360 60p	3.99
3840x2160 50i	13.22	640x360 30p	3.00
3840x2160 60i	14.33	640x360 30p	3.00
3840x2160 50p	18.77	640x360 60p	3.99
3840x2160 60p	21.00	640x360 60p	3.99

NDI HX GOP size recommendation is between 1 and 2 second.

6.4. NDI HX3 h.264 (8bit 4:2:0)

Resolution Framerate	Maximum Bandwidth Mbps	Proxy Resolution Framerate	Maximum Bandwidth Mbps
1080 50i	26.00	640x360 25/30p	3.00
1080 60i	31.00	640x360 25/30p	3.00
1080 50p	52.00	640x360 25/30p	3.00
1080 60p	62.00	640x360 25/30p	3.00
3840x2160 50p	92.00	640x360 25/30p	3.00
3840x2160 60p	110.00	640x360 25/30p	3.00

6.5. NDI HX3 h.265 (8bit 4:2:0)

Resolution Framerate	Maximum Bandwidth Mbps	Proxy Resolution Framerate	Maximum Bandwidth Mbps
1080 50i	20.00	640x360 25/30p	3.00
1080 60i	25.00	640x360 25/30p	3.00
1080 50p	41.00	640x360 25/30p	3.00
1080 60p	50.00	640x360 25/30p	3.00
3840x2160 50p	70.00	640x360 25/30p	3.00
3840x2160 60p	84.00	640x360 25/30p	3.00

NDI HX3 stream specification:

GOP size must be 20 frames
Glass to Glass latency, less than 100ms
I Frame request response, less than 80ms

6.6. NDI Proxy and bandwidth optimization.

By default, an NDI Sender has the capability to generate two types of streams: a full-quality stream and a proxy stream. A receiver can readily activate the proxy stream. A simple way to visualize a proxy stream is by utilizing NDI Studio Monitor and enabling the Low Bandwidth mode within the Video settings menu.

The Proxy stream serves as a means for an application to enhance the efficiency of NDI distribution across the network. For instance, an NDI-based Video Mixer can leverage the proxy stream for sources that are intended for use in a preview monitor rather than an actual output. Then, when a source is transitioned to an output, the application can seamlessly switch from the proxy stream to the full bandwidth stream.

This approach empowers receivers to effectively manage a more significant number of NDI sources while utilizing less network bandwidth.

7. Network Interface Settings

NDI is designed to enable successful video transport using the default configuration of network interface drivers; however, most recent network interface drivers do support the configuration of advanced properties that can help optimize NDI transmission.

Consider the following adjustments but note that adjusting individual adapters can significantly affect performance and reliability positively and negatively. It is important to consider testing performance with a network analyzer before and after each setting change. The following adjustments are intended to help; however, performance will depend on network and usage (names and available settings vary between vendors, adapter models, and even between different driver versions):

7.1. Speed and Duplex

This setting allows for the selection of the desired speed and duplex of the network adapter. Usually, this is set to Auto Negotiation. This setting should be set to 1 Gbps Full Duplex or higher if supported to ensure the maximum available throughput.

7.2. Energy Efficient Ethernet

When enabled, this allows the adapter to engage power-saving features while keeping connections active. This technology uses the standard IEEE 802.3az to allow for less power during periods of low data activity. Adapters that utilize the IEEE 802.3az standard should have no impact on the performance of NDI; however, some integrated circuits exist that were developed before the standard was finalized or do not adhere to the standard at all. In these cases, it is best to disable the energy efficiency while determining the best network optimization.

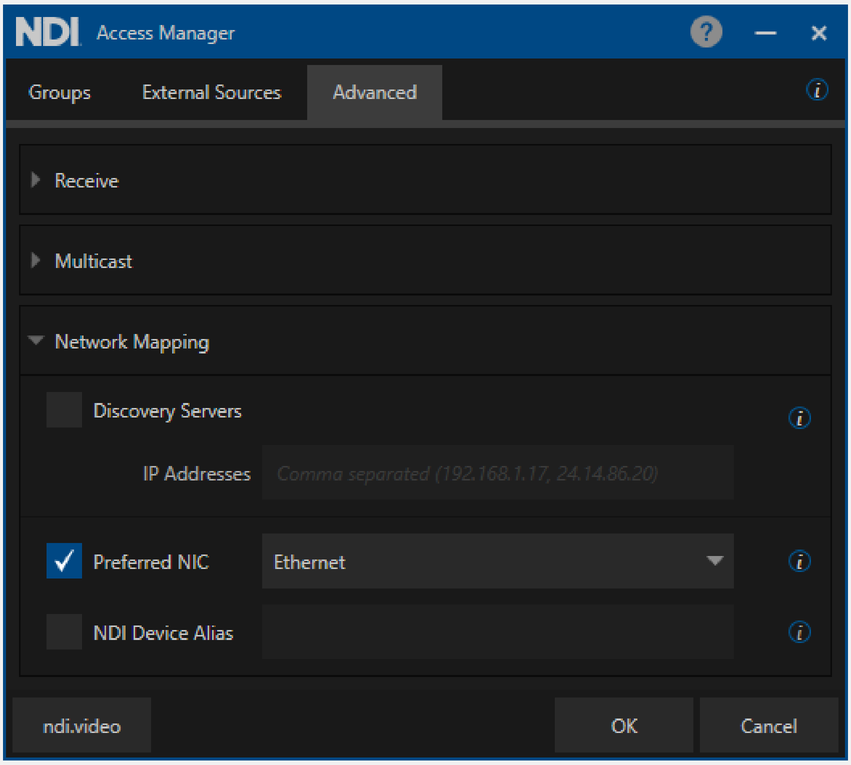
8. NIC Selection

Starting in NDI version 5, this lists all the network adapters that will be used for network transmission.

One or more NICs can be used to transmit and receive video and audio data. This capability can be used to ensure that the NDI primary stream data remains on a particular group of network adapters, for instance, allowing you to ensure that dedicated audio is on a separate network card from the NDI video. It is generally preferred that you let NDI select the network adapters automatically, which can smartly select which to use and how to choose the ones that result in the best bandwidth.

While in some modes, NDI can automatically balance bandwidth across multiple NICs, it usually is better for you to use NIC teaming at a machine configuration level which can result in much better performance than what is possible in software. If this setting is misconfigured to specify NICs that might not exist, then NDI might fail to function correctly.

Also, please note that the operation of computer systems that are separately on entirely different networks with different IP address ranges is often not handled robustly by the operating system, and NDI might not fully function in these configurations.



□ Discovery Servers

IP Addresses: Comma separated (192.168.1.17, 24.14.86.20)

☑ Preferred NIC: Ethernet

□ NDI Device Alias

NIC Selection in configuration is part of NDI Access Manager (Windows).

In MacOS and Linux the NIC Selection can be manually added in the NDI configuration file located in the home directory of the effective user: "ndi-config.v1.json"

Here is the way to manually setup NIC Selection in the configuration file:

```
},
  "adapters": {
   "allowed": [
    "192.168.30.8,10.10.122.123"
   ]
```

Using the NIC selection, in combination with the Discovery Server, is ultimately the best solution for controlling the NDI network.

9. Encoding and Decoding

9.1. SpeedHQ Compression

NDI uses compression to enable the transmission of many video streams across existing infrastructure, specifically discrete cosine transform (DCT), which converts video signals into elementary frequency components. This method of compression is commonly used in encoding formats and mezzanine codecs within the industry.

One of the most efficient codecs in existence, NDI achieves significantly better compression than many codecs that have been accepted for professional broadcast use. On a typical, modern Intel-based i7 processor, the codec can compress a video stream to the following benchmarks:

The NDI codec's peak signal-to-noise ratio (PSNR) exceeds 70dB for typical video content.

Uniquely and importantly, NDI is the first ever codec to provide multi-generational stability.
This means there is no further loss once a video signal is compressed. As a practical example, generation 2 and generation 1000 of a decode-to-encode sequence would be identical.

The NDI codec is designed to run very fast and is largely implemented in hand-written assembly to ensure that the process of compressing video frames occurs as quickly as possible. Latency is both a factor of the network connection and the endpoint products. NDI has a technical latency of 16 video scan lines, although in practice, most implementations would be one field of latency. Hardware implementations can provide full end-to-end latency of within 8 scan lines.

9.2. NDI HX

NDI is available in some devices and applications using a different compression codec than NDI High Bandwidth. This format is known as NDI HX. Devices using this NDI format will be labeled with the HX moniker. HX offers similar video quality at a much lower bit rate, which can be useful in contexts with limited bandwidth, like Fast Ethernet networks, Wi-Fi, or WAN connections.

NDI HX is commonly found in hardware devices, like PTZ cameras and mobile phones, but it is possible to have HX in software applications as well. Software applications using NDI HX will leverage the GPU on the computer for enhanced encoding performance. For this reason, having a good GPU on the system is an advantage.

There are two variations of NDI HX: NDI HX and NDI HX3; both can be decoded by software applications thanks to the codec provided by the NDI Tools.

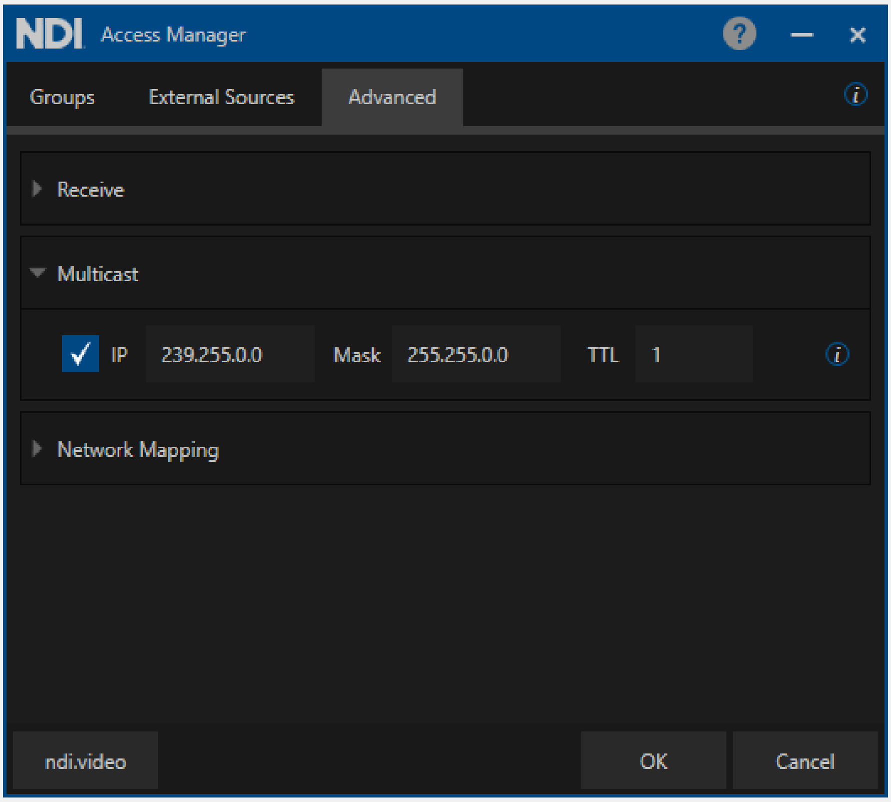
10.Multicast

To ensure packet loss, NDI supports multicast-based video sources using multicast UDP with forward error correction. Multicasting allows for a single NDI source to be delivered to multiple receivers by replicating the NDI packets from the sender to any number of receivers. It is essential to be aware that using multicast on a poorly configured network can produce undesirable results and cripple network performance. For this reason, multicast sending is disabled by default.

For successful multicasting, the use of Internet Group Management Protocol (IGMP) is encouraged. IGMP allows the receiving NDI systems to request access to the sender. Without IGMP querying and snooping, Multicast traffic is treated the same as broadcast transmission resulting in packet forwarding to all ports on the network. With IGMP snooping, multicast NDI traffic is forwarded only to the receivers that subscribe to the multicast NDI stream.

NDI subscribes to a multicast group and will unsubscribe when that stream is no longer needed. The management of multicast subscriptions is handled by a routing querier on the network.

While video and audio data are delivered to the network via multicast delivery, each receiver also connects to the sender via a unicast TCP connection for bi-directional communication of metadata (e.g., PTZ control, tally, etc.).



☑ IP 239.255.0.0 Mask 255.255.0.0 TTL 1

In Windows and MacOS machines, multicast can be setup in the NDI Access Manager.
In Linux, Multicast can be configured in the NDI configuration file located in the home directory of the effective user: "ndi-config.v1.json"

Here is the way to configure Multicast in the configuration file:

```
},
  "multicast": {
   "send": {
     "ttl": 1,
     "enable": true,
     "netmask": "255.255.0.0",
     "netprefix": "239.255.0.0"
   }
```

11.Synchronization

NDI transmitters or receivers do not need any synchronization method to be connected and work. **An NDI infrastructure can perfectly work "sync free"** without the complexity and cost of network infrastructure that supports a synchronization layer. However, with NDI, software developers and hardware manufacturers have several ways to approach synchronization:

11.1. NDI Advanced SDK – Genlock API

When using NDI to send video onto the network, it is very common that one uses the computer clock to know what speed to send frames at; at this point, the SDK will use the system clock to pace the sending of frames for you.

It is common in video systems that users wish to make sure that all your NDI sources are synchronized together so that they all send video at the same rate and time. While it is tempting to solve this by having a very high precision "reference clock" (e.g., PTP), this often works very well on local networks but does not easily extend to systems that are remote from each other (e.g., your local network and in the WAN).

The NDI SDK now allows you to easily clock any number of video sources on the network to match a centralized clock and even interface those with external video clocks like local SDI sources commonly used as a genlock signal.

The NDI genlock allows one to create a "genlock clock" attached to any NDI sender on the network. That genlock clock can then be used to correctly time all senders on the network so that they are correctly timed with the NDI sender. By sharing this NDI source into the cloud, you can ensure that you have full genlock support that spans both on-premises, remote networks, and in-cloud connections. By driving an NDI source using an SDI (or PTP, 2110, HDMI) converter, it is even simple to genlock your entire NDI network to a physical genlock signal.

For an NDI source to correctly operate as an NDI Genlock, it is essential to bear in mind a couple of key ingredients as outlined below:

- It is strongly recommended that the NDI source is a stream from NDI version 5, which has been significantly improved to support genlock capabilities. It is possible that some NDI streams from previous versions are not fully compliant with how NDI genlock operates.
- Considering that the default NDI is unicast, it is essential that the source has enough network bandwidth to deliver a signal to all the NDI instances that need to be genlocked. Configuring this source for multicast might also help, although multicast often is complex to full support.
- NDI genlock is very robust and supports correct cross-frame-rate locking. For instance, a sender might be 30Hz, and you are genlocking a 60Hz signal to it. This is, however, not a recommended workflow where it can be avoided.
- Some NDI sources like Test Pattern Generator and NDI Screen Capture do not always send a regular stream of frames. They do this to save network bandwidth and CPU time. Sources such as these cannot be used as a basis for genlock.
- If the genlock clock cannot correctly genlock to an NDI sender, it will fall back to using the system clock and so can continue to work reasonably.
- Since there is some (low) overhead associated with each genlock instance, it is recommended only to have one for each source that wish to lock, too.

11.2. Timestamp, NDI Advanced SDK – AV Sync API

NDI transmitters and receivers utilize the system clock as a point of reference. This system clock can be aligned with robust sources like NTP or PTP. The timing details from these sources are integrated into the NDI Stream through timestamps embedded in each video and audio frame (with audio frames being smaller than video frames).

Subsequently, the receiver can realign audio and video sources using this timestamp information. This method provides an ingenious approach to achieving synchronization, making it an ideal solution for remote or cloud-based production scenarios.

12.NDI in Cloud

Setting up an NDI-based video production in a Virtual Private Cloud is quite easy; the first step is to define how to make NDI Discovery and Registration work in a VPC. Cloud providers allow the creation of a multicast domain; multicast is required to use mDNS-based discovery and registration. This setup requires the creation of a transit gateway with multicast enabled. Enabling multicast in the cloud might require specific knowledge; for this reason, the easiest solution to enable NDI Discovery and registration is to set up a Discovery Service. NDI Discovery Service requires just a basic Windows or Linux-based instance to run.

13.Glossary

Cache

Cache refers to a reserved section of computer memory or an independent high-speed storage device used to accelerate access and retrieval of commonly used data.

Domain

A domain refers to a LAN subnetwork of users, systems, devices, and servers. Domain can also refer to the IP address of a website on the Internet.

DNS

DNS (Doman Name System) is a system used by the Internet and private networks to translate domain names into IP addresses.

mDNS

mDNS (multicast DNS) refers to the use of IP multicast with DNS to translate domain names into IP addresses and provide service discovery in a network that does not have access to a DNS server.

Ethernet

Ethernet, standardized as IEEE 802.3, refers to a series of LAN (Local Area Network) technologies used to connect computers and other devices to a home or business network.
Ethernet is a physical and data link layer networking protocol that supports data transfer rates starting at 10 Mbps, typically over twisted pair cabling, but also fiber optic and coaxial cabling.

IGMP

IGMP (Internet Group Management Protocol) is the protocol used in IP multicasting that allows a host to report its multicast group membership to networked routers in order to receive data, messages, or content addressed to the designated multicast group.

IP

IP (Internet Protocol) is the communications protocol for the Internet, many wide area networks (WANs) and most local area networks (LANs) that define the rules, formats, and address scheme for exchanging datagrams or packets between a source computer or device and a destination computer or device.

IPv4

IPv4 (Internet Protocol Version 4) is the fourth and most used version of the Internet Protocol. IPv4 uses a 32-bit IP address scheme for network identification and communication, with each unique IP address expressed as four numbers (between 0 and 255) separated by decimal points.

IPv6

IPv6 (Internet Protocol version 6) is the latest version of the Internet Protocol, developed to eventually replace IPv4 (Internet Protocol version 4). IPv6 uses a 128-bit IP address scheme for network identification and communication, with each unique IP address expressed as eight groups of four hexadecimal digits (numbers from 0-9 or letters from A-F) separated by colons. In addition to exponentially increasing the number of available IP addresses, IPv6 simplifies and streamlines network communication while increasing security, compatibility, and efficiency.

LAN

LAN (Local Area Network) is a network that connects computers and devices in a room, building or group of buildings. LANs are typically deployed in homes, offices, and schools, where users share access to the same server, resources, and data storage. A system of LANs can also be connected to form a WAN (Wide Area Network).

Layer 2

Layer 2 refers to the OSI networking model's second layer or Data Link layer. A layer 2 switch uses hardware-based switching to transmit data between connected devices based on their MAC (Media Access Control) layer addresses.

Layer 3

Layer 3 refers to the OSI networking model's third layer or Network layer. A layer 3 switch uses hardware-based switching to transmit data between connected devices based on their IP (Internet Protocol) addresses. A layer 3 switch can support packet inspection and routing protocols to prioritize and forward traffic.

MAC Address

MAC (Media Access Control) address refers to a unique physical address identifying a network node.

Mbps

Mbps (Megabits per second) is a unit of measurement for data transfer speed, with one megabit equal to one million bits. Network transmissions are commonly measured in Mbps.

NDI

NDI (Network Device Interface) is an open protocol for IP transmission and live production using standard LAN networking. NDI allows networked video systems to identify and communicate with each other over IP, and encode, transmit and receive multiple streams of broadcast-quality, low-latency, frame-accurate video, and audio in real time.

OSI

The OSI (Open System Interconnection) reference model is a standard that defines worldwide network communication, developed by ISO (International Organization for Standardization). The OSI reference model divides network communication into seven layers: 1) Physical, 2) Data Link, 3) Network, 4) Transport, 5) Session, 6) Presentation, and 7) Application.

Packet (Frame)

A packet, also known as a frame or datagram, is a unit of data transmitted over a packet-switched network, such as a LAN, WAN, or the Internet.

Port

A port is a communications channel for data transmission to and from a computer on a network. Each port is identified by a 16-bit number between 0 and 65535, with each process, application or service using a specific port, or multiple ports, for data transmission. Port can also refer to a hardware socket used to physically connect a device or device cable to your computer or network.

QoS

QoS (Quality of Service) is the measure of performance for systems or networks, with considerations that include availability, bandwidth, latency, and reliability. QoS can also refer to prioritizing network traffic to ensure a minimum or required level of service, predictability, and/or control.

Subnet

Subnet (short for subnetwork) refers to a distinct subdivision of an IP network, usually created for performance or security purposes. Subnets typically include the computers, systems, and devices in one location, office, or building, with all nodes sharing the same IP address prefix.

TCP

TCP (Transmission Control Protocol) is a network communications protocol that enables two host systems to establish a connection and exchange data packets, ensuring that data is delivered to the correct destination. TCP is typically grouped with IP (Internet Protocol) and is known collectively as TCP/IP.

UDP

UDP (User Datagram Protocol) is an alternative protocol to TCP that is used when reliable delivery of data packets is not required. UDP is typically used for applications where timeliness is of higher priority than accuracy, such as streaming media, teleconferencing, and voice-over IP (VoIP).

WAN

WAN (Wide Area Network) is a network that spans a relatively broad geographical area, such as a state, region, or nation. WANs typically connect multiple smaller networks, such as LANs (Local Area Network) and MANs (Metropolitan Area Network). The Internet is an example of a WAN.

Disclaimer

The information in this document is believed to be correct as of the date of publication. However, our policy is one of continual development so the information in this guide is subject to change without notice and does not represent a commitment on the part of Vizrt.

Technical Support

For technical support and the latest news of upgrades, documentation, and related products, visit the NDI website at www.ndi.video