NDI NETWORKING

Introduction

NDI® is an efficient, easy-to-deploy IP media transport standard supporting high-quality audio and video over low-cost networks. However, it is important to understand how NDI transport works and the requirements of the network for a successful deployment. In this paper, we will walk through the different flavors of NDI, the underlying network transports it utilizes and what you need to know about building and configuring a network for NDI.

History of NDI

NDI was introduced in 2016, and has gone through many revisions throughout the years, with highlights outlined below:

Version	Year	Features
1	2016	Initial High-Bandwidth (SpeedHQ codec)
2	2016	Cross-subnet support
3	2017-18	NDI-HX (H.264), multicast support, PTZ camera support, embedded FPGA SDK
4	2019-20	Multi-TCP, NDI-HX2 (H.264/H.265), UHD, Discovery Server
5	2021-22	rUDP, NDI-HX3, NDI Bridge, NDI Router

NDI Versions are backwards compatible, so, for example, an NDI 5 device should interact with an NDI 3 device, based on the NDI 3 capabilities.

NDI Profiles

NDI implements several different transport & codec profiles. These are tailored to different applications.

NDI High-Bandwidth (NDI-HB)

This is the original NDI codec (sometimes referred to as NDI High Bandwidth or NDI-HB) and the most commonly used. It utilizes the SpeedHQ codec (a long-GOP MPEG-2 variant) that emphasizes image quality and low latency at relatively high bitrate. It supports full-resolution and low-bandwidth preview streams.

NDI HX

NDI HX (High Efficiency) utilizes H.264 long-GOP video and AAC audio codecs to achieve lower bitrates. However, it is subjectively lower quality and higher latency than NDI High-Bandwidth.

NDI HX2

NDI HX2 is an extension of NDI HX which adds support for HEVC video codec and Opus audio codec. HDI HX2 also supports UHD resolutions.

NDI HX3

NDI HX3 utilizes the same codecs as HX2 (H.264, HEVC, AAC, OPUS), but utilizes a short-GOP structure (1 or 2-frame GOP) at higher bitrate to optimize image quality and latency. It does this at about half the bitrate of NDI High-Bandwidth. In the future, HX3 is expected to be the preferred profile for high-quality, low-latency applications, replacing NDI High-Bandwidth.

Profile	Bitrate	Latency	Codec	Quality
NDI-HB	100-160Mb/s	16ms	SpeedHQ (HD)	Great
NDI-HX/HX2	9-16Mb/s 20-30Mb/s 6-10Mb/s 15-20Mb/s	80-200ms	H.264 (HD) H.264(UHD) HEVC (HD) HEVC (UHD)	Good
HDI-HX3	25-60Mb/s 90-110Mb/s 20-50Mb/s 70-85Mb/s	<100ms	H.264 (HD) H.264 (UHD) HEVC (HD) HEVC (UHD)	Great

NDI Transport

NDI utilizes a number of transport protocols, which fall into one of 2 IP protocol types: unicast or multicast.

Unicast

Unicast transports (including TCP, multipath-TCP, UDP and rUDP) are point-to-point. Each stream has exactly 1 sender and one receiver. As shown in Figure 1, if multiple receivers wish to capture a stream, the NDI Source must transmit it multiple times. This can have scaling issues if a large number of receivers are interested in the same stream.

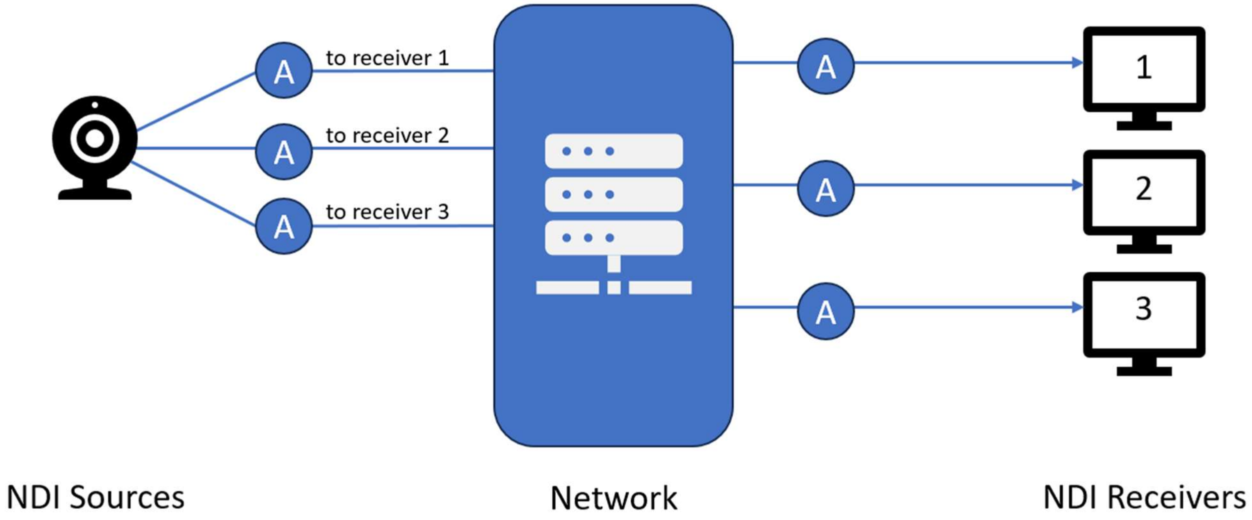

Figure 1 - Unicast Transport

Multicast

Multicast transport allows a stream to have one sender and multiple receivers, requiring a source to only send a stream once, regardless of the number of receivers. This reduces the processing of the source and its network utilization. As shown in Figure 2, only receivers that subscribe to the stream will receive it.

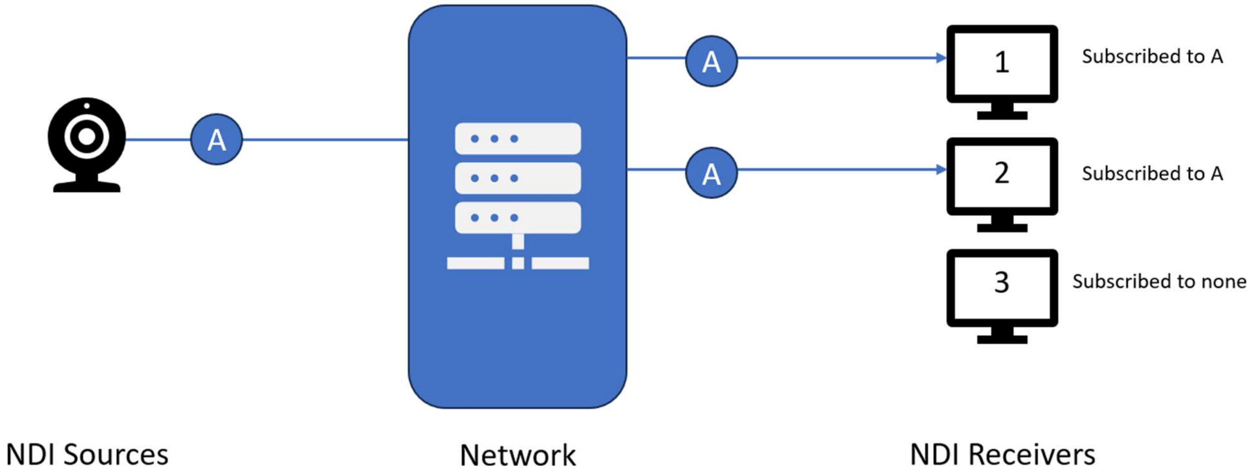

Figure 2 - Multicast Transport

As we add additional sources to the network, individual receivers only receive the streams they are interested in, providing efficient use of network bandwidth, as illustrated in Figure 3.

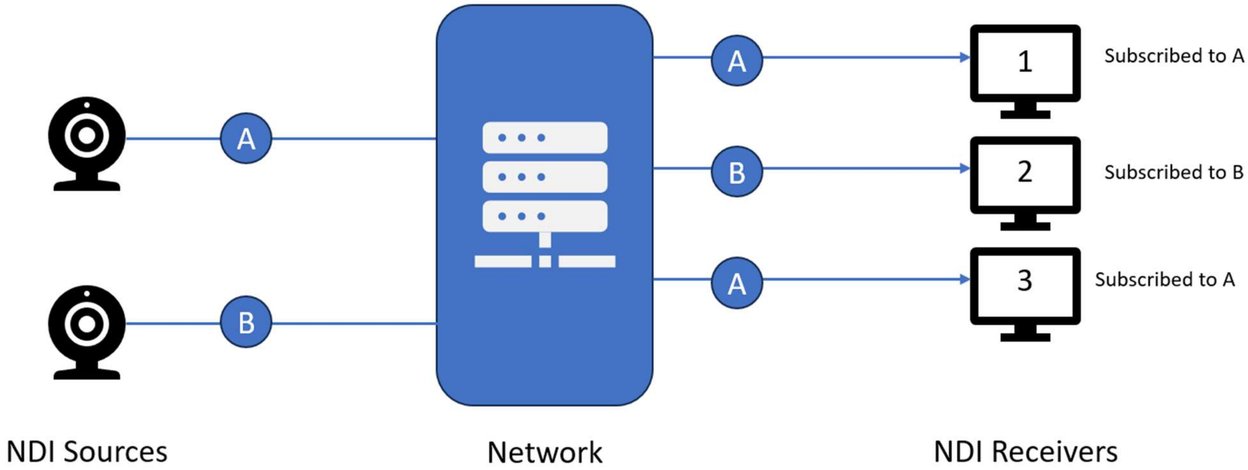

Figure 3 - Multicast with multiple devices

In order to ensure that a receiver only receives the streams it is interested in, multicast devices utilize Internet Group Management Protocol (IGMP) to tell the network which multicast streams it wants to receive. This requires the network to have an IGMP querier and implement IGMP snooping. Networking equipment that does not support this (or does not enable it) will typically fall back to a “flood” mode, where all multicast streams are sent to all devices, regardless of whether they want it or not, as shown in Figure 4.

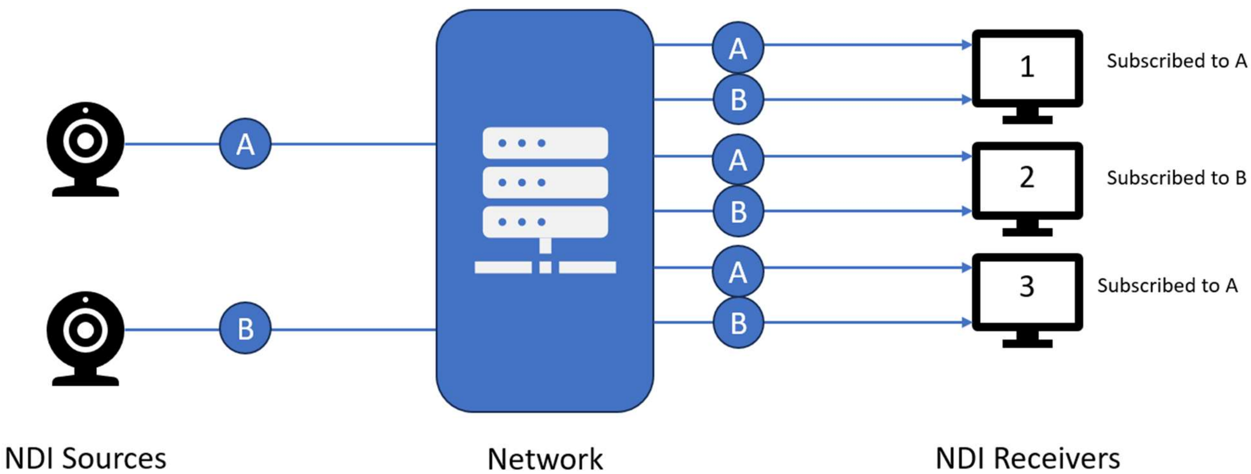

Figure 4 - Multicast port flooding

Multicast flooding can very quickly saturate all the bandwidth on a network, bringing it to its knees. It is therefore absolutely necessary that proper multicast support with IGMP snooping is supported by the network equipment if multicast NDI streams are being used.

NDI Protocols

TCP / Multipath TCP

TCP and multipath-TCP are *unicast* transports that guarantee delivery. If a packet is lost, the source is asked to retransmit it. This offers high reliability at the expense of large (and unpredictable) latency. Multipath TCP allows transport across multiple NICs and network paths.

UDP / UDP+FEC

UDP is a *unicast* transport that can be thought of as "fire and forget". The source transmits packets, but there is no guarantee of delivery. If the receiver does not receive a packet, it is lost forever and will not be retransmitted. This offers much lower (and predictable) network latency as there are no packet acknowledgments or retransmissions. This is generally preferred for video streaming as a momentary small glitch is preferable to unpredictable delays in the transmission path.

UDP may also include Forward Error Correction (UDP+FEC). This includes extra information in the packet that allows a receiver to reconstruct a certain amount of information that may have been lost in transmission. This has a penalty of higher latency and higher bandwidth than standard UDP, but is more predictable and generally lower latency than TCP.

rUDP

Reliable UDP (rUDP) is an optimized *unicast* UDP transport that provides dynamic bandwidth and buffering techniques along with sequencing and flow control to optimize delivery over congested and high-latency networks. It is the default transport recommended for most networks.

Multicast

Multicast transport utilizes UDP network transport, but rather than being point-to-point as in unicast UDP, it allows one-to-many transport, with the lower latency of UDP. Multicast NDI offers the most efficient use of network bandwidth if the network equipment supports it.

What Settings Should I Use?

Although each installation is unique with its own requirements, here are some good starting points:

Each NDI profile offers different advantages:

- NDI High-Bandwidth is the most universally supported and offers low latency with high quality.

- NDI HX and HX2 offer significantly lower network bandwidth utilization at the expense of latency and quality.
- NDI HX3 provides quality and latency similar to NDI High-Bandwidth at approximately half the bandwidth utilization. However, it is not yet supported broadly. In the future, this will likely be favored over NDI High-Bandwidth for low-latency/high-quality applications.

You are also free to mix profiles in a facility (for example, to trade-off latency on certain critical signals vs. lower bandwidth on others). However, it is important to check which profiles are supported by all devices on the network. If you decide, for example, to implement HX3 in your design, you may come across some devices that cannot receive it.

The NDI protocol selection for most users will be one of the following:

- rUDP is efficient for simple networks with few (10's of) devices. It could hit scaling issues particularly in situations where a single source must be sent to many receivers (such a black or bars). This is the default protocol for NDI.
- Multicast provides the most efficient utilization of network bandwidth on both sources and receivers. However, it requires specific multicast handling by the network hardware. Multicast can be challenging to configure and troubleshoot, and is usually not supported in cloud environments.

Discovery

For NDI receivers to connect to network streams, the receiver needs to discover what streams and devices are available. NDI implements 2 different mechanisms to facilitate discovery and registration of NDI streams:

- mDNS
- NDI Discovery Service

mDNS

Multicast Domain Name System (mDNS) is a mechanism that can create a zero-configure discovery environment. A device can discover the available services on the network by sending an mDNS multicast message (multicast address 224.0.0.251, port 5353). Discoverable NDI devices then reply to the sender using a unicast response, disclosing their available resources.

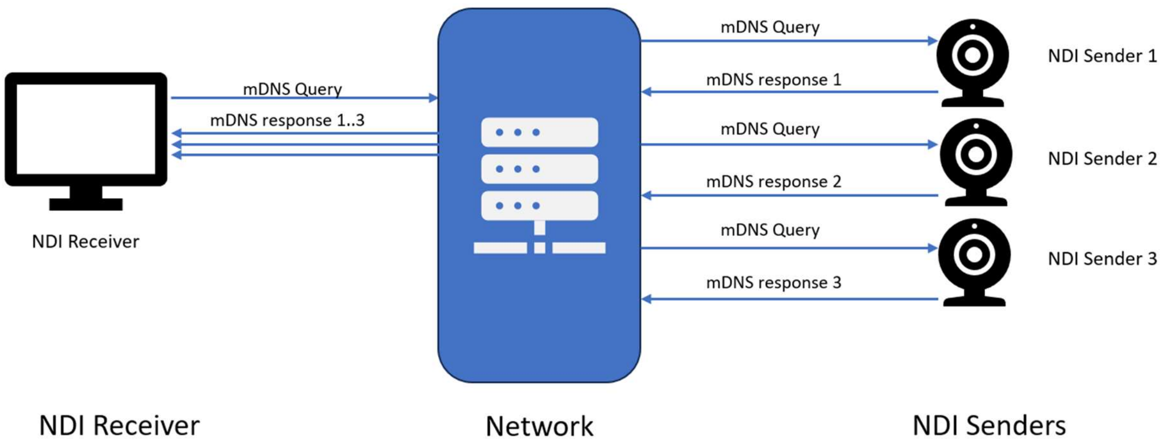

Figure 5 - mDNS Discovery

In simple network environments, mDNS will work without any special configuration. However, it is important to verify the following:

- Multicast (including mDNS) will not traverse across subnets, unless the network is specifically configured to enable multicast routing.
- mDNS is not usually supported in cloud deployments.
- Firewalls must be configured to pass mDNS messages.
- In Microsoft Windows, the Network Location should be set to Private.
- mDNS usually does not require multicast snooping to be enabled on a network switch. Simple networks will simply forward the mDNS traffic to all ports (these are small messages). However, if a multicast querier is enabled, ensure it is configured to allow mDNS (224.0.0.251) messages to be forwarded.

For small NDI installations, mDNS discovery works fine, however, for larger systems, or where there are multiple subnets, NDI Discovery Server should be used.

NDI Discovery Service

One of the disadvantages of mDNS for discovery is that it is a many-to-many discovery mechanism. Each receiver device reaches out the network, and every sender must respond to each receiver. This can cause unwanted overhead for systems with a large number of devices. mDNS also is reliant on multicast, which might be undesirable or unsupported in some network environments (such as some cloud environments).

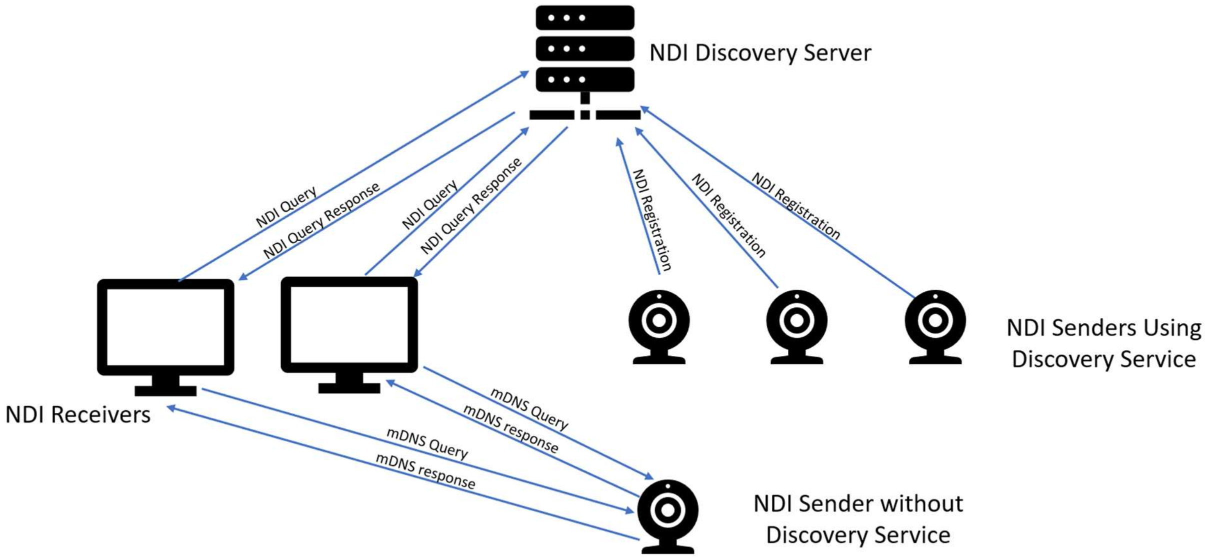

Figure 6 - NDI Discovery Service

The NDI Discovery Service provides a centralized registry for NDI resources on a network. Senders actively register themselves with the discovery server and receivers query the discovery server for available streams, all using unicast messaging. This can be significantly more efficient than the mDNS approach, reducing the overhead of many-to-many discovery.

When a sender device is configured to use NDI Discovery Service, it registers itself with the discovery server, and will then no longer respond to mDNS query requests. A receiver will only learn about these senders via the discovery server. The receiver can still learn of senders which are not configured for Discovery Service through mDNS.

The main downside of NDI Discovery Service is that it requires an additional component to be added to the network. This might be a physical server, VM, or cloud instance.

Which Discovery Mechanism Should I Use?

NDI Discovery Service should be considered if any of the following are true:

- There are a large number of NDI devices (>20 or so) on the network
- NDI devices span across multiple subnets
- Multicast is not supported or undesirable on the network

If none of the above scenarios apply, then mDNS peer-to-peer discovery can be used, and it would therefore not be necessary to stand up a NDI discovery server.