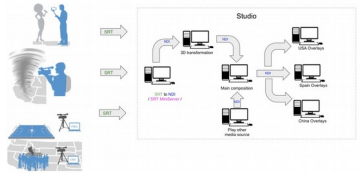
Consept of NDI on SRT

https://srtminiserver.com/blog_ndi_srt/#:~:text=NDI is glue for build production workflow inside,task%3A low-latency delivery of data (not only video).



Movie

NDI

NDI

NDI

header
NDI
SRT

header
NDI
SRT

listener(or caller)

pack
NDI → SRT

Wide Area
Network

caller(or listener)

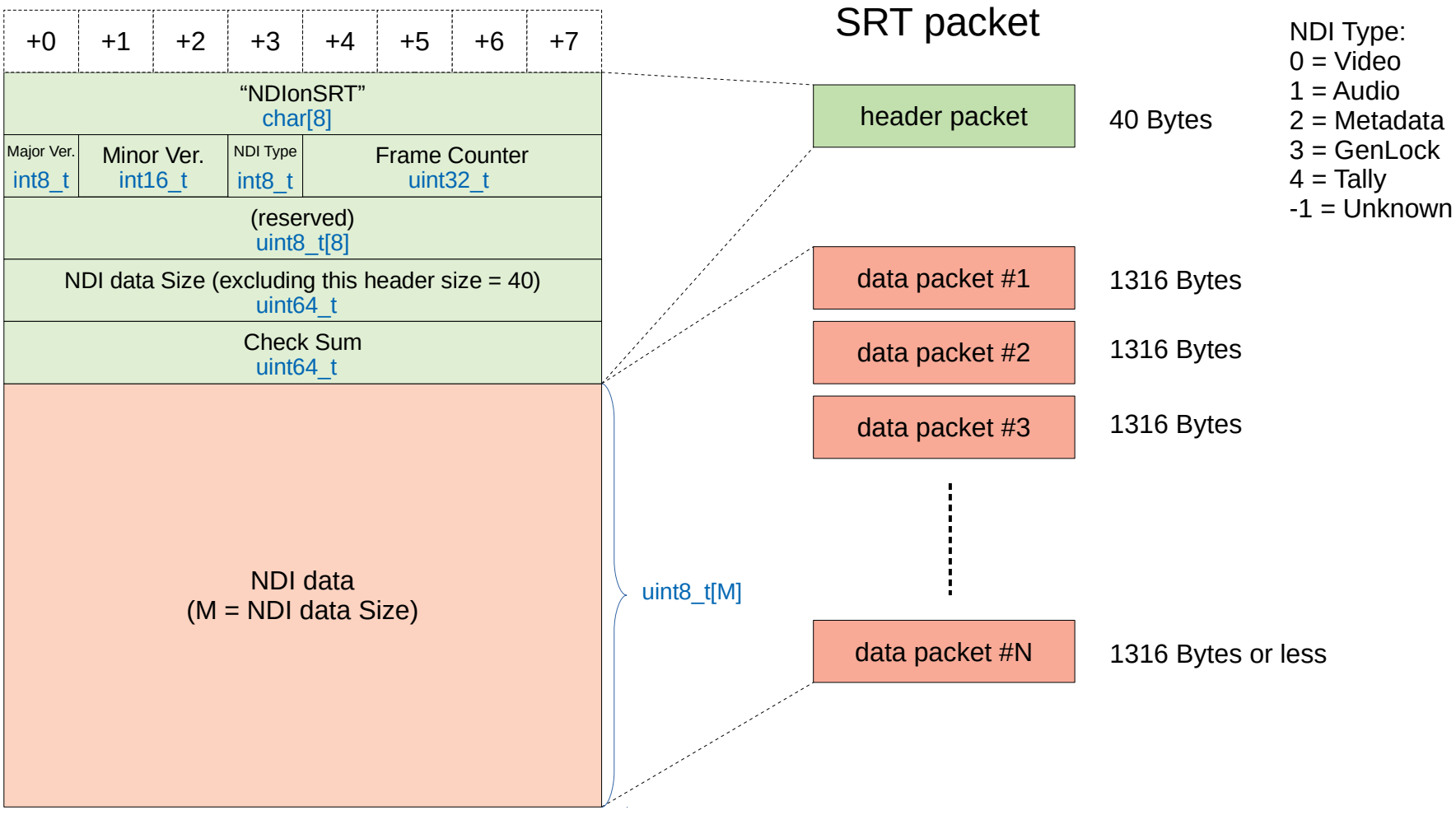
unpack
SRT → NDI

NDI

NDI

NDI

Structure of NDI on SRT



Structure of NDI data : Video

```
typedef struct NDIlib_video_frame_v2_t
{
	// The resolution of this frame
	int xres, yres;

	// What FourCC describing the type of data for this frame
	NDIlib_FourCC_video_type_e FourCC;

	// What is the frame-rate of this frame.
	// For instance NTSC is 30000,1001 = 30000/1001 = 29.97fps
	int frame_rate_N, frame_rate_D;

	// What is the picture aspect ratio of this frame.
	// For instance 16.0/9.0 = 1.778 is 16:9 video
	// 0 means square pixels
	float picture_aspect_ratio;

	// Is this a fielded frame, or is it progressive
	NDIlib_frame_format_type_e frame_format_type;

	// The timecode of this frame in 100ns intervals
	int64_t timecode;

	// The video data itself
	uint8_t* p_data;

	union
	{
		// If the FourCC is not a compressed type, then this will be the inter-line stride of the video data
		// in bytes. If the stride is 0, then it will default to sizeof(one pixel)*xres.
		int line_stride_in_bytes;

		// If the FourCC is a compressed type, then this will be the size of the p_data buffer in bytes.
		int data_size_in_bytes;
	};

	// Per frame metadata for this frame. This is a NULL terminated UTF8 string that should be in XML format.
	// If you do not want any metadata then you may specify NULL here.
	const char* p_metadata; // Present in >= v2.5

	// This is only valid when receiving a frame and is specified as a 100ns time that was the exact moment
	// that the frame was submitted by the sending side and is generated by the SDK. If this value is
	// NDIlib_recv_timestamp_undefined then this value is not available and is NDIlib_recv_timestamp_undefined.
	int64_t timestamp; // Present in >= v2.5
} NDIlib_video_frame_v2_t;
```