

What's In It For the End User? Taking RIST Out of the Lab And Projecting It Into the Real World

libRIST > Step by Step > Script by Script

Sergio Ammirata, Ph.D.

SipRadius

September 22, 2020



lib>>>RIST

Today's Presentation

- **4K 30+ Mbps video over libRIST with network emulated loss, corruption, reordering... yet *perfect*.**
- **Today's end user is content with 720P, 1080P over HLS/DASH – “good enough.”**
- **Tomorrow that same end user expects 4K, and perhaps even 8K in the future.**



What Developers Have to Do

- **Efficient GUIs to help content providers acquire and transport 4K to end users.**
- **Combination of open standards *and* FOSS engines best ensure interoperability**



Underlying the Demo

- **LibRIST, which is FOSS.**
- **Coral OS, SipRadius' proprietary linux flavor, calling libRIST.**
- **We use the exact same applications (ristsender, ristreceiver) in Coral OS as are in the libRIST repo. The idea is to promote “production quality” tools, not just the code.**



The Demo

- Note that this is a 4K desktop capture (*thanks, nVidia!*), though you may see a different resolution via Zoom.
- It's all one take, and I will narrate above it...



Insert video and play full screen here



What We Learned

- Whatever libRIST supports in command line can be made easier in a GUI.
- By torture-testing an overweight h.264 4K video, we prove that h.265, 4K and RIST will be a good combination.
- Bottom line: if we make it easier for the content providers, when the RIST-enabled players reach the public, RIST will work.
- Email questions to: don.cardone@denz.tv

