DIFI: The Universal Language for Streaming Digitized RF
Digital IF Interoperability (DIFI)

• Goal: Wide adoption of an interoperable Digital IF standard
  • Match the interoperability that is native to analog IFs
  • Create an open, simple, interoperable digital IF standard
  • Encourage adoption of the standard throughout the satellite industry and beyond

• Purpose:
  • Define an interoperable standard based on VITA-49
  • Design standard for easy adoption
  • Publish as an open, referenceable standard
  • Provide a way to certify compliance
  • Reduce vendor lock-in that plagues the satellite industry

• Consortium Structure:
  • Leverage IEEE-ISTO to manage the Consortium and specification
Standards Without Interoperability

Framework standards allow everyone to claim compliance but without interoperability. Result is vendor lock-in. Drives up development risk. Prevents economies of scale.
Why Subset of VITA Specification

• Two most common standards: VITA-49 and eCPRI
  • Both are framework standards allowing unique implementations (i.e. neither ensure interoperability)
  • VITA-49.2 is an established ANSI standard that is simple and well suited for satcom

• VITA-49 is the only widely deployed Digital IF standard in satellite market today
  • 100+ Digital IF systems in operations today
  • Used across multiple different customers and applications
    • Choice of US military, Cloud, aaS for satellite applications
  • Specification tailored for satellite industry requirements
### Some Protocols for Carrying IQ

<table>
<thead>
<tr>
<th>Protocols</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ZMQ</strong></td>
<td>1. API for development 2. Remote streaming 3. Flow control capability at lower levels. High level is fire and forget. 4. Efficient</td>
<td>1. Not widely adopted 2. No context or M&amp;C</td>
</tr>
<tr>
<td><strong>UHD/CHDR</strong></td>
<td><strong>Pros:</strong> 1. API for development 2. Remote streaming 3. Flow control capability with a flag.</td>
<td>1. CHDR isn’t widely adopted but there is a DIFI RFNoC block <a href="https://github.com/DIFI-Consortium/rfnoc-difi">github.com/DIFI-Consortium/rfnoc-difi</a></td>
</tr>
</tbody>
</table>
DIFI Packet Protocol

- Signal Data Packets
  - Carries the IQ data of the stream.

- Signal Context Packets
  - Carries the RF parameters of the stream.

- Version Context packets
  - Carries version of standard and timing information.

- Context packets are informational.
DIFI Packet Protocol

Ethernet Frame Header (inc. 802.1Q / 802.1ad)

Ethernet Frame Payload

FCS (Checksum)

Ethernet Packet Size (128 octets ~ 9000 octets (jumbo frame size = vendor dependant))

IP Header

IP Payload

UDP Header

UDP Payload

Signal Data Packet Size - N+7 words

Overhead within Ethernet frame
56 octets IPv4, min 76 octets IPv6

Maximum available Signal Data Payload size
Jumbo frame size minus Overhead
20 octets

VITA Header

Signal Data Payload

VITA Header

Signal Context Packet

VITA Header

Version Context Packet

20 octets IPv4, 40 octets IPv6

8 octets

7 words (28 octets)

N octets

7 words

20 words

4 words
Example Use-Case

Other Potential Configurations

- Channelizer
  - DIFI
  - N DIFI Streams

- DSP
  - Anything
  - DIFI

- DIFI Producer
  - RF

- DIFI Consumer
  - RF

- Digitizer
  - DIFI
  - Standard Context Packets
  - Version Context Packets

- Modem
  - DIFI Consumer
  - DIFI Producer

- Network

- DIFI
  - Anything
  - DIFI
Existing DIFI Tooling

- **DIFI Python Validator**
  - The Packet validator is a collection of Python scripts that generate ‘Standard Context' packets, 'Version Context' packets and 'Data' packets (dcs.py, dvs.py & dds.py) for consumption by a DIFI packet receiver (drx.py). The collection of scripts were released as open source by Kratos Defense to the DIFI Consortium.
  
  https://github.com/DIFI-Consortium/DIFI-Certification

- **DIFI GNU Radio Out of Tree blocks (gr-difi)**
  - The GNU Radio out of tree blocks consist of a "Data" packet Source and Sink blocks and can be found in gr-difi. The collection of GNU Radio blocks were part of gr-azure and released as open source by Microsoft to the DIFI Consortium.
  
  https://github.com/DIFI-Consortium/gr-difi

- **DIFI Wireshark Dissector**
  - The DIFI Wireshark Dissector is a LUA plugin created for Wireshark that allows for packet inspection of DIFI Standard Context, Standard Data and Version Context packets. The plug-in was created by Johanna Rivera.
  
  https://github.com/DIFI-Consortium/DIFI-Certification/tree/main/wireshark-dissector
gr-difi

- gr-difi allows users to communicate with DIFI devices by streaming and receiving DIFI signal data packets within a GNU Radio flow graph.

Flow graph example from gr-difi running in one VM on Azure
DIFI packets running from one VM to another VM in Azure
DIFI Plus SigMF

- Signal Metadata Format (SigMF) is used to share recordings of IQ binary data with a standard meta data format
- DIFI for real-time streaming, SigMF for offline recording/analysis
- SigMF combined with DIFI is a natural pair, and GNU Radio can be used to incorporate both into your application
Trends in RF & The Cloud
Members

Corporate Members

- AMERGINT
- AvL Technologies
- AWS
- AllSpace
- Bascom Hunter
- BlueHalo
Get involved!

- Help define DIFI features for your RF needs
- Help create tooling for interacting with DIFI streams and/or provide feedback on existing tooling
- Incorporate DIFI into a project and provide feedback on the standard
- Help shape the future of DIFI at the edge and in the cloud