

# Developing an IADS Plugin

FOR CHAPTER 10 DATA DISPLAY AND ANALYSIS

Stephen Frees  
Avionics Test & Analysis Corp.



# Developing an IADS Plugin

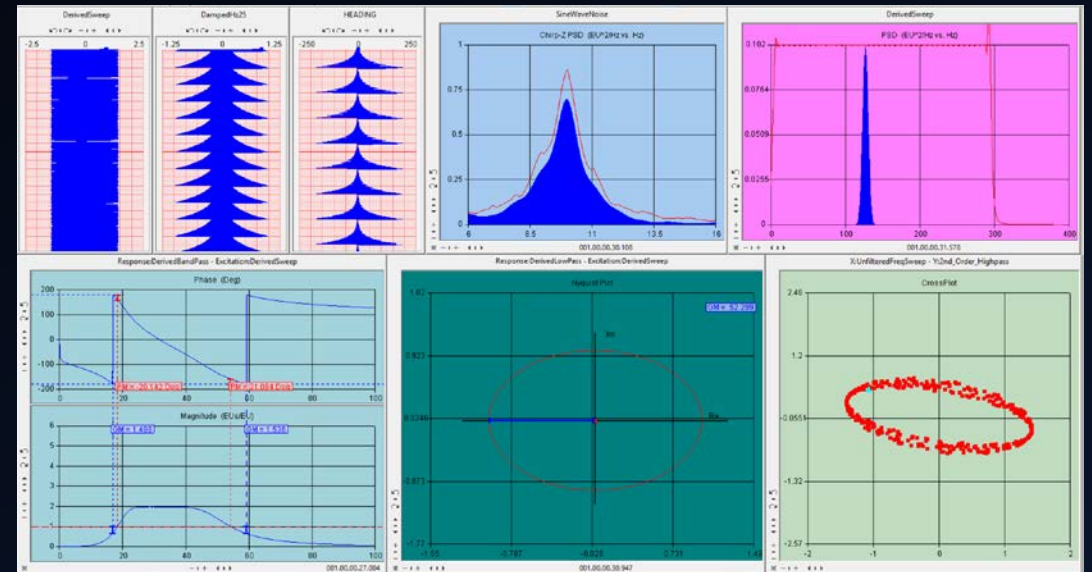
- Introduction to IADS
- What is Missing?
- Implementing a Solution
- The Finished Product
- Conclusion

# Introduction to IADS

- What is IADS?
  - Developed by SYMVIONICS, Inc.
  - Real time and post test display and analysis product
  - Scalable and portable
  - Supports a wide range of Telemetry and COTS data sources
  - Drag-and-drop display builder
  - Robust analysis tools


- IADS in the Industry

- Used by every major test program in the US
- Used in many countries world-wide
- Deployed on a wide scale of test environments



# What is Missing?

- IADS cannot automatically handle:
  - Non-standard data streams
  - Proprietary modifications of common standards
  - Chapter 10 Message packet reduction

- 
- A Plugin as a Supplement
    - Can utilize IADS Chapter 10 support
    - Ability to automate IADS parameter creation
    - Unpack Message data with C++ code
      - Fast
      - Flexible
      - Any data format

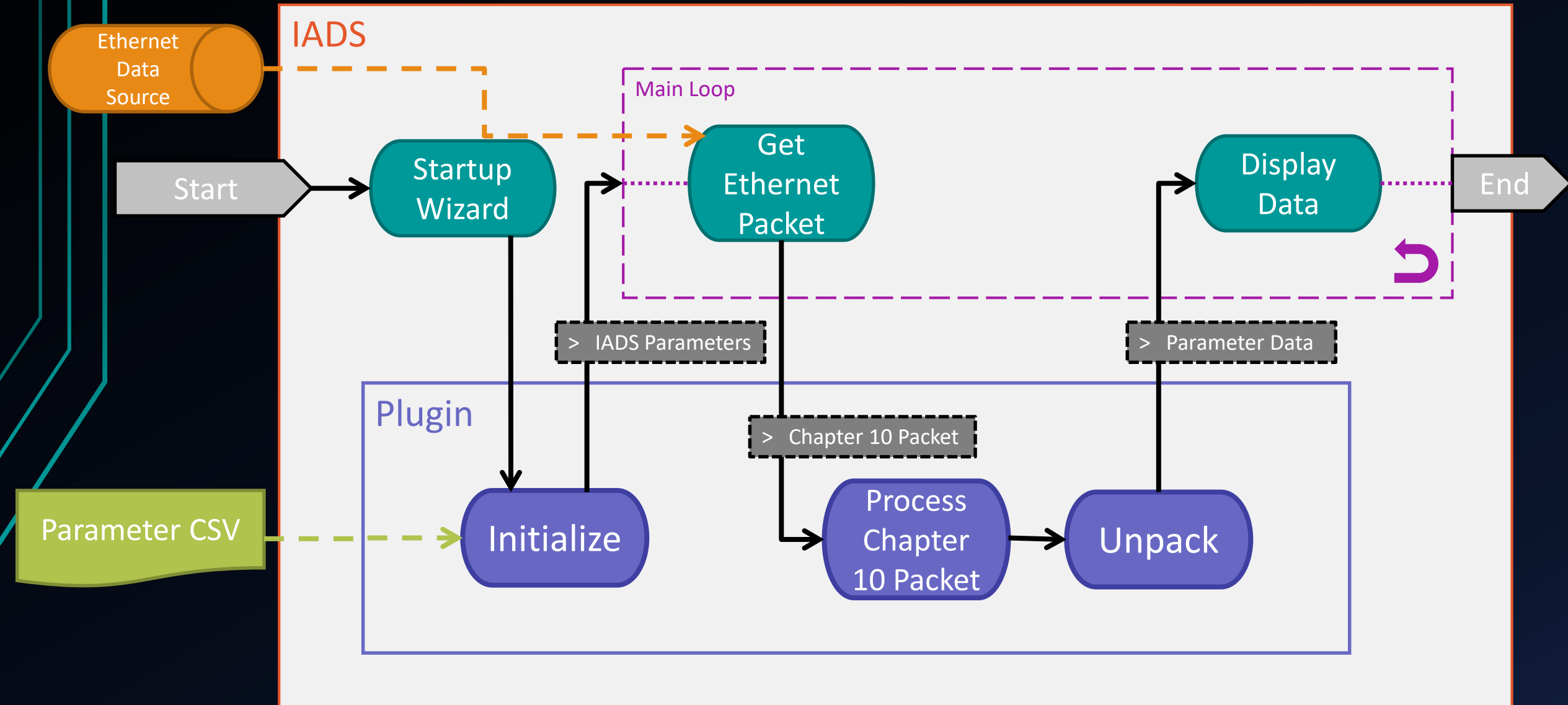
- Challenges of a Plugin
  - Operator Parameter Control
    - Modify parameter list external to the plugin
    - CSV format allows scripting
    - Enables maintenance for thousands of parameters
  - Development Environment
    - Component Object Model (COM)
    - Active Template Library (ATL)
    - Microsoft Visual Studio

# Implementing a Solution

- Design



# Basic Plugin Architecture



- IADS Interface (Chapter 10 Ethernet)
  - IPluginMeasurement
  - IladsTppCh10PluginDataStream
    - CreateBasicMeasurement(7)
      - name, shortName, longName, units, dataType, sampleRate, pMeasurement
    - addMeasurement(1)
      - measurement
    - PutTime(1)
      - iadsFormattedTime
    - PutData(2)
      - measurement, value

- Plugin Interface (Chapter 10 Message data)
  - Param
    - msgID
    - firstWord
    - startBit
    - numBits
    - unpackMode
    - measurement

- Plugin Interface (continued)

- class Unpacker

- IladsTppCh10PluginDataStream

- vector<Param>

- buildParams(1)

- IladsTppCh10PluginDataStream

- addParam(5)


- msgID, firstWord, startBit, numBits, unpackMode, measurement

- unpack(2)

- pData, dataLen

- extractParam(3)

- pData, dataLen, pParam

- 
- Implementation
    - IADS Plugin Tutorial
      - Visual Studio project creation
      - IADS Interface targeting
      - Registry script modifications
    - Supplemental Material
      - Procedural notes
      - SYMVIONICS, Inc. support personnel
    - Example code

# The Finished Product

- Example Plugin



- Portability

- Can be run on any IADS compatible machine
- Post test through data files
- Real time through Ethernet data connection

- Scalability

- User configurable parameters from CSV
- Additional Plugins can be developed in a single project
- Using standard interface minimizes development effort

# Conclusion

- Plugins can supplement IADS internal data types
- Simple interface to IADS
- Allows unpacking any data type
- Portable and scalable