Technical Steering Meeting

April 15, 2014

Antitrust Compliance Notice

- AllSeen Alliance meetings involve participation by industry competitors, and it is the intention of AllSeen Alliance to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of and not participate in any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.

- Examples of types of actions that are prohibited at AllSeen Alliance meetings and in connection with AllSeen Alliance activities are described in the AllSeen Alliance Antitrust Policy. If you have questions about these matters, please contact your company counsel, or if you are a member of AllSeen Alliance, feel free to contact Lee Gesmer or Andrew Updegrove, of the firm of Gesmer Updegrove LLP, which provides legal counsel to AllSeen Alliance.

Reminder:

This call is being recorded

Agenda

- Approve minutes from last call
- Infrastructure Repository
  - For information only
- Data-driven API Proposal
  - Presentation from Technicolor
- Wireshark protocol dissector for the AllJoyn framework
  - Brief description and pointers for download

Infrastructure Repository

Infrastructure Repository

- The Tools and Infrastructure team needs to store scripts and configuration files used to implement build automation
- These files need to be shared openly, but are not distributed as part of AllSeen official releases and do not fall under a working group.
- Name: infra/automation.git

Data-driven API Proposal

Dominique Chanet

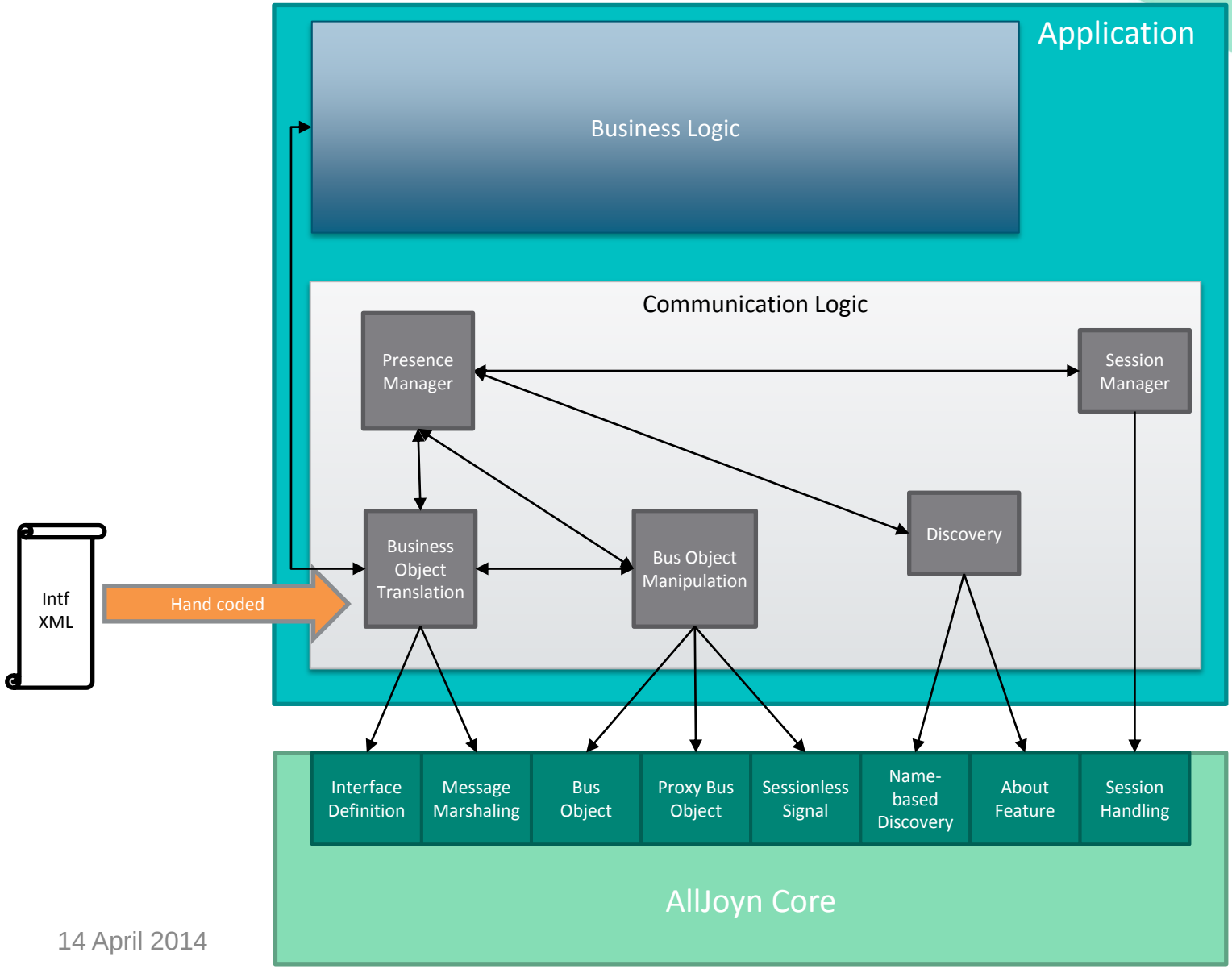
Raise conceptual level of the API

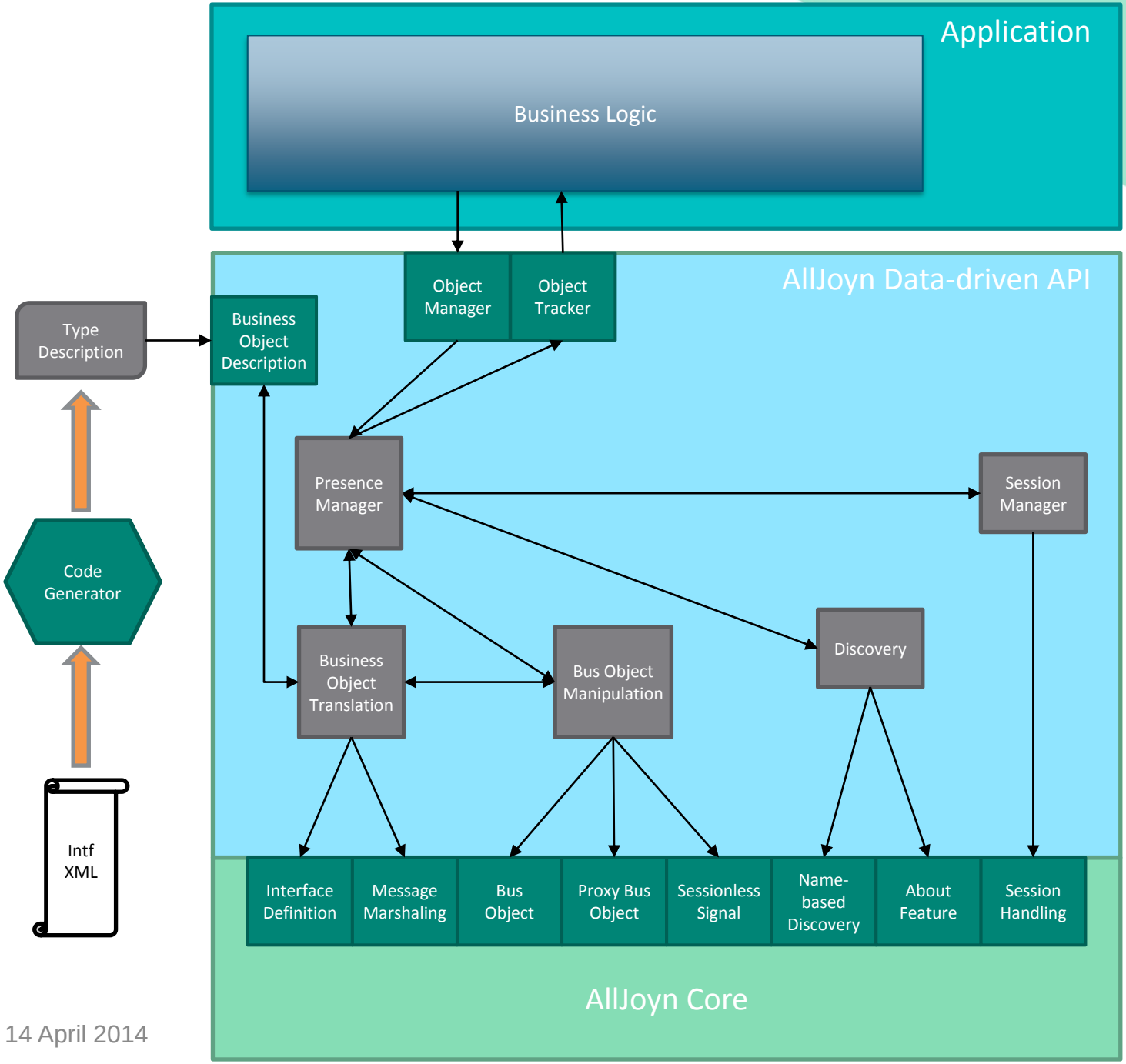
Smaller number of distinct API concepts

reduces learning curve for developers

- One approach for discovery and session setup
  better interoperability, easier for developers
- Introduce data-oriented, publish subscribe paradigm
  good fit for IoT use cases
- Extend expressiveness of type system
  make interfaces more self-describing
- Leverage code generator tooling
  less boilerplate code, friendlier C/C++ API

Classic AllJoyn Application




Application
Business Logic
AllJoyn Data-driven API
Object Manager
Object Tracker
Type Description
Business Object Description
Presence Manager
Session Manager
Code Generator
Business Object Translation
Bus Object Manipulation
Discovery
Intf XML
Interface Definition
Message Marshaling
Bus Object
Proxy Bus Object
Sessionless Signal
Name-based Discovery
About Feature
Session Handling
AllJoyn Core

Main Driver

A lot of IoT's potential value lies in devices that respond intelligently to their surroundings

- data-driven approach
  communicate with others based on the data they offer/need
- understandable data
  make sense of the data you gather
- accessible data
  know how to get the data
- ubiquitous data
  the more inputs, the better

Data-driven approach

Discover peers based on the data they offer

- current discovery is mainly based on who you are (name-based discovery)
- change this to what you offer (interface-based discovery)

Introduce the ability to define interfaces as data-oriented instead of service-oriented

- current AllJoyn interfaces focus on interactions (methods), observable properties are an afterthought
- make observable properties first-class citizens
  build publish-subscribe system to observe them

Understandable data

Data is only useful if I can make sense of it

- formalized data model description language
  based on existing Introspection XML language
- extensions to type system to make data models more self-describing
  named types, enumerations, optional fields
- best practices for data model design
  discourage use of variant types, expose state as observable properties, …
- common repository for standardized data models (out of scope for this proposal)
  - one way to describe temperature, light bulbs, GPS position, …
  - data-centric take on the AllSeen standardized services

Accessible Data

Data is only useful to me if I can access it

- AllJoyn has a toolkit-based approach to discovery and session setup
  - lots of building blocks, build the system that suits you best out of this
  - great for ad hoc distributed applications
  - reusable distributed components need a standardized, well-understood way of doing discovery and session setup
- The same concept applies to security
  - security is mandatory for IoT
  - need a standardized mechanism for authentication & access control
  - not covered in this proposal, separate efforts underway

Ubiquitous Data

Devices become more useful if they have more data available as input

- AllSeen ecosystem gets more valuable as it expands
- Need to make it easy for developers and device manufacturers to step in
- Simplified API with reduced learning curve can lower barrier to entry

Practical Approach

R14.06

- extensions to type system
  - proposal on Alliance Wiki, discussions on mailing list
  - implementation of named types and enumerations in the code generator
- marshaling improvements for C/C++
  - generic marshaling/unmarshaling code as library
  - code generator to generate type descriptions out of interface XML
  - not full application skeleton as for Thin Client
- prototype of data-driven API
  - Initially just for C++
  - as a separate library on top of AllJoyn Core
  - primary goal: gather feedback from the community

Practical Approach

R14.10

- data-driven API
  - Integrate data-driven C++ API in core
    taking advantage of e.g. Next-Gen Name Service
  - Add language bindings (Android/C/Thin Client/…)
- extensions to the type system
  - optional fields
  - extend to all language bindings
- extensive documentation
  - best practices for data model definition
  - data-driven API reference guide
  - data-driven API user guide

Wireshark Plugin for the AllJoyn Protocol

Joseph Huffman

What is Wireshark?

It's hard to overstate how useful this tool is.

Craig Dowell, AllJoyn Senior Staff Engineer
April 14, 2014

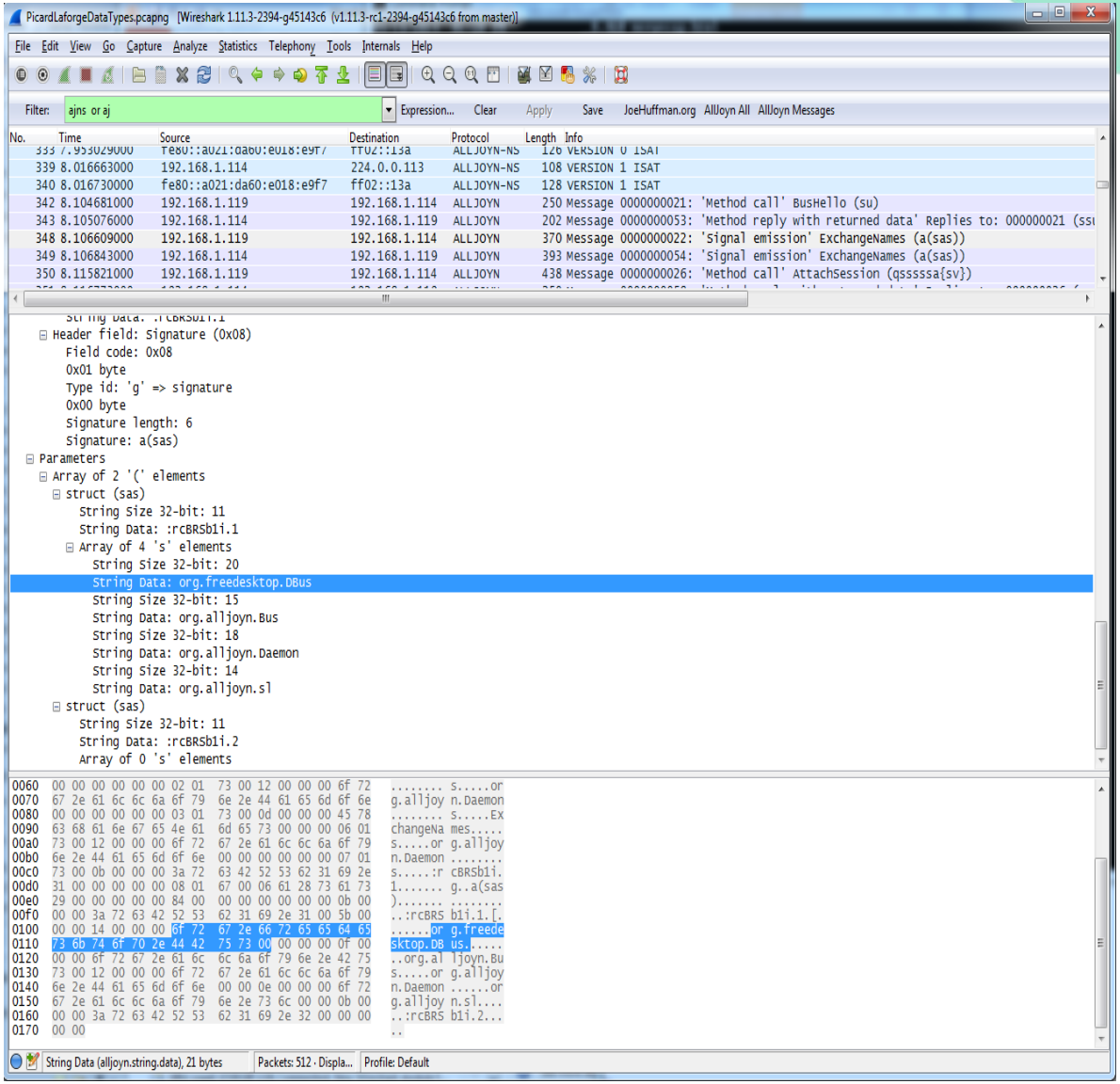
Wireshark is a described as "The World's Most Popular Network Protocol Analyzer".

Wireshark captures and displays network traffic in human readable format with powerful protocol specific filters to make debugging network communication dramatically easier. See http://www.wireshark.org/ for more information.

An AllJoyn written protocol dissector has just been accepted for inclusion in the next Wireshark stable release, version 1.12. This is scheduled for some time in June.

A developmental release is scheduled for April 15th.

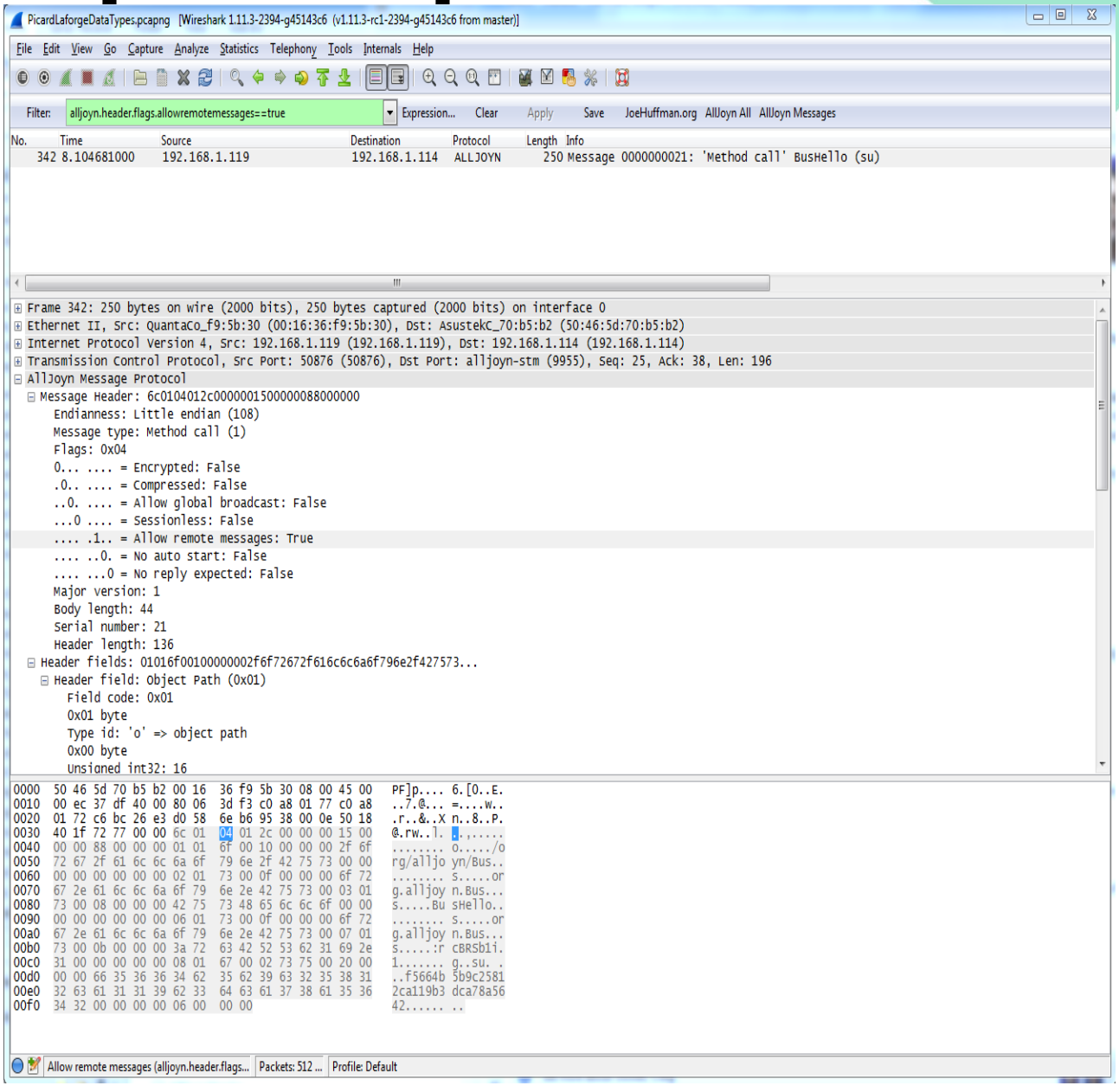
Sample output



Filters

- To display only AllJoyn traffic type "aj or ajns" in the "Filter:" toolbar and then click "apply".
- In the filter expression window type "alljoyn." to see AllJoyn data types available to filter upon. E.g. applying "alljoyn.string.data“ limits the packets shown to only packets that contain that data type.
- Apply the filter "alljoyn.string.data==org.datatypes.test" to see only packets with AllJoyn string data values of "org.datatypes.test".
- Detailed filters down to the bit level cases with a filter like: "alljoyn.header.flags.allowremotemessages==true".

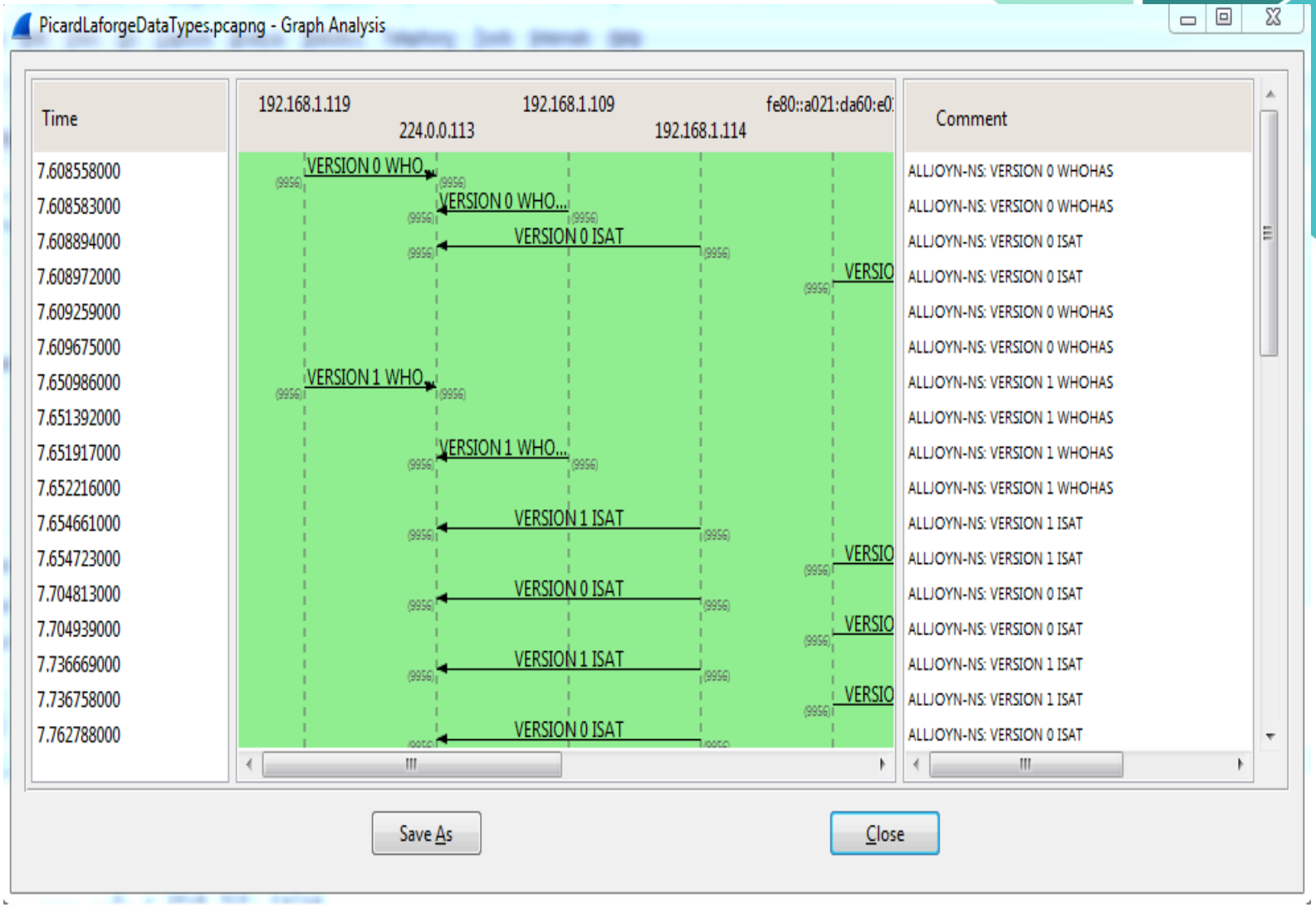
Sample output



Flow Graphs

To see a sequence diagram click on menu item "Statistics", click on "Flow Graph", then click "Okay" on the dialog box that comes up.

Sample output



Report bugs

If you encounter bugs please file them here:
https://bugs.wireshark.org/bugzilla/

You can also download the source to fix bugs and/or add features by getting involved in the Wireshark project:
http://www.wireshark.org/develop.html

Thank You.