asn1.io/ CDR Inspector

zooming into call records

by OSS NOKALVA, INC.

Overview

Purpose

- Troubleshoots Call/Charging Data Records (CDRs)
- Various formats supported (GSMA TAP, 3GPP TS 32 205, 3GPP TS 32 298, and more) •
- •
- Improves the accuracy of billing records Allows scripting over decoded results (exported as JSON)

Scenarios

- Find errors across multiple records Find data mismatches with a schema •
- •
- Detect and recover malformed TLV data
- Handle large encodings and/or large number of records
- Detect the record type
- Detect CDR file headers
- Programmatically process and interpret the decoding results



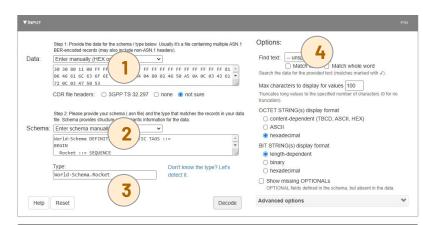
Focusing on data

With CDR Inspector the schema becomes "secondary", while a variety of file formats, including large files, can be decoded and/or errors detected.

- The **schema is optional**. Decoding raw BER can be done without the schema. This helps to catch low level encoding errors, and make general sense of data, e.g. search for text fragments, etc.
- If the **data mismatch the schema type**, the closest matching type can be auto-detected.
- CDR files can contain **one or more records.**
- Large files with thousands of records can be examined in chunks.
- CDR files with headers are supported, including **auto detection of headers**.

Example 1

- Specify the data by either uploading a file or pasting as hex text.
- 2. Optionally pick the corresponding schema and type.
- 3. Note, CDR Inspector can help you with detecting the type by scanning the schema for the best match for your data.
- 4. Use options to control the output, to search for text, and more.
- 5. Navigate records, nodes, and low level encoding details.

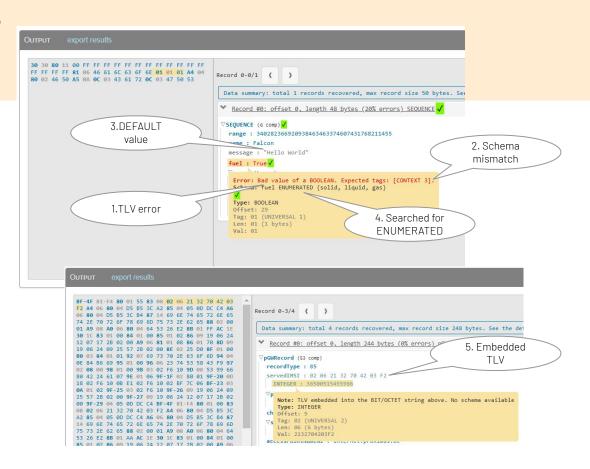




Advanced features

| Advanced features | Generic CDR decoders | CDR Inspector |
|---|-----------------------------------|------------------------------------|
| TLV errors | 🗶 skipping, breaking | ✔ detecting, revealing, recovering |
| Schema mismatch | 🗶 breaking | ✔ detecting, revealing, recovering |
| Decoding DEFAULTs and OPTIONALs | x excluded from the output | ✔ shown in the output |
| TBCD strings | × undetected | ✓ detected |
| Search decoding results | × none | ✓ all |
| TLVs embedded inside OCTET/BIT STRINGS | ✗ undetected | ✓ detected |
| Merging constructed encodings of multi-fragmented strings | 🗴 unsupported | ✓ supported |

Example 2



- 1. Low level errors in catching bad **TLV encodings**
- 2. High level errors in catching schema mismatches
- 3. Showing **DEFAULTs and OPTIONALs**
- Search for "ENUMERATED" the record and the field containing the searched text is marked with ✓
- 5. Detecting **TLVs embedded** in OCTET/BIT STRINGS

Power Tips

- **Re-inspect parts** of your data by selecting the HEX portion of a record, and decoding it with CDR Inspector again, potentially specifying a different schema type.
- Use "Sort by errors" to find bad records quickly.
- **Export** the inspector's output in **JSON format** for programmatic processing and analysis of decoding results.

DIFFERENTIATORS

OSS offers more

ACCESSIBILITY

The online platform requires no installation and provides access from any browser on any platform.

SUPPORT

OSS offers comprehensive technical support with a long list of customers and decades of history.

EXPERTISE

OSS employed the industry experts for all aspects of using ASN.1 in a right way

EASE OF USE

OSS' web tools are user-friendly and produce intuitive results. The UI offers a wealth of options to fit your scenario.