Specifying binary file formats for TAIGA data sharing and reuse

Andrey Mikhailov
mikhailov@icc.ru

Matrosov Institute for System Dynamics and Control Theory
SB RAS

May 30, 2018

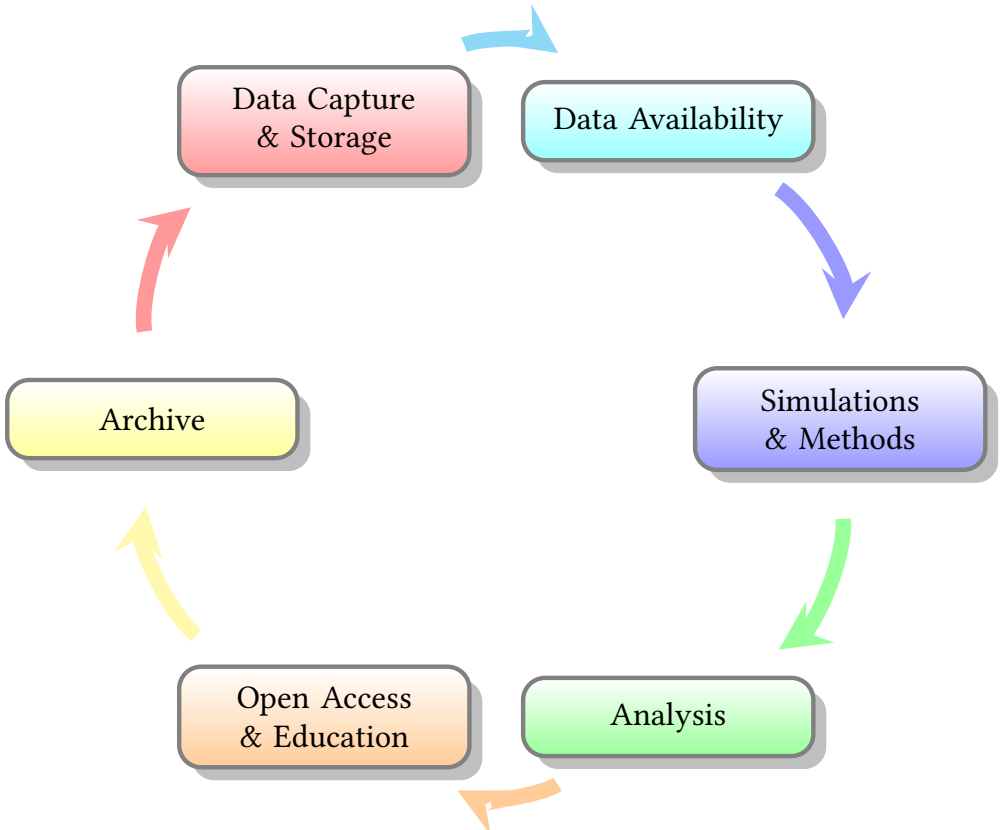
Motivation

TAIGA formats:

- TAIGA-IACT
- TUNKA-HISCORE
- TUNKA-133
- TUNKA-REX
- TUNKA-GRANDE

Issue:

- No documentation, no open raw data access API
- How to verify and reuse raw data?



Astroparticle Data Lifecycle

Solution

Binary data

```
0E1F BA0E 00B4 09CD 21B8
6973 2070 726F 6772 616D
7420 6265 2072 756E 2069
6D6F 6465 2E0D 0D0A 2400
```

Data format specification

```
meta:
  id: tcp_segment
  seq:
    - id: src_port
    type: u2
    - id: dst_port
    type: u2
    - id: seq_num
    type: u4
```

Parsing library in a target language: C, C++, Java, Go, JS, Python, Ruby, etc.

```
public class TcpSegment extends KaitaiStruct {
  //...private void _read() throws IOException {
    this.srcPort = _io.readU2be();
    this.dstPort = _io.readU2be();
    this.seqNum = _io.readU4be();
    this.ackNum = _io.readU4be();
  } ...
}
```

What is Kaitai Struct

Declarative: describe the very structure of the data, not how you read or write it

Packed with tools and samples: includes a *compiler*, an *IDE*, a *visualizer* and library of format specs

Free & open source: licensed under the following terms: GPLv3+ (Compiler and visualizer), MIT or Apache v2 (Runtime libraries)

Language-neutral: write once, use in all supported languages:

- C++/STL
- C#
- Go (*)
- Java
- JavaScript
- Lua
- Perl
- PHP
- Python
- Ruby

(*) entry-level support

Mikhail Yakshin, "'Kaitai Struct – a new way to develop parsers for binary structures."', 2018. [Online]. Available: https://kaitai.io. [Accessed: 29- May- 2018]

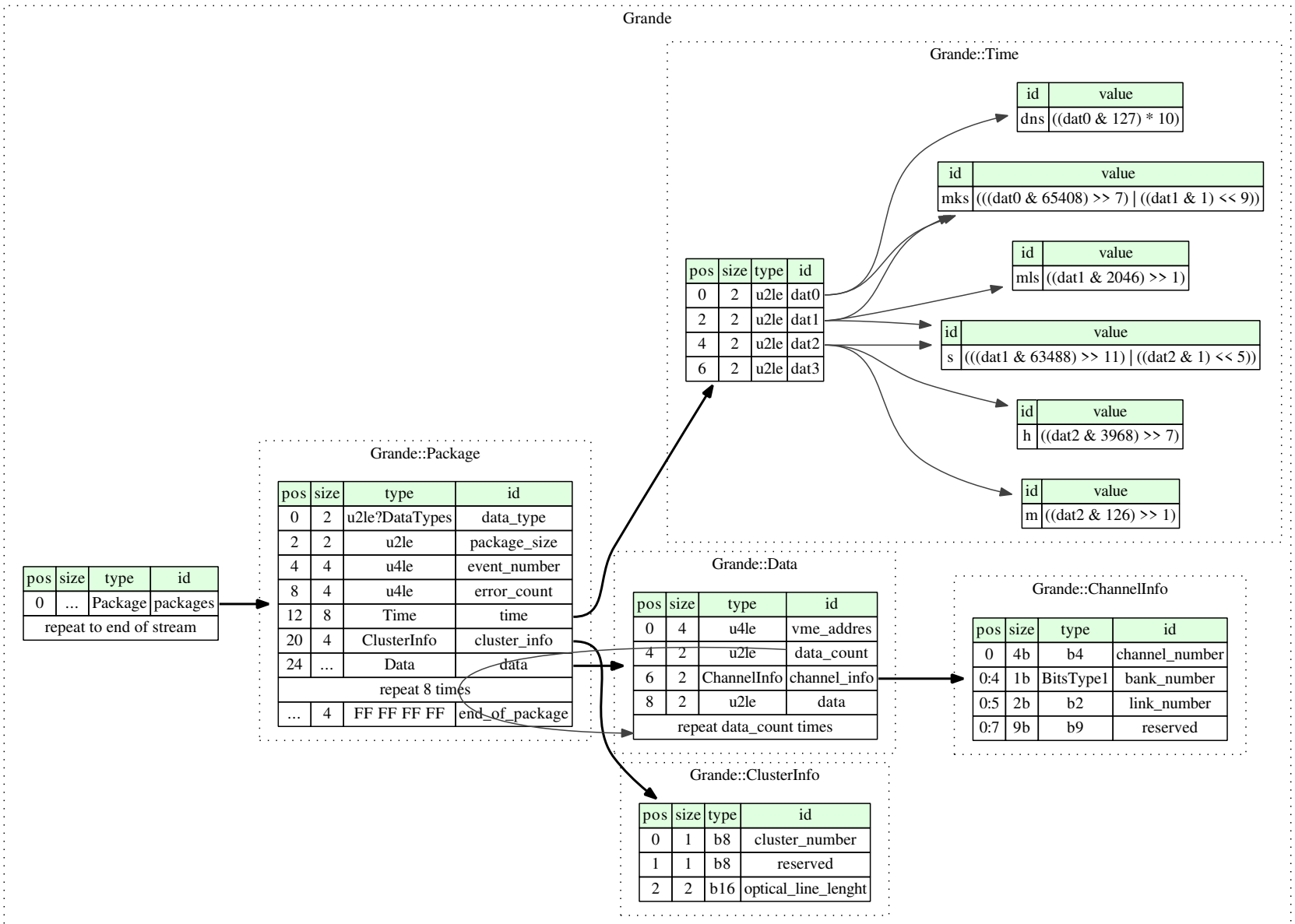
What have been done

- TAIGA format specifications
- Reading libraries
- Examples for C++, Java, and Python
- The libraries tested on real data

- GRANDE, T133, and TREX tested on ≈ 89 000 files for season 2016
- HiSCORE, IACT tested on ≈ 120 000 files

- 4 % of GRANDE, T133, and TREX files have BAD file format
- 0,6 % of HiSCORE and IACT files have BAD file format

GRANDE specification diagram



How to use. C++ example

```
// Add KS runtime library
...
#include <kaitaistruct.h>
...
// Include generated class
#include "hiscore.h"
...
ifstream ifs(fileName, ifstream::binary);
// Make Kaitai Struct stream
kaitai::kstream ks(&ifs);
// Read HiSCORE file by generated library
hiscore_t hiscore = hiscore_t(&ks);
// Get all packages
vector<hiscore_t::package_t*>* packages = hiscore.packages();
vector<hiscore_t::package_t*>::iterator it = packages->begin();
// Print some infos
for (it; it != packages->end(); ++it) {
  hiscore_t::package_t* package = (hiscore_t::package_t*)*it;
  hiscore_t::header_t* header = package->hdr();
  printf("Event number: %d\n", header->event_number());
  printf("IP:           %d\n", header->ip());
  printf("Magic:        %d\n", header->magic());
}
```

Specifying binary file formats for TAIGA data sharing and reuse

Andrey Mikhailov
mikhailov@icc.ru

Matrosov Institute for System Dynamics and Control Theory
SB RAS

May 30, 2018