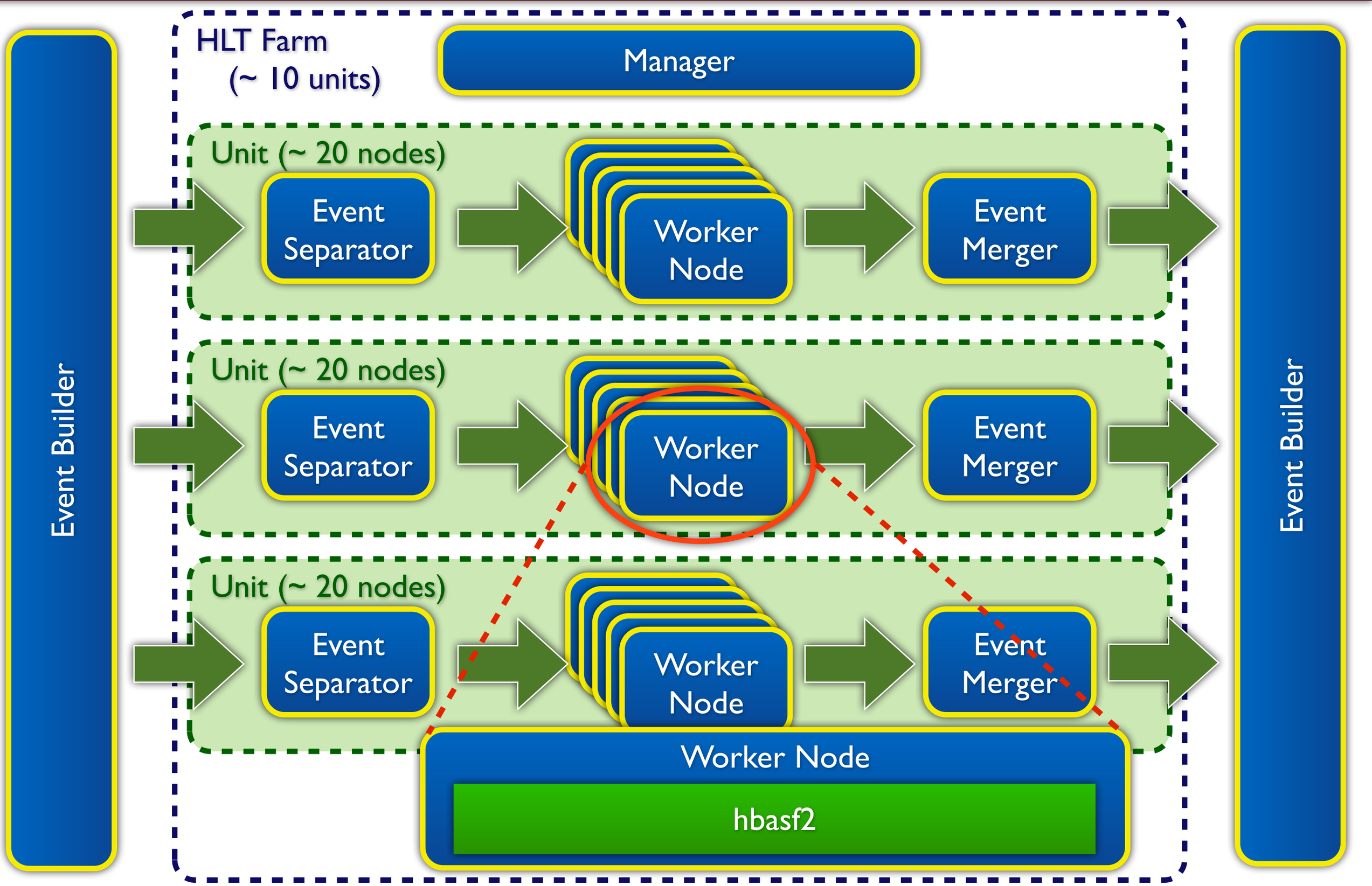
HLT Status

Belle II Trigger/DAQ Workshop, Jan 16-18 2012
University of Hawaii, Hawaii, US

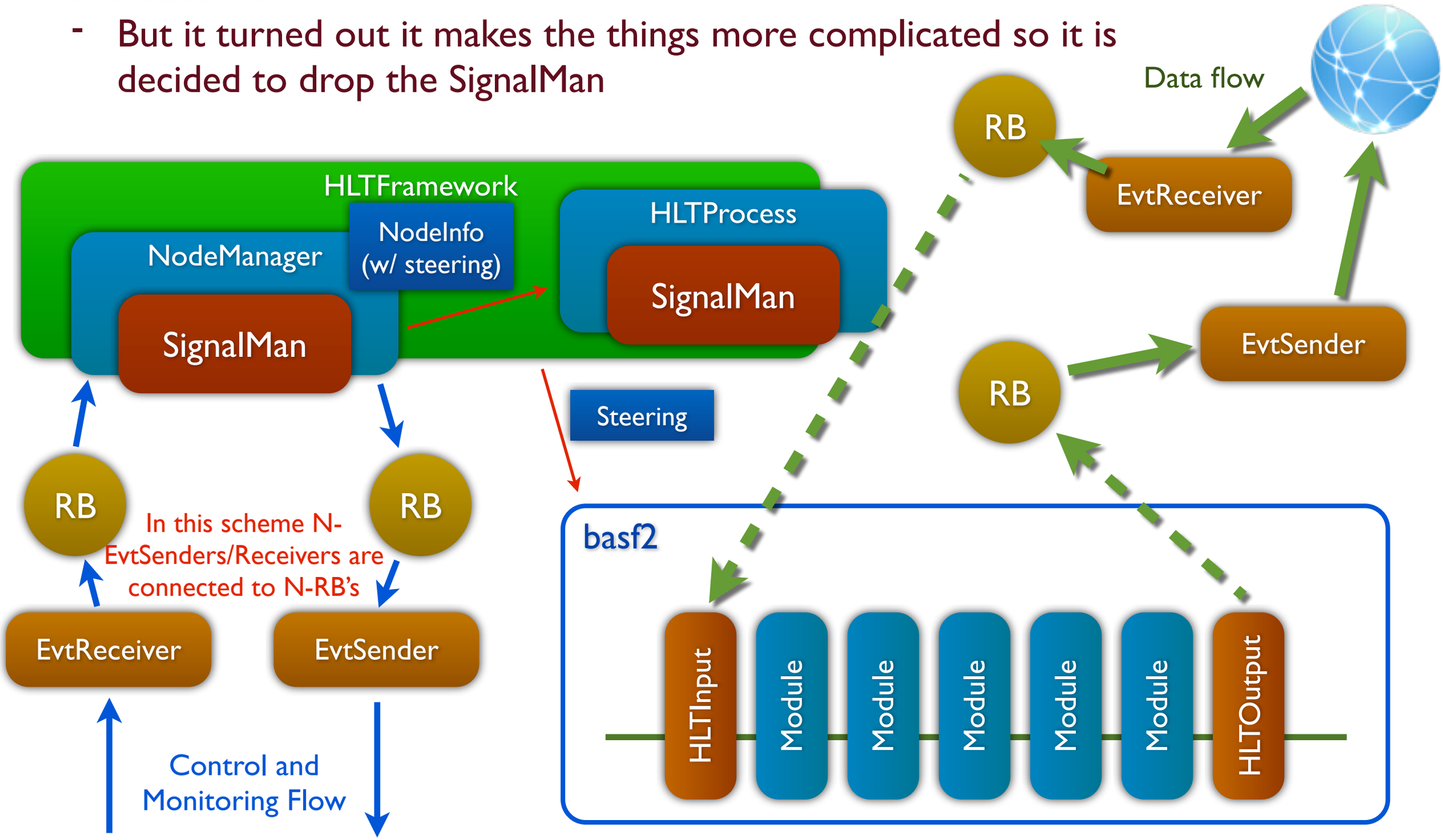
Soohyung Lee (Korea University)

HLT Structure



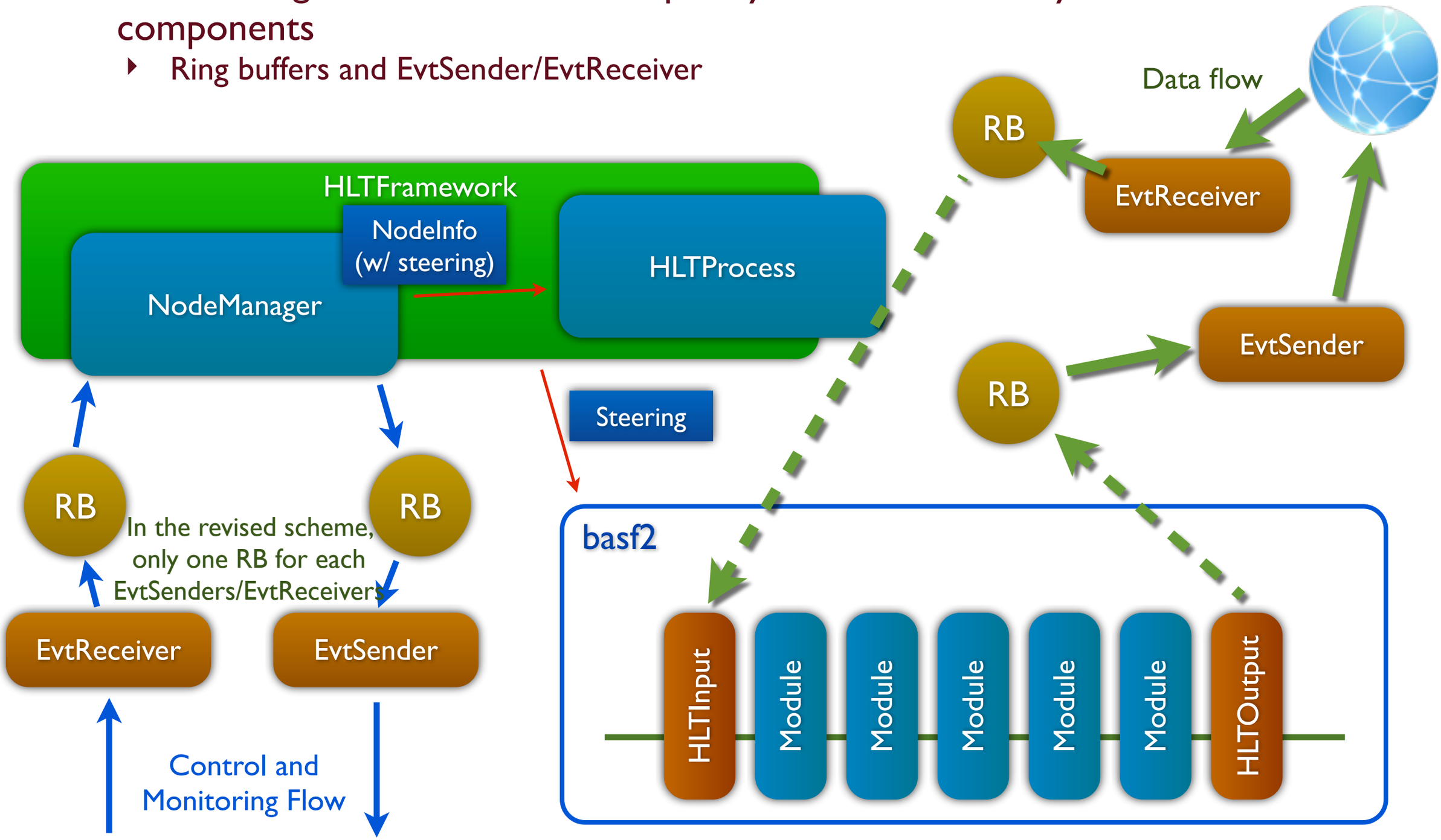
Data Flow

- In the previous design, SignalMan is a unit component of the data flow
  - But it turned out it makes the things more complicated so it is decided to drop the SignalMan



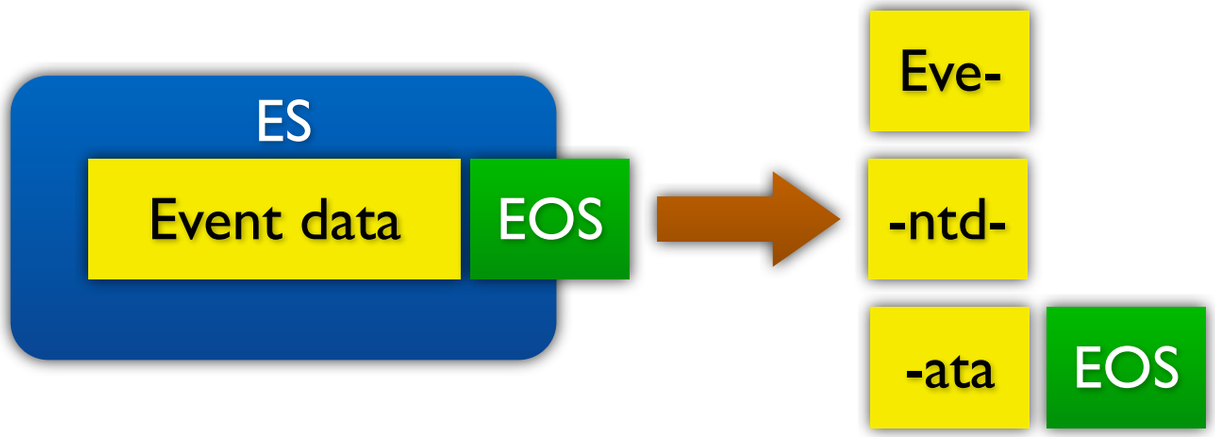
Data Flow (revised)

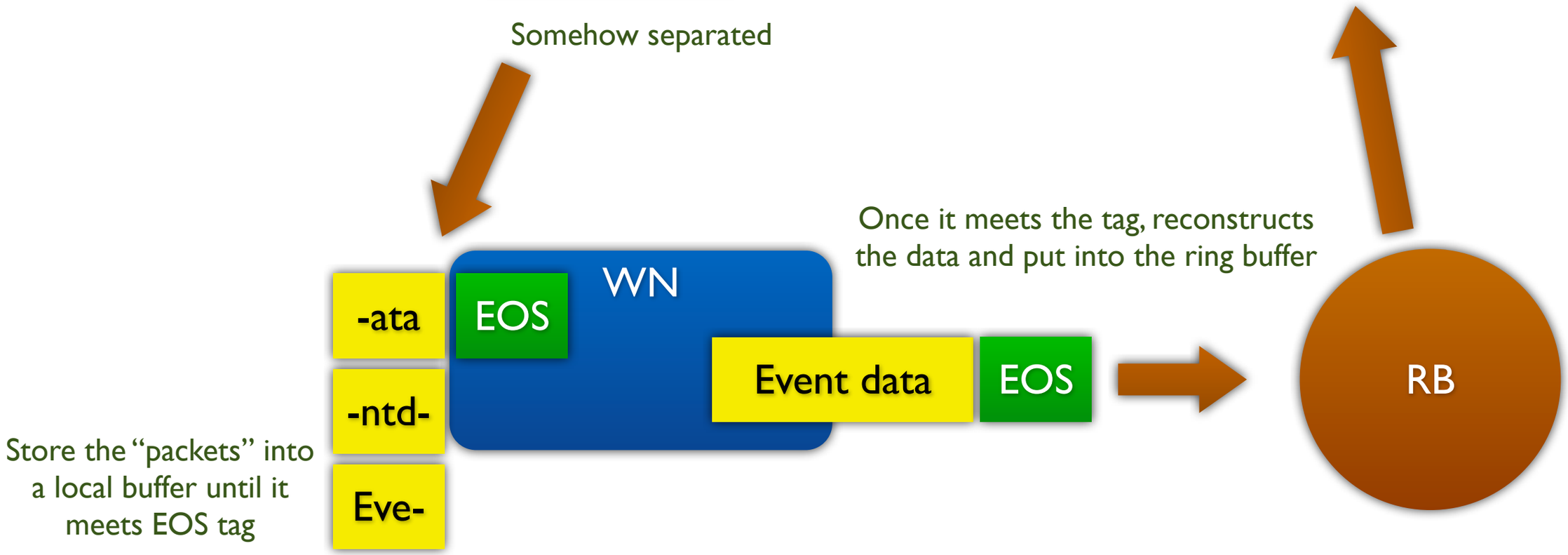
- No SignalMan anymore
  - NodeManager and HLTProcess explicitly initialize necessary components
    - Ring buffers and EvtSender/EvtReceiver



Unexpected Separation of Data

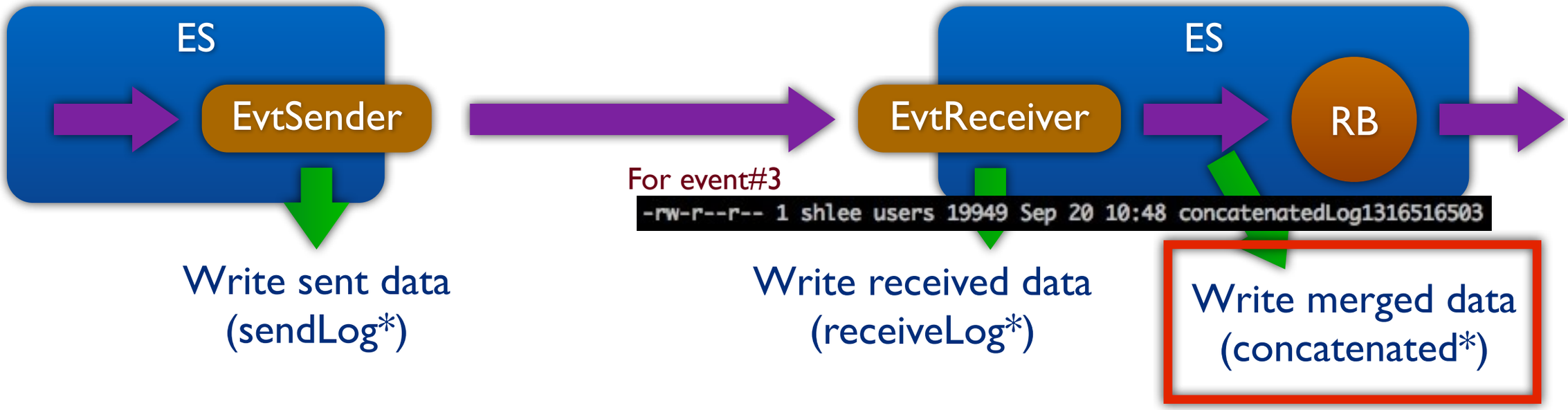
- I assumed that one TCP operation guarantees a "capsule" of a event data but I found that TCP arbitrary separate the data somehow
  - In order to fix this, very naive solution is applied for now





Quick Test

- By writing the data into binary files, this can be confirmed



- 3 events are generated and simulated for the test

```
-rw-r--r-- 1 shlee users 19949 Sep 20 10:48 sendLog1316516085-19949
-rw-r--r-- 1 shlee users 19573 Sep 20 10:48 sendLog1316516085-19573
-rw-r--r-- 1 shlee users 19365 Sep 20 10:48 sendLog1316516085-19365
```

Event#1: 19365 bytes
Event#2: 19573 bytes
Event#3: 19949 bytes

  - Event#2 and #3 are separated during the transfer

```
[INFO] [EvtReceiver] 19365 bytes received
[INFO] SignalMan: INTERNAL BUFFER FLUSHING
```

Event#1

```
[INFO] [EvtReceiver] 2896 bytes received
[INFO] SignalMan: Storing partial data into internal container
[INFO] SignalMan: Merging 2896 at offset 0
454cf010
[INFO] [EvtReceiver] 16677 bytes received
[INFO] SignalMan: Merging 16677 at offset 2896
454cfb60
[INFO] SignalMan: INTERNAL BUFFER FLUSHING
```

Event#2 (2896+16677=19573)

```
[INFO] [EvtReceiver] 2896 bytes received
[INFO] SignalMan: Storing partial data into internal container
[INFO] SignalMan: Merging 2896 at offset 0
[INFO] [EvtReceiver] 17053 bytes received
[INFO] Storing arrays...0
Collection name='Belle2::CDCHits', class='TC
 OBJ: Belle2::CDCHit    Belle2::CDCHit
```

Event#3 (2896+17053=19949)

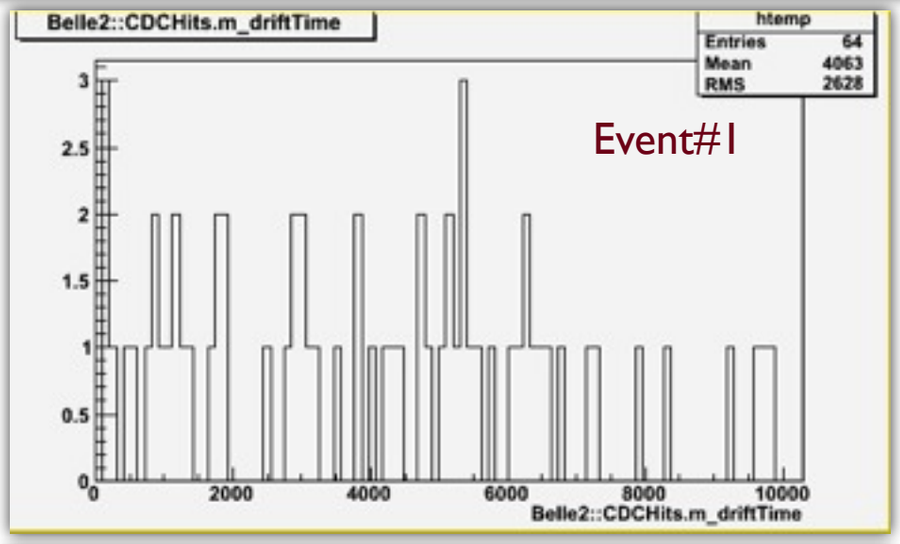
Quick Test (Cont'd)

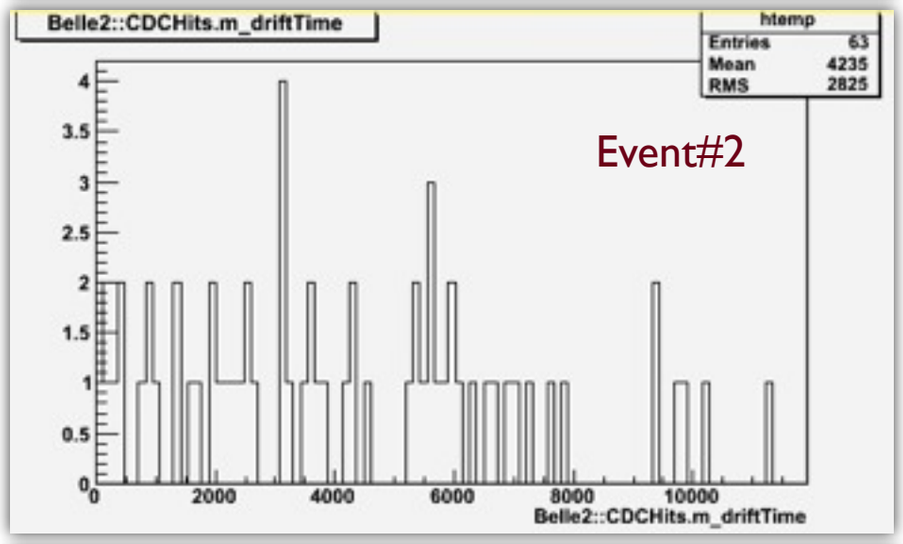
- Output ROOT files (@HLTInput level) are well written

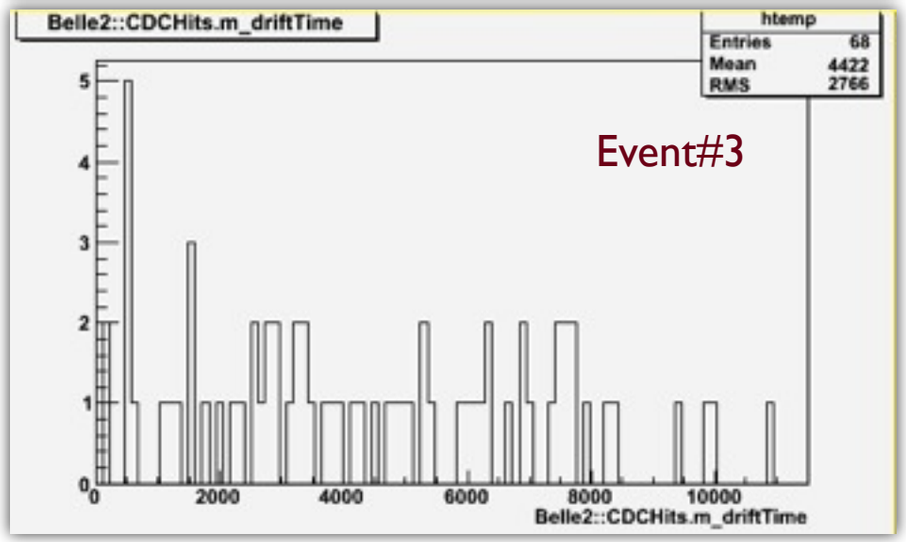
```
-rw-r--r-- 1 shlee users 66645 Sep 20 10:48 HLTInput1316516503-19364.root
-rw-r--r-- 1 shlee users 66916 Sep 20 10:48 HLTInput1316516504-19569.root
-rw-r--r-- 1 shlee users 67369 Sep 20 10:48 HLTInput1316516504-19946.root
```

- Working behavior is getting stable
  - No crash after 1,000 event processing

```
[INFO] SignalMan: Merging 2896 at offset 0
970be010
[INFO] [EvtReceiver] 17005 bytes received
[INFO] SignalMan: Merging 17005 at offset 2896
970beb60
[INFO] SignalMan: INTERNAL BUFFER FLUSHING
[INFO] [EvtReceiver] 2896 bytes received
[INFO] SignalMan: Storing partial data into internal container
[INFO] SignalMan: Merging 2896 at offset 0
970be010
[INFO] [EvtReceiver] 17313 bytes received
[INFO] SignalMan: Merging 17313 at offset 2896
970beb60
[INFO] SignalMan: INTERNAL BUFFER FLUSHING
[INFO] [EvtReceiver] 2896 bytes received
[INFO] SignalMan: Storing partial data into internal container
[INFO] SignalMan: Merging 2896 at offset 0
970be010
[INFO] [EvtReceiver] 16149 bytes received
[INFO] SignalMan: Merging 16149 at offset 2896
970beb60
[INFO] SignalMan: INTERNAL BUFFER FLUSHING
[INFO] [EvtReceiver] 2896 bytes received
[INFO] SignalMan: Storing partial data into internal container
[INFO] SignalMan: Merging 2896 at offset 0
970be010
[INFO] [EvtReceiver] 28785 bytes received
[INFO] SignalMan: Merging 28785 at offset 2896
970beb60
[INFO] SignalMan: INTERNAL BUFFER FLUSHING
```

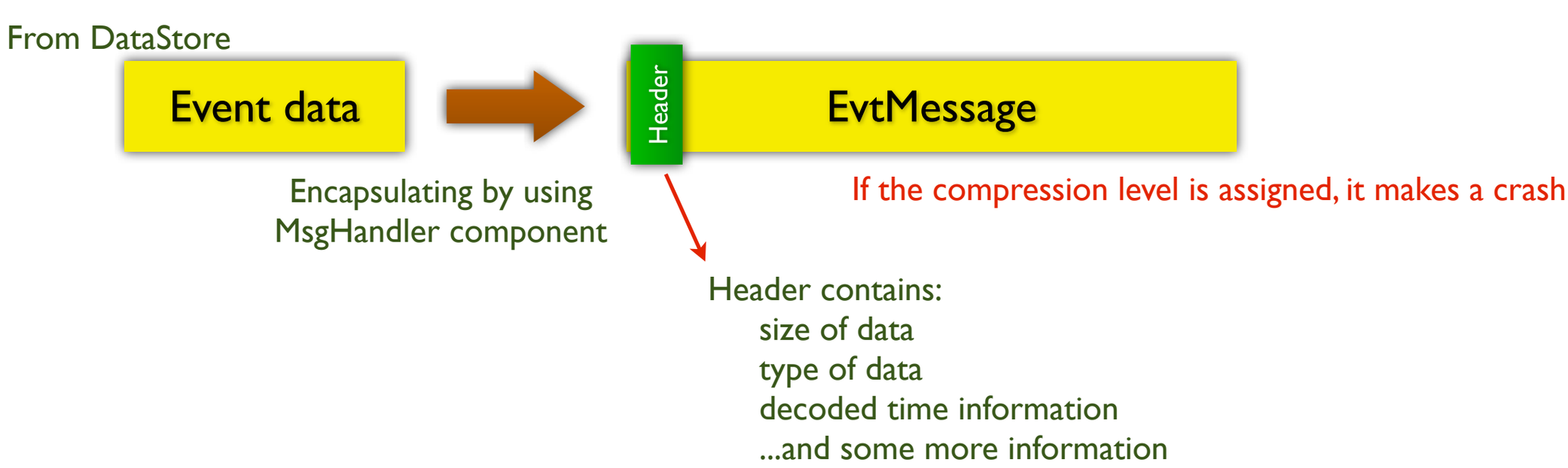






DataStore Problem

- At the moment we did a quick test, the data in DataStore weren't reconstructed in other node well
  - It turned out there is some problem in TMessage class related to the compression level
  - With compression level = 0 (no compression of data), the data reconstruction in other node started to work

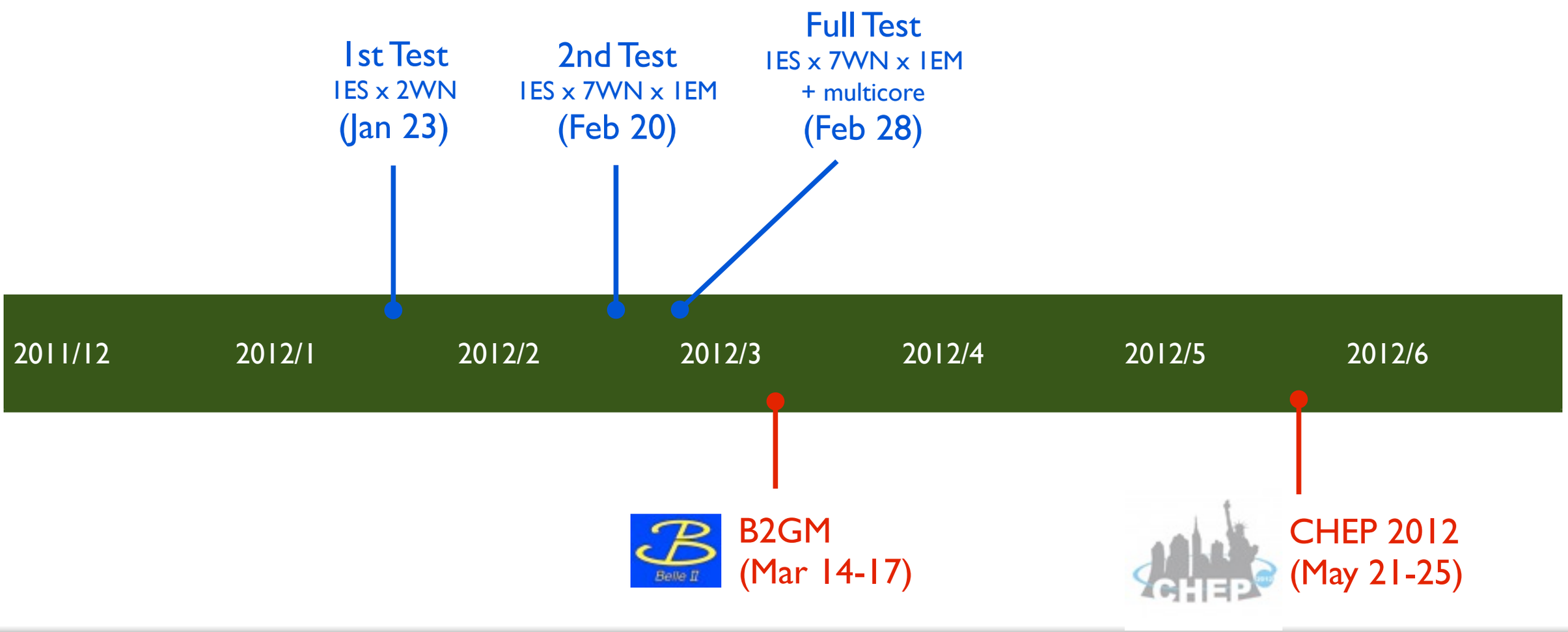


Current Status

- Previous (partial) working version
  - Management System
    - ☑ XML parsing works well
    - ☑ Node information reconstruction @ manager node works well
    - ☑ Node information distribution over network works well
    - ☑ Node information reconstruction @ process node works well
    - ☐ Node information serialization using EvtMessage is required
  - Data Flow over Network
    - ☑ Data serialization using EvtMessage works well
    - ☑ Data transfer over network looks working
    - ☑ Separated data handling looks working (but need further improvement)
    - ☐ Data reconstruction is not clearly confirmed
    - ☐ Multiprocess handling is problematic
    - ☐ Process termination is incomplete
    - ☐ EM node implementation is not done
    - ☐ Persistent connection of B2Socket is required

Development Roadmap

- Tests
  - 1st test (Jan 23)
    - Process management and complete data flow
    - There will (should) be a lot of problems as usual
  - 2nd test (Feb 20)
    - Supposed to be complete as hbasf2
  - Full test (Feb 28)
    - Combining with multicore-based parallel processing




KOREA
UNIVERSITY
1905