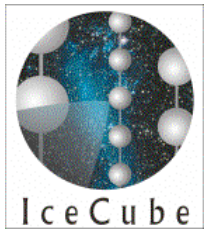


DOM-Cal: Update and Tutorial

John Kelley and Jim Braun

UW-Madison

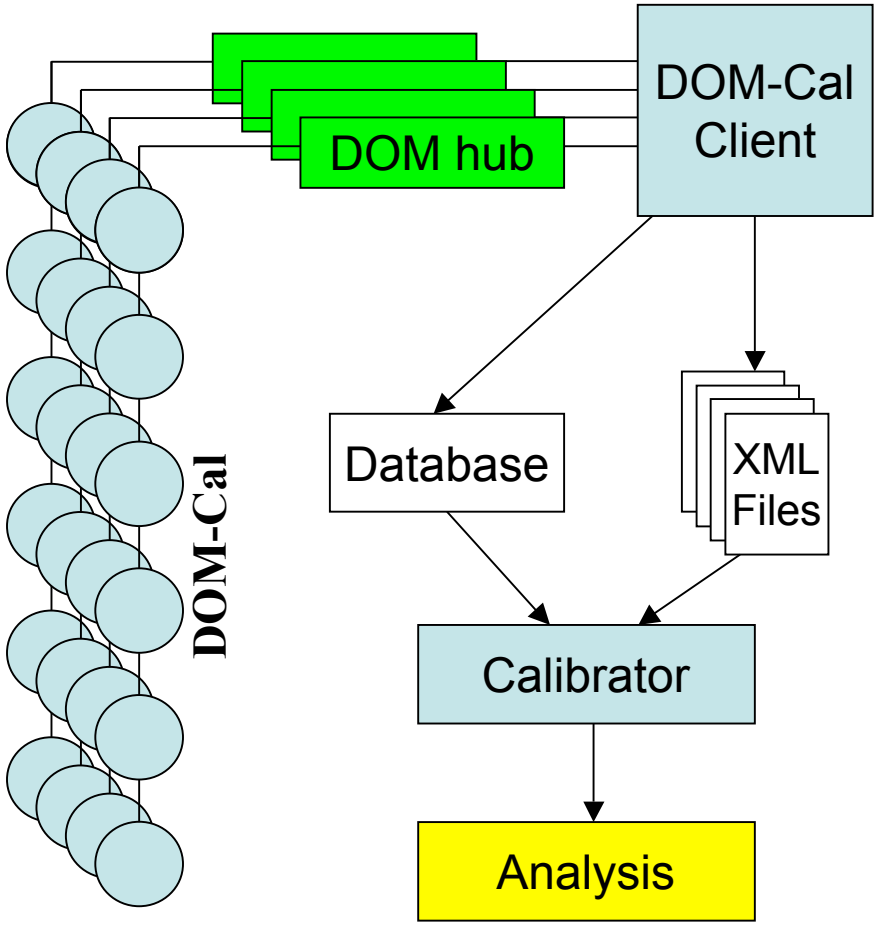
September 26, 2005

IceCube Collaboration Meeting

London

Overview

- Status of DOM-Cal
  - Review of capabilities
  - Recent improvements
  - Transit time calibration
- Tutorial on use of calibration results



DOM-Cal

- Raw waveforms to time & voltage

- Calibrates:
  - ATWD
    - voltage
    - sampling freq.
  - channel gains (improved!)
  - PMT gain (wider range!)
  - transit time (new!)

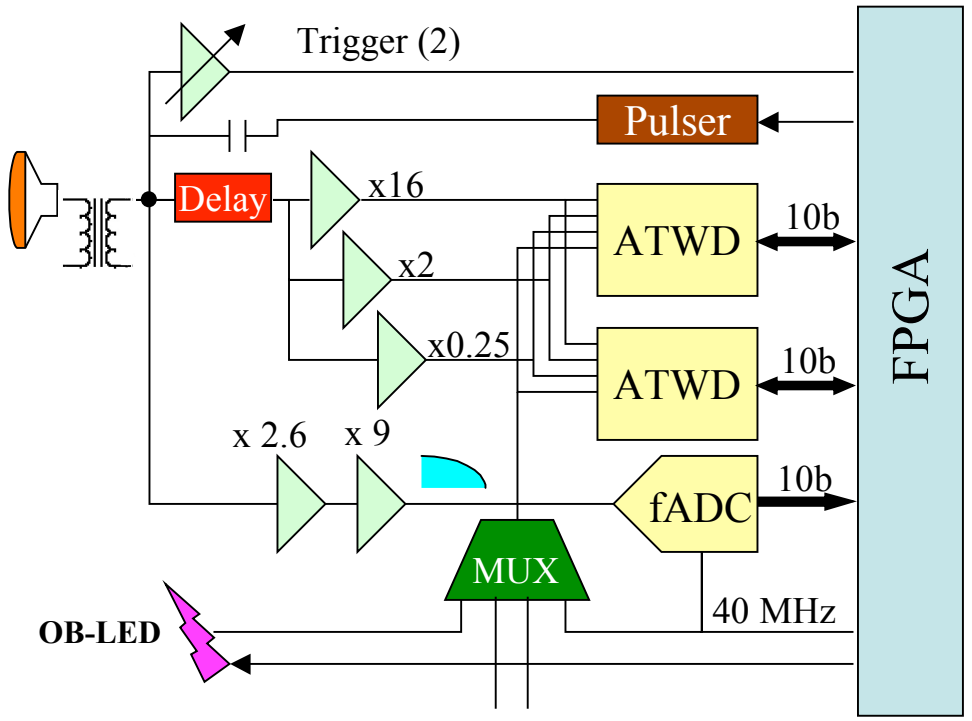
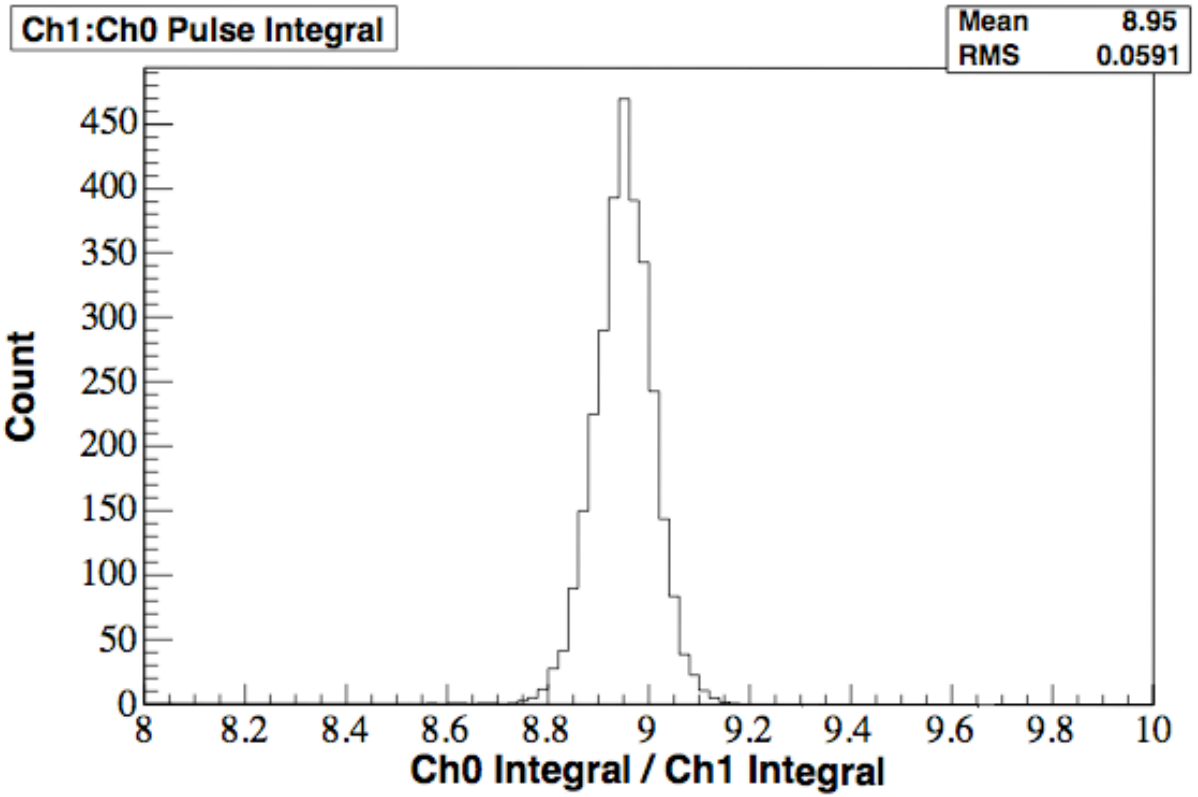


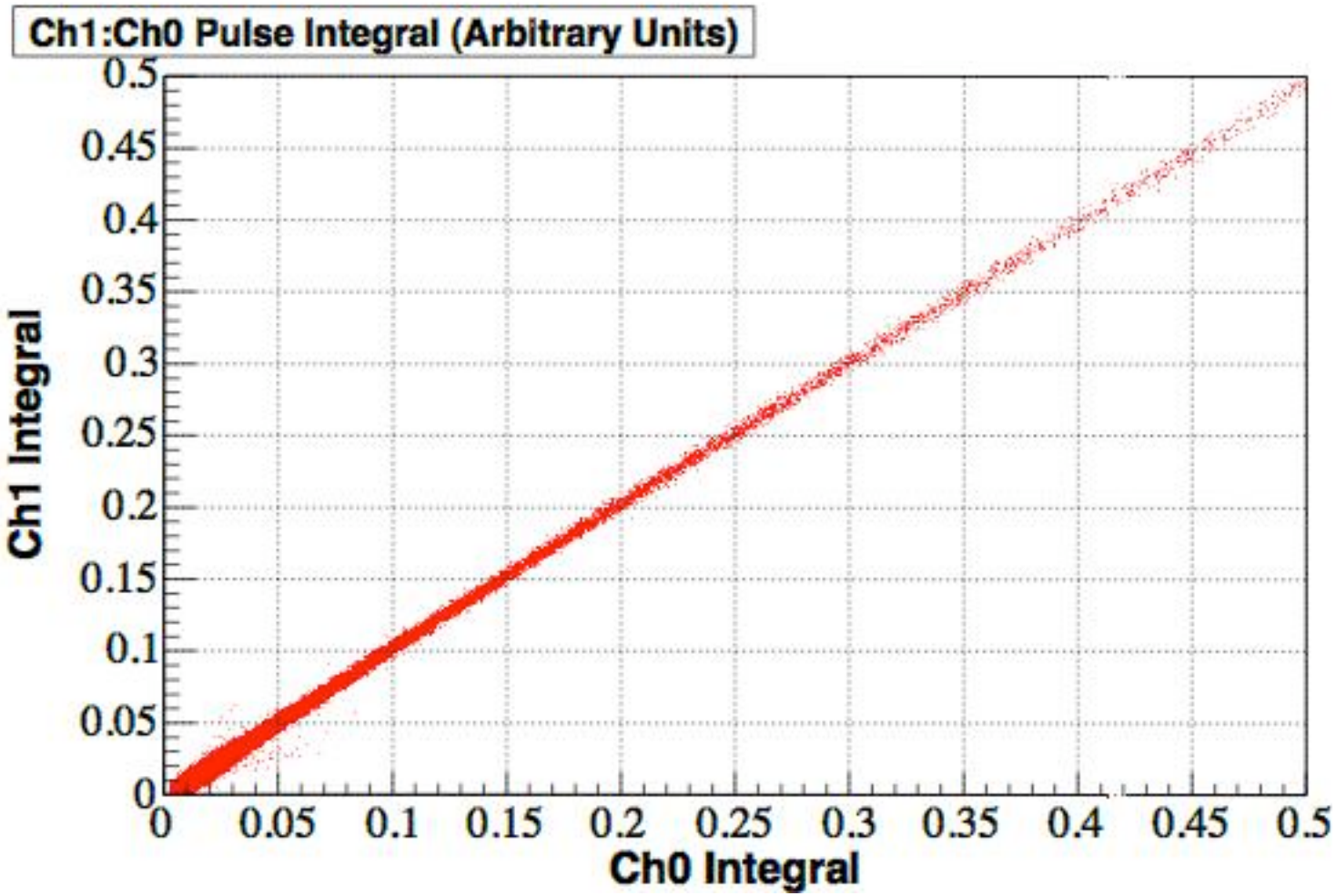
Figure: modified from G. Przybylski

Channel Gain

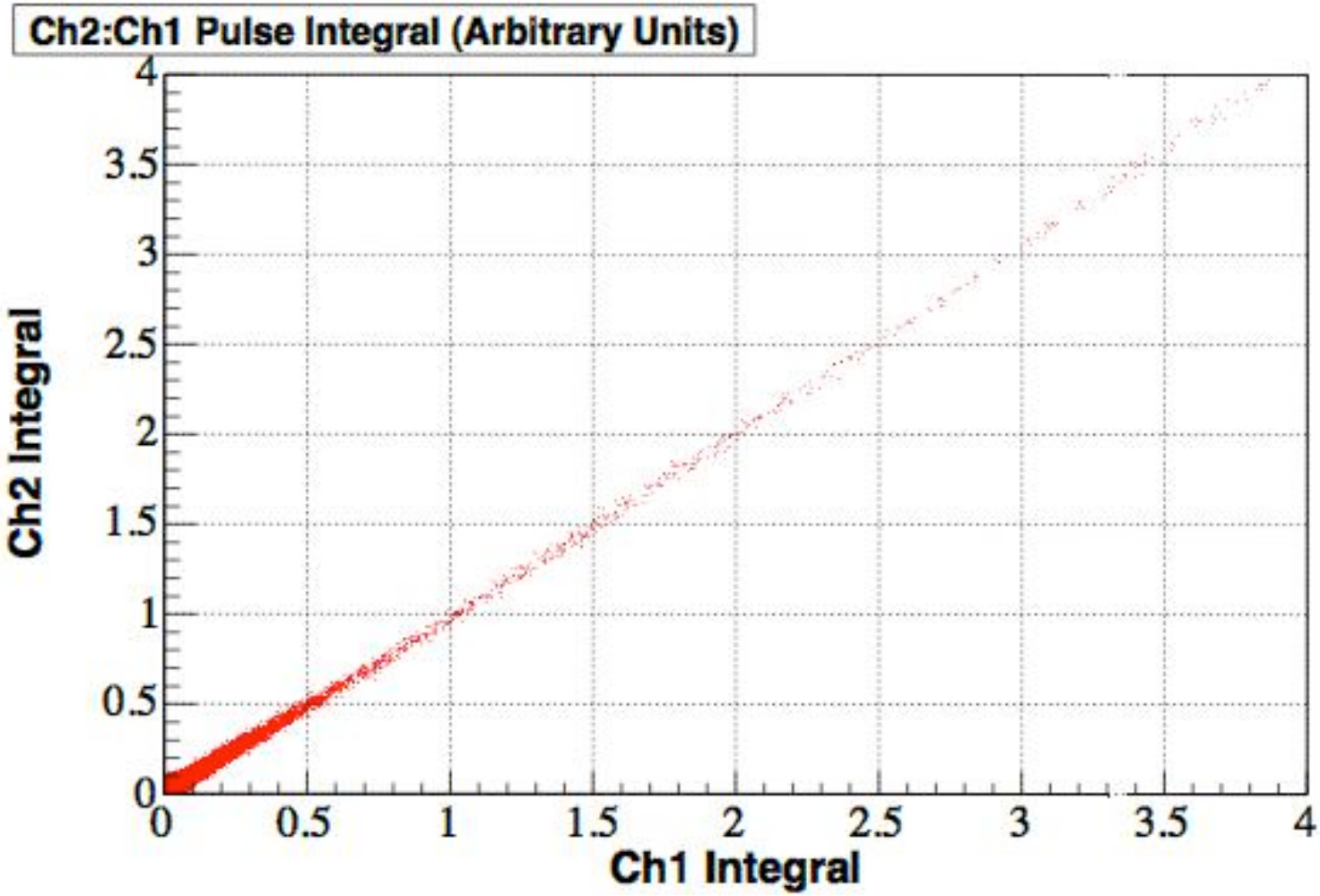


- Older versions (<5.0) showed poor ch1:ch0, ch2:ch1 charge ratio
- Many improvements
  - use LED for ch1, ch2
  - careful baseline subtraction
  - conserve charge, not amplitude

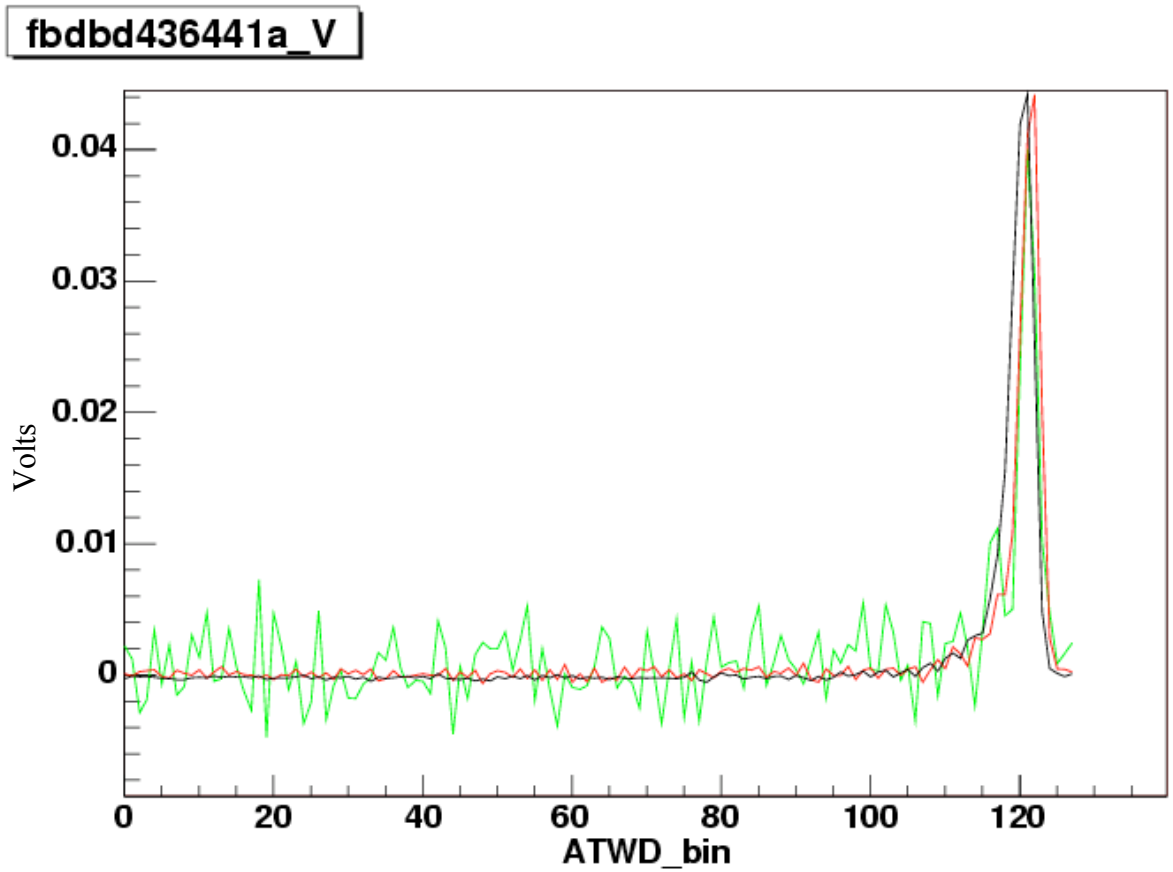
Ch1:Ch0 Charge



Ch2:Ch1 Charge

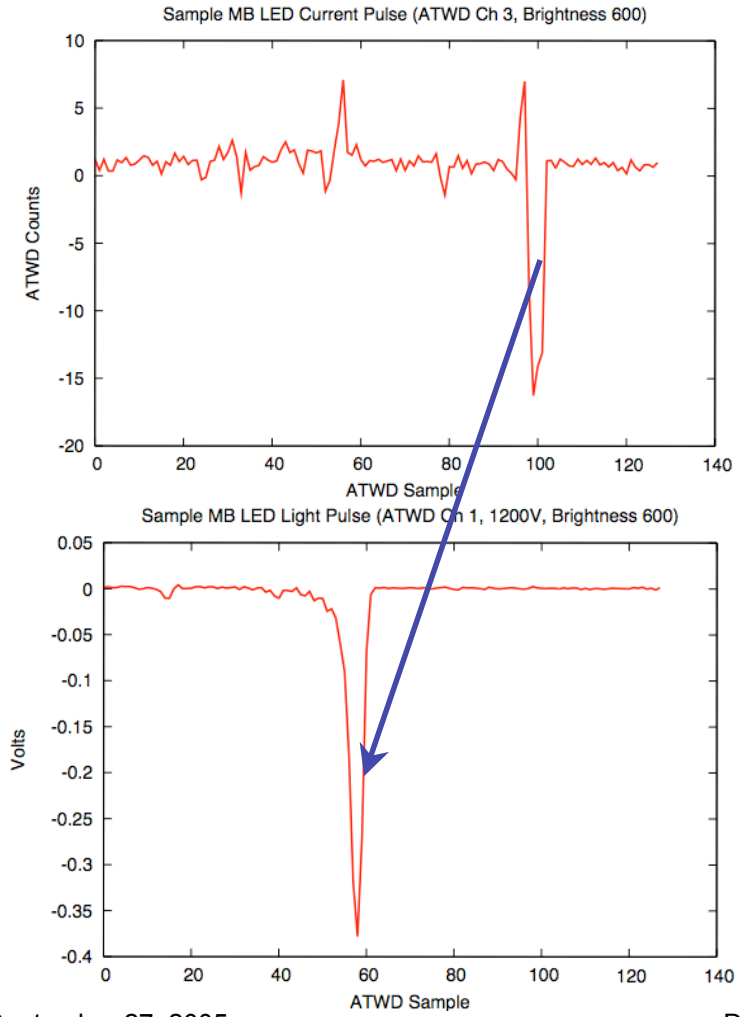


Calibrated Amplitude



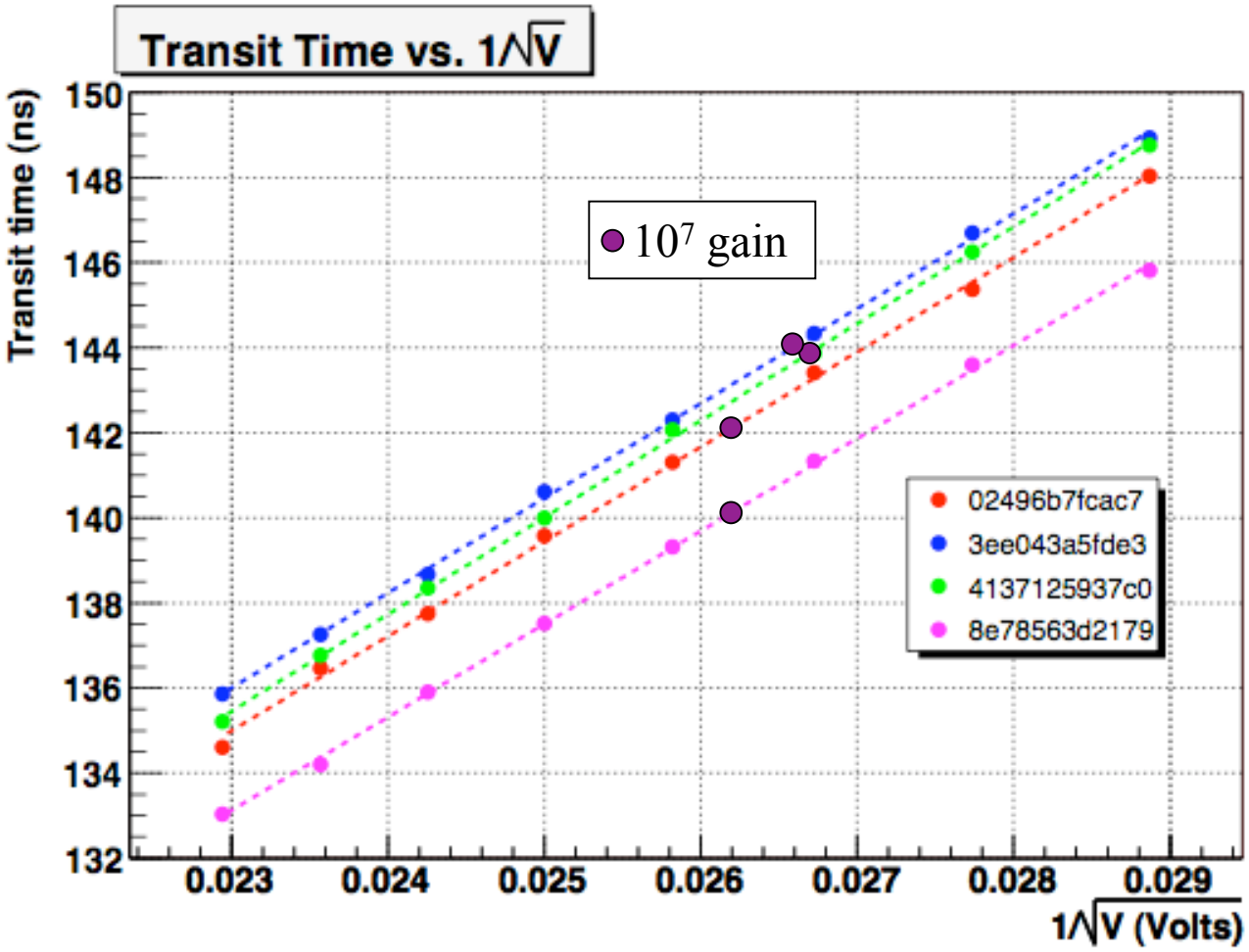
Amplitudes also agree well in all channels (although amplifier response not uniform!)

Transit Time

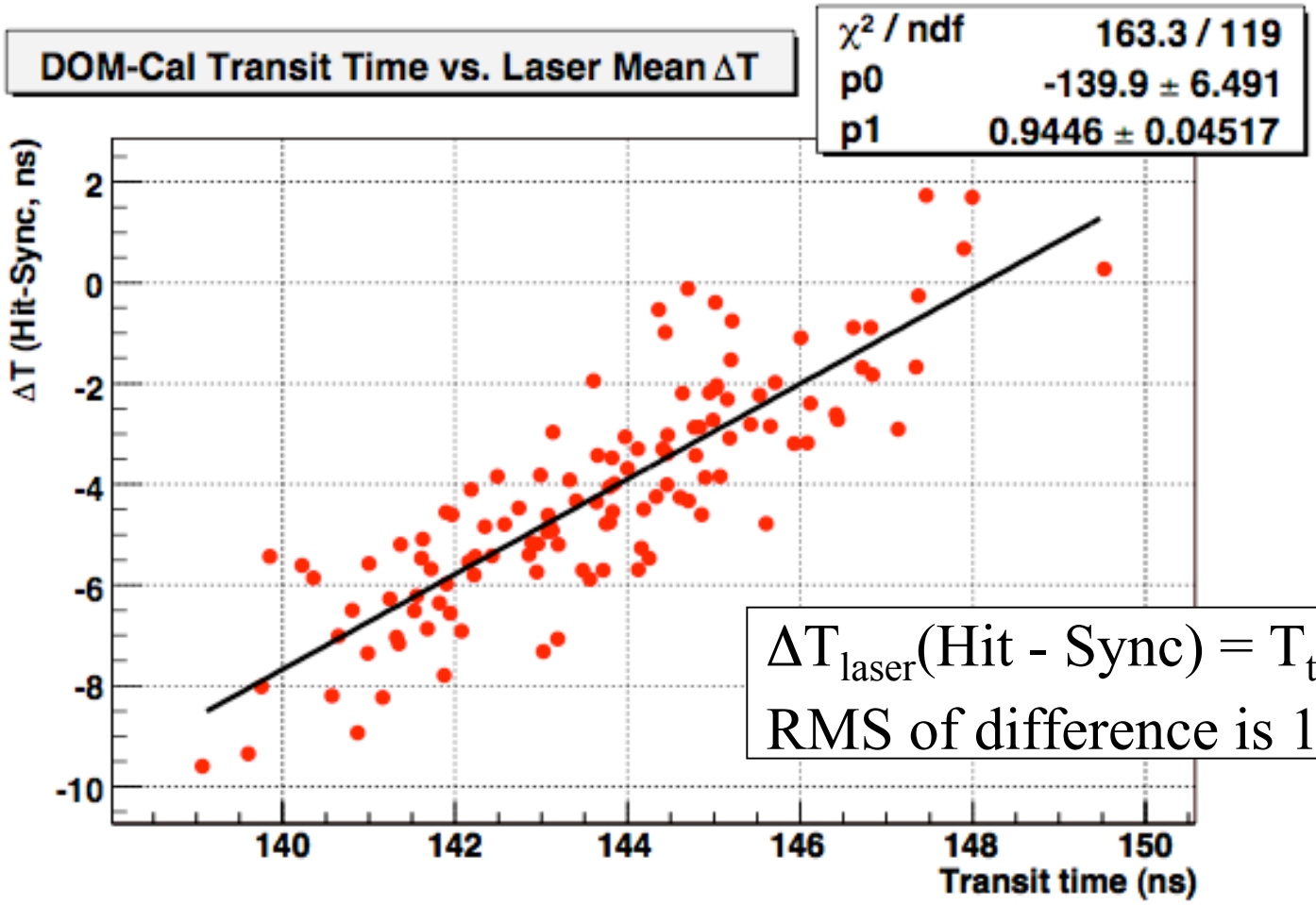


- Measure PMT transit time + delay board
- Flash mainboard LED
- LED-trigger the ATWD
- Record current pulse in channel 3, light pulse in channel 1
- Find 50% point of leading edges, convert difference to nanoseconds

Transit Time Data



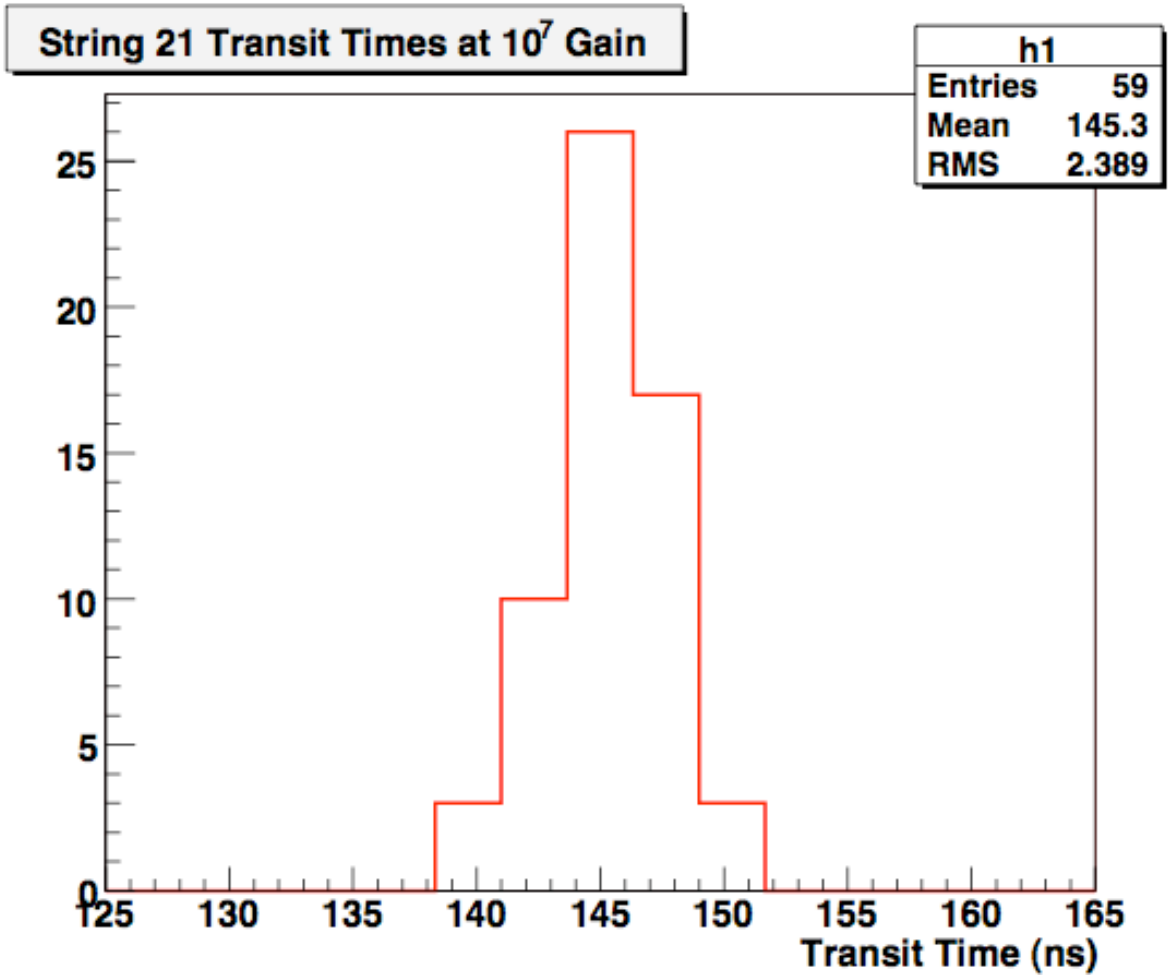
DFL Laser Comparison



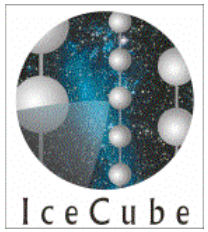
$\Delta T_{laser}(\text{Hit - Sync}) = T_{transit} + C$

RMS of difference is 1.2 ns

String 21 Transit Times

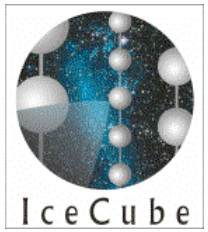


- First transit-time run on String 21 (20 Sept)
- Sadly, 20% of DOMs failed calibration, but DOM-Cal 5.13 addresses this



Open Calibration Issues

- In-ice transit time calibration issues fixed, in testing
- Signal droop for long pulses (base toroid) is not calibrated
- FADC is not calibrated (yet)
- Otherwise, DOM-Cal is rather mature — but feel free to contact us if you see anything weird!



Using DOM-Cal Results

- **`Icecube.daq.domcal.Calibrator`** java class provides access to calibration data.
- Create a Calibrator for each DOM:
  - **`new Calibrator(XML_File);`**
- Most important routine:
  - **`atwdCalibrateToPmtSig()`**
    - Takes an array of raw ATWD data and applies calibration to yield true voltage signal
- Other methods to access raw data, described in javadoc
  - http://www.amanda.wisc.edu/jbraun/domcaldoc/

XML Result File

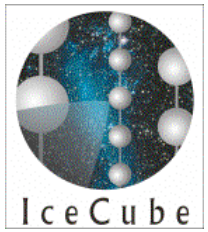
```
<domcal version="5.11">
  <date>8-22-2005</date>
  <domid>02496b7fcac7</domid>
  <temperature format="Kelvin">236.91</temperature>
  <dac channel="0">850</dac>
  <dac channel="1">2300</dac>
  <dac channel="2">350</dac>
  <dac channel="3">2250</dac>
  <dac channel="4">850</dac>
  <dac channel="5">2300</dac>
  <dac channel="6">350</dac>
  <dac channel="7">2130</dac>
  <dac channel="8">0</dac>
  <dac channel="9">556</dac>
  <dac channel="10">0</dac>
  <dac channel="11">0</dac>
  <dac channel="12">0</dac>
  <dac channel="13">0</dac>
  <dac channel="14">0</dac>
  <dac channel="15">0</dac>
  <adc channel="0">4</adc>
  <adc channel="1">1021</adc>
  <adc channel="2">860</adc>
  <adc channel="3">416</adc>
  <adc channel="4">78</adc>
  <adc channel="5">19</adc>
  <adc channel="6">80</adc>
  <adc channel="7">156</adc>
  <adc channel="8">683</adc>
  <adc channel="9">886</adc>
  <adc channel="10">628</adc>
```

- XML result file format documented, in docushare*
- Step-by-step instructions for use of calibration constants

*http://docushare.icecube.wisc.edu/docushare/dsweb/View/Collection-2597

Before You Start

- ATWD DAC settings (1-3,5-7) from run must match DAC settings in calibration file
- Temperature of run must be similar to calibration
- Ideally, calibration should be recent
- Know whether the data is pedestal-subtracted



Calibrating Waveforms

To calibrate raw waveforms (not pedestal-subtracted) to voltage:

1) Use **<atwd>** calibration constants for each bin:

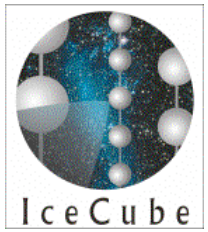
   V(id, channel, bin) = *m* * counts(id, channel, bin) + *b*

2) Subtract bias voltage (using DAC value):

   V_{bias}(Volts) = DAC[7] * (5.0 / 4096.0)

3) Remove any remnant baseline (use **<baseline>** calibration for HV setting, or roll your own)

4) Divide by channel gain from **<amplifier>** calibration



Feature Time

To calibrate time axis of a waveform:

1) Determine bin offset of whatever feature you like (be careful about counting from the trigger end of the waveform)

2) Determine sampling speed of the ATWD using **<atwdfreq>** calibration:
   sampling speed (MHz) = 20.0 (*m* * DAC[0,4] + *b*)

3) Calculate GPS feature time in ns using above along with timestamp of trigger

4) Subtract transit time at the operating high voltage using **<pmtTransitTime>** calibration:
   transit time (ns) = *m* / sqrt(V) + *b*

Waveform Charge

To calculate the charge under a waveform:

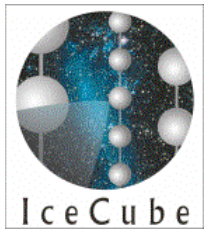
1) Transform the raw waveform to Volts

2) Convert summed voltages into amplified charge:

$$\text{charge (pC)} = 10^{12} \cdot \frac{1}{50\ \Omega} \cdot \frac{1}{\text{freq (Hz)}} \cdot \sum_i V_i$$

3) Divide by the PMT gain at the operating HV using the **<hvGainCal>** calibration:

$\log_{10}(\text{gain}) = m * \log_{10}(\text{HV}) + b$



Thanks!

- Latest version of DOM-Cal is V05-13-01 (V05-11-01 at pole)

- Please contact jkelley@icecube.wisc.edu or jbraun@icecube.wisc.edu with any questions!