Weather and Seeing Stats, Control System Upgrades

Nils Turner

18 March 2015 / CHARA Winter Meeting, Atlanta

Weather Station Uptime

	Cum.	2011	2012	2013	2014
E1	95.4	98.9	97.3	99.9	99.7
E2	89.8	94.7	97.7	93.0	62.2
S1	94.0	96.8	99.0	99.7	79.8
S2	93.6	98.2	92.8	91.7	98.5
W1	95.8	97.1	98.8	93.3	97.9
W2	94.8	97.3	98.8	92.3	99.6
L1	67.0	47.0	30.5	12.9	99.7

Table: Weather station uptimes as a percentage of time.

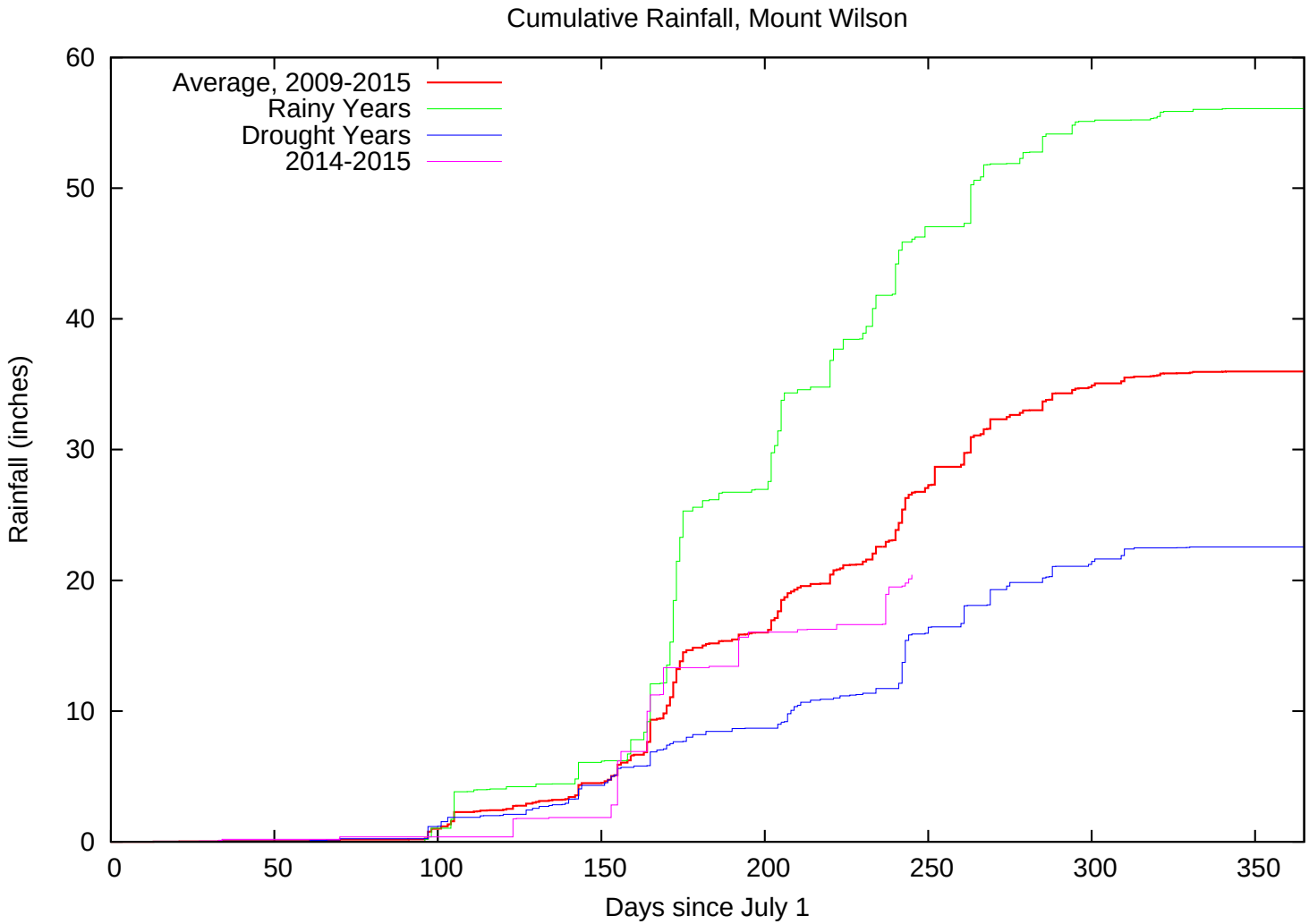
Cross-year Vital Stats

	2009	2010	2011	2012	2013	2014
Measurable Wind	19.2	23.2	30.4	35.6	27.5	11.4
High Wind†	0.2	0.7	0.5	0.3	0.3	0.2
High Humidity♮	16.5	21.6	18.2	15.6	13.5	16.0

Table: Table entries are percentages of time. Values quoted are the largest of the six bunker weather stations. † High wind is defined as being above 20 kph. ♮ High Humidity is defined as being above 90%.

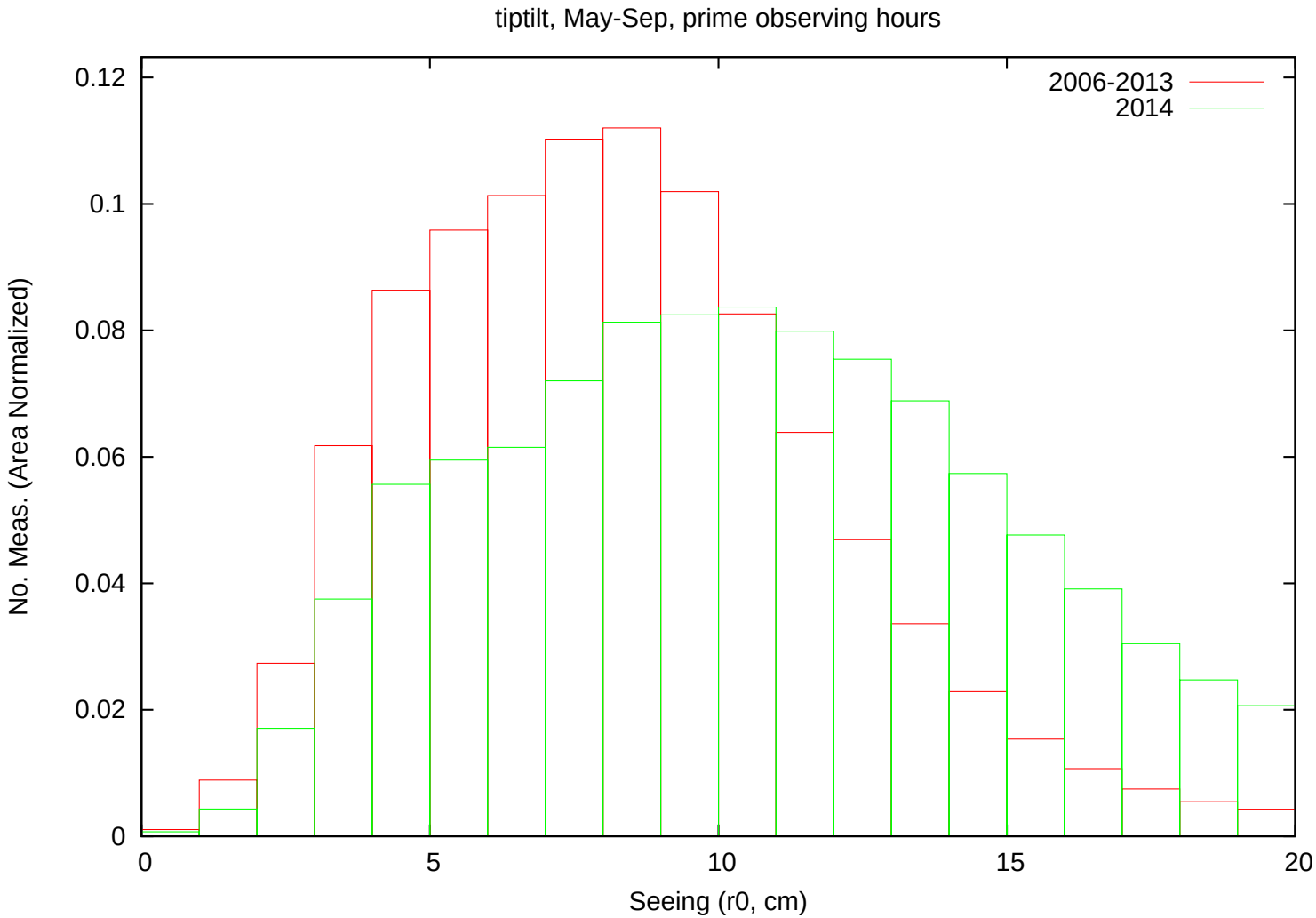
Rainfall

data courtesy of L. Webster



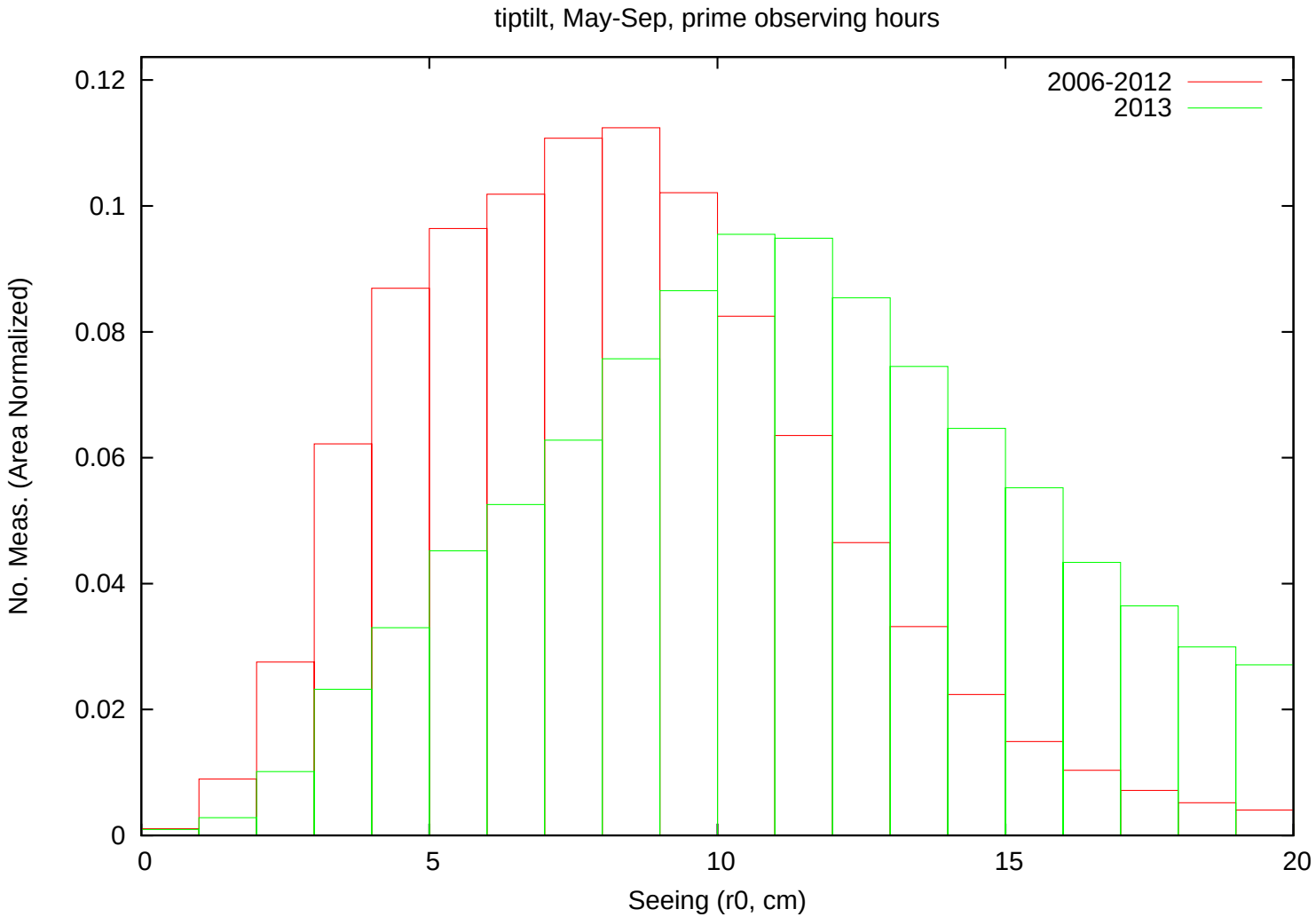
Seeing

r_0 values



Seeing

r_0 values, last year



Control System Changes

Hardware

- New computers at each telescope – actually mounted ON the telescope
  - “shoe-box” style computer (industrial node wall-mount chassis)
  - 5-position backplane and PICMG 1.3 single-board computer
  - Core i3 dual-core CPU, 2.8-ish GHz
  - Eurosys CameraLink card

Control System Changes

Hardware

- New computers at each telescope – actually mounted ON the telescope
  - “shoe-box” style computer (industrial node wall-mount chassis)
  - 5-position backplane and PICMG 1.3 single-board computer
  - Core i3 dual-core CPU, 2.8-ish GHz
  - Eurosys CameraLink card
- New computer at the western OPLE computer area rack
  - Installed to control all 6 LabAO deformable mirrors
  - Dual Xeon quad-core CPUs, 2.4 GHz, 24 GB memory

Control System Changes

Software – Current Status

- CentOS 5 with 2.6.18 kernel
  - Released April 2007
  - Full support ended March 2014
  - Bug fixes end March 2017
- Overlayed with 2.6.33RT kernel
  - New Mexico Tech/FSM Labs (Yodaiken) model
  - Intricate patches – inflexible updating
  - Not in active development

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25 : Preemptive read-copy update

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25 : Preemptive read-copy update
  - 2.6.30

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25 : Preemptive read-copy update
  - 2.6.30 : IRQ threads

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25 : Preemptive read-copy update
  - 2.6.30 : IRQ threads
  - 2.6.33

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25 : Preemptive read-copy update
  - 2.6.30 : IRQ threads
  - 2.6.33 : Raw spinlock annotation

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25 : Preemptive read-copy update
  - 2.6.30 : IRQ threads
  - 2.6.33 : Raw spinlock annotation
  - 2.6.39

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25 : Preemptive read-copy update
  - 2.6.30 : IRQ threads
  - 2.6.33 : Raw spinlock annotation
  - 2.6.39 :
    - Forced IRQ threads
    - R/W semaphore cleanup

Control System Changes

Software – Update Path

- CentOS 7 with 3.10 kernel
- Much of the “RT”-esque functionality mainlined to the x86_64 kernel since 2.6.18
  - 2.6.24 : High-resolution timer
  - 2.6.25 : Preemptive read-copy update
  - 2.6.30 : IRQ threads
  - 2.6.33 : Raw spinlock annotation
  - 2.6.39 :
    - Forced IRQ threads
    - R/W semaphore cleanup
- Should full RT be necessary, “CONFIG_PREEMPT_RT” available as a patch to a wider variety of kernels
  - `http://www.kernel.org/pub/linux/kernel/projects/rt/`

Control System Changes

Software – `git`

- Transitioning from `cvs` to `git`
- Advantages of `git` over `cvs`
  - Changes in `git` are atomic – more robust to interruption

Control System Changes

Software – `git`

- Transitioning from `cvs` to `git`
- Advantages of `git` over `cvs`
  - Changes in `git` are atomic – more robust to interruption
  - Changesets are “per file” in `cvs` while they are “whole project” under `git` – easier to revert, but no partial checkouts

Control System Changes

Software – `git`

- Transitioning from `cvs` to `git`
- Advantages of `git` over `cvs`
  - Changes in `git` are atomic – more robust to interruption
  - Changesets are “per file” in `cvs` while they are “whole project” under `git` – easier to revert, but no partial checkouts
  - More logical tagging

Control System Changes

Software – `git`

- Transitioning from `cvs` to `git`
- Advantages of `git` over `cvs`
  - Changes in `git` are atomic – more robust to interruption
  - Changesets are “per file” in `cvs` while they are “whole project” under `git` – easier to revert, but no partial checkouts
  - More logical tagging
  - Easier branching

Control System Changes

Software – `git`

- Transitioning from `cvs` to `git`
- Advantages of `git` over `cvs`
  - Changes in `git` are atomic – more robust to interruption
  - Changesets are “per file” in `cvs` while they are “whole project” under `git` – easier to revert, but no partial checkouts
  - More logical tagging
  - Easier branching
  - More fault tolerant renaming

Control System Changes

Software – `git`

- Transitioning from `cvs` to `git`
- Advantages of `git` over `cvs`
  - Changes in `git` are atomic – more robust to interruption
  - Changesets are “per file” in `cvs` while they are “whole project” under `git` – easier to revert, but no partial checkouts
  - More logical tagging
  - Easier branching
  - More fault tolerant renaming
  - Binary file support

Control System Changes

Software – `git`

- Transitioning from `cvs` to `git`
- Advantages of `git` over `cvs`
  - Changes in `git` are atomic – more robust to interruption
  - Changesets are “per file” in `cvs` while they are “whole project” under `git` – easier to revert, but no partial checkouts
  - More logical tagging
  - Easier branching
  - More fault tolerant renaming
  - Binary file support
  - etc.

Control System Changes

Software – `git`

- Transitioning from `cvs` to `git`
- Advantages of `git` over `cvs`
  - Changes in `git` are atomic – more robust to interruption
  - Changesets are “per file” in `cvs` while they are “whole project” under `git` – easier to revert, but no partial checkouts
  - More logical tagging
  - Easier branching
  - More fault tolerant renaming
  - Binary file support
  - etc.
- About 1/3 done . . . reduction pipelines and libraries