

# NRAO DiFX

Walter Brisken (NRAO)

DiFX workshop, MPIfR Bonn

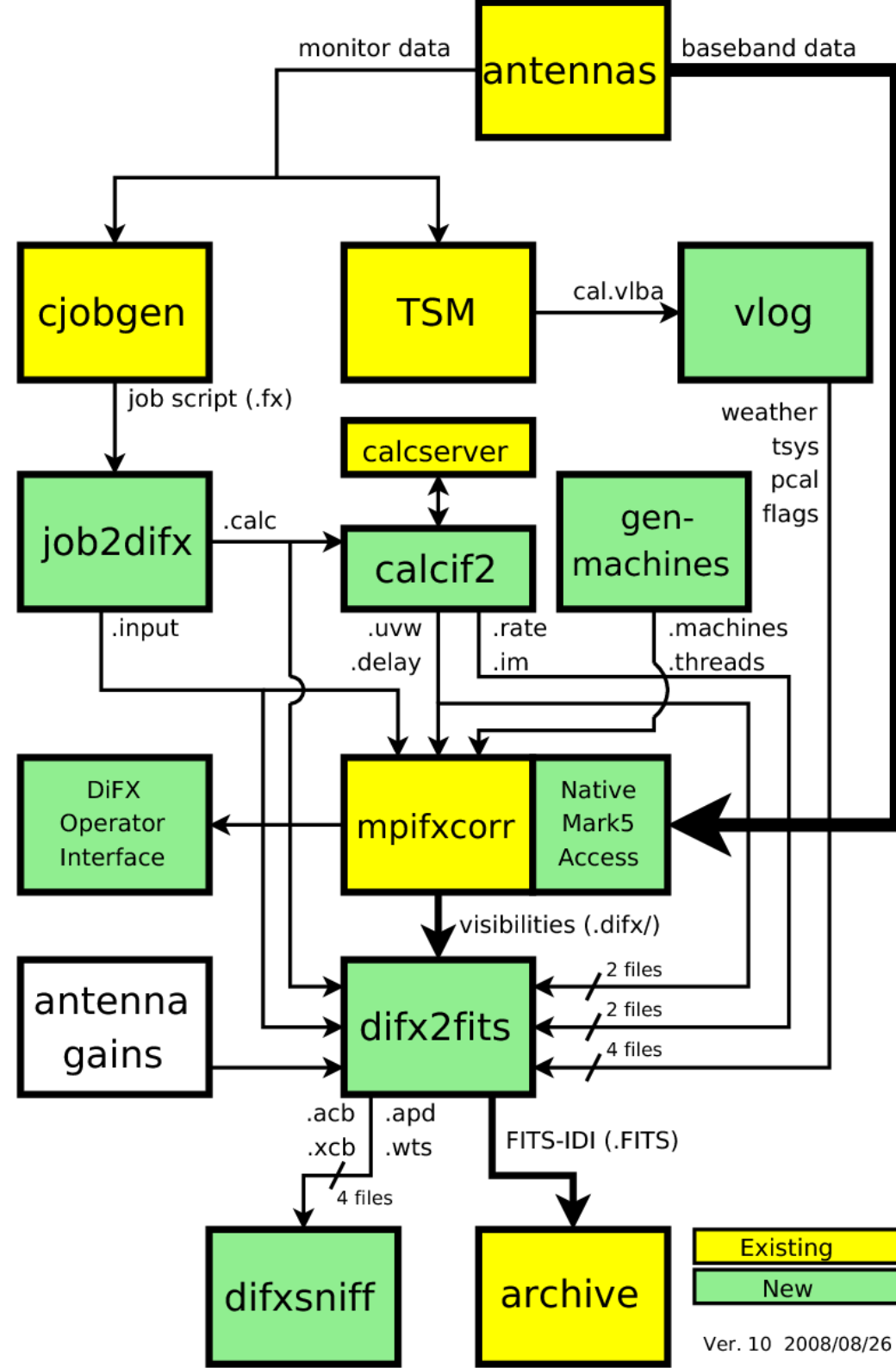
2008 Sep 9

# NRAO DiFX

- What is it?
- A DiFX distribution, like RedHat is a Linux distribution
- Several software components
  - mpifxcorr      the core of difx
  - difxmessage    an XML messaging library
  - difxio          difx .input (and other) file parser/writer
  - mark5access    library for decoding Mark5 data
  - difx2fits        program to generate FITS-IDI files
  - job2difx        VLBA job script converter and utils

# Block Diagram

- See NRAO-DiFX User Guide for more explanation



# New in version 1.1

<http://www.aoc.nrao.edu/~wbrisken/NRAO-DiFX-1.1/>

- Blanking of data replaced by headers (MarkIV format only)
- Proper data weights
- Mark5B support (incomplete)
- Support for oversampled data via decimation
- Multicast XML messaging
- Correlation of moving and near-field sources
- Improved Mark5 playback (missing disks, improved robustness)
- Polynomial-based delay model and baseline vectors
- Lots of bug fixes

# Directory tree

- Operations files in difx/
- Version specific files in their own directories
- Example tree (right) incomplete, but shows general layout

```
/home/swc/ difx root directory: a 1.5 TB partition on swc000
├ difx/ all data related to operations ends up under this directory
│   ├── archive/ staging area for data going to archive
│   ├── directories/ cached Mark5 module directories; environment variable
│   │               MARK5_DIR_PATH points here
│   │   ├── NRA0-120.dir
│   │   ├── NRA0-123.dir
│   │   └── VIPSU-05.dir
│   ├── gaincurves/ directory of symlinks to gain curve files; environment
│   │               variable GAIN_CURVE_PATH points here
│   │   └── gain.ar symlink to /home/jansky3/vlbaops/TCAL/gain.ar
│   ├── projects/ active project data; environment variable JOB_ROOT
│   │             points here
│   │   ├── bc120e/
│   │   └── mt831/
│   ├── tests/ active test data; environment variable TESTS points here
├ NRA0-DiFX-1.1/ version 1.1 files; set prefix to this during installation
│   ├── bin/ ver 1.1 programs
│   │   ├── calcif2
│   │   ├── job2difx
│   │   ├── mpifxcorr
│   │   └── startdifx
│   ├── lib/ libraries for ver 1.1
│   ├── setup_difx script to set environment for version 1.1
│   └── src/ frozen source code for ver 1.1
│       ├── calcserver/
│       ├── difx2fits/
│       ├── difxio/
│       ├── difxmessage/
│       ├── job2difx/
│       │   └── calcif/
│       ├── mark5access/
│       ├── mk5daemon/
│       ├── mpifxcorr/
│       └── openmpi-1.2.7/
├ NRA0-DiFX-trunk/ development branch
│   ├── bin/ programs
│   ├── lib/ libraries
│   ├── setup_difx script to set environment for development version
│   └── src/ source
```

# VLBA in 1994

- Mean record rate = 128 Mbps on tape
- Peak record rate = 256 Mbps
- \$ → VLBA
- \$ → VLA

# VLBA in 2004

- Mean record rate = 128 Mbps on Mark5
- Peak record rate = 512 Mbps
- \$ → VLBA
- \$ → EVLA
- \$ → ALMA

# VLBA at end of 2008

- Mean record rate = 256 Mbps on Mark5
  - Hardware correlator at processing limit
- Peak record rate = 512 Mbps
- \$ → VLBA
- \$ → EVLA
- \$ → ALMA



# VLBA in 2010

- Mean record rate = 512 Mbps on Mark5C
- Peak record rate = 4 Gbps
- \$ → VLBA
- \$ → EVLA
- \$ → ALMA

# VLBA Upgrade Project: DBE

- Replace BBCs, samplers, formatters with digital back end (DBE)
  - NRAO-Haystack-Berkeley-South Africa collaboration
  - 2 modes of operation
    - “DBE2” poly-phase filterbank for wide bandwidths
    - “VDBE” digital BBC for flexible tuning
  - 10GbE data output

# VLBA Upgrade Project: Mark5C

- Replace Mark5A with Mark5C
  - “Dumb” packet-based recorder up to 4 Gbps
  - Data packetized by channel
- Conduant corporation and Haystack under NRAO contract
- Hardware is “Amazon” Streamstor card with new 10GbE daughter board for input
  - Complete as of now, near end of testing phase
- Software being written by Haystack
  - Completion date: end of January 2009

# VLBA Upgrade Project: Correlator

- Software correlator
  - NRAO-DiFX
  - Current hardware
    - 22 quad-core intel XEON @ 2.5 GHz
    - 1 GbE switch
    - 20 Mark5 units
    - 10 stations at ~500 Mbps processing rate (2-bit samples)
  - Funded hardware (by mid 2009)
    - Roughly double or tripple processor power
    - Mean rate  $\geq$  1024 Mbps processing rate

# VLBA Upgrade Project: Control SW

- A large, mostly overlooked, part of project
  - 0.5 to 1.5 man-years effort
- How should new HW (DBE, Mark5C) be controlled?
  - Add code to ancient, fragile VME station computers?
    - No long term benefit; painful, but minimal, SW effort
  - Control new HW with new SW and old with old
    - Scalable approach, but must reinvent M&C code
  - Use EVLA control code
    - Most effective, but requires EVLA cooperation; this is the most likely option

# Side issue: SATA Mark5 Modules

- SATA hard disks nearly ubiquitous now
- SATA Mark5 modules Rev 1 available in 2007
  - Use SATA to PATA converter chip
    - No high-speed blind-mate connectors available
  - Plagued with numerous problems, HW and SW
- SATA module Rev 4/5 available mid 2008
  - HW problems fixed
  - SW problems remained, but thought to be fixed now
- NRAO to get 36x 6TB modules in next 60 days.
  - ~50% capacity increase