

IRAF: Lessons for Project Longevity

Michael Fitzpatrick

NOAO

Introduction

- * IRAF has been around for 25 years (30 if you include the design phase)
- * Designed to be highly portable
 - * More than 22 *platform* ports over its lifetime
- * Designed to be user-extensible
 - * More than 120 external packages, countless user scripts
 - * More external code written than is in the core system
- * Still highly productive for scientists
- * No obvious replacement *system* on the horizon

Introduction

- * How did IRAF survive this long?
- * What can we learn from its history?
- * Which of those lessons are being ignored?
- * Where do we go from here?

Everyone hates IRAF

- * *"Once. Just once. I would like IRAF to behave like it's supposed to"* -- Twitter - May 18, 2011
- * *"right... spectrum reduced. IRAF being stupid."* -- Twitter - Jun 06, 2011
- * *"Installing #IRAF makes me grumpy."* -- Twitter – Oct 31, 2011
- * *"Also after the last two days I'm tempted to write an IRAF manual for real astronomers that starts with 'Find some virgins for sacrificing' "* -- Twitter – Aug 7, 2011

But everyone still uses it

- * More than 1700 registered users at the help forum
- * Estimate of 5000 "regular" users not unreasonable

But everyone still uses it

- * More than 1700 registered users at the help forum
 - * Estimate of 5000 "regular" users not unreasonable
- * More than 6,070 mentions of IRAF in *astro-ph* since 2001

But everyone still uses it

- * More than 1700 registered users at the help forum
 - * Estimate of 5000 "regular" users not unreasonable
- * More than 6,070 mentions of IRAF in *astro-ph* since 2001
- * More than 17,283 mentions of IRAF in ADS refereed papers

But everyone still uses it

- * More than 1700 registered users at the help forum
 - * Estimate of 5000 "regular" users not unreasonable
- * More than 6,070 mentions of IRAF in *astro-ph* since 2001
- * More than 17,283 mentions of IRAF in ADS refereed papers
- * By the most conservative estimate, this implies:

One paper for every day since IRAF was first released

But everyone still uses it

"The spectra were reduced, combined, and extracted using standard procedures in IRAF,"
arXiv:1106.3073

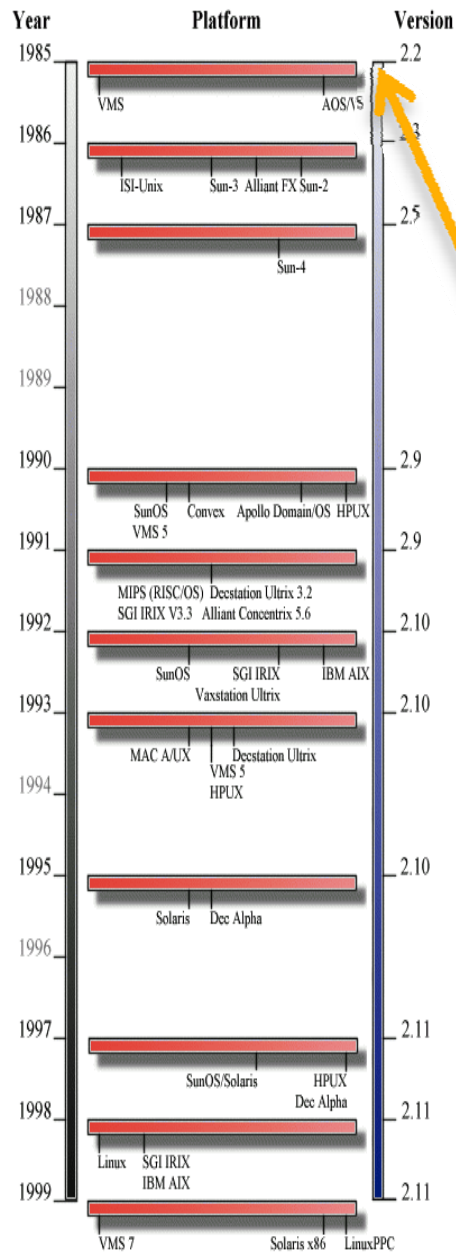
"Standard data reduction procedures were performed by using the available IRAF tasks...."
arXiv:1107.5116

Early IRAF History

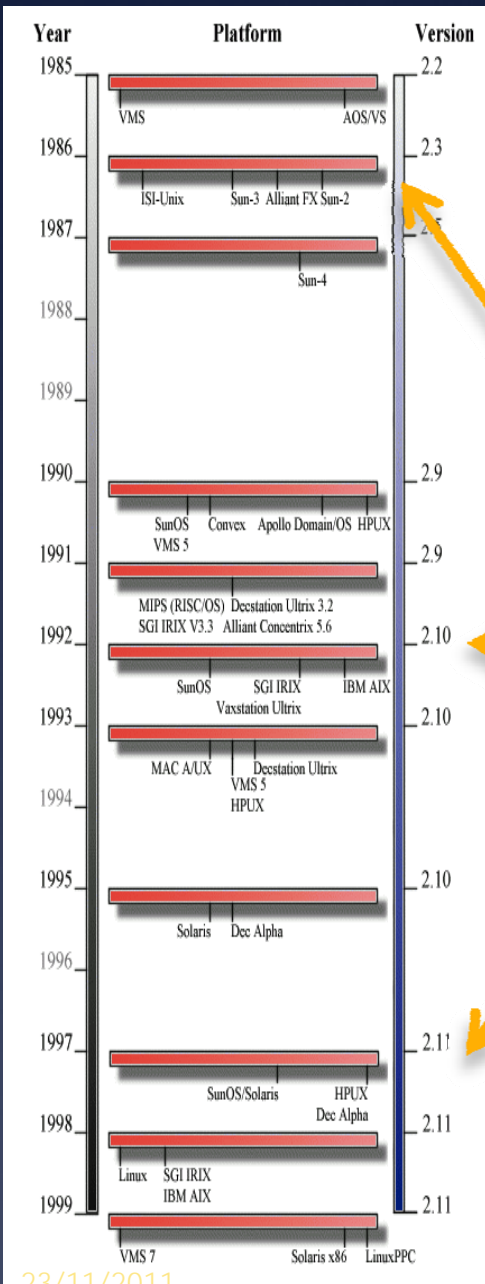
- 1981 IRAF Project begins in the Fall
- 1982 Preliminary design completed, first version of CL
- 1983 IRAF Programming Group formed
- 1984 First internal release, VMS port begun at STScI
- 1985 Limited outside beta release
- 1985 Ports to SunOS workstation and AOS/VS on DG MV/10000
- 1986 Limited public release to ~40 sites (Unix and VMS)
- 1987 General release
- 1988 GTerm and IMTOOL developed (SunView)
- 1989 Layered software support improvements
- 1990 X11IRAF Tools released

: : : :

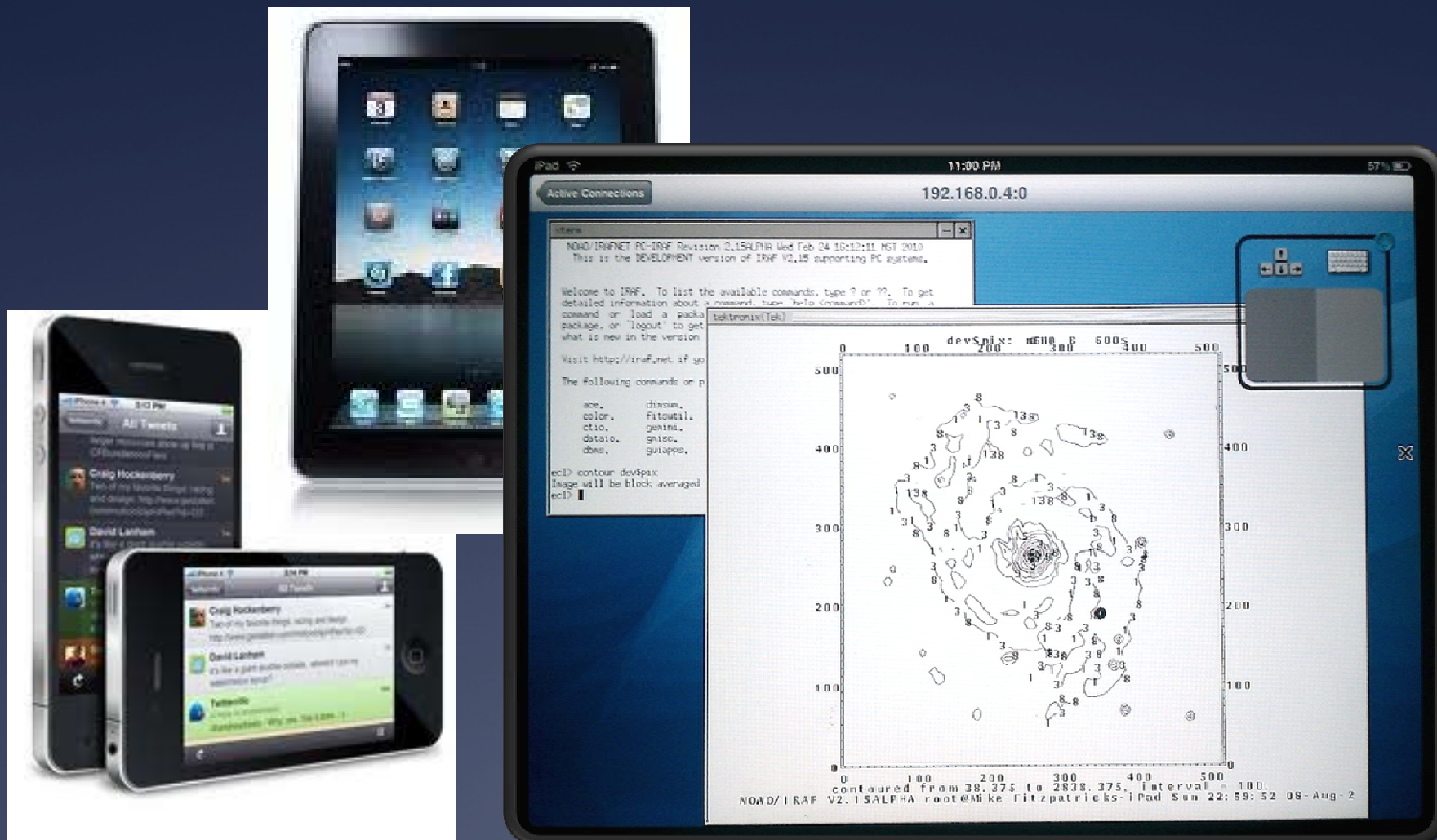
IRAF Portability



IRAF Portability



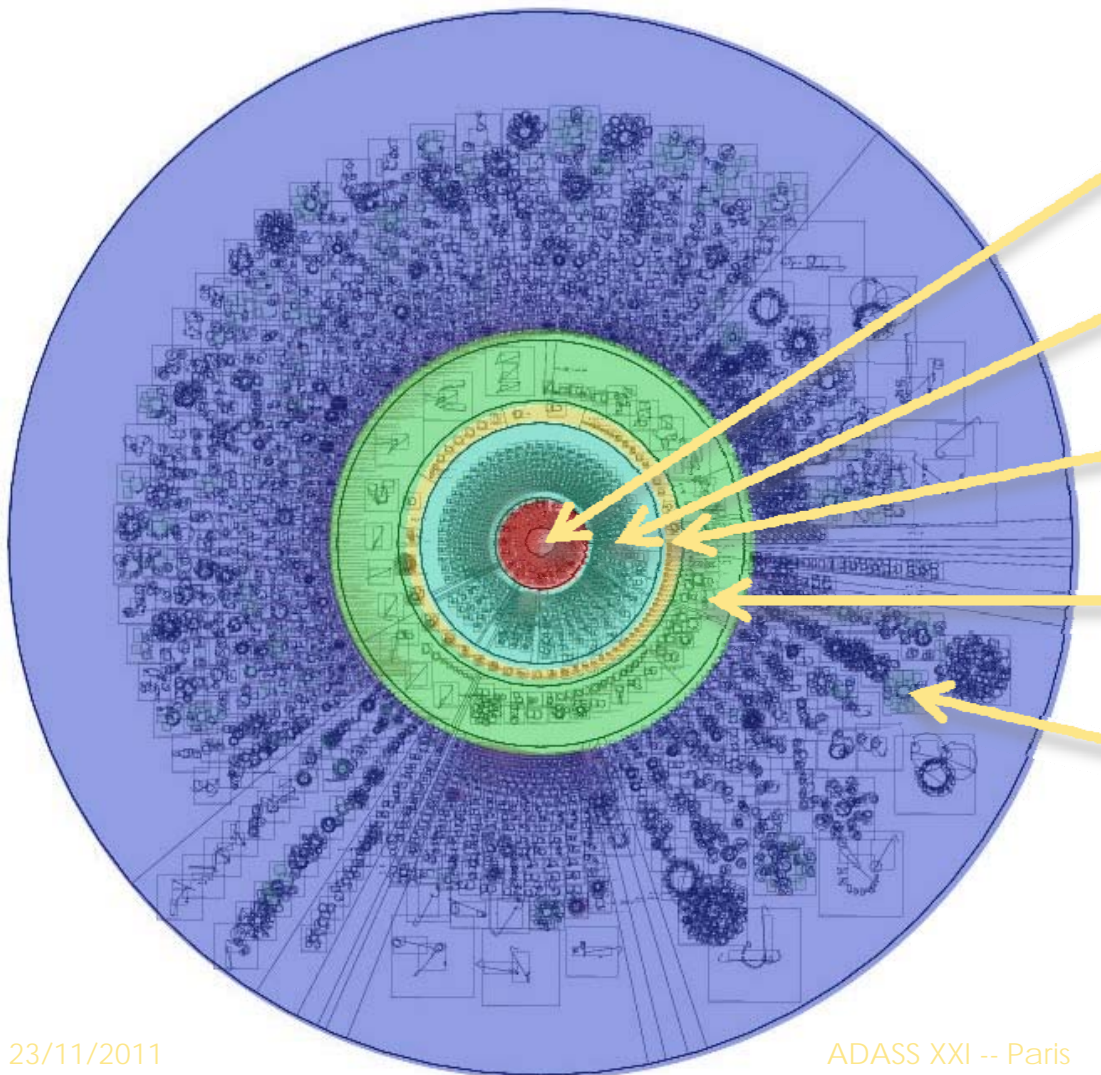
IRAF Portability



How did IRAF survive this path?

- * *Really* good, fundamental, design work
 - * Lots of up-front project cost, but paid big dividends
- * Interfaces, interfaces, interfaces ...
 - * VOS isolated all the applications from porting concerns
 - * Those who violated interfaces, paid the price
- * Low turnover rate in the IRAF Group
 - * Lots of Institutional Memory to continue best practices
- * Of course, some luck

IRAF Architecture



IRAF Kernel

VOS Interfaces

Core System Tasks

NOAO Package

External Packages

Not scaled by LOC

IRAF Architecture



Applications

IMIO, FIO, GIO, MWCS, (VOS)

IRAF Kernel (HSI)

IRAF Lessons Learned

True portability must span:

Not only:

- * Operating Systems
- * Hardware
- * Compilers (and/or other system software)

But Also:

- * Staffing levels
- * Management follies
- * Computing paradigms
 - * The invention of the Web

IRAF Lessons Learned

- * Outside projects will use your system in ways you don't anticipate
 - * Language requirements / desires
 - * Deployment as service, embedded app
 - * Interoperability, scriptable GUI, etc
- * Your user aren't who you think they are
 - * Not just end-user scientists, fellow developers
 - * Multiple levels of interfaces are required
- * Do it right the first time, even if you don't do it all

IRAF *What Ifs....*

What if, the system had

IRAF *What Ifs....*

What if, the system had

- *been written in standard C?

IRAF *What Ifs....*

What if, the system had

- *been written in standard C?
- *C bindings to the VOS interfaces?

IRAF *What Ifs....*

What if, the system had

- *been written in standard C?
- *C bindings to the VOS interfaces?
- *More than one scripting language?

IRAF *What Ifs....*

What if, the system had

- *been written in standard C?
- *C bindings to the VOS interfaces?
- *More than one scripting language?
- *a rich Aunt and we could do major new development?

Replacement Challenges

- * Attitudes have changed, so have needs
 - * Community Projects → Dedicated SW → Toolboxes
 - * Is there Institutional will to do (and fund!) this?
- * FOSS Efforts
 - * Aggregate rather than design creates a false portability regardless of language
 - * Lots of Cat Herding
 - * Are they building for the future?
 - * GPUs, detector sizes, data volumes, changes to the development environment/language itself?
- * The 99% versus the 1%

What Next?

- * No major *technical* issues seen with keeping the system running indefinitely
- * Plans/ideas for development far exceed our resources
- * Current work being driven by internal NOAO needs and collaborations, e.g.
 - * Pipeline development for new instruments (ODI, DES?)
 - * Virtual Observatory
 - * Ongoing maintenance of the system

What Next?

VO-IRAF Integration – What does it mean?

- * Support for VO Data Formats (VOTable)
- * Support for Remote data
 - * Data services
 - * General URL access
- * Application interoperability via SAMP
- * New tasks that incorporate VO capabilities
- * More to come

" IRAF was designed to protect the substantial software investment that any large data analysis system represents, and this it will do, both for the developers and for the users."

'A User's Introduction to the IRAF Command Language, v2.3

Shames & Tody, 1987

Thank You



