Maintaining FITS! Some Lessons from (and Perils of) Successful Long-term Software Maintenance

Michael Scroggins and Bernie Boscoe

Prepared for Maintainers III: Practice, Policy, and Care

October 6-9, 2019

"*Oh, that's another sign that things are in bad shape*." – A FITS maintainer on learning that the Vatican has adopted FITS as its archiving format

In this paper, we use the Flexible Image Transport Format's (FITS) forty-year maintenance history to discuss lessons that can be drawn from and unexpected perils engendered by successful long-term software maintenance. One thing is certain, for as long as there has been software there has been software maintenance. Writing from a business management perspective, Lientz, Swanson, and Tompkins observed in 1976:

> Maintenance and enhancement of application software consume a major portion of the total life cycle cost of a system. Estimates of the total systems and programming resources consumed range as high as 75-80% in each category. However, the area has been given little attention in the literature.

Little has changed in the ensuing four decades. In 1976, software maintenance was an overlooked and underappreciated problem. In 2019, it still is the same. In 1976, software maintenance soaked up a preponderance of total system programming time and costs. In 2019, it still is the same. Despite the dreams of disruptors and innovators, maintenance is a problem of eternal reoccurrence, and maintenance time is a flat circle.

Over the last forty years, FITS has been the de facto standard file format for astronomical computing. FITS was designed as an interchange format not only to mediate between non-interoperable computing systems but also to mediate between the differing imaging practices of radio and optical astronomers. Born in the FORTRAN era, through the ruggedness of its design and the care of its maintainers, FITS has weathered changes in both the technical basis of computer systems and organization of astronomical labor, transforming from an interchange format into an archival format and part-time analytical format. FITS has been maintained so successfully that it was recently adopted by the Vatican as its archival formation of choice.

Before proceeding further, a brief overview of the FITS format and its governance system is in order. The computing environment of the late 1970s was characterized by interoperability problems at every level of computation: operating systems, storage formats, and endian. Three astronomers, using the reigning computing language of the day, FORTRAN, solved the interoperability problem by defining a file format that could glide over the problems of non-interoperable operating systems, storage media, and endian. They did so by leveraging the ubiquity and simplicity of FORTRAN and the contextualization inherent to natural language. Technically, a FITS file consists of tabular binary data prefaced by a human readable and modifiable header. Useful to a fault, FITS quickly became the format of choice for observatories and individual astronomers alike. This meant that observatories, individual astronomers, and research groups could use the header system to tailor FITS to their own needs. The result was a de facto digital standard with intentionally unstandardized headers. This combination of tight digital specifications and loose header specifications that had been a strength in the computing environment of the late 1970s has since become a weakness, as computation has become more sophisticated and the organization of astronomical labor changed.

Throughout its life, FITS has been governed through two international academic organizations, the first established early in the 20th century and the other towards the dawn of the 21st century. The older organization is the International Astronomical Union's Committee, where FITS is governed through a working group. Suggestions for features are taken at the local level, percolate up through the local to national committees, and are taken up or rejected at the international level. The other organization is the Astronomical Data Analysis Software and Systems conference, where FITS is overseen by a Birds of a Feather group. Yet today, the work of maintaining FITS is usually performed on an ad hoc basis by a precariously employed younger generation who increasingly use tools such as GitHub and Google messenger, rather than academic committees, to coordinate software creation and maintenance.

Some Lessons in Software Maintenance from FITS

Despite the unsettled future of FITS maintenance, we should not lose sight of the last forty years. We have extracted a few brief lessons that we think are widely applicable to maintenance problems in general and software maintenance in particular.

Lesson 1: Stable working conditions and stable institutions are necessary, but not sufficient, for successful long-term maintenance.

In the forty years since FITS was introduced, the organization of astronomical labor has changed considerably. A broad discussion is outside the scope of this paper, but it will suffice to say that the general trend has been from stable employment to precarious employment. The reorganization of astronomical labor has brought a sea change in how astronomical software is built and maintained. Maintaining FITS through the same academic organizations that maintained it in the early 1980s is a logistical challenge in a world where even critical software is routinely developed and maintained collaboratively through distributed collaborative tools like GitHub. On the other hand, keeping scientific (or any critical) software in the hands of a few maintainers with deep expertise avoids the compatibility (or security) problems that plague many open source projects.

Lesson 2: There must also be agreement on what is important to maintain and a mechanism for reaching that agreement.

There is a small pool of FITS maintainers who are active in the two academic conference previously mentioned. We have interviewed numerous FITS maintainers and, in the interest of maintaining the confidentiality of those within this small group, we will just say that the majority of long-term FITS maintainers are scientists with PhDs who have forged permanent careers within astronomy off the tenure track. The steady cadence of academic conferences serve as regular touchpoints for formal and informal deliberations over common problems. As the conditions of academic labor have changed over the last decades, a younger generation existing precariously from postdoc to postdoc without the stability of a tenured job or a long-term research appointment are finding it difficult to access that mechanism for making agreement.

Lesson 3: Disciplinary (or professional) concerns must lead technical concerns.

Disciplinary disputes – in this case, software engineers versus astronomers – raise the question of whose norms of work and organization lead today and into the future. Who is FITS designed for? Astronomers or computer scientists? As well, generational inflection points must be attended to early and often. Slogans such as "Once FITS, Always FITS" are not enough to communicate how and why FITS came to be integral to astronomical computing. Each new generation of astronomers must receive and education into the conditions that led to FITS's creation. This is part of astronomical common sense and must be transmitted vertically from generation to generation. Software is not exempt from the historical circumstances of its creation.

Some Perils Revealed by FITS Maintenance

Much of the maintenance literature has positioned itself as a counterweight to popular (and romantic) accounts of invention or innovation, pointing out that much of what passes for invention and innovation stems from the everyday work of maintenance and repair. The history of FITS bears this out. From the simple interchange format defined in a 1981 paper, FITS has been pressed into service through innovative uses of its flexible header system as an archival and analytic format.

Peril 1: Maintainers can innovate in new directions that are themselves unmaintainable.

The stability and ubiquity of FITS, artifacts of its well-organized maintenance efforts, have led to the maintainers of FITS innovating new uses for the format. But these uses have led to unexpected consequences. The flexible header system, a strong point when used for interchange, causes unexpected problems when used for archiving. While the Vatican might be able to dictate a small number of keywords for archival use, it is impossible to impose this level of standardization on individual astronomers and small astronomy groups. Likewise, attempts to standardize a world coordinate system within FITS have stretched into their second decade, and in the interim, dozens of incompatible hacked and improvised coordinate systems have appeared, leading to maintenance nightmares when exchanging FITS files across groups.

Peril 2: Maintenance can freeze key pieces of infrastructure.

The FITS header is taken directly from the ANSI FORTRAN 1977 standard for list input and takes the form of: keyword = value/comment. The FITS header is made of card images, each card taking 80 columns (bytes), originating from systems using 80 column punch cards. The upshot is that FITS headers are stuck in the 1970s, unable to support hierarchical information. This makes instituting, let alone standardizing, a world coordinate system within FITS a herculean challenge.

Peril 3: The danger of a one-size-fits-all solution

FITS is the TXT or the CSV of astronomical formats. It started as an interchange format, was quickly pressed into service as an archival format, and finally as an analytic format. It is a default and the format of last resort when transferring images and tabular data. So while FITS unites astronomers, FITS is unusable or insufficient for many scenarios. Despite its image in popular culture, astronomy is a surprisingly diverse discipline with some astronomers simulating massive data sets, others analyzing images, and still others running complex statistical analysis over data combined from multiple telescopes. FITS does most astronomical tasks somewhat well, but nothing exceptionally well.

Final Thoughts

Returning to Lientz, Swanson, and Tompkins' 1976 warning about the ballooning costs and time required by software maintenance, we can add a further warning: maintenance is a problem of eternal reoccurrence on two levels: first, at the level of the technical skill and time commitment required to keep software components and dependencies in synch, and second, more onerously, at the level of maintaining the common sense of a field, a science, a domain, or a discipline. FITS is no different; today, it strains under the weight of new software components and dependencies, yet it maintains a key piece of astronomical common sense – scarce astronomical observations must be preserved and made available.

Astronomical common sense about preserving and sharing scarce observations has found purchase beyond astronomy. Over the last decade, FITS has served as inspiration for a potential new interchange format for material testing at the National Institute of Standards and as the Vatican's archival format of choice for preserving digital images of rare documents and religious relics. Less surprising is that the oldest idea in astronomical computing, FITS, is also the newest idea in astronomical computing. FITS is being remade into the Advanced Scientific Data Format, a new astronomical format that supplants the FITS header system with the markup language YAML, which allows technical elements for archiving and analysis, like hierarchical and nested headers, that FITS prohibits.

The history of FITS reminds us that we cannot ignore the older meanings of maintenance, those associated with providing the necessities of life. The problems of maintaining FITS are those of maintaining the science of astronomy under changing technological and organizational conditions and of maintaining the Vatican archives under changing climactic and financial conditions and, perhaps, of maintaining standards in material science. Maintenance time is a flat circle.