Does PDS Need to Support a Medium Term "Archive?" R. G. Deen[1], P. F. Penteado[2], F. J. Calef III[3], [1]Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr, Pasadena, CA 91109, Bob.Deen@jpl.nasa.gov, [2]same, Paulo.Penteado@jpl.nasa.gov, [3]same, Fred.J.Calef@jpl.nasa.gov.

Introduction: PDS is designed to be a long-term archive (50+ years). As such, its holdings are limited to static data.

Flight missions provide a lot of services while they are active, both to the operations and science teams. In the modern era, many of these services are implemented by servers, databases, or other interactive mechanisms. These resources can be considered a short-term "archive": they are available during the mission, and sometimes 6 months to a year after end of mission to support closeout activities.

Although some thought has been put into it (e.g. [1]), nobody has convincingly figured out how to archive active software such as servers over such a long time scale, given how the computer industry changes over time.

This "white paper" poses the question: is there a need for a medium-term "archive" in between these, in the 5-10 year timeframe?

The Problem with Software: Software tends to "rot" over time, becoming unsable as technology marches forward. The display technology we at MIPL (Multimission Image Processing Lab, at JPL) used in the 1980's for example, consisted of dedicated special purpose frame buffers made by companies such as DeAnza or IRIS, attached to VAX/VMS computers. Code written to display on those computers is useless today – the hardware no longer exists, and the software was not standardized. 9-track tape drives were commonly used, which are virtually impossible to find today. The Internet was in its infancy, with DECnet, BITnet, and other networks competing for dominance. Nothing written to those standards would be usable today.

So it is simply not credible to imagine that software or active servers in use today would still be usable in the PDS' 50+ year timeframe. Even given today's virtual machines, it has yet to be shown that the virtual machine file formats in use today will stand the test of time and still be runnable in that timeframe.

However, it is perfectly reasonable to assume that current software can survive with little maintenance over a 5 year or even 10 year timeframe.

Why Medium Term?: For most missions, interest in their data is at its highest during and soon after the mission ends. We assert (without proof) that the majority of science results come in the 5 to 10 year time frame after a mission ends.

If one accepts this assertion, it stands to reason that there is significant benefit to be had in making mission data easy to use in the 5-10 year timeframe – exactly when it is most feasible to do so given software rot.

Of course there will always be users looking at very old data. We are not ***in any way*** proposing that the long-term archive function of PDS be diluted or de-emphasized. Long-term users of the data should be no worse off than they are now. We are simply looking to make the task easier for medium-term users.

In fact, long-term users may benefit as well. Even if the servers no longer run, the source code should still be available. Examination of it may help to understand the data or mission better. Plus, a future mission revisiting the same body may wish to resurrect the services, in which case preserving them (even in non-running form) would be critical.

What Kinds of Services?: Most modern missions have servers and software to access their data, which would be of benefit to users after the mission ends. We will discuss a few with which the authors are most familar.

Marsviewer. This is a system consisting of data servers coupled with a Java, JavaScript, or iOS client, that allow easy browsing of data from MSL, MER, Phoenix, and (soon) InSight and Mars 2020. It visualizes not only the image data but also the derived data – XYZ, range, slope, etc. [2]

PLACES. This is an active database containing MSL rover localizations – where the rover is at any point. The data is extracted to the PDS archive now, but the server provides additional functionality in terms of coordinate translation services and maintaining user-created localizations that is not available in the static data. [3]

MSLICE. This is the tool used by the science team to command the instruments – but it also has a significant visualization component, which would be useful after the mission ends. [4]

MMGIS. This GIS (Geographic Information Service) system is a series of programs, workflows, and web-based interfaces to unite mission basemaps (orbital imagery) with science products and "quicklook" instrument results in their proper geospatial context. [5]

W10N (Webification). This is a general service that provides Web-enabled access to resources. It is used as a foundation for the Marsviewer services as well as other uses. [6]

JMars. This is a significant GIS system deployed by Arizona State University. It may have achieved enough success to be self-sustaining at this point. But if ASU decides to no longer support it, it could be picked up by PDS under this medium-term framework. [7]

VICAR. While not a server, the image processing software developed and used by MIPL (and recently released Open Source) certainly constitutes a significant software suite. It is actually 50 years old as of last year, and has been continuously maintained (although there is no known 50-year old code left). There is no indication this support will stop any time soon. But if it were to, PDS could continue to offer it in Open Source form as long as it keeps running for users. [8]

Cassini VIMS Data Server. This server has been the primary source for science data used by the VIMS team throughout the entire mission. It provides calibrated observations on-the-fly, controlled by user parameters. The pipeline itself is a combination of several independent libraries and custom software. The VIMS operations team has been maintaining this server, but support will end in the near future as the mission ends.

Titanbrowse. A database and visualization service originally developed for Cassini VIMS observations of Titan, in order to provide queries based on the full hyperspectral dataset and arbitrary processing to generate derived products and visuaizations on demand. Software of this kind can be applied to make any hyperspectral imaging dataset accessible, not just VIMS [9].

Cassini Image DataBase (IMDB). Used by the Cassini ISS team as one of the primary references to locate and visualize Cassini ISS data.

How Would This Work?: We do not claim to have all the answers to this. However, in broad brush, when a mission ends, part of the close-out activities would be to package up relevant servers and software into a virtual machine (VM) framework that could be easily redeployed. The relevant PDS nodes would then deploy these servers and keep them running as long as they can without investing too many resources. We are not suggesting that PDS maintain them forever; that would constitute a long-term archive and be prohibitively expensive. The original authors could update the servers if they so chose. But the nodes could do little fixes here and there, and simply keep the power on – a seemingly trivial thing that is quite hard to do once the mission ends and nobody is paying the bill.

Additionally, if the code behind the servers can be open sourced, it may be adopted by the community and maintained that way. PDS could still host the servers (since it has the data resources), accepting updates from the community.

Maintenance of the servers would be simpler than during operations, for the simple reason that they are not getting any new data. The ingest mechanisms, often the most troublesome part for maintenance, are not needed. Only the user-facing parts of the services need to be maintained.

At some point, the service will become unmaintainable. Perhaps the JavaScript features the client depends on become deprecated in modern browsers, or the virtual machine format changes, or an unacceptable security hole is discovered. When that happens, the servers would simply be turned off, their medium-term job done. Users would then have to revert to the current data-only long-term archive. However, the VM images themselves, as well as the source code, could still be available for download, in case users want to try to fix it themselves.

Precedents: There are precedents for this concept. The Imaging Node has already deployed the Java Marsviewer client [2], along with the W10N server it needs. Support for the Web and iOS versions is planned. There is no expectation that these tools will be available forever, but they are very useful to the community now, which makes support worthwhile. Because Marsviewer is multimission, and its missions are still active, maintenance is not a concern at the moment. But it is certainly not likely to be maintained over a 50+ year timeframe.

In addition, this concept is applied in microcosm within VICAR [8]. When missions end, potentially useful code is simply left in place. It is built with each release but is not otherwise maintained or tested. If the code stops building (and it's not a trivial fix), it is obsoleted at that point. Although application programs are simpler than servers, the concept has proven to be useful.

References: [1] Million, C.C. (2015), 2nd Planetary Data Workshop, Abstract #7011.
[2] http://pds-imaging.jpl.nasa.gov/tools/marsviewer/
[3] http://pds-imaging.jpl.nasa.gov/data/msl/MSLPLC_1XXX/
[4] https://software.nasa.gov/software/NPO-45908-1
[5] Calef, F.J. *et al*, (2017), LPSC Abstract #2541.
[6] http://w10n.org/spec/w10n-v0.9-20131221.html
[7] https://jmars.asu.edu
[8] http://www-mipl.jpl.nasa.gov/vicar_open.html
[9] Penteado, P.F. (2016), 48th DPS, Abstract #123.43.