KDP: A DISTRIBUTED PIPELINE PROCESSING TOOL FOR KUBERNETES. Zachary M. Taylor[1], Kevin M. Grimes[1], Brad Lunsford[1], [1]Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Dr, Pasadena, CA 91109, zachary.m.taylor@jpl.nasa.gov.

Introduction: The Planetary Data Systems ("PDS") Cartography and Imaging Sciences Node's ("IMG") archives retains hundreds of terabytes of planetary imagery captured by dozens of spacecraft over as many years. Such massive quantities of data pose unique engineering challenges when undertaking tasks such as data validation, integrity checks, metadata extraction, transformation, and other operations that would otherwise be straightforward were they not required to touch hundreds of millions of files. The need to perform such tasks on vast archive holdings is the impetus for development of new analytical capabilities that are highly repeatable, reliable, and *embarrassingly parallel*.

This work is presented as KDP: *Kubernetes Does Pipelines*.[1] According to its creators, "Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation" [1]. KDP is, in short, an extension of this functionality, achieved by way of the Operator Pattern [2], which allows for custom resources defined within Kubernetes to be managed via customized plugins. KDP adds pipeline definition and execution on top of Kubernetes' industry-leading container orchestration, lifecycle management, and scalability.

Impact: KDP brings the following new processing capabilities to PDS IMG and the wider science community:

Horizontal Scalability. As KDP uses lightweight container technology for each of its processing steps, individual pipeline steps achieve theoretically infinite horizontal scalability *independent of one another*. This is a significant advantage over tools which require processing power to scale as a whole - KDP allows fine-tuning of pipeline scaling.

Embarrassingly Parallel. KDP has been designed with 'embarrassingly parallel' execution as a top-level requirement. Where common use cases requiring processing thousands of files at a time, KDP allows the work to be broken down into the smallest logical 'chunk' (oftentimes the operation to be performed on a single file in the subset), then goes as wide as possible while still guaranteeing order of execution and completeness of processing.

Deployment-Venue Agnostic. KDP can be deployed anywhere a Kubernetes cluster can be deployed, which makes it compatible with both on-premises and cloud computing environments. With most cloud providers offering managed Kubernetes clusters as a service and open-source tools such as Rancher [3] making on-premises deployments more approachable, the prerequisites for KDP can often be spun up and down with just a few clicks. KDP is also agnostic to data location, so processing can be carried out regardless of whether data resides locally, in the cloud, or any combination thereof. This is especially useful for missions such as Mars 2020 that deliver data to the cloud, where egress to the ground would be costly and slow.

Use Cases: The following is a non-exhaustive list of use cases that KDP is being used to address.

Product-Level Validation of Mars 2020 Data Deliveries. Validation of data deliveries to the PDS is a time-consuming process, as existing tools operate in a single-threaded manner, handling files one at a time. KDP has enabled a highly parallel approach to product-level validation, decreasing the time taken by validation by orders of magnitude.

Processing, Transformation, and Validation of MSAM PDART Data Set. The MSAM data set [4] contains roughly 60TB of MSL mosaics and ancillary files which must be transformed and labeled with PDS4-compliant metadata. Performing this task for over 2000 sols of data would be onerous with existing tools, and KDP has allowed for reliable execution across the entire dataset. The containerized nature of processing steps further enables rapid iteration when data providers need to make updates to the process.

Archive Metadata Extraction. It is necessary to keep up-to-date various metadata about the data holdings at PDS IMG, and KDP makes such a task possible. Crawling the entire archive holdings would be impossible without distributed processing, as there are hundreds of millions of files to process. KDP enables archivists to extract metadata in a distributed manner much faster than previously possible with sequential processes, which would often take weeks to complete.

Conclusion: KDP brings new analytical capabilities to the PDS IMG that are horizontally scalable, embarrassingly parallel, deployment-venue agnostic, and collectively backed by the reliability of the Kubernetes platform. Such capabilities will enrich the archive and its computational capabilities, broadening the scope of analyses which can be performed on PDS IMG's vast holdings.

Availability: KDP is planned for open source release to the general public in the current fiscal year.

Acknowledgments: This development was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the

[1] Or, more fun: *KDP Does Pipelines*.

National Aeronautics and Space Administration.

The author would especially like to thank coworkers Kevin M Grimes and Brad Lunsford for their contributions to this work. Their continued support, high standards, and subject-matter expertise helped make this work what it is today.

References:

[1] The Kubernetes Authors. (2021). *What is Kubernetes?* https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/ [2] The Kubernetes Authors. (2021). *Operator pattern*. https://kubernetes.io/docs/concepts/extend-kubernetes/operator/

[3] Rancher Labs, Inc. (2021). Rancher. https://github.com/rancher/rancher

[4] R. G. Deen et al. (2018). Mastcam Stereo Analysis and Mosaics (MSAM). In *49th Lunar and Planetary Science Conference*.