Key Provenance of Earth Science Observational Data Products

Mehmet Aktas[2], Beth Plale[1, 2], Helen Conover[3], Prajakta Purohit[1, 2]
[1]School of Informatics and Computing, Indiana University Bloomington, Indiana, USA
[2]Data to Insight Center, Indiana University Bloomington, Indiana, USA
[3]Information Technology and Systems Center, University of Alabama in Huntsville, USA

Abstract

As local arrangements for sharing data break down due to the sheer volume of data that is becoming available, these local arrangements need to be replaced with reliable records about the what, who, how, and where of a data set or collection. This is frequently called the provenance of a data set. While observational data processing systems in the earth sciences have a long history of capturing metadata about the processing pipeline, this metadata is often difficult to access and interpret. In this paper we discuss the issues and an approach to capture and representation methodologies for the provenance of a satellite imagery processing pipeline in an effort funded by NASA.

Keywords:
Provenance Capture, Provenance Management, Digital Data Preservation, Earth Sciences

1. Introduction

As the sheer volume of data increases, particularly evidenced in the earth and environmental sciences, communities are growing increasingly aware of the breakdown of local approaches to data sharing where data is exchanged between groups and researchers who know one another. As a result there is growing interest around ways in which data sets can be more richly described, and descriptions can travel with the data. The setting in which data is generated is often highly distributed, requiring large amounts of observational data and computational resources are needed to run sophisticated numerical models, and mining and visualization can be carried out over Petabytes of data. The data products produced in the course of research are digital, and are of lasting value in advancing the scholarly research and in addressing pressing societal problems.

For data collections to have lasting value, provenance capture and management has become an acknowledged component of enhancing the long-term preservation of digital data.

Data provenance is the lineage of a data product or collection of data [1] and applies to observational and imagery data arriving in real time from sensors, networks of sensors, and instruments; results from computational models and data mining; field studies such as documenting human use of a plot of land over time; regional positional data; scholarly reports in journals, etc. Provenance can identify event causality; enable broader forms of sharing, reuse, and long-term preservation of scientific data; can be used to attribute ownership and determine the quality of a particular data set [2].

What role does provenance collection play in the earth sciences? Provenance captures how a particular scientific collection was created, when, and by whom. That is, the “what”, “where”, “how”, and “who” of a data product or collection. It reflects the transformations that a data collection underwent prior to its current form and the sequence of tasks that were executed and data products applied to generate a new collection.

The larger context in which our research is carried out is enhancing the preservation record for the Earth sciences. We discuss provenance capture in the context of Earth science applications, in particular the Advanced Microwave Scanning Radiometer - Earth Observing System (AMSR-E) Sea Ice Processing application that we are working with. The NASA-funded Instant Karma project is used to introduce a model for thinking about provenance instrumentation. We discuss the model by applying to data processing workflows with the use of Karma provenance collection framework.

The remainder of the paper is organized as follows. Section 2 gives the motivation behind this research. Section 3 discusses the NASA data products, the application in which capture is applied, and limitations that motivated the work. Section 4 gives a brief overview of related work, while Section 5 introduces the Karma Provenance Collection Tool.

Section 6 explains the Instant Karma Project. Section 7 introduces a model for thinking about mechanisms for collection and representation of the provenance data that is driven by instrumenting the sea ice data processing application. Finally, Section 8 concludes the paper with ongoing and future work.

2. Motivation

NASA has been collecting, storing, archiving and distributing vast amounts of data collected from sensing satellites for several decades now. The raw data collected from the different sensors undergoes many different transformations before it is distributed to the science community as climate research quality data products. The science community then uses these data products to address important science questions, and their analyses can influence major policy decisions. The transformations that these data undergo range from simple to complex, from preprocessing that includes calibration, reprojection, subsetting, etc., to actual conversion of the instrument counts into meaningful scientific parameters. These transformations are based on scientific algorithms and may also utilize ancillary data. The different data processing centers use well-engineered processing procedures to handle these transformations, but inevitably variability is introduced. There may be changes made to the algorithms, different ancillary datasets may be used by these algorithms, underlying hardware and software may get upgraded, etc. These variabilities impact the data product and thereby influence the scientific analyses. These changes need to be captured, documented and made accessible to the scientific community so they can be properly accounted for in analyses.

The current procedures for capturing and disseminating provenance, or data product lineage, are limited in both what is captured and how it is disseminated to the science community. These limitations have been recognized by organizations like NASA who recently stated that it is *"imperative that users have substantial information about product quality, usability, and legacy of inputs and processes to these data.*" It seeks better ways for users to interpret data product metadata as well as methods for adding value and exposing users to provenance information. Specifically, we target the Advanced Microwave Scanning Radiometer - Earth Observing System (AMSR-E), a passive microwave radiometer aboard the Aqua satellite that generates data products and key data sets for research in climate variability.

3. Advanced Microwave Scanning Radiometer - Earth Observing System (AMSR-E)

The AMSR-E instrument is a Japanese designed instrument flying aboard NASA's Aqua satellite. It provides measurements of terrestrial, oceanic, and atmospheric parameters for the investigation of global water and energy cycles, including precipitation rate, sea surface temperature, sea ice concentration, snow water equivalent, soil moisture, surface wetness, wind speed, atmospheric cloud water, and water vapor.

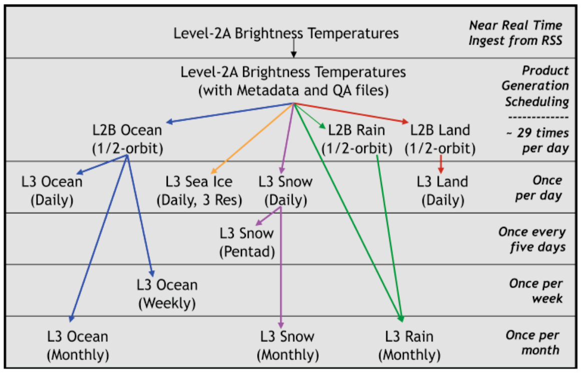


Figure 1 NASA AMSR-E Data Products

NASA has funded the AMSR-E Science Investigator-led Processing System (SIPS) to generate standard data products using science algorithms developed by the AMSR-E science team. As depicted in Figure 1, the Brightness Temperature data products are used to generate Level-2 (Ocean, Rain, and Land products) and Level-3 (daily, pentad, weekly and/or monthly Ocean, Snow, Sea Ice, Rain and Land) products before they are transferred to the National Snow and Ice Data Center for distribution to the science teams and to the public.

3.1 Framework

The AMSR - E Sea Ice processing framework is based on a flexible software architecture design that accommodates generating any number of the science products at any one time, as well as multiple instances of the same product (as in the case of reprocessing). Each individual processing operation is atomized so that it may be invoked independently or by layers of wrapper scripts [3]. The lowest - level software scripts contain one science algorithm or other atomic operation (such as browse image generation) and operate on either one file or a list of files.

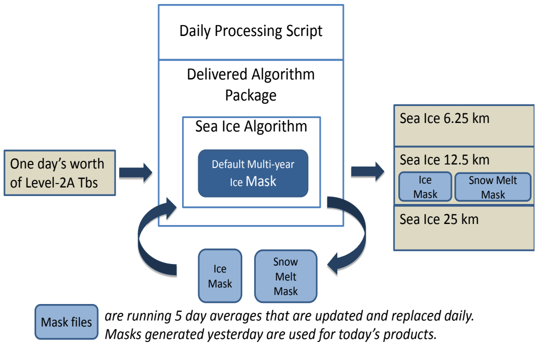

Daily Processing Script
Delivered Algorithm Package
Sea Ice Algorithm
Default Multi-year Ice Mask
One day's worth of Level-2A Tbs
Sea Ice 6.25 km
Sea Ice 12.5 km
Ice Mask
Snow Melt Mask
Sea Ice 25 km
Ice Mask
Snow Melt Mask
Mask files
are running 5 day averages that are updated and replaced daily.
Masks generated yesterday are used for today's products.

Figure 2 AMSR-E Sea Ice Processing Workflow and Dependencies

Figure 2 depicts the AMSR-E Sea Ice processing workflow and shows how the sea ice science algorithm is encapsulated with higher level scripts that operate on a list of input files. Groups of low‐level scripts are combined into workflows that orchestrate the actions required to generate each daily, weekly, and monthly product (and all the files associated with the product, such as metadata, quality assurance, production history, browse images, and science data subsets). Higher‐level workflows combine calls to lower‐level workflows to produce, say, all the daily products (or any combination of one, two, or three) in one execution. Figure 3 illustrates an AMSR-E high resolution sea ice concentration product, which is widely used for development of new and improved sea ice concentrations (e.g., higher resolution products).

Over time AMSR-E sea ice algorithms improve as a result of new algorithm research and algorithm validation studies. The ingest flow line will receive the new algorithm, and the information about the updated algorithm needs to be disseminated to users. For Earth sciences and climate studies in particular, monitoring temporal changes and variabilities, in our case from satellites, provides some of the most important pieces of information. Therefore, it is absolutely essential that users of these data are made aware of algorithm changes, whether errors are corrected, whether algorithm coefficients were updated or other functional improvements were applied.

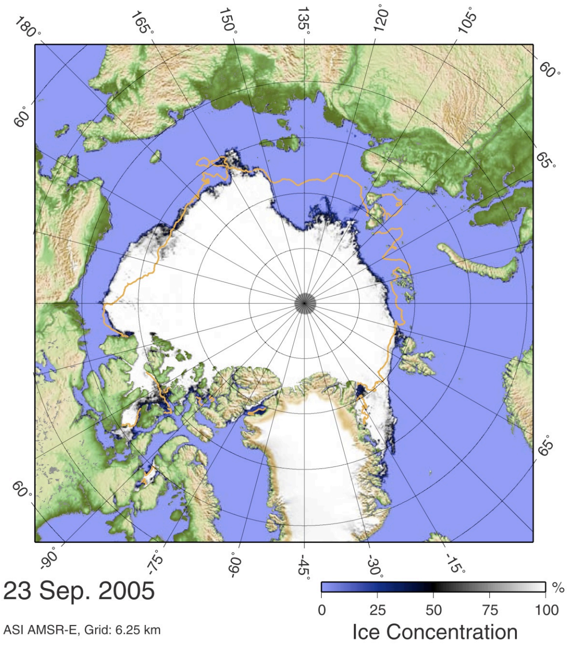


Figure 3 AMSR_E Sea Ice Data Product

To this end, a system is needed to document quantitatively through plots or maps the impact of each data product change on the data quality. Because of the importance of AMSR-E sea ice data products to the different climate studies, it is imperative that the process of provenance information collection and dissemination is improved. Hence, in this study, we mainly focus on improving the collection, preservation, utility and dissemination of provenance information for the AMSR-E sea ice data production workflows.

3.2 Limitations of Original Metadata Capture

The AMSR-E data processing workflows have existed for some time. When viewed from the perspective of provenance that we have today, the system that built in support for some provenance-like information recording is limited in the following ways:

- *Full lineage information is not collected.* There is no direct traceability in the inventory metadata back to brightness temperature data (Level 2) for the Ocean, Land or Snow products (Level 3) because the metadata only lists the input pointer to these products.
- *Sometimes complete input information is not collected.* An example is the Rain product - Level 3 - that uses as input ~900 Level 2 objects and ~900 other Level 2 objects. Due

to a limitation in the length of the input pointer field in the metadata, only the first 900 are listed. It is up to the user to determine which version of second 900 was used as input to this product.

- *Change notification is based on a "Push" model.* There is no simple, automated mechanism for the data user to request information on previous or more recent versions of a given product.
- *Provenance information is embedded in the data.* Existing provenance-like information is captured in the inventory metadata and stored in the files. The drawback with this approach is that once the data is removed (e.g., replaced by a newer version), so is the provenance information. A scientist researching a paper citing a specific dataset will not find the data if it is no longer available.
- *Some provenance and quality information is not available to users.* Production history and QA files are generated but not easily available to data users. While production history information is embedded in the metadata stored within the file, the QA files may contain information not available elsewhere.
- *Comparison of two versions of data products is cumbersome.* To compare two different versions of a data product, a data user must first find the release notes for each version and then use a tool like HDF-View to analyze the limited provenance metadata stored in the files.
- *Comparison of more than two versions of data product and generating patterns that describes and distinguishes the general properties of data production process is not possible.* To make use of large number of different versions of a data product is necessary for tasks such as performing inference to make predictions on new data production process, detecting faulty production processes and describing the general properties of production process. There is no mechanism to perform these tasks.

To address these limitations for provenance collection, we applied a proven provenance tool to NASA's AMSR-E Sea Ice Data Production workflows.

4. Related Work

Provenance systems for scientific workflows can be categorized into two main categories: a) incorporated into scientific workflow systems, and b) standalone provenance systems. Provenance systems coupled to scientific workflows can easily record provenance during workflow execution. Examples of such systems include Taverna [4] and VisTrails [5]. These systems have commonalities, that is, they are tightly coupled with a particular workflow system and designed only for managed workflows. Provenance systems designed as standalone systems can capture provenance not only from the managed workflows but also from the unmanaged workflows where there is no end-to-end control over the execution. Examples of such systems include the Karma [6] and Provenance Aware Service Oriented Architecture (PASOA) [7] systems. A disadvantage of PASOA system is that it can only capture provenance from the workflows in which all components are web services. In this study, we use Karma to harvest provenance from AMSR-E Sea Ice Production Workflows, since Karma has broader suite of tools, which support different workflow settings.

5. Karma Provenance Capture Tool

Karma is a provenance capture and representation tool designed and developed for data driven computing such as the pipeline that generates NASA standard products. Karma records uniform and usable provenance metadata independent of the processing system while minimizing both the modification burden on the processing system and the overall performance overhead. Karma collects both process and data provenance. Process provenance contains information about workflow execution and associated algorithm invocations. Data provenance captures metadata about the derivation history of the data product, including algorithms used and input data sources transformed to generate it.

Five key features distinguish the system from many other provenance systems. First, Karma is a standalone general collection system, uncoupled from a specific workflow system or workflow model, such as occurs in Taverna, VisTrails. Therefore, it can capture provenance from different workflow systems, and has done so from systems that implement user-driven workflows, orchestration engine workflows, or, in ongoing work, a mix of the two. Secondly, Karma provenance collection is implemented through modular instrumentation, which makes it useable in diverse workflow architectures composed of, for instance, Axis2 web services, Java classes, and message bus listeners. Thirdly, Karma captures provenance by accumulating discrete runtime activities during the life cycle of workflows, thus, it can capture provenance in streaming workflows whose structure is not defined before execution. Fourthly, Karma stores provenance data using a two-layer information model which includes both execution details for utilizing the data and registry information for long term preservation [6]. Finally, Karma implements Open Provenance Model (OPM), which is a good model for core provenance entities and relationships [8]. Emerging from the e-Science provenance community, OPM is evolving as a standardized representation of provenance. The aforementioned requirements of provenance collection in AMSR-E data production streams are satisfied to a large extent by a provenance point of view such as is embodied in the OPM v1.1 that captures data and/or control flow of the ingest process. Extensive discussion on Karma tool can be found in [9].

6. Instant Karma: Application of Karma to NASA's Data Production Stream

The NASA-funded Instant Karma project aims to instrument the existing AMSR-E data production workflows with the Karma tool to capture all the provenance information and to disseminate this information to the AMSR-E data user community using a web based Provenance Browser as illustrated in Figure 4. In this project, the initial focus is on incorporating Karma into the AMSR-E data production system related to Sea Ice products. To this end, the project aims to modify and configure Karma to capture the requisite provenance and quality metadata. The project fully leverages existing lineage information that is currently being captured about process runs and embedded in the data products (e.g., input pointers) or provided in separate files (e.g., data quality information in the QA files). Such information is augmented with additional data and process provenance to better capture historical traces and connect together the existing disparate sources of provenance information.

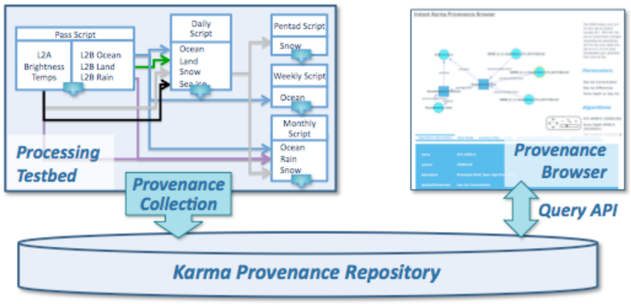


Figure 4 Karma Provenance Collection Tool applied to AMSR-E SIPS domain

Karma captures and stores provenance information in a database. This information is exposed to the AMSR-E data user community via the web. Each AMSR-E data product generated is referenced using a unique ID, which can be used to retrieve all historical information about the data product, including for instance, software release notes about the algorithm used to generate the product and a complete list of input files. AMSR-E data users are able to browse, query and visualize the provenance for a particular data file.

7. Provenance Capture in Production Workflow

7.1 Capture

A key challenge in provenance collection is the method used to collect provenance information. Plale et al [9] discuss different mechanisms for collecting provenance where collection mechanisms fall into one of three categories: user annotation, scavenging, or full provenance instrumentation. Based on this analysis, provenance collection through *"User annotation"* is a human data entry activity where users enter textual annotations. This approach imposes a low burden on the application, but a high burden on the humans responsible for annotation. Provenance collection through "full *instrumentation"* requires implementing an instrumentation library or routines that are then inserted into an application to collect provenance. This approach allows for far better completeness and consistency of the resulting provenance, but can impose fairly substantial performance overhead on the application and on the programmer who must add calls invoking the provenance library to his/her application. Provenance collection through "full *scavenging"* requires piggybacking onto existing collection mechanisms, such as a logging tool or an auditing tool. This approach minimizes the burden on the application and user; however it has a disadvantage of resulting in incomplete information. However, it represents the most promising form of provenance collection.

In AMSR-E data production framework, we use combination of *"full provenance instrumentation"* and *"scavenging"* approaches to achieve complete provenance while minimizing the performance overhead.

As for the *"full provenance instrumentation"* approach, we first instrumented the high level wrapper scripts (Daily, L3, L3-SeaIce and Santa) written in Perl. To reduce the performance overhead on the application, we chose after-the-fact provenance collection as opposed to real-time provenance collection. The after-the-fact collection requires a logging tool to capture provenance in log files, while the real-time collection mandates direct interaction with a third-party provenance repository during execution. As the time required for capturing the provenance in log files was less than the time needed to interact with a provenance tool, we used a Perl log tool to dump into a log file, which is then harvested for provenance.

As for the *"scavenging"* approach, we implemented an adapter that harvests the log files for provenance. The adapter code is written in Java and invoked from the high level wrapper Perl script after the computing task completed.

The adaptor provides an interface that uses the log files to derive provenance information and maps them into the Karma Provenance Repository. It consists of two main parts: Log Parser and Notification Generator. The Log Parser is used to process log files to extract provenance information, while the Notification Generator is used for generating and sending provenance notifications to Karma Provenance Repository.

In AMSR-E data production workflows, provenance is collected from multiple sources to obtain a picture of experiments and conditions of the experiment. These sources include configuration files, data files, mask files and so forth. Our adaptor software implements XML parsers that process several configuration files (PerGranule.xml, PerProduct.xml) to harvest provenance information and ingest to Karma repository.

Another important source for provenance information is the compressed data files (HDF files), which is used to contain input and output data products. Currently, we are working on processing metadata portion of the compressed data files to harvest more provenance information that may be valuable for the researcher.

7.2 Provenance Capture

To do provenance harvesting in AMSR-E Data Production Workflows, we use the Karma provenance collection and management system, developed at Indiana University [6], which captures provenance data in scientific workflows through modular instrumentation, and stores the data in its own database for further utilization and long term preservation.

In this scenario, Karma collects provenance from AMSR-E Sea ice processing *application*, in which code for collecting provenance information is inserted into the desired monitoring points.

```
dataConsumed,seaice-
20110621185112,level1,http://d2i.org/amsreprovenance/i
u/daily,1,daily,20110621185112,Data
Consumed,AMSR_E_L2A_BrightnessTemperatures_V1
1_201106170048_D.hdf,Consumed,AMSR_E_L2A_Bri
ghtnessTemperatures_V11_201106170048_D.hdf==545
70123#::none::none

serviceInvoked,seaice-
20110621185112,level1,File,1,/amsr/karmadev/products/
bin/seaice/@L3seaice,20110621185112,Service
Invoked,/amsr/karmadev/products/bin/seaice/@L3seaice,
Invoked,none::none::none

dataProduced,seaice-
20110621185112,level1,http://d2i.org/amsreprovenance/i
u/daily,1,daily,20110621185826,Data
Produced,AMSR_E_L3_SeaIce6km_V13_20110617.hdf
,Produced,AMSR_E_L3_SeaIce6km_V13_20110617.hd
f==46472342#::none::none
```

Figure 5 Raw provenance data captured in experiment log files

During workflow execution, the instrumentation dumps the provenance activities into a log file. Figure 5 illustrates raw provenance data, capturing three different types of provenance events, in a log file.

With the adaptor approach, provenance is collected after-the-fact when provenance data is gathered in a log file after the compute task is completed. At the end of all the sequence of activities completed, the adapter is invoked by the workflow script and processes the log files to publish provenance activities as *notifications* (in XML).

Notifications travel from the adapter to the remote Karma provenance service. This is done through a synchronous or asynchronous send/receive message protocol by utilizing web service programming model. When Karma receives a notification, the corresponding handler retrieves the provenance data and stores or updates these data in the provenance database.

Karma system utilizes the two-layer information model introduced in [6]. Provenance data stored in a relational database is accessed through an interface, a *graph visualization client* which is implemented by University of Alabama in Huntsville (See Figure 7).

The AMSR-E data production streams are based in a workflow execution environment that includes a higher level script (workflow engine), and multiple services (science code executing Perl scripts, housekeeping and processing automation scripts). The high level script acts as an agent who executes the workflow on the user's behalf, so provenance collection occurs there as well. Finally, collection is done within the services (sometimes through proxy services).

In our approach, we follow the traditional way of collecting provenance data that is to instrument each of the components in this environment and have them dump the provenance data into log files which are then harvested by an adapter to send provenance notifications at appropriate times. Although information capture is complete, assuming reliable transfer, this approach requires intensive instrumentation effort, that is, the aforementioned "full provenance instrumentation" approach.

7.3 Provenance Representation

Provenance data can be represented and stored using different technologies, including relational databases, semantic web technologies RDF and OWL, internal private formats, and relational databases together with XML views [10].

To represent the AMSR-E domain provenance data, we adopt Karma's data model. Karma implements the two-layer information model we proposed in [6] containing a *registry* level which contains metadata about the instance, and an *execution* level. The registry level has similarities to registries used in web service architectures in that it contains information about services and data products at a sufficient level of detail to support discovery and automated decisions about whether or not to bind a particular data product or service. The registry for provenance is not used for binding, but needs to contain sufficient information for building a data object that can be preserved indefinitely. The execution level captures instance invocation and execution details of a particular sequence of actions.

The two-layer model recognizes commonalities in workflows and stores that common information consistently and without redundancy. In summary, the registry level captures the metadata of services, the methods inside a service, the name and type of input parameters and output results of each method, and the structure of workflows in terms of services for predefined workflows. The order of method execution is recorded in the execution level by method invocation. Karma Service currently supports four notification types, each consisting of a number of notification messages:

- *workflow activities*, which includes *InvokingWorkflow*, *WorkflowInvoked*, *WorkflowInitiated*, and *WorkflowTerminated*;
- *service activities*, which includes *InvokingService*, *ServiceInvoked*, *ServiceInitialized*, and *ServiceTerminated*;
- *message passing activities*, which includes *SendingResponse*, *ReceivedResponse*, etc.; and
- *data activities*, which include *DataConsumed*, *DataProduced*.

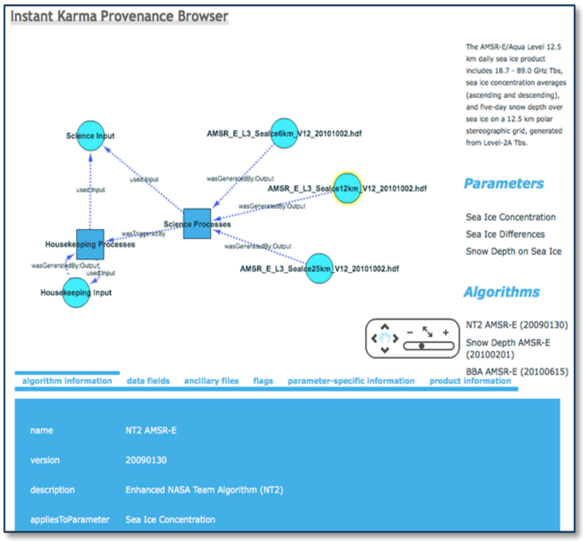
Figure 6 is an example of an *InvokingService* notification generated by the Client handler. Each invoker and invokee contains the information of workflow ID, service ID, method ID, workflow node ID, timestep. Each workflow run has a unique instance ID. This notification says that L3seaice service was invoked at 1:08:52 on June 4, 2011.

```
<kar:invokingService
xmlns:kar="http://www.dataandsearch.org/karma/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <invoker>
    <workflowID>seaice-20110621185112</workflowID>
  <serviceID>/amsr/karmadev/products/bin/seaice/@L3seaice
</serviceID>
    <methodID>unknown-method</methodID>
    <workflowNodeID>WorkflowNode</workflowNodeID>
    <timestep>-1</timestep>
    <instance>75e8662c-86ac-48ce-b010-d3443bff0f2d</instance>
  </invoker>
  <invokee>
    <workflowID>seaice-20110621185112</workflowID>
    <serviceID>/amsr/karmadev/products/bin/seaice/@L3seaice
</serviceID>
    <timestep>1</timestep>
    <instance>75e8662c-86ac-48ce-b010-d3443bff0f2d</instance>
  </invokee>
  <invokingServiceTimestamp>2011-06-04T01:08:52.147-
04:00</invokingServiceTimestamp>
  <descAndAnnotation/>
</kar:invokingService>
```

Figure 6. Example notification for *InvokingService*

7.4 Provenance Browser

Figure 7 illustrates a visualization of provenance information captured from the AMSR-E Sea ice Production workflow using the provenance browser implemented in University of Alabama at Huntsville. We have two kinds of nodes, artifacts and processes. The edges show the dependency between them. The square shaped nodes represent processes, while the rounded shapes represent artifacts.



Instant Karma Provenance Browser

The AMSR-E/Aqua Level 12.5 km daily sea ice product includes 18.7 - 89.0 GHz Tbs, sea ice concentration averages (ascending and descending), and five-day snow depth over sea ice on a 12.5 km polar stereographic grid, generated from Level-2A Tbs.

Parameters

Sea Ice Concentration
Sea Ice Differences
Snow Depth on Sea Ice

Algorithms

NT2 AMSR-E (20090130)
Snow Depth AMSR-E (20100201)
BBA AMSR-E (20100615)

name	NT2 AMSR-E
version	20090130
description	Enhanced NASA Team Algorithm (NT2)
appliesToParameter	Sea Ice Concentration

Figure 7 Visualization of provenance captured by Karma Provenance Collection Tool

In addition to displaying the provenance graph itself, the browser also provides access to additional metadata about the data products and the processes used to generate them. For example, hovering the mouse over an output artifact (sea ice data file name) will display the metadata associated with that data file. Similarly, hovering the mouse over a process artifact will display the names and versions of science or housekeeping processes represented. The left side of the browser screen displays a brief product description, list of parameters contained in the data files, and list of algorithms used to generate these parameters. At the bottom of the screen is more detailed information about the science algorithms, ancillary files used, and other processing information, with links to full documentation. All of this additional information is drawn from static metadata associated with each science data product type.

8. Conclusion and Future Work

Several things have become clear during the course of the project. One is that basic OPM entities and relationships are not adequate for expressing the kinds of provenance that is of interest. OPM supports name-value pair annotations that can be used to augment what is known about the provenance entities and relationships, but in Karma, annotations cannot be added during capture, only after the fact which limits the capture system's ability to record something it learned later. Too, annotations are not described by a common vocabulary so one of our future work items is to identify a suiteable vocabulary. A third future work item arises from the fact that not all provenance is created equal. In processing pipelines, some provenance is repetitive and uninteresting. Because of the volume of provenance, this obscures what are the interesting pieces of provenance. Methodologies to reveal the interesting provenance and supress the uninteresting are in work. Finally, provenance harvesting has occurred as part of the capture system and by the browser directly. In current work these are being brought together.

Acknowledgements

The Instant Karma project is funded by the NASA Advancing Collaborative Connections for Earth System Science (ACCESS) program.

References

[1] Y. Simmhan, B. Plale, D. Gannon, A survey of data provenance in e-Science, ACM SIGMOD Record, 34(3) (2005) 31-36.

[2] Y. Simmhan, B. Plale, D. Gannon, Towards a Quality Model for Effective Data Selection in Collaboratories, IEEE Workshop on Workflow and Data Flow for Scientific Applications (SciFlow06), held in conjunction with ICDE, Atlanta, GA, 2006.

[3] Regner, Kathryn, H. Conover, B. Beaumont, S. J.

Graves, L. Hawkins, P. Parker, et al., 2005, Flexible Processing at the AMSR-E SIPS, IEEE International Geoscience and Remote Sensing Symposium (IGARSS '05), Seoul, Korea.

[4] T. Oinn, M. Greenwood, M. Addis, M. Alpdemir, J. Ferris, K. Glover, C. Goble, A. Goderis, D. Hull, D. Marvin, P. Li, P. Lord, M. Pocock, M. Senger, R. Stevens, A. Wipat, C. Wroe, Taverna: Lessons in creating a workflow environment for the life sciences, Concurrency and Computation: Practice and Experience, 18(10) (2006) 1067–1100.

[5] C. Scheidegger, D. Koop, E. Santos, H. Vo, S. Callahan, J. Freire, C. Silva, Tackling the Provenance Challenge one layer at a time, Concurrency and Computation: Practice and Experience, 20(5) (2008) 473 – 483.

[6] B. Cao, B. Plale, G. Subramanian, E. Robertson, Y. Simmhan, Provenance Information Model of Karma, IEEE 2009 Third International Workshop on Scientific Workflows (SWF'09), Los Angeles, CA, July 2009.

[7] S. Miles, P. Groth, Miguel Branco, and Luc Moreau. The requirements of recording and using provenance in e-science experiments. Journal of Grid Computing, 5(1):1-25, 2007.

[8] L. Moreau (Editor), B. Plale, S. Miles, C. Goble, P. Missier, R. Barga, Y. Simmhan, J. Futrelle, R. McGrath, J. Myers, P. Paulson, S. Bowers, B. Ludaescher, N. Kwasnikowska, J. Van den Bussche, T. Ellkvist, J. Frieire, P. Groth, The Open Provenance Model (v1.01), Technical Report, Electronics and Computer Science, University of Southampton, 2008. http://eprints.ecs.soton.ac.uk/16148

[9] Beth Plale, Bin Cao, Mehmet Aktas, Provenance Collection of Unmanaged Workflows with Karma, under review, Jan 2011.

[10] L. Moreau et al., Special Issue: The First Provenance Challenge, Concurrency and Computation: Practice and Experience, 20 (5) (2008).