Scientific data exchange: a schema for HDF5-based storage of raw and analyzed data

Francesco De Carlo,[a]* Doğa Gürsoy,[a] Federica Marone,[b] Mark Rivers,[c] Dilworth Y. Parkinson,[d] Faisal Khan,[a] Nicholas Schwarz,[a] David J. Vine,[a] Stefan Vogt,[a] Sophie-Charlotte Gleber,[a] Suresh Narayanan,[a] Matt Newville,[c] Tony Lanzirotti,[c] Yue Sun,[e] Young Pyo Hong[e] and Chris Jacobsen[a,e,f]

[a]Advanced Photon Source, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, IL 60439, USA, [b]Swiss Light Source, Paul Scherrer Institut, Villigen, Switzerland, [c]The University of Chicago, Center for Advanced Radiation Sources, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, IL 60439, USA, [d]Advanced Light Source, 6 Cyclotron Road, Berkeley, CA 94720, USA, [e]Department of Physics and Astronomy, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208, USA, and [f]Chemistry of Life Processes Institute, Northwestern University, 2170 Campus Drive, Evanston, IL 60208, USA. *E-mail: decarlo@aps.anl.gov

Data Exchange is a simple data model designed to interface, or 'exchange', data among different instruments, and to enable sharing of data analysis tools. Data Exchange focuses on technique rather than instrument descriptions, and on provenance tracking of analysis steps and results. In this paper the successful application of the Data Exchange model to a variety of X-ray techniques, including tomography, fluorescence spectroscopy, fluorescence tomography and photon correlation spectroscopy, is described.

Keywords: data storage; provenance; HDF5.

1. Introduction

When it comes to the digital storage of experimental data and analyses results at synchrotron light sources around the world, the situation resembles Babel (Bible, n.d.). As different research teams and techniques have grown at various facilities, they have often developed local data storage formats based on instrument hardware specificity and expediency rather than rational planning, often drawing upon the particular preferences of a scientist or engineer writing software at the project's outset. In some cases, simple text files are used because of their human readability in spite of their inefficiency with respect to storage size and the cost of parsing text. In other cases, images are stored using common image files, but without systematically saving the metadata describing experiment conditions or analysis parameters.

The Data Exchange is a simple data model that is designed to interface, or 'exchange', data among different instruments, and to enable sharing of data analysis tools. This is particularly important since more and more scientific users perform experiments at different synchrotron light sources (Kanitpanyacharoen et al., 2013). The Data Exchange implementation uses the Hierarchical Data Format 5, or HDF5 (The HDF Group, 2013), a widely used and supported storage format for scientific data. The Data Exchange is highly simplified and focuses on technique rather than instrument descriptions, and on provenance tracking for understanding analysis steps and results. Provenance information is stored in a manner that can

be used with a workflow pipeline to automatically run analysis steps while maintaining human readability.

Here we describe the successful application of the Data Exchange model to a variety of synchrotron-based techniques, including X-ray tomography, X-ray fluorescence spectroscopy, X-ray fluorescence tomography, coherent diffraction imaging and X-ray photon correlation spectroscopy.

2. Background

A popular image format used in the scientific community is the Tagged Image File Format (TIFF). The TIFF standard allows for the addition of 'private' tags beyond those used by generally available software. This feature has been used by the Open Microscopy Environment's data format (OME-TIFF, 2013) and the GeoTIFF format (GeoTIFF, 2013). However, TIFF files suffer from file size limitations, and may require the use of separate metadata files to describe data that do not logically fit into a series of separate image files.

The Crystallographic Information File (CIF) is a successful example of a standard file format for representing crystallographic data. In addition, the Protein Data Bank uses the pdb format (PDB, 2013) to store three-dimensional molecular structures in a standardized way. Both formats store data in standard text files, which is adequate due to the relatively small size of these files with respect to imaging datasets.

Within the synchrotron light source and neutron facility community, NeXus (Tischler, 1984) has a long history of

development, though its adoption at synchrotron light sources in particular has been uneven and its design is centered primarily on the storage of as-acquired data (NeXus, 2013). New European initiatives like the Photon and Neutron data infrastructure (PANdata, 2013) and the High Data Rate Initiative for Photon, Neutrons and Ions (PNI-HDR, 2013) are planning to extend the NeXus interfaces between various instrument controls and NeXus libraries creating NeXus definitions to fully describe the instruments where the data were collected. For X-ray fluorescence microprobes (XFM) no distinct file format exists, even though basic data structure is similar among beamlines. Most XFM beamlines operating in an imaging mode typically save full energy-dispersive spectra as a data array consisting of energy bins *versus* intensity (either as a single spectra per pixel or as a list-mode event stream) along with pixel (or motor) coordinates, scaler information, *etc*. Yet despite this commonality the situation today is that, by and large, every beamline saves data in a different format, making the development of a common set of data-processing tools difficult.

The HDF5 library is a widely used and supported open-source binary file format supporting utilities for large scientific datasets available for use in a myriad of programming languages (The HDF Group, 2013). HDF5 files are self-describing and portable, which anyone who has tried to decode an undocumented binary data file can immediately appreciate. HDF5 files are also used in the financial services industry (Bethel *et al.*, 2011), and their advantages have been appreciated for biological imaging (Dougherty *et al.*, 2009; Eliceiri *et al.*, 2012), for X-ray spectroscopy (Ravel *et al.*, 2012; Medjoubi *et al.*, 2013), for X-ray fluorescence microscopy analysis packages (Vogt, 2003; Solé *et al.*, 2007) and for coherent diffraction imaging (Steinbrener *et al.*, 2010; Maia, 2012). While the NeXus format noted above allows for implementation *via* XML and HDF4 files as well as HDF5 files, most new implementations of NeXus are concentrating on using the HDF5 file format.

3. The Data Exchange model

Because of the impossibility of designing a schema that incorporates all possible types of data collection, analysis results and data representations, the Data Exchange definition limits the amount of required structures, while allowing correct Data Exchange files to include as much additional information as desired.

The key principle of Data Exchange is that, in most cases, for each experiment technique, there is one particular data array that an analysis or visualization program, or researcher, will want to access. For example, this may be a series of normalized projections, a spectrum, a spectrum image, a scattering pattern or a reconstructed three-dimensional volume. This key array is placed into an HDF5 group called exchange for easy identification.

This gives maximum flexibility for various analysis programs to add various derived results to an original data file

Table 1
A simple Data Exchange file.

HDF5 object	Description
exchange	HDF5 Group for primary data
projections	Three-dimensional array of image projections taken over time
time	One-dimensional array of time stamps for projections
theta	One-dimensional array of angles corresponding to projections
measurement	HDF5 Group
instrument	HDF5 Group for instrument definitions
sample	HDF5 Group for static sample information
provenance	HDF5 Group for provenance information
process_table	Table storing list of provenance processes
copy	Group with parameters for a copy operation
reconstruction	Group with parameters for reconstruction

without compromising or altering the section of the file devoted to original storage of the acquired data. Instrument and data collection metadata tags are given fixed names with a naming scheme meant to be maximally compatible with both NeXus (NeXus, 2013) and CXI (Maia, 2012) files, and are meant to be descriptive enough for easy human readability using an application such as *HDFView*.

Many scientists have probably seen data files that refer to a wavelength without specifying whether it is in Ångstroms or inverse centimeters, or that refer to an angle without indicating whether it is in degrees or microradians. In Data Exchange, all variables are required to specify the physical units. The text strings that describe the units should conform to those defined by the *UDUNITS* package (UCAR, 2013).

Finally, Data Exchange uses plain HDF5 calls, rather than a separate Application Programming Interface (API). In this way one avoids the rather substantial effort that would otherwise be required to debug and maintain an API across many platforms for an ever changing set of requirements.

Again, our goal is that one should maximize both the future extensibility of a file to meet evolving data acquisition schemes and data analysis tools, and also human readability (*via*, for example, *h5dump* or *HDFView*) so that one can manually examine a file without access to the computer code used to write it.

3.1. Storing data

The full definition for Data Exchange HDF5 files is published online (Argonne, 2014). Here we outline the key characteristics of a basic Data Exchange file, with the understanding that Argonne (2014) contains the reference description as well as a number of example Data Exchange read and write programs in several computer languages. A simple Data Exchange file layout is shown in Table 1.

While HDF5 gives great flexibility in data storage, straightforward file readability and exchange requires adhering to an agreed-upon naming and organizational convention. To achieve this goal, Data Exchange adopts a layered approach by defining a set of *mandatory* and optional fields. That is, even a simple Data Exchange file

should be able to be considered 'correct' according to the schema, while allowing for as rich a set of metadata tags as any technique might desire.

In a minimal Data Exchange file, the only mandatory items are a HDF5 group called *exchange* and a HDF5 dataset called *implements*. The *implements* dataset is a string that describes which 'exchange' groups have been added to the file. This expedient obviates the need to parse the file structure to see if a particular group exists. As mentioned previously, *exchange* is the data of primary interest in the file. The *exchange* group may contain any number of HDF5 datasets that contain actual data or links to data points or arrays. Associated values that relate to the data in the *exchange* group, and that may be considered axis values for the data, should be stored in additional datasets in the *exchange* group. These data may be the times and angles at which projections were taken, for example. Data Exchange uses the dimension scales HDF5 feature (an HDF5 feature that is incidentally not presently supported by NeXus) to associate the array of data values with its associated axis information. In part because of this requirement, Data Exchange requires use of HDF5 version 1.8 or later; all other software requirements are listed by Argonne (2014). Multiple *exchange* groups can be added to a file by appending a '_N' to the name of the group, for example, *exchange_4*.

A more general Data Exchange file might also contain the optional `measurement` group. This group contains sample and instrument information which is expected to be static throughout the measurement (*e.g.* sample preparation information and instrument configuration). As with the *exchange* group, multiple `measurement` groups can be added to a file by appending a '_N' to the group name, for example, `measurement_4`.

Beyond this, additional groups may be added to meet individual needs, with guidelines suggesting the best structure.

3.2. Maintaining provenance

Another optional group is the `provenance` group. This group contains information about all transformations, analyses and interpretations of data performed by a sequence of process functions. Maintaining this history allows for reproducible representations of data. The Data Exchange format tracks provenance with a provenance process table. The provenance process table tracks the execution order of a series of processes *via* a list of sequential entries in the process table. The Data Exchange model uses this approach instead of using a separate traditional relational database to maintain provenance information for two main reasons. One is so that all relevant information regarding actions taken on the data may be stored together along with the data in one container. The other is that standard relational databases require stricter up-front definitions of tables and data types which requires more effort when extending as opposed to adding more key-value pairs.

An example of the provenance process table is shown in Table 2. Rows in the table represent actions performed on the data. Each row has a number of properties associated with it,

Table 2
An example provenance process table.

Name	Status	Message	Reference
copy	FAILED	auth. error	/provenance/copy
copy	SUCCESS	OK	/provenance/copy
norm	SUCCESS	OK	/provenance/norm
reconstruction	SUCCESS	OK	/provenance/reconstruction
convert	RUNNING		/provenance/export
copy	QUEUED		/provenance/copy_2

some of which are name, status, message and reference. Other properties are omitted for the sake of brevity.

The most important property in the process table is the reference. The reference is a text string that refers to another group in the HDF5 file that describes the parameters needed to perform a particular process on the data. For something simple like a file transfer, the group may contain source and destination Uniform Resource Identifiers (URIs). For an analysis step, the group may contain parameters for the analysis algorithm, including the input and output datasets within the file used for that specific instance of a process. Increasing the parameters within the appropriate reference group allows for the representation of arbitrarily complex analysis processes. Likewise, increasingly complex workflows of many analysis steps may be constructed by adding more entries to the process table. Workflow pipelines, such as mentioned in the use cases below, may use this information to re-run analysis attempts due to either a failure or a desire to modify parameters resulting in additional entries in the table, and possibly other parts of the file. In order to maintain human understandability, it is encouraged to make parameters for analysis steps the actual algorithm parameters and not only the text of the command line arguments required to run a specific tool.

Scientific users are not generally expected to maintain data in this group, however. The expectation is that analysis pipeline tools will automatically modify processing steps using this group. An analysis workflow pipeline compatible with Data Exchange files has been demonstrated for X-ray tomography experiments by Schwarz *et al.* (2013), and for X-ray photon correlation spectroscopy analysis by Khan *et al.* (2013).

4. Data Exchange for full-field X-ray tomography

A tomographic raw dataset consists of a series of projections (data), dark-field (data_dark) and white-field (data_white) images. Since dark and white fields can be collected at any time before, after or during the projection data collection, Data Exchange uses the angular position of the tomographic rotation axis, theta, to keep track of when the dark and white images were collected. Data Exchange saves the raw dataset in three-dimensional arrays using, by default, the natural HDF5 order of a multi-dimensional array (rotation axis, ccd y, ccd x), *i.e.* with the fastest changing dimension being the last dimension, and the slowest changing dimension being the first dimension. The definition of the Data Exchange implementation for tomography (see Argonne, 2014) allows for

storing of intermediate processing steps like normalized projections. Normalized projections represent the first data entry array for any subsequent three-dimensional reconstruction algorithm.

At the Advanced Photon Source (APS), the *areaDetector* software (Rivers, 2010) is used to integrate various cameras into the APS control system. A specialized *areaDetector* plug-in is under development to write fully compliant Data Exchange HDF5 files.

The raw data are subsequently processed using tomoPy, an open source framework developed at the APS for the analysis of synchrotron tomographic data (Gürsoy *et al.*, 2014).

At the Swiss Light Source, the need for a new approach for digital storage of experimental data has become evident during the recent developments towards an ultrafast tomography endstation (Mokso *et al.*, 2010). The new in-house-developed readout system [GIGAbit Fast Read-Out SysTem (GIGAFROST)] for a CMOS detector permits the readout of the sensor in a fully continuous unlimited mode achieving rates as high as 8 GB s^{-1}. To efficiently handle and in particular post-process this large amount of raw data, at rates consistent with the acquisition, the current reconstruction pipeline has been revisited at different levels. A major input/output (I/O) bottleneck has clearly been identified in the originally used TIFF format, with thousands of small files for individual projections. The HDF5 technology enables significant improvements in the current I/O performance, bringing performance close to hardware limits of modern shared file systems, 8 GB s^{-1} for the current General Parallel File System (GPFS) set-up. In our implementation of the data collection, individual User Datagram Protocol (UDP) datagrams dispatched by GIGAFROST are assembled to images in shared memory where the data are represented as 3d numpy arrays, *i.e.* stacks or series of images in Python. When the tomographic dataset is complete, the data are dumped in a sequential way to disk to a HDF5 file using the direct chunk write function (Donath *et al.*, 2013) and an nbit-filter, for instance, for native 12-bit images. No additional compression algorithms are used for X-ray tomography data, since the achieved compressibility factor is in general very low (less than 1.5).

The Data Exchange model is particularly attractive for its simplicity combined with flexibility and completeness. In the first phase, we focused exclusively on the raw data, in addition to the metadata characterizing the acquisition set-up and parameters. In the simplest implementation we store the raw data (projections, dark and white fields) in their respective datasets in the *exchange* group using the natural order of multi-dimensional arrays in HDF5. The flexibility of the Data Exchange model with its object attributes permits us to test different array orders to optimize performance, without need for changes in the post-processing software. If, for instance, the default first and second dimensions are swapped, the data are actually directly organized as uncorrected sinograms. This arrangement brings an advantage during post-processing, if no projection based pre-processing (*e.g.* standard phase retrieval) is needed. We measured a 35% acceleration in the generation

of corrected sinograms if raw data in default HDF5 Data Exchange compliant format are used as opposed to individual TIFF files. An improvement as high as 57% is instead obtained if this alternative organization of the multi-dimensional arrays is used. Fine-tuning of the raw data organization, including different chunking strategies, is currently on-going to reach best I/O performance. Best results seem to require an intermediate chunk size, while small chunks implying a large amount of chunks for our typical datasets are penalizing the performance. The tests presented here have been run on a distributed environment, with multiple processes reading simultaneously. The used file system is GPFS, a high-performance parallel file system. Other file systems (*e.g.* NTFS or tiered file systems) are not suitable for our applications. The flexibility of even the simplest Data Exchange format has also already been proven in recent tomographic dynamic experiments, where the onset of the process of interest was not easy to control. In such a case it is difficult to obtain two high-quality three-dimensional volumes right before and after the changes occurring. To obviate this difficulty, we continuously acquired and stored projections in a well documented single HDF5 file, while rotation of the sample over several revolutions, instead of standard tomographic datasets implying a rotation of 180°. After *a posteriori* recognition of the event time point, the post-processing software could easily extract the relevant projections for the reconstruction of the volumes of interest.

In a second phase, we plan also to take advantage of additional features provided by the Data Exchange model. The aspect related to provenance is of particular interest, which we envisage to link to our database as well as graphical user interface, so to easily have a control over the outcome of the different steps of the post-processing pipeline. In addition, on-going internal discussions are devoted to establishing the optimal internal file organization, both for storing raw data for complicated experiments involving multiple scans (for instance under different conditions) and, when needed, for intermediate post-processing results.

At the Swiss Light Source, Python has been the main language for implementation of the data backend system including writing of Data Exchange compliant files. C/C++ has been used when the achieved performance was not sufficient as for instance for receiving, from the camera, UDP datagrams and assembling them in shared memory to images. Python has also been the language of choice in combination with the flexible Message Passing Interface (MPI) for data reading and post-processing. To improve the computational performance, for some selected parts, Cython has been used.

At the Advanced Light Source (ALS), data rates and volumes are increasing at an unprecedented rate. At the same time, analysis and simulation software for lightsource data are increasingly sophisticated and represent greater investment in development as well as providing greater capability. To make these advanced capabilities available to more scientists, we have developed a suite of data management, processing, analysis and simulation tools named *SPOT Suite* (ALS, 2014), in collaboration with Lawrence Berkeley Laboratory's

Computational Research Division and the National Energy Research Scientific Computing Center (NERSC). HDF5 was chosen as the data format for *SPOT Suite*, but during initial developments a non-specialized HDF5 format was used. Because the data processing and analysis needs of the hard X-ray tomography beamline at the ALS are similar to those at other synchrotron tomography beamlines, a common file format optimized for tomographic data could greatly facilitate sharing of and collaboration on processing software, and improves the user experience for those who use multiple different facilities. Initial software is being developed to allow *SPOT Suite* to begin using Data Exchange for tomography data, and these changes will soon be put into the production version of the code.

5. Data Exchange for X-ray fluorescence microscopy

An X-ray fluorescence (XRF) dataset typically consists of a series of elemental maps (images), derived from raw X-ray fluorescence spectra acquired with energy-dispersive detector systems in a scanning probe geometry. Depending on the specific instrumentation, these may comprise single or multiple individual detector elements. These data are typically complemented with scalar per-pixel information, such as incident flux, transmitted flux, additional detector information (live time, count-rate, ...). Typically, each of these elemental maps give the planar distribution (sample *x versus* sample *y*) of one specific element of interest in the sample studied, although data acquisition strategies that record individual scan lines (1D) or tomographic line projections (sample *x versus* sample theta) are also encountered. Full XRF tomography is typically acquired in a series of individual (separately saved and processed) projections, and discussed below. We note that, intrinsically, there is significant variation in the type and amount of data acquired in scanning microprobes. For example, one system may use just one single-element energy-dispersive spectroscopy detector counting for a fixed elapsed live time, whereas another system may use a 96-element XRF detector requiring per channel normalization. The goal of the exchange implementation, at minimum, is to make it possible to open the file with a variety of software tools, and work with the data no matter the source, and specific instrument configuration. In addition, we note that the built-in support of compression into HDF5 can be a significant boon particularly for XRF microprobes. Significant savings in file size can typically be achieved on fluorescence spectra.

The primary data produced by an X-ray fluorescence microscope is a number of photons detected with a given energy originating from a particular spatial location on the sample. These raw data can then be analyzed in a number of ways to produce the (quantified) spatial distribution of elemental concentration that is typically the goal of the measurement. Data Exchange for X-ray fluorescence microscopy strives to make the raw data available for (re-)analysis with the knowledge that most end users and visualization software will work with analyzed data. To this end we reserve 'exchange_0' for the raw data and 'exchange_{1,2,...,N}' for

the analyzed data. The 'exchange' group is designated as a soft link to the dataset most relevant to the recipient of the Data Exchange file. In this way we preserve the flexibility of the Data Exchange model and enabling immediate access to the raw data without the need to define special tags which necessitate parsing and searching the entire file for them.

To accommodate arbitrary scan types we have adopted list-based storage of the raw data. In the most general case each pixel of a scan consists of an energy measurement recorded as a function of an independent variable (position, temperature, angle *etc.*). In the Data Exchange file each measurement is stored in the 'data' dataset of the 'exchange_0' group with an attribute describing the independent axis as, *e.g.* 'theta:y:x'. A separate dataset 'axes' in the 'exchange_0' group records the independent variable for each entry in 'data'. For the most common scan types like a two-dimensional image this approach requires additional processing to arrange the data into an array; however, it has the advantage of being able to handle a range of scan geometries: Cartesian and spiral, raster and zigzag, one-dimensional and up. Each independent channel of multi-element energy-resolving detectors are stored individually as 'data_N' for $N = 0, 1, \ldots, M$ and the integrated spectra stored as 'data'.

Per-pixel scalar channels (storage ring current, ion chamber, *etc.*) are stored as a list-based dataset 'channels' in the raw data 'exchange_0' group. For N scalar channels each entry in the 'channels' dataset will be an N-element array and the corresponding labels for each channel stored in a separate 'channel_names' dataset under 'exchange_0'. Optional recommended channel names for standard scalar quantities like the storage-ring current are defined in the Data Exchange reference document. Standardizing the channel names greatly aids in the development of automated software by making the data facility and instrument agnostic.

List-based storage of data is optional for analyzed datasets. In most cases visualization or tomographic reconstruction software will expect data to be arranged in a two- or three-dimensional array and in these cases it makes sense to store the data natively in a Cartesian array of the appropriate dimensionality. For higher-dimensional scans (*e.g.* tomograms as a function of time) the list-based approach is recommended.

The analyzed data can be normalized in different ways or not at all. We recommend that each analyzed data exchange group contain the un-normalized data and optionally a normalized dataset. This approach allows easy re-normalization and also accommodates storing a fully analyzed dataset. The normalization method can be described in the attribute of the normalized dataset and the units attribute set accordingly.

As an example of the utility of the Data Exchange model, consider X-ray fluorescence tomography. Adopting the Data Exchange format enabled APS scanning probe beamlines to easily make use of the sophisticated and actively developed tomography analysis software created by the more mature full-field tomography beamlines. A data analysis pipeline was quickly adjusted to also handle XRF tomography data. Centralizing the tomography reconstruction software devel-

opment effort reduces duplication of effort and ensures that all tomography users benefit from the continual improvement to the software.

6. Data Exchange for X-ray photon correlation spectroscopy

X-ray photon correlation spectroscopy (XPCS) is a unique technique to probe the equilibrium and non-equilibrium dynamics in materials over a wide range of time scales and length scales down to nanometers. Typical XPCS data consist of a time series of 2D detector images acquired at a constant time interval. By time correlation of the speckle pattern arising due to the scattering of a fully or a partially coherent beam from a disordered material, the wavevector- or length-scale-dependent dynamical time scales of the system being probed can be extracted. During the last decade, XPCS has been successfully applied to probe a wide range of soft and hard matter systems.

The Data Exchange model has been implemented for XPCS using the following HDF5 groups in conjunction with a high-performance computing (HPC) based data analysis system:

(i) measurement: comprising instrument, acquisition, detector and source sub-groups;

(ii) sample: stores sample-related information such as thickness, temperature and other relevant parameters;

(iii) provenance: containing steps pertaining to the data-processing steps;

(iv) xpcs: a rich set of metadata that contains information regarding how the data need to be processed in a HPC environment:

(*a*) region of interest of the data to be processed;

(*b*) pixellated wavevector map dividing the 2D array into static and dynamic wavevector maps based on which the individual pixel correlations will be binned to yield the final wavevector-dependent correlation functions;

(*c*) details of the start and the end frames to be processed in the case when only a subset of the data is processed;

(v) exchange: contains the following detailed computational results saved in several datasets:

(*a*) wavevector-dependent correlation functions computed from the raw data;

(*b*) modeled correlation functions based on non-linear least-square fitting to exponential and stretched/compressed exponential functions;

(*c*) dynamical wavevector-dependent time scales extracted from curve fitting.

Similar to the implementations for the other techniques presented earlier, multiple analysis performed on the same dataset under different processing steps are tracked in a provenance table.

The APS had pioneered the development of the XPCS technique using area detectors and has likewise established the lead in developing HPC-based data reduction and analysis yielding time auto-correlations in near real-time. The successful implementation of the data exchange model to XPCS will enable potential collaboration with the recently built XPCS beamline P-10 at PETRA III and the upcoming hard X-ray coherent scattering beamline at NSLS-II.

7. The future of the Data Exchange model

The initial goal of Data Exchange is to facilitate exchange of data and analysis software tools among facilities performing the same or similar technique. The model has been applied to data from a range of facilities including ALS, APS, ANKA, Australian Synchrotron, Diamond, ESRF ID-19, Elettra, PETRA III, SLAC, SLS and X-radia systems. For example, it has already resulted in demonstrable improvements at APS where the tomoPy software developed by the full-field tomography beamlines has been shared with the scanning probe beamlines. We envision that adoption of Data Exchange will empower facility users to select the best analysis tool for their data irrespective of where the data were acquired or where the software was developed.

The Data Exchange strategy is not to solve the 'Babel situation' described in the *Introduction* by creating yet another format but to create an intermediary that everyone can export to and read from. For this reason, even within the same technique, Data Exchange does not impose rigid definition for instrument components nor of saving instrument status at each data collection point (unlike the NeXus implementation, for example); instead, its goal is in finding consensus within each class of instrument in what is the most meaningful and basic raw dataset to share.

To expand the Data Exchange model and possibly turn it into the native data format of choice at large facilities we adopted the following strategy:

(i) Maintain, expand and freely distribute tomoPy (Gürsoy *et al.*, 2014), the APS tomography reconstruction software that natively uses Data Exchange.

(ii) Expand the ability of tomPy to natively read both Data Exchange and facility-specific data. These data importers are published and constantly updated in the Argonne (2014) demo section.

(iii) Develop a data exchange plug-in for area detectors (Rivers, 2010) making it possible for all facility using this software to control their camera and to save tomographic data directly in Data Exchange.

With more and more users performing experiments at different synchrotron facilities, we expect the need to exchange data and software tools will grow and facilitate this approach.

8. Conclusions

The definition of Data Exchange as a simple data model designed to interface, or 'exchange', data among different instruments has the potential to improve the inter-operability of software toolboxes currently under development at various synchrotron facilities. Perhaps most importantly, the migration to a more standardized file format among beamlines for the techniques described here will empower users. It can be argued that users can then more freely utilize a broader suite

of analysis tools that may have been developed at different facilities for different beamlines, but may be best suited to their particular scientific study. This would allow, for example, not only the interfacing of varying data types that are collected simultaneously but also the integration of complementarity data that may have been acquired at different beamlines (*e.g.* TXM and XRFT) or with non-synchrotron-based instruments (optical microscopy, electron beam instruments, *etc.*). As examples, consider that this would also allow for more seamless registration of three-dimensional image tomographic datasets or for X-ray microfluorescence and microdiffraction data, one collected using energy-dispersive detectors and the other using area detectors. For example, at the APS, tomoPy (Gürsoy *et al.*, 2014), in connection with Data Exchange, is being developed to provide the ability to analyze tomography data from all major synchrotron facilities around the world, for various tomographic techniques.

References

ALS (2014). *The SPOT Suite Portal*, http://spot.nersc.gov/.
Argonne (2014). *The Scientific Data Exchange*, http://www.aps.anl.gov/DataExchange/.
Bethel, E. W., Leinweber, D., Rübel, O. & Wu, K. (2011). *Proceedings of the Fourth Workshop on High Performance Computational Finance (WHPCF '11)*, pp. 23–30. New York: ACM. http://doi.acm.org/10.1145/2088256.2088267.
Bible (undated). *The Bible*, **11**, 1–9.
Donath, T., Rissi, M. & Billich, H. (2013). *Synchrotron Radiat. News*, **26**, 34–35.
Dougherty, M. T., Folk, M. J., Zadok, E., Bernstein, H. J., Bernstein, F. C., Eliceiri, K. W., Benger, W. & Best, C. (2009). *Commun. ACM*, **52**, 42–47.
Eliceiri, K. W., Berthold, M. R., Goldberg, I. G., Ibáñez, L., Manjunath, B. S., Martone, M. E., Murphy, R. F., Peng, H., Plant, A. L., Roysam, B., Stuurmann, N., Swedlow, J. R., Tomancak, P. & Carpenter, A. E. (2012). *Nat. Methods*, **9**, 697–710.
GeoTIFF (2013). *GeoTIFF*, http://trac.osgeo.org/geotiff/.
Gürsoy, D., De Carlo, F., Xiao, X. & Jacobsen, C. (2014). *J. Synchrotron Rad.* **21**, 1188–1193.
Kanitpanyacharoen, W., Parkinson, D. Y., De Carlo, F., Marone, F., Stampanoni, M., Mokso, R., MacDowell, A. & Wenk, H.-R. (2013). *J. Synchrotron Rad.* **20**, 172–180.
Khan, F., Hammonds, J., Narayanan, S., Sandy, A. & Schwarz, N. (2013). *Proceedings of the 14th International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS2013)*, 6–11 October 2013, San Francisco, CA, USA.
Maia, F. R. N. C. (2012). *Nat. Methods*, **9**, 854–855.
Medjoubi, K., Leclercq, N., Langlois, F., Buteau, A., Le, S., Poirier, S., Mercere, P., Sforna, M. C., Kewish, C. M. & Somogyi, A. (2013). *J. Synchrotron Rad.* **20**, 293–299.
Mokso, R., Marone, F. & Stampanoni, M. (2010). *AIP Conf. Proc.* **1234**, 87–90.
NeXus (2013). *NeXus*, http://www.nexusformat.org/.
OME-TIFF (2013). *The OME-TIFF Format*, http://www.openmicroscopy.org/site/support/ome-model/ome-tiff/.
PANdata (2013). *NeXus/HDF5 developments*, http://pan-data.eu/NeXus/.
PDB (2013). *Protein Data Bank Contents Guide: Atomic Coordinate Entry Format Description*, http://www.wwpdb.org/documentation/format33/v3.3.html.
PNI-HDR, (2013). *Pni-hdr*, http://www.pni-hdri.de/.
Ravel, B., Hester, J. R., Solé, V. A. & Newville, M. (2012). *J. Synchrotron Rad.* **19**, 869–874.
Rivers, M. (2010). *AIP Conf. Proc.* **19**, 51–54.
Schwarz, N., De Carlo, F., Glowacki, A., Hammonds, J., Khan, F. & Yue, K. (2013). *Proceedings of the 14th International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS2013)*, 6–11 October 2013, San Francisco, CA, USA.
Solé, V. A., Papillon, E., Cotte, M., Walter, P. & Susini, J. (2007). *Spectrochim. Acta B*, **62**, 63–68.
Steinbrener, J., Nelson, J., Huang, X., Marchesini, S., Shapiro, D., Turner, J. J. & Jacobsen, C. (2010). *Opt. Express*, **18**, 18598–18614.
The HDF Group (2013). *HDF5*, http://www.hdfgroup.org/HDF5/.
Tischler, J. Z. (1984). *Nucl. Instrum. Methods Phys. Res.* **222**, 339–340.
UCAR (2013). *UDUNITS*, http://www.unidata.ucar.edu/software/udunits/.
Vogt, S. (2003). *J. Phys. IV*, **104**, 635–638.