Published in final edited form as:

Comput Sci Eng. 2012 May 1; 15(1): 76–83. doi:10.1109/MCSE.2012.60.

A Pipeline Software Architecture for NMR Spectrum Data Translation

Heidi J.C. Ellis [Chair and Associate Professor],
Department of Computer Science and Information Technology, Western New England University, 1215 Wilbraham Road Springfield, MA 01119-2684, ph: 413-782-1748, fax: 413-796-2118

Gerard Weatherby,
Department of Molecular, Microbial and Structural Biology, University of Connecticut Health Center, 263 Farmington Avenue, Farmington, Connecticut 06030-3305, ph: 860-679-8759, fax: 860-679-3408

Ronald J. Nowling,
429 Napoleon St., South Bend, IN 46617, ph: 954-496-2314

Jay Vyas,
Department of Molecular, Microbial and Structural Biology, University of Connecticut Health Center, 263 Farmington Avenue, Farmington, Connecticut 06030-3305, ph: 860-679-8759, fax: 860-679-3408

Matthew Fenwick, and
Department of Molecular, Microbial and Structural Biology, University of Connecticut Health Center, 263 Farmington Avenue, Farmington, Connecticut 06030-3305, ph: 860-679-8759, fax: 860-679-3408

Michael R. Gryk
Department of Molecular, Microbial and Structural Biology, University of Connecticut Health Center, 263 Farmington Avenue, Farmington, Connecticut 06030-3305, ph: 860-679-4785, fax: 860-679-3408

Heidi J.C. Ellis: ellis@wne.edu; Gerard Weatherby: gerardw@alum.mit.edu; Ronald J. Nowling: rnowling@nd.edu; Jay Vyas: jayunit100@gmail.com; Matthew Fenwick: mfenwick@student.uchc.edu; Michael R. Gryk: gryk@uchc.edu

Abstract

The problem of formatting data so that it conforms to the required input for scientific data processing tools pervades scientific computing. The CONNecticut Joint University Research Group (CONNJUR) has developed a data translation tool based on a pipeline architecture that partially solves this problem. The CONNJUR Spectrum Translator supports data format translation for experiments that use Nuclear Magnetic Resonance to determine the structure of large protein molecules.

Keywords

Computer Applications: J.3.a Biology and genetics; Data: E.m Miscellaneous

1. Introduction

The issue of translating data into different formats to be used by a range of scientific software applications has been recognized as an issue for computing since data first were stored in files. The rapid growth in the amount of scientific data being generated and the development of a profusion of tools to process this data has magnified the data translation problem. The domain of Nuclear Magnetic Resonance (NMR) spectroscopy suffers from this problem.

Nuclear Magnetic Resonance (NMR) spectroscopy is used to determine the structure and function of organic molecules, including proteins and nucleic acids. The process of NMR allows researchers to identify the position of atoms in a molecule based on the response of the atoms to a strong magnetic field. Understanding the structure of proteins is important to molecular biology and biochemistry as such information leads to the development of new medications, furthers the knowledge of enzyme activities and aids in understanding of proteins.

NMR is based on the principle that certain chemical isotopes have the ability to absorb and re-emit electromagnetic radiation when within a magnetic field. The frequency of this radiation characterizes the nucleus, such that ^{1}H and ^{15}N have identifiably different frequencies. An NMR spectrometer consists of a strong magnet and electronics which energize the nuclei in a molecule to create magnetic induction which the spectrometer then measures. The frequency of a particular nucleus is slightly *shifted* due to the spatial and chemical relationships with other nuclei based on the location of the nuclei within the molecule, allowing the individual nuclei to be distinguished from one another.

The signals received from the magnetic induction are called Free Induction Decays (FIDs) and are considered the *direct* dimension of the data. In an NMR experiment, if only this data is collected, the sample is defined as *one dimensional* and typically includes1024 data points. However, a one-dimensional spectrum is often insufficient for analysis due to the difficulty in distinguishing thousands of overlapping signals, each corresponding to different atoms in the molecule.

A second dimension of data can be generated by exposing the sample to a pair of magnetic pulses with a varying time delay which will allow atoms to be more clearly identified within a molecule. This second dimension is defined as the first *indirect* dimension and the experiment is defined as *two dimensional*. This second dimension typically contains 256 data points. As the second dimension is measured for each point in the first dimension, the addition of a second dimension results in a data set that is 1024 × 256 = 262,144 data points. Figure 1 shows a graphical representation of the data for a 2D experiment. The experimenter must identify the nuclei represented by the various strength of color shown in Figure 1.

Increasing the complexity of the pulse sequence allows additional indirect dimensions to be investigated and three and four dimensional experiments to be performed. Each dimension in an experiment multiplies the total data size and execution time of the experiment by the number of data points. Therefore, four dimensional experiments are rare due to the large amount of data produced and long experiment time.

An additional complexity occurs when two signals are used to determine whether the nuclei in a molecule are rotating clockwise or counter-clockwise. These two signals are modeled mathematically as a complex number, with one signal being designated the real component and the other the imaginary. The combination of real and imaginary data collection typically occurs for each dimension further increasing the size and variety of data that must be collected and processed. In signal processing terminology the real, imaginary modeling of the data is defined as *quadrature*.

The actual process of using NMR to determine protein structure is time-consuming and detailed. Multiple different software tools [1] are used resulting in the generation of many different data files, frequently using different formats. Many of these data files contain tens of MB of data and one key aspect of correct processing of data is that the data must conform to the input format required for each tool.

There are four high-level steps involved in the process of NMR experimentation. The output of each step serves as input to the next and steps may be repeated. Phases 2 through 4 may involve the use of multiple different tools used in an iterative manner within each phase.

1. **Spectrometer Acquisition:** A protein sample is prepared and then processed by an NMR spectrometer to produce a representation of the free inductive decay over time.

2. **Spectral Reconstruction:** Software tools are used to convert the output from the spectrometer from time-domain data to frequency-domain data. The result of the spectral reconstruction step is that the data are now embodied as spectra where individual peaks represent nuclei and their relationships to each other.

3. **Spectral Analysis:** More software tools are used to identify peaks and assign them to the nuclei which give rise to them.

4. **Biophysical Characterization:** Geometric relationships between the nuclei are inferred from the spectra and used to determine the actual physical structure and dynamics of the protein.

In the domain of NMR, the number and variety of different data formats used in processing NMR data presents a problem as the ability to translate between data formats limits the tools and approaches that NMR researchers can use to determine molecular structure. While some tools exist that will convert data from one format to one or two other formats (e.g., TopSpin (http://www.bruker-biospin.com/), NMRPipe [2], Rowland NMR ToolKit [3]), these tools offer only limited translation facilities. In addition, many of these tools require the user to provide additional information in order to correctly complete the conversion process. In order to provide NMR researchers maximum flexibility in processing data, an automatic data translator that provides the ability to convert among multiple spectrometer and tool formats is critical.

The CONNecticut NMR Joint University Research (CONNJUR) team is a group of biochemical and software engineering researchers investigating the partial automation of the experimental process of employing NMR to determine the structure of large protein molecules. Our vision is to develop a comprehensive application that integrates a variety of data analysis tools with data management to support the process of protein structure determination using NMR [4, 5]. One main component of this integrated environment is a spectrum converter, the CONNJUR Spectrum Translator (CONNJUR-ST) tool. CONNJUR-ST supports the translation of data between the formats used by both different spectrometers as well as the variety of tools used to process the data. The application of CONNJUR-ST to processing NMR data for an NMR audience as well as initial development efforts is described in [4]. In this paper, we provide a comprehensive description of the CONNJUR-ST architecture, how it works, and its possible application to other domains, as well as a detailed discussion of the difficulties of data translation in the NMR domain.

2. Challenges to Translation of NMR Data

Scientific data conversion is time-consuming with one effort reporting data conversion tasks as responsible for over 30% of tasks involved in a scientific workflow [6]. In addition, the explosive increase in open source software has also boosted the growth of tools used to process scientific data. The end result is that scientific data is generated and used in a wide range of different formats and it is frequently necessary to translate between the formats required for various tools.

One challenge to processing NMR data is posed by the data itself. There are a number of different NMR spectrometers and tools that produce and/or process data. The different ways in which these spectrometers and tools represent experiment data within files must be understood in order to allow data in one format to be translated into another format. In addition, NMR spectral data is quite complicated. NMR spectrometers generate two types of data, *core* data which is the actual data collected on nuclear induction decay, and *meta-data* which may either refer to information about experimental setup or to the layout and structure of the core data. The core binary files are difficult to understand without a complete description of the meaning in the file, a common problem found in scientific data [7]. Clearly, both the core data and the meta-data need to be transformed when used as input to a particular tool.

Logically, the core data can be viewed as sets of floating point, integer or short integer values in 1 to N spatial dimensions with a typical maximum of 4, depending on the type of experiment. The meta-data describe information necessary for interpreting core data. For example, meta-data tells us whether the core data represents time-domain or frequency-domain information, or whether the core data consists of real, complex, hyper-complex, or hyper-hyper-complex data points. The need to translate the combination of both core data and meta-data for a variety of tools and spectrometers complicates the data translation process.

The way that core data and meta-data are stored within a file presents another challenge to translating among formats. The core data generated by the spectrometer and used by tools are typically stored in one or more binary files as floating point, integer, or short integer values. Different formats arrange the experimental FID data differently, and the arrangement may vary depending on the dimension. For example, the NMRPipe format lists all the real components in the first dimension first and then lists all the imaginary. In subsequent dimensions however, the real and imaginary values are alternated. The Varian format, for indirect dimensions, iterates through all quadrature values first and then indexes spatially. Rowland NMR Toolkit alternates on every dimension. One of two formats supported by Bruker arranges the data in smaller sub-matrixes within a larger matrix.

To illustrate schematically, a fictitiously small data set is presented in Figure 2 below. The numerals represent the spatial indexes and the R (real) and I (imaginary) represent quadrature indexing. For example, in NMRPipe files, in the direct dimension, all the real numbers are clustered, followed by all the imaginaries. In Rowland NMR Toolkit files, the numbers alternate real, imaginary, real, imaginary, etc.

In addition to differences in data ordering, meta-data conventions vary among the formats. For example, some formats consider the direct dimension to be the lowest numbered and others consider it to be the highest. Different formats will imply different signs of the imaginary numbers when used in performing subsequent processing such as Fourier transforms.

Meta-data may be stored in the header of the binary files containing the core data or may be located in a separate ASCII text file (or both). The order in which the core data is organized and/or indexed spatially varies among tools. This ordering determines the orientation of the data and how it is stored in a file. In order to accurately translate between formats, this ordering and indexing must be modeled. In addition, the Endianness of floats and/or integers must be known and properly converted.

The formats generated by the various spectrometers also introduce complexity to the process of translating among formats. For instance, both Bruker and Varian place restrictions on the first dimension. The Varian spectrometer format requires that the first dimension be complex, not real only while the Bruker spectrometer only performs digital signal processing in the first dimension. In addition, formats generated by the Varian and Bruker spectrometers use the opposite sign convention for imaginaries.

The differences in the representation and meaning of meta-data pose another challenge to translating among data formats. Different spectrometers and tools have different purposes and therefore capture different information about the core data being processed. For example, meta-data generated by the Bruker spectrometer describes various acquisition modes and function modes using parameters that take one of a set of string values (e.g., function mode: {States, TTPI, States-TTPI, Echo-Antiecho}). In contrast, NMRPipe models these characteristics more directly via explicit flags, with a Boolean flag for each of the four function modes. This difference in representation must be accommodated in a spectrum translator.

3. Quality Factors for the CONNJUR-ST Architecture

When considering the CONNJUR-ST tool, there were several critical quality attributes that guided the selection of the architecture. A set of user characteristics defined the important capabilities of CONNJUR-ST. These characteristics include:

- Breadth of translation: The ability to handle translation among a wide variety of spectrometer output and NMR data processing tool formats. The translator should be able to handle the unique aspects of a variety of formats in a graceful manner while notifying the user of unexpected formatting issues.
- Heuristics to guide conversions: The translator should be able to understand and utilize the meta-data that accompanies a collection of data and to notify or prompt the user only when key portions of the meta-data are missing.
- Current capability: An ideal broad translation tool for NMR spectroscopy data should provide functionality equivalent to the existing tools that only perform a limited number of conversions. The transposition of data so that the ordering of dimensions within the data matches the input expected by other tools must also be supported. In order for a translation tool to be useful, it must provide at least the functionality that users are accustomed to when using other tools.
- Efficiency: The translation of the data must occur in a reasonable amount of time, seconds to a minute. When dealing with NMR experiments using three or four dimensions, the size of the data set may reach a gigabyte or more and a billion or more data points.

The CONNJUR-ST architecture selection was also driven by a set of development characteristics. These include:

- Modularity: In our case, modularity of the software is critical to supporting parallel development. Since membership in the CONNJUR group is fluid due to its academic make-up, modularity is also important as it improves understandability. This understandability allows us to integrate new developers into the project more easily.
- Extensibility: Since it is reasonable to assume that new file formats will continue to emerge as new NMR data processing tools are developed, an ideal format translator must be easily extensible. In the case of the CONNJUR group, CONNJUR-ST is a newly emerged open source project and extensibility would aid in building a developer base by allowing developers to easily add support for their desired formats.

- Separation of Concerns: This characteristic is necessary to allow modification of the reading of input files without impacting the writing of output files.

4. The CONNJUR-ST Architecture

The CONNJUR-ST architecture, shown in Figure 3, uses a variation of the Pipeline architectural style [8]. The boxes indicate the major software components in the architecture. The box labeled Spectral Object holds a representation of the spectrum that is common across all input and output file formats. The lines show how data moves within the architecture and the arrows indicate the direction of the data flow. In CONNJUR-ST, data in an input format is read into the system, converted into a Spectral Object and then transformed into the desired output format.

Our architecture abstracts the common features of NMR experiment data, supplying a representation of NMR data that supports a set of commonly used data formats. The architecture demonstrates high cohesion as each component is responsible for a unified task while also displaying low coupling.

The boxes in Figure 3 represent the components that process data. Data movement is indicated via the dashed arrows which are labeled with the data being passed. The file system is represented by the slanted rectangle and the Spectral Object is represented using a dashed outline as it is a component in the system that is mainly responsible for managing data.

The translation of a file from a specified input format to a specified output format takes two high-level steps. These steps are controlled by a manager process that governs the sequence of actions. First, the file must be read into CONNJUR-ST and converted to a Spectral Object which contains a model of the data. Second, the Spectral Object is then converted into the desired output format and the transformed file is written to the file system. Optional Semantic Operators take a Spectral Object and perform a permutation on the data held in the Spectral Object to return a new Spectral Object. The use of a common internal data model within the Spectral Object enables our desired modular development approach and reduces the complexity of the overall system, allowing new input and output formats to be added to CONNJUR-ST independently. We discuss the two translation steps in more detail below.

The Reader/Input Layout Generator obtains the meta-data for a spectrum from either the headers of one or more binary files on the file system or from one or more separate ASCII text files, or both. There are two types of meta-data: 1) experiment meta-data which is data that describes experimental aspects of the core data; and 2) file meta-data which describes how the data is organized within the file. The Reader/Input Layout Generator passes the experiment meta-data to the Spectral Object where it is interpreted and stored and eventually be used to generate the appropriate output format. The Reader/Input Layout Generator generates a description of the input file layout, Input Layout Descriptor, which is passed to the Binary File Reader which retrieves the relevant information about the file layout. The Binary File Reader reads the core data from the file and populates the Spectral Object. At this point, the Spectral Object contains all information necessary to support translation of the data into a new format.

In some cases, in order for the data to be translated properly, some form of mathematical processing must be applied to the data (e.g., negating imaginary numbers). The Semantic Operator takes the Spectral Object as input, applies the processing and returns a new Spectral Object. Two operators are currently provided. The Negate Imaginaries operator negates the imaginary value of specified dimensions. The Rance-Kay operator allows processing of data collected with the Rance-Kay sensitivity enhancement [9] in which the recorded quadrature fids are the sum and difference of the actual quadrature components.

The actions taken in the writing stage of the pipeline are analogous to the steps taken in the reading stage, however the actions taken are reversed in order. The Writer/Output Layout Generator uses the experiment meta-data in the Spectral Object to create an Output Layout Descriptor which contains a description of how the data will be organized in the output file. The Writer/Output Layout Generator retrieves the experiment meta-data from the Spectral Object and translates it into the format required by the desired output file. This information is stored in the Output Layout Descriptor which already contains the file meta-data. The Writer/Output Layout Generator is also responsible for writing to ASCII files if the output format requires meta-data to be stored in separate ASCII files. The Binary File Writer takes the Output Layout Descriptor and the Spectral Object and writes out the appropriate binary file(s) for the desired output format.

Example

Consider the example of a collaborating researcher who receives a Rowland NMR Toolkit formatted file and wishes to process it using Bruker's TopSpin software. The Rowland NMR Toolkit data will consist of a pair of files, e.g. input.secand input.par. Bruker data consists of a multiple file directory hierarchy, so the target of the output command will be a directory.

The following shell command will be executed:

```
connjurst -srcfile input.sec -srctype rowland -destdir bruker_data -desttype
bruker
```

The command line driver parses the arguments, verifies the presence of required elements and creates a Converter object. The Converterwill use factories and the source and destination strings to create subtype objects that implement a Rowland version of the Reader/Input Layout Generator component. The specialized Reader/Input Layout Generator creates an Input Layout Descriptor. As the meta-data is parsed, a Spectral Object is created and the dimension information set. The Spectral Object and Input Layout Descriptor are then passed to the Binary Data Reader, which populates the spectrum values portion in the Spectral Object.

The Spectral Object is next utilized by a Bruker version of the Writer/Output Layout Generator component which produces an Output Layout Descriptor which contains the name and location of the binary data files. The Binary Data Writer then writes the binary data to disk, completing the conversion.

5. Usefulness of the Architecture

The use of the central Spectral Object and a pipeline architecture greatly reduced the amount of development necessary to support the currently implemented conversions. Although twenty conversions are currently supported, only nine reader or writer implementations were necessary. In addition, all four readers user the same Binary File Reader and four of the five writers use the same The Binary Data Writer.

CONNJUR-ST was tested by doing parallel conversions with existing tools, where available and doing a byte-wise comparison of the data portions of the file(s). Where existing tools only perform a one way conversion, they were used to verify the inverse conversion by

CONNJUR-ST; e.g. if a tool only supports A to B conversion, CONNJUR-ST was used to convert B to A, the existing tool converted the output A to B′, and B′ was compared with B to ensure no loss of data integrity. Additionally, spectra were examined visually by both developers and expert users with software tools which support the converted format.

Currently CONNJUR-ST supports all conversions for the Spectral Reconstruction phase for Bruker and Varian NMR spectrometers, NMRPipe and RowandNMRToolKit processing tools, and for tabular ASCII data output. Transformation of data among most of the Spectrometer Acquisition and Spectral Reconstruction phases is also supported. It was decided not to perform the computation required to convert data from one phase to another as existing tools do this efficiently (e.g., between the Spectral Reconstruction and Spectral Analysis phases). However, conversion between tool formats within a phase is supported.

The CONNJUR-ST code base has been used extensively in an additional CONNJUR software product, the CONNJUR Workflow Builder. A subclass of the Spectral Object provides necessary indexing for storage of the data in a relational database. The CONNJUR-ST readers and writers are used to implement importing and exporter of data from the Workflow Builder in memory/database format to the format of external tools used by Workflow Builder.

The CONNJUR-ST architecture was designed to be sufficiently general to apply to a range of scientific data translations. Most scientific data contains both core data and meta data with the meta data including data about the format of the core data, and data about the tools or experiment that generated the data. For instance, the CONNJUR-ST architecture could easily be adapted to support data translation for geospatial applications such as maps. Map data is commonly stored in grids or vectors, similar to the quadratures used in CONNJUR-ST to hold NMR data. Geospatial meta-data includes information such as the resolution of the collection instrument (e.g., satellite), identification of any data compression algorithms used and whether the data compression is lossless or lossy. All of this latter meta data is necessary in order to properly convert data from the format of one tool to another.

The CONNJUR-ST implementation of the architecture presented above provides key structure and functionality that is shared across a variety of input to output file translations. The framework provides common components that contain the functionality shared by all subtypes. Individual subclasses may be created to support specific functionality for reading or writing a particular file format. Therefore the architecture supports separation of concerns and is extensible and modular, reducing effort for developers desiring to add new functionality to read from or write to a new file format. Experience so far has shown that the CONNJUR-ST architecture is able to support development in a very complex domain.

6. Status and Future Work

Development on CONNJUR-ST began in June 2009. During 2009–2010 we had two part-time developers working on the project and in August 2010, CONNJUR-ST V1.0 was presented at the 24th International Conference on Magnetic Resonance in Biological Systems (ICMRBS) to a very positive response. This initial version of CONNJUR-ST supported conversion of the time-domain data of the Spectral Reconstruction phase. In spring 2011, an updated version, CONNJUR-ST WithFrequency was released for publication in the Journal of Biomolecular NMR. The WithFrequency version of CONNJUR-ST added conversion of frequency-domain data of the Spectral Reconstruction phase.

Researchers at the Hoch Lab (developers of the Rowland NMR Toolkit) have successful used CONNJUR-ST to perform difficult conversions beyond the scope of the converter incorporated in the Rowland NMR Toolkit. A recent advance in NMR spectroscopy is collecting indirect dimensions with uneven spacing in the time delays of the pulse sequences. This non-uniform sampling reduces time requirements for collecting larger data sets but requires the use of more sophisticated algorithms for converting data from time to frequency domain. An additional meta-data text file is used to record information about the spacing between successive samples. CONNUR-ST Nonuniform added support for these formats and was presented at the High Throughput Structural Biology Keystone Symposium in January 2012. The process of developing CONNJUR-ST has reaped several benefits. The determination of the structure of meta-data and various files generated by NMR spectrometers as well as data processing tools has provided us with an in-depth understanding of NMR experiment data. This knowledge will benefit the further construction of the CONNJUR environment, easing development of the workflow framework. In addition, the process of designing CONNJUR-ST has provided us with a model and language for describing NMR experiment data and file formats. The process has helped us model the data which has additionally benefited the CONNJUR project as a whole.

CONNJUR-ST currently has a small user base which is providing us feedback on future directions for development. The CONNJUR-ST Nonuniform version is in maintenance and future plans are to add additional formats if sufficient user demand is present and resources are available.

Acknowledgments

This research was funded by U.S. National Institutes of Health grants EB-001496 and GM-083072. The authors wish to thank Dr. Frank Delaglio, Jeffrey Hoch, Mark W. Maciejewski and Alan Stern for useful conversations and assistance with NMRPipe and the Rowland NMR Toolkit file formats. The authors also wish to thank Agilent Technologies and Bruker Biospin for assistance with understanding the Varian and Bruker file formats.

References

1. Gryk MR, Vyas J, Maciejewski MW. Biomolecular NMR data analysis. Progress in Nuclear Magnetic Resonance Spectroscopy. May; 2010 56(4):329–345. [PubMed: 20625476]

2. Hoch, J.; Stern, A. The Rowland NMR Toolkit. 2010. http://rnmrtk.uchc.edu/rnmrtk/RNMRTK.html

3. Delaglio F, Grzesiek S, Vuister GW, Zhu G, Pfeifer J, Bax A. NMRPipe: a multidimensional spectral processing system based on UNIX pipes. J Biomol NMR. 1995; 6:277–293. [PubMed: 8520220]

4. Nowling RJ, Vyas J, Weatherby G, Ellis HJC, Fenwick M, Gryk M. CONNJUR Spectrum Translator: An open source application for reformatting time-domain NMR data. Journal of Biomolecular NMR. May; 2011 50(1):83–89. [PubMed: 21409563]

5. Nowling, RJ.; Vyas, J.; Weatherby, G.; Ellis, HGC.; Gryk, MR. CONNJUR Spectrum Translator: Open Source software for converting the format of time-domain NMR data. International Conference on Molecular Resonance for Biological Systems; Cairns, Australia. Aug. 2010;

6. Wassink, I.; van der Vet, PE.; van Dijk, EMAG.; van der Veer, GC.; Roos, M. New interactions with workflow systems. Proceedings of the ECCE '09 European Conference on Cognitive Ergonomics: Designing beyond the Product --- Understanding Activity and User Experience in Ubiquitous Environments; 2009.

7. Benger W. On Safari in the File Format Jungle: Why Can't You Visualize My Data? Computing in Science & Engineering. 2009; 11(6):98–102.

8. Shaw, M.; Garlan, D. Software Architecture: Perspectives on an Emerging Discipline. Prentice Hall; Apr. 1996

9. Rancea M, Loriab JP, Palmer AG. Sensitivity Improvement of Transverse Relaxation-Optimized Spectroscopy. Journal of Magnetic Resonance. Jan; 1999 136(1):92–101. [PubMed: 9887294]

Biographies

Heidi J. C. Ellis is Associate Professor and Chair of the Computer Science and Information Technology department at Western New England University. Dr. Ellis has an M.S. and Ph.D. from the University of Connecticut in Computer Science and Engineering. Her research interests are in software engineering education and she is currently participating in an NIH grant for developing database-driven software for biological NMR analysis. Dr. Ellis is a founding member of the Connecticut Joint University Research (CONNJUR) project. She is also a member of the IEEE Computer Society and has been involved in the development of the Society's Certified Software Development Professional exam.

Gerard Weatherby is a software developer with 20 years of experience. He holds a BS Physics from the Massachusetts Institute of Technology and an MS in Computer and Information Science from Rensselaer Polytechnic Institute. He is currently contracted as a consultant to Western New England University to support the Connecticut Joint University Research (CONNJUR) project.

Ronald J. Nowling is a Ph.D. student in Computer Science & Engineering at the University of Notre Dame. Mr. Nowling graduated from Eckerd College in 2010 with a B.S. in Computer Science and Mathematics and worked as an undergraduate research assistant at the University of Connecticut Health Center during the summers, four of which were spent with the Connecticut Joint University Research (CONNJUR) project, of 2005 to 2010. His research interests include algorithms and numerical methods, software engineering, programming languages, and compilers, particularly with applications to the biological sciences.

Jay Vyas has an M.S. in Computer Science from Rensselaer at Hartford and a Ph.D. in Molecular Biology and Biochemistry from the University of Connecticut. Dr. Vyas has been a member of the CONNJUR project since 2004 and has implemented several prototypes of the CONNJUR-WB tool.

Matt Fenwick is a graduate student in the Department of Molecular, Microbial, andStructural Biology, University of Connecticut Health Center. Mr. Fenwick has been a member of the CONNJUR project since 2009.

Michael R. Gryk is Associate Professor of Molecular, Microbial and Structural Biology at the University of Connecticut Health Center. Dr. Gryk has a B.S. in Biophysics and an M.S. in Chemistry from the University of Connecticut and a Ph.D. in Biophysics from Stanford. Dr. Gryk is a founding member of the Connecticut Joint University Research (CONNJUR) project. His research interests include structural biology of proteins using NMR and bioinformatics.

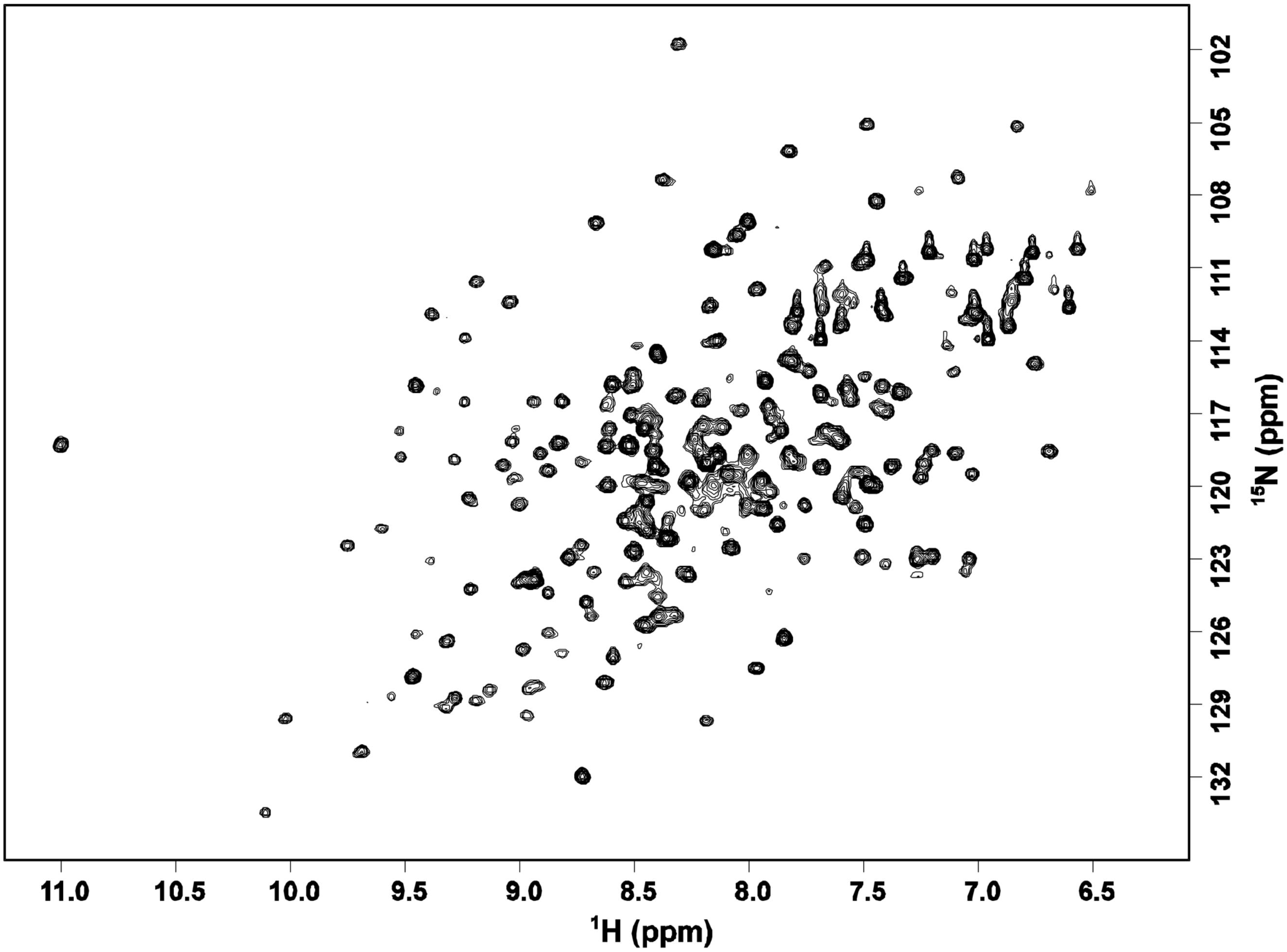
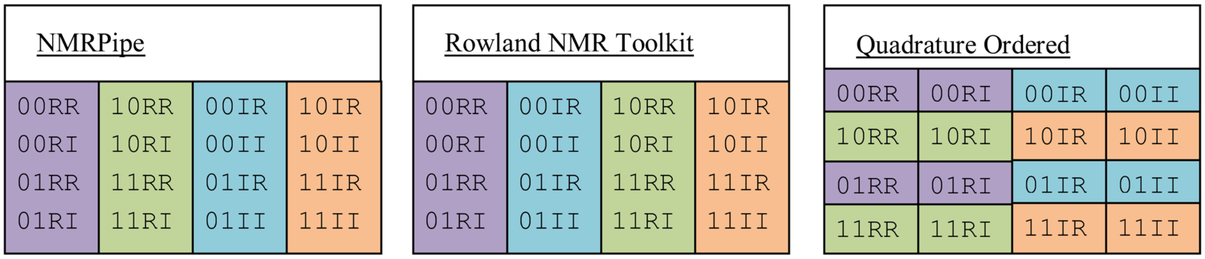


Figure 1.

Example Data for 2D-NMR experiment.



NMRPipe

00RR	10RR	00IR	10IR
00RI	10RI	00II	10II
01RR	11RR	01IR	11IR
01RI	11RI	01II	11II

Rowland NMR Toolkit

00RR	00IR	10RR	10IR
00RI	00II	10RI	10II
01RR	01IR	11RR	11IR
01RI	01II	11RI	11II

Quadrature Ordered

00RR	00RI	00IR	00II
10RR	10RI	10IR	10II
01RR	01RI	01IR	01II
11RR	11RI	11IR	11II

Figure 2.

Comparison of NMRPipe, Rowland NMR Toolkit and Quadrature Ordered Data.

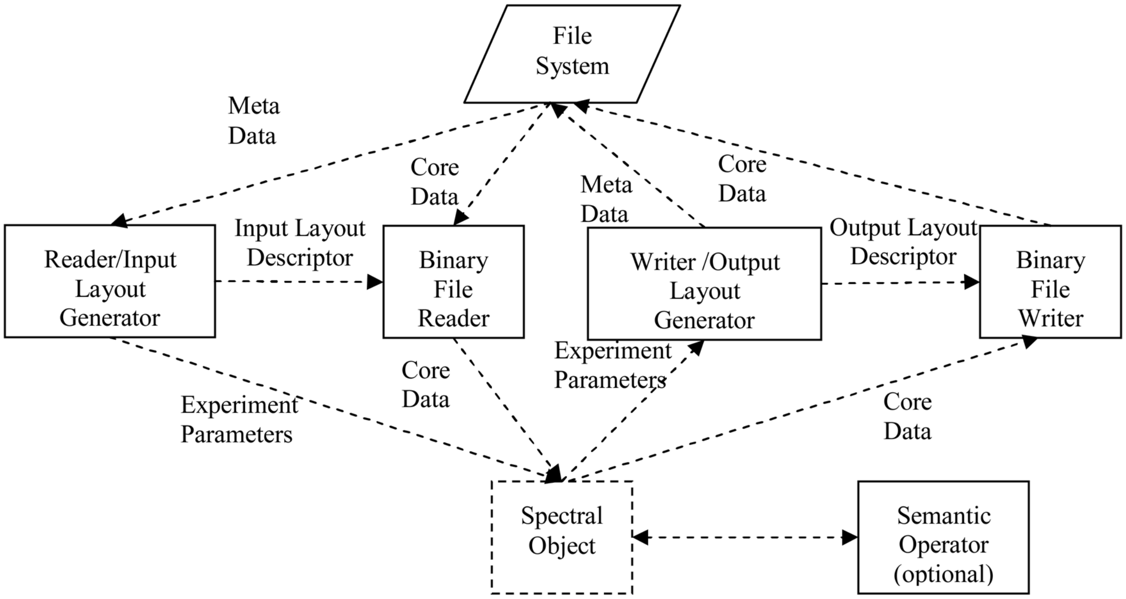


Figure 3.
CONNJUR-ST Architecture