FACETS

FP6-2004-IST-FETPI 15879
Fast Analog Computing with Emergent Transient States

`D11:`

Definition of data formats for WP3 results in the FACETS knowledge base

Report Version: 1.01

Report Preparation: Andrew Davison and Thierry Viéville

Classification: PU

Contract Start Date: 01/09/2005 Duration: 4 Years

Project Co-ordinator: Karlheinz Meier (Heidelberg)

Partners: U Bordeaux, CNRS (Gif-sur-Yvette, Marseille), U Debrecen, TU Dresden, U Freiburg, TU Graz, U Heidelberg, EPFL Lausanne, Funetics S.a.r.l., U London, U Plymouth, INRIA, KTH Stockholm

Information Society Technologies

Project funded by the European Community

under the "Information Society Technologies' Programme

DELIVERABLES TABLE

Project Number: FP6-2004-IST-FETPI 15879

Project Acronym: FACETS

Title: Fast Analog Computing with Emergent Transient States

Del. No.	Revision	Title	Type[1]	Classification[2]	Due Date	Issue Date
11	1.0	Definition of data formats for WP3 results in the FACETS knowledge base	R	PU	Month 12	12/10/2006
11	1.01	Definition of data formats for WP3 results in the FACETS knowledge base	R	PU	Month 12	13/10/2006

[1] *R: Report; D: Demonstrator; P: Prototype; O: Other – Specify in footnote*

[2] *PU: Public*
PP: Circulation within programme participants, including the Commission Services
RE: Restricted circulation list (specify in footnote), including the Commission Services
CO: Confidential, only for members of the consortium, including the Commission Services

DELIVERABLE SUMMARY SHEET

Project Number: FP6-2004-IST-FETPI 15879

Project Acronym: FACETS

Title: Fast Analog Computing with Emergent Transient States

Deliverable N^o: 11
Due date: Month 12
Delivery Date: 12/10/2006

Short Description:

For efficient sharing of data between experimental groups and between experimentalists and modellers within the FACETS project, a common data format is required for each data-type, together with tools for converting data between the common formats and the local formats used by individual groups.

This report concerns the common formats for the data obtained in Workpackage 3. There is some overlap with Workpackage 2 (electrophysiological recordings of membrane potentials and membrane currents), and here the same common format will be used in both workpackages.

The structure of this report is as follows:

1. Survey of data acquisition and analysis software currently used in FACETS, and of the data formats that can be read/written by that software.
2. Description of the common formats that have been chosen (based on the HDF5 standard).
3. Summary of the future actions that are required to fully implement use of the common formats.

Partners owning: CNRS(a) and INRIA

Partners contributed: CNRS(a), INRIA, ULON, EPFL(b), UD, CNRS(b), ALUF

Made available to: public

Contents

1

Introduction

Within the FACETS consortium, there are several groups carrying out biological experiments at the single-neuron level (Workpackage 2) and at the network level (Workpackage 3). These groups use different software packages for data acquisition and analysis, and different data file formats for saving the data. For efficient sharing of data between experimental groups and between experimentalists and modellers, a common data format is required for each data-type, together with tools for converting data between the common formats and the local formats used by individual groups.

This report concerns the common formats for the data obtained in Workpackage 3. There is some overlap with Workpackage 2 (electrophysiological recordings of membrane potentials and membrane currents), and here the same common format will be used in both workpackages.

The structure of this report is as follows:

1. Survey of data acquisition and analysis software currently used in FACETS, and of the data formats that can be read/written by that software.

2. Description of the common formats that have been chosen (based on the HDF5 standard).

3. Summary of the future actions that are required to fully implement use of the common formats.

This report is specific to Workpackage 3 data, but is also part of the wider program of integration and standardisation being carried out in Workpackage 8. There is some overlap between this report, deliverable D12, and the report on milestone M8.

2

Existing tools and data formats used in FACETS

2.1 Data analysis and acquisition software used within FACETS

We surveyed all the partners within FACETS which are part of either Workpackage 2 or Workpackage 3 or both. The survey results are summarised in the following table:

Group	**Data type**	**Acquisition software**	**Analysis software**
Partner 2 (UD)	Electrophysiology	Spike2	Spike2
	Optical imaging	VDAQ	IDL, OIZ, Matlab, WINMIX
Partner 5 (ALUF)	Electrophysiology	Spike2, MCS	Spike Tools, MEA Tools
Partner 6a (CNRS-UNIC)	Electrophysiology	Elphy	Elphy, Matlab, Python
	Optical imaging	Elphy	Elphy, Python
Partner 6b (CNRS-DyVA)	Electrophysiology	REX, Elphy	Matlab, Python
	Optical imaging	VDAQ	Matlab, Python
	Eye-movements	REX	Matlab, Python
Partner 8b (EPFL-LNM)	Electrophysiology	Igor	Matlab
Partner 11 (ULON)	Electrophysiology	Spike2	*(In-house software)*

We now provide relevant details about each of the software packages used in FACETS.

Spike2

Supplier:	Cambridge Electronic Design
URL:	http://www.ced.co.uk/
Operating system(s):	Microsoft Windows
Import formats:	—
Export formats:	text, spreadsheet

Widely-used proprietary data acquisition and analysis tool for electrophysiology data. User-contributed Matlab[1] and Python[2] libraries are available to read and write the file format.

[1] http://www.kcl.ac.uk/depsta/biomedical/cfnr/lidierth.html
[2] http://www.ced.co.uk/files/sonpy.tar.gz

Spike Tools

Supplier:	BCCN Freiburg/Multi Channel Systems GmbH
URL:	http://www.bccn.uni-freiburg.de/research/b5.php
Operating system(s):	Platform-independent
Import formats:	*in development*
Export formats:	*in development*

Open source analysis toolbox for ensemble spike and population activity recordings, being developed by Partner 5 (ALUF). The toolbox will accommodate import of multiple proprietary data formats, based on the Neuroshare Project (neuroshare.sourceforge.net).

Elphy

Supplier:	CNRS-UNIC
URL:	http://www.unic.cnrs-gif.fr/software/elphy_engl.htm
Operating system(s):	Microsoft Windows
Import formats:	DAC2 , Acquis1, Axon, (programmable)
Export formats:	Matlab, text, (programmable)

Freely-distributed[3] data acquisition and analysis tool, developed by Partner 6a (CNRS-UNIC).

MCS

Supplier:	Multichannel Systems
URL:	http://www.multichannelsystems.com/
Operating system(s):	Microsoft Windows
Import formats:	n/a
Export formats:	binary

Proprietary data acquisition and analysis software for micro-electrode array recordings. For analysis of MCS data, Partner 5 (ALUF) uses the Matlab-based open source MEA-Tools, developed at ALUF and available to the community at http://www.brainworks.uni-freiburg.de/projects/mea/meatools/overview.htm

REX

Supplier:	U.S. National Eye Institute
URL:	http://www.nei.nih.gov/intramural/software.asp
Operating system(s):	QNX
Import formats:	n/a
Export formats:	binary

REX is an acquisition program (also allowing some on-line analysis), created mainly for the eye movement community. It controls the experiment, acquires eye movement data and other data (spikes can be acquired in REX for example). It is a master program that controls behavioural experiments. Routines are available to import the binary output format in Matlab.

[3] except for visual stimulation module

VDAQ

Supplier:	Optical Imaging, Inc.
URL: http://www.opt-imaging.com/	
Operating system(s):	Microsoft Windows
Import formats:	n/a
Export formats:	Matlab

Data acquisition software for optical imaging experiments.

Igor

Supplier:	Wavemetrics
URL:	http://www.wavemetrics.com/products/igorpro/
Operating system(s):	Microsoft Windows, Mac OS, Mac OS X
Import formats:	text, general binary, Excel spreadsheet, HDF, HDF5, Matlab, JCAMP, Nicolet Instruments, SDTS DEM and DLG
Export formats:	(programmable)

"*interactive software environment for experimentation with scientific and engineering data and for the production of publication-quality graphs and page layouts.*"

IDL

Supplier:	ITT Visual Information Solutions
URL:	http://www.ittvis.com/idl/
Operating system(s):	Microsoft Windows, Mac OS X, UNIX/Linux
Import formats:	multiple, including HDF5
Export formats:	multiple, including HDF5

Note that the information above applies to version 6.3., and needs to be confirmed for version 5.3., used by Partner 2.

IDL is a software platform for data visualisation and analysis.

Matlab

Supplier:	The MathWorks
URL:	http://www.mathworks.com/products/matlab/
Operating system(s):	Microsoft Windows, Mac OS X, Unix/Linux
Import formats:	text, HDF5, (programmable)
Export formats:	text, HDF5, (programmable)

A proprietary high-level language and interactive environment for numerical computing, including data analysis.

Python

Supplier:	Python Software Foundation
URL:	http://www.python.org/
Operating system(s):	Microsoft Windows, Mac OS X, Unix/Linux
Import formats: text, HDF5, (programmable)	
Export formats: text, HDF5, (programmable)	

An open-source, interpreted, interactive, object-oriented, extensible programming language with multiple tools available for numerical computing, data analysis, image analysis, etc.

3

Common formats

3.1 File format

The requirements for a common file format to be used within FACETS are:

- Freely available and useable for further software development
- Widely used
- Efficient storage and I/O

The first requirement rules out any proprietary format, while the last requirement rules out plain ASCII text format. For electrophysiology data (voltage or current recordings, spike-times), we identified two candidate formats which meet all the requirements: the ROOT data format (http://root.cern.ch/) and the HDF5 data format (http://hdf.ncsa.uiuc.edu/HDF5/).[1]

ROOT is an object-oriented data-analysis framework, including a standard file format, developed at CERN and widely used in the high-energy physics community. HDF5 is '*a general purpose library and file format for storing scientific data*', developed by the U.S. National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign, and '*created to address the data management needs of scientists and engineers working in high performance, data intensive computing environments*'.

Both the ROOT and HDF5 formats allow for hierarchical structuring of datasets, sequential or non-sequential data access, efficient storage and I/O (transparent compression), and are designed for use in distributed (parallel) computing systems.

After reviewing the features of these two options, we chose to use the HDF5 format, since it relatively lightweight (being only a library and file-format, whereas ROOT is an entire data-analysis framework), and has been interfaced with and incorporated within a wider variety of tools (e.g. Matlab has built-in support for reading/writing HDF5 files).

3.2 Conversion between local and common file formats

It is essential to have a conversion pathway, as short as possible, in both directions between HDF5 and each of the formats used locally in FACETS. This indeed seems to be possible in principle, although not all the following pathways have yet been tested:

- Spike2 ↔ Matlab or Python ↔ HDF5 (using Matlab/Python SON libraries)
- SpikeTools ↔ HDF5 (*in development*)

[1] A further interesting project is NeuroShare (http://neuroshare.sourceforge.net/), an open-source project to '*support the collaborative development of open library and data file format specifications for neurophysiology*'. These efforts may be of interest in future, but appeared to be too complex for our current needs; additionally, the platform-independence of the libraries was not clear.

- Elphy $\rightarrow$ Matlab $\rightarrow$ HDF5 $\rightarrow$ ASCII text $\rightarrow$ Elphy
- MCS $\leftrightarrow$ binary $\leftrightarrow$ Matlab $\leftrightarrow$ HDF5
- REX $\leftrightarrow$ binary $\leftrightarrow$ Matlab $\leftrightarrow$ HDF5
- IDL $\leftrightarrow$ HDF5
- Igor $\leftrightarrow$ HDF5
- Matlab $\leftrightarrow$ HDF5 (Matlab version $\geq$ 6.5.1 required)
- Python $\leftrightarrow$ HDF5 (using the PyTables package)

3.3 Data representation

Having decided on a file format, the next step is to decide on a data representation, since it is possible to lay out a given data set in many ways within an HDF5 file. For details of the proposed data representations, see the report on Milestone 8: "Provide and evaluate concept of the FACETS electronic knowledge repository".

4

Next steps

Two important steps now remain:

1. fully test the conversion pathways outlined in §3.2, using actual data files produced in WP3 and WP2, to ensure that conversion between HDF5 and each local file format is possible, and that no corruption or change in numerical precision takes place.

2. expand the outline proposals for data representation given in report M8, to give precise specifications including the size of fields (number of bits to be used for each data type) and the layout of file header information. The proposed representations may evolve within this step, in order to obtain the best compromise between storage efficiency and ease of manipulation of the data files. Further versions of this document will be released to reflect any changes.

There may be some interaction between these two steps, since different programs implement different subsets of the HDF5 specification.