



Published in final edited form as:

Cytometry A. 2010 January ; 77(1): 97–100. doi:10.1002/cyto.a.20825.

Data File Standard for Flow Cytometry, Version FCS 3.1¹

Josef Spidlen,

Terry Fox Laboratory, BC Cancer Agency, Vancouver, BC, Canada

Wayne Moore,

Genetics Department, Stanford University School of Medicine, Stanford, CA, USA

David Parks,

Stanford Shared FACS Facility, Stanford University, Stanford, CA, USA

Michael Goldberg,

Becton Dickinson Biosciences, San Jose, CA, USA

Chris Bray,

Verity Software House, Topsham, ME, USA

Pierre Bierre,

Cytek Development, Fremont, CA, USA

Peter Gorombey,

Soft Flow Informatika, Burnsville, MN, USA

Bill Hyun,

Laboratory for Cell Analysis, Helen Diller Family Comprehensive Cancer Center, University of California, San Francisco, CA, USA

Mark Hubbard,

iCyt, Champaign, IL, USA

Simon Lange,

Partec GmbH, Görlitz, Germany

Ray Lefebvre,

Guava Technologies, Hayward, CA, USA

Robert Leif,

Newport Instruments, San Diego, CA, USA

David Novo,

De Novo Software, Los Angeles, CA, USA

Leo Ostruszka,

Accuri Cytometers, Ann Arbor, MI, USA

¹This work was supported by the Michael Smith Foundation for Health Research and by NIH grant 1 R01EB005034 from the National Institute of Biomedical Imaging and Bioengineering to RRB. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institute of Biomedical Imaging and Bioengineering or the National Institutes of Health.

*Corresponding Author: Ryan R. Brinkman, Terry Fox Laboratory, BC Cancer Research Centre, 675 West 10th Avenue, Vancouver, BC, V5Z 1L3, Canada, Phone: +1-604-675-8132, Fax: +1-604-877-0712, rbrinkman@bccrc.ca.

Authors' Disclosures of Potential Conflicts of Interest

Members of ISAC Data Standards Task Force (ISAC DSTF) include representatives of companies selling flow cytometry instrumentation and software as well as academic researchers. FCS 3.1 was developed in an open manner with full and equal participation and approval by all members. The ISAC DSTF membership is always open to participation to members of the ISAC community.

Adam Treister,
Treestar, Ltd., Ashland, OR, USA

James Wood,
Department of Cancer Biology, Wake Forest University School of Medicine, Winston-Salem, NC,
USA

Robert F. Murphy,
Carnegie Mellon University, Pittsburgh, PA, USA

Mario Roederer,
National Institutes of Health, Bethesda, MD, USA

Damir Sudar,
Lawrence Berkeley Laboratory, Berkeley, CA, USA

Robert Zigon, and
Beckman Coulter, Chaska, MN, USA

Ryan R. Brinkman*
Terry Fox Laboratory, BC Cancer Agency, Vancouver, BC, Canada

Abstract

The flow cytometry data file standard provides the specifications needed to completely describe flow cytometry data sets within the confines of the file containing the experimental data. In 1984, the first Flow Cytometry Standard format for data files was adopted as FCS 1.0. This standard was modified in 1990 as FCS 2.0 and again in 1997 as FCS 3.0. We report here on the next generation Flow Cytometry Standard data file format. FCS 3.1 is a minor revision based on suggested improvements from the community. The unchanged goal of the Standard is to provide a uniform file format that allows files created by one type of acquisition hardware and software to be analyzed by any other type.

The FCS 3.1 standard retains the basic FCS file structure and most features of previous versions of the standard. Changes included in FCS 3.1 address potential ambiguities in the previous versions and provide a more robust standard. The major changes include simplified support for international characters and improved support for storing compensation. The major additions are support for preferred display scale, a standardized way of capturing the sample volume, information about originality of the data file, and support for plate and well identification in high throughput, plate based experiments. Please see the normative version of the FCS 3.1 specification in supplementary material to this manuscript (or at <http://www.isac-net.org/> in the Current standards section) for a complete list of changes.

Keywords

Flow cytometry; FCS; data standard; file format; bioinformatics

Introduction

The goal of the Flow Cytometry Data File Standard is to facilitate the development of software for reading and writing flow cytometry data files in a standardized format. Application of a standard file format allows files created on one type of instrument to be read and analyzed by software implemented on a different computer. The original FCS standard was published in 1984 as FCS 1.0 (1) and amended in 1990 as FCS 2.0 (2) and again in 1997 as FCS 3.0 (3).

Over the past ten years, FCS 3.0 has served its purpose well, with only few minor update requests from the scientific community. To address these requests, the International Society for the Advancement of Cytometry Data Standards Task Force (ISAC DSTF) has developed a minor revision of the specification. Below, we summarize the major changes in FCS 3.1. The normative version of the FCS 3.1 specification can be found in supplementary material to this manuscript and at the ISAC website in the Current standards section (4). Additional supplementary material to this manuscript contains examples of data transformations all the way from channel values in the FCS data file to the computer display of the end user. This document can be used as tutorial guiding software developers through some of the new features of the FCS 3.1 specification.

Summary of Changes

The changes in FCS 3.1 include suggested improvements from the community, addressing some potential ambiguities in the previous versions and to provide a more robust standard. Below, we summarize the changes between FCS 3.0 and FCS 3.1 data file standard.

Improved Support for Storing Compensation

Most multi-color fluorescent data requires compensation to map from measurement space to dye space. Compensation is accomplished by linear algebra; for each event, a vector of the relevant measurements is multiplied by the compensation matrix to give a vector of the corresponding dye quantities. The compensation matrix is the inverse of the spillover matrix.

Many users apply compensation at the time of data acquisition. However, most acquisition software packages now store the data uncompensated to provide the most flexibility in storage and retrieval of data. The compensation transformation can theoretically be recomputed at the time of analysis, given the same control samples. However, it is far more efficient for the acquisition software to describe the transformation in the FCS header segment so that the exact same transformation can be implemented by the analysis software.

Historically, there were two methods for the specification of this compensation. With FCS 2.0, the transformation could be completely and uniquely specified by the \$DFC*i*TO*j* set of keywords, with one keyword for every element in the spillover matrix. With FCS 3.0, these keywords were eliminated and replaced by the \$COMP keyword. Unfortunately, the \$COMP keyword was inadequately specified, and cannot uniquely specify the compensation transformation under many situations.

Therefore, the FCS 3.1 standard remedies this situation with the \$SPILLOVER keyword. The \$SPILLOVER keyword specifies the number of parameters included in the transformation, which parameters are to be included, and the spillover coefficient matrix. In FCS 3.1, the \$SPILLOVER keyword is the only standardized way to specify compensation.

It is conceivable that multiple transformation matrices might be desired in a single file (each of which would address non-overlapping sets of parameters). For example, if both area and height parameters were collected, these would require distinct spillover coefficients. However, since the parameter set is non-overlapping, the two matrices could be merged into a single matrix addressing all parameters (and with zero spillover values between the non-overlapping parameter sets). At this time, there is no justification for requiring distinct spillover matrices operating on shared parameters; therefore, there will be no mechanism for providing more than one \$SPILLOVER matrix per dataset.

Preferred Display Scale

Many acquisition software packages now store data as high-resolution linear data (e.g., 18 bit integer or IEEE floating point). However, users often wish to display the data in a transformed mode, for example, logarithmically-scaled with 4 decades of display. The transformation is often different for different parameters; for example, forward scatter is usually displayed as linear, immunofluorescence channels as logarithmic, and DNA fluorescence channels as linear.

Most often, the users have already defined their preference for visualization of each parameter at the time of acquisition. Rather than expecting the users to re-define these preferences at the time of analysis, or having the analysis software “guess” at what should be done, the FCS 3.1 standard defines the optional \$PnD keyword that identifies the user's preferred display scaling for each parameter. Analysis software should interpret this keyword value as a hint or preference.

Simplified Support for International Characters

FCS 3.1 now provides uniform Unicode (5) support for all keyword values. All keyword values are encoded in the Universal Character Set (UCS), also known as Unicode, as defined in ISO 10646 at implementation level 3 in UTF-8 (UTF = UCS Transformation Format) encoding defined in ISO 10646-1:2000. The \$UNICODE keyword is no longer necessary, and is discontinued as a valid FCS keyword.

UTF-8 is backward compatible with ASCII since all characters 00 through 7F (hex) inclusive are encoded the same way in UTF-8 and ASCII. Therefore, no change is required in existing software if it chooses not to support international characters. Moreover, UTF-8 support is included in many programming languages in a transparent way. Therefore, it is to be expected that many software tools will gain the international character support without much effort required from the software developers.

Calibration

Calibration of parameter values to well defined units such as mean equivalent soluble fluorochrome (MESF) or antibody molecules represents an area of current interest and research. Therefore, adding the calibration information will preempt the creation of multiple non-standard ways of handling calibration and is expected to be a significant advancement to the field. In the FCS 3.1, calibration has been included using the set of optional \$PnCALIBRATION keywords in a way that does not complicate or interfere with existing and new uses of FCS 3.1. It can be considered as hint or user preference, not a mandated part of the specification.

Support for Plate and Well Identification

The \$PLATEID, \$PLATENAME, and \$WELLID keywords have been added to systematically address the need of sample identification in high throughput, plate based experiments. These keywords can be used in addition to the existing \$SRC and \$SMNO keywords.

Sample Volume

A standardized way to communicate the sample volume has been introduced. The value of a new optional \$VOL keywords is a floating point number expressing the sample volume in nanoliters.

Originality of the Data

ISAC is considering mechanisms to enforce immutability of data sets in the future, which is a long term goal along with separation of data and metadata (i.e., data about data) information. While altering data is not being encouraged, there may be use cases where modifications of FCS files are essential (e.g., adding computed parameters, adding analytical information, etc.) These shall however be stored in a new copy of the data file, always keeping the original file untouched.

As a response to the need of clearly distinguishing between an original data set (as acquired by the instrument) and an altered copy of this data set, the optional \$ORIGINALITY keyword has been added to FCS3.1. It allows specifying whether the file is original or altered, distinguishing the case where the DATA segment has been modified from the case where only meta data have been touched, such as adding new keywords (e.g., \$SPILLOVER) or adding the ANALYSIS segment. In addition, the optional \$LAST_MODIFIER keyword may specify the name of the person performing the last modification, and the optional \$LAST_MODIFIED keyword may store the date and time of that modification. This mechanism is mostly intended to prevent accidentally mixing original and derived data sets and is not meant to replace additional originality certification mechanisms that may be in place in particular (e.g., clinical) environments.

Parameter Names

FCS3.0 and earlier versions required specified parameter names (values of the required \$PnN keyword) to be used for certain measured parameters. For example, these included FS for forward scatter, SS for side scatter, etc. Historically, these requirements helped with the establishment of useful naming conventions; however, they have never been followed completely. For example, the forward scatter parameter name would vary between FS and FSC and a suffix “-A”, “-H”, or “-W” would commonly be added to indicate whether the area, height or width of the signal pulse is being captured.

Reflecting current practices, the requirements of certain parameter names for certain parameters have been removed in FCS3.1 except for “TIME” being required as the value of \$PnN keyword for the time parameter. This “TIME” name requirement has been kept since it provides the only identification of the parameter related to the \$TIMESTEP keyword.

Deprecation of Obsolete Features

Historically, the FCS file format has supported certain features that are not considered useful from contemporary perspective. Some of these features are being deprecated in FCS3.1, which means, that users (i.e., software and hardware developers) are still allowed to use these in FCS3.1; however, their usage is not recommended since they may not be preserved in future versions of the FCS data file standard.

Specifically, the FCS3.1 specification deprecates the ASCII data type (\$DATATYPE = ‘A’). This data type has been useful at the early era of FCS for debugging purposes and it is generally not being used or supported by current instruments or analytical software.

The use of gating parameters (\$GnE, \$GnF, \$GnN, \$GnP, \$GnR, \$GnS, \$GnT, and \$GnV keywords) is deprecated in FCS3.1. However, this does not mean that acquisition gates are deprecated in FCS 3.1. The intent of \$Gn keywords was to store information about gating parameters, which may not have been included in the final data set. From the contemporary perspective, this is not a recommended practice and gates are applied on “regular” (i.e., \$Pn) parameters either during data acquisition (supported via the \$GATING, \$RnI and \$RnW keywords) or, more frequently, during the post-acquisition analytical process. These gates

are typically saved and communicated as part of a project (or workspace) definition file by the software application used to perform the analysis. Alternatively, these gates can be exported and communicated in the Gating-ML standard (6), which is increasingly being supported by third party software applications.

The use of histograms (\$MODE other than 'L') and the related \$PKn, \$PKNn keywords are deprecated in FCS3.1. Implementors are encouraged to avoid histograms since they may be discontinued in next versions of FCS. Moreover, storing and providing raw list mode data is one of the key aspects of transparent and reproducible data analysis.

The use of multiple data sets within a single data file is also deprecated, unless these are derived from each other. While multiple data sets within a single data file may seem a convenient feature, it has led to alternative and conflicting implementations rather than providing useful aspects.

Introduced Restrictions

There are two new restrictions in the FCS3.1. The values of the \$BYTEORD keyword have been restricted to either "1,2,3,4" (little endian) or "4,3,2,1" (big endian). The series of the PDP-11 computers manufactured in 1970s and 1980s was the only widely used platform with an unusual byte order: "3,4,1,2" meaning that in the two 16-bit words comprising a 32-bit word, the most significant 16-bit word was written first; however, within the 16-bit word, the least significant byte was written first. While previous versions of FCS allowed for any byte order to be specified, mostly in order to support the PDP-11 platform, the little and big endian are currently the only byte orders being effectively used and supported by analytical hardware and software.

The second restriction affects floating point data. If the floating point data type is used (either \$DATATYPE = 'F' or \$DATATYPE = 'D'), then all parameters shall be stored as linear with the value of \$PnE equal to '0,0'. The combination of logarithmic scale with floating point data brings confusion rather than any benefit. Please note that storing data on logarithmic scale (values of \$PnE other than '0,0') is still supported for the integer data type.

Improved Documentation

In several sections, we have improved documentation of the file format. Improvements include more explicit description of several keywords, such as \$PnE. In previous versions of FCS, insufficient description of this keyword caused developers to use a "4,0" value, which, strictly speaking, is not a valid entry. In addition, we are specific in several details that have previously been considered as "implicitly clear", e.g., last byte of a segment being the end of the segment or dot ('.') being the decimal separator of an ASCII representation of a floating point number.

Conclusion

FCS 3.1 represents a minor revision of the successful flow cytometry standard file format. This revision includes features and improvements requested by the community. While FCS continues to be the main venue for raw flow cytometry data and metadata describing the acquisition conditions, additional metadata are required to unambiguously describe experiments and analyses as specified by the Minimum Information about a Flow Cytometry Experiment (MIFlowCyt) guidelines (7). Since these details are typically available long time after the acquisition of the raw data, it is preferential to avoid storing them directly in FCS files. Ensuring that the FCS files are immutable facilitates reproducible research as well as typical clinical use cases. Moreover, data structures capturing experimental metadata and analysis require a high degree of flexibility, which would not be transparent if encoded in

FCS. A solution is to move these components out of the data file into more appropriate data formats based on up-to-date technologies, such as the Extensible Markup Language (XML). The Gating-ML (6) file format used to communicate post acquisition gating represents an example of such a component. To simplify the organization of multiple files, the Analytical Cytometry Standard (ACS) is being designed as a container file format (essentially a ZIP file). Currently, the ISAC DSTF is focused on the development of a “Table of Contents” (i.e., a manifest or an index) that would describe the contents of the ACS container. Typically, FCS represents the primary source of data in these containers and additional information can be provided in a flexible manner. While we are also reviewing additional data file formats for their potential to accommodate future instruments (e.g., integrate list mode data with spectral, waveform or image data), it should be stressed that ISAC is not considering retiring FCS, which will likely be the most important file format in flow cytometry for the next generations of cytometers.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

RRB is an ISAC Scholar. The authors would like to thank for comments on FCS 3.1 draft received from members of the ISAC wide membership during the open commentary period.

References

1. Murphy RF, Chused TM. A proposal for a flow cytometric data file standard. *Cytometry*. 1984; 5:553–555. [PubMed: 6489069]
2. Dean PN, Bagwell CB, Lindmo T, Murphy RF, Salzman GC. Data File Standard for Flow Cytometry. *Cytometry*. 1990; 11:323–332. [PubMed: 2340769]
3. Seamer LC, Bagwell CB, Barden L, Redelman D, Salzman GC, Wood JC, Murphy RF. Proposed new data file standard for flow cytometry, version FCS 3.0. *Cytometry*. 1997; 28:118–122. [PubMed: 9181300]
4. International Society for Advancement of Cytometry. Data File Standard for Flow Cytometry, Version FCS 3.1 - Normative Reference. August 13, 2009 Available at http://www.isac-net.org/images/stories/documents/Standards/fcs3.1_normativespecification_20090813.pdf
5. The Unicode Consortium. The UNICODE Standard, Version 1.0. Vol. 1. Reading, MA: Addison-Wesley Publishing Co. Inc.; 1991.
6. Spidlen J, Leif RC, Moore W, Roederer M, Brinkman RR. Gating-ML: XML-based Gating Descriptions in Flow Cytometry. *Cytometry A*. 2008; 73A(12):1151–1157. [PubMed: 18773465]
7. Lee JA, Spidlen J, Boyce K, Cai J, Crosbie N, Dalphin M, Furlong J, Gasparetto M, Goldberg M, Goralczyk EM, Hyun B, Jansen K, Kollmann T, Kong M, Leif R, McWeeney S, Moloshok TD, Moore W, Nolan G, Nolan J, Nikolich-Zugich J, Parrish D, Purcell B, Qian Y, Selvaraj B, Smith C, Tchuvatkina O, Wertheimer A, Wilkinson P, Wilson C, Wood J, Zigon R, International Society for Advancement of Cytometry Data Standards Task Force. Scheuermann RH, Brinkman RR. MIFlowCyt: the Minimum Information about a Flow Cytometry Experiment. *Cytometry A*. 2008; 73A(10):926–930. [PubMed: 18752282]