International Society for Advancement of Cytometry

Classification Results File Format CLR

International Society for Advancement of Cytometry Candidate Recommendation

Document Status

This document is an ISAC Candidate Recommendation. Features and design aspects specified in this document may be changed in the final version of the Recommendation. The CLR file format has not changed since CLR 1.0 version 110318, only the format documentation and examples have been improved.

Disclaimer of Liability

The International Society for Advancement of Cytometry (ISAC) disclaims liability for any injury, harm, or other damage of any nature whatsoever, to persons or property, whether direct, indirect, consequential or compensatory, directly or indirectly resulting from publication, use of, or reliance on this Specification, and users of this Specification, as a condition of use, forever release ISAC from such liability and waive all claims against ISAC that may in any manner arise out of such liability. ISAC further disclaims all warranties, whether express, implied or statutory, and makes no assurances as to the accuracy or completeness of any information published in the Specification.

In issuing and making this Specification available, ISAC is not undertaking to render professional or other services for or on behalf of any person or entity, nor is ISAC undertaking to perform any duty owed by any person or entity to someone else. Anyone using this document should rely on his or her own independent judgment or, as appropriate, seek the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Attention is called to the possibility that implementation of this Specification may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. ISAC shall not be responsible for identifying patents or patent applications for which a license may be required to implement an ISAC standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

This work is supported by NIH/NIBIB supplemental award to grant no 1R01EB008400.

CLR 1.0, ISAC Candidate Recommendation, version 140903, September 3, 2014.

Abstract

Gating in flow cytometry is an important process for sorting and selecting populations of interests for further data acquisition and analysis. Traditionally, manual gating has been the core of the analysis supported by virtually all flow cytometry analysis software applications. Consequently, a standard way of exchanging unambiguous descriptions of gates became crucial for interoperability among these applications. This need was addressed by the Gating-ML specification that allows forming unambiguous gate definitions based on their boundaries in multidimensional space.

Recently, the increased amount of high-throughput and high-content flow cytometry data motivated the development of various automated methods to supplement manual gating. Similarly to manual gating, the results of these methods are often per-event-based classifications, potentially with soft classifications expressed as the probability of an event being a member of a class.

The Classification Results File Format has been developed to exchange the results of manual gating and algorithmic classification approaches in a standard way in order to be able to report and process the classification. Simplicity of the format and its compatibility with common spreadsheet tools have been the major requirements driving the design of the specification. Although it was originally designed for the field of flow cytometry, it is applicable in any domain that needs to capture either soft or unambiguous classifications of virtually any kinds of objects.

Keywords: classification, cytometry, analytical cytology, file format, data standard

TABLE OF CONTENTS

Classification Results File Format CLR

1. Overview

1.1 Introduction

Traditionally, manual gating has been the core of flow cytometry data analysis that is supported by virtually all flow cytometry analytical software. With manual gating, boundaries are drawn to select populations of interest. A standard way of exchanging unambiguous descriptions of these manually created gates became crucial for interoperability among flow cytometry software. The Gating-ML [1] specification was developed to allow for unambiguous gate definitions based on population boundaries in multidimensional space.

Recently, the increased amount of high-throughput and high-content flow cytometry data motivated the development of various automated methods to supplement manual gating. Similarly to manual gating, the results of these methods are often per-event-based classifications, potentially with soft classifications expressed as the probability of an event being a member of a class. Consequently, Gating-ML is not suitable to capture these algorithmically classified events.

This document describes the Classification Results File Format that has been developed to address the new requirement of providing standard means for the computational exchange of the results of classification, both manual and automated. This format has been developed to be simple to process by any software application written in any programming language as well as to be editable by humans using common spreadsheet programs such as the Open Office Calc or Microsoft Excel. Consequently, the format has not been optimized for performance or for storage requirements. The file format is expected to be used for the exchange of classification results only, and various software tools may transform these into internal structures optimized for their own purposes.

The Classification Results File Format captures the results of event-based classifications only. Additional file formats or additional measures are often required to describe the workflow that lead to the creation of these results. The Archival Cytometry Standard (ACS [2]) specification is being designed to address this issue (see section 1.2 and Annex A for more details).

While the Classification Results File Format has been developed to address event-based classification in the field of flow cytometry, it is generally applicable in any biological and non-biological domain that needs to capture either soft or unambiguous classifications of any objects.

1.2 Scope

This document provides detailed specifications of the Classification Results (CLR) File Format. This file format may be used for the electronic exchange of assignments of events (or any objects) to a predefined set of classes. Both, overlapping and non-overlapping classes are supported. In addition, soft classification is supported in that events (objects) may be assigned to classes with specified probability values. Finally, the format accounts for cases where the event classification is unknown or where the probability of an event being a member of a class is unknown.

The CLR File Format is based on the CSV [3] file format. It introduces specific restrictions on the CSV file format and adds specific semantics so that compliant CSV spreadsheets can be interpreted as classification results files. Consequently, each valid CLR file is also a valid CSV file and therefore, it is editable using a common spreadsheet program, such as the Open Office Calc or Microsoft Excel.

The Classification Results File Format captures the results of event-based classifications only. Additional file formats or additional measures are required to describe the workflow that lead to the creation of these results. For example, it is often essential to capture the link between the results file format and the original (e.g., list-mode) data that have being classified. In addition, one may want to capture custom metadata providing information about the classification (e.g., who did the gating, what clustering algorithm has been used, what were the exact input parameters for that particular clustering algorithm, etc.). Software tools may, for example, store all this information internally in some form of a database. However, since the classification results cannot be interpreted without context, they should be accompanied by appropriate metadata when being exchanged or published for third parties to use. For example, the classification results may be bundled with related data and additional information using the Archival Cytometry Standard (ACS [2]) specification, in which case the files will be compressed effectively and the storage requirements will be minimized. See Annex A for examples. ACS will be the ISAC-recommended bundling mechanism for Classification Results once the ACS specification is finalized.

1.3 Purpose

The purpose of this document is to provide standard means for the electronic exchange of classification results of events (or any objects) into a predefined set of classes where each event (or each object) may be a member of one or more classes with a certain probability. This format has been developed to facilitate interoperability among software tools for algorithmic event classification in flow cytometry; however, it may be used for the electronic exchange of classification results in virtually any domain.

1.4 Normative References

The following referenced documents are indispensable for the application of this standard. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

— Request for Comments: 4180: Common Format and MIME Type for Comma-Separated Values (CSV) Files [3].

— Request for Comments: 20: ASCII format for Network Interchange [4]

— Request for Comments: 3629: UTF-8, a Transformation Format of ISO 10646 [5].

The following documents are useful for the application of this standard. They represent other standards and standard proposals relevant for understanding of this specification and/or relevant for the application of this standard, especially in the field of cytometry.

— ISAC Candidate Recommendation DRAFT, Archival Cytometry Standard [2].

— Spidlen J, Moore W, Parks D, Goldberg M, Bray C, Bierre P, Gorombey P, Hyun B, Hubbard M, Lange S, Lefebvre R, Leif R, Novo D, Ostruszka L, Treister A, Wood J, Murphy RF, Roederer M, Sudar D, Zigon R, Brinkman RR. Data File Standard for Flow Cytometry, version FCS 3.1 [6].

— Spidlen J, Leif RC, Moore W, Roederer M, Brinkman RR. Gating-ML: XML-based Gating Descriptions in Flow Cytometry [1].

1.5 The Content of this Specification

This specification consists of the following parts:

a) Normative: This document providing a detailed description of the CLR specification.

b) Informative: Examples of CLR files.

c) Informative: Examples of flow cytometry data and results of their classification in CLR files bundled together using the ACS [2] specification.

All the components of this standard are available from World Wide Web at http://www.isac-net.org/. The specification may also be downloaded from http://flowcyt.sf.net/.

1.6 Acronyms and Abbreviations

ACS	Archival Cytometry Standard
ASCII	American Standard Code for Information Interchange
CLR	Classification Results (File Format)
CR	Carriage Return (ASCII character code 0D hex)
CSV	Comma-Separated Values
DSTF	Data Standards Task Force
FCS	Flow Cytometry Standard
IEEE	Institute of Electrical and Electronics Engineers
ISAC	International Society for Advancement of Cytometry
LF	Line Feed (ASCII character code 0A hex)
RFC	Request for Comments
UTF-8	Unicode Transformation Format (8-Bit)

1.7 Keywords Indicating Requirement Levels

The key words *shall*, *should*, and *may* in this document are to be interpreted as described in RFC 2119 [7] and are also compatible with the IEEE Standards Style Manual. The word *shall* is used to indicate mandatory requirements to be followed in order to conform to the standard and from which no deviation is permitted (*shall* equals *is required to*). The word *should* is used to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain course of action is deprecated but not prohibited (*should* equals *is recommended that*). The word *may* is used to indicate a course of action permissible within the limits of the standard (*may* equals *is permitted to*).

2. Conformance

2.1 File Conformance

To be conformant with this standard, a CLR file shall be named with the `.csv` file name extension and its format shall meet requirements stated in section 3.

2.2 Software and Hardware Conformance

To be compliant with this standard, a software application or hardware instrument shall be able to "read", "write", or "read and write" the Classification Results (CLR) files. When CLR are produced (written) then these shall be valid CLR files according to section 2.1. When CLR files are read then the software application (or the hardware instrument) shall be able to read any CLR file that is valid according to section 2.1.

3. CLR File Format

3.1 General Structure

The CLR file format shall be a valid CSV spreadsheet file as specified by RFC: 4180 [3]. In the CSV spreadsheet, columns shall correspond to classes, column headings to class names, and rows to events (or other objects that were classified). The data in the spreadsheet shall express the probability of the particular event (or object) being a member of the particular class.

3.2 Class Names

The names of the classes that events (or objects) are assigned to shall be stated as column headings in the first row of the CLR file. UTF-8 [5] encoding shall be used if characters outside of the standard ASCII [4] character set are required (i.e., if international characters are part of the class names). Class names shall be unique within a single CLR file. Class names such as *Class 1*, *Class 2*, etc. should be used if there are no known meaningful class names.

If line breaks, double quotes, or commas are part of a class name then these shall be handled according to CSV specification [3]. Specifically, class names containing line breaks, double quotes, or commas shall be enclosed in double-quotes. If double-quotes are used to enclose a class name, then a double-quote appearing inside the class name shall be escaped by preceding it with another double quote.

3.3 Order of Classified Objects

The rows in the CSV spreadsheet shall correspond to objects being classified. The order of the objects shall be maintained as in the original data containing these objects. For example, if flow cytometry events from an FCS [6] file are being classified, then the events in the CLR file shall be described in the same order as they appear in the FCS file used for the classification. Consequently, let n be the number of events in the FCS file, then there shall be n+1 rows in the CLR file (one for the heading, n for the event classification).

3.4 Line Endings

The CSV specification [3] requires that line endings be encoded as a sequence of two characters: CR (Carriage Return, ASCII code 0D hex) and LF (/Line Feed, ASCII code 0A hex). Therefore, in order to be fully compliant, line endings in a CLR file should be encoded as a sequence of these two characters. This is also a common practice of encoding line endings on MS DOS and MS Windows platforms.

Unfortunately, Unix-like operating systems use only a single LF character to encode line endings in text based files and, furthermore, some spreadsheet tools use the platform-specific line endings when saving CSV files. In order to increase the interoperability, software applications reading CLR files should be able to process CLR files with line endings encoded either as a sequence of CR, LF characters or as a single LF character only.

3.5 Class Assignments

Starting on the second row (i.e., after the headings), the fields (cell values) in the spreadsheet shall express the probability of the particular event (or object) being a member of the class stated in the appropriate column heading. Let c_1, c_2, …, c_k be the classes names stated in the header of the CLR file and let e_1, e_2, …, e_n be the events (or objects) as in the original dataset used to perform classification. Then, the field $f_{i+1,j}$ ($i,j \in \mathbf{N}$, $i \in [1, n]$, $j \in [1, k]$) in the CLR file (i.e., the field in the row i+1-th row and j-th column) shall express the probability that the event (or object) e_i is a member of class c_j.

3.5.1 Definite Class Assignments

The value zero (0) shall be used in order to state that a specific event (or object) is not a member of a specific class according to these classification results. The value shall be ASCII-encoded, i.e., the character ASCII code 30 hex shall be used. The value one (1) shall be used in order to state that a specific event (or object) is a member of a specific class according to these classification results. The value shall be ASCII-encoded, i.e., the character ASCII code 31 hex shall be used.

3.5.2 Class Probability and Fractional Membership Assignments

Any floating-point number from the interval [0, 1] may be used to express that an event (or object) is a member of a specific class with a specified probability. Specifically, the value v, $v \in \mathbf{R}$, $v \in [0, 1]$, may be used to state that a specific event is a member of a specific class with a probability v. Similarly, a floating-point number from the interval [0, 1] may also be used to express fractional class membership, i.e., if an event is only partially member of a defined class. This concept may, for example, be useful to capture transitional stages during cell differentiation, oncogenic transformation etc.

The value v shall be encoded in ASCII with the point character (ASCII code 2E hex) used as a decimal separator. No other separators shall be used. A leading zero may or may not be used (i.e., the value may start with the decimal separator). There shall be no white space characters in the ASCII representation of the value.

The value v may be expressed using so called *E notation* [8]. In this form of scientific notation, values are expressed in the form of aEb, where $a \in \mathbf{R}$ is any real number, $b \in \mathbf{N}$ is an integer, and the construct shall be interpreted as a times ten to the power of b. Either the character E (ASCII code 45 hex) or the character e (ASCII code 65 hex) may be used to separate the coefficient a from the exponent b. Both a and b shall be encoded in ASCII with the point character used as a decimal separator. No other separators, neither white space characters shall be used.

Consequently, only the following ASCII characters may be used to encode v in a CLR file: “-”, “.”, “E”, “e”, “0”, “1”, “2”, “3”, “4”, “5”, “6”, “7”, “8”, “9”.

3.5.3 Unknown Class Assignments

An empty value shall be used to state that the probability of an event (or an object) being a member of a specific class is not known according to these classification results. An empty value shall be separated from preceding and/or following values by a comma.

The CLR format does not distinguish between events with unknown class labels and events with unintentionally unassigned class labels. The former may indicate that there is not enough information to classify an event, while the latter typically indicates an outlying event that is being ignored in the classification. It is recommended to define a separate "outlier" class in applications where such a distinction is required.

3.6 Examples

3.6.1 Definite Example with Distinct Classes

A listing of a CLR file with definite classification results of 5 events into 3 distinct classes (T cell, B cell, and NK cell class) is shown below. In this example, the first and the last events are T cells, the second one is a B cell, the third one is an NK cell, and the fourth one is not classified as any of the cell types.

```
T cell,B cell,NK cell
1,0,0
0,1,0
0,0,1
0,0,0
1,0,0
```

3.6.2 Definite Example with Overlapping Classes

A listing of a CLR file with definite classification results of 4 events into 3 overlapping classes (Lymphocyte, T cell, and B cell class) is shown below. In this example, all events are classified as lymphocytes and in addition, the second event is also classified as a T cell and the third one as a B cell.

```
Lymphocyte,T cell,B cell
1,0,0
1,1,0
1,0,1
1,0,0
```

3.6.3 Example with Probability, Unknown Values, and E Notation

This example demonstrates how to state the probability that an event is a member of a certain class or when the probability is unknown. The listing shows classification results of 3 events and 4 classes. In this example, the probability that the first event is a member of the first class is 0.32 (i.e., 32 %). The probabilities that it is a member of the third and second class are 1.23 % and 97 %, respectively. Finally, it is certain according to these classification results that the first event is a member of the fourth class. The probabilities of the second event being a member of the first and second class are unknown; however, it is certain according to these results that the second event is not a member of the third class. The second event is also a member of the fourth class with a probability of 34 %. Finally, nothing is known about the probability of the last event being a member of any of the classes.

```
Class 1,Class 2,Class 3,Class 4
0.32,1.23E-2,.97,1
,,0,3.4e-1
,,,
```

Annex A

CLR in ACS – Informative

Please note that this Annex is provided for informative purposes only and the ACS specification has not been finalized at the time of writing this text. Use the Archival Cytometry Standard (ACS [2]) specification (once available) as a normative reference for the creation of ACS files.

An ACS container file to bundle an FCS data file with corresponding CLR file containing classification results may be created as follows:

a) Copy the FCS data file and the CLR classification results file to an empty working directory.
b) Create a *TOC1.xml* file in the working directory with the contents as described below.
c) Zip the contents of the working directory and name the resulting ZIP file with an *.acs* file extension. Note: zip only the contents of the working directory, not the working directory itself, so that the files appear in the root of the ZIP archive.

The contents of the *TOC1.xml* file may be as simple as:

```
<?xml version="1.0" encoding="UTF-8"?>
<toc:TOC
  xmlns:toc          = "http://www.isac-net.org/std/ACS/1.0/toc/"
  xmlns:xsi          = "http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation = "http://www.isac-net.org/std/ACS/1.0/toc/
                        http://flowcyt.sf.net/acs/toc/TOC.v1.0.xsd">

  <toc:file toc:URI      = "file:///filename.fcs"
            toc:mimeType = "application/vnd.isac.fcs">
    <toc:associated toc:with         = "file:///filename.cvs"
                    toc:relationship = "classification results" />
  </toc:file>

  <toc:file toc:URI = "file:///filename.cvs" toc:mimeType = "text/csv" />

</toc:TOC>
```

where *filename.fcs* stands for the name of the classified FCS data file and *filename.csv* stands for the name of the CLR file with classification results. Multiple data files and classification result files may be included in a single ACS container. Additional information may also be included as permitted by the Archival Cytometry Standard (ACS [2]) specification. Below is the contents of the TOC1.xml file that shows how additional information may be provided.

```
<?xml version="1.0" encoding="UTF-8"?>
<toc:TOC
  xmlns:toc          = "http://www.isac-net.org/std/ACS/1.0/toc/"
  xmlns:xsi          = "http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation = "http://www.isac-net.org/std/ACS/1.0/toc/
                        http://flowcyt.sf.net/acs/toc/TOC.v1.0.xsd">

  <toc:file toc:URI         = "file:///filename.fcs"
            toc:mimeType    = "application/vnd.isac.fcs"
            toc:description = "Data from Experiment 063 Lymph">
    <toc:additional_info>Additional data information</toc:additional_info>
    <toc:associated toc:with         = "file:///filename.cvs"
                    toc:relationship = "classification results" />
  </toc:file>

  <toc:file toc:URI         = "file:///filename.cvs"
            toc:mimeType    = "text/csv"
            toc:description = "Results for Experiment 063 Lymph">
    <toc:additional_info>Free text description of results</toc:additional_info>
```

```
    <toc:additional_info>
            <creator>
                <name>SamSPECTRAL</name>
                <url>http://bioconductor.org/packages/release/bioc/html/
                     SamSPECTRAL.html</url>
                <version>1.4.1</version>
                <R-version>2.12.1</R-version>
                <Platform>x86_64 (Ubuntu Linux)</Platform>
                <parameters>
                    <dimension>c(1, 2, 3)</dimension>
                    <normal.sigma>200</normal.sigma>
                    <separation.factor>0.39</separation.factor>
                </parameters>
            </creator>
            <keyword name="operator">John McBoss</keyword>
        </toc:additional_info>
  </toc:file>
</toc:TOC>
```

Annex B

Bibliography – Informative

[1] Spidlen J, Leif RC, Moore W, Roederer M, Brinkman RR, International Society for the Advancement of Cytometry Data Standards Task Force. Gating-ML: XML-based gating descriptions in flow cytometry. Cytometry A. 2008;73:1151-1157.

[2] Spidlen J, Moore W, Bray Chris, International Society for Advancement of Cytometry Data Standards Task Force, Brinkman R. Archival Cytometry Standard. Available at: http://flowcyt.sf.net/acs/latest.pdf.

[3] Shafranovich Y. RFC 4180: Common Format and MIME Type for Comma-Separated Values (CSV) Files. Available at: http://tools.ietf.org/html/rfc4180.

[4] Cerf V. ASCII format for Network Interchange. Available at: http://tools.ietf.org/rfc/rfc20.txt.

[5] The Internet Engineering Task Force. Request for Comments: 3629 - UTF-8, a transformation format of ISO 10646. Available at: http://www.ietf.org/rfc/rfc3629.txt.

[6] Spidlen J, Moore W, Parks D, et al. Data file standard for flow cytometry, version FCS 3.1. Cytometry A. 2010;77:97-100.

[7] Bradner S., The Internet Engineering Task Force. Request for Comments: 2119 - Key words for use in RFCs to Indicate Requirement Levels. Available at: http://www.ietf.org/rfc/rfc2119.txt. Accessed 07/31, 2007.

[8] Wikipedia. Scientific notation. Available at: http://en.wikipedia.org/wiki/Scientific_notation.