JCAMP-DX: A Standard Form for Exchange of Infrared Spectra in Computer Readable Form*

ROBERT S. McDONALD§ and PAUL A. WILKS, JR.

9 Woodside Dr.. Burnt Hills. New York 12027 (R.S.M.); and *General Analysis Corp., 140 Water St., So. Norwalk, Connecticut 06856 (P.A.W.)*

JCAMP-DX is a standard file form for exchange of infrared spectra and related chemical and physical information between spectrometer data systems of different manufacture, main-frame time-sharing systems, general purpose lab computers, and personal computers. It is compatible with all media: telephone, magnetic and optical disk, magnetic tape, and even the printed page (via optical reader).

All data are stored as labeled fields of variable length using printable ASCII characters. A JCAMP-DX spectrum is a text file which can be viewed, corrected, and annotated with a text editor. The present focus is on infrared spectra, but JCAMP-DX can easily accommodate Raman, UV, NMR, mass, and other types of spectra, x-ray powder patterns, chromatograms, thermograms, and other plots which require the capability of representing contours as well as peak position and intensity. JCAMP-DX also provides for combining adequate information about the sample and method of observation with its spectrum.

Index Headings: Digital spectra; Computer data processing; Telecommunications; Data exchange; Standard data files.

1. BACKGROUND

The revolution which converted a generation of infrared spectrometers to digital spectra has enabled universal sharing of *accurate* spectral data. *Incompatibility* of internal digital forms of spectra measured by different instruments is an obstacle to such sharing. The Joint Committee on Atomic and Molecular Physical Data (JCAMP) has addressed this problem, with the assistance of the infrared spectrometer manufacturers cited under acknowledgments. JCAMP-DX is the result. It is a standard *external* form for exchanging spectra between data systems, whether they use the same *internal* form or not.

All sponsoring spectrometer manufacturers have developed, or are in the final stages of developing, software based on these specifications for converting between their own *internal* data forms and JCAMP-DX.

Spectra are converted to JCAMP-DX form for transmission to another system. The receiving system normally converts spectra to its own internal form for data manipulation, but the data can be plotted directly, stored in JCAMP-DX form, entered into databases, etc.

The focus of these specifications is on infrared spectra, because needs of infrared spectroscopists stimulated this development. However, the design is readily adaptable to other types of data (see Section 10).

Table I summarizes the design criteria for JCAMP-DX. All information is represented as a series of labeled fields of variable length, consisting of printable ASCII

* Copyright (C) 1986, 1987 by the Joint Committee on Atomic and Molecular Physical Data (JCAMP).

§ Author to whom correspondence should be sent.

TABLE I. Design criteria for JCAMP-DX.

1. Ability to represent *exactly* digital output of dispersive, FT-IR, tunable filter, and laser spectrometers. including spectra, interferograms, transformed spectra, and peak tables.
2. Ability to represent descriptive information on sample identity, sample preparation, instrumental parameters, computational procedures, and comments, as well as spectral data, as appropriate for applications.
3. Internal documentation, readable by humans.
4. Acceptability by a wide variety of computers, communication systems, and storage media.
5. Expandability of each data field to whatever length is required. Ability to add new data fields as the need arises.
6. Sufficiently precise specifications for design of software to produce JCAMP-DX files which can be decoded by other systems designed independently to the same specifications.

characters only. Restriction to printable ASCII characters guarantees that such files are acceptable to *all* computers and data systems. A spectrum in JCAMP-DX form is a *text* file which can be viewed, corrected, and augmented with a standard text editor, and it can be listed in hard copy form without special software.

First priority is given to representing the spectral data itself, without loss of significant detail. Attention is also given to inclusion of descriptive information on sample identity, origin, preparation, method of observation, etc.-information which is normally highly abbreviated or missing from *internal* files.

JCAMP-DX files are intended for ease of *exchange* of data. While they can also be used for building data banks and archiving spectra, they are not designed for rapid-search or other special uses. Rather, they are viewed as a source for creating data bases, as well as for informal exchange of data.

2. SCOPE

(2.1) These specifications define Version 4.24 of JCAMP-DX, a standard file form for exchange of infrared spectra and related chemical information between spectrometer data systems of different manufacture, main-frame time-sharing computer systems, general purpose lab computers, and personal computers. It is compatible with all media: telephone (via modem), magnetic and optical disk, magnetic tape, and the printed page (via optical reader).

(2.2) The JCAMP-DX protocol is meant for public use. These specifications are copyrighted by the Joint Committee on Atomic and Molecular Physical Data (JCAMP) for the purpose of linking them with the name *JCAMP-DX.* The right to copy these specifications for scientific purposes is hereby granted.

(2.3) Use of the name *JCAMP-DX* in the description

TABIE IIa. CORE portion of JCAMP-DX INFRARED FILE. CORE DATA is the irreducible minimum for a JCAMP-DX file, generated automatically by software. See Sectio ȵ.4.*

1. ##TITLE= Epichlorohydrin vapor	$$ 6.1.1
2. ##JCAMP-DX= 4.24.	$$ 6.1.2
3. ##DATA TYPE= INFRARED SPECTRUM	$$ 6.1.3
4. ##ORIGIN= Sadtler Research Laboratories	$$ 7.1.2
5. ##OWNER= EPA/Public Domain	$$ 7.1.3
$$ Lines 6-24 are *optional.* See Table IIb.	
25. ##XUNITS= l/CM	$$ 6.2.1
26. ##YUNITS= ABSORBANCE	$$ 6.2.2
27. ##XFACTOR= 1.0	$$ 6.2.5
28. ##YFACTOR= 0.001	$$ 6.2.5
29. ##FIRSTX= 450.	$$ 6.2.3
30. ##LASTX= 4000.	$$ 6.2.3
31. ##NPOINTS= 1776	$$ 6.2.6
32. ##FIRSTY= 0.058	$$ 6.2.7
33. ##XYDATA= (X++(Y..Y))	$$ 6.4.1
34. 450 58 44 34 39 26 24 22 21 21 19 16 15 15 17 16 etc.	
35. $$ See Table VI for more detailed examples	
36. 3998 16 15 14	
37. ##END=	$$ 6.1.5

‡ **$$ = Section No. of JCAMP-DX Specification 4.24.**

of data files and software implies the capability of converting between *internal* data files and JCAMP-DX according to specifications described in Sections 3 through 8 and in the rest of this section.

(2.3.1) Restriction to single spectra. These specifications apply in detail to *simple* JCAMP-DX files as described in Section 3.3.1. This limitation to a *single* spectrum per JCAMP-DX file is in recognition of the fact that present infrared data systems are designed to store only *one spectrum per internal file.*

(2.3.2) Encoding JCAMP-DX. JCAMP-DX software must be able to generate at least the CORE information, as defined in Section 3.4, from an *internal* file of a particular data system. The resulting JCAMP-DX file must be decodable by software written independently to these specifications. The originator may encode spectral data in *any* ASDF form described in Section 5.

(2.3.3) Decoding JCAMP-DX. JCAMP-DX software must be able to generate an *internal* spectrum file in a particular data system that is equivalent to a spectrum encoded in a *simple* JCAMP-DX file containing spectral data in *any combination* of ASDF forms described in Section 5, with due allowance for limitations of the internal data structure of the receiving system.

(2.3.4) Multiple conversions must not result in progressive loss of accuracy when conversion does not involve interpolation to accommodate different abscissa spacing or truncation to reduce the number of significant figures.

(2.3.5) These specifications define the *form* of NOTES, (Section 3.5), but since NOTES may vary with users, samples, equipment, and applications, the specifications do not attempt to specify *which* NOTES *must* be included, *how* NOTES are generated by the encoding system, or *how* they are to be handled by the receiving system.

(2.3.6) When files are exchanged via wire, an error-correcting protocol is *required,* to guarantee the integrity of the data. Choice of protocol is left to arrangement between the parties involved. Kermit[1] is recommended. Other media, such as local area networks, magnetic disk, and tape, have their own internal error checks which are applied automatically.

TABLE IIb. NOTES portion of JCAMP-DX INFRARED FILES. NOTES vary for different samples, applications, sites, and equipment. They may or may not be part of internal spectrometer data files. In general, they are the responsibility of the user. This material is inserted between lines 5 and 25 of Tabl ȵIa. See Sectio ȵ.5.*

(Continued from TABLE IIa)

6. ##CLASS= Coblentz 3ȵ, IUPAC Bȵ2	$$ 7.1.1
7. ##DATE= 77/08/11 $$TIME is optional.	$$ 7.1.4
8. ##SAMPLE DESCRIPTION= $$This data-set is a	$$ 7.2.1
$$description of the sample, source, properties,	
$$etc. For pure compounds, it may be replaced or	
$$supplemented by lines 9-12 below, as appropriate	
9. ##CAS NAME= 1-chloro-2,3-epoxypropane	$$ 7.2.2
10. ##MOLFORM= C 3 H 5 Cl 0	$$ 7.2.4
11. ##CAS REGISTRY NO= 13403-37-7	$$ 7.2.5
12. ##WISWESSER= T30TJ BlG	$$ 7.2.6
13. ##MP= -25.6C	$$ 7.2.8
14. ##BP= 117C	$$ 7.2.9
15. ##SOURCE REFERENCE= No. 2 (EPA Vapor Library)	$$ 7.1.6
16. ##SPECTROMETER/DATA SYSTEM= Digilab FTS-11;	$$ 7.3.1
17. ##INSTRUMENT PARAMETERS=	$$ 7.3.2
18. ##SAMPLING PROCEDURE= MODE=transmission;	$$ 7.4.1
19. ACCESSORIES: GC capillary flow cell;	
20. TEMP: 200C; PATH LENGTH: 7.6CM; PRESSURE: ?;	
21. ##DATA PROCESSING= BACKGROUND: Empty cell;	$$ 7. 6
22. Smoothing: none;	
23. ##RESOLUTION= 4.0	$$ 6.3.1
24. ##DELTAX= 2.0	$$ 6.3.2

(Continued in TABLE IIa)

' $$ = Section No. of JCAMP-DX Specification 4.24.

3. **STRUCTURE OF JCAMP-DX FILES**

JCAMP-DX is a *FILE* specification. The basic element of JCAMP-DX files is the LABELED-DATA-RECORD (Section 4.2). LABELED-DATA-RECORDS are combined into BLOCKS (Section 3.2). A FILE consists of one or more BLOCKS.

These specifications apply mainly to FILES which consist of a *single* BLOCK.

Guidelines are included to define how BLOCKS are combined to form *compound* FILES representing multiple spectra, or combinations of spectra and other data.

(3.1) **INTERNAL FILE.** The term *internal file* is used exclusively in this document to refer to the data files employed *internally* by spectrometer data systems. (A JCAMP-DX file is an *external* file.)

(3.2) JCAMP-DX BLOCK. A BLOCK is a collection of LABELED-DATA-RECORDS (Section 4.2) starting with ##TITLE= and ending with ##END=. BLOCKS may be nested within outer BLOCKS. ##TITLE= and ##END= serve the functions of BEGIN and END in Pascal. A BLOCK contains as many LABELED-DATA-RECORDS as necessary, but each DATA-LABEL (except ##TITLE=, ##END=, and ##=) may appear no more than once in a BLOCK.

(3.2.1) Innermost BLOCKS are *data* BLOCKS, containing spectra or other forms of data. Data BLOCKS may contain any LDR except *##BLOCKS= (6.1.4).* Only one LABELED-DATA-RECORD containing *spectral data* may appear in a *data* BLOCK.

(3.2.2) Outer BLOCKS are *link* BLOCKS which define the BLOCK structure in much the same way that a file directory defines a group of files. *Link* BLOCKS may contain which are defined under NOTES. Such

LDRs must appear before the first inner BLOCK. *Link* BLOCKS may *not* contain spectral parameters or spectra. *LINK* is the ##DATA TYPE= for *link* BLOCKS. This convention provides a flag to warn software which handles only files whose ##DATA TYPE= is IN-*FRARED SPECTRUM.*

(3.2.3) The SCOPE of a DATA-LABEL is restricted to the current BLOCK. This requires duplication of spectral parameters in *each* data BLOCK. This restriction is necessary in order to make each data BLOCK complete in itself and, thus, separable as a *simple* JCAMP-DX FILE (Section 3.3.1).

(3.3) JCAMP-DX FILES. A JCAMP-DX file is either *simple* or *compound,* as described below.

(3.3.1) SIMPLE JCAMP-DX FILE. A *SIMPLE* JCAMP-DX file consists of a single JCAMP-DX *data* BLOCK containing a single, contiguous section of a spectrum with its defining parameters, information on the sample, instrumental conditions, etc.

(3.3.2) COMPOUND JCAMP-DX FILE. A *COM-POUND* JCAMP-DX file is a hierarchical BLOCK structure containing multiple BLOCKS of spectral and/or other types of data, nested within *link* BLOCKS. A *compound* JCAMP-DX file may contain all data from a GC/IR run, including the chromatogram and spectra of numerous fractions, each in a separate inner *data* BLOCK (Section 11 and Table XI).

(3.4) CORE. The CORE portion of a JCAMP-DX BLOCK (Table IIa) is the irreducible minimum JCAMP-DX file. It contains all information required to plot and label a spectrum. CORE components are defined in Section 6.

(3.5) NOTES. The NOTES portion of a JCAMP-DX BLOCK complements the CORE portion (Table IIb). NOTES describe the sample in greater detail than does ##TITLE=, as well as describing equipment, method of observation, and data processing details. NOTES may contain information which is not found in the *internal* file from which a JCAMP-DX file is derived. Components of the NOTES portion are defined in Section 7.

NOTES are inserted ahead of the spectral data, as indicated in Table IIa, to permit them to be viewed without listing of the whole file. Content of NOTES depends on the user and will vary for different samples, sites, data systems, and applications, as appropriate.

(3.6) FREE FORM. All data in a JCAMP-DX file is free form. The precise column in which a given data element begins is of no significance. Software for processing JCAMP-DX files must scan each line, character by character, to isolate and recognize each data element. Parsers for JCAMP-DX can be written in virtually any high-level language, including FORTRAN and BASIC. Files can be *written,* but not *pursed,* via FORTRAN *fixed-format* or BASIC *free-format* numerical input/output procedures.

4. JCAMP-DX DEFINITIONS

This section is concerned with the *form* of components of JCAMP-DX data files.

(4.1) CHARACTER SET. The *character set* consists of standard printing ASCII characters (Table III). Characters with special meanings are listed in Table IV. Some advantages of ASCII are outlined below.

TABLE III. ASCII printing characters.*

!"#$%&'()*+, — . /0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\] ^ - 'abcdefghijklmnopqrstuvwxyz{ |} ~

* Note: The characters [\]'{}~ are not included in the EBCDIC set used by IBM mainframes.

TABLE IV. Special characters.*

##	At the start of a line, preceded only by blanks indicate the start of a data-label.
=	Indicates the end of a data-label.
=	At end of a line indicates continuation without break on next line. Used only with special string fields. Never used for numeric data.
$$	Indicates remainder of line is a comment.
$	First character of a user defined label (4.4.2).
?	Replaces numerical entry which is unavailable or out-of-range. Used only in tabular data.
<>	Used instead of quotes to delimit strings.
()	Used to delimit data groups which contain strings.

*Note: The characters #, =, (,), <, >, and $ may be used elsewhere without special meaning.

(4.1.1) ASCII can be listed without special software.

(4.1.2) Communication hardware and software is normally transparent to printable ASCII characters.

(4.1.3) ASCII representation of numbers is of variable length, allowing unlimited accuracy while using only as much space as necessary.

(4.1.4) Knowledge of the *internal* hardware representation of numbers is not required in order to write software for ASCII conversion.

ASCII files present only minor problems for IBM mainframes which use the EBCDIC character set. JCAMP-DX files can be processed in EBCDIC. Character conversion is automatic for communication via Kermit.[l] The ASCII characters [\]'{} ~are not defined for EBCDIC, and should generally be avoided in exchanges with IBM mainframes. Handling of these characters depends on the ASCII/EBCDIC translation table of the I/O system.

(4.2) LABELED-DATA-RECORDS (LDR). An *LDR* consists of a flagged *data-label* and an associated *data-set,* defined below. An LDR begins with a *data-label-flag (##)* and ends with the next *data-label-flag* (4.4).

(4.3) LINES. LDRs are divided into *lines* of 80 or fewer characters, terminated by <crlf>. End-of-line is equivalent to a blank, except for certain special cases. Each LDR occupies as many lines as necessary.

(4.4) DATA-LABELS. A *data-label* is the name of an LDR. It is delimited by a *data-label-flag (##)* and a *data-label-terminator (=).* ##TITLE= is a DATA-LABEL with flags. A LINE contains no more than one DATA-LABEL. When LABELS are parsed, alphabetic characters are converted to upper case, and all spaces, dashes, slashes, and underlines are discarded. (XUNITS, xunits, xUNITS, and X-UNITS are equivalent.)

(4.4.1) RESERVED-DATA-LABELS. Data-labels described in Sections 6-8 are *global* in scope, not *local* to a particular instrument, location, or use. RESERVED-DATA-LABELS consist of alphabetic and numeric characters only. The first character is always alphabetic. A full list of RESERVED-LABELS is given in Table V.

(4.4.2) USER-DEFINED-DATA-LABELS. USER-

TABLE V. Reserved labels, JCAMP-DX version 4.24.*

HEADER INFORMATION		SAMPLE INFORMATION	
*6.1.1	##TITLE=	7.2.1	##SAMPLE DESCRIPTION=
*6.1.2	##JCAMP-DX=	7.22	##CAS NAME=
*6.1.3	##DATA TYPE=	7.2.3	##NAMES=
6.1.4	##BLOCKS=	7.2.4	##MOLFORM=
*6.1.5	##END=	7.2.5	##CAS REGISTRY NO=
		7.2.6	##WISWESSER=
REQUIRED SPECTRAL PARAMETERS		7.2.7	##BEILSTEIN LAWSON NO=
*6.2.1	##XUNITS=	7.2.8	##MP
*6.2.2	##YUNITS=	7.2.9	##BP
*6.2.3	##FIRSTX=, ##LASTX=	7.2.10	##REFRACTIVE INDEX=
6.2.4	##MAXX=, ##MINX=	7.2.11	##DENSITY=
6.2.4	##MAXY=, ##MINY=	7.2.12	##MW=
*6.2.5	##XFACTOR=, ##YFACTOR=	7.2.13	##CONCENTRATIONS=
*6.2.6	##NPOINTS=		
*6.2.7	##FIRSTY=	SAMPLING INFORMATION	
		7.4.1	##SAMPLING PROCEDURE=
OPTIONAL SPECTRAL PARAMETERS		7.4.2	##STATE=
6.3.1	##RESOLUTION=	7.4.3	##PATH LENGTH=
6.3.2	##DELTAX=	7.4.4	##PRESSURE=
		7.4.5	##TEMPERATURE=
TABULAR SPECTRAL DATA		7.4.6	##DATA PROCESSING=
*6.4.1	##XYDATA=		
6.4.2	##XYPOINTS=	COMMENTS	
6.4.3	##PEAK TABLE=	7.5.1	##=
6.4.4	##PEAK ASSIGNMENTS=	7.5.2	$$
		7.5.3	##XLABEL=, ##YLABEL=
MISCELLANEOUS			
7.1.1	##CLASS	FOURIER DOMAIN	
*7.1.2	##ORIGIN=	8.1	##RUNITS=
*7.1.3	##OWNER=	8.2	##AUNITS=
7.1.4	##DATE=	8.3	##FIRSTR=, ##LASTR=
7.1.3	##TIME=	8.4	##DELTAR=
7.1.6	##SOURCE REFERENCE=	8.5	##MAXA=, ##MINA=
7.1.7	**##CROSS REFERENCE=**	8.6	##RFACTOR=, ##AFACTOR=
		8.7	##FIRSTA=
EQUIPMENT		8.8	##ALIAS=
7.3.1	##SPECTROMETER/DATA SYSTEM=	8.9	##ZPD=
7.3.2	##INSTRUMENT PARAMETERS=	8.10	##RADATA=

* Required for INFRARED SPECTRA.

DEFINED-DATA-LABELS are local to a particular instrument, location, or use. They can be generated by *users* for such fields as operator's name, shoporder, slit program, scan-speed, number of scans, etc., as needed for private use. The first character in a USER-DEFINED-DATA-LABEL is $, to distinguish it from a RESERVED-DATA-LABEL.

(4.5) DATA-SETS. The actual data portion of an LDR is called a DATA-SET. A DATA-SET usually starts on the same line as the DATA-LABEL, but for tabular data, i.e., ##XYDATA= (6.4.1), the DATA-SET starts on the line after the LABEL. A DATA-SET is of one of the following four forms, as specified for each LDR.

(4.5.1) TEXT data-sets contain descriptive information for humans, not normally intended to be parsed by computer, i.e., title, comments, origin, etc.

(4.5.2) STRING data-sets contain alphanumeric fields intended to be parsed by computer and read by a human. The form of each string field is specified under the LABELED-DATA-RECORD in which it is used.

(4.5.3) AFFN (ASCII FREE FORMAT NUMERIC) is similar to free-form input of BASIC. A field which starts with a +, −, decimal point, or digit is treated as a numeric field and converted to the internal form of the target computer. E is the only other allowed character. A numeric field is terminated by E, comma, or blank. If E is followed immediately by either + or − and a two- or three-digit integer, it gives the power of 10 by which the initial field must be multiplied.

(4.54) ASDF (ASCII SQUEEZED DIFFERENCE

FORM) is a special form for compressing tabular spectral data described in Section 5.

5. ASDF (ASCII SQUEEZED DIFFERENCE FORM)

(5.1) TABULAR DATA. Infrared spectra are normally tabulated in a series of lines, each starting with an abscissa and followed by a series of ordinates at equal abscissa intervals:

$$X1, Y1, Y2, \ldots Yi \qquad \text{(See Table VI for examples)}$$
$$X_{i+1} \quad Y_{i+1} \quad Y_{i+1} \ldots$$

(5.1.1) *VARIABLE LIST.* The form of TABULAR DATA is represented symbolically as a *variable list* as follows: $(X+ + (Y..Y))$, where . . indicates indefinite repeat until the line is filled and + + indicates that X is incremented by (LASTX-FIRSTX)/(NPOINTS-1) between adjacent Ys.

Other forms of *variable list* for tables of discrete points are (XY..XY), (XYZ..XYZ), etc., meaning that each line consists of *groups* consisting of XY pairs, XYZ triplets, etc.

Groups should not be split between lines. The following delimiters are recommended: semicolon or blank between groups, and comma between individual components of a group. Extra blanks can be used for readability; they are ignored during processing. Strings are allowed as components of a variable list. When a string contains blank, comma, or semicolon, the whole string is enclosed in angle brackets (< >) rather than quotes. This provides unique opening and closing delimiters. Groups which extend for more than one line are enclosed in parentheses. End-of-line terminates a group, except within the scope of angle brackets, or parentheses. If a DATA LABEL is encountered while one is processing a group, both the group and the current LDR are terminated. Delimiters can be redefined for a particular LDR; for example, ASDF form is used for long tables.

(5.2) **DATA COMPRESSION**. TABULAR DATA can be compressed by (1) converting decimals to integers, (2) expressing data as *differences* between adjacent points, (3) combining *duplicate* ordinates, and (4) replacing leading digits by *pseudo-digits* which represent delimiter, sign, and the digit itself (Table VII). Subsequent digits of a multidigit number are standard ASCII digits.

It is not necessary to specify ASDF forms separately, because the first character of each numerical value contains this information. A *single* software procedure can be written to decode *any* combination of the ASDF forms described below. The compressed forms can be encoded or decoded faster than ASCII digits alone, and also faster than most other forms of compressed data.

(5.3) **AFFN**. ASDF processors should accept ASCII FREE FORMAT NUMERIC (4.5.3) to simplify user generation of JCAMP-DX files by systems which are not supported by manufacturers.

(5.4) **PACKED Form (PAC)**. Adjacent values are separated by + or − or blank.

(5.5) **SQUEEZED Form (SQZ)**. Delimiter, leading digit, and sign are replaced by a pseudo-digit from Table VII, lines

(5.6) **DIFFERENCE Form (DIF)**. Delimiter, leading

TABLE VI. Examples of ASDF options for a sample spectrum of 53 points (106 bytes as 2 byte binary integers).

```
##YUNITS=  ABSORBANCE                                              VI.1
##XFACTOR= 1.0
##YFACTOR= 0.001
##XYDATA=  (X++(Y..Y))                       Fixed length (I6), 372 bytes
599.860     0     0     0     0     2     4     4     4     7
617.220     5     4     4     5     5     1    10    11    11
634.579     6     5     7     6     9     9     7    10    10
651.938     9    10    11    12    15    16    16    14    17
669.298    38    38    35    38    42    47    54    59    66
686.657    75    78    88    96   104   110   121   128
```

```
##YUNITS=  ABSORBANCE                                              VI.2
##XFACTOR= 1.0
##YFACTOR= 0.001
##XYDATA=  (X++(Y..Y))                            PAC form, 167 bytes
599.860 0 0 0 0 2 4 4 4 7 5 4 4 5 5 7 10 11 11 6 5 7 6 9 9 7
648.081 10 10 9 10 11 12 15 16 16  14 17 38 38 35  38 42 47 54
682.799 59 66 75 78 88 96 104 110  121 128
```

```
##YUNITS=  ABSORBANCE                                              VI.3
##YFACTOR= 0.001
##XYDATA=  (X++(Y..Y))                           DIFDUP Form, 74 bytes
599.860@VKT%TLkj%J%KLJ%njKjL%kL%jJULJ%kLKl%lLMNPNPRLJ0QTOJ1P
700.158A28      $$ checkpoint
```

```
##YUNITS=  ABSORBANCE                                              VI.4
##XFACTOR= 1.92881
##YFACTOR= 0.001
##XYDATA=  (X++(Y..Y))                 DIFDUP Form, Scaled X, 66 bytes
311@VKT%TLkj%J%KLJ%njKjL%kL%jJULJ%kLK1%lLMNPNPRLJ0QTOJIP
363A28      $$ checkpoint
```

TABLE VIIa. Pseudo-digits or ASDF forms.*

	0	1	2	3	4	5	6	7	8	9
1. ASCII digits	0	1	2	3	4	5	6	7	8	9
2. Positive SQZ digits	@	A	B	C	D	E	F	G	H	I
3. Negative SQZ digits		a	b	c	d	e	f	g	h	i
4. Positive DIF digits	%	J	K	L	M	N	O	~	Q	R
5. Negative DIF digits		j	k	l	m	n	o	p	q	r
6. Positive DUP digits		S	T	U	V	W	X	Y	Z	s

* Note: The above characters replace the leading digit, sign, and preceding space for SQZ, DIF, and DUP forms (Section 5.). Remaining digits of a multidigit number are standard ASCII.

TABLE VIIb. Simple examples of ASDF form.

```
FIX form (I3):     1    2    3    3    2    1    0   - 1   - 2   - 3
PAC form:          1+2+3+3+2+1+0-1-2-3
       or:         1  2  3  3  2  1  O -1 - 2 - 3
SQZ form:          lBCCBA@abc
DIF form:          1JJ%jjjjjj
DIFDUP form:       1JT%jX
```

preceding the undefined ordinate. Except for the break caused by the undefined ordinate itself, this procedure preserves the chain of *Y-value* checks which would otherwise be broken.

(5.9) DUPLICATE SUPPRESSION (DUP). When two or more adjacent Y-values in a table are identical, all but the first are replaced by a *duplicate-count* whose initial digit is a pseudo-digit from Table VII line 6. *Duplicate-count* is the number of successive identical table values, *including* the first. It can be used with all ASDF forms. *Count* for four identical values is 4, i.e., 50 50 50 50 becomes 50V in DUP form.

(5.10) DIFDUP Form. When *duplicate suppression* is combined with *difference* form, the *duplicate* count is obtained by counting identical *differences.* The above example, 50 50 50 50, becomes 50% % % in DIF form, and 50 % U in DIFDUP form.

DIFDUP form takes the least space and is processed most rapidly. However, spectral data in this form are not easily inspected visually.

(5.11) ASDF EXAMPLES. Table VI shows JCAMP-DX spectral data using the major options, each containing the same set of 53 ordinate values. In the example, DIFDUP form requires 66 bytes (including <crlf>), compared to 106 and 212 for 2- and 4-byte integers, respectively.

6. RESERVED-LABELS FOR CORE DATA

This section defines reserved data-labels for JCAMP-DX CORE data for spectra, together with content and form of corresponding data-sets. This information is always present or implied in *internal* spectrometer data files. Software provided by manufacturers should automatically convert CORE data.

##ORIGIN= (7.1.2) and ##OWNER= (7.1.3) are rightly part of the CORE; they are defined under NOTES because they usually cannot be derived from *internal* files.

(6.1) BLOCK HEADER INFORMATION.

(6.1.l)##TITLE=(TEXT). *Required* as the *first* LDR of all JCAMP-DX FILES and BLOCKS. Software which decodes JCAMP-DX files should check for the initial ##TITLE= to prevent accidental processing of JCAMP-DX files. For a BLOCK containing a spectrum, ##TITLE= should contain a concise description of the

digit, and sign of the difference between adjacent values are represented by a pseudo-digit from Table VII, lines 4,5. The abscissa at the start of each line is in AFFN form, and the first ordinate is an *actual* value in SQZ form. Remaining ordinates are *differences,* in DIF form.

(5.7) INVALID-DATA SYMBOL. In all forms, ? represents a missing or off-scale ordinate. It is effectively a SQZ form. In DIF form, the ordinate following ? is an actual value in SQZ form, *not* a difference.

(5.8) CHECK-POINTS. The start of each line after the first is a CHECK-POINT. Checks are of two types: *X-sequence* checks, and *Y-value* checks.

(5.8.1) X-SEQUENCE CHECK. At the beginning of each line, *X-sequence* is checked in all modes to make sure that a whole line is not omitted or duplicated.

(5.8.2) Y-VALUE CHECK. In DIF mode, a Y-value check is performed at the beginning of each line to guard against a single error which would invalidate all following values. When, and only when, the last ordinate of a line is in *DIF* form, the abscissa does not advance before the next line. The first ordinate of the next line (after the leading abscissa value) is always an *actual* value, equal to the last *calculated* ordinate of the previous line.

(5.8.3) The last line of a block of *DIF* data contains only abscissa and ordinate for the Y-value check of the last ordinate.

Example of DIF checkpoint:

X_1 Y_1, $D_{2,1}$ . . . $D_{i,i-1}$ ($D_{,,}$ represents Y_2-Y_1)

Xi Yi   @checkpoint

(5.8.4) If the *last* or *next-to-last* ordinate in a line is ?, the check-point at the start of the next line is a simple *X-sequence* check, not a Y-value check. It is recommended that one avoid this situation by terminating the line just before the ordinate which is represented by ?, starting a new line with a checkpoint for the ordinate

spectrum, *suitable as a title for a plotted spectrum.* Software which generates a *simple* JCAMP-DX file should copy the internal field which most closely meets this requirement.

(6.1.2) ##JCAMP-DX= (STRING). Required immediately after ##TITLE= of each JCAMP-DX BLOCK to show version of JCAMP-DX, for example: ##JCAMP-DX= 4.24.

Digit 1 refers to LDRs of CORE data. Digit 2 refers to LDRs defined as string fields. Digit 3 refers to LDRs defined as text fields and to editorial changes in the specifications. Software procedures which process CORE data should check Digit 1, and procedures which process string fields should check Digit 2. Digit 3 generally does not require checking.

(6.1.3) ##DATA TYPE= (STRING). *Required* to define type of data in a BLOCK (Table IIa, line 3). ##DATA TYPE= for a *data* BLOCK containing an infrared spectrum is *INFRARED SPECTRUM* ##DATA TYPE= for all *link* BLOCKS is *LINK.*

Other ##DATA TYPES= are: RAMAN SPECTRUM (10.2), INFRARED PEAK TABLE (10.3), INFRARED INTERFEROGRAM (10.4), and INFRARED TRANSFORMED SPECTRUM (10.5).

(6.1.4) ##BLOCKS= (AFFN). This LDR gives the number of *link* and *data* BLOCKS in the first level of the current BLOCK of a *compound* JCAMP-DX file (Section 11). This LDR is *required* in *link* BLOCKS; it is not allowed in *data* BLOCKS.

(6.1.5) ##END=. End of a JCAMP-DX BLOCK. There must be one ##END= for each ##TITLE= in a JCAMP-DX file.

(6.2) REQUIRED SPECTRAL PARAMETERS. X and Y represent abscissa and ordinate of spectral data. Spectral parameters are defined to accommodate *internal* forms of spectra for all IR systems, including transmittance or absorbance, wavelength or wavenumber, rational wavenumber (6.2.5) or laser abscissa intervals, and descending or ascending abscissa.

(6.2.1) ##XUNITS= (STRING). Specifies abscissa units for ##XYDATA=:

> 1/CM   (representing CM-l)
> MICROMETERS
> NANOMETERS
> SECONDS

(6.2.2) ##YUNITS= (STRING). Specifies ordinate units of ##XYDATA=, where I = intensity, and R = reflectance:

> TRANSMITTANCE   (WI,)
> REFLECTANCE   (WI,)
> ABSORBANCE $(\log_{10}(Io/I))$
> KUBELKA-MUNK $((1-R^2)/(2R))$
> ARBITRARY UNITS

This list is presently limited to the units which are implemented for most current infrared spectrometer systems. Other units defined in ASTM E-131 (Committee E13) are candidates for inclusion.

(6.2.3) ##FIRSTX=, ##LASTX= (AFFN). First and last *actual* abscissa values of ##XYDATA=. First tabulated abscissa times ##XFACTOR= should equal ##FIRSTX=. Spectral data are tabulated in order of either increasing or decreasing abscissa values. ##FIRSTX= specifies abscissa of the first datum of ##XYDATA=, not necessarily the smallest abscissa.

(6.2.4) ##MAXX=, ##MINX=, ##MAXY=, ##MINY= (AFFN). These *optional* LDRs give the largest and smallest *actual* X and Y-values in a spectrum. When used, they precede ##XYDATA=. They are used for scaling when spectra are plotted directly from JCAMP-DX files and as follows: (1) ##MAXX= and ##MINX= are required to give the full X-range if different regions of a spectrum are contained in separate BLOCKS of a *compound* file, and (2) ##MAXY= and ##MINY= are *required* if Y-range exceeds *recommended ordinate scaling* (6.4.1.1).

(6.2.5) ##XFACTOR=, ##YFACTOR= (AFFN). Factors by which components of ##XYDATA= are multiplied to obtain actual values.

##XFACTOR= may be used to convert X-values in ##XYDATA= to integers. *Rationally* spaced X-values (intervals of 0.5,1.0, etc.) are usually left unscaled. Laser spaced X-values require several decimal places. Such X-values can be converted to integers to reduce file size by a suitably chosen ##XFACTOR= (which must be stated with sufficient precision for accurate reconstruction of X-values).

##YFACTOR= is normally chosen to convert Y-values to integers to save space required by decimal points, i.e., if ##YFACTOR= is 0.0001, transmittance of 0.9999 is tabulated as 9999.

(6.2.6) ##NPOINTS= (AFFN). Number of components of ##XYDATA=, ##XYPOINTS=, ##PEAK TABLE=, or ##RADATA=.

(6.2.7) ##FIRSTY= (AFFN). Actual Y-value corresponding to ###FIRSTX=. ##FIRSTY= should equal ##YFACTOR= times the first Y-value value in ##XYDATA=.

(6.3) *RECOMMENDED* SPECTRAL PARAMETERS. The following parameters are desirable from the standpoint of the spectroscopist. Inclusion is *recommended* but they are not used in converting spectral data.

(6.3.1) ##RESOLUTION= (STRING). Nominal resolution in units specified by ##XUNITS=, as a single number for spectra at constant resolution throughout, or as pairs of the form: $R_1,X_1; \ldots ;R_i,X_i$, where R_i stands for resolution at abscissa X_i. ##RESOLUTION= is strongly recommended for FT-IR spectra.

(6.3.2) ##DELTAX= (AFFN). This LDR gives the nominal spacing between points for inspection by the user. Actual X-values are calculated from the spacing given by (LASTX-FIRSTX)/(NPOINTS-1) (see Section 5.1.1).

(6.4) TABULAR SPECTRAL DATA.

(6.4.1) ##XYDATA= (AFFN or ASDF). Contains a table of ordinates at equal X-intervals specified by parameters defined in Section 6.2. This LDR will extend over many lines. ##XYDATA= is followed on the same line by a *variable list* i.e., (X++(Y..Y)), representing the form of the table as described in Section 5.1. (Table IIa, line 33.)

Under no circumstances should ##XYDATA= start with a region of undefined data. Software should adjust ##FIRSTX= so that the first point is valid data.

(6.4.1.1) *RECOMMENDED ORDINATE SCALING.*

It is *recommended* that ordinates be scaled between -32767 and 32767. This is an adequate dynamic range for most applications and keeps the size of JCAMP-DX files within reasonable limits. With this scaling, there is no danger of overflowing 2-byte integer arrays used by some software. Ordinate range may exceed these limits, however. If so, actual range is stated via ##MAXY= and ##MINY=. Decoding software should scale ordinate data automatically if it exceeds *internal* word size.

(6.4.1.2) *ABSCISSA SPACING.* A data system may tabulate ##XYDATA= with whatever spacing is used in *internal files.* The intent is to allow the original data to be represented *exactly* by JCAMP-DX, without rounding of any sort. Since certain data processing operations require exact abscissa registration, interpolation may be required before spectra from different systems can be compared. Thus, data from a FT-IR system which employs laser spacing must be interpolated before use by a system with *rational* spacing, and vice versa. JCAMP-DX places the burden of such interpolation on the receiving system. This arrangement avoids the need for one FT-IR system to undo interpolation by another which uses precisely the same *internal* interval.

(6.4.1.3) *ABSCISSA ORDER.* Historically, the plotting of infrared spectra has been done with the frequency decreasing from left to the right. Some systems store spectra in this order. Others store spectra in conventional increasing order. JCAMP-DX allows encoding in the order used *internally* by the encoding system. Decoding software must accept data in either order and convert to that of the receiving system.

(6.4.1.4) *SPECTRUM SIZE.* JCAMP-DX places no restrictions on size of ##NPOINTS=, which can vary by more than an order of magnitude, depending on resolution. Spectra which are readily handled by the originating system may overflow available storage in the receiving system. Software for decoding JCAMP-DX should be prepared to extract usable regions, or to degrade resolution of spectra which exceed *internal* limits. Most systems accept 1 cm-l spacing (4-5000 points).

(6.4.2) ##XYPOINTS= (AFFN). Contains a table of XY pairs for spectral data which are not spaced at constant abscissa increments. This LDR is intended for spectra taken with infrared filter spectrometers. ##XYPOINTS= is followed on the same line by the *variable* list (XY..XY) (Section 5.1.1). X and Y are separated by commas; pairs are separated by semicolons or blanks. This LDR should not be used for peak tables (use ##PEAK TABLE=).

(6.4.3) ##PEAK TABLE= (AFFN). Contains a table of peaks. For a table of peak abscissa vs. intensity, ##PEAK TABLE= is followed on the same line by the *variable list (XY..XY)* (Section 5.1.1). Peak data starts on the following line. For peak position, intensity, and width, the corresponding form would be (XYW..XYW), etc. Groups are separated by semicolon or space; components of a group are separated by commas. X and Y units are given by ##XUNITS= and ##YUNITS=. Peak width and other peak kernel functions should be defined by a $$ comment on the line(s) below ##PEAK TABLE=.

(6.4.4) ##PEAK ASSIGNMENTS= (STRING). List of peak assignments for components, functional groups, or vibrational modes in the form:

##PEAK ASSIGNMENTS = (XYA) or (XYWA)
(X1 [,Y1] [,W1],\<A1\>)

(Xi [,Yi] [,Wi],\<Ai\>)

Parentheses indicate the start and end of each peak entry; square brackets indicate optional elements. Each peak entry starts on a new line, and may continue on following lines if necessary. X and Y units are specified by ##XUNITS= and ##YUNITS=, respectively. W stands for width. The *width-function* is specified by a comment on the line(s) below ##PEAK ASSIGN-MENTS=. The symbol *A* represents a string describing the assignment (enclosed in angle brackets).

7. JCAMP-DX RESERVED LABELS FOR NOTES

NOTES are the *variable* part of JCAMP-DX files. The choice of a standard set of labels for NOTES is a compromise between having labels for everything and holding the list within bounds acceptable for maintenance of software and documentation.

Choice of labels for a particular spectrum depends on the sample, site, or application. The information may or may not be present in *internal* spectrometer data files. The *user* is responsible for inclusion of NOTES.

(7.1) CLASS, ORIGIN, **and OWNER of SPECTRA, etc.** ##ORIGIN= and ##OWNER= are required LDRs. They are defined in this section rather than in Section 6, with CORE definitions, because they cannot normally be generated by software from *internal* spectrum files.

(7.1.1) **##CLASS= (STRING).** Specifies the Coblentz Class of the spectrum (1,2,3, or 4)[2] and the IUPAC Class of digital representation (A, B, C).[3] This LDR is *required* for spectra of *pure* compounds.

(7.1.2) ##ORIGIN= (TEXT). Name of organization, address, telephone number, name of individual contributor, etc., as appropriate. *This information is not optional; the originator of a JCAMP-DX file should always see that it is included.*

(7.1.3) **##OWNER= (TEXT).** Name of owner of a proprietary spectrum. The organization or person named under ##ORIGIN= is responsible for accuracy of this field. If data are copyrighted, this line should read "COPYRIGHT (C) (year) by (name)." *This information is not optional; the originator of a* JCAMP-DX file *should always see that it is included.*

If ##OWNER= contains *PUBLIC DOMAIN,* the implication is that the data may be copied without permission on the authority of whoever is named under ##ORIGIN=. *Absence of ##OWNER= does not imply permission to copy.*

(7.1.4) **##DATE= (STRING).** Date when spectrum was measured, in the form: YY/MM/DD (note order).

(7.1.5) **##TIME= (STRING).** Time when spectrum was measured, in the form: HH:MM:SS.

(7.1.6) **##SOURCE REFERENCE= (TEXT).** Adequate identification to locate the original spectrum, i.e., name of file containing the spectrum, or library name and serial number of the spectrum.

(7.1.7) ##CROSS REFERENCE= (TEXT). Cross references refer to additional spectra of the same sample, i.e., different thickness, mulling agent, polarization, temperature, time, etc., or serve to link a peak table or interferogram with a spectrum.

(7.2) SAMPLE INFORMATION. At least one of the *optional* LDRs described in this section should be included in each JCAMP-DX file.

(7.2.1) ##SAMPLE DESCRIPTION= (TEXT). If the sample is not a pure compound, this field should contain its description, i.e., composition, origin, appearance, results of interpretation, etc. If the sample is a known compound, the following LDRs specify structure and properties, as appropriate.

(7.2.2) ##CAS NAME= (STRING). Name according to Chemical Abstracts naming conventions as described in Appendix IV of the 1985 CAS Index Guide. Examples can be found in Chemical Abstracts indices or the Merck Index. Greek letters are spelled out, and standard ASCII capitals are used for small capitals. Sub-/superscripts are indicated by prefixes / and A, respectively. Example: alpha-D-glucopyranose, 1-(dihydrogen phosphate).

(7.2.3) ##NAMES= (STRING). Common, trade, or other names. Multiple names are placed on separate lines.

(7.2.4) ##MOLFORM= (STRING). Molecular formula. Elemental symbols are arranged with carbon first, followed by hydrogen, and then remaining element symbols in alphabetic order. The first letter of each elemental symbol is capitalized. The second letter, if required, is lower case. One-letter symbols **must** be separated from the next symbol by a blank or digit. Sub-/superscripts are indicated by the prefixes: / and A, respectively. Sub-/superscripts are terminated by the next nondigit. Slash may be omitted for subscripts. For readability, each atomic symbol may be separated from its predecessor by a space. For substances which are represented by dot-disconnected formulas (hydrates, etc.), each fragment is represented in the above order, and the dot is represented by *. Isotopic mass is specified by a leading superscript. D and T may be used for deuterium and tritium, respectively. Examples:

C2H402 or C2 H4 02 (acetic acid)
C6 H9 Cr 06 * H2 0 (chromic acetate monohydrate)
H2 A170 (water, mass 17 oxygen)

(7.2.5) ##CAS REGISTRY NO= (STRING). CAS Registry Numbers for many compounds can be found in Chemical Abstracts indices, Merck Index, or CAS ON-LINE.

(7.2.6) ##WISWESSER= (STRING). The Wiswesser line notation is a precise and concise means of expressing structural formulas as character strings. The basic idea is to use letter symbols to denote functional groups and numbers to express the lengths of chains and the sizes of rings. Wiswesser notation is described in detail by Smith.[4] Wiswesser notation for many compounds is tabulated by Smith[4] and by Grasselli and Ritchey.[5] Wiswesser notation should be enclosed in angle brackets if it extends for more than one line, and care should be taken to terminate lines between expressions separated by blank.

(7.2.7) ##BEILSTEIN LAWSON NO= (STRING). The Beilstein Lawson Number is a structural formula clustering code based on the Beilstein System. It is used for locating information in Beilstein databases, including the STN online database. The code contains only structural information and can be generated locally on microcomputers from connection table data alone. A simple description of the algorithm has been published.[6]

(7.2.8) ##MP= (AFFN). Melting point in degrees C.

(7.2.9) ##BP= (AFFN). Boiling point in degrees C.

(7.2.10) ##REFRACTIVE INDEX= (AFFN). In the form: ND = 1542^20 (index at 20°C for NaD line).

(7.2.11) ##DENSITY=. (AFFN). Density in g/cc.

(7.2.12) ##MW= (AFFN). Molecular weight.

(7.2.13) ##CONCENTRATIONS= (STRING). List of known components and impurities and their concentrations in the following form, where N stands for name (enclosed in angle brackets if more than one word); C, for concentration; and U, for units of concentration:

$$\text{\#\#CONCENTRATIONS = (NCU)}$$
$$\text{(N1, C1, U1)}$$

$$\text{(Ni, Ci, Ui)}$$

The group for each component is enclosed in parentheses. Each group starts a new line and may continue on following lines.

(7.3) EQUIPMENT.

(7.3.1) ##SPECTROMETER/DATA SYSTEM= (STRING). This field contains manufacturer's name, model of the spectrometer, software system, and release number, as appropriate, in the form used by the manufacturer.

(7.3.2) ##INSTRUMENTAL PARAMETERS= (STRING). This optional field is a list of pertinent instrumental settings. Only settings which are **essential** for applications should be included.

(7.4) SAMPLING INFORMATION.

(7.4.1) ##SAMPLING PROCEDURE= (TEXT). First entry in this field should be MODE of observation (transmission, specular reflection, PAS, matrix isolation, photothermal beam deflection, etc.), followed by appropriate additional information, i.e., name and model of accessories, cell thickness, and window material for fixed liquid cells, ATR plate material, angle and cone of incidence, and effective number of reflections for ATR measurements, polarization, and special modulation techniques, as discussed by Grasselli **et al.[7]** The LDRs defined below are used in place of, or to supplement, ##SAMPLING PROCEDURE=.

(7.4.2) ##STATE= (STRING). Solid, liquid, gas, solution, KBr pellet, powder, nujol mull, etc.

(7.4.3) ##PATH LENGTH= (STRING). Pathlength in cm.

(7.4.4) ##PRESSURE= (STRING). Sample pressure in appropriate units if significantly different from room temperature.

(7.4.5) ##TEMPERATURE= (STRING). Sample temperature in degrees C if significantly different from room temperature.

(7.4.6) ##DATA PROCESSING= (TEXT). Description of background correction, smoothing, subtraction, deconvolution procedures, apodization function, zero-fill, or other data processing, together with reference to original spectra used for subtractions.

(7.5) COMMENTS.

(7.5.1) ##=. Comments may be specified by a data-**label-flag** plus a **data-label-terminator,** with a null **data-**

label .Such comments may continue for more than one line, terminating at the next *data-lab e--flag.*

(7.5.2) $$. Comments may be entered at any point in a line by prefixing the first word of the comment by $$. Such comments continue only to the end of the current line, and they do not terminate an LDR.

(7.5.3) ##XLABEL=, ##YLABEL= (TEXT). Optional labels for X- and Y-axes, for use when spectra are plotted directly from JCAMP-DX files. These LDRs should *not* be used in place of ##XUNITS= or ##YUNITS=.

8. RESERVED LABELS FOR FOURIER DOMAIN DATA

This section defines a separate set of labels using R and A as abscissa and ordinate in the Fourier domain. For interferograms, R is used for *optical reardation* and A for signal *amplitude* .R and A are also used for transformed spectra. X and Y represent *valid spectral data,* defined by filter window, alias, detector, etc. Both RA and XY are required simultaneously for interferograms (10.4) and transformed spectra (10.5).

(8.1) ##RUNITS= (STRING). Defines units of *optical* retardationfor interferograms (MICROMETERS).

(8.2) ##AUNITS =(STRING). Specifies the units of amplitude of interferograms (ARBITRARY UNITS or VOLTS).

(8.3) ##FIRSTR=, ##LASTR= (AFFN). First and last actual optical retardation values of an interferogram, usually measured relative to the first tabulated point of the interferogram.

(8.4) ##DELTAR= (AFFN). Optical retardation per data point expressed in ##RUNITS=. ##DELTAR is a function of laser frequency and interferometer geometry.

(8.5) ##MAXA=, ##MINA= (AFFN). Maximum and minimum amplitude of interferogram ordinate values in ##RUNITS=. ##MAXA= and ##MINA= are *required* because of the large interferogram amplitude near ZPD.

(8.6) ##RFACTOR=, ##AFACTOR= (AFFN). Factors by which spectral data under ##RADATA= are multiplied to obtain actual values. Normally, ##RFACTOR= equals ##DELTAR=.

For interferograms, ##AFACTOR= should reflect amplifier gain, etc., to allow transmittance to be calculated as the ratio of sample to reference intensities.

(8.7) ##FIRSTA= (AFFN). Actual A-value corresponding to ##FIRSTR=. ##FIRSTA= is the product of ##AFACTOR= and the first A in ##RADATA=.

(8.8) ##ALIAS= (AFFN). A fraction whose denominator is the number of *possible* aliases for a given sample and whose numerator is the number of the alias *represented by the data* For sampling at alternate laser fringes, the number of aliases is 2, and the alias for the region starting at zero frequency is 1, giving an ##ALIAS= of 1/2.

(8.9) ##ZPD= (AFFN). The number of data points before zero path difference (ZPD). ZPD can be an integer giving approximate position or a real number giving precise position of zero path difference.

(8.10) ##RADATA= (ASDF). Contains interferogram or transformed spectrum as a table of ordinates (A) vs. retardation (R). Each line of ##RADATA= starts with an R-value followed by a series of A-values at equal R spacing in *DIFDU P* form (see Table X, lines 19-22).

9. INFRARED SPECTBA

Tables IIa and IIb have already been cited to show the structure of JCAMP-DX files for infrared spectra. ##DATA TYPE= for infrared spectra is *INFRARED SPECTRUM* .LDRs were defined in Sections 6 and 7, with infrared spectra in mind. The flexibility of parameters defining ##XYDATA=, i.e., ascending vs. descending abscissa, laser vs. rational spacing, and limitation to a single block of spectral data, is intended to accommodate infrared data in known infrared data systems (which represent spectra as a single array of equally spaced ordinates). Flexibility of NOTES is intended to accommodate pure substances, known mixtures, natural products, and unknowns. It is expected that special user groups will define more restrictive subsets of these LDRs for their own purposes. For example, a group, such as the Coblentz Society, might provide a "Guide for Contributors," to specify the order, spacing, precision, abscissa range, etc., for ##XYDATA=, together with required LDRs for physical properties, etc., in the same way that a journal sets forth rules for manuscripts in a "Guide for Authors."

The following are comments on the implementation of Sections 6 and 7.

(9.1) COMPLEXITY OF NOTES. In many cases, *interna l*files will not contain all information required for NOTES. For example, origin and owner of data will not normally be stated explicitly in *interna l* files. *Addition* of such NOTES via text editor is relatively simple, but there is no simple solution to the problem of what to do about NOTES containing information which has no place in *internal* files of the *receivin g*system.

(9.1.1) *ENCODING NOTES* .Users are responsible for supplying NOTES to supplement information from *internal* files. Generation of JCAMP-DX from an *internal* file is like generating a plotted spectrum; the same data are processed, but they are processed by a different algorithm and are sent to a different device. Information which is not contained in *intemal* files is often added to plots from "USE," "MACRO," etc.-files which may even request the operator to supply sample-specific information. The same technique seems applicable to generation of JCAMP-DX files. NOTES can also be inserted with a text editor *after* generation of CORE data (see Section 9.4).

(9.1.2) *DECODING NOTES* .When practical, NOTES should be transferred directly to fields of the *internal* file of the receiving system. These specifications do not attempt to define how to handle NOTES which do not correspond to any *intemal* field, except to point out that NOTES should be accessible to the operator while a spectrum is being displayed or processed, to permit inclusion of selected fields in a plotted spectrum. This can be accomplished by collecting NOTES in a data area which is accessible to a resident mini-editor.

(9.2) SPECTRAL PARAMETERS. The CORE of *simple* JCAMP-DX files can always be generated from *internal* files, regardless of number of points, spacing, etc. This is because JCAMP-DX sets no limits on the parameters which define spectra.

Conversion of CORE data to an internal file of another system is complicated because spectral parameters of the *originating* system may not be compatible with the *internal* file structure of the *receiving* system. Problems due to different *sequence* of the data (increasing vs. decreasing abscissa), different *spacing* (laser vs. rational intervals), and *excessive dynamic range* should be handled automatically by software. When a JCAMP-DX file exceeds the capacity of the receiving system, however, a decision by the *operator* is required (i.e., Should the spectrum be de-resolved to fit available space, or should only part of the data be converted, and if so, what region?, etc.)

(9.3) ASDF FORMS. The apparent complexity of ASDF forms is something of an illusion. *A single* software procedure can decode all forms of ASDF (together with FORTRAN F and E format), taking a cue as to the form of each datum from the data itself. The source can be written in such a way that the procedure will function properly not only when used on *ASCII* systems but also when compiled on IBM mainframes which use the EBCDIC character set (assuming communications via Kermit?).

(9.4) CORRECTING ERRORS IN JCAMP-DX FILES. Software which decodes JCAMP-DX must be on the alert to inform the operator about erroneous data. Errors can arise from misunderstanding of these specifications, from misspelling in manually entered information, from communication problems, etc. Good programming practice requires that the *nature and exact locution* of errors be given. This information, together with knowledge of JCAMP-DX specifications, may allow an error to be corrected with a text editor.

In this regard, a word of warning is necessary. Text editors, and word processors in particular, sometimes use non-ASCII characters to control formatting. Software in higher level languages may not be able to cope with such irregularities. In order that such problems might be prevented, word processors should be set to produce *pure* ASCII text, i.e., in nondocument mode.

(9.5) TESTING JCAMP-DX SOFTWARE.[8] Each new software implementation of JCAMP-DX may err in some way at first. Testing by direct *text* comparison of JCAMP-DX files from different systems is not recommended, because such a comparison can fail due to trivial formatting problems *even though the spectral data are identical.*

Testing involves comparison of the orginal data set with one reconstructed from a JCAMP-DX file, either by subtraction or by plotting of both spectra. The latter method can be used to compare spectra from different systems, requires no knowledge of internal details, and is applicable even when one spectrum is interpolated to different data spacing, as well as when the data have been truncated to remove excess noise. Testing by sub-*traction* is required when sufficient precision for *perfect* reconstruction of spectra is needed.

10. OTHER DATA TYPES

Coates[9] has pointed out the advantages of representing all types of data in the same standard form. He shows data from several techniques plotted from data in JCAMP-DX form using infrared software.

(10.1) Although JCAMP-DX was designed with infrared spectra in mind, it can easily accommodate data from other techniques which generate continuous curves resembling infrared spectra, such as chromatograms; x-ray powder patterns; visible, ultraviolet, and NMR spectra; ESCA; TGA, etc. The same LDRs can be used in many cases.

Techniques in which the data consist mainly of peak position and height, such as mass spectra, atomic emission spectra, etc., can be handled in the same way that infrared peak tables are handled (Section 10.3).

JCAMP is active in many other fields, and considerable interest has been expressed with respect to expanding the scope of JCAMP-DX. This would require development of specifications for units, data types, and use of currently defined LDRs, together with definition of new technique-specific LDRs. User groups and interested individuals are invited to participate in this development for their own fields. Further details are available from the authors.

Sections 10.2-10.5 illustrate extension of JCAMP-DX specifications to four other types of data related to infrared spectra. These vary from Raman spectra (which require virtually no change) to infrared interferograms (which require many new LDRs).

(10.2) RAMAN SPECTRA. Raman spectra are represented in the same way that infrared spectra are, with the use of the same DATA-LABELS (Table IIa and IIb). ##DATA TYPE= specifies RAMAN SPECTRUM, and ##SAMPLING PROCEDURE= contains information on the exciting line, polarization, etc. ##XYDATA= contains the Raman spectrum itself. Ordinate units will normally be arbitrary intensity units. The data are so similar to infrared spectra that the same software can handle Raman spectra simply by allowing ##DATA TYPE= *Raman Spectrum* to be processed.

(10.3) INFRARED PEAK TABLES. Table VIII shows a JCAMP-DX PEAK-TABLE file. Peaks are tabulated as XY pairs under ##PEAK TABLE=. ##FIRSTX= and ##LASTX= define limits of the abscissa region. ##NPOINTS= gives the number of peaks. The spectrum from which ##PEAK TABLE= is derived should be identified as ##CROSS REFERENCE=. PEAK TABLES will require a separate software procedure to read peak data.

(10.4) JCAMP-DX INTERFEROGRAM FILES. Table IX shows a JCAMP-DX file for an interferogram. R represents optical retardation in micrometers and A represents amplitude of the interferogram in arbitrary units. ##DELTAR= is the optical retardation between adjacent points. Normally, ##RFACTOR= is set equal to ##DELTAR=, to scale R to successive integers. ##AFACTOR= contains gain and other factors for proper scaling of transmittance. ##MAXA= and ##MINA= are required because of the large dynamic range in the region of ZPD. ##FIRSTX= and ##LASTX= are abscissa limits of usable data in the spectrum itself. ##NPOINTS= is the number of interferogram points. ##CROSS REFERENCE= links sample and reference interferograms, and possibly the spectrum itself. Data systems with a small word size require a special procedure to read ##RADATA= when the region near ZPD must be handled in double precision; systems with suf-

TABLE VIII. JCAMP-DX PEAK TABLE file.

```
 1. ##TITLE= Epichlorohydrin vapor
 2. ##JCAMP-DX= 4.24
 3. ##DATATYPE=INFRAREDPEAKTABLE
 4. ##SOURCE REFERENCE= #2 (EPA Vapor Library)
 5. ##CASNAME=  l-chloro-2,3-epoxypropane
 6. ##CAS REGISTRY NO= 13403-37-7
 7. ##XUNITS=  l/CM
 8. ##YUNITS= ABSORBANCE
 9. ##FIRSTX= 450.
10. ##LASTX= 4000.          1.912
11. ##NPOINTS= 10
12. ##PEAK TABLE=   (XY..XY)
13.  765.7, 1.912    841.0, 1.180  956.7, 0.966   1134.1, 0.161
14. 1195.9, 0.127   1267.2, 1.076 1402.2, 0.261   1446.6, 0.326
15. 3074.7, 0.896   3082.2, 0.790
16. ##END=
```

TABLE IX. JCAMP-DX file **containing an interferogram.**

```
 1. ##TITLE= Sample interferogram for Epichlorohydrin vapor
 2. ##JCAMP-DX= 4.24
 3. ##DATA TYPE= INFRARED INTERFEROGRAM
 4. ##CROSS REFERENCE= $.$link to reference interferogram
 5. ##RESOLUTION= 4.0
 6. ##RUNITS=  MICROMETERS
 7. ##DELTAR= 1.265757
 8. ##RFACTOR= 1.265757
 9. ##AFACTOR= 1.0
10. ##MAXA= 29552
11. ##MINA= – 27430
12. ##XUNITS=  l/CM
13. ##FIRSTX= 449.413
14. ##LASTX= 4000.355
15. ##NPOINTS= 4096
16. ##ALIAS= 1/1
17. ##ZPD= 254
18. ##RADATA= (R+ +(A..A))
19. 0 -268 38 26 -291 -323 -297 0 31 68 88
20. $$ etc.
21. 4090 4 '270 -275 -266 -268 -278
22. ##END=
```

ficiently large word size can process ##RADATA= with the same procedure that is used for infrared spectra.

(10.5) **TRANSFORMED SPECTRUM.** A JCAMP-DX file for a transformed spectrum (Table X) resembles that for an interferogram. R represents abscissa as a series of sequential integers starting at zero. The transform of a spectrum normally covers only the *spectral region between* ##FIRSTX= and ##LASTX=. ##FIRSTY= is required to define ordinate scaling of the spectrum. ##RFACTOR=, ##RUNITS=, ##ZPD=, and ##ALIAS= are not used, but ##XUNITS= and ##YUNITS= are required. There is the definite possibility that some elements of ##RADATA= will exceed the range of 2-byte integers, so ##MINA= and ##MAXA= are required.

11. *COMPOUND* JCAMP-DX FILES

As indicated in Section 2, these specifications are directed mainly toward *simple* JCAMP-DX files which contain only a *single* spectrum. This restriction is intended to ease the problem of developing software for *internal* files which contain only a single spectrum.

There is growing interest in combining multiple spectra and/or other types of data in a single *compound* JCAMP-DX file. Section 3 provides guidelines for such

TABLE X. JCAMP-DX **file containing transformed spectrum.**

```
 1. ##TITLE= Transformed IR spectrum of Epichlorohydrin va-
    por
 2. ##JCAMP-DX= 4.24
 3. ##DATA TYPE= INFRARED TRANSFORMED SPEC-
    TRUM
 4. ##CROSS REFERENCE= $$link to spectrum
 5. ##RESOLUTION= 4.0
 8. ##AFACTOR= 1.0
 9. ##MAXA= 29552
10. ##MINA= -27430
11. ##XUNITS=  l/CM
12. ##YUNITS=  TRANSMITTANCE        ## FIRSTR= 0
13. ##YFACTOR= 0.001                ## LASTR= 255
14. ##DELTAX= 1.0
15. ##FIRSTX= 600.0
16. ##LASTX=  8 5 5 . 0
17. ##FIRSTY= -268
18. ##NPOINTS= 256
19. ##RADATA=  (R++(A..A))
20. 0 -268 38 26 -291 -323 -297 0 31 68 88
21. $$ etc.
22. 250    -270 -275 -266 -268 -278
23. ##END=
```

files, designed to allow easy separation into *simple* JCAMP-DX files regardless of complexity.

Table XI shows a schematic *compound* file for a GC/IR experiment. The outer BLOCK of Table XI starts with ##TITLE= in line 1, and ends with ##END= in line 21. Line 3 identifies this as a *link* BLOCK. Each inner BLOCK starts with ##TITLE= (lines 6, 11, and 16), and ends with ##END= (lines 10, 15, and 20). ##DATA TYPE= of the first inner block is *Chromatogram* (line 8). ##DATA TYPE= of the second through Nth inner BLOCKS is *Infrared Spectrum* (lines 13 and 18). Thus, inner BLOCKS are identified as *data* BLOCKS, each of which is equivalent to a *simple* JCAMP-DX file.

Table XI. JCAMP-DX file with multiple blocks of data (Schematic).

1.	##TITLE= GCIR Test XYZ	
2.	##JCAMP-DX= 4.24	
3.	##DATA TYPE= LINK	
4.	##BLOCKS= N + 1	
5.	##SAMPLE DESCRIPTION= $$ of original sample	
6.	##TITLE= Chromatogram XYZ	
7.	##JCAMP-DX= 4.24	DATA
8.	##DATA TYPE= Chromatogram	BLOCK
9.	$$ Chromatogram parameters and data	1
10.	##END= $$Chromatogram	
11.	##TITLE= IR Spectrum of Fraction XYZ-1	
12.	##JCAMP-DX= 4.24	DATA
13.	##DATA TYPE= Infrared Spectrum	BLOCK
14.	$$ parameters & spectrum, fraction 1	2
15.	##END= $$ Spectrum	
	– – – – – – – – – –	
16.	##TITLE= IR Spectrum of Fraction XYZ-N	
17.	##JCAMP-DX= 4.24	DATA
18.	##DATA TYPE= Infrared Spectrum	BLOCK
19.	$$ parameters & spectrum, fraction N	N + 1
20.	##END= $$ Spectrum N	
21.	#END= $$ GCIR TEST XYZ	

Structure of *compound* files can be preserved by mapping onto a UNIX-type directory structure, with *link* BLOCKS becoming subdirectories, and *data* BLOCKS becoming files. Otherwise, structure can be ignored by skipping *link* BLOCKS and converting *data* BLOCKS to files. In either case, spectra are processed as *simple* JCAMP-DX files.

This *compound* form is applicable to LC/IR, GC/MS/IR, time or temperature studies, step scans, fluorescence excitation surfaces, two-dimensional NMR experiments, combination of parallel and perpendicular polarized spectra, combination of data for optical constants *(n and k)*, combination of sample and reference interferogram with the corresponding spectrum, combination of spectra used in a subtraction together with the result of subtraction, etc. It is also applicable for archiving all information for a complex separation, including report, spectra, analytical results, etc., in a single JCAMP-DX file.

12. DISCUSSION

(12.1) ORDINATE SCALING. If excessive noise is not stripped before conversion, ASCII representation of spectra can produce excessively large files. DIFDUP form (Section 5.10) is intended to minimize this problem, but even in this mode one extra digit of pure noise can increase the size of a 4000-point JCAMP-DX file by several thousand bytes! The size problem is of particular concern for *internal* files in which the data have been *scaled* to fill a large binary word such as IEEE real number format. Software for generating ##XYDATA= *requires* a user option for stripping excess noise in such cases.

(12.2) OWNERSHIP OF FILES. When copyright or proprietary files are converted to *internal* form from JCAMP-DX, *ownership* information may be lost. To guard against reconversion to JCAMP-DX form and subsequent distribution without the ownership notice, one should save the JCAMP-DX file rather than the *internal* file.

(12.3) EXCHANGING JCAMP-DX SPECTRA. Concurrently with development of JCAMP-DX specifications, rapid strides have been made in communications. Local and wide area networks, computer bulletin boards, and computer mail can all be used to exchange spectra in JCAMP-DX form. Last, but not least, standardization of floppy disks for PC-DOS/MS-DOS systems has simplified exchange of spectra by mail.

(12.4) USES of JCAMP-DX. A major force behind JCAMP-DX is the need for common use of infrared spectra in organizations which employ more than one make of infrared spectrometer. A related factor for organizations with a single make of spectrometer is the need for freedom to consider purchase of other makes, on the basis of performance rather than data compatibility.

The combination of properties of JCAMP-DX makes it a candidate for a number of special applications. However, it should be recognized that JCAMP-DX is not designed for rapid searching of very large databases, which requires random access and avoidance of operations such as ASCII-to-binary conversions.

The unlimited precision of JCAMP-DX can be useful for round-robin committee work on interchangeability of spectral data among different spectrometers, for testing algorithms for manipulating data, for development of software, etc. Ability to combine unlimited amounts of descriptive information with spectral data can be useful in record keeping.

Publication of accurate digital spectra can also be facilitated. Here, the term *publication* refers to making information available to the scientific community at large. When the effort has been invested in preparing and characterizing a pure compound or a natural or a commercial product and measuring its spectrum accurately, the data should be able to be shared with others. The form of publication could be digital media, ranging from laser disks to on-line databases, rather than conventional journals. DX is seen as a form for submitting such contributions and for transmitting spectra to end users.

(12.5) PERSONAL COMPUTERS AND JCAMP-DX.

It seems likely that before long, personal computers will be, if they are not already, a major factor in the processing and interpretation of spectral data. Ready exchange of data between spectrometer data systems and personal computers is exciting, because it promotes *standurdizution of the software environment.* Already, personal computers have become the frontier in development of new languages and data processing techniques. Extension of *data* standardization to other laboratory techniques will promote interaction of different scientific disciplines through use of the same common software.

1. F. da Cruz, *Kermit, A File* Transfer Protocol (Digital Press, Bedford, Massachusetts, 1986).
2. "The Coblentz Society Specifications for Evaluation of Research Quality Analytical Infrared Spectra (Class II)," Anal. Chem. 47, 2339 (1975).
3. W. B. Derendjaev, Appl. Spectrosc. 40, 707 (1986).
4. Elbert G. Smith, *The Wiswesser Line-Formula Chemical Notation* (McGraw-Hill, New York, 1968), 2nd ed.; and (Chemical Information Management, Inc., Cherry Hill, New Jersey, 1975), 3rd ed.
5. *Atlas of Spectral Data* and *Physical Constants for Organic Compounds,* J. G. Grasselli and W. G. Ritchey, Eds. (Chemical Rubber Co., Boca Raton, Florida, 1975), 2nd ed.
6. A. J. Lawson, "Structure Graphics," in *Pointers to Beilstein Out,* ACS Symposium 341, *Graphics for.Chemical Structures-Integration with Text and Data,* W. Warr, Ed. (ACS, Washington, D.C., 1987), pp. 80-87. STN International is an online information network operated cooperatively by ACS, West German, and Japanese oraanizations. Chem. and Eng. News, July 13, 1987, pp. 30-31.
7. J. G. Grasselli, P. R. Griffiths, and R: W. Hannah, Appl. Spectrosc. 36, 87 (1982).
8. IBM PC disks containing test spectra in a variety of JCAMP-DX forms are available.
9. John P. Coates, Spectroscopy 2, 14 (1987).