Fuzzy date management in military history databases

FZY-D, a proposed variation on the ISO 8601:2004(E) datetime format

"A rich vocabulary for precisely describing imprecise dates"

January 2016 version 2.0

MGT-WP-01

MGT-WP-01: Fuzzy date management in military history databases
FZY-D: a proposed variation on the ISO 8601:2004(E) datetime format

MGT-WP-01: Fuzzy date management in military history databases – FZY-D: a proposed variation on the ISO 8601:2004(E) date-time format.

Notes

This document was created in Adobe InDesign. The fonts are Gill Sans and Courier New, set at 10pt/14pt. Margins have been set to make this document printable on both A4 and US Letter paper.

Version History

v1.0 Jul 2009	Original text, not released
v1.1 Feb 2010	Revised for clarity, with some additional material
v1.2 Apr 2010	Minor revisions and additional material
v1.3 Jan 2011	Revised to include discussion of Allen Operators and rethink some previous Qualifier Codes
v1.4 Jul 2012	Revised FZY-D format for consistency
v1.5 Feb 2013	Further refinement
v1.6 Apr 2014	Orientation and Confidence Qualifier positions swapped over
v2.0 Jan 2016	Various revisions and additional material, Bibliography

MacGregorTech
3b Glendower Place
South Kensington
London SW7 3DU
United Kingdom

Website: www.macgregortech.com
Email: fzyd@macgregortech.com
Telephone: +44 (0)7803-163 938

Contents

Fuzzy date management in military history databases

FZY-D: a proposed variation on the ISO 8601:2004(E) datetime format

PART ONE – FUZZY PROBLEMS

1. Introduction

1.1 Types of fuzzyness

The author is a database and web developer, technical author and amateur military historian, with over 30 years of experience in the IT business and a particular interest in managing data related to World War Two British Special Forces. One of the problems encountered by the author with recording and analysing military history data is that the dates associated with people, events, documents and images etc., are often imprecise (the Precision Problem), with a wide variation in the nature and extent of that imprecision, sometimes also covering intervals[1] rather than specific points in time. Related to this (but a different issue) is what might be called the Granularity Problem.

This is further complicated by variation in the levels of confidence in the accuracy of those dates (the Confidence Problem), and again by some dates needing to look relatively nearer or further in time with qualifying phrases such as "*or later*" and "*week beginning*" (defined here as the Orientation Problem).

Together these issues create a 'date fuzziness' that does not seem to be addressed by current date and time data formatting standards. In fact this problem appears to be considered as such an inconvenience to many in the IT world that they wish it would just go away rather than stand up and confront it properly. In the author's view this is a significantly short-sighted technical and commercial mistake.

Fuzzy dates are not new in historical research (see the Bibliography), but a general trawl of the internet suggests that few (J. F. Allen excepted) have systematically tackled the problems around date fuzziness. There is even some fuzzy thinking as to what constitutes a fuzzy date: a few technical forum discussions, purporting to be about fuzzy dates, have been found regarding how to calculate intervals that could result in displays like "*nearly 4 hours ago*" or "*last week*" or "*next Wednesday*". These, it seems to the author, are not so much about fuzzy dates as about fuzzy descriptions of *very precise but relative* dates, which are not the same thing.

This White Paper, then, attempts to tease out, examine and clarify many underlying aspects of these issues, and proposes an adaptation of an existing standard to allow, paradoxically, a precise and concise way of storing, manipulating and displaying vague historical date data through a comprehensive conceptual yet practical framework.

Although this is primarily designed to address the author's technical problems in his personal area of interest, some account has been taken of general historical contexts, and one purpose of this paper is to draw out wider and deeper thinking, knowledge and expertise.

There is no attempt here to prescribe how the proposed format could or should be implemented in any programming language or database system, as these topics are better left to specialists in their fields. The author has, however, developed software that does implement the proposed fuzzy date format, and has used it for many years. This practical experience informs much of the thinking laid out here. To start with, two examples may help to illustrate the precision and confidence problems.

---

1 The word "interval" is used throughout instead of "period" to indicate a date or time range, as this conforms with their very specific meanings within the ISO 8601 specification.

1.2 The Precision Problem

A published memoir briefly describes a minor World War Two skirmish behind the German lines in the North African desert, during which a soldier is injured and leaves the special unit to which he is attached. The unit's official War Diary for that month is missing from The National Archives though, possibly lost in the unit's later move from Tunisia to Italy. There is no doubt about in which month the incident happened and therefore roughly when the soldier left, but the actual day is not known, pending further research, and the event date could be recorded in several narrative ways…

'*definitely sometime during* February 1943', or
'*definitely sometime between (and including)* 1 February 1943 and 28 February 1943', or perhaps
'*almost certainly* 12th February 1943, *give or take 7 days*'.

The confidence level is high but the date is imprecise, and the preference would be to represent this event in a database with an imprecise, but qualified, date rather than as a precise interval, which could give it a subtly different meaning. This is relevant to investigating the sequence and significance of other events nearby in time and space.

1.3 The Confidence Problem

A grainy, scratched, out-of-focus, dog-eared black and white photograph shows five grinning soldiers dressed in skiing gear on a snowy wooded slope, but nothing is written on the back. The photo is loose in a box owned by the widow of a soldier who is not in the picture, but a mate of those who are, so the unit is known. But who are his pals, and where and when was the picture taken?.

From the unit's War Diary it's known that only two mountain training exercises were held; both in Italy and roughly one year apart, with definite start and end dates for each exercise. Initially the photo could only be identified as 'Five unknown men in either Piedemonte d'Alife or Terminillo in Italy, *definitely sometime between (and including)* 19 March 1944 and 3 April 1945'. Later re-examination of the photo positively identifies two of the men and probably a third. From personnel records it's discovered that one of the identified men apparently only joined the unit in July 1944, so the photo must be from the second exercise, in Terminillo, narrowing down (a) the interval to 8 March 1945 to 3 April 1945, and therefore (b) the possible identities of the other three men.

However, further research establishes that the personnel records are not always reliable: sometimes the paperwork catches up with the person weeks or even months after they joined the unit. Then comparison with annotated photos in another collection shows that pictures taken around Terminillo are in sharper focus, with mountains rather than gentle hills in the background, so the conclusion is that it may be from the first exercise after all, and two of the unidentified men could not then be X and Y as suggested by a colleague.

The photo could be narratively dated therefore as '19 March 1944 *or later* and *probably (or, almost certainly) not later than* 30 April 1944', or alternatively, '*probably during the period of* 19 March 1944 to 30 April 1944'. These interval dates are quite precise but there is not an absolute confidence about one or both of them, there are orientation issues, and there are various ways to describe all that fuzziness as the photo goes through its many stages of dating analysis.

1.4 Date and time data types

How are these fuzzy dates and their qualifying remarks of precision, orientation and confidence to be codified and stored in a database in such a way as to be easily, meaningfully and consistently searched for, sorted on, displayed and shared with others?

Date, time and datetime standard data types in the main database systems and in common spreadsheet software are often strictly numeric, stored as the number of seconds elapsed since an arbitrary base position such as 1 January 1904 or 1970. Although this allows great precision, efficient storage, fast processing, convenient date mathematics and strict sorting rules, it is not much use for recording the dates of birth of senior army officers in World War Two (i.e. possibly in the 19th Century), and cannot handle dates that may only be known to within one month or worse.

Any coded qualification of such a date, even if numeric, also has to be handled in a separate field rather than being incorporated directly as an integral part of the datetime (i.e. to simplify searching and sorting). Other standard data types (as opposed to user-defined data types) may have an alphanumeric representation but often have very strict data entry rules that in fact only allow precise[2] content.

These kinds of problems led the author to investigate and experiment with alternatives, not as a datetime expert, but merely as an end-user seeking a solid rather than fudged solution. What follows is a discussion of the perceived issues and their implications, conclusions drawn from research and practice, and proposals that are designed to help data sharing beyond the author's private research area. It is a thorough revision of an earlier implementation that was not quite strong enough to support use by the wider public.

1.5 Meanings and definitions

In this paper "date" generally includes "time", so means "datetime" unless otherwise specified, to allow a less tedious narrative.

As no statements have been found as to what constitutes a precise date, let alone a fuzzy one – only general background assumptions that these are self-evident – here is an attempt at some working definitions : –

" A PRECISE DATE IS A POINT IN TIME WHERE ALL OF ITS RELEVANT COMPONENTS ARE KNOWN AND UNQUALIFIED "

The inclusion of 'relevant' gets us out of the trap of implying that a date without a time is therefore fuzzy and, similarly, a date that only has year and month components[3] may be perfectly accurate and precise in its context.

An interval therefore is : –

" A PRECISE INTERVAL COMPRISES TWO POINTS IN TIME WHERE ALL OF ITS RELEVANT COMPONENTS ARE KNOWN AND UNQUALIFIED "

A fuzzy date definition follows on from that then, fairly smoothly : –

" A FUZZY DATE IS A POINT IN TIME WHERE SOME OR ALL OF ITS RELEVANT COMPONENTS ARE KNOWN BUT QUALIFIED, OR ARE UNCERTAIN OR UNKNOWN "

Again, intervals seem to be catered for. These definitions though are entirely separate from any textual description of a date (fuzzy or not), and fuzzy *relative* date descriptions, as noted above, could be derived from either precise or fuzzy dates, but we are looking here for unambiguous codification of inherently fuzzy dates. This definition of a fuzzy date underlies all of the thinking in the following sections.

---

2 Where a "precise" date is one that includes year, month and day and a "precise" time includes hours, minutes and seconds.
3 The word "component" is used specifically to conform with ISO 8601 terminology..

1.6 Background

The author specialised, for about twenty years, in developing cross-platform (Mac and PC), client-server, multi-user desktop applications using a Rapid Application Database Development System called 4D[4]. Trained originally as a Systems Analyst and Architect on mainframes, the author has designed many desktop applications, and developed, debugged or enhanced them in a wide variety of business sectors (banking, medical, logistics, entertainment etc.) for many small, medium and large organizations. One the earliest adopters of desktop-publishing in the UK, the author has also designed and written technical and historical papers, newsletters, press releases, manuals, user guides, release notes, web articles and obituaries.

Privately the author maintains a small database, written in 4D, to hold some WWII Special Forces veterans' personnel details as well as contact details for relatives and other interested parties. It was in the management of this personnel database and related historic event, image and document analysis and recording that the problems of how to store and display fuzzy dates were first encountered and solutions devised.

The author's original 4D-based format for fuzzy dates, developed in 2004, is slightly shorter than that proposed here, but its essential elements have been in practical use since then. In 2009 a complete rethink of that format was started, in light of experience, resulting in this White Paper, itself slowly refined and expanded over the last few years.

That original suite of fuzzy-date functions have now been completely redeveloped from scratch to the new, more generic, standard. They have then been refined and thoroughly tested in 4D, to convert data between conventional and this latest fuzzy date format as well as tightly validate input and display the results.

More recently the author has moved into open-source web development, using the Drupal content management system, and developed a Special Forces military history and remembrance website[5], partly to bring that 4D database content to the wider public.

The next step will be to convert these functions, and develop others, in Drupal (PHP), to enable this data to be managed in an open-source web environment, and perhaps allow standardised fuzzy-date data interchange with other similar organizations, probably using the JSON format.

This is the driver for this White Paper: to fundamentally and comprehensively offer a solution to many, if not all, fuzzy date management problems and share the results with like-minded others.

R.M.Paterson
January 2016

4 *www.4d.com*
5 *www.popski.org*

PART TWO – DATETIME FORMATS

2. The ISO 8601:2004(E) datetime format

2.1 Introduction

The most rational starting point for the formatting of fuzzy dates is perhaps the current International Standard for date and time representation and interchange, ISO 8601:2004(E)[6] (from here on simply referred to as ISO 8601), and this paper assumes that the reader is conversant with its details, so its contents are not repeated here except for the final paragraph of the Introduction to the Standard : –

" *To avoid confusion between the representations and the actual text, its punctuation marks and associated graphic characters, all the representations are contained in brackets [ ]. The brackets are not part of the representation, and should be omitted when implementing the representations. All matter outside the brackets is normal text, and not part of the representation. In the associated examples, the brackets and typographical markings are omitted.* "

2.2 ISO 8601

Although ISO 8601 does appear to cater for fuzzy dates, in its definition of "*Representations with reduced accuracy*" *(section 4.1.2.3)*, in practice this 'representation-by-omission' leaves room for ambiguity of meaning. While a date can be fuzzy, the nature of that fuzziness should still be as precise as possible. For example[7] : –

`1985-04-12` clearly this is 12th April 1985…
`1985-04` …but does this mean 'April 1985' or 'a specific day in April 1985, not yet known'?

While a solution to this problem, like `1985-04-00`, might sort of work for a date, it doesn't deal with the difference between "*don't know, but wish I did*" and "*actually not needed, thank you*", and falls down immediately with a time, where `T12:30:00`, for instance, already has a specific meaning. What is proposed, to manage this and other problems, is a variation of, and extension to, the 20-character ISO 8601 extended date format (and to the 41-character ISO 8601 extended interval format), including time zone : –

`YYYY-MM-DDThh:mm:ssz`	ISO 8601 "extended" date format (this is what the "(E)" bit means)
`1985-04-12T12:34:06Z`	example

`YYYY-MM-DDThh:mm:ssz/YYYY-MM-DDThh:mm:ssz`	ISO 8601 extended interval format
`1985-04-12T12:34:00Z/1985-05-06T17:45:00Z`	example

Note that the extent of these base formats' level of precision goes as far as, but no further than, seconds, and this is deliberate. Although the author's experience of recording military history only extends to minutes (for example: "*The time-on-target bombardment started at precisely 0355 hours*"), the seconds time component is included for balance and completeness (compared with the date components) and because experience says that someone somewhere will have a use for it. Precision beyond that, into decimal parts of seconds, is not catered for at all in this paper, but is by no means inherently excluded from the following proposals.

6 *www.iso.org/iso/iso8601*
7 *The examples shown throughout are not necessarily displayed in the full ISO 8601 or FZY-D formats, for clarity.*

3. The FZY-D *datetime format*

3.1 Introduction

The ISO 8601 standard explicitly allows for some variation, subject to "…*mutual agreement of the partners in information interchange." (section 3.7),* and the proposed new format is built very firmly upon those foundations.

Starting with 'the naming of the parts' it is proposed that this new format is identified by the name '**FZY-D**' (meaning Fuzzy Date, pronounced *fuzzy-dee*), with the suffix included to distinguish it from a possible, complementary '**FZY-L**' format (meaning Fuzzy Location, pronounced *fuzzy-ell*), for managing uncertainty in geolocation data (see ISO 6709). More formally it could possibly be called '**FZY-D (ISO 8601)**' to indicate its parentage without claiming that it's a ratified standard. It probably cannot be called ISO 8601 (FZY-D) because, as you will see, it does not merely extend the standard but also modifies some of its basic elements. Collectively, this 'fuzzy format family' could be called '**FZY-X**' (pronounced *fuzzy-ex)*.

Drawing on long experience and accepted good practice, it was decided that the ground rules for this FZY-D format were that it should aim to be : –

- precise – in the sense of precision of meaning
- explicit – if a component is not precisely known, this should not be just implied by its absence
- unambiguous – precision of meaning again
- comprehensive – to cover as wide a range of circumstances as reasonably possible
- adaptable – the format should allow for usage not envisaged by the author
- common-sense – use of symbols whose meaning is as clear, obvious or intuitive as possible
- human-readable – using extended format only, for ease of understanding the raw data
- sort-sensible – fuzzy dates and precise dates should group together in rational ways
- fixed-length – to help searches and manipulation with Regular Expressions, etc.
- ASCII friendly – all special symbols should be above ASCII 32 and below ASCII 128, where possible

3.2 FZY-D

It is proposed that the FZY-D format should extend the 20-character ISO 8601 base format by a further 10 characters, so that a 30-character FZY-D Datetime comprises the following parts : –

FZY-D Date – 10 characters as in an extended format ISO 8601 date, including hyphen separators
FZY-D Time – 9 characters as in an extended format ISO 8601 time, including 'T' prefix and colon separators
FZY-D Zone Designator – 1 character as per the ISO 8601 Time Zone Designator, but with extra symbol options
FZY-D Era Designator – 3 characters, plus prefix hyphen separator (represented as '-eee')
FZY-D Orientation Qualifiers – 2 characters, plus prefix hyphen separator (represented as '-oo')
FZY-D Confidence Qualifiers – 2 characters, plus prefix hyphen separator (represented as '-cc')

…but also including…

FZY-D Precision Indicators – substitute characters in various positions to show imprecision.

The extra 10 characters and the Precision Indicators are all explained in the following sections.

Here then is the FZY-D format with a typical example :–

```
YYYY-MM-DDThh:mm:ssz-eee-oo-cc   FZY-D date format
1985-04-12T12:34:00Z-CE#-<< -?#  example
```

A FZY-D Interval is simply the concatenation of two FZY-D Dates to give a 61-character string, including the standard ISO 8601 solidus [/] separator (but see PART THREE – FORMAT EXTRAS) :–

```
YYYY-MM-DDThh:mm:ssz-eee-oo-cc/YYYY-MM-DDThh:mm:ssz-eee-oo-cc  FZY-D Interval format
1985-04-12T12:34:06J-AD#-##-**/1985-05-06T17:45:00J-AD#-##-..  example
```

The fixed-length requirement follows naturally from other ground rules: leaving out any component would fatally conflict with unambiguity and human-readability, as well as with searching and sorting. Storage cost-per-byte ceased to be an issue a long time ago. Similarly, the space character (decimal ASCII 32) would not be allowed anywhere in the FZY-D format, because it would also break one or more of the ground rules. All alphabetic characters would be uppercase only.

It is recognised that 'FZY-D Date' and 'FZY-D Datetime' would probably come to be used interchangeably, diluting the narrower definition of 'FZY-D Date' above – such is life – but would then usefully contrast with the term 'FZY-D Interval'.

The component parts of the FZY-D format are explained and examined in detail in the sections that follow, but first it is necessary to introduce the FZY-D Precision Indicators, because these go to the heart of the FZY-D concept.

3.3 FZY-D Precision Indicators

FZY-D Precision Indicators are the substitute characters for the FZY-D Date, FZY-D Time and FZY-D Zone parts that make it possible to explicitly show which components of a FZY-D Datetime are imprecise. They are also used, identically or similarly, in the FZY-D Era Designators and the FZY-D Orientation and Confidence Qualifier parts.

Only one or the other of two characters (usually in pairs) are needed to indicate precision and get away from 'representation-by-omission' in ISO 8601 :–

Chars	*Decimal ASCII*	*Meaning*
##	35 + 35	Irrelevant, or not applicable
??	63 + 63	Relevant but unknown

For example :–

```
1985-04-##   April 1985, day not relevant
1985-07-??   July 1985, but on what day?
```

The following sections shows how these Precision Indicators are implemented in each of the component parts of a FZY-D Date.

3.4 FZY-D Dates

A FZY-D date element is generally exactly the same as a standard ISO 8601 date element except only for the use of Precision Indicators (and some special variations that are explored in PART THREE – FORMAT EXTRAS). A key requirement of fuzzy date handling for many users is "granularity" – the ability to manage dates at the year-only level or year-and-month level. However, this seems to be a strictly hierarchic concept, whereas 'precision' here is not, and the two should not be confused. FZY-D Precision Indicators offer both hierarchic granularity and more subtle representations of fuzziness in dates.

For example, granularity would look like this : –

`1985-04-##`	April 1985, day not relevant
`1985-##-##`	1985 – a year
`198#-##-##`	The 1980s – a decade
`19##-##-##`	20th Century
`1###-##-##`	2nd Millennium

…but there is nothing to say that a date couldn't be represented like this : –

`1985-##-25`	The 25th of every month in 1985
`1985-??-25`	A day in 1985 but the month is not yet known (for instance, the source document could be damaged)

…allowing an unambiguous, precise way of recording imprecision, and clearly discriminating between "*don't know, don't care*" and "*hmmm, wish I knew*".

3.5 FZY-D Times

Similarly with a FZY-D time element : –

`1985-04-12T12:34:##`	12.34pm on 12 April 1985, seconds irrelevant
`1985-04-12T12:34:00`	This is also 12.34pm on 12 April 1985, of course (standard ISO 8601)
`1985-04-??T##:##:##`	Unknown day in April 1985, time not relevant
`1985-04-12T00:00:00`	Midnight at the start of 12th April 1985 (standard ISO 8601)
`1985-04-12T??:??:??`	12 April 1985, time unknown but relevant down to the seconds
`????-04-12T##:##:##`	Only the month and day are known

Either of the Precision Indicator characters could be used in any and all of the date and time numeric positions, not excluding : –

`????-??-??T??:??:??`	Date (i.e. datetime) is completely unknown
`####-##-##T##:##:##`	Date is completely irrelevant (someone will surely have a use even for this)

However, mixing or part-using FZY-D Precision Indicators within a component would not be allowed : –

`1985-04-?#T12:34:00`	Not allowed (character positions 9 and 10) – whatever would it mean?

3.6 FZY-D Zone Designators

The 'FZY-D Zone' Designator character is standard ISO 8601, in character position 20.

This symbol set comprises the full complement of all 25 NATO main time zone designators, with 'Z' (that is, UTC) as the default[8]. The 26th character is also one of the FZY-D Zone Designators : –

Chars	*Decimal ASCII*	*Meaning*
J	74	Local Time – this is the NATO designator for an observer's current local time

1985-04-12T12:34:##**J**	12.34pm local time
1985-04-12T12:34:##**R**	12.34pm New York time (UTC -5 hours: Eastern Standard Time)

In addition though, either of the two FZY-D Precision Indicators could be substituted : –

1985-04-12T12:34:##**?**	12.34pm, time zone unknown
1985-04-12T12:34:##**#**	12.34pm, time zone irrelevant

Consideration was given to how to represent time concepts like 'Double Summer Time' (as was the case in Britain during World War Two), or 'Pacific Daylight Time', possibly by using single lower-case characters, numbers, or by adding more characters to the FZY-D Zone code, but nothing was concluded. These are probably best handled by computation from date and location data, and the 'Local Time' concept helps us out of that trap.

3.7 FZY-D Era Designators

The Era element of the FZY-D format is the first part of the FZY-D extension to standard ISO 8601. It is also the most obvious departure from the author's narrow WWII historical view, but would seem to be quite logical to include, and differs from the use of the negative sign in ISO 8601 for years before year 0000 in the Common Era.

ISO 8601 is of course based upon the Gregorian Calendar, and the four digits of the year component, along with the Era Designator codes, limit us to a 20,000 year timespan (-2), but surely this is adequate for most of general as well as military history. In any case, ISO 8601 allows extensions to the year characters, by mutual agreement, and the intention in this paper is to get to a basic and baseline agreement on historical fuzzy date encoding.

It is proposed that one of five 3-character FZY-D Era Designator codes is used in character positions 22 to 24 (preceded by a hyphen separator in character position 21) : –

Chars	*Decimal ASCII*	*Meaning*
AD#	41 + 44 + 35	Anno Domini
BC#	42 + 43 + 35	Before Christ
BCE	42 + 43 + 45	Before Common Era
CE#	43 + 45 + 35	Common Era

…plus…

8 *www.timeanddate.com/library/abbreviations/timezones/military/*

###	35 + 35 + 35	Irrelevant, or not applicable (but probably defaulting to AD/CE of course)
???	63 + 63 + 63	Relevant but unknown

…derived from the FZY-D Precision Indicators (see **3.3** above).

For example, in 20 + 4 characters : –

`1985-04-12T12:34:##Z-CE#` 12.34pm GMT on 12 April 1985 in the Common Era

The 3-character format also allows ample room for further FZY-D Era Designator codes such as `VIC` for "Victorian", or "`WW2`" for instance.

3.8 FZY-D Orientation Qualifiers

Whereas *irrelevance* and *unknownness* are about levels of precision and can be represented as substitute characters within the FZY-D format (the FZY-D Precision Indicators – see **3.3** above), *uncertainty* is about levels of confidence. The representation of uncertainty is surely a central part of fuzzy date management.

Qualification of a date, regardless of its precision, cannot be adequately managed by further substitute characters, so must be encoded separately. However, research and experimentation with concepts of date uncertainty show that they can be split into the two sub-categories of 'confidence' and 'orientation'.

The word 'orientation' was chosen deliberately, after considering and discarding 'breadth', 'width', 'range', 'scope' and 'direction' as not quite conveying the appropriate concept. Please note that the Orientation Qualifiers come before the Confidence Qualifiers because "Orientation" is specifically about dates whereas "Confidence" is a more general concept.

Date *orientation*, then, is exemplified by words and phrases such as '*before*', '*or later*' and '*week ending*', and the following eleven 2-character FZY-D Orientation Qualifier codes are proposed for character positions 26 and 27 (preceded by a hyphen in position 25) : –

Chars	*Decimal ASCII*	*Meaning*
<#	60 + 35	Week ending[9]
<<	60 + 60	Before
<=	60 + 61	Or earlier
<>	60 + 62	Mid
=<	61 + 60	Early
=>	61 + 62	Late
>#	62 + 35	Week beginning
>=	62 + 61	Or later
>>	62 + 62	After

…plus…

##	35 + 35	Irrelevant, not applicable
??	63 + 63	Relevant but unknown

9 'Week ending' is relevant to recording data about WWII British Army Forms W3008 and W3009 "Field Returns of Other Ranks" and 'Field Returns of Officers'.

…exactly as in the FZY-D Precision Indicators, so here [##] *does* mean 'Irrelevant', and [??] *does* mean 'Unknown'.

For example, in 20 + 7 characters : –

`1985-04-12T12:34:##Z-AD#->=`	12:34pm on 12 April 1985 (or later)
`1985-04-12T##:##:##Z-AD#-<<`	Before 12 April 1985
`1985-04-12T##:##:##Z-AD#-<#`	Week ending 12 April 1985
`1985-04-##T##:##:##Z-CE#-<>`	Mid April 1985
`19##-##-##T##:##:###-###-=>`	Late 20th Century

3.9 FZY-D Confidence Qualifiers

Date confidence is exemplified by words and phrases such as '*probably*', '*approximately*' and '*to be confirmed*', and the following eight 2-character FZY-D Confidence Qualifier codes are proposed for character positions 29 and 30 (preceded by a hyphen separator in position 28) : –

Chars	*Decimal ASCII*	*Meaning*
`##`	35 + 35	Unqualified
`**`	42 + 42	Definitely
`*#`	42 + 35	Almost certainly
`..`	46 + 46	To be confirmed
`?#`	63 + 35	Probably
`??`	63 + 63	Maybe
`~#`	126 + 35	Approximately, circa, around
`~~`	126 + 126	Very roughly

…but note that here, [##] means 'Unqualified', not 'Irrelevant', and [??] means 'Maybe', not 'Unknown': further reasons why the format is fixed-length, for unambiguous positional meaning. Conveniently there is no clash between the rising values of the ASCII numbers and the falling levels of confidence, for sorting purposes.

In addition, perhaps there should be two more Confidence Qualifiers to handle situations that are just too difficult to convey with any of the other codes : –

Chars	*Decimal ASCII*	*Meaning*
`+!`	43 + 21	Known to be an invalid date! (e.g.: "31st February" in a source document)
`++`	43 + 43	Careful, see notes!

Conventional date format software is not so much incapable of handling invalid dates as specifically designed to prevent them being included – a classic example of the technical world forcing the real world to adjust rather than the other way round.

So, now we have the full 30-character FZY-D Date format. For example, in 20 + 10 characters : –

`1985-04-12T12:34:##Z-AD#-##-~#`	approximately 12:34pm on 12 April 1985
`1985-04-12T##:##:##Z-AD#-##-?#`	12 April 1985 (probably)

`1985-04-??T##:##:##Z-CE#-##-..`	unknown day in April 1985 (to be confirmed)
`1985-04-12T##:##:##Z-AD#-##-##`	12 April 1985
`196#-##-##T##:##:###-###-##-~#`	Around the 1960s
`1960-##-##T##:##:###-###-##-~#`	Circa 1960
`1945-09-##T##:##:###-###-##-~~`	Very roughly September 1945
`1066-??-??T##:##:##?-AD#-##-++`	Needs more research, did anything happen that year?
`1985-02-31T##:##:##Z-AD#-##-+!`	Yes I know it's wrong but that's what's in the records
`1985-04-12T12:34:##Z-AD#->=-##`	12:34pm on 12 April 1985 (or later)
`1985-04-12T##:##:##Z-AD#-<#-~#`	Approximately the week ending 12 April 1985
`1985-04-##T##:##:##Z-CE#-<>-??`	Maybe mid-April 1985?
`19##-##-##T##:##:###-###-=>-**`	Definitely late 20th Century

The ability to explicitly handle dates that are actually invalid is arguably one of the best features of this new date format.

It's recognised that some combinations of FZY-D Precision Indicators and FZY-D Confidence Qualifiers (as in one example above, with [??] in the day position and [..] as the uncertainty qualifier), might need to be proscribed as redundant or too confusing to be meaningfully interpreted, but these are issues of implementation rather than format. There is also a subtle limitation here: it's not possible to indicate an approximate date with a definite time (or vice versa) – it is all or nothing, and probably doesn't much matter anyway (unless, of course, someone knows better).

One implementation issue is whether the displayed descriptive text should prefix or suffix the date itself. See PART THREE – FORMAT EXTRAS for further discussion of the implications of this.

The alert reader will notice that there might be an overlap of meaning or concept between some of the FZY-D Orientation and FZY-D Confidence Qualifiers (for example: 'Approximately' and 'Mid'), but it is felt that in practice these sub-categories are different enough to separate them like this, and that either way they provide a rich vocabulary and great flexibility to fuzzy date encoding and description, without significant loss of precision in meaning. How this information is stored and how it is displayed are separated, like HTML and CSS.

PART THREE – FORMAT EXTRAS

4. Other considerations

This part considers some exceptions to and variations of the core format proposals and examines issues that don't fit elsewhere. It should be noted that foreign language and non-Gregorian/Julian calendar matters are not considered at all, though they are certainly not inherently excluded from further adaptations to the FZY-D format.

4.1 Special format variations

The FZY-D format also allows for some special variations (not showing all 30 characters, for clarity) : –

`1985-H1-##` 1st Half of 1985
`1985-H2-##` 2nd Half of 1985

`1985-Q2-##` 2nd Quarter of 1985
`1985-Q3-##` 3rd Quarter of 1985

…and although standard ISO 8601 already caters for Week Numbering, it does so by using a special short form that may not integrate well with more regular precise and fuzzy dates, so perhaps this variation could be handled like this : –

`1985-WK-01` First Week of 1985
`1985-WK-11` 11th Week of 1985
`1985-WK-29` 29thWeek of 1985
`1985-WK-53` Last Week of 1985

These concepts could be extended further to handle the seasons as loose date-groups too, including solving the tricky problem of 'winter' crossing a year boundary at one end or the other : –

`1985-SG-##` Spring 1985
`1985-SR-##` Summer 1985
`1985-AN-##` Autumn 1985 (or `-FL-` Fall)
`1985-W<-##` Winter 1984/5
`1985-W>-##` Winter 1985/6

…or maybe, for rational sorting…

`1985-S1-##` Winter 1984/5
`1985-S2-##` Spring 1985
`1985-S3-##` Summer 1985
`1985-S4-##` Autumn 1985
`1985-S5-##` Winter 1985/6

…where `1985-S5-##` and `1986-S1-##` are the same thing of course.

4.2 Searching, sorting and date mathematics

Quite a lot of thought went into the selection of the FZY-D substitute, indicator and qualifier characters with regard to the implementation issues of searching and sorting. The FZY-D Era codes remain a problem though, and two other practical issues surfaced when the author was transcribing a unit's War Diary into a database using the original version of the FZY-D format :–

- should FZY-D Intervals sort before or after FZY-D Dates?
- should dates containing fuzzy components sort before or after precise dates?

For example, although one might think that a War Diary simply has only one slot per day, in practice the author has seen data something like this (format: dd.mm.yy) :–

```
3.5.44      this happened to unit company A
4.5.44      that happened to unit companies A and B
5.5.44      unit HQ moved to location X
1st - 7th   unit company C away on training exercise
```

…and is in favour of being able to maintain that sequence as found in the original data. Using the FZY-D format as proposed would, of course, sort the interval (in the fourth line above) to the top :–

```
1944-05-01T##:##:##Z-AD#-##-##/1944-05-07T##:##:##Z-AD#-##-##
1944-05-03T##:##:##Z-AD#-##-##
1944-05-04T##:##:##Z-AD#-##-##
1944-05-05T##:##:##Z-AD#-##-##
```

Consideration was therefore given to allowing alternative characters, above or below the decimal ASCII range of 30 to 39 (the numbers 0 to 9), to the proposed 'irrelevant' [#] and 'unknown' [?] characters. However, this is probably not best handled as a format issue, not least because it's essentially a complicating fudge, and using another field to manage sequence is probably better for allowing users to define how to group fuzzy and non-fuzzy dates.

Relatedly and in a regular regiment's War Diary, within the same day different squadrons have separate time-sequences, which also override a conventional sort. For example.…

```
1945-05-08T08:00 A Sqn…
1945-05-08T11:00 A Sqn…
1945-05-08T08:00 B Sqn…
1945-05-08T10:00 B Sqn…
1945-05-08T09:45 D Sqn…
```

Perhaps this needs a "sequence-nb-within-date-that-overrides-time" (sacrificing seconds, allowing up to 99 items within a group). The [+] symbol is what indicates this special-case of time-sequence :–

```
1944-05-01T01+08:00
1944-05-01T02+11:00
1944-05-01T03+08:00
1944-05-01T04+18:00
1944-05-01T05+09:45
```

How can meaningful date range searching and date mathematics be undertaken with fuzzy dates? Again, these are implementation issues best handled elsewhere, perhaps using Regular Expressions, but it may be useful to note that the author devised some (tacky) methods that dynamically substituted 'start-of-month' or 'end-of-month' values (for example: `YYYY-MM-01` or `YYYY-MM-31`) into fuzzy dates at runtime to get a handle on these kinds of problems. Clearly this needs looking at to devise industrial-strength solutions.

4.3 Miscellaneous

The author's investigations unearthed a 20-year-old document about historical research (see the Bibliography) that contained an interesting variation on the representation of intervals. In trying to convey 'sometime between this date and that', one can just concatenate two qualified FZY-D Dates to make a qualified FZY-D Interval, but this document described a triple-date format which conveyed 'this particular date, give or take a specific amount of time', and comprised a 'Probable Date', followed by a 'Minimum Date' and a 'Maximum Date'.

Not wishing to introduce an extension to the FZY-D format, consideration was given to a possible way of achieving this effect within the proposed format and general spirit of ISO 8601, like this :–

```
1985-04-12T##:##:##Z-AD#-?#-##±####-##-14T##:##:###-###-~#-##
```

…meaning '*probably 12 April 1985, plus or minus approximately two weeks*', by using the [±] character (decimal ASCII 177) as an alternative but quite intuitive separator. Obviously this doesn't allow an asymmetric span of the central date, but otherwise seems to adequately manage the problem (asymmetry could be handled by multiple date records, for instance). Although the alternative separator does break one of the ground rules by being above decimal ASCII 127, this exception is probably justifiable. Note that this should not be confused with the matching ISO 8601 format placeholder character for either of the plus *or* minus characters: this is the actual character [±].

Consideration was also given to the concept of 'alternative date' so, extending thinking from the [±] exception, we could usefully have another :–

```
1985-04-12T##:##:##Z-AD#-##-##|1985-04-13T##:##:##Z-AD#-##-##
```

…meaning '*either the 12th or the 13th April 1985*', with the pipe [|] character (ASCII 124) again quite intuitive.

The related concept of 'alias date' falls here too: we could use the combination of pipe separator with uncertainty qualifiers to convey subtleties such as '*probably this date but maybe that one*', or '*if you see this date you should also look at that one*'.

To clarify: if these variations were implemented then there would be three legal interval separators, where the solidus [/] and the pipe [|] both indicate that the second part of the format is also a date, while the plus/minus character [±] indicates that the second part of the format contains a positive integer value in one or more of the date and time component positions. For input validation purposes the solidus and pipe formats would indicate that the second date part must be greater than the first (equal would of course be meaningless) for FZY-D Era codes AD and CE, and the reverse for FZY-D Era codes BC and BCE.

The concept of a 'disputed' date, originally assigned [!!], was considered but dropped as it would have to be complementary, rather than alternative, to the confidence qualifiers, implying yet another extension to the format, and this was felt to be going too far, as well as also probably best handled in implementation. [!!] was then briefly recycled as meaning 'Definitely', with [!#] as 'Almost certainly', but these were changed because their ASCII values meant that they would sort before [##], meaning 'Unqualified', which really had to come first.

4.4 Allen Operators

Feedback from the release of v1.1 of this document to a member of "Antiquist"[10] drew the author's attention to the work of James F. Allen. The Antiquist association is dedicated to work on "computing in the heritage sector", and Allen has worked for many years on the study of "temporal relations"[11], a term that had not previously surfaced in the search for information on "fuzzy dates".

It now appears that the present author has, in part, merely reinvented a fuzzy wheel. Put simply, Allen devised terms about 28 years ago, now widely known as "Allen Operators", to describe relationships between dates. Some of these Allen Operators, such as "Before" and "After", are unsurprisingly identical to those in **3.8** above, whilst others such as "Meets", "During" and "Overlaps" cover nuances that had escaped the author's conceptual ruminations.

The full set of Allen Operators is : –

BEFORE
DURING
AFTER
MEETS
OVERLAPS
MERGES

Allen considers all events to be intervals not dates. In his studies Allen explores what might be regarded as a sub-level of the proposed FZY-D format: the idea that both of the dates of an interval (as in **3.2** above) might themselves be considered as intervals, with their own upper and lower bounds (fuzzy or precise). The FZY-D Interval format, as proposed, handles only the outer boundaries of a given timespan, but not the possible inner boundaries.

However, it would appear that Allen's thinking could easily be accommodated by further extending the FZY-D format to allow the concatenation of two FZY-D Intervals, perhaps separated by a two-character "Allen Operator Designator" to give a 124-character string (which could be designated a "**FZY-A** Datetime"). It seems that this would have little or no effect on the definition of a fuzzy date or fuzzy interval as in **1.5** above.

5. Conclusion

All this gives us a comprehensive representation of a datetime in a way that is not possible in ISO 8601 alone: we have date, time, zone, era, precision, orientation and confidence, all neatly bundled in 30, 61 (or 124) characters, catering for a very wide range of circumstances in a rich vocabulary and amenable to automated searching, sorting and filtering.

This all started, as these things do, as a small and casual solution to a minor irritation. Pondering the ramifications led the author inexorably into a much bigger arena, and it is believed that there is something useful here for a wide range of historians and database designers. The paper is not presented as a definitive fait-accompli, but put forward as a sturdy framework for robust examination, collaborative refinement and then consensus adoption.

R.M.Paterson
January 2016

10 *www.antiquist.org*
11 *www.ics.uci.edu/~alspaugh/cls/shr/allen.html*

6. Bibliography and Links

J.F.Allen. Maintaining Knowledge about Temporal Intervals. *Comm. ACM, 26: pp. 832-843, 1983*

J.F.Allen. Towards a General Theory of Action and Time. *Artificial Intelligence, 23: pp.123-154, 1984*

Doerr and Yiortsou. Implementing a Temporal Datatype. *Technical Report ICS-FORTH/TR 236, November 1998*

Shaw, Troncy and Hardman. LODE: Linking Open Descriptions of Events. *UC Berkeley, 2009*

Signore and Barioli. Managing Art History Fuzzy Dates. *Historical Social Research Vol 14, 1989, No 3. 98-104*

www.macgregortech.com	Author's software development company
www.popski.org	Author's Special Forces remembrance association
www.4d.com	Rapid Application Development Database company
www.iso.org/iso/iso8601	Home page for the ISO 8601 Standard
www.timeanddate.com/time/zones/military	NATO time zone codes
www.antiquist.org	Association for computing in the heritage sector
www.ics.uci.edu/~alspaugh/cls/shr/allen.html	Details about Allen Operators
cidoc-crm.org/	CIDOC Conceptual Reference Model