

HL7 Patient Demographics Information Models

HL7 includes two approaches to modeling patient demographics. HL7 version 2.x is a traditional message segment oriented approach while HL7 version 3.x is based on an object oriented model.

HL7 version 2.x Patient Identification (PID) segment

PID Segment

- PID-5 -- Patient Name -- XPN data type --required field -- repeating
- PID-7 -- Date/Time of Birth -- DTM data type -- optional - nonrepeating
- PID-8 -- Administrative Sex -- IS data type -- optional -- nonrepeating
- PID-16 -- Marital Status -- CWE data type -- optional -- nonrepeating

Notes:

- (1) *Patient Name consists of components for family name, given name, second and further given names, name suffix, name prefix, name type code (see HL7 Table 0200 below), name assembly order (for non-Western names), effective date, expiration date.*
- (2) *Date/time of Birth may be an imprecise date (not the same as accuracy)*
- (3) *Administrative Sex (see User-defined Table 0001 below) does not currently match the Sex Structure codes adopted by HITSP*
- (4) *Martital Status (see User-defined Table 0002 below) can be extended by users*

[See pp 2 - 5 for details]

HL7 version 3.x Patient model

Person Class

- name -- PN datatype -- required (may send nullFlavor) -- repeating
- birthTime -- TS datatype -- optional -- nonrepeating
- administrativeGenderCode -- CE CWE data type -- optional -- nonrepeating
- maritalStatusCode -- CE CWE data type -- optional -- nonrepeating

Notes:

- (1) *Person.name uses the PN (Person Name) data type. It consists of (a) labeled name parts (such as given name, family name, prefix, suffix, etc) with a qualifier (such as 'nickname', 'birth name', etc.), (b) a name use code (such as legal, pseudonym, license), (c) a valid time interval and (d) a character string value with the person name formatted with proper spacing.*
- (2) *Person.birthTime may be an imprecise date (not the same as accuracy)*
- (3) *Person.administrativeGenderCode is not constrained to any value set in the HL7 International standard so it can use the Sex Structure value set adopted by HITSP*
- (4) *Person.maritalStatusCode is not constrained to any value set in the HL7 International standard so can use the Marital Status value set once it is adopted by HITSP*

[See pp 6 -15 for details]

3.4.2 Patient Identification Segment

HL7 Attribute Table - PID - Patient Identification

SEQ	LEN	DT	OPT	RP/#	TBL#	ITEM#	ELEMENT NAME
1	4	SI	O			00104	Set ID - PID
2	20	CX	B			00405	Patient ID
3	250	CX	R	Y		00106	Patient Identifier List
4	20	CX	B	Y		00407	Alternate Patient ID - PID
5	250	XPN	R	Y	0200	00108	Patient Name
6	250	XPN	O	Y		00109	Mother's Maiden Name
7	24	DTM	O			00110	Date/Time of Birth
8	1	IS	O		0001	00111	Administrative Sex
9	250	XPN	B	Y		00412	Patient Alias
10	705	CWE	O	Y	0005	00113	Race
11	250	XAD	O	Y		00114	Patient Address
12	4	IS	B		0289	00415	County Code
13	250	XTN	O	Y		00116	Phone Number - Home
14	250	XTN	O	Y		00117	Phone Number - Business
15	705	CWE	O		0296	00118	Primary Language
16	705	CWE	O		0002	00119	Marital Status
17	705	CWE	O		0006	00120	Religion
18	250	CX	O			00121	Patient Account Number
19	46	ST	B			00422	SSN Number - Patient
20	25	DLN	B			00423	Driver's License Number - Patient
21	250	CX	O	Y		00124	Mother's Identifier
22	705	CWE	O	Y	0189	00125	Ethnic Group
23	250	ST	O			00126	Birth Place
24	1	ID	O		0136	00127	Multiple Birth Indicator
25	2	NM	O			00128	Birth Order
26	705	CWE	O	Y	0171	00129	Citizenship
27	705	CWE	O		0172	00130	Veterans Military Status
28	705	CWE	B		0242	00739	Nationality
29	24	DTM	O			00740	Patient Death Date and Time
30	1	ID	O		0136	00741	Patient Death Indicator
31	1	ID	O		0136	01535	Identity Unknown Indicator
32	20	IS	O	Y	0445	01536	Identity Reliability Code
33	24	DTM	O			01537	Last Update Date/Time
34	241	HD	O			01538	Last Update Facility
35	705	CWE	C		0446	01539	Species Code
36	705	CWE	C		0447	01540	Breed Code
37	80	ST	O			01541	Strain
38	705	CWE	O	2	0429	01542	Production Class Code
39	705	CWE	O	Y	0171	01840	Tribal Citizenship

HL7 version 2.6 Chapter 3 - Patient Administration

User-defined Table 0001 - Administrative Sex

Value	Description	Comment
F	Female	
M	Male	
O	Other	
U	Unknown	
A	Ambiguous	
N	Not applicable	

User-defined Table 0002 - Marital Status

Value	Description	Comment
A	Separated	
D	Divorced	
M	Married	
S	Single	
W	Widowed	
C	Common law	
G	Living together	
P	Domestic partner	
R	Registered domestic partner	
E	Legally Separated	
N	Annulled	
I	Interlocutory	
B	Unmarried	
U	Unknown	
O	Other	
T	Unreported	

2.A.88 XPN - extended person name data type

HL7 Component Table - XPN– Extended Person Name

SEQ	LEN	DT	OPT	TBL#	COMPONENT NAME	COMMENTS
1	194	FN	O		Family Name	
2	30	ST	O		Given Name	
3	30	ST	O		Second and Further Given Names or Initials Thereof	
4	20	ST	O		Suffix (e.g., JR or III)	
5	20	ST	O		Prefix (e.g., DR)	
6	6	IS	B	0360	Degree (e.g., MD)	
7	1	ID	O	0200	Name Type Code	
8	1	ID	O	0465	Name Representation Code	
9	705	CWE	O	0448	Name Context	
10	40	DR	B		Name Validity Range	
11	1	ID	O	0444	Name Assembly Order	
12	24	DTM	O		Effective Date	
13	24	DTM	O		Expiration Date	
14	199	ST	O		Professional Suffix	

2.A.30 - FN - family name

HL7 Component Table - FN – Family Name

SEQ	LEN	DT	OPT	TBL#	COMPONENT NAME	COMMENTS
1	50	ST	R		Surname	
2	20	ST	O		Own Surname Prefix	
3	50	ST	O		Own Surname	
4	20	ST	O		Surname Prefix from Partner/Spouse	
5	50	ST	O		Surname from Partner/Spouse	

Definition: This data type allows full specification of the surname of a person. Where appropriate, it differentiates the person's own surname from that of the person's partner or spouse, in cases where the person's name may contain elements from either name. It also permits messages to distinguish the surname prefix (such as "van" or "de") from the surname root.

HL7 version 2.6 Chapter 2A - Datatypes

HL7 Table 0200 - Name Type

Value	Description	Comment
A	Alias Name	
B	Name at Birth	
C	Adopted Name	
D	Display Name	
I	Licensing Name	
K	Artist Name	
L	Legal Name	
M	Maiden Name	
N	Nickname / "Call me" Name/Street Name	
P	Name of Partner/Spouse (retained for backward compatibility only)	
R	Registered Name (animals only)	
S	Coded Pseudo-Name to ensure anonymity	
T	Indigenous/Tribal/Community Name	
U	Unspecified	

HL7 Table 0444 – Name assembly order

Value	Description	Comment
G	Prefix Given Middle Family Suffix	
F	Prefix Family Middle Given Suffix	

2.A.22 - DTM - date/time data type

HL7 Component Table - DTM – Date/Time

SEQ	LEN	DT	OPT	TBL#	COMPONENT NAME	COMMENTS
	24				Date/Time	

Definition: Specifies a point in time using a 24-hour clock notation.

Maximum Length: 24

The number of characters populated (excluding the time zone specification) specifies the precision.

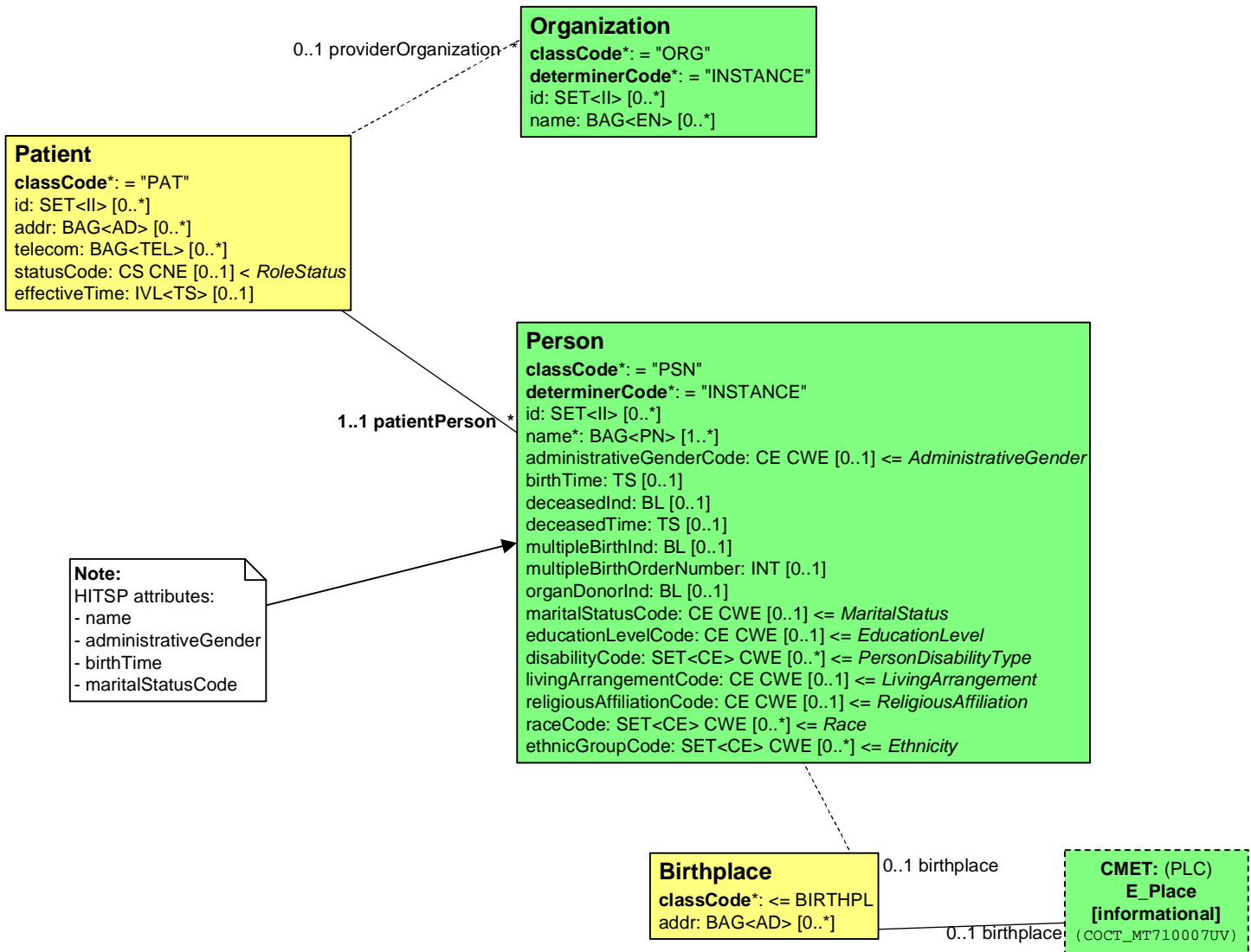
Format: YYYY[MM[DD[HH[MM[SS[.S[S[S[S]]]]]]]]][+/-ZZZZ].

Thus:

- only the first four are used to specify a precision of "year"
- the first six are used to specify a precision of "month"
- the first eight are used to specify a precision of "day"
- the first ten are used to specify a precision of "hour"
- the first twelve are used to specify a precision of "minute"
- the first fourteen are used to specify a precision of "second"
- the first sixteen are used to specify a precision of "one tenth of a second"
- the first nineteen are used to specify a precision of "one ten thousandths of a second"

Example: |199904| specifies April 1999.

HL7 v3 Patient Model



2.22 Entity Name Part (ENXP) specializes [ST](#)

Definition: A character string token representing a part of a name. May have a type code signifying the role of the part in the whole entity name, and a qualifier code for more detail about the name part type. Typical name parts for person names are given names, and family names, titles, etc.

Table 25: Property Summary of Entity Name Part

Name	Type	Description
partType	CS	Indicates whether the name part is a given name, family name, prefix, suffix, etc.
qualifier	SET<CS>	The qualifier is a set of codes each of which specifies a certain subcategory of the name part in addition to the main name part type. For example, a given name may be flagged as a nickname, a family name may be a pseudonym or a name of public records.

```
protected type EntityNamePart alias ENXP specializes ST {
    CS      type;
    SET<CS> qualifier;
};
```

2.22.1 Name Part Type : [CS](#)

Definition: Indicates whether the name part is a given name, family name, prefix, suffix, etc.

Table 26: Domain EntityNamePartType

code	name	definition
FAM	family	Family name, this is the name that links to the genealogy. In some cultures (e.g. Eritrea) the family name of a son is the first name of his father.
GIV	given	Given name (don't call it "first name" since this given names do not always come first)
PFX	prefix	A prefix has a strong association to the immediately following name part. A prefix has no implicit trailing white space (it has implicit leading white space though). Note that prefixes can be inverted.
SFX	suffix	A suffix has a strong association to the immediately preceding name part. A prefix has no implicit leading white space (it has implicit trailing white space though). Suffixes can not be inverted.
DEL	delimiter	A delimiter has no meaning other than being literally printed in this name representation. A delimiter has no implicit leading and trailing white space.

HL7 version 3 - Datatypes

Not every name part must have a type code, if the type code is unknown, not applicable, or simply undefined this is expressed by a NULL value (type.isNull). For example, a name may be "Rogan Sulma" and it may not be clear which one is a given name or which is a last name, or whether Rogan may be a title.

Entity names are conceptualized as text with added mark-up. The mark-up may describe in detail the role of each name part if it is known. Name parts occur in the order in which they would be printed on a mailing label. The model is similar to HTML or XML markup of text.

2.22.2 Qualifier : [SET](#)<[CS](#)>

Definition: The qualifier is a set of codes each of which specifies a certain subcategory of the name part in addition to the main name part type. For example, a given name may be flagged as a nickname, a family name may be a pseudonym or a name of public records.

Table 27: Domain EntityNamePartQualifier		
code	name	definition
AC	academic	Indicates that a prefix like "Dr." or a suffix like "M.D." or "Ph.D." is an academic title.
NB	nobility	In Europe and Asia, there are still people with nobility titles (aristocrats). German "von" is generally a nobility title, not a mere voorvoegsel. Others are "Earl of" or "His Majesty King of..." etc. Rarely used nowadays, but some systems do keep track of this.
PR	professional	Primarily in the British Imperial culture people tend to have an abbreviation of their professional organization as part of their credential suffices.
VV	voorvoegsel	A Dutch "voorvoegsel" is something like "van" or "de" that might have indicated nobility in the past but no longer so. Similar prefixes exist in other languages such as Spanish, French or Portugese.
AD	adopted	The name the person was given at the time of adoption.
BR	birth	A name that a person had shortly after being born. Usually for family names but may be used to mark given names at birth that may have changed later.
SP	spouse	The name assumed from the partner in a marital relationship (hence the "M"). Usually the spouse's family name. Note that no inference about gender can be made from the existence of spouse names.

Table 27: Domain EntityNamePartQualifier		
code	name	definition
CL	callme	A callme name is (usually a given name) that is preferred when a person is directly addressed.
IN	initial	Indicates that a name part is just an initial. Initials do not imply a trailing period since this would not work with non-Latin scripts. Initials may consist of more than one letter, e.g., "Ph." could stand for "Philippe" or "Th." for "Thomas".
TITLE	title	Indicates that a prefix or a suffix is a title that applies to the whole name, not just the adjacent name part.

2.23 Entity Name (EN) specializes [LIST](#)<ENXP>

Definition: A name for a person, organization, place or thing. A sequence of name parts, such as given name or family name, prefix, suffix, etc. Examples for entity name values are "Jim Bob Walton, Jr.", "Health Level Seven, Inc.", "Lake Tahoe", etc. An entity name may be as simple as a character string or may consist of several entity name parts, such as, "Jim", "Bob", "Walton", and "Jr.", "Health Level Seven" and "Inc.", "Lake" and "Tahoe".

Table 28: Property Summary of Entity Name		
Name	Type	Description
use	SET<CS>	A set of codes advising a system or user which name in a set of names to select for a given purpose.
validTime	IVL<TS>	An interval of time specifying the time during which the name is or was used for the entity. This accomodates the fact that people change names for people, places and things.
formatted	ST	<p>A character string value with the entity name formatted with proper spacing. This is only a semantic property to define the function of some of the name part types.</p> <p>Remember that semantic properties are bare of all control flow semantics. The formatted could be implemented as a "procedure" that would "return" the formatted name, but it would not usually be a variable to which one could assign a formatted name. However, HL7 does not define applications but only the semantics of exchanged data values. Hence, the semantic model abstracts from concepts like "procedure", "return", and "assignment" but speaks only of property and value.</p>

HL7 version 3 - Datatypes

Entity names are conceptualized as text with added logical mark-up. Name parts occur in a natural order in which they would be displayed, as opposed to in a order determined by name part. The ordering of the name parts is significant a feature that replaces the need for a separate "display name" property. Applications may change that ordering of name parts to account for their user's customary ordering of name parts. The approach is similar to HTML or XML markup of text (but it is not technically limited to XML representations.)

Entity names are essentially sequences of entity name parts, but add a "use" code and a valid time range for information about when the name was used and how to choose between multiple aliases that may be valid at the same point in time.

```
type EntityName alias EN specializes LIST<ENXP> {  
  SET<CS>   use;  
  IVL<TS>   validTime;  
  BL        equal(ANY x);  
  ST        formatted;  
};
```

2.23.1 Use Code : [SET<CS>](#)

Definition: A set of codes advising a system or user which name in a set of names to select for a given purpose.

Table 29: Domain EntityNameUse		
code	name	definition
C	License	As recorded on a license, record, certificate, etc. (only if different from legal name)
I	Indigenous/Tribal	e.g. Chief Red Cloud
L	Legal	Known as/conventional/the one you use
P	pseudonym	A self asserted name that the person is using or has used.
A	Artist/Stage	Includes writer's pseudonym, stage name, etc
R	Religious	e.g. Sister Mary Francis, Brother John
SRCH	search	A name intended for use in searching or matching.
PHON	phonetic	A name spelled phonetically.
SNDX	Soundex	A name spelled according to the SoundEx algorithm.
ABC	Alphabetic	Alphabetic transcription of name (Japanese: romaji)
SYL	Syllabic	Syllabic transcription of name (e.g., Japanese kana, Korean hangul)

Table 29: Domain EntityNameUse		
code	name	definition
IDE	Ideographic	Ideographic representation of name (e.g., Japanese kanji, Chinese characters)

A name without specific use code might be a default name useful for any purpose, but a name with a specific use code would be preferred for that respective purpose.

2.23.2 Valid Time : [IVL](#)<[TS](#)>

Definition: An interval of time specifying the time during which the name is or was used for the entity. This accommodates the fact that people change names for people, places and things.

EN conforms to the history item data type extension ([HXIT](#)).

2.23.3 Equality : [BL](#), inherited from [ANY](#)

Two name values are considered equal if both contain the same name parts, independent of ordering. Use code and valid time are excluded from the equality test.

```
invariant(EN x, y)
  where x.nonNull.and(y.nonNull) {
    x.equal(y).equal((
      forall(ENXP p) where x.contains(p) {
        y.contains(p);
      }).and.(
        forall(ENXP p) where x.contains(p) {
          y.contains(p);
        }));
```

2.23.4 Formatting Entity Names : [ST](#)

Definition: A character string value with the entity name formatted with proper spacing. This is only a semantic property to define the function of some of the name part types.⁴¹

The *EN* data type's main purpose is to capture names of people, places, and things (entities), so that one can address and refer to these entities in speech and writing. Humans will look at names in printed form, such as on a mailing label. The *EN* data type therefore defines precise rules of how its data is formatted.⁴²

Entity names are ordered lists of entity name parts. Each entity name part is printed in the order of the list from left to right (or in any other language-specific reading direction.) Every entity name part (except for those marked "invisible") is printed. Most entity name parts are framed by whitespace. The following six rules govern the setting of whitespace.

1. Whitespace never accumulates, i.e. two subsequent spaces are the same as one.
2. Literals may contain explicit whitespace subject to the same white space reduction rules.
3. Except for *prefix*, *suffix* and *delimiter* name parts, every name part is surrounded by implicit whitespace. Leading and trailing explicit whitespace is insignificant in all those name parts.
4. Delimiter name parts are not surrounded by any implicit white space. Leading and trailing explicit whitespace is significant in delimiter name parts.
5. Prefix name parts only have implicit leading whitespace but no implicit trailing whitespace. Trailing explicit whitespace is significant in prefix name parts.
6. Suffix name parts only have implicit trailing whitespace but no implicit leading whitespace. Leading explicit whitespace is significant in suffix name parts.
7. This means that all entity name parts are generally surrounded by whitespace, but whitespace does never accumulate. Delimiters are never surrounded by implicit whitespace, prefixes are not followed by implicit whitespace and suffixes are not preceded by implicit white space. Every whitespace contributed by preceding or succeeding name parts around those special name parts is discarded, whether it was implicit or explicit.

2.23.5 Examples

A very simple encoding of "Adam A. Everyman" would be:

Example 7.

```
<name>
  <given>Adam</given>
  <given>A.</given>
  <family>Everyman</family>
</name>
```

None of the special qualifiers need to be mentioned if they are unknown or irrelevant. The next example shows extensive use of multiple given names, prefixes, suffixes, for academic degrees, nobility titles, *vorvoegsels* ("van"), and professional designations.

Example 8.

```
<name>
  <prefix qualifier="AC">Dr. phil. </prefix>
  <given>Regina</given>
  <given>Johanna</given>
  <given>Maria</given>
  <prefix qualifier="NB">Gr&auml;fin </prefix>
  <family qualifier="BR">Hochheim</family>-<family
qualifier="SP">Weilenfels</family>
  <suffix qualifier="PR">NCFSA</suffix>
</name>
```

The next example is an organization name, "Health Level Seven, Inc." in simple string form:

Example 9.

```
<name>Health Level Seven, Inc.</name>
```

and as a fully parsed name

Example 10.

```
<name>Health Level Seven, <suffix  
qualifier="LS">Inc.</suffix></name>
```

The following example shows a Japanese name in the three forms: ideographic (Kanji), syllabic (Hiragana), and alphabetic (Romaji).

Example 11.

```
<name use="IDE">  
  <family>木村</family>  
  <given>通男</given>  
</name>  
<name use="SYL">  
  <family>きむら</family>  
  <given>みちお</given>  
</name>  
<name use="ABC">  
  <family>KIMURA</family>  
  <given>MICHIO</given>  
</name>
```

2.25 Person Name (PN) specializes [EN](#)

Definition: An [EN](#) used when the named Entity is a Person. A sequence of name parts, such as given name or family name, prefix, suffix, etc. A name part is a restriction of entity name part that only allows those entity name parts qualifiers applicable to person names. Since the structure of entity name is mostly determined by the requirements of person name, the restriction is very minor.

Since most of the functionality of entity name is in support of person names, the person name (PN) is only a very minor restriction on the entity name part qualifier.

```
type PersonName alias PN specializes EN;  
  
invariant(PN this) {  
  forall(ENXP part)  
    where this.contains(part) {  
      part.qualifier.contains("LS").not;  
    }  
};
```

2.36 Point in Time (TS) specializes [QTY](#)

Definition: A quantity specifying a point on the axis of natural time. A point in time is most often represented as a calendar expression.

2.36.3 Calendar : [CS](#)

Definition: A code specifying the calendar used in the literal representation of this *TS*.⁴⁸

Table 37: Domain Calendar		
code	name	definition
GREG	Gregorian	The Gregorian calendar is the calendar in effect in most countries of Christian influence since approximately 1582. This calendar superceded the Julian calendar.

The purpose of this property is mainly to faithfully convey what has been entered or seen by a user in a system originating such a *TS* value. *calendar* also advises any system rendering a *TS* value into a literal form of which calendar to use. However, this is only advice; any system that renders *TS* values to users may choose to use the calendar and literal form demanded by its users rather than the calendar mentioned in *calendar*. Hence, *calendar* is not constant in communication between systems, the calendar is not part of the equality test.

For the purpose of defining the relationship between calendar expression and epoch/offset form, two private data types, [CAL](#) and [CLCY](#) are defined. These calendar data types exist only for defining this specification. These private data types may not be used at all outside this specification.

2.36.4 Precision of the Calendar Literal Form : [INT](#)

Definition: The number of significant digits of the calendar expression representation.

precision is formally defined based on the [literal](#).

precision is only the precision of a decimal digit representation, *not the accuracy of the TS value*.

The purpose of *precision* is to faithfully capture the whole information presented to humans in a calendar expression. The number of digits shown conveys information about the uncertainty (i.e., precision and accuracy) of a measured *TS*.

NOTE: The precision of the representation is independent from uncertainty (precision accuracy) of a measurement result. If the uncertainty of a measurement result is important, one should specify uncertain values as [PPD](#)

precision is dependent on [calendar](#). A given *precision* relative to one calendar does not mean the same in another calendar with different periods.

For example "20000403" has 8 significant digits *in the representation*, but the *uncertainty of the value* may be in any digit shown or not shown, i.e., the uncertainty may be to the day, to the week, or to the hour. Note that external representations *should* adjust their representational precision with the uncertainty of

HL7 version 3 - Datatypes

the value. However, since the precision in the digit string depends on the calendar and is granular to the calendar periods, uncertainty may not fall into that grid (e.g., 2000040317 is an adequate representation for the value between 2000040305 and 2000040405.)

NOTE: A character based ITS need not represent *precision* as an explicit attribute if *TS* values are represented as literal calendar expressions. A *TS* representation must not produce more or less significant digits than were originally in that value. Conformance can be tested through round-trip encoding - decoding - encoding.