Description of Quest II Data File Format

Constantijn Sikkel

Version 1.35 of June 3, 2016

Abstract

This document describes the `qdf` file format used in the exchange of data with the University of Greifswald, Germany for the Quest II project. The same file format is used for the supply of data to Logos Bible Software for the Stuttgart Electronic Study Bible, which has superseded Quest II. The production of the data is taken care of by the Werkgroep Informatica, Vrije Universiteit Amsterdam. The underlying database model is not explained; a discussion at length can be found in Chapter 3 and 4 of [Doedens94].

1 Characteristics of the file

Every file covers a single book of the Hebrew Bible. The file name consists of the name of the book followed by the extension `qdf`, for instance `exodus.qdf`. The files are complete, that is, all the linguistic data made available are contained within a single file. Each word in the book is described on a single line in the file. The order of the lines in the file reflects the word order in the book. Every line has a fixed length of 372 characters and is terminated by an additional newline. The size of the file is therefore equal to the number of words in the book times 373.

2 Format of the lines

Every line holds 61 fields, each of which has a fixed width. Three types of fields are in use: single character, integer and string fields. String fields are padded to the right with spaces, and integer fields are padded to the left with spaces. The only white space characters used in the file are *space* (ASCII character 32) and *newline* (ASCII character 10). All fields are separated from their successor by a single space.

Apart from the values described below, there is a special value that applies to every type of field. A single dot (ASCII character 46) in a field indicates that the feature in question is absent or does not apply to the current word. (Phrase features, for instance, are only listed with the last word of the phrase head.) The value for *not applicable* will be assigned to the feature if an Emdros object requires it.

In some of the following sections the description of a value is followed by a number in parentheses (see, for instance, Section 2.48). These codes are well known numbers used by the production programs, and not part of the data presented here. Entries with a dagger† are there *pro memoria*. They do not actually occur in the data.

For every Emdros feature also the identifier, values, and object type are given as used in the schema of the database. With morphemes, the values for *not applicable* and *absent* are the strings "`n/a`" and "`absent`" in this schema; with grammatical functions, the values for *not applicable* (-1) and *unknown* (0) are the identifiers `NA` and `unknown`.

2.1 Verse label

Type: string. Width: 10 (1–10).
Object type: `verse`. Feature identifier: `label`. Feature value type: string of fixed length 10.

2.2 Half-verse label

Type: character. Width: 1 (12–12).
Object type: `half_verse`. Feature identifier: `label`. Feature value type: string of fixed length 1. The values factually present in the data range from 'A' to 'C'.

2.3 Graphical representation of the word

Type: string. Width: 35 (14–48).
Object type: `word`. Feature identifier: `g_word`. Feature value type: string of variable length. Longest graphical word is אָבְלָה בֵּית הַמַּעֲכָה in 2S 20:15, with a length of 35.

2.4 Paradigmatic form of the preformative

Type: integer. Width: 2 (50–51).
Object type: `word`. Feature identifier: `pfm`. Feature value type: string of variable length. The two markers `!!` in the list are not part of the feature value.

−1	Not applicable	3	`!T!`	7	`!M!`
0	Absent	4	`!>!`	8	`!T=!`
1	`!!`	5	`!N!`	9	`!L!`
2	`!J!`	6	`!H!`		

2.5 Graphical representation of the preformative

Type: string. Width: 7 (53–59).
Object type: `word`. Feature identifier: `g_pfm`. Feature value type: string of variable length. The two markers `!!` in the field are not part of the feature value.

2.6 Paradigmatic form of the root formation morpheme

Type: integer. Width: 2 (61–62).
Object type: `word`. Feature identifier: `vbs`. Feature value type: string of variable length. The two markers `]]` in the list are not part of the feature value. (The identifier `vbs` is a mnemonic for verbal stem.)

−1	Not applicable	5	*Not used*	11	*Not used*
0	Absent	6	`]HT]`	12	`]NT]`
1	*Not used*	7	*Not used*	13	`]>T]`
2	`]H]`	8	*Not used*	14	`]T]`
3	`]N]`	9	`]>CT]`†	15	`]>]`
4	*Not used*	10	`]HCT]`	16	`]C]`

2.7 Graphical representation of the root formation morpheme

Type: string. Width: 10 (64–73).
Object type: `word`. Feature identifier: `g_vbs`. Feature value type: string of variable length. The two markers `]]` in the field are not part of the feature value.

2.8 Lexical set

Type: integer. Width: 2 (75–76).
Object type: `word`. Feature identifier: `ls`. Feature value type: enumeration. In order to arrive at the value for lexical set, the number in this field needs to be combined with *part of speech* (2.28), as shown in the table below.

−6	2	`nmdi`	Distributive noun
−5	2	`nmcp`	Copulative noun
−4	2	`padv`	Potential adverb
−4	4	`afad`	Anaphoric adverb
−3	2	`ppre`	Potential preposition
−3	4	`cjad`	Conjunctive adverb
−3	13	`ordn`	Ordinal
−2	1	`vbcp`	Copulative verb
−2	2	`mult`	Noun of multitude
−2	4	`focp`	Focus particle
−2	12	`ques`	Interrogative particle
−2	13	`gntl`	Gentilic
−1	1	`quot`	Quotation verb
−1	2	`card`	Cardinal

Otherwise the value for lexical set is `none`. The values Topographical Name, Person's Name and People's Name, which were listed in earlier versions of this document, have been removed from the feature lexical set, because they were once added as semantic embellishments, but not actually used in the parsing and as such do not belong to lexical set.

2.9 Lexeme

Type: string. Width: 15 (78–92).
Object type: `word`. Feature identifier: `lex`. Feature value type: string of variable length.

2.10 Graphical representation of the lexeme

Type: string. Width: 35 (94–128).
Object type: `word`. Feature identifier: `g_lex`. Feature value type: string of variable length.

2.11 Paradigmatic form of the verbal ending

Type: integer. Width: 2 (130–131).
Object type: `word`. Feature identifier: `vbe`. Feature value type: string of variable length. The marker `[` in the list is not part of the feature value.

−1	Not applicable	7	`[W`	15	`[NH`
0	Absent†	8	`[TM`	16	*Not used*
1	`[`	9	`[TN`	17	*Not used*
2	`[H`	10	`[NW`	18	`[H=`
3	`[T`	11	*Not used*	19	`[N`
4	*Not used*[1]	12	`[J`	20	`[N>`
5	`[T=`	13	`[JN`	21	`[T==`
6	`[TJ`	14	`[WN`	22	`[TWN`

2.12 Graphical representation of the verbal ending

Type: string. Width: 8 (133–140).
Object type: `word`. Feature identifier: `g_vbe`. Feature value type: string of variable length. The marker `[` in the field is not part of the feature value.

2.13 Paradigmatic form of the nominal ending

Type: integer. Width: 2 (142–143).
Object type: `word`. Feature identifier: `nme`. Feature value type: string of variable length. The marker `/` in the list is not part of the feature value.

−1	Not applicable	7	`/~H`†	15	`/JN`
0	Absent	8	*Not used*	16	`/TJ`
1	`/`	9	`/T~H`†	17	`/TJM`
2	`/H`	10	`/JM~H`†	18	`/W`
3	`/T`	11	`/W=`	19	`/JN=`
4	`/JM`	12	`/WTJ`	20	`/N`
5	`/J`	13	`/J=`	21	`/T=`
6	`/WT`	14	`/JM=`	22	`/TJN`

The morphemes `W=`, `J=`, `JM=`, and `JN=` are the homographs of the dual. The ending `T=` is the one of the Aramaic feminine plural in construct state (with a typical realisation of `-@T`).

2.14 Graphical representation of the nominal ending

Type: string. Width: 8 (145–152).
Object type: `word`. Feature identifier: `g_nme`. Feature value type: string of variable length. The marker `/` in the field is not part of the feature value.

2.15 Paradigmatic form of the univalent final

Type: integer. Width: 2 (154–155).[2]
Object type: `word`. Feature identifier: `uvf`. Feature value type: string of variable length. The marker `~` in the list is not part of the feature value. This 'morpheme type' holds a number of single-valued morphemes which occur in final position (not counting the pronominal suffix).

0	Absent		4	`~W`	Waw compaginis
1	*Not used*		5	`~J`	Yod compaginis
2	`~>`	Emphatic marker	6	`~N`	Energic `-EN`
3	`~H`	Locative			

2.16 Graphical representation of the univalent final

Type: string. Width: 5 (157–161).
Object type: `word`. Feature identifier: `g_uvf`. Feature value type: string of variable length. The marker `~` in the field is not part of the feature value.

[1] In earlier versions of the data this value belonged to `[TH`, but that ending represented only a realisation variant of `[T`, not a separate morpheme.

[2] The two pairs of fields for univalent final (formerly locative) and pronominal suffix (2.15–2.18) have switched positions with respect to earlier versions of the data.

singular							plural						
	ps	gn	S	A	Q	*f*		ps	gn	S	A	Q	*f*
J	1	-	2	6	3	70	N>	1	-	9	13	25	10
NJ	1	-	3	14	2	18							
K	2	m	4	3	4	99	KM	2	m	-	10	10	5
							KWN	2	m	10	4	23	10
KJ	2	f	-	5	24	0	KN	2	f	-	11	11	0
H	3	m	8	8	19	159	HM	3	m	-	15	12	12
HJ	3	m	7	17	22	77	HWN	3	m	13	12	21	51
H=	3	f	6	7	20	47	HN	3	f	15	18	15	8

Table 1: Pronominal suffixes in Aramaic

2.17 Paradigmatic form of the pronominal suffix

Type: integer. Width: 2 (163–164).
Object type: `word`. Feature identifier: `prs`. Feature value type: string of variable length. The marker + in the list is not part of the feature value.

−1	Not applicable	8	+H	17	*Not used*
0	Absent	9	+NW	18	*Not used*
1	*Not used*	10	+KM	19	*Not used*
2	+NJ	11	+KN	20	+H=
3	+J	12	+HM	21	+HWN
4	+K	13	+M	22	+HJ
5	+K=	14	+MW	23	+KWN
6	+W	15	+HN	24	+KJ†
7	+HW	16	+N	25	+N>

The morpheme H is that of the third person feminine singular (typical realisation -@H. in Hebrew and -AH. in Aramaic), whereas H= is the one of the Aramaic third person masculine singular (typical realisation -;H.).[3]

Historical note: For Aramaic these values differ from the traditional values of `syn01(1)` and the values calculated by the first version of `at2ps(1)`. A survey of the historical values for Aramaic and Hebrew is given in Tables 1 and 2. The value calculated by `syn01(1)`, based on `morfset(5)`, is in column S. Column A has the value from `at2ps(1)` and column Q the value in this field as they were with earlier versions of the data (19 and 20 have changed, as pointed out by Footnote 3).

Person of the pronominal suffix

Object type: `word`. Feature identifier: `prs_ps`. Feature value type: enumeration. This feature assumes the same values as *person* (2.21).

Number of the pronominal suffix

Object type: `word`. Feature identifier: `prs_nu`. Feature value type: enumeration. This feature assumes the same values as *number* (2.22).

Gender of the pronominal suffix

Object type: `word`. Feature identifier: `prs_gn`. Feature value type: enumeration. This feature assumes the same values as *gender* (2.23).

[3] For Aramaic this is different from the assignment in earlier versions of the data.

singular							plural						
	ps	gn	S	A	Q	*f*		ps	gn	S	A	Q	*f*
J	1	-	3	3	3	6577	NW	1	-	9	9	9	1635
NJ	1	-	2	2	2	1214							
K	2	m	4	4	4	7035	KM	2	m	10	10	10	2652
K=	2	f	5	5	5	1308	KN	2	f	11	11	11	19
W	3	m	6	6	6	12435	M	3	m	13	13	13	3938
HW	3	m	7	7	7	1081	HM	3	m	12	12	12	3034
							MW	3	m	14	14	14	117
H	3	f	8	8	8	3304	HN	3	f	15	15	15	183
							N	3	f	16	16	16	46

Table 2: Pronominal suffixes in Hebrew

2.18 Graphical representation of the pronominal suffix

Type: string. Width: 8 (166–173).
Object type: `word`. Feature identifier: `g_prs`. Feature value type: string of variable length. The marker + in the field is not part of the feature value.

2.19 Verbal stem

Type: integer. Width: 2 (175–176).
Object type: `word`. Feature identifier: `vs`. Feature value type: enumeration.

−1	Not applicable		13	`etpa`	Etpa“al
0	`qal`	Qal	14	`tif`	Tif‘al
1	`piel`	Pi“el	15	`afel`	Af‘el
2	`hif`	Hif‘il	16	`shaf`	Shaf‘el
3	`nif`	Nif‘al	17	`peal`	Pe‘al
4	`pual`	Pu“al	18	`pael`	Pa“el
5	`haf`	Haf‘el	19	`peil`	Pe‘il
6	`hit`	Hitpa“el	20	`htpa`	Hitpa“al
7	`htpe`	Hitpe‘el	21	`etpe`	Etpe‘el
8	`hof`	Hof‘al	22	`esht`	Eshtaf‘al†
9	`pasq`	Passive Qal	23	`etta`	Ettaf‘al†
10	`hsht`	Hishtaf‘al	24	`poel`	Po‘el
11	`hotp`	Hotpa“al	25	`poal`	Po‘al
12	`nit`	Nitpa“el	26	`htpo`	Hitpo‘el

2.20 Verbal tense

Type: integer. Width: 2 (178–179).
Object type: `word`. Feature identifier: `vt`. Feature value type: enumeration.

−1	Not applicable		5	`infa`	Infinitive absolute
0	Unknown†		6	`ptca`	Participle
1	`impf`	Imperfect	11	`wayq`	Wayyiqtol
2	`perf`	Perfect	12	`weyq`	Weyiqtol†
3	`impv`	Imperative	62	`ptcp`	Passive participle
4	`infc`	Infinitive construct			

These values have been taken from the manual page `ps1(5)`, which is authoritative. We have cancelled the value *weyiqtol*, because we are convinced it is only a

clause type, not a verbal tense.

2.21 Person

Type: integer. Width: 2 (181–182).
Object type: `word`. Feature identifier: `ps`. Feature value type: enumeration.

−1	Not applicable	1	`p1`	First	3	`p3`	Third
0	Unknown	2	`p2`	Second			

These values have been taken from the manual page `ps1(5)`, which is authoritative. The Emdros values are {`p1, p2, p3`} rather than {`first, second, third`} because `first` is a reserved word in MQL.

2.22 Number

Type: integer. Width: 2 (184–185).
Object type: `word`. Feature identifier: `nu`. Feature value type: enumeration.

−1	Not applicable	1	`sg`	Singular	3	`pl`	Plural
0	Unknown	2	`du`	Dual			

These values have been taken from the manual page `ps1(5)`, which is authoritative.

2.23 Gender

Type: integer. Width: 2 (187–188).
Object type: `word`. Feature identifier: `gn`. Feature value type: enumeration.

−1	Not applicable	1	`f`	Feminine
0	Unknown	2	`m`	Masculine

These values have been taken from the manual page `ps1(5)`, which is authoritative.

2.24 State

Type: integer. Width: 2 (190–191).
Object type: `word`. Feature identifier: `st`. Feature value type: enumeration.

−1	Not applicable	1	`c`	Construct	3	`e`	Emphatic
0	Unknown	2	`a`	Absolute			

These values have been taken from the manual page `ps1(5)`, which is authoritative.

2.25 Consonants of the surface form

Type: string. Width: 14 (193–206).
Object type: `word`. Feature identifier: `g_cons`. Feature value type: string of variable length. The underscore, as in `TWBL_QJN`, represents an embedded space.

2.26 Old lexeme

Type: string. Width: 14 (208–221).
This field is obsolete. It used to house the lexeme in Quest I style when it was different from field *lexeme* (2.9).

2.27 Word number

Type: integer. Width: 5 (223–227).
Object type: `word`. Feature identifier: `number`. Feature value type: integer. Specifies the word number within the book. The count starts at 1.

2.28 Part of speech

Type: integer. Width: 2 (229–230).
Object type: `word`. Feature identifier: `sp`. Feature value type: enumeration.

0	`art`	Article	7	`prps`	Personal pronoun
1	`verb`	Verb	8	`prde`	Demonstrative pronoun
2	`subs`	Noun	9	`prin`	Interrogative pronoun
3	`nmpr`	Proper noun	10	`intj`	Interjection
4	`advb`	Adverb	11	`nega`	Negative
5	`prep`	Preposition	12	`inrg`	Interrogative
6	`conj`	Conjunction	13	`adjv`	Adjective

These values have been taken from the manual page `ps2(5)`, which is authoritative.

2.29 Phrase dependent part of speech

Type: integer. Width: 2 (232–233).
Object type: `word`. Feature identifier: `pdp`. Feature value type: enumeration. Values as with *part of speech* (2.28).

2.30 Phrase atom number

Type: integer. Width: 5 (235–239).
Object type: `phrase_atom`. Feature identifier: `number`. Feature value type: integer. Specifies the phrase atom number within the book. The count starts at 1.

2.31 Phrase atom type

Type: integer. Width: 3 (241–243).
Object type: `phrase_atom`. Feature identifier: `typ`. Feature value type: enumeration. Values as with *phrase type* (2.46). The Emdros feature identifier is `typ` because `type` is a reserved word in MQL.

2.32 Phrase atom determination

Type: string. Width: 2 (245–246).
Object type: `phrase_atom`. Feature identifier: `det`. Feature value type: enumeration.

iD	und	Undetermined (1)	D	det	Determined (2)

2.33 Distance to mother

Type: integer. Width: 3 (248–250).
Object type: phrase_atom. Feature identifier: dist. Feature value type: integer. The distance is defined as the difference between the phrase atom number of the mother and the daughter, or the word number of the mother and the last word of the daughter, depending on whether the mother is a phrase atom or a word. The object type of the mother is determined by the *unit* (2.34) in which the distance is measured. A distance in phrase atoms implies that the mother is a phrase atom, and a distance in words implies that the mother is a word.
Object type: phrase. Feature identifier: dist. Feature value type: integer. The distance is defined as the difference between the clause atom number of the first atom of the mother and that of the daughter, or between the phrase atom number of the first atom of the mother and the daughter, depending on whether the mother is a clause or a phrase. The object type of the mother is determined by the *unit* (2.34) in which the distance is measured. A distance in clause atoms implies that the mother is a clause, and a distance in phrase atoms implies that the mother is a phrase.

2.34 Unit used in counting distance to mother

Type: character. Width: 1 (252–252).
Object type: phrase_atom.

C	Clause atoms	P	Phrase atoms	W	Words

The unit is not a separate Emdros feature, but it does indicate the object type of the mother. A distance in phrase atoms implies that the mother is a phrase atom or a phrase (apparent from the relation), and a distance in words implies that the mother is a word. Gn 11:28 has an example of the latter, where the phrase atom אביו is an apposition to the word תרח.

2.35 Phrase (atom) relation

Type: string. Width: 4 (254–257).
Object type: phrase_atom. Feature identifier: rela. Feature value type: enumeration.

Appo	Apposition (500)	Sfxs	Suffix specification (535)
Link	Conjunction (567)	Spec	Specification (582)
Para	Parallel (566)		

Object type: phrase. Feature identifier: rela. Feature value type: enumeration.

PrAd	Predicative adjunct (525)	Resu	Resumption (572)

This field contains both phrase atom relations, which build complex phrases, and phrase relations, where one constituent refers to another. In case of the *phrase* relations, the value for rela has been derived from the value of *phrase function within clause* (2.48) of the daughter (PrAd yields PrAd) or the mother (Frnt yields Resu). The mother of a resumption can be a clause, namely when the constituent in question resumes a casus pendens clause.

2.36 First subphrase relation. Type

Type: string. Width: 3 (259–261).
Object type: `subphrase`. Feature identifier: `rela`. Feature value type: enumeration {`NA=-1, reg=2, mod=4, adj=5, par=6, dem=8, atr=13`}.

`ADJ`	Adjunct (5)	`adj`	Adjunct (−5)
`ATR`	Attribute (13)	`atr`	Attribute (−13)
`DEM`	Demonstrative (8)	`dem`	Demonstrative (−8)
`MOD`	Modifier (4)	`mod`	Modifier (−4)
`PAR`	Parallel (6)	`par`	Parallel (−6)
`REG`	Regens (2)	`rec`	Regens/rectum (−2)

In case of the regens/rectum relation, the mother is not a subphrase, but a word. The upper case values apply to the mother subphrase and the lower case values apply to the daughter subphrase. The Emdros feature applies to the daughter only; the mother has the value `NA`.

2.37 First subphrase relation. Head

Type: integer. Width: 3 (263–265).
Object type: `subphrase`. Specifies the distance in words to the beginning of this subphrase. Not an Emdros feature as such, but used to construct the object.

2.38 First subphrase relation. Mother

Type: integer. Width: 3 (267–269).
Object type: `subphrase`. Feature identifier: `dist`. Feature value type: integer. Specifies the distance in words to the mother of this subphrase. It is defined as the difference in word number between the last word of the mother and the last word of the daughter.

2.39 Second subphrase relation. Type

Type: string. Width: 3 (271–273).
Object type: `subphrase`. As with the first subphrase relation.

2.40 Second subphrase relation. Head

Type: integer. Width: 3 (275–277).
Object type: `subphrase`. As with the first subphrase relation.

2.41 Second subphrase relation. Mother

Type: integer. Width: 3 (279–281).
Object type: `subphrase`. As with the first subphrase relation.

2.42 Third subphrase relation. Type

Type: string. Width: 3 (283–285).
Object type: `subphrase`. As with the first subphrase relation.

2.43 Third subphrase relation. Head

Type: integer. Width: 3 (287–289).
Object type: `subphrase`. As with the first subphrase relation.

2.44 Third subphrase relation. Mother

Type: integer. Width: 3 (291–293).
Object type: `subphrase`. As with the first subphrase relation.

2.45 Phrase number within clause

Type: integer. Width: 2 (295–296).
Object type: `phrase`. Feature identifier: `number`. Feature value type: integer.

2.46 Phrase type

Type: integer. Width: 3 (298–300).
Object type: `phrase`. Feature identifier: `typ`. Feature value type: enumeration. The Emdros feature identifier is `typ` because `type` is a reserved word in MQL.

1	`VP`	Verbal phrase
2	`NP`	Nominal phrase
3	`PrNP`	Proper-noun phrase
4	`AdvP`	Adverbial phrase
5	`PP`	Prepositional phrase
6	`CP`	Conjunctive phrase
7	`PPrP`	Personal pronoun phrase
8	`DPrP`	Demonstrative pronoun phrase
9	`IPrP`	Interrogative pronoun phrase
10	`InjP`	Interjectional phrase
11	`NegP`	Negative phrase
12	`InrP`	Interrogative phrase
13	`AdjP`	Adjective phrase

These values have been taken from the manual page `ps3(5)`, which is authoritative.

2.47 Phrase determination

Type: string. Width: 2 (302–303).
Object type: `phrase`. Feature identifier: `det`. Feature value type: enumeration. Values as with *phrase atom determination* (2.32).

2.48 Phrase function within clause

Type: string. Width: 4 (305–308).
Object type: `phrase`. Feature identifier: `function`. Feature value type: enumeration.

`Adju`	Adjunct (505)	`Exst`	Existence (550)
`Cmpl`	Complement (504)	`Frnt`	Fronted element (572)
`Conj`	Conjunction (509)	`Intj`	Interjection (512)
`EPPr`	Enclitic personal pronoun (541)	`IntS`	Interjection with *SS* (522)
`ExsS`	Existence with SS^4 (552)	`Loca`	Locative (507)

`Modi`	Modifier (508)	`PreO`	Predicate with object suffix (531)
`ModS`	Modifier with *SS* (528)	`PreS`	Predicate with *SS* (532)
`NCop`	Negative copula (540)	`PtcO`	Participle with object suffix (534)
`NCoS`	Negative copula with *SS* (542)	`Ques`	Question (511)
`Nega`	Negation (510)	`Rela`	Relative (519)
`Objc`	Object (503)	`Subj`	Subject (502)
`PrAd`	Predicative adjunct (525)	`Supp`	Supplementary constituent (515)
`PrcS`	Predicate complement with *SS* (522)	`Time`	Time reference (506)
`PreC`	Predicate complement (521)	`Unkn`	Unknown (599)
`Pred`	Predicate (501)	`Voct`	Vocative (562)

Among these values we can distinguish the following subsets: a Nominal Copula Set {`ExsS`, `Exst`, `NCoS`, `NCop`}, a Predicate Complement Set {`PrcS`, `PreC`, `PtcO`}, a Predicate Set {`Pred`, `PreO`, `PreS`}, an Object Set {`Objc`, `PreO`, `PtcO`}, and a Subject Set {`ExsS`, `IntS`, `ModS`, `NCoS`, `PrcS`, `PreS`, `Subj`}.

The table has been copied from the manual page `ct(5)`, which is authoritative. The former labels `Irp[COPS]` have been removed, because they are equivalent to `function = Cmpl and type = IPrP`, and so on. The labels `Frn[ACOS]` have also been removed, because they are superseded by the phrase relation 'resumption'.

2.49 Clause atom number

Type: integer. Width: 4 (310–313).
Object type: `clause_atom`. Feature identifier: `number`. Feature value type: integer. Specifies the clause atom number within the book. The count starts at 1.

2.50 Clause atom type

Type: string. Width: 4 (315–318).
Object type: `clause_atom`. Feature identifier: `typ`. Feature value type: enumeration. The Emdros feature identifier is `typ` because `type` is a reserved word in MQL. The values have been taken from the manual page `PX(5)`, which is authoritative.

`AjCl`	Adjective clause (213)	`WQtX`	We-qatal-X clause (162)
`CPen`	Casus pendens (302)	`WxI0`	We-x-imperative-null clause (183)
`Defc`	Defective clause atom (0)	`WXIm`	We-X-imperative clause (173)
`Ellp`	Ellipsis (303)	`WxIX`	We-x-imperative-X clause (193)
`InfA`	Infinitive absolute clause (105)	`WxQ0`	We-x-qatal-null clause (182)
`InfC`	Infinitive construct clause (104)	`WXQt`	We-X-qatal clause (172)
`MSyn`	Macrosyntactic sign (304)	`WxQX`	We-x-qatal-X clause (192)
`NmCl`	Nominal clause (200)	`WxY0`	We-x-yiqtol-null clause (181)
`Ptcp`	Participle clause (106)	`WXYq`	We-X-yiqtol clause (171)
`Reop`	Reopening (305)	`WxYX`	We-x-yiqtol-X clause (191)
`Unkn`	Unknown (99)	`WYq0`	We-yiqtol-null clause (151)
`Voct`	Vocative clause (301)	`WYqX`	We-yiqtol-X clause (161)
`Way0`	Wayyiqtol-null clause (157)	`xIm0`	x-imperative-null clause (133)
`WayX`	Wayyiqtol-X clause (167)	`XImp`	X-imperative clause (123)
`WIm0`	We-imperative-null clause (153)	`xImX`	x-imperative-X clause (143)
`WImX`	We-imperative-X clause (163)	`XPos`	Extraposition (306)
`WQt0`	We-qatal-null clause (152)	`xQt0`	x-qatal-null clause (132)

[4] Subject suffix.

XQtl	X-qatal clause (122)	ZImX	Zero-imperative-X clause (113)
xQtX	x-qatal-X clause (142)	ZQt0	Zero-qatal-null clause (102)
xYq0	x-yiqtol-null clause (131)	ZQtX	Zero-qatal-X clause (112)
XYqt	X-yiqtol clause (121)	ZYq0	Zero-yiqtol-null clause (101)
xYqX	x-yiqtol-X clause (141)	ZYqX	Zero-yiqtol-X clause (111)
ZIm0	Zero-imperative-null clause (103)		

With these labels, a capital X refers to a stretch of constituents containing an explicit subject, whereas a lower case x refers to a stretch of constituents without a subject.

2.51 Distance to mother clause atom

Type: integer. Width: 4 (320–323).
Object type: clause_atom. Feature identifier: dist. Feature value type: integer. Specifies the distance in clause atoms to the mother clause atom. A distance of zero clause atoms and a zero relation mark the atom as the root of this tree of clause atom relations.

2.52 Clause atom relation

Type: integer. Width: 3 (325–327).
Object type: clause_atom. Feature identifier: code. Feature value type: integer. This section has been derived from the manual pages CARC(5) and CodesList(5), which are authoritative. Clause atom relation is denoted by a decimal code, which is constructed from the values displayed below. The method used for constructing the code is explained in the listing of the actual codes that follows.

Values used in constructing the codes

- Verbal tense of the predicate of the *whole* clause

0	none	3	imperative	6	participle[5]
1	imperfect	4	infinitive construct	7	wayyiqtol
2	perfect	5	infinitive absolute	8	weyiqtol†

  We have cancelled the value *weyiqtol*, as explained under *verbal tense* (2.20).

- Preposition class of the infinitive clause *atom*

0	none	9	B<D/†	18	*Not used*
1	>XR/	10	ZWLH/†	19	*Not used*
2	>L	11	J<N/	20	<D
3	>YL/†	12	K, KMW	21	<L
4	>T	13	*Not used*	22	<M
5	B, BMW	14	L, LMW	23	*Not used*
6	BJN/†	15	LM<N	24	TXT/
7	BL<DJ†	16	*Not used*		
8	*Not used*	17	MN		

- Conjunction class of the clause *atom* opening

[5] Both active and passive participle.

400 >W, W
500 >CR, DJ, H, ZW, KJ, C
600 >JN=, >LW, >M, HN, LHN=, LW, LWL>
800 PN

The conjunction class is determined by the conjunction that ends the clause opening conjunction phrase.

- Preposition class of the clause *atom* opening

700 >XR/, >L, B, BMW, VRM/, K, KMW, L, LMW, <D
800 BLT/, ZWLH/, LM<N, MN
900 J<N/, <L, <QB/

The preposition class is determined by the preposition that heads the clause opening conjunction phrase.

These classes are of a distributional, not functional, nature. They group lexemes which are hypothesised to share one or more functional aspects into tentative sets, so that the resulting clause atom relations codes constitute a useful collection of data for further research.

Possible constructed values

- 0. No relation.

  Zero indicates the absence of a clause atom relation. A zero relation and a distance of zero clause atoms mark the atom as the root of this tree of clause atom relations.

- 10–16. Relative clause atoms.

  A relative clause atom is characterised by an opening phrase of type `CP` and function `Rela`. The last digit is the tense of the verbal predicate of the daughter clause.

- 50–74. Infinitive construct clause atoms.

  Clause atoms of which the verbal predicate is an infinitive construct. The offset from 50 is the class of the preposition used in the construction.

- 100–167. Asyndetic clause atoms.

  Construction without a conjunction. The first digit in the offset from 100 is the tense of the verbal predicate of the daughter clause, the second that of the mother clause.

- 200–201. Parallel clause atoms.

  - 200 Identical clause atom opening.
  - 201 Identical clause atom opening when disregarding the coordinating conjunction(s).

  Two clause atoms are parallel if they concur in subject presence and have equivalent phrases up to the predicate, provided that the daughter is not subordinated. If either predicate is absent, the clause atoms must be of the same *clause atom type* (2.50).

- 220–223. Defective clause atoms.

  - 220 No verbal predicate in mother or daughter.
  - 221 Unclassified clause atom relation.†
  - 222 Verbal predicate in daughter clause atom.
  - 223 Verbal predicate in mother clause atom.

  A clause atom is defective if there is another clause atom which contains the predicate (or the main part) of the clause.

- 300–367. Conjunctive adverbs.

  Asyndetic construction, but with a conjunctive adverb. The first digit in the offset from 300 is the tense of the verbal predicate of the daughter clause, the second that of the mother clause.

- 400–487. Coordinate clause atoms.

  Construction with a conjunction from class 400, the 'coordinating' conjunctions. The first digit in the offset from 400 is the tense of the verbal predicate of the daughter clause, the second that of the mother clause.

- 500–567. Postulational clause atoms.

  Construction with a conjunction from class 500, the 'postulational' conjunctions. The first digit in the offset from 500 is the tense of the verbal predicate of the daughter clause, the second that of the mother clause.

- 600–667. Conditional clause atoms.

  Construction with a conjunction from class 600, the 'hypothetical' conjunctions. The first digit in the offset from 600 is the tense of the verbal predicate of the daughter clause, the second that of the mother clause.

- 700–767. Temporal clause atoms.

  Construction with a conjunction from class 700, the 'temporal' conjunctions. The first digit in the offset from 700 is the tense of the verbal predicate of the daughter clause, the second that of the mother clause.

- 800–867. Final clause atoms.

  Construction with a conjunction from class 800, the 'final' conjunctions. The first digit in the offset from 800 is the tense of the verbal predicate of the daughter clause, the second that of the mother clause.

- 900–967. Causal clause atoms.

  Construction with a conjunction from class 900, the 'causal' conjunctions. The first digit in the offset from 900 is the tense of the verbal predicate of the daughter clause, the second that of the mother clause.

- 999. Direct speech.

  This relation is issued to the daughter of an introduction of direct speech with a verbum dicendi. Also to the daughter confirming direct speech, opening a (small) paragraph with a verbum dicendi (for example Jes 40:1).

2.53 Clause number within sentence

Type: integer. Width: 3 (329–331).
Object type: `clause`. Feature identifier: `number`. Feature value type: integer.

2.54 Clause type

Type: string. Width: 4 (333–336).
Object type: `clause`. Feature identifier: `typ`. Feature value type: enumeration. The Emdros feature identifier is `typ` because `type` is a reserved word in MQL. The same values as *clause atom type* (2.50), with the exception of `Defc`, which applies to atoms only.

Clause kind

Object type: `clause`. Feature identifier: `kind`. Feature value type: enumeration. The values of *clause type* can be grouped into *clause kinds* as follows: {`unknown=0, VC=1, NC=2, WP=3`}.

Unknown `Unkn`.

Verbal clause `InfA`, `InfC`, `Ptcp`, `Way0`, `WayX`, `WIm0`, `WImX`, `WQt0`, `WQtX`, `WxI0`, `WXIm`, `WxIX`, `WxQ0`, `WXQt`, `WxQX`, `WxY0`, `WXYq`, `WxYX`, `WYq0`, `WYqX`, `xIm0`, `XImp`, `xImX`, `xQt0`, `XQtl`, `xQtX`, `xYq0`, `XYqt`, `xYqX`, `ZIm0`, `ZImX`, `ZQt0`, `ZQtX`, `ZYq0`, and `ZYqX`.

Nominal clause `AjCl` and `NmCl`.

Utterance without predication `CPen`, `Ellp`, `MSyn`, `Reop`, `Voct`, and `XPos`.

2.55 Clause constituent relation

Type: string. Width: 4 (338–341).
Object type: `clause`. Feature identifier: `rela`. Feature value type: enumeration.

`Adju`	Adjunctive clause (505)	`PreC`	Predicate complement clause (521)
`Attr`	Attributive clause (−13)	`ReVo`	Referral to vocative (562)
`Cmpl`	Complement clause (504)	`Resu`	Resumptive clause (572)
`Coor`	Coordinated clause (−6)	`RgRc`	Regens/rectum connection (−2)
`Objc`	Object clause (503)	`Spec`	Specification clause (−5)
`PrAd`	Predicative adjunct clause (525)	`Subj`	Subject clause (502)

These values have been taken from the manual page `PX(5)`, which is authoritative.

2.56 Distance to mother

Type: integer. Width: 4 (343–346).
Object type: `clause`. Feature identifier: `dist`. Feature value type: integer.

2.57 Unit used in counting distance to mother

Type: character. Width: 1 (348–348).
Object type: `clause`.

`C`	Clause Atoms	`P`	Phrase Atoms	`W`	Words

The unit is not a separate Emdros feature, but it does indicate the object type of the mother, just as it does with *phrase atom relations* (2.34). A distance in clause atoms implies that the mother is a clause, and a distance in phrase atoms implies that the mother is a phrase.

2.58 Tabulation

Type: integer. Width: 4 (350–353).
Object type: `clause_atom`. Feature identifier: `tab`. Feature value type: integer. The number of tab stops in the hierarchical display of the tree of clause atoms. This column used to be reserved for the clause label, which was intended to mark a clause as *question*, *command* and the like, but was never used.

2.59 Sentence atom number

Type: integer. Width: 4 (355–358).
Object type: `sentence_atom`. Feature identifier: `number`. Feature value type: integer. Specifies the sentence atom number within the book. The count starts at 1.

2.60 Sentence number within chapter

Type: integer. Width: 4 (360–363).
Object type: `sentence`. Feature identifier: `number`. Feature value type: integer.

2.61 Text type

Type: string. Width: 8 (365–372).
Object type: `clause`. Feature identifier: `txt`. Feature value type: string of variable length. *Text type* is a feature of a clause. That means that within one and the same clause, every clause atom must have the same value for text type. Its values are constructed in a cumulative manner. The clause לֹא תֹאכְלוּ מִכֹּל עֵץ הַגָּן in Genesis 3:1, for instance, is a quotation within a quotation in a narrative passage, and therefore marked `NQQ`. Text type is a concatenation of any of the following characters.

`?`	Unknown	`N`	Narrative
`D`	Discursive	`Q`	Quotation

These values have been taken from the manual page `PX(5)`, which is authoritative.

3 Example

There is a sample from the data (Gn 1:1–2) on page 18 as an example. It is presented in two blocks, because the lines are extremely long.

References

[Doedens94] C. F. J. Doedens. *Text Databases. One Database Model and Several Retrieval Languages*. PhD thesis, Rijksuniversiteit Utrecht, November 1994.

[Manual] UNIX manual pages at the WIVU: `at2ps(1)`, `syn01(1)`, `CARC(5)`, `CodesList(5)`, `PX(5)`, `ct(5)`, `morfset(5)`, `ps1(5)`, `ps2(5)`, `ps3(5)`.

```
GEN 01,01 A B.:-                    -1 .      -1 .         0 B           B.:-                    -1 .      -1 .       0 .     0 .       -1 -1 -1 -1 -1 -1
GEN 01,01 A R;>CI73JT               -1 .      -1 .         0 R>CJT/      R;>CIJT                 -1 .       1 /       0 .     0 .       -1 -1 -1  1  1  2
GEN 01,01 A B.@R@74>                 0 .       0 .         0 BR>[        B.@R@>                   1 [       0 .       0 .     0 .        0  2  3  1  2 -1
GEN 01,01 A >:ELOHI92JM             -1 .      -1 .         0 >LHJM/      >:ELOH                  -1 .       4 /IJM    0 .     0 .       -1 -1 -1  3  2  2
GEN 01,01 B >;71T                   -1 .      -1 .         0 >T          >;T                     -1 .      -1 .       0 .     0 .       -1 -1 -1 -1 -1 -1
GEN 01,01 B HA-                     -1 .      -1 .         0 H           HA-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,01 B C.@MA73JIM              -1 .      -1 .         0 CMJM/       C.@MAJ                  -1 .       4 /IM     0 .     0 .       -1 -1 -1  3  2  2
GEN 01,01 B W:-                     -1 .      -1 .         0 W           W:-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,01 B >;71T                   -1 .      -1 .         0 >T          >;T                     -1 .      -1 .       0 .     0 .       -1 -1 -1 -1 -1 -1
GEN 01,01 B H@-                     -1 .      -1 .         0 H           H@-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,01 B >@75REY00               -1 .      -1 .         0 >RY/        >@REY                   -1 .       1 /       0 .     0 .       -1 -1 -1  1  0  2
GEN 01,02 A W:-                     -1 .      -1 .         0 W           W:-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,02 A H@-                     -1 .      -1 .         0 H           H@-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,02 A >@81REY                 -1 .      -1 .         0 >RY/        >@REY                   -1 .       1 /       0 .     0 .       -1 -1 -1  1  0  2
GEN 01,02 A H@J:T@71H                0 .       0 .        -2 HJH[        H@J                      2 [:T@H    0 .       0 .     0 .        0  2  3  1  1 -1
GEN 01,02 A TO33HW.03               -1 .      -1 .         0 THW/        TOHW.                   -1 .       1 /       0 .     0 .       -1 -1 -1  1  2  2
GEN 01,02 A W@-                     -1 .      -1 .         0 W           W@-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,02 A BO80HW.                 -1 .      -1 .         0 BHW/        BOHW.                   -1 .       1 /       0 .     0 .       -1 -1 -1  1  2  2
GEN 01,02 A W:-                     -1 .      -1 .         0 W           W:-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,02 A XO73CEK:                -1 .      -1 .         0 XCK/        XOCEK:                  -1 .       1 /       0 .     0 .       -1 -1 -1  1  2  2
GEN 01,02 A <AL&                    -1 .      -1 .         0 <L          <AL                     -1 .      -1 .       0 .     0 .       -1 -1 -1 -1 -1 -1
GEN 01,02 A P.:N;74J                -1 .      -1 .         0 PNH/        P.:N                    -1 .       5 /;J     0 .     0 .       -1 -1 -1  3  2  1
GEN 01,02 A T:HO92WM                -1 .      -1 .         0 THWM/       T:HOWM                  -1 .       1 /       0 .     0 .       -1 -1 -1  1  1  2
GEN 01,02 B W:-                     -1 .      -1 .         0 W           W:-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,02 B R74W.XA                 -1 .      -1 .         0 RWX/        RW.XA                   -1 .       1 /       0 .     0 .       -1 -1 -1  1  0  1
GEN 01,02 B >:ELOHI80JM             -1 .      -1 .         0 >LHJM/      >:ELOH                  -1 .       4 /IJM    0 .     0 .       -1 -1 -1  3  2  2
GEN 01,02 B M:RAXE73PET              7 !M:!    0 .         0 RXP[        RAXEP                    1 [       3 /ET     0 .     0 .        1  6  0  1  1  2
GEN 01,02 B <AL&                    -1 .      -1 .         0 <L          <AL                     -1 .      -1 .       0 .     0 .       -1 -1 -1 -1 -1 -1
GEN 01,02 B P.:N;71J                -1 .      -1 .         0 PNH/        P.:N                    -1 .       5 /;J     0 .     0 .       -1 -1 -1  3  2  1
GEN 01,02 B HA-                     -1 .      -1 .         0 H           HA-                     -1 .      -1 .       0 .    -1 .       -1 -1 -1 -1 -1 -1
GEN 01,02 B M.@75JIM00              -1 .      -1 .         0 MJM/        M.@J                    -1 .       4 /IM     0 .     0 .       -1 -1 -1  3  2  2
```



```
B           B            1  5  5    1   . .   . . .   .     .   . .     .   . .     .   . 1   . . .     1 .       .   .   1 .    .       . .    .    1    1 .
R>CJT       R>CJT/       2  2  2    1   5 iD  . . .   .     .   . .     .   . .     .   . 1   5 iD Time   1 .       .   .   1 .    .       . .    .    1    1 .
BR>         BR>[         3  1  1    2   1 .   . . .   .     .   . .     .   . .     .   . 2   1 .  Pred   1 .       .   .   1 .    .       . .    .    1    1 .
>LHJM       >LHJM/       4  2  2    3   2 iD  . . .   .     .   . .     .   . .     .   . 3   2 iD Subj   1 .       .   .   1 .    .       . .    .    1    1 .
>T          >T           5  5  5    4   . .   . . .   .     .   . .     .   . .     .   . 4   . . .     1 .       .   .   1 .    .       . .    .    1    1 .
H           H            6  0  0    4   . .   . . .   .     .   . .     .   . .     .   . 4   . . .     1 .       .   .   1 .    .       . .    .    1    1 .
CMJM        CMJM/        7  2  2    4   . .   . . .   PAR  -2   0 .     .   . .     .   . 4   . . .     1 .       .   .   1 .    .       . .    .    1    1 .
W           W            8  6  6    4   . .   . . .   .     .   . .     .   . .     .   . 4   . . .     1 .       .   .   1 .    .       . .    .    1    1 .
>T          >T           9  5  5    4   . .   . . .   .     .   . .     .   . .     .   . 4   . . .     1 .       .   .   1 .    .       . .    .    1    1 .
H           H           10  0  0    4   . .   . . .   .     .   . .     .   . .     .   . 4   . . .     1 .       .   .   1 .    .       . .    .    1    1 .
>RY         >RY/        11  2  2    4   5 D   . . .   par  -2  -4 .     .   . .     .   . 4   5 D  Objc   1 xQtX    0   0   1 xQtX .       . .    0    1    1 ?
W           W           12  6  6    5   6 .   . . .   .     .   . .     .   . .     .   . 1   6 .  Conj   2 .       .   .   1 .    .       . .    .    2    2 .
H           H           13  0  0    6   . .   . . .   .     .   . .     .   . .     .   . 2   . . .     2 .       .   .   1 .    .       . .    .    2    2 .
>RY         >RY/        14  2  2    6   2 D   . . .   .     .   . .     .   . .     .   . 2   2 D  Subj   2 .       .   .   1 .    .       . .    .    2    2 .
HJTH        HJH[        15  1  1    7   1 .   . . .   .     .   . .     .   . .     .   . 3   1 .  Pred   2 .       .   .   1 .    .       . .    .    2    2 .
THW         THW/        16  2  2    8   . .   . . .   PAR   0   0 .     .   . .     .   . 4   . . .     2 .       .   .   1 .    .       . .    .    2    2 .
W           W           17  6  6    8   . .   . . .   .     .   . .     .   . .     .   . 4   . . .     2 .       .   .   1 .    .       . .    .    2    2 .
BHW         BHW/        18  2  2    8   2 iD  . . .   par   0  -2 .     .   . .     .   . 4   2 iD PreC   2 WXQt   -1 422   1 WXQt .       . .    2    2    2 ?
W           W           19  6  6    9   6 .   . . .   .     .   . .     .   . .     .   . 1   6 .  Conj   3 .       .   .   1 .    .       . .    .    3    3 .
XCK         XCK/        20  2  2   10   2 iD  . . .   .     .   . .     .   . .     .   . 2   2 iD Subj   3 .       .   .   1 .    .       . .    .    3    3 .
<L          <L          21  5  5   11   . .   . . .   .     .   . .     .   . .     .   . 3   . . .     3 .       .   .   1 .    .       . .    .    3    3 .
PNJ         PNH/        22  2  2   11   . .   . . .   REG   0   0 .     .   . .     .   . 3   . . .     3 .       .   .   1 .    .       . .    .    3    3 .
THWM        THWM/       23  2  2   11   5 iD  . . .   rec   0  -1 .     .   . .     .   . 3   5 iD PreC   3 NmCl   -1 402   1 NmCl .       . .    3    3    3 ?
W           W           24  6  6   12   6 .   . . .   .     .   . .     .   . .     .   . 1   6 .  Conj   4 .       .   .   1 .    .       . .    .    4    4 .
RWX         RWX/        25  2  2   13   . .   . . .   REG   0   0 .     .   . .     .   . 2   . . .     4 .       .   .   1 .    .       . .    .    4    4 .
>LHJM       >LHJM/      26  2  2   13   2 iD  . . .   rec   0  -1 .     .   . .     .   . 2   2 iD Subj   4 .       .   .   1 .    .       . .    .    4    4 .
MRXPT       RXP[        27  1  1   14   1 .   . . .   .     .   . .     .   . .     .   . 3   1 .  PreC   4 .       .   .   1 .    .       . .    .    4    4 .
<L          <L          28  5  5   15   . .   . . .   .     .   . .     .   . .     .   . 4   . . .     4 .       .   .   1 .    .       . .    .    4    4 .
PNJ         PNH/        29  2  2   15   . .   . . .   REG   0   0 .     .   . .     .   . 4   . . .     4 .       .   .   1 .    .       . .    .    4    4 .
H           H           30  0  0   15   . .   . . .   .     .   . .     .   . .     .   . 4   . . .     4 .       .   .   1 .    .       . .    .    4    4 .
MJM         MJM/        31  2  2   15   5 D   . . .   rec  -1  -2 .     .   . .     .   . 4   5 D  Cmpl   4 Ptcp   -1 460   1 Ptcp .       . .    4    4    4 ?
```