The ETCBC Data Model

Current developments

Hendrik Jan Bosman Constantijn Sikkel

Vrije Universiteit

February 18 2020

Overview

- Introduction
- Background of the ETCBC data model
- The ETCBC data model: Points to consider
- Current developments and the issues that spring from it
- Time for questions and discussion in preparation of the plenary discussion later this afternoon

Introduction

Our point of departure:

- The model stems from 1977 and has been gradually evolving ever since.
- It has been serving its purpose, but is not perfect.
- Progress is continually requiring adaptations.

What to expect this afternoon:

- A few glances at the model in development.
- The current developments and the issues we are facing.
- That I present work in progress.

Past and Present Implementations

Over the years, parts of the data model have been implemented using

- Punch cards (from 1977)
- Structured plain text files in 6-bit Display character set on a mainframe from the CDC 6000 series (from 1984)
- Structured plain text files in ASCII on a UNIX server (from 1990)
- An Emdros database engine running on top of an SQL database server (from 2001)
- Text Fabric in a Python programming environment (from 2013)

Some Features of the Model

- Stand-off markup (it predates XML)
- Overlapping hierarchies
- Facilitate a form-to-function approach
- Several implementations

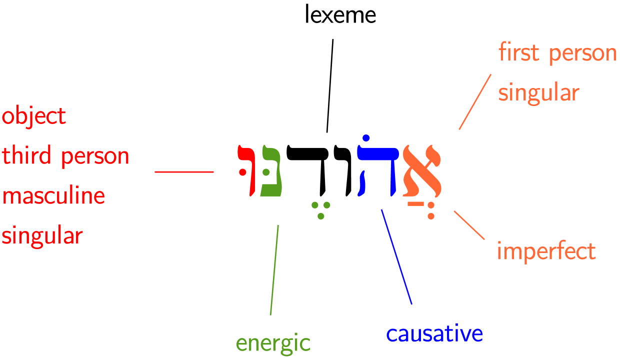
Rich Morphology

Ps 28:7

“I will glorify Him” (JPS 1999)

אֲהוֹדֶנּוּ

ahodennu



lexeme

first person

singular

object

third person

masculine

singular

אֲהוֹדֶנּוּ

imperfect

energic

causative

Figure: Grammatical functions marked by morphemes

Methodological Considerations

- Form to function

  The rich morphology of the semitic languages calls for a form-to-function approach.

  Hence *morphemes* are the smallest objects (analytic non-primary data) in the model.

- Pattern recognition

  The desire to *discover* the grammar beyond word level, rather than dictating it, drove us to use pattern recognition and not a rule-based approach.

  Hence the model also contains strictly linear object types, which we call *atoms*.

Parallel Hierarchies

Coalescence

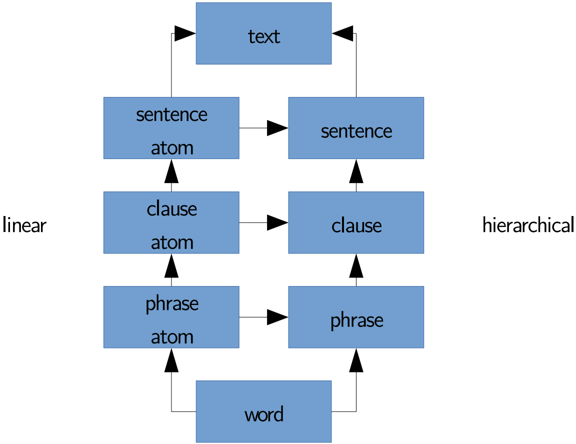


Figure: Coalescent hierarchies

Parallel Hierarchies

Document Structure

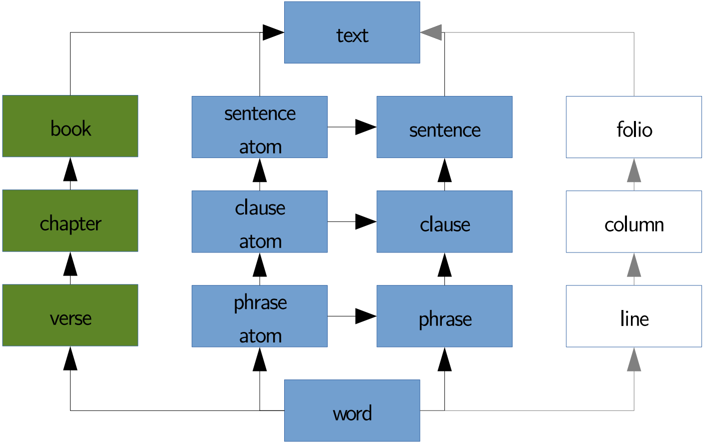


Figure: Parallel hierarchies

Parallel Hierarchies

Syllables and morphemes

Syllables and morphemes present two parallel hierarchies.

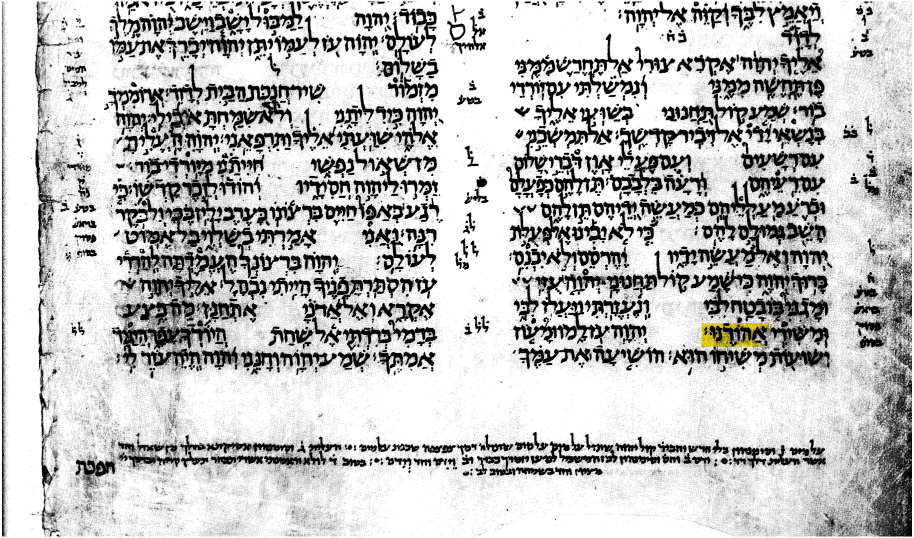
Take, for instance, the German word Unterhaltungssendung (entertainment broadcast):

- Un-ter-hal-tungs-sen-dung (syllables)
- Unter-halt-ung-s-send-ung (morphemes)

The ETCBC Model: Points to Consider

- Preparation of the primary data
- Objects in the database
- Linguistic levels of analysis
- Query languages

Primary Data

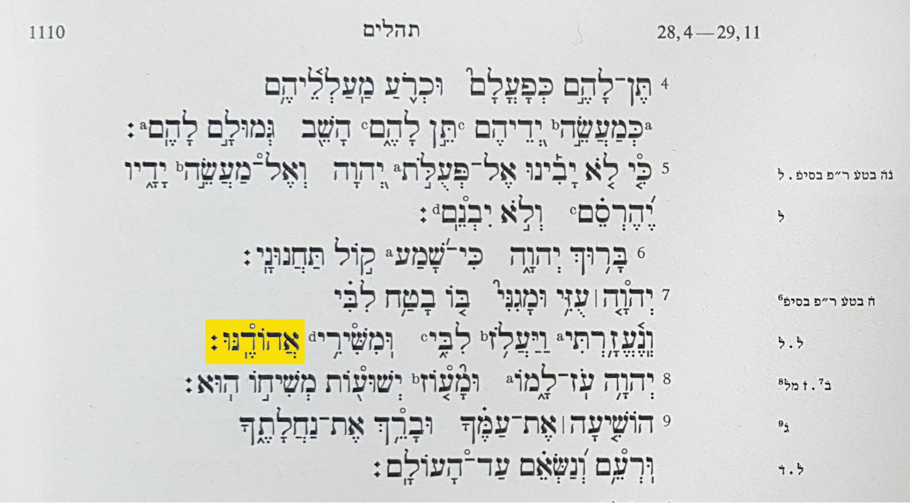


יַאֲמֵץ לִבֶּךָ וְקַוֵּה אֶל־יְהוָה׃
לְדָוִד
אֵלֶיךָ יְהוָה אֶקְרָא צוּרִי אַל־תֶּחֱרַשׁ מִמֶּנִּי
פֶּן־תֶּחֱשֶׁה מִמֶּנִּי וְנִמְשַׁלְתִּי עִם־יוֹרְדֵי
בוֹר׃ שְׁמַע קוֹל תַּחֲנוּנַי בְּשַׁוְּעִי אֵלֶיךָ
בְּנָשְׂאִי יָדַי אֶל־דְּבִיר קָדְשֶׁךָ׃ אַל־תִּמְשְׁכֵנִי
עִם־רְשָׁעִים וְעִם־פֹּעֲלֵי אָוֶן דֹּבְרֵי שָׁלוֹם
עִם־רֵעֵיהֶם וְרָעָה בִּלְבָבָם׃ תֶּן־לָהֶם כְּפָעֳלָם
וּכְרֹעַ מַעַלְלֵיהֶם כְּמַעֲשֵׂה יְדֵיהֶם תֵּן לָהֶם
הָשֵׁב גְּמוּלָם לָהֶם׃ כִּי לֹא יָבִינוּ אֶל־פְּעֻלֹּת
יְהוָה וְאֶל־מַעֲשֵׂה יָדָיו יֶהֶרְסֵם וְלֹא יִבְנֵם׃
בָּרוּךְ יְהוָה כִּי־שָׁמַע קוֹל תַּחֲנוּנָי׃ יְהוָה עֻזִּי
וּמָגִנִּי בּוֹ בָטַח לִבִּי וְנֶעֱזָרְתִּי וַיַּעֲלֹז לִבִּי
וּמִשִּׁירִי אֲהוֹדֶנּוּ׃ יְהוָה עֹז־לָמוֹ וּמָעוֹז
יְשׁוּעוֹת מְשִׁיחוֹ הוּא׃ הוֹשִׁיעָה אֶת־עַמֶּךָ

כָּבוֹד׃ יְהוָה לַמַּבּוּל יָשָׁב וַיֵּשֶׁב יְהוָה מֶלֶךְ
לְעוֹלָם׃ יְהוָה עֹז לְעַמּוֹ יִתֵּן יְהוָה יְבָרֵךְ אֶת־עַמּוֹ
בַשָּׁלוֹם׃
מִזְמוֹר שִׁיר־חֲנֻכַּת הַבַּיִת לְדָוִד׃ אֲרוֹמִמְךָ
יְהוָה כִּי דִלִּיתָנִי וְלֹא־שִׂמַּחְתָּ אֹיְבַי לִי׃ יְהוָה
אֱלֹהָי שִׁוַּעְתִּי אֵלֶיךָ וַתִּרְפָּאֵנִי׃ יְהוָה הֶעֱלִיתָ
מִן־שְׁאוֹל נַפְשִׁי חִיִּיתַנִי מִיָּרְדִי־בוֹר׃
זַמְּרוּ לַיהוָה חֲסִידָיו וְהוֹדוּ לְזֵכֶר קָדְשׁוֹ׃ כִּי
רֶגַע בְּאַפּוֹ חַיִּים בִּרְצוֹנוֹ בָּעֶרֶב יָלִין בֶּכִי וְלַבֹּקֶר
רִנָּה׃ וַאֲנִי אָמַרְתִּי בְשַׁלְוִי בַּל־אֶמּוֹט
לְעוֹלָם׃ יְהוָה בִּרְצוֹנְךָ הֶעֱמַדְתָּה לְהַרְרִי
עֹז הִסְתַּרְתָּ פָנֶיךָ הָיִיתִי נִבְהָל׃ אֵלֶיךָ יְהוָה
אֶקְרָא וְאֶל־אֲדֹנָי אֶתְחַנָּן׃ מַה־בֶּצַע
בְּדָמִי בְּרִדְתִּי אֶל־שָׁחַת הֲיוֹדְךָ עָפָר הֲיַגִּיד
אֲמִתֶּךָ׃ שְׁמַע־יְהוָה וְחָנֵּנִי יְהוָה הֱיֵה־עֹזֵר לִי׃

Figure: Psalm 28 in Codex Leningradensis

Biblia Hebraica Stuttgartensia



1110 תהלים 28,4—29,11

4 תֶּן־לָהֶ֣ם כְּפָעֳלָם֮ וּכְרֹ֪עַ מַֽעַלְלֵיהֶ֥ם
[a]כְּמַעֲשֵׂ֣ה[b] יְ֭דֵיהֶם [c]תֵּ֣ן לָהֶ֑ם[c] הָשֵׁ֖ב גְּמוּלָ֣ם לָהֶֽם[a]׃
5 כִּ֤י לֹ֤א יָבִ֡ינוּ אֶל־פְּעֻלֹּ֣ת[a] יְ֭הוָה וְאֶל־מַעֲשֵׂ֣ה[b] יָדָ֑יו נה בטע ר״פ בסיפ . ל
יֶ֝הֶרְסֵ֗ם[c] וְלֹ֣א יִבְנֵֽם[d]׃ ל
6 בָּר֥וּךְ יְהוָ֑ה כִּי־שָׁ֝מַע[a] ק֣וֹל תַּחֲנוּנָֽי׃
7 יְהוָ֤ה ׀ עֻזִּ֥י וּמָגִנִּי֮ בּ֤וֹ בָטַ֪ח לִבִּ֡י ח בטע ר״פ בסיפ[6]
וְֽנֶעֱזָ֗רְתִּי[a] וַיַּעֲלֹ֣ז[b] לִבִּ֑י[c] וּֽמִשִּׁירִ֥י[d] אֲהוֹדֶֽנּוּ׃ ל . ל
8 יְהוָ֥ה עֹֽז־לָ֑מוֹ[a] וּמָ֘ע֤וֹז[b] יְשׁוּע֖וֹת מְשִׁיח֣וֹ הֽוּא׃ ב[7] . ז מל[8]
9 הוֹשִׁ֤יעָה ׀ אֶת־עַמֶּ֗ךָ וּבָרֵ֥ךְ אֶת־נַחֲלָתֶ֑ךָ ג[9]
וּֽרְעֵ֥ם וְ֝נַשְּׂאֵ֗ם עַד־הָעוֹלָֽם׃ ל . ד

Figure: Psalm 28 in the Biblia Hebraica Stuttgartensia

Types of Graphemes

Ps 28:7

“I will glorify Him.” (JPS 1999)

אֲהוֹדֶֽנּוּ׃

‘ahodennu’

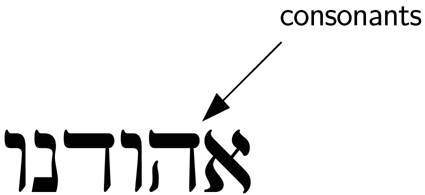
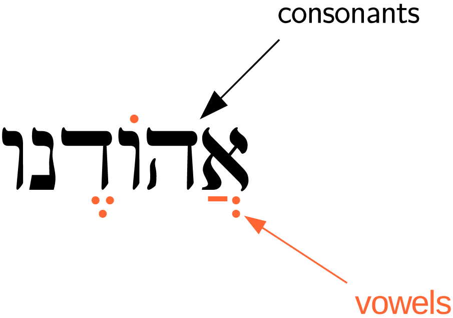


Figure: Five types of graphemes

Types of Graphemes

Ps 28:7 אֲהוֹדֶנּוּ׃

"I will glorify Him." (JPS 1999) *'ahodennu'*



consonants

vowels

Figure: Five types of graphemes

Types of Graphemes

Ps 28:7

“I will glorify Him.” (JPS 1999)

אֲהוֹדֶנּוּ׃

‘ahodennu’

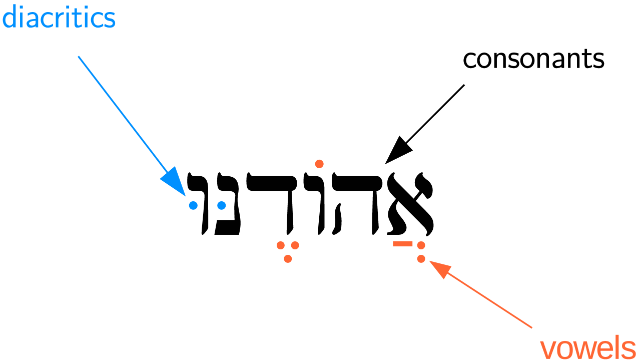


Figure: Five types of graphemes

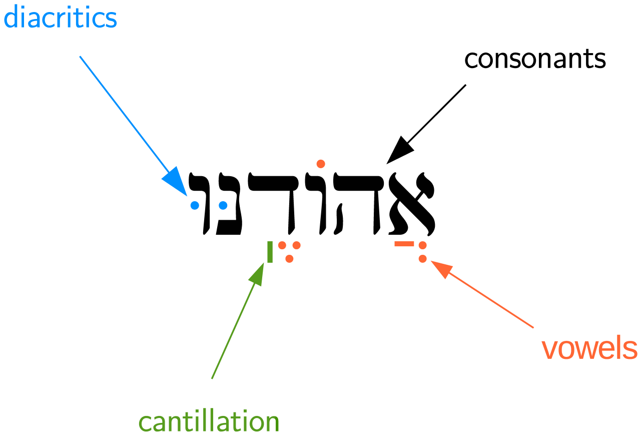
Types of Graphemes

Ps 28:7

“I will glorify Him.” (JPS 1999)

אֲהוֹדֶֽנּוּ׃

‘ahodennu’



diacritics

consonants

cantillation

vowels

Figure: Five types of graphemes

Types of Graphemes

Ps 28:7

“I will glorify Him.” (JPS 1999)

אֲהוֹדֶֽנּוּ׃

‘ahodennu’

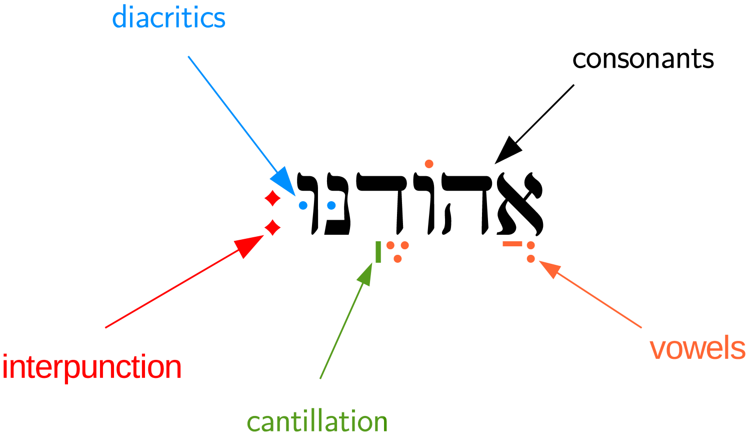


Figure: Five types of graphemes

To One Dimension

- The text comes to us on a two-dimensional substrate as an arrangement of characters which are read in a certain order.
- The two-dimensional text is reduced to a one-dimensional string of graphemes.
- This yields a sequence of objects of which their textual position is mapped to the mathematical set of the integers.
- These integers, called monads, are the coordinate system of the database.

…	>	:A	H	O	W	D	E	75	N	.	W	.	00	…
37	38	39	40	41	42	43	44	45	46	47	48	49	50	51

Grapheme

ܦܝܫܘܢ *Pishon* ܩܝܫܘܢ *Qishon* ܒܝܫܘܢ *Bashan*

```
[grapheme
   id = qof, position = initial, folio = "34v",
   line = 12, index = 7, style = "estrangela",
   ids = {qof,beth,pe}, certainty = {0.71,0.24,0.05},
   x = 37.518, y = 15.773, height = 60, width = 67,
   pixels = "b3a5b3ff302c30ff..."
]
```

This grapheme in the database is a Syriac letter *qof* in initial position, written in estrangela and is the seventh grapheme on line twelve of folio 34 *verso*. The letter was not recognised with absolute certainty. It could also be a *beth* or a *pe*, but with a lower probability (estimated 24% and 5% repectively). The last five features give some more details of the optical character recognition.

Database Objects

Every object has:

- An *object type*, which determines to which class of object it belongs. For example, morpheme, word, clause.
- A *unique identifier*.
- A *monad set*, which determines its position in the text and hence the graphemes which are part of it.
- One or more *features*, with their values.

Database Objects

...	>	:A	H	O	W	D	E	75	N	.	W	.	00	...
37	38	39	40	41	42	43	44	45	46	47	48	49	50	51

So, for instance:

```
[word
   self = 0x24633f88, monad_set = {38-47},
   surface = ">:AHOWDEN.", part_of_speech = verb,
   verbal_tense = imperfect, person = first
]
[word
   self = 0xd357091d, monad_set = {48-49},
   surface = "W.", part_of_speech = personal_pronoun,
   person = third, number = singular, gender = masculine
]
```

Phrase Level Objects

...	>	:A	H	O	W	D	E	75	N	.	W	.	00	...
37	38	39	40	41	42	43	44	45	46	47	48	49	50	51

But also:

```
[phrase
   self = 0xc3071235, monad_set = {38-47},
   type = verbal_phrase, function = predicate
]
[phrase
   self = 0x176d84f1, monad_set = {48-49},
   type = personal_pronoun_phrase, function = object
]
```

Current Developments

Atoms	The relationship between the linear and hierarchical analysis, which used to be practical, now becomes formalised.
Elisions	Analytical objects (words, phrases) that do not actually appear in the text, but influence the linguistic analysis of that text as if they did, need to be recorded.
Dislocation	The *casus pendens* construction, with which we address left dislocation, gets generalised so we can deal with right dislocation as well.
Participants	Research into coreference resolution and participant analysis makes it necessary to have objects and relations which can store its outcome and make retrieval possible.
Valency	In order to link predicates to the active valency pattern, we are going to rearrange our parsing labels into three dimensions: grammatical relations, complementariness, and semantic roles.

Atoms

- Atoms represent the text as a linear stream of tokens pertaining to a certain object type.
- They are called *atoms* because their monad sets are continuous.
- They exist if some object types are ordered in such a way that the relational operations *less than*, *equal to*, and *greater than* are defined on them.

Atom Algorithm

Find headnode

```
procedure find_head_node(node, type)
   atom_set: monad_set_t;
begin
   if node.type <> type then
      for every child of node do
         find_head_node(child, type)
   else begin
      atom_set := node.monad_set;
      visit(node, type, atom_set);
      print_atom_set(node, atom_set)
   end
end
```

Pseudo-code of the first step in the algorithm for the division into atoms: finding the headnodes.

Atom Algorithm

Visit headnode

```
procedure visit(node, type, monad_set)
begin
  for every child of node do
    if child.type <= type then
      visit(child, child.type, monad_set)
    else begin
      monad_set := monad_set - child.monad_set;
      find_head_node(node, child.type)
    end
end
```

Pseudo-code of the second step in the algorithm for the division into atoms: visiting the headnodes.

Elision of the Article

After a one-letter preposition, the article is absorbed by the two encompassing morphemes. It is no longer there, but has left its traces.

Dt 32:10	desert	MID:B.@R	מִדְבָּר
?	in a desert	B.:MID:B.@R	בְּמִדְבָּר
Gn 14:6	the desert	HAM.ID:B.@R	הַמִּדְבָּר
Gn 16:7	in the desert	B.AM.ID:B.@R	בַּמִּדְבָּר

Yet elision does not *always* occur:

Chr 23:10	to the altar	LAM.IZ:B.;XA	לַמִּזְבֵּחַ
Chr 29:27	to the altar	L:HAM.IZ:B.;XA	לְהַמִּזְבֵּחַ

Elements without a Textual Representation

Gn 31:10 וָאֶשָּׂא עֵינַי וָאֵרֶא בַּחֲלוֹם

“I lifted up mine eyes, and saw in *a* dream” (JPS 1917)

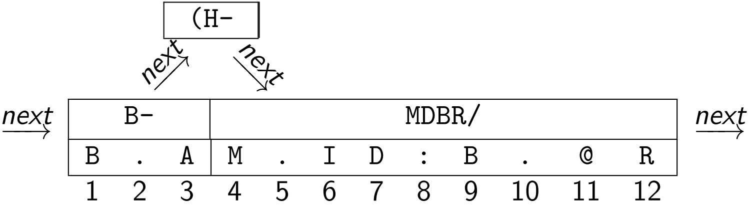
Gn 31:11 וַיֹּאמֶר אֵלַי מַלְאַךְ הָאֱלֹהִים בַּחֲלוֹם

“And the angel of God said unto me in *the* dream” (JPS 1917)

Objects Without Realisation

Virtual objects

Dt 1:1 *In the desert* בַּמִּדְבָּר



(H-

next *next*

next *next*

B-			MDBR/								
B	.	A	M	.	I	D	:	B	.	@	R
1	2	3	4	5	6	7	8	9	10	11	12

- *‘next’-edges* determine word sequence: B-(H-MDBR/
- (H- has an *empty monad set* {}.
- (H- can be located, within monads {3-4}, ‘between’ 3 and 4
- B-, (H- and MDBR/ are *consecutive*

Adjacency

Several notions of adjacency:

- Objects can be *contiguous* (actually touching): the last monad of O_1 is one less than the first monad of O_2. They are side-by-side in the *primary data*.
- Objects can be *adjacent* (like two houses with a driveway in between them): O_1 and O_2 are adjacent when on the monads between O_1 and O_2 no objects of the object type of O_1 or O_2 can be found. They are side-by-side within their *object type*.
- Objects can be *consecutive* (the one comes immediately after the other): O_1 and O_2 are consecutive if the relation 'next' of O_1 points to O_2. They are side-by-side on an *analytical path*.

Dislocation

Left dislocation:
Gn 42:11

כֻּלָּנוּ בְּנֵי אִישׁ־אֶחָד נָחְנוּ

We all, sons of one man are we.

Right dislocation:
Gn 35:6

וַיָּבֹא יַעֲקֹב לוּזָה אֲשֶׁר בְּאֶרֶץ כְּנַעַן הִוא בֵּית־אֵל
הוּא וְכָל־הָעָם אֲשֶׁר־עִמּוֹ

“Thus Jacob came to Luz—that is, Bethel—in the land of Canaan, he and all the people who were with him.” (JPS 1999)

New in the data model:

- Introduction of a clause type Right Dislocation.
- Introduction of a grammatical relation Dislocated Element.

Dislocation

```
Clause atom 50   LDis    [KLNW <DE>]
Clause atom 51   NmCl      [BNJ >JC >XD <PC>] [NXNW <Su>]
```

Figure: Gn 42:11 (left dislocation)

```
Clause atom 30   WayX   [W-<Cj>] [JB> <Pr>] [J<QB <Su>] [LWZH <Co>]
Clause atom 31   NmCl         [>CR <Re>] [B->RY KN<N <PC>]
Clause atom 32   NmCl       [HW> <Su>] [BJT_>L <PC>]
Clause atom 33   RDis     [HW> W-KL H-<M <DE>]
Clause atom 34   NmCl       [>CR <Re>] [<MW <PC>]
```

Figure: Gn 35:6 (right dislocation)

Legend	DE	=	Dislocated Element
	LDis	=	Left Dislocation
	NmCl	=	Nominal Clause
	RDis	=	Right Dislocation

Communication Types

Narrative The narrator is telling a story. *(N)*

Quotation Direct speech: A participant is speaking. *(Q)*

Discursive The narrator suspends the story and addresses the reader directly. *(D)*

Concepts and Notions

Main participants

speaker Actor who is the *source* of the communication, viewed from outside the domain.

audience Actor to whom the communication is directed, viewed from outside the domain.

sender Actor who is the *source* of the communication, viewed from within the domain.

addressee Actor to whom the communication is directed, viewed from within the domain.

Participants

Review of Domain

Domain A domain is characterised by the four main participants that constitute the communication. In theory there are two sets of ‘owners’, one viewed from the outside (*Speaker* and *Audience*), and one viewed from the inside of the domain (*Sender* and *Addressee*).

Speaker		Audience
Domain 4	Q	Atom 7
§4.2.3		Text type NQ
Sender		Addressee

Table: Properties of a Domain

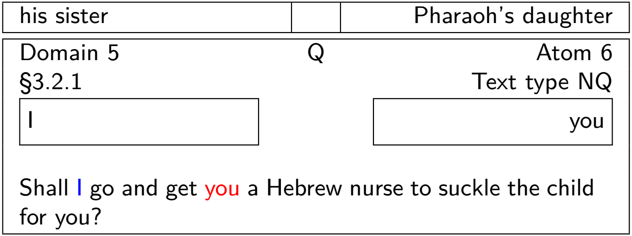
Complete Model

All main participants are explicit

Ex 2:7 shows a domain in which all main participants are explicit.

```
WayX N     32     30   5.#   [W-<Cj>] [T>MR <Pr>] [>XTW <Su>] [>L BT PR<H <Co>]
     =====                    |+==============================================================\
xYq0 NQ    321    31   6.q     | [H-<Qu>] [>LK <Pr>]
WQt0 NQ    321    32   7..     |     [W-<Cj>] [QR>TJ <Pr>] [LK <Co>] [>CH MJNQT MN H-<BRJT <Ob>]
WYq0 NQ    321    33   8..     |         [W-<Cj>] [TJNQ <Pr>] [LK <Aj>] [>T H-JLD <Ob>]
     =====                    |+==============================================================/
```

Then his sister said to Pharaoh's daughter:



his sister

Pharaoh's daughter

Domain 5

Q

Atom 6

§3.2.1

Text type NQ

I

you

Shall I go and get you a Hebrew nurse to suckle the child for you?

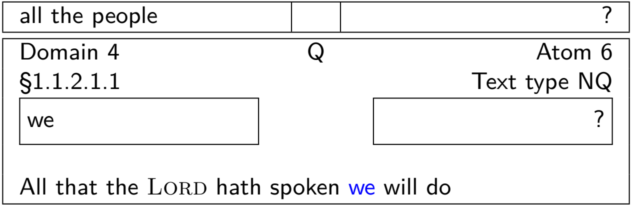
Incomplete Model

Audience and addressee remain unnamed

Ex 19:8 shows a domain in which only the speaker and the sender are explicit.

```
EXO 19,08 WayX N     1121  29  5.#  [W-<Cj>] [J<NW <Pr>] [KL H-<M <Su>] [JXDW <Mo>]
EXO 19,08 Way0 N     1121  30  6..      [W-<Cj>] [J>MRW <Pr>]
               =====                       +========================================\
EXO 19,08 Defc NQ    11211 31  7dq        || [KL <Ob>]
EXO 19,08 xQtX NQ    11211 32  9.e        ||      |  [>CR <Re>] [DBR <Pr>] [JHWH <Su>]
EXO 19,08 ZYq0 NQ    11211 33  8..        ||     [N<FH <Pr>]
               =====                       +========================================/
```

And all the people answered together, and said:



all the people ?

Domain 4 Q Atom 6

§1.1.2.1.1 Text type NQ

we ?

All that the LORD hath spoken we will do

Concepts and Notions

Referential

- PRef (participant reference): phrase or subphrase that introduces or refers to a participant.
- PSet: set of participant references within one domain, that refer to the same actor.
- PAct (actor): collection of sets of participant references identified across domain borders, referring to the same actor.
- Participant: set of actors that share the same referent in the text.

Partcicipants

PRef: Participant References

Ex 2:7 וַתֹּאמֶר אֲחֹתוֹ אֶל־בַּת־פַּרְעֹה

“Then his sister said to Pharaoh’s daughter” (JPS 1999)

Here אֲחֹתוֹ represents two phrases and two participant references.

ps	nu	gn		
	sg	f	אֲחֹתוֹ	his sister
3	sg	m	וֹ-	he

PRef Participant references are phrases with the grammatical functions of person, number or gender. This means that phrases can be nested and inherit these grammatical functions from the way they are constructed.

Participants

Review of PSet

Ex 2:7 הַאֵלֵךְ וְקָרָאתִי לָךְ אִשָּׁה מֵינֶקֶת מִן הָעִבְרִיֹּת וְתֵינִק לָךְ אֶת־הַיָּלֶד

“Shall I go and call thee a nurse of the Hebrew women, that she may nurse the child for thee?” (JPS 1917)

PRef	PSet	ps	nu	gn	phrase
81	22	1	sg		אֵלֵךְ
82	22	1	sg		קָרָאתִי
83	23	2	sg	f	־ָךְ
84	24		sg	f	אִשָּׁה
86	26		pl	f	עִבְרִיֹּת
87	24	3	sg	f	תֵינִק
88	23	2	sg	f	־ָךְ
89	27		sg	m	הַיָּלֶד

PSet Within the confines of a single domain, the participant reference set unites the participant references which refer to the same actor.

Participants

Review of PAct

Ex 2:5–10

PSet	ps	nu	gn		
9	3	sg	f	תַּחְמֹל, לִרְחֹץ, בַּת־פַּרְעֹה, הָ-, ָהּ-, תִּקְרָא, תִּקָּחֶהָ, תִּפְתַּח, תֵּרֶד, תֵּרֶא, תֹּאמֶר, תִּשְׁלַח, תִּרְאֵהוּ	her, her, Pharaoh's daughter, ..., she said
23	2	sg	f	ָךְ-	you
34	1	sg		אֲנִי, אֶתֵּן, ִי-	I, I shall give, me
38	1	sg		מְשִׁיתִהוּ	I drew him

Table: PAct 9, 27×, label = בת פרעה

PAct A PAct is a collection of sets of participant references identified across domain borders, which refer to the same actor.

Valency

Current Analysis

Ez 8:17 כִּי־מָלְאוּ אֶת־הָאָרֶץ חָמָס

“...that they must fill the country with lawlessness” (JPS 1999)

```
[KJ <Cj>] [ML>W <Pr>] [>T H->RY <Ob>] [XMS <Ob>]
```

Gn 6:11 וַתִּמָּלֵא הָאָרֶץ חָמָס

“...the earth was filled with lawlessness” (JPS 1999)

```
[W-<Cj>] [TML> <Pr>] [H->RY <Su>] [XMS <Ob>]
```

Gn 1:17 וַיִּתֵּן אֹתָם אֱלֹהִים בִּרְקִיעַ הַשָּׁמָיִם

“And God set them in the expanse of the sky” (JPS 1999)

```
[W-<Cj>] [JTN <Pr>] [>TM <Ob>] [>LHJM <Su>] [B-RQJ< H-CMJM <Co>]
```

Conjecture

The textgrammatical rules that govern the clauses (sentences) that connect domains*, differ from the classical textgrammatical rules, because those are only valid within the confines of a domain.*

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

Dick Grune and Ceriel J. H. Jacobs.
Parsing Techniques. A Practical Guide.
Springer, second edition, 2007.

Eep Talstra.
Approaching the mountain of Exodus 19: thou shalt explore syntax first.
HIPHIL Novum, 3(1):2–24, June 2019.