Conceptual Logical Structures:

The Lexico-Conceptual Linkage in Role and Reference Grammar

Carlos Periñán[1] & Ricardo Mairal[2]

[1] Universidad Católica San Antonio (Spain)

[2] UNED (Spain)

Logical Structures

Betty asked Bill for an apple.

$<_{\text{IF DECL}}$ $<_{\text{TNS PAST}}$ <[do' (Betty, [say' (Betty, Bill)])] PURP [do' (Bill, 0)]

CAUSE [BECOME have' (Betty, apple)]>>>

Logical Structures

Arreglar

(i) put into a proper or systematic order: (= **ARRANGE**)

e.g. *Mi madre arregló las flores del jarrón*

[do' (x, 0)] CAUSE [BECOME arranged' (y)]

(ii) restore by replacing a part or putting together what is torn or broken (= **REPAIR**)

e.g. *Mi padre arregló el televisor*

[do' (x, 0)] CAUSE [BECOME repaired' (y)]

Conceptual Logical Structure

Betty asked Bill for an apple.

$<_{\text{IF DECL}}$ $<_{\text{TNS PAST}}$ <[do (%BETTY_00$_{\text{Theme}}$, [+REQUEST_01 (%BETTY_00$_{\text{Theme}}$, %BILL_00$_{\text{Goal}}$)])] PURP [do (%BILL_00$_{\text{Goal}}$, 0)] CAUSE [BECOME +REQUEST_01 (%BETTY_00$_{\text{Theme}}$, +APPLE_00$_{\text{Referent}}$)]>>>

- **Language-independent representation**
- **Lexico-conceptual linkage**

FunGramKB Lexicon English
dry

FunGramKB Lexicon Spanish
secar

FunGramKB Editor

Ontology

298 events

SEARCH

- +COOL_00
- +CURE_00
- +DAMAGE_00
- +DECORATE_00
- +DECREASE_00
- +DRY_01
- +FILL_00
- +HARDEN_00
- +HEAT_00
- +IMPROVE_00
- +INCREASE_00
- +INFECT_00
- +LENGTHEN_00

Entities Events Qualities

Edit structure

Conceptual Information:

CONCEPT:	+DRY_01
SUPERORDINATE(S):	+CHANGE_00
THEMATIC FRAME:	(x1)Theme (x2: +CORPUSCULAR_00)Referent
MEANING POSTULATE:	+(e1: +CHANGE_00 (x1)Theme (x2)Referent (f1: (e2: n +BECOME_00 (x2)Theme (x3: +WET_00) Attribute))Result)
DESCRIPTION:	remove the moisture from and make dry; "dry clothes"; "dry hair"

Edit properties

LCM CORE GRAMMAR:

AktionsArt:

- [ ] State
- [ ] Activity
- [x] Accomplishment
- [ ] Achievement

You determine the canonical lexical class(es) of the verb.

Lexical Template:

Variables: x, y

Idiosyncratic features:

[MR <-- no value selected --> **], [U =** <-- no value selected --> **]**

Thematic frame mapping:

X = Theme **, Y =** Referent **, Z =** [no function]

A REMINDER OF FUNGRAMKB PARTICIPANTS:

THEME: Entity that transforms another entity.

REFERENT: Entity that is transformed by another entity.

LCM CORE GRAMMAR:

AktionsArt:

- [ ] State
- [ ] Activity
- [x] Accomplishment
- [ ] Achievement

You determine the canonical lexical class(es) of the verb.

Lexical Template:

Variables: x, y

Idiosyncratic features:

[MR <-- no value selected --> **], [U =** <-- no value selected --> **]**

Thematic frame mapping:

X = Theme **, Y =** Referent **, Z =** [no function]

A REMINDER OF FUNGRAMKB PARTICIPANTS:

THEME: Entity that transforms another entity.

REFERENT: Entity that is transformed by another entity.

LCM CORE GRAMMAR:

AktionsArt:

- [ ] State
- [ ] Activity
- [x] Accomplishment
- [ ] Achievement

You determine the canonical lexical class(es) of the verb.

Lexical Template:

Variables: x, y

Idiosyncratic features:

[MR <-- no value selected --> **], [U =** <-- no value selected --> **]**

Thematic frame mapping:

X = Theme **, Y =** Referent **, Z =** [no function]

A REMINDER OF FUNGRAMKB PARTICIPANTS:

THEME: Entity that transforms another entity.
REFERENT: Entity that is transformed by another entity.

LCM CORE GRAMMAR:

AktionsArt:

- [ ] State
- [ ] Activity
- [x] Accomplishment
- [ ] Achievement

You determine the canonical lexical class(es) of the verb.

Lexical Template:

Variables: x, y

Idiosyncratic features:

[MR <-- no value selected --> **], [U =** <-- no value selected --> **]**

Thematic frame mapping:

X = Theme **, Y =** Referent **, Z =** [no function]

A REMINDER OF FUNGRAMKB PARTICIPANTS:

THEME: Entity that transforms another entity.
REFERENT: Entity that is transformed by another entity.

LCM CORE GRAMMAR:

AktionsArt:

- [ ] State
- [ ] Activity
- [x] Accomplishment
- [ ] Achievement

You determine the canonical lexical class(es) of the verb.

Lexical Template:

Variables: x, y

Idiosyncratic features:

[MR <-- no value selected --> **], [U =** <-- no value selected --> **]**

Thematic frame mapping:

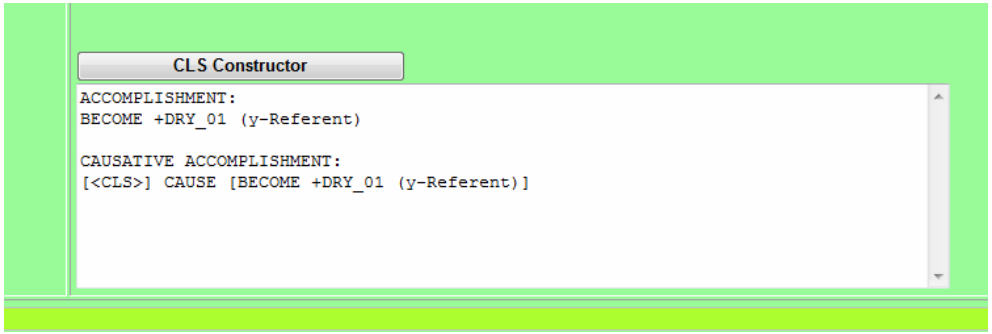
X = Theme **, Y =** Referent **, Z =** [no function]

A REMINDER OF FUNGRAMKB PARTICIPANTS:

THEME: Entity that transforms another entity.

REFERENT: Entity that is transformed by another entity.

Concept: +OPEN_01
Thematic Frame: (x1)Agent (x2: +DOOR_00 ^ +WINDOW_00)Theme (x3)Location (x4)Origin (x5)Goal
Thematic-Frame Mapping: x = Agent, y = Theme



FunGramKB Lexicon$_{English}$

dry

LCM Core Grammar

Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent

Conceptual Logical Structures

BECOME +DRY_01 ($y_{Referent}$)

[do (x_{Theme}, 0)] CAUSE [BECOME +DRY_01 ($y_{Referent}$)]

CLS Constructor

FunGramKB Lexicon$_{Spanish}$

LCM Core Grammar

Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent

secar

CLS Constructor

State	<C> (<VAR>)	<STA>
Activity	do (x [<STA>])	<ACT>
Accomplishment	BECOME <STA>	<ACC>
Achievement	INGR <STA>	<ACH>
Semelfactive	SEML <STA>	<SEM>
Active accomplishment	do (x [<C> (x)]) & <ACH>	<ACA>
Causative state	[<CLS>] CAUSE [<STA>]	<CSTA>
Causative activity	[<CLS>] CAUSE [<ACT>]	<CACT>
Causative accomplishment	[<CLS>] CAUSE [<ACC>]	<CACC>
Causative achievement	[<CLS>] CAUSE [<ACH>]	<CACH>
Causative semelfactive	[<CLS>] CAUSE [<SEM>]	<CSEM>
Causative active accomplishment	[<CLS>] CAUSE [<ACA>]	<CACA>

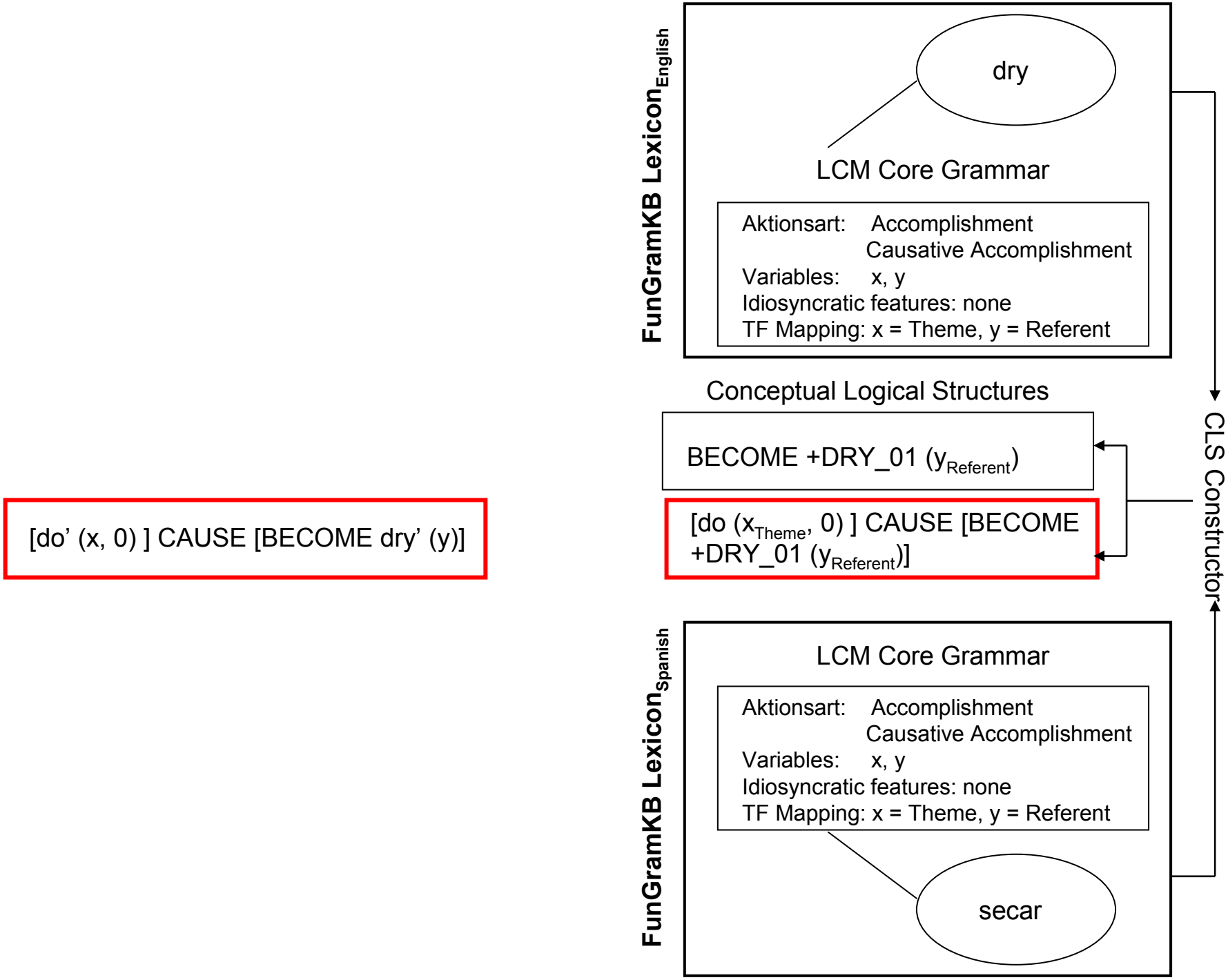
CLS Constructor

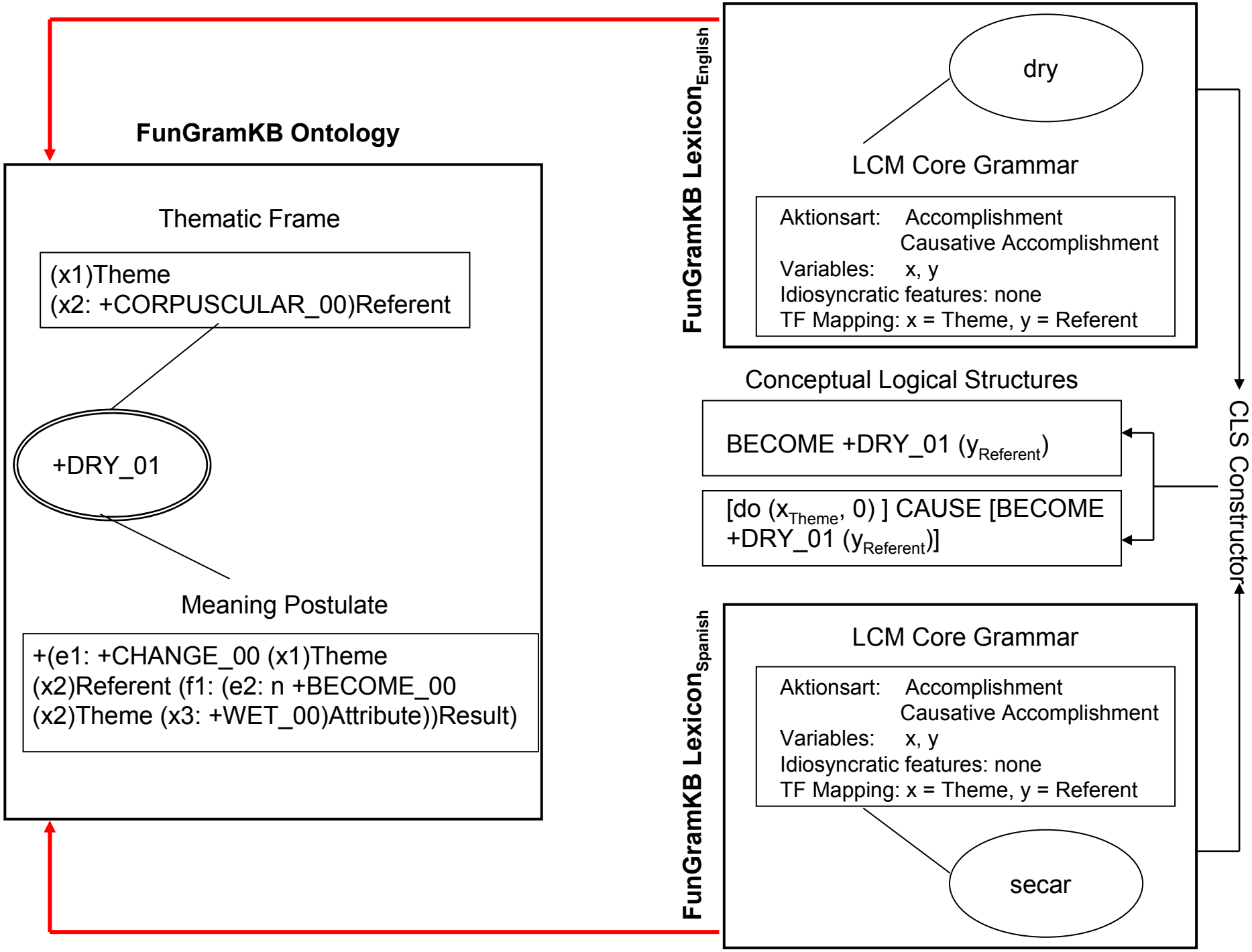
1. Expanding LS metavariables
   (e.g. <STA>, <ACT>, <ACC>, <ACH>, <SEM>, <ACA>)

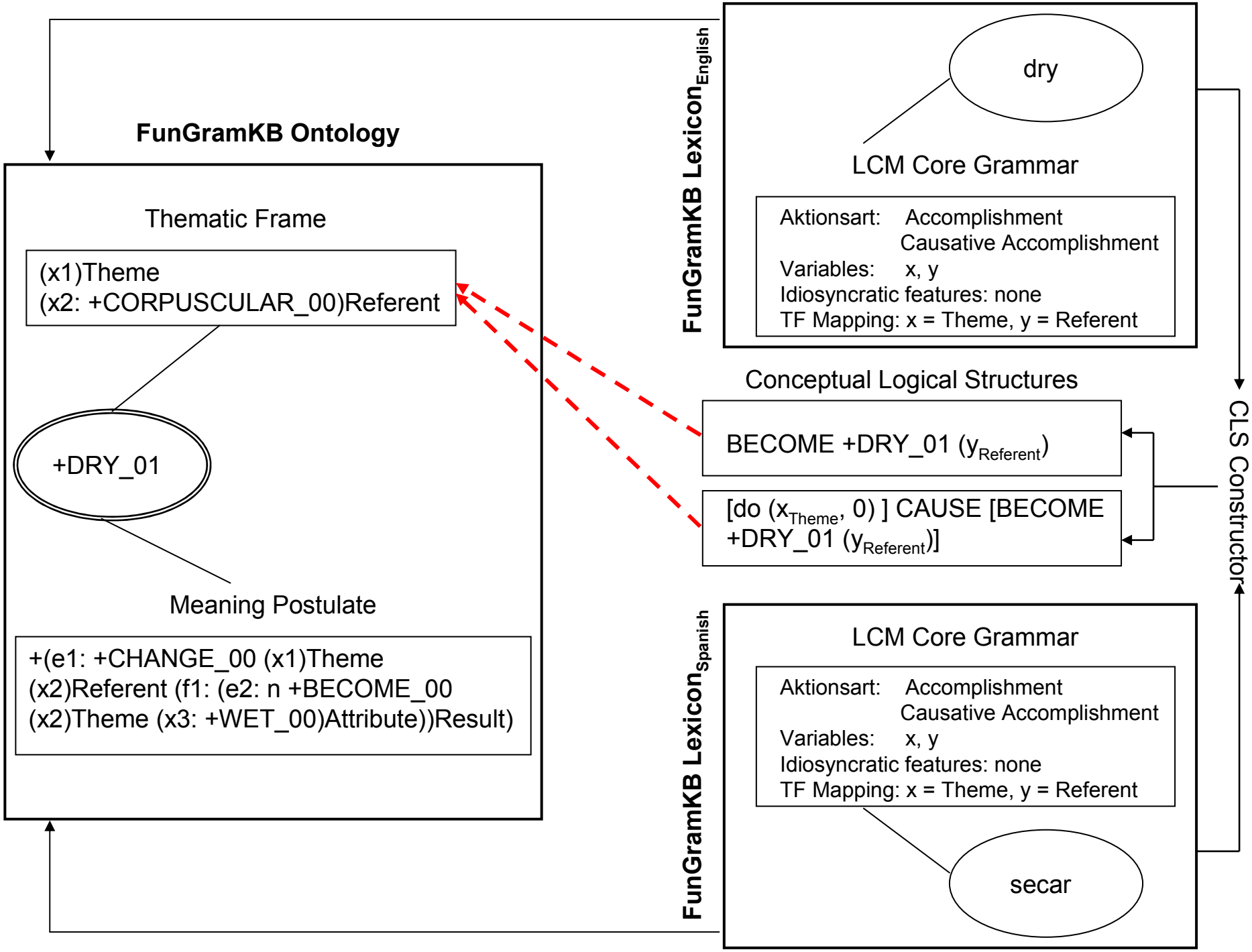
2. Inserting the concept
   (i.e. <C>)

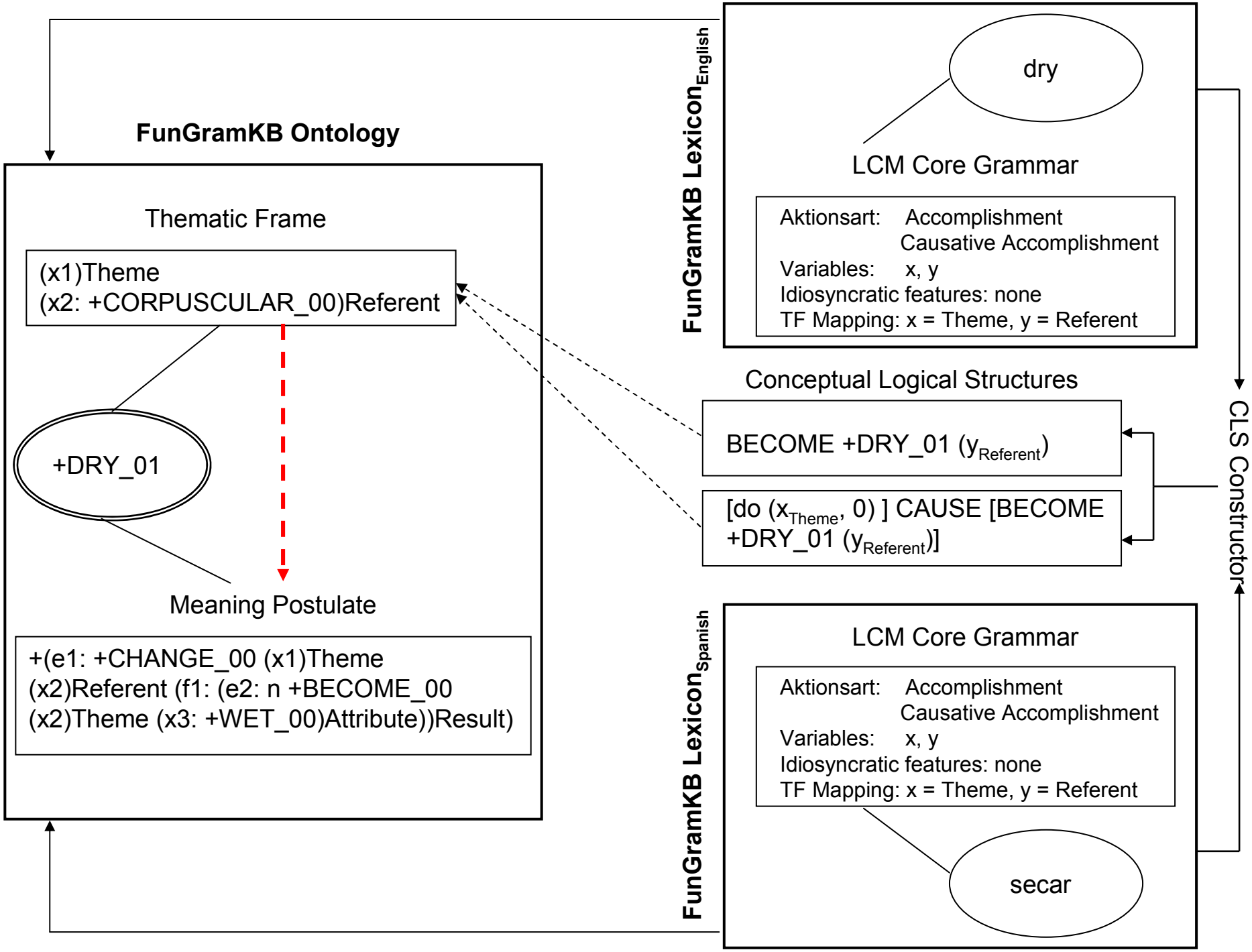
3. Inserting variables
   (i.e. <VAR>)

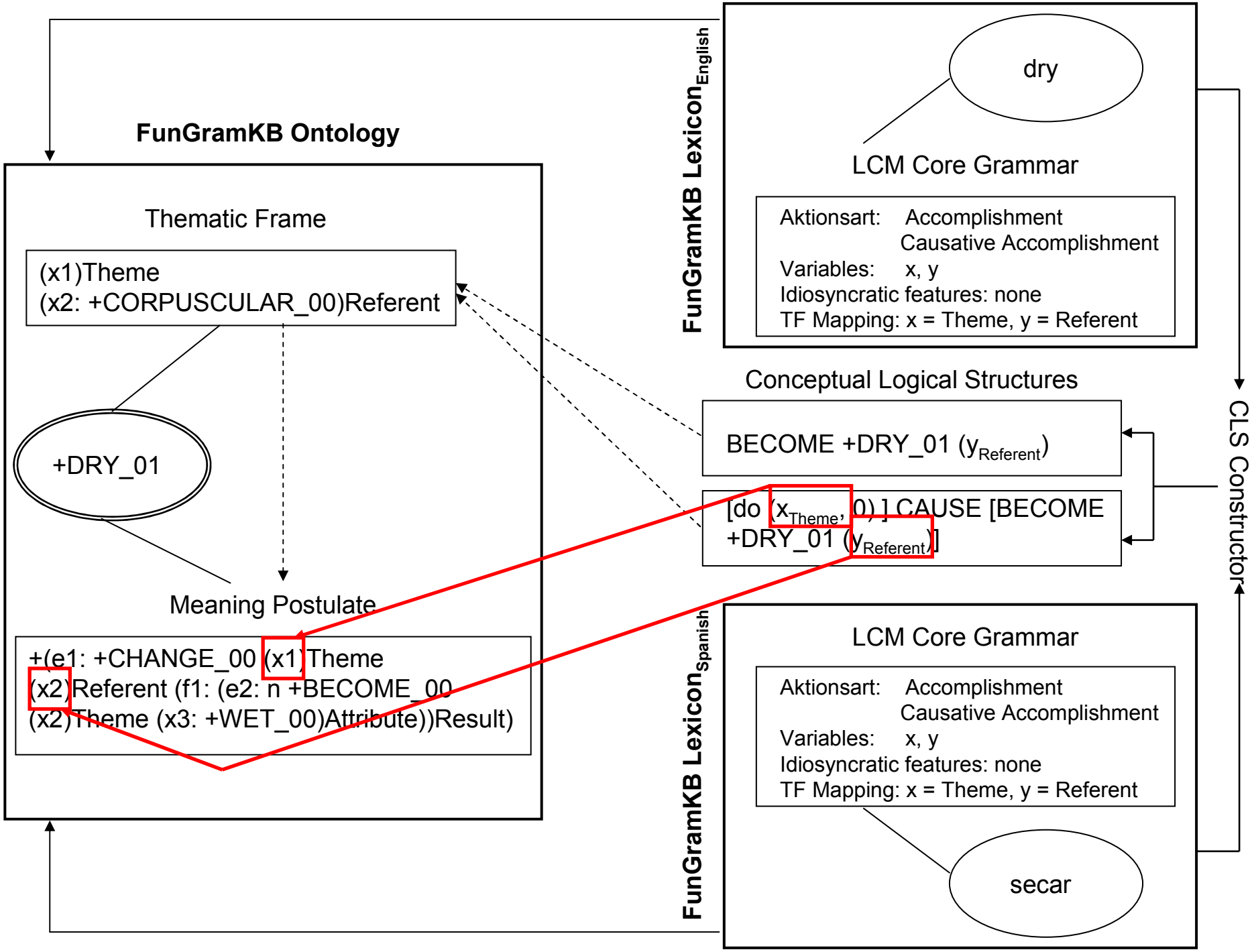
4. Inserting thematic roles


FunGramKB Lexicon English
dry
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
Conceptual Logical Structures
BECOME +DRY_01 (y Referent)
[do (x Theme, 0)] CAUSE [BECOME +DRY_01 (y Referent)]
[do' (x, 0)] CAUSE [BECOME dry' (y)]
CLS Constructor
FunGramKB Lexicon Spanish
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
secar


FunGramKB Ontology
Thematic Frame
(x1)Theme
(x2: +CORPUSCULAR_00)Referent
+DRY_01
Meaning Postulate
+(e1: +CHANGE_00 (x1)Theme
(x2)Referent (f1: (e2: n +BECOME_00
(x2)Theme (x3: +WET_00)Attribute))Result)
FunGramKB Lexicon English
dry
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
Conceptual Logical Structures
BECOME +DRY_01 (y Referent)
[do (x Theme, 0)] CAUSE [BECOME
+DRY_01 (y Referent)]
CLS Constructor
FunGramKB Lexicon Spanish
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
secar


FunGramKB Ontology
Thematic Frame
(x1)Theme
(x2: +CORPUSCULAR_00)Referent
+DRY_01
Meaning Postulate
+(e1: +CHANGE_00 (x1)Theme
(x2)Referent (f1: (e2: n +BECOME_00
(x2)Theme (x3: +WET_00)Attribute))Result)
FunGramKB Lexicon English
dry
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
Conceptual Logical Structures
BECOME +DRY_01 (y Referent)
[do (x Theme, 0)] CAUSE [BECOME
+DRY_01 (y Referent)]
CLS Constructor
FunGramKB Lexicon Spanish
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
secar


FunGramKB Ontology
Thematic Frame
(x1)Theme
(x2: +CORPUSCULAR_00)Referent
+DRY_01
Meaning Postulate
+(e1: +CHANGE_00 (x1)Theme
(x2)Referent (f1: (e2: n +BECOME_00
(x2)Theme (x3: +WET_00)Attribute))Result)
FunGramKB Lexicon English
dry
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
Conceptual Logical Structures
BECOME +DRY_01 (y Referent)
[do (x Theme, 0)] CAUSE [BECOME
+DRY_01 (y Referent)]
CLS Constructor
FunGramKB Lexicon Spanish
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
secar


FunGramKB Ontology
Thematic Frame
(x1)Theme
(x2: +CORPUSCULAR_00)Referent
+DRY_01
Meaning Postulate
+(e1: +CHANGE_00 (x1)Theme
(x2)Referent (f1: (e2: n +BECOME_00
(x2)Theme (x3: +WET_00)Attribute))Result)
FunGramKB Lexicon English
dry
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
Conceptual Logical Structures
BECOME +DRY_01 (y Referent)
[do (x Theme, 0)] CAUSE [BECOME
+DRY_01 (y Referent)]
CLS Constructor
FunGramKB Lexicon Spanish
LCM Core Grammar
Aktionsart: Accomplishment
Causative Accomplishment
Variables: x, y
Idiosyncratic features: none
TF Mapping: x = Theme, y = Referent
secar

Sentence-driven CLS

Betty asked Bill for an apple.

Meaning #1: To say a question

Meaning #2: To make a request

$<_{\text{IF DECL}}$ $<_{\text{TNS PAST}}$ <[do (%BETTY_00$_{\text{Theme}}$, [+REQUEST_01 (%BETTY_00$_{\text{Theme}}$, %BILL_00$_{\text{Goal}}$)])] PURP [do (%BILL_00$_{\text{Goal}}$, 0)] CAUSE [BECOME +REQUEST_01 (%BETTY_00$_{\text{Theme}}$, +APPLE_00$_{\text{Referent}}$)]>>>

The Word-Sense Disambiguation algorithm

```
IF the word is linked to more than one meaning THEN
    IF the word occurs in an idiom/higher-level construction THEN
        Translate the entire phrase with its idiomatic meaning
    ELSE
        Use morphosyntactic constraints
        IF only one meaning is left THEN
            Use the appropriate meaning
        ELSE
            Take into account selectional preferences of arguments
            IF only one meaning is left THEN
                Use the appropriate meaning
            ELSE
                IF the word has already occurred in the text THEN
                    Use the same meaning used in its previous occurrence
                ELSE
                    IF any of the meanings of the word belongs to the same
                    technical domain as the input text THEN
                        Use the appropriate meaning
                    ELSE
                        Apply a spreading activation method
                        IF there is a winning candidate THEN
                            Use the appropriate meaning
                        ELSE
                            Take the most frequent meaning
                        END IF
                    END IF
                END IF
            END IF
        END IF
    END IF
END IF
```

The Word-Sense Disambiguation algorithm

```
IF the word is linked to more than one meaning THEN
    IF the word occurs in an idiom/higher-level construction THEN
        Translate the entire phrase with its idiomatic meaning
    ELSE
        Use morphosyntactic constraints
        IF only one meaning is left THEN
            Use the appropriate meaning
        ELSE
            Take into account selectional preferences of arguments
            IF only one meaning is left THEN
                Use the appropriate meaning
            ELSE
                IF the word has already occurred in the text THEN
                    Use the same meaning used in its previous occurrence
                ELSE
                    IF any of the meanings of the word belongs to the same
                    technical domain as the input text THEN
                        Use the appropriate meaning
                    ELSE
                        Apply a spreading activation method
                        IF there is a winning candidate THEN
                            Use the appropriate meaning
                        ELSE
                            Take the most frequent meaning
                        END IF
                    END IF
                END IF
            END IF
        END IF
    END IF
END IF
```

LEXICON

The Word-Sense Disambiguation algorithm

```
IF the word is linked to more than one meaning THEN
    IF the word occurs in an idiom/higher-level construction THEN
        Translate the entire phrase with its idiomatic meaning
    ELSE
        Use morphosyntactic constraints
        IF only one meaning is left THEN
            Use the appropriate meaning
        ELSE
            Take into account selectional preferences of arguments
            IF only one meaning is left THEN
                Use the appropriate meaning
            ELSE
                IF the word has already occurred in the text THEN
                    Use the same meaning used in its previous occurrence
                ELSE
                    IF any of the meanings of the word belongs to the same
                    technical domain as the input text THEN
                        Use the appropriate meaning
                    ELSE
                        Apply a spreading activation method
                        IF there is a winning candidate THEN
                            Use the appropriate meaning
                        ELSE
                            Take the most frequent meaning
                        END IF
                    END IF
                END IF
            END IF
        END IF
    END IF
END IF
```

ONTOLOGY

Thank you!

E-mail: jcperinan@pdi.ucam.edu