Enhancing and Extending the Cardiff Model of SFL
A Generation-Oriented View

Víctor M. Castel
Universidad Nacional de Cuyo
Mendoza, Argentina

The Cardiff Symposium: Influences and Opportunities - Cardiff, Wales, UK, 2017
Outline

1 Introduction
   1.1 The Cardiff Grammar Generator (CGG)
   1.2 CGG constitutive text samples: MicroE and MiniE
   1.3 Location of the constitutive texts

2 Enhancements of CGG: Meaning onto form mappings
   2.1 Representing output instances
   2.2 A more delicate meaning-form interface for CGG output instances

3 Conclusions and future work on the enhancement of CGG

4 Extensions of CGG: Issues for the open and/or closing panel discussion
   4.1 Towards a grammar of Spanish
      Writing a Micro-Grammar of River Plate Spanish clitics
      System networks as logical form planners
      Realization rules as logical and linguistic forms builders
   4.2 Grammar writing
      Towards a grammar development environment for generation-oriented Cardiff grammars (GraDeR ©2014)
      A generation-oriented grammar of grammars
      Computer-aided grammar writing
Constitutive text sample of the Micro-Grammar of English (Fawcett 2004)

System Network Rules

sn2 : situation -> MOOD_1 & TRANSITIVITY & PERIOD_MARKING & INFORMATION_FOCUS.

sn3 : MOOD_1 -> 1# proposal_for_action (sp1) / 99# information (2).

sn16 : TRANSITIVITY -> 80# action ([sp3, 6]) / 20# relational (14) / 0# others.

sn23 : PERIOD_MARKING -> 90# simple_pd / 10# period_marked (17).

sn24 : INFORMATION_FOCUS -> 1# contrastive_newness / 99# no_contrastive_newness.

Same Pass Preference Re-Setting Rules

sp1 : proposal_for_action or attributive : for same_pass prefer sn23 <99% simple_pd & 1% period_marked>.

Realization Rules

2 : information : (if [seeker or confirmation_seeker or negative or contrast_on_polarity or validity_assessed or future_trp or being or affected_S_theme or retrospective or period_marked] then if giver then 0 @ 3, if [seeker or confirmation_seeker] then 0 @ 1), if [seeker or confirmation_seeker or negative or contrast_on_polarity] then apply do_support_subrule.

6 : action : M @ 7.

14 : relational : Ca by S, C @ 8, (if information then for Ca prefer thing, for Ca re_enter_at_entity).

17 : period_marked : (if information and not [validity_assessed or future_trp or retrospective] then PdX by 0, if present_trp then PdX < "is", if past_trp then PdX < "was"), (if [validity_assessed or future_trp or retrospective or proposal_for_action] then PdX @ 5, PdX < "be"), if affected_S_theme then PaX <+ "ing".

Back to Outline: MiniE
Constitutive text sample of the Mini-Grammar of English (Fawcett 2004)

System Network Rules

... 

sn2: MODE -> 70# spoken (0.1) / 30# written (0.2).

...

sn8: situation -> SITUATION_TYPE.

sn8_1: SITUATION_TYPE -> 100# congruent_situation (sp1_1, 1.2) / 0# reified_situation.

sn9: congruent_situation -> DEPENDENCE & TRANSITIVITY & TIME_POSITION_SPECIFICATION & CO_ORDINATION_OF_SITUATIONS & INFORMATION_FOCUS_SIT.

...

sn90: TRANSITIVITY -> 90# action / 10# relational / 0# mental / 0# environmental / 0# influential.

sn91: action -> 10# one_role_process / 90# two_role_process.

...

Same Pass Preference Re-Setting Rules

...

sp1_1: congruent_situation : (if written then for same_pass prefer sn12 <99.98% information & 0.02% proposal_for_action>, sn14 <99.9% giver & 0.1% seeker & 0% confirmation_seeker>).

...

sp1_3: proposal_for_action : for same_pass prefer sn90 <99.9% action & 0.1% relational>, sn50 <0.1% period_marked & 99.9% not_period_marked>.

...

Realization Rules

1.2 : congruent_situation : CI, CI places 250, S @ 33, [if spoken and not_coordinated_with_a_previous_situation and fills TE then St @ 3, St < "JJ"], if not at being then M @ 100, if information and (at being or unmarked_passive or future_trp or validity_assessed or retrospective_from_trp or period_marked or negative or (seeker and not ncs_theme_on_a_subject_theme_sought_r) or confirmation_seeker or contrastive_newness_on_polarity) then apply Operator_placement_subrule, if information and (negative or confirmation_seeker or (seeker and not ncs_theme_on_a_subject_theme_sought_r) or contrastive_newness_on_polarity) then apply dc_support_subrule, if simplex_situation or final_coordinated_situation then E @ 250, apply Ender_subrule), [if spoken and (simplex_situation or final_coordinated_situation)] then (if no_contrastive_newness_sit then MN @ 200, MN < "MT"), K @ 201).

...

1_sub : Ender_subrule : if spoken then E < "JJ", [if written then if unmarked_mood_wr then E < ".", if (seeker or confirmation_seeker or request) then E < "?", if (fun_mood_wr or enthusiastic_mood_wr) then E < "!"),

...
Architecture of the Cardiff Lexico-Grammar, and the enhancement of output instance representations

1. potential
   - system network of semantic features

2. instance
   - selection expression of semantic features

3. realization rules & potential structures

4. one layer of a richly labelled tree structure

Figure 2-3: The components and their outputs in a systemic functional grammar (Fawcett 2008).

(1a) \[ p \rightarrow q, \]
(1b) if \( p \), then \( q \),
(1c) if \( p \) is true, then \( q \) is true of the structure being built
The two existing handmade representations of the full semantic and syntactic analyses of simple clauses

Figure 10 from Fawcett (2000, Syntax...)

Figure 17-2 from Fawcett (2008, Invitation...)

Key

Σ = Text-Sentence; — = Filled With; Cl = Clause; | = Composed Of; / = Conflates With; S = Subject; Ag = Agent; Ph = Phenomenon; Δ = Expounded By; O = Operator; X = Auxiliary; M = Main Verb; C = Complement; Af = Affected; Perc = Perceiver; A = Adjunct; TP = Time Position
The function **Triggering** form functions

At an abstract level, realization rules are implications that can be represented, read, and interpreted as in (1), (11) or (111), respectively:

1. $p \rightarrow q$,
2. if $p$, then $q$,
3. if $p$ is true, then $q$ is true of the form structure being built,

where $p$ and $q$ are variables ranging over conditions and consequences, respectively. Condition $p$ can be a single semantic feature, a disjunction of semantic features, or a conjunction of semantic features. Consequence $q$ can be a (conjunction of) form function(s), and/or a (conjunct of) implication(s) like (11).

**Figure 1:** Function $F$ maps argument $A$ onto value $V$. If $F$ is a form function, then $A$ and $V$ are categories of form. If $F$ is the function **Triggering**, then $A$ is a configuration of semantic features, and $V$ is a form function.

**Figure 4:** The distinguished function **Triggering** maps $SF$ onto the form function $F$, which in turn, maps the argument $A$ onto the value $V$, where $SF$ is a (configuration of) semantic feature(s), and $A$ and $V$ are form categories.

**Examples of Meaning onto Form Mappings, i.e. Mappings of SF onto Filling, Component, Exponent, and Conflation:**

The rest of the presentation addresses these examples.

**Back to CG Architecture:** 4

**Start visualization of output instances**
Representation options for output instances

Sample text-sentences: *The dog rested.* and *Ike is kicking Victoria.*
Representing the selection expression structure underlying the text-sentence *The dog rested.*

Key

sth = subject-theme

cc = cultural classification

cr = creature

c = count

Back to dialog window, Handout: 7, 2
Representing the form structure of the text-sentence *The dog rested.*

A handmade standard tree diagram *vis à vis* an automatically generated graph

```
\Sigma
 |
Cl
 |
S/Ag M E
 |
\Delta \Delta
ngp rest+ed
 |
\Delta \Delta
dd h
the dog
```

**Key**

\(\Sigma = \text{TE} =\) Text-Element;  
\(- = \text{FdWt} =\) Filled With;  
\(\text{Cl} = \text{Clause};\)  
\(\mid = \text{CpOf} =\) Composed Of;  
\(S = \text{Subject};\)  
\(\slash = \text{CfWt} =\) Conflates With;  
\(\text{Ag} = \text{Agent}; \)  
\(O = \text{Operator}; \)  
\(\text{PdX} = \text{Period Auxiliary}; \)  
\(M = \text{Main Verb}; \)  
\(C = \text{Complement}; \)  
\(\text{Af} = \text{Affected}; \)  
\(E = \text{Ender};\)  
\(\Delta = \text{SfWt} =\) Suffixed With;  
\(\Delta = \text{ExBy} =\) Expounded By;  
\(\text{E} = \text{Ender};\)  
\(\text{ngp} = \text{Nominal Group}; \)  
\(\text{dd} = \text{Deictic Determiner}; \)  
\(h = \text{head}\)
Representing the full mapping of meaning structures onto form structures underlying the text-sentence *The dog rested.*

**Key**
- ReS = Reference to Situation
- CfWt = Conflates With = /
- TE = Text-Element = Σ
- FdWt = Filled With = —
- Cl = Clause
- CpOf = Composed Of = |
- S = Subject
- M = Main Verb
- E = Ender
- SfWt = Suffixed With = Δ
- ExBy = Expounded By = Δ
- ngp = Nominal Group
- dd = Deictic Determiner
- h = Head
Using interface pointers to understand the Triggering function in the generation of *The dog rested.*

0: See underlying RRs - H:12

1: Triggering component of *Cl* - H:13
2: Triggering conflation of *Ag* with *S* - H:14

3: Triggering exponence of *E* - H:15
4: Triggering filling of *S* with *ngp* - H:16

5: Triggering component of *ngp* and exponence of *dd* and *h* - H:17

**Key**

- *Cl* = Clause
- *Ag* = Agent
- *CfWt* = Conflates With = /
- *CpOf* = Composed Of = |  
- *S* = Subject
- *M* = Main Verb
- *E* = Ender
- *FdWt* = Filled With = —
- *SfWt* = Suffixed With = Δ
- *ExBy* = Expounded By = Δ
- *ngp* = Nominal Group
- *dd* = Deictic Determiner
- *h* = Head

Interface# = Node waiting for connection with a (configuration of) semantic feature(s) of the current selection expression structure.
Constitutive text of the realization rules underlying the generation of

*The dog rested.* Mini-Grammar of English (Fawcett 2004)

All passes

0.2 : written : for any_re_entry prefer written.

0.33 : consultative : for any_re_entry prefer consultative.

1st Pass: Creation of the element TE

1.1 : text_semantics : creates TE, [if ReS then ReS by TE, for ReS prefer situation, for ReS re_enter_at entity], [if ReT then ReT by TE, for ReT prefer thing, for ReT re_enter_at entity], [if ReMT then ReMT by TE, for ReMT prefer minor_relationship_with_thing, for ReMT re_enter_at entity].

2nd Pass: Called by ReS

1.2 : congruent_situation : Cl, Cl places 250, S @ 33, (if spoken and not_coordinated_with_a_previous_situation and fills TE then St @ 3, St < "JJ"), if not at being then M @ 100, if information and (at_being or unmarked_passive or future_trp or validity_assessed or retrospective_from_trp or past_from_trp or marked or unmarked or or (seeker and not ncs_theme_on_a_subject_theme_sought_r) or confirmation_seeker or contrastive_newness_on_polarity) then apply Operator_placement_subrule, if information and (negative or confirmation_seeker or (seeker and not ncs_theme_on_a_subject_theme_sought_r) or contrastive_newness_on_polarity) then apply do_support_subrule, (if simplex_situation or final_coordinated_situation then E @ 250, apply Ender_subrule), (if spoken and (simplex_situation or final_coordinated_situation) then (if no_contrastive_newness_st then MN @ 200, MN < "MT"), K @ 201).

Operator_placement_subrule : if giver or (seeker and ncs_theme_on_a_subject_theme_sought_r) then O @ 35, (if not at_being then M @ 100), if information and (at_being or unmarked_passive or future_trp or validity_assessed or retrospective_from_trp or past_from_trp or marked or unmarked or or (seeker and not ncs_theme_on_a_subject_theme_sought_r) or confirmation_seeker or contrastive_newness_on_polarity) then apply Operator_placement_subrule, if information and (negative or confirmation_seeker or (seeker and not ncs_theme_on_a_subject_theme_sought_r) or contrastive_newness_on_polarity) then apply do_support_subrule, (if simplex_situation or final_coordinated_situation then E @ 250, apply Ender_subrule), (if spoken and (simplex_situation or final_coordinated_situation) then (if no_contrastive_newness_st then MN @ 200, MN < "MT"), K @ 201).

do_support_subrule : if not (future_trp or validity_assessed or retrospective_from_trp or past_from_trp or marked or unmarked or or (seeker and not ncs_theme_on_a_subject_theme_sought_r) or confirmation_seeker or contrastive_newness_on_polarity) then apply finite_do_forms.

finite_do_forms : if present_trp and (simple_singular_sth or most_selected_thing_is_singular_sth or simple_mass_sth or most_selected_thing_is_mass_sth) then O < "does", if present_trp and (singular_performer_sth or singular_addressee_sth or simple_plural_interactant_sth or simple_plural_sth or most_selected_thing_is_plural_sth) then O < "do", if past_trp then O < "did".

Ender_subrule : if spoken then E < "JJ", (if written then if unmarked_mood_wr then E < ",", if (seeker or confirmation_seeker or request) then E < "?", if (fun_mood_wr or enthusiastic_mood_wr) then E < "!").

6.1 : agent_only : Ag by S, [if not proposal_for_action then if agent_only_unmarked then apply Ag_preferences_subrule, if agent_only_sought then apply Ag_sought_preferences_subrule, for Ag re_enter_at entity].

Ag_preferences_subrule : [if agent_subject_theme or agent_only_unmarked or (agent_unmarked and affected_covert) then apply subject_theme_subrule_Ag, else apply non_agent_theme_subrule_Ag].

subject_theme_subrule_Ag : [if interactant_sth then apply interactant_sth_subrule_Ag, else apply outsider_sth_subrule_Ag].

outsider_sth_subrule_Ag : if count_sth then apply count_sth_subrule_Ag, if mass_sth then apply mass_sth_subrule_Ag.

count_sth_subrule_Ag : for Ag prefer thing & BASIC_TYPICALLY_HUMAN_PREF_BLOCK & outsider & TYPICALLY_HUMAN_CC_PREF_BLOCK & sn159 <95% particularizedsingular & 5% unparticularizedsingular > & sn163 <95% particularizedplural & 5% unparticularizedplural >, if simple_singular_sth then for Ag prefer BASIC_SING_OUTSIDER_PREF_BLOCK, if simple_plural_sth then for Ag prefer BASIC_PL_OUTSIDER_PREF_BLOCK, if (singular_unselected_from_sth or plural_unselected_from_sth) then for Ag prefer NOT.Selected FROM_PREF_BLOCK, if singular_selected_from_sth then apply singular_selected_from_sth_subrule_Ag, if plural_selected_from_sth then apply plural_selected_from_sth_subrule_Ag.

3rd Pass: Called by Ag

60 : congruent_thing : ngp, ngp places 100, h @ 57.

65 : recoverable_cc : dd @ 27, dd < "the".

73.33 : dog_c : h < "dog".

Back to interface pointers, Handout: 11, 1
RR#1.2: Triggering componence operations $S@33, M@100, E@250$
RR#6.1: Triggering conflation operation $Ag$ by $S$
Ender subrule: Triggering exponence operation $E < "\cdot"$
RR#60: Triggering filling operation \textit{ngp}
RRs#60, 65, and 73.33: Triggering componence operations $dd \at 27$ and $h \at 57$, and exponence operations $dd < "the"$ and $h < "dog"$
Conclusions and future work on the enhancement of CGG

1. Formalization of the Triggering function to capture the explicit descriptive delicacy of the form potential

2. Implementation of the Triggering function so that output instances reflect the explicit descriptive delicacy of the form potential

3. No output instances yet reflecting the explicit delicacy of the lower pass preference potential
THANK YOU FOR YOUR PATIENCE!
Representing the selection expression structure underlying the text-sentence *Ike is kicking Victoria.*
Representing the form structure underlying the text-sentence *Ike is kicking Victoria.*

A handmade standard tree diagram vis-à-vis an automatically generated graph

\[
\begin{array}{c}
\Sigma \\
- \\
\text{Cl} \\
| \\
\text{S/Ag} & \text{O/Pdx} & \text{M} & \text{C/Af} & \text{E} \\
- & \Delta & \Delta & - & \Delta \\
\text{ngp} & \text{is} & \text{kicking} & \text{ngp} \\
| & | \\
\text{h} & \text{h} \\
\Delta & \Delta \\
\text{Ike} & \text{Victoria}
\end{array}
\]

**Key**

\(\Sigma = \text{TE} = \text{Text-Element} ; \quad - = \text{FdWt} = \text{Filled With} ; \quad \text{Cl} = \text{Clause} ; \quad | = \text{CpOf} = \text{Composed Of} ; \quad \text{S} = \text{Subject} ; \quad / = \text{CfWt} = \text{Conflates With} ;

\text{Ag} = \text{Agent} ; \quad \text{O} = \text{Operator} ; \quad \text{Pdx} = \text{Period Auxiliary} ; \quad \text{M} = \text{Main Verb} ; \quad \text{C} = \text{Complement} ; \quad \text{Af} = \text{Affected} ; \quad \text{E} = \text{Ender} ; \quad \Delta = \text{SfWt} = \text{Suffixed With} ;

\Delta = \text{ExBy} = \text{Expounded By} ; \quad \text{E} = \text{Ender} ; \quad \text{ngp} = \text{Nominal Group} ; \quad \text{h} = \text{head}

*Back to dialog window, Handout: 7, 3'*
Representing the full Mapping of meaning structures onto form structures underlying the text-sentence *Ike is kicking Victoria.*
Using interface pointers to understand the Triggering function in the generation of *Ike is kicking Victoria.*

0: See underlying RRs - H:22

1: See conflation of PdX - H:23

2: See exponence of PdX - H:24

3: See filling of S - H:25
Constitutive text of the realization rules underlying the generation of

*Ike is kicking Victoria*. Micro-Grammar of English (Fawcett 2003)

1st Pass: Creation of the element TE

1.2: text_semantics:: creates TE, if ReS then ReS by TE, for ReS prefer situation, for ReS enter_at entity, if ReT then ReT by TE, for ReT prefer thing, for ReT enter_at entity, if ReMT then ReMT by TE, for ReMT prefer minor_relationship_with_thing, for ReMT enter_at entity.

2nd Pass: Called by ReS

1.3: situation:: Cl, Cl places 9, S @ 2, E @ 9, giver then E < ".", if seeker or confirmation_seeker then E < "?", if directive then E < "!".

2: information:: (if (seeker or confirmation_seeker or negative or contrast_on_polarity or validity_assessed or future_trp or being or affected_S_theme or retrospective or period_marked) then if giver then O @ 3, if (seeker or confirmation_seeker) then 0 @ 1), if (seeker or confirmation_seeker or negative or contrast_on_polarity) then apply do_support_subrule.

6: action:: M @ 7.

7: kicking:: M < "kick", apply regular_vb_subrule.

11: agent_S_theme:: Ag by S, if information then for Ag prefer thing, for Ag enter_at entity, C @ 8, Af by C, for Af prefer thing, for Af enter_at entity.

17: period_marked:: (if information and not (validity_assessed or future_trp or retrospective) then PdX by O, if present_trp then PdX < "is", if past_trp then PdX < "was"), (if (validity_assessed or future_trp or retrospective or proposal_for_action) then PdX @ 5, PdX < "be"), if affected_S_theme then PaX <+ "ing".

do_support_subrule:: (if information and not (validity_assessed or future_trp or retrospective or period_marked or affected_S_theme or being) then if present_trp then 0 < "does", if past_trp then 0 < "did"), (if directive and (negative or contrast_on_polarity) then 0 @ 1, 0 < "do").

regular_vb_subrule:: if giver and not validity_assessed or future_trp or negative or contrast_on_polarity or retrospective or period_marked or affected_S_theme or being) then if present_trp then (if kicking then M <+ "s"), (if kissing or touching or washing then M <+ "es"), if past_trp and validity_unassessed then M <+ "ed"), if affected_S_theme then M <+ "ed", if (period_marked and not affected_S_theme) then M <+ "ing", if (retrospective or (past_trp and validity_assessed)) and not (period_marked or affected_S_theme) then M <+ "ed".

3rd Pass: Called by Ag

24: thing:: ngp, ngp places 1, h @ 1.

25: ingroupness:: if ike then h < "Ike", if ivy then h < "Ivy", if tony then h < "Tony", if george then h < "George", if david then h < "David", if victoria then h < "Victoria".

4th Pass: Called by Af

24: thing:: ngp, ngp places 1, h @ 1.

25: ingroupness:: if ike then h < "Ike", if ivy then h < "Ivy", if tony then h < "Tony", if george then h < "George", if david then h < "David", if victoria then h < "Victoria".
RR#17: Triggering the conflation operation $PdX$ by $O$
RR#17: Triggering the exponence operation $PdX < \text{"is"}$

See details in RR#17, Handout: 22  
Go back to interface pointers, Handout: 21, 3
RR#24: Triggering the filling operation \textit{ngp}
Outline

1 Introduction

2 Towards a grammar of Spanish from the CG view
   2.1 The data
   2.2 The architecture
   2.3 Writing a Micro-Grammar of Spanish clitics
      2.3.1 System networks as logical form planners
      2.3.2 Realization rules as logical and linguistic forms builders

3 Conclusions and future work
The Standard Cardiff Grammar Architecture vis-à-vis a Modified Architecture of a Grammar of Spanish

Figure 2-3: The components and their outputs in a systemic functional grammar (Fawcett 2008).

The System Network of a Micro-Grammar of Spanish
Initial Stage Logical and Linguistic Forms

Realization Rule #1

Key
FdWt = Filled With
ccc = current commanding category
LogF = Logical Form
Ev = Event (variable)
LinF = Linguistic Form
CpOf = Composed Of
M = Main Verb Root
Vnd = Verb Ending

Ordinary prose definitions

Insuing output sub-graph

Back to System Network: 3
A Logical Form for *regalar* (= 'give as a gift')

Realization Rule #3

**Key**

- regalar_m = equivalent to regalar'
- PRED = PREDICATE
- Ev = Event (variable)
- Ob = Object (variable)
- SUJ = SUBJECT
- NON_SUJ = NON_SUBJECT
Logical and Linguistic Forms for Verb Endings and Clitics

Realization Rule #5
Output Graph for the Text-Sentence *te las regalamos*

Realization Rules #1, 3, 5

You them give+as+a+gift+first-person+plural+present-trp
we give to you as a gift
Conclusions
Initial Logical and Linguistic Forms for a Congruent Semantics

Ordinary Language Paraphrases of Realization Rule #1

Key
FdWt = Filled With
ccc = current commanding category
LogF = Logical Form
Ev = Event (variable)
LinF = Linguistic Form
CpOf = Composed Of
M = Main Verb Root
Vnd = Verb Ending

Ordinary language paraphrase I

If you want to realize a congruent semantics, then

[i] the current commanding element, namely, a text-sentence, must be filled with a sign;
[ii] the sign’s logical form is an event;
[iii] the event is congruent;
[iv] the sign’s linguistic form is a clause;
[v] the clause has ten ten potential places for form elements to land on;
[vi] the fifth landing site is to be occupied by a main verb root; and
[vii] the sixth landing site is to be occupied by a verbal ending.

Ordinary language paraphrase II

For a text-sentence to express a congruent semantics, it must result from a sign with a congruent type event, and a clause with ten landing sites for clause elements. The fifth and sixth clause landing sites will be occupied by a main verb root and a verbal ending, respectively.
Mapping a Congruent Event onto a Verb Root and its Ending

Output Sub-Graph of Realization Rule #1

Key
FdWt = Filled With
LogF = Logical Form
Ev = Event (variable)
LinF = Linguistic Form
Cl = Clause
CpOf = Composed Of
M = Main Verb Root
Vnd = Verb Ending
A Logical Form for *regalar* (= 'give as a gift')

Ordinary Language Paraphrases of Realization Rule #3

**Key**
- `regalar_m` = equivalent to `regalar'`
- `PRED` = PREDICATE
- `Ev` = Event (variable)
- `Ob` = Object (variable)
- `S U J` = SUBJECT
- `NO N_S U J` = NON_SUBJECT

**Ordinary language paraphrase I**
The logical form of the process `regalar_m` is built with:

(i) the feature `regalar_m` as the core predicate of the event `Ev`;
(ii) the variable `Ob1` as the Agent of `Ev`;
(iii) the variable `Ob2` as the Affected-Possessed of `Ev`;
(iv) the variable `Ob3` as the Affected-Carrier of `Ev`; furthermore, if the agent participant [agent *sth*] of `regalar_m` is to be the subject of the event `Ev`, then add to the sub-graph built in steps (i)-(iv) the following sub-structures:
(v) the variable of the Agent participant of `Ev` is also its SUBJECT;
(vi) the variable of the Affected-Possessed of `Ev` is also its NON_SUBJECT;
(vii) the variable of the Affected-Carrier participant of `Ev` is also its NON_SUBJECT.

**Ordinary language paraphrase II**
The semantics of the process `regalar` is basically a predicate with three participant roles: an agent, an affected-possessed, and an affected-carrier. The agent participant of the process is also its subject, while the affected-possessed and the affected-carrier participants are both non-subjects.
Logical Form for Participant Role Variables of *regalar_m*

Output Sub-graph of Realization Rule #3

**Key**
- FdWt = Filled With
- LogF = Logical Form
- Ev = Event (variable)
- PRED = PREDICATE
- Ob = Object (variable)
- SUJ = SUBJECT
- NON_SUJ = NON_SUBJECT
- regalar_m = equivalent to regalar'
Logical Form of Participant Role Variables of the Predicate *regalar_m*

Output Sub-graph of Realization Rule #5

Key
Ev = Event (variable)
PRED = PREDICATE
Ob = Object (variable)
SUJ = SUBJECT
NON_SUJ = NON_SUBJECT
*regalar_m* = equivalent to regalar'

Back to RR#5: H38
Explicit andImplicit meanings of the Verb Ending *amos*

Output Graph of Realization Rule #5

---

**Key**
- Ev = Event (variable)
- Ob = Object (variable)
- SUJ = SUBJECT
- Vnd = Verb Ending
- ExBy = Expounded By

- = conjunction of explicit meanings
- = implicit meanings

---

Back RR#5: H39
Explicit and Implicit meanings of the Clitic *te*

Output Graph of Realization Rule #5

Key

- Ev = Event (variable)
- Ob = Object (variable)
- NON_SUJ = NON_SUBJECT
- Clt2 = Clitic Second
- ExBy = Expounded By

- = conjunction of explicit meanings
- = implicit meanings
Explicit and Implicit Meanings of the Clitic *las*

Output Graph of Realization Rule #5

Key:
- **Ev** = Event (variable)
- **Ob** = Object (variable)
- **NON_SUJ** = NON_SUBJECT
- **CltA** = Clitic Accusative
- **ExBy** = Expounded By
- = conjunction of explicit meanings
- = implicit meanings