Semantic Theory
Week 7 – Discourse Representation Theory

Noortje Venhuizen
Harm Brouwer

Universität des Saarlandes

Summer 2020

DRS Syntax

A discourse representation structure (DRS) K is a pair $\langle U_K, C_K \rangle$, where:

- $U_K \subseteq U_D$ and U_D is a set of discourse referents, and
- C_K is a set of well-formed DRS conditions

Well-formed DRS conditions:

• $R(u_1, \ldots, u_n)$	*where:*	R is an n-place relation, $u_i \in U_D$
• $u = v$		$u, v \in U_D$
• $u = a$		$u \in U_D$, a is a constant
• $\neg K_1$		K_1 is a DRS
• $K_1 \Rightarrow K_2$		K_1 and K_2 are DRSs
• $K_1 \vee K_2$		K_1 and K_2 are DRSs

Anaphora and accessibility

Mary knows a professor. If she owns a book, he reads it. ?It fascinates him.

w u s t

w = Mary

professor(u)

knows(w, u)

x y		z v
x = w book(y) owns(x, y)	⇒	reads(z, v) z = u v = y

fascinates(s, t)

t = u

s = **???**

Non-accessible discourse referents

Cases of non-accessibility:

(1) *If a professor owns a book, he reads it. It has 300 pages.*

(2) *It is not the case that a professor owns a book. He reads it.*

(3) *Every professor owns a book. He reads it.*

(4) *If every professor owns a book, he reads it.*

(5) *Peter owns a book, or Mary reads it.*

(6) *Peter reads a book, or Mary reads a newspaper article. It is interesting.*

Accessible discourse referents

The following discourse referents are accessible for a condition:

- DRs in the same local DRS
- DRs in a superordinate DRS
- DRs in the universe of an antecedent DRS, if the condition occurs in the consequent DRS.

We need a formal notion of DRS subordination

Subordination

A DRS K_1 is an immediate sub-DRS of a DRS $K = \langle U_K, C_K \rangle$ iff C_K contains a condition of the form

- $\neg K_1$, $K_1 \Rightarrow K_2$, $K_2 \Rightarrow K_1$, $K_1 \vee K_2$ or $K_2 \vee K_1$.

K_1 is a sub-DRS of K (notation: $K_1 \leq K$) iff

- $K_1 = K$, or
- K_1 is an immediate sub-DRS of K, or
- there is a DRS K_2 such that $K_1 \leq K_2$ and K_2 is an immediate sub-DRS of K (i.e. reflexive, transitive closure)

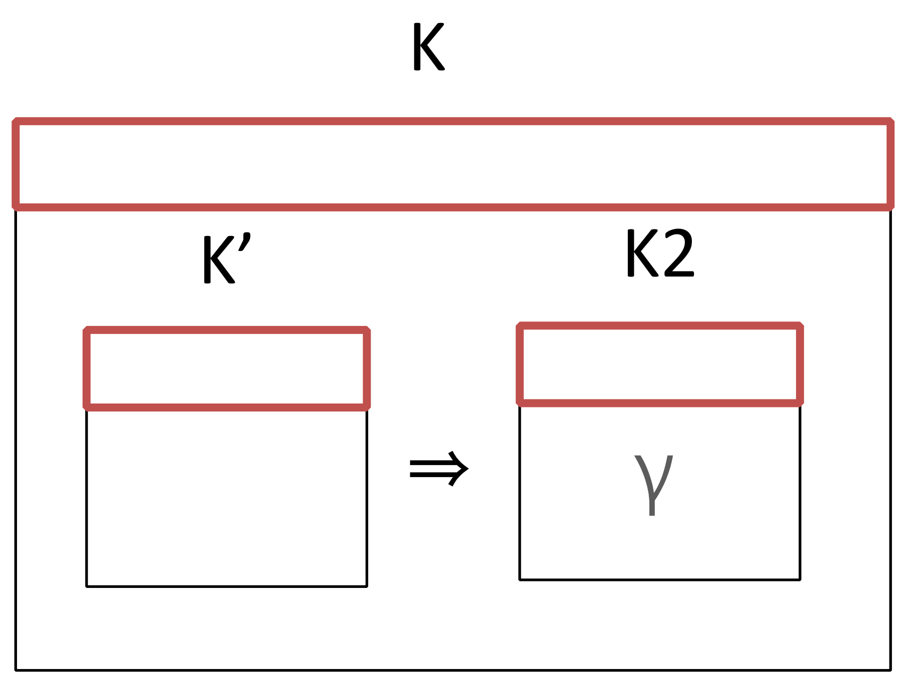
K_1 is a proper sub-DRS of K iff $K_1 \leq K$ and $K_1 \neq K$.

Accessibility

Let K, K_1, K_2 be DRSs such that $K_1, K_2 \leq K$, $x \in U_{K_1}$, $\gamma \in C_{K_2}$

x is accessible from γ in K iff

- $K_2 \leq K_1$ or
- there are K_3, $K_4 \leq K$ such that $K_1 \Rightarrow K_3 \in C_{K4}$ and $K_2 \leq K_3$



Free and bound variables in DRT

A DRS variable x, introduced in the conditions of DRS K_i, is bound in global DRS K iff there exists a DRS $K_j \leq K$, such that:

(i) $x \in U(K_j)$, and

(ii) K_j is accessible for K_i in K

Properness: A DRS is *proper* iff it does not contain any free variables

Purity: A DRS is *pure* iff it does not contain any *otiose declarations* of variables

$x \in U(K_1)$ and $x \in U(K_2)$ and $K_1 \leq K_2$

From text to DRS

Text	$\Sigma = \langle\ S_1,$	$S_2,$	$\ldots,$	$S_n\ \rangle$
	$\downarrow$	$\downarrow$		$\downarrow$
Syntactic Analysis	$P(S_1)$	$P(S_2)$	$\ldots$	$P(S_n)$
	$\downarrow$	$\downarrow$		$\downarrow$
DRS Construction	$K_1 \longrightarrow$	$K_2 \longrightarrow$	$\ldots \longrightarrow$	K_n

DRS Construction Algorithm

Let the following be a well-formed, *reducible* DRS condition:

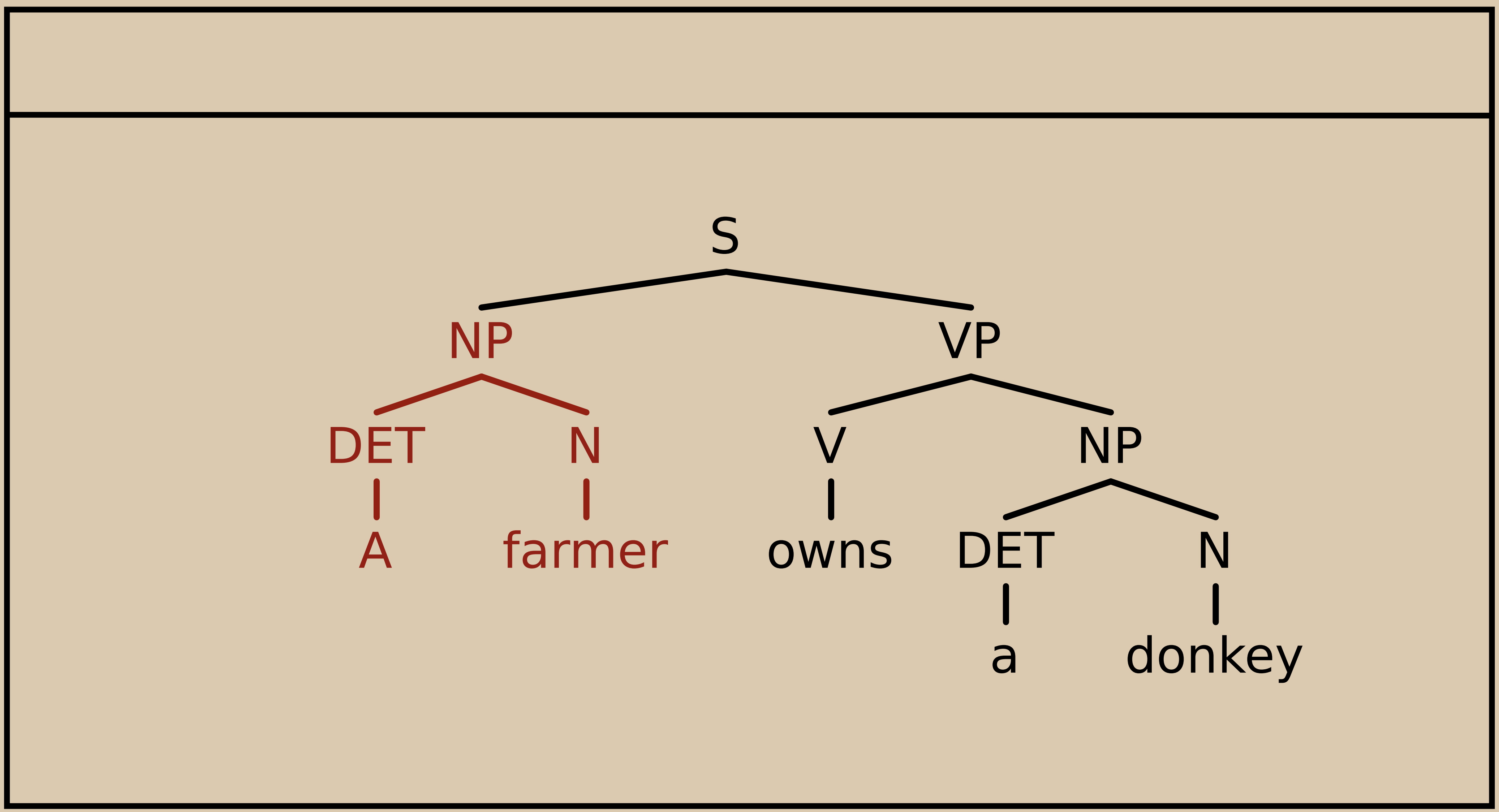
- Conditions of form α or α(x1, …, xn), where α is a context-free parse tree.

DRS construction algorithm:

- Given a text $\Sigma = \langle S_1, \ldots, S_n \rangle$, and a DRS K_0 (= $\langle \varnothing, \varnothing \rangle$, by default)
- Repeat for i = 1, …, n:
  - Add parse tree $P(S_i)$ to the conditions of K_{i-1}.
  - Apply DRS construction rules to reducible conditions of K_{i-1}, until no reduction steps are possible any more.
  - The resulting DRS K_i is the discourse representation of text $\langle S_1,\ldots,S_i \rangle$.

DRS Construction Example

- A farmer owns a donkey. He beats it.

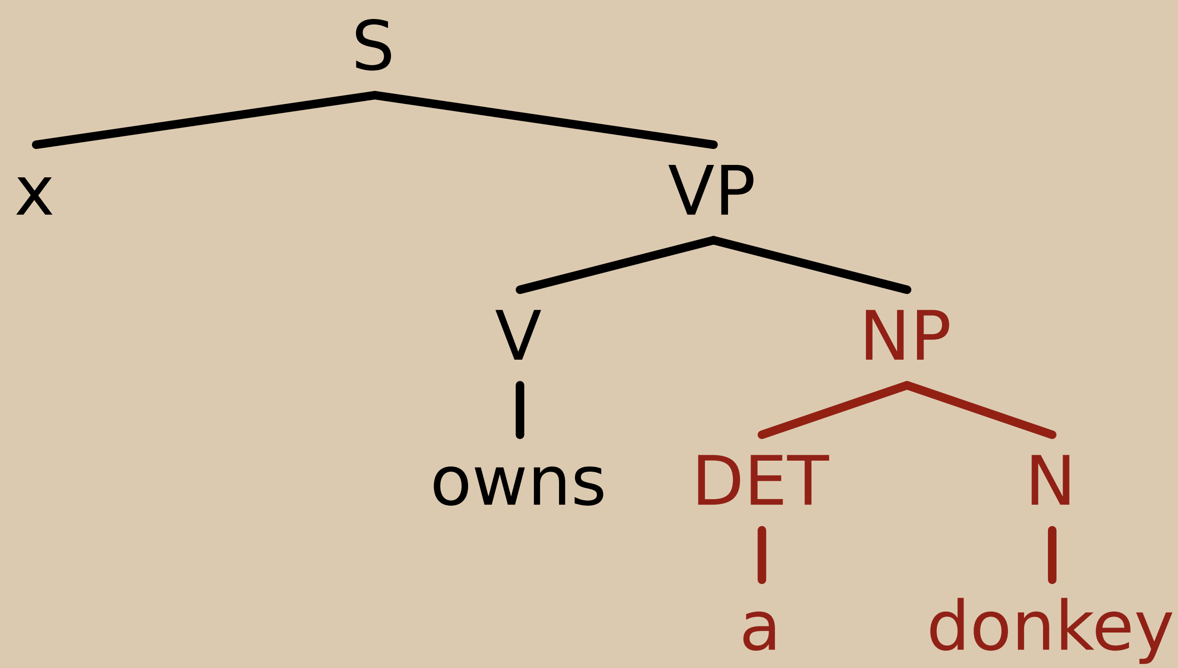


DRS Construction Example

- A farmer owns a donkey. He beats it.

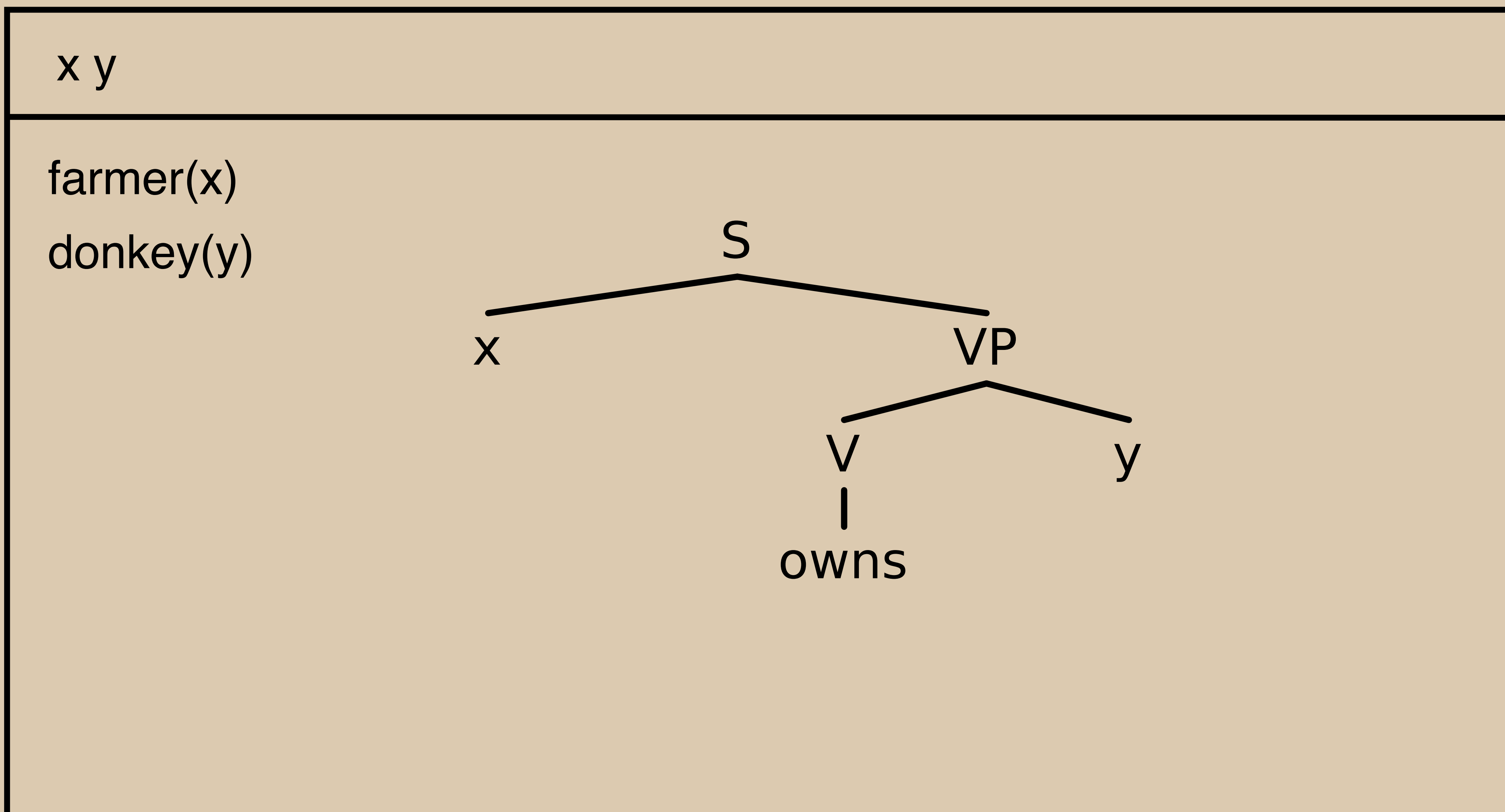
x

farmer(x)



DRS Construction Example

- A farmer owns a donkey. He beats it.



DRS Construction Example

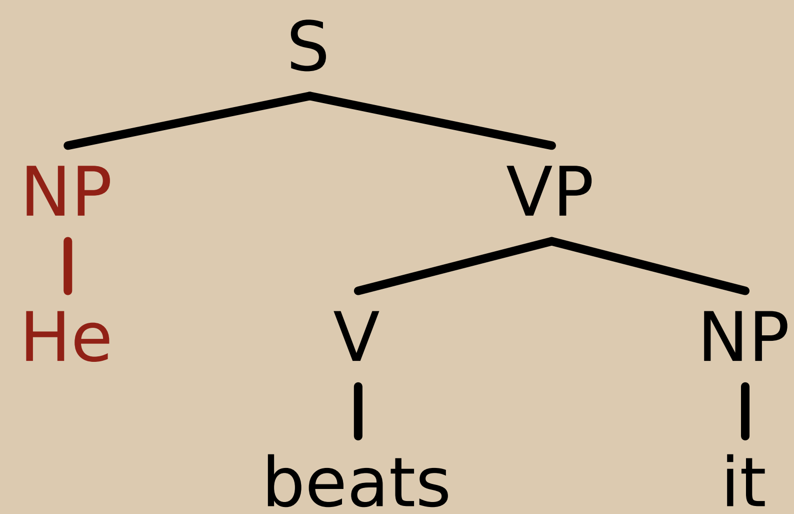
- A farmer owns a donkey. He beats it.

x y
farmer(x) donkey(y) owns(x, y)

DRS Construction Example

- A farmer owns a donkey. He beats it.

x y
farmer(x) donkey(y) owns(x, y)

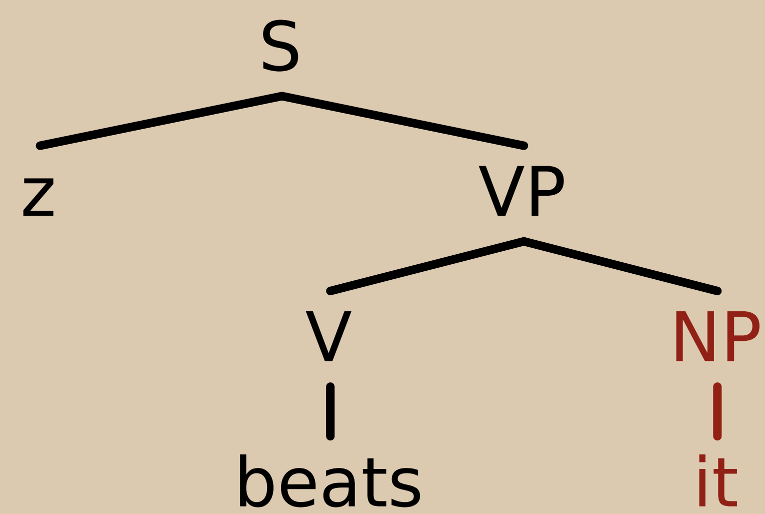


DRS Construction Example

- A farmer owns a donkey. He beats it.

x y z

farmer(x)
donkey(y)
owns(x, y)
z = x



DRS Construction Example

- A farmer owns a donkey. He beats it.

x y z u

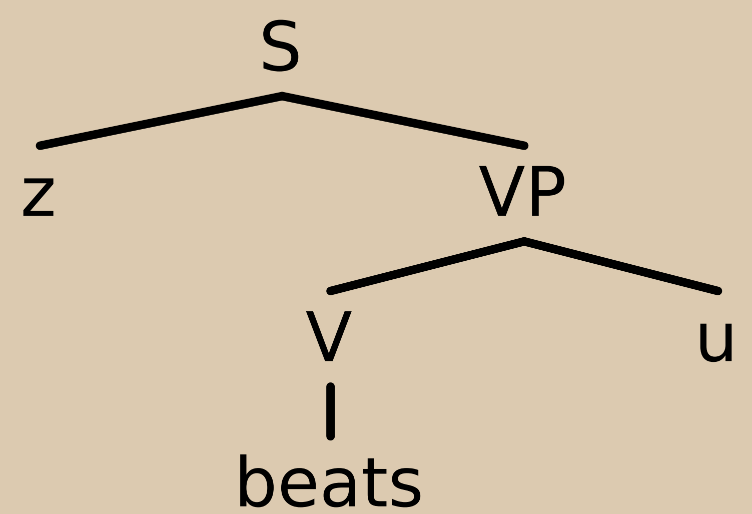
farmer(x)

donkey(y)

owns(x, y)

z = x

u = y



DRS Construction Example

- A farmer owns a donkey. He beats it.

x y z u
farmer(x) donkey(y) owns(x, y) z = x u = y beat(z, u)

Construction Rules: Examples

Indefinite NPs

- **Trigger:** a reducible condition α in DRS K that has a substructure [NP β], such that β is εδ, where ε is an indefinite article
- **Action:** Add new DR x to U_K; Replace β in α by x; Add δ(x) to C_K

Personal Pronouns

- **Trigger:** a global DRS K*, and some K ≤ K*, with a reducible condition α in K that has substructure [NP β], such that β is a personal pronoun
- **Action:** Add a new DR x to U_K; Replace β in α by x; Select an appropriate DR y that is accessible from α in K*; Add x = y to C_K

A constraint on DRS construction

Problem: The basic DRS construction algorithm can derive DRSs for both of the following sentences, with the indicated anaphoric binding:

(1) [A professor]$_i$ recommends a book that she$_i$ likes

(2) She$_i$ recommends a book that [a professor]$_i$ likes

Solution: If two different triggering configurations occur in a reducible condition, then first apply the construction rule to the highest triggering configuration.

- *The highest triggering configuration* is the one whose top node dominates the top nodes of all other triggering configurations.

From text to DRS to models

Text	$\Sigma = \langle\ S_1,$	$S_2,$	$\ldots,$	$S_n\ \rangle$
	$\downarrow$	$\downarrow$		$\downarrow$
Syntactic Analysis	$P(S_1)$	$P(S_2)$	$\ldots$	$P(S_n)$
	$\downarrow$	$\downarrow$		$\downarrow$
DRS Construction	$K_1 \longrightarrow$	$K_2 \longrightarrow$	$\ldots \longrightarrow$	K_n
				$\downarrow$

Interpretation by
model embedding:
Truth-conditions of Σ

DRS Interpretation

Given a DRS $K = \langle U_K, C_K \rangle$, with $U_K \subseteq U_D$

Let $M = \langle U_M, V_M \rangle$ be a FOL model structure appropriate for K, i.e. a model structure that provides interpretations for all predicates and relations occurring in K

DRS K is *true* in model M *iff*

- there is an embedding function for K in M which verifies all conditions in K

… where: an embedding of K into M is a (partial) function **f** from U_D to U_M such that $U_K \subseteq \mathrm{Dom}(\mathbf{f})$.

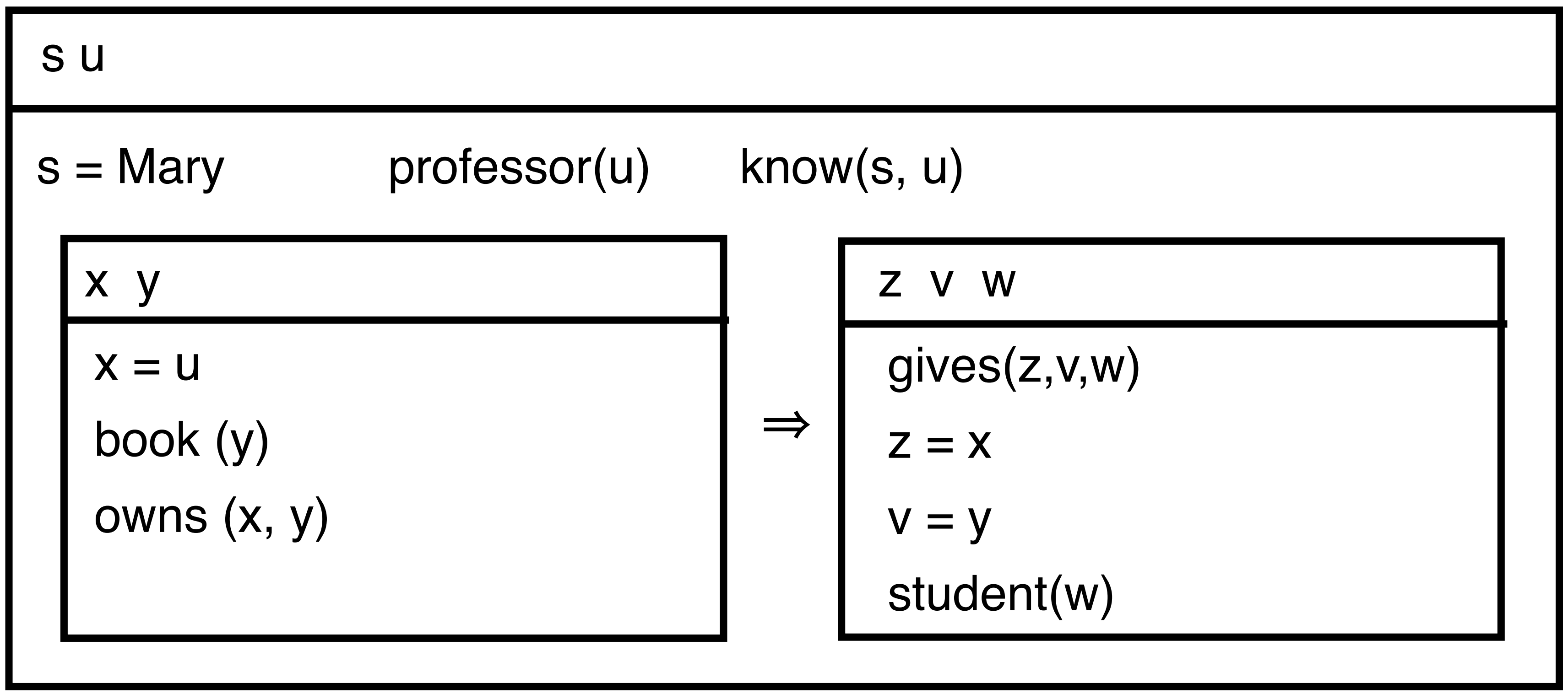
Verifying embedding

An embedding **f** of K in M verifies K in M (**f** $\vDash_M$ K) iff **f** verifies every condition $\alpha \in C_K$

- **f** $\vDash_M R(x_1, \ldots, x_n)$ iff $\langle \mathbf{f}(x_1), \ldots, \mathbf{f}(x_n) \rangle \in V_M(R)$
- **f** $\vDash_M x = y$ iff $\mathbf{f}(x) = \mathbf{f}(y)$
- **f** $\vDash_M x = a$ iff $\mathbf{f}(x) = V_M(a)$
- **f** $\vDash_M \neg K_1$ iff there is no **g** $\supseteq_{U_{K1}}$ **f** such that g $\vDash_M K_1$
- **f** $\vDash_M K_1 \Rightarrow K_2$ iff for all **g** $\supseteq_{U_{K1}}$ **f** such that **g** $\vDash_M K_1$
  there is a **h** $\supseteq_{U_{K2}}$ **g** such that **h** $\vDash_M K_2$
- **f** $\vDash_M K_1 \vee K_2$ iff there is a $\mathbf{g_1} \supseteq_{U_{K1}}$ **f** such that $\mathbf{g_1} \vDash_M K_1$
  or there is a $\mathbf{g_2} \supseteq_{U_{K2}}$ **f** such that $\mathbf{g_2} \vDash_M K_2$

Verifying embedding: example

Mary knows a professor. If he owns a book, he gives it to a student.



…is **true** in $M = \langle U_M, V_M \rangle$ *iff* there is an $\mathbf{f} :: U_D \to U_M$, (with $\{s,u\} \subseteq \mathrm{Dom}(\mathbf{f})$) such that:

$\mathbf{f}(s) = V_M(\text{Mary})$ & $\mathbf{f}(u) \in V_M(\text{prof'})$ & $\langle \mathbf{f}(s), \mathbf{f}(u) \rangle \in V_M(\text{know})$,

and for all $\mathbf{g} \supseteq_{\{x,y\}} \mathbf{f}$ s.t. $\mathbf{g}(x) = \mathbf{g}(u)$ $(=\mathbf{f}(u))$ & $\mathbf{g}(y) \in V_M(\text{book})$ & $\langle \mathbf{g}(x), \mathbf{g}(y) \rangle \in V_M(\text{own})$,

there is a $\mathbf{h} \supseteq_{\{z,\, v,\, w\}} \mathbf{g}$ s.t. $\langle \mathbf{h}(z), \mathbf{h}(v), \mathbf{h}(w) \rangle \in V_M(\text{give})$ & $\mathbf{h}(z) = \mathbf{h}(x)$ $(=\mathbf{g}(x))$ & … etc.

Translation of DRSs to FOL

Consider DRS K = $\langle\{x_1, \ldots, x_n\}, \{c_1, \ldots, c_k\}\rangle$

$$
\begin{array}{|l|}
\hline
x_1 \ \ldots \ x_n \\
\hline
c_1 \\
\vdots \\
c_n \\
\hline
\end{array}
$$

K is truth-conditionally equivalent to the following FOL formula:

$$\exists x_1 \ldots \exists x_n \, [c_1 \wedge \ldots \wedge c_k]$$

DRT and compositionality

- DRT is non-compositional on truth conditions: The difference in discourse-semantic status of the text pairs is not predictable through the (identical) truth conditions of its component sentences.

- Since structural information which cannot be reduced to truth conditions is required to compute the semantic value of texts, DRT is called a *representational* theory of meaning.

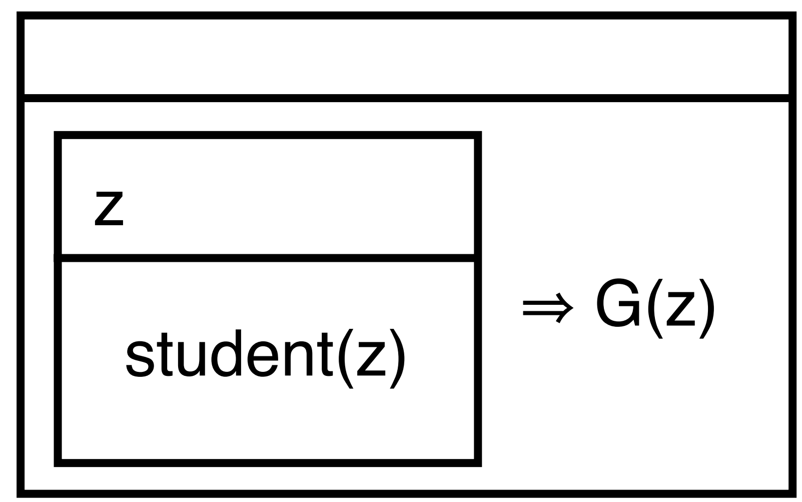
However…

Wait a minute …

- Why can't we just combine type theoretic semantics and DRT?
- Use λ-abstraction and reduction as we did before, but:
- Assume that the target representations which we want to arrive at are not First-Order Logic formulas, but DRSs.
- The result is called λ-DRT.

λ-DRSs

An expression in λ-DRT consists of a lambda prefix and a partially instantiated DRS.

- *every student* :: $\langle\langle e, t\rangle, t\rangle \mapsto$ λG. 

Alternative notation: λG [∅ | [z | student(z)] ⇒ G(z)]

- *works* :: $\langle e, t\rangle \mapsto$ λx [∅ | work(x)]

Muskens, *Linguistics and philosophy* (1996)

λ-DRT: β-reduction

Every student works

$\mapsto \lambda G[\, \varnothing \mid [\, z \mid \text{student}(z)\,] \Rightarrow G(z)\,]](\lambda x\, [\, \varnothing \mid \text{work}(x)\,])$

$\Rightarrow^{\beta} [\, \varnothing \mid [\, z \mid \text{student}(z)\,] \Rightarrow (\lambda x\, [\, \varnothing \mid \text{work}(x)\,])(z)\,]$

$\Rightarrow^{\beta} [\, \varnothing \mid [\, z \mid \text{student}(z)\,] \Rightarrow [\, \varnothing \mid \text{work}(z)\,]]$

How do we define conjunction on DRSs?

(Naïve) Merge

The “merge” operation on DRSs combines two DRSs (conditions and universes).

- Let $K_1 = [\ U_1 \mid C_1\ ]$ and $K_2 = [\ U_2 \mid C_2\ ]$.

Merge: $K_1 + K_2 = [\ U_1 \cup U_2 \mid C_1 \cup C_2\ ]$

Merge: An example

- *a student* $\mapsto$ $\lambda G\, ([\, z \mid student(z)\,] + G(z))$
- *works* $\mapsto$ $\lambda x\, [\, \varnothing \mid work(x)\,]$

A student works $\mapsto \lambda G\, ([\, z \mid student(z)\,] + G(z))\, (\lambda x[\, \varnothing \mid work(x)])$

$\Rightarrow^{\beta} [\, z \mid student(z)\,] + \lambda x[\, \varnothing \mid work(x)](z)$

$\Rightarrow^{\beta} [\, z \mid student(z)\,] + [\, \varnothing \mid work(z)]$

$\Rightarrow^{\beta} [\, z \mid student(z), work(z)]$

Compositional analysis

- *Mary* $\mapsto$ $\lambda G\,([\ z \mid z = Mary\ ] + G(z))$
- she $\mapsto$ $\lambda G.G(z)$

Mary works. She is successful.

$\mapsto$ $\lambda K\ \lambda K'(K + K')([\ z \mid z = Mary,\ work(z)])([\ \mid successful(z)])$

$\Rightarrow^{\beta}$ $\lambda K'([\ z \mid z = Mary,\ work(z)] + K')([\ \mid successful(z)])$

$\Rightarrow^{\beta}$ $[\ z \mid z = Mary,\ work(z)] + ([\ \mid successful(z)])$

$\Rightarrow^{\beta}$ $[z \mid z = Mary,\ work(z),\ successful(z)]$

Merge again

The "merge" operation on DRSs combines two DRSs (conditions and universes).

- Let $K_1 = [\ U_1 \mid C_1\ ]$ and $K_2 = [\ U_2 \mid C_2\ ]$.

Merge: $K_1 + K_2 \Rightarrow [\ U_1 \cup U_2 \mid C_1 \cup C_2\ ]$
under the assumption that no discourse referent $u \in U_2$ occurs free in a condition $\gamma \in C_1$.

Note that under this definition Merge is directional:
$K_1 + K_2 \nLeftrightarrow K_2 + K_1$

Variable capturing

In λ-DRT, discourse referents are captured via the interaction of β-reduction and DRS-binding:

$\lambda K'([z \mid student(z), work(z)] + K')([\ \mid successful(z)])$

$\Rightarrow^{\beta} [z \mid student(z), work(z)] + [\ \mid successful(z)]$

$\Rightarrow^{\beta} [z \mid student(z), work(z), successful(z)]$

- But the β-reduced DRS must be *equivalent* to the original DRS!
- This means that the potential for capturing discourse referents must be captured in the interpretation of λ-DRSs.

➡ Possible, but tricky.

Playing in the sandbox

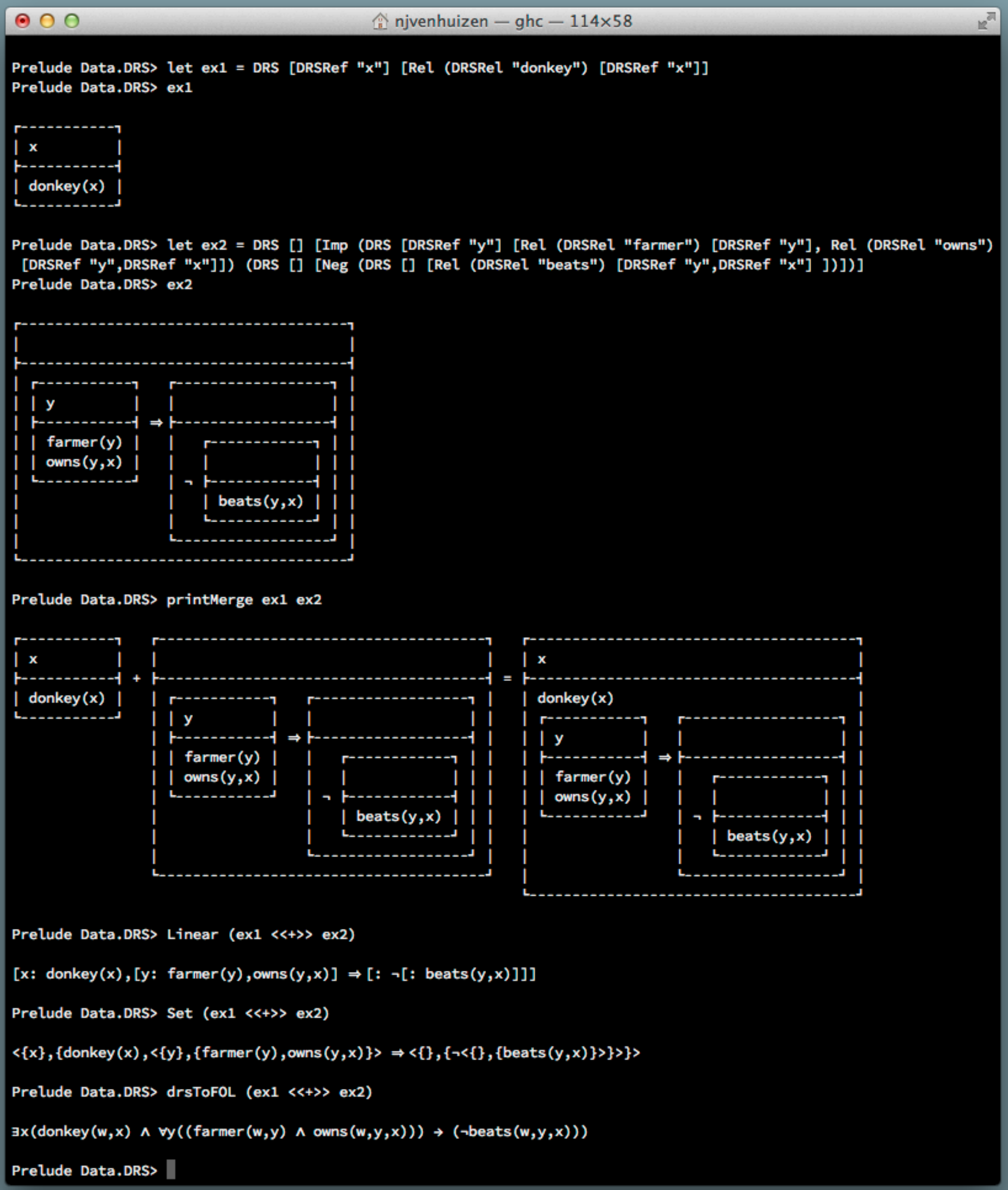
PDRT-SANDBOX is a Haskell library that implements Discourse Representation Theory (and the extension Projective DRT)

http://hbrouwer.github.io/pdrt-sandbox/
also available via: login.coli.uni-saarland.de:/proj/courses/semantics19

- Define your own DRSs, using the internal syntax or the set-theoretic notation
- Show the DRSs in different output formats (boxes, linear boxes, set-theoretic, internal syntax)
- Composition of DRSs (using lambda's)
- Translate DRSs to FOL formulas

PDRT SANDBOX

Playing in the sandbox



PDRT
SANDBOX

DRS Syntax in PDRT-SANDBOX

DRS: `DRS […] […]` referents conditions

Referents: `DRSRef “x”, DRSRef “Mary”`

Conditions:

- Relation: `Rel (DRSRel "man") [DRSRef “x"]`
- Identity: `Rel (DRSRel "=") [DRSRef “x”,DRSRef “y"]`
- Negation: `Neg (DRS […] […])`
- Implication: `Imp (DRS […] […]) (DRS […] […])`
- Disjunction: `Or (DRS […] […]) (DRS […] […])`

Properties: `isPure(DRS […] […]), isProper(DRS […] […])`

Using PDRT-SANDBOX on **coli**

```
~$ cp -r /proj/courses/semantics-19/pdrt-sandbox/ .
~$ cp /proj/courses/semantics-19/ghci .ghci
~$ cd pdrt-sandbox/
~/pdrt-sandbox$ make
[…]
~/pdrt-sandbox$ cd tutorials/
~/pdrt-sandbox/tutorials$ ghci DRSTutorial.hs
GHCi, version 7.10.3: http://www.haskell.org/ghc/  :? for help
[1 of 1] Compiling Main             ( DRSTutorial.hs, interpreted )
Ok, modules loaded: Main.
*Main>
```

Literature

References:

- Hans Kamp and Uwe Reyle. From Discourse to Logic, Kluwer: Dordrecht 1993.
- Reinhard Muskens. "Combining Montague semantics and discourse representation." *Linguistics and philosophy* (1996): 143-186.

Background reading:

- https://plato.stanford.edu/entries/discourse-representation-theory/