A Compositional Analysis of Dependence Statements

MSc Thesis *(Afstudeerscriptie)*

written by

Terence Hui
(born October 16th, 1996 in Toronto, Canada)

under the supervision of **Dr. Maria Aloni**, and submitted to the Examinations Board in partial fulfillment of the requirements for the degree of

MSc in Logic

at the *Universiteit van Amsterdam.*

Date of the public defense:	**Members of the Thesis Committee:**
August 27th 2021	Dr. Benno van den Berg (Chair)
	Dr. Maria Aloni (Supervisor)
	Dr. Paul Dekker
	Dr. Floris Roelofsen

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Contents

Abstract

This thesis investigates the semantic structure of dependence statements which feature the verb *depends on*. The analysis of these statements will be done in a typed system which will be based on an inquisitive notion of meaning. Various types of dependence statements will be presented which are unable to be represented in a standard typed inquisitive semantics. These dependence statements feature concealed questions and propositional anaphora. Two typed inquisitive semantics systems will be presented. The first system that we will present is designed to represent dependence statements which feature concealed questions. This system will be able to represent a number of different readings of dependence statements which contain concealed questions. The second system will be based on a dynamic inquisitive semantics framework and is designed to represent dependence statements which feature propositional anaphora. This framework will allow us to represent the dynamic effect of propositional anaphora in dependence statements.

Acknowledgments

First, I would first like to thank Maria Aloni for all the time she spent helping me with the numerous technical details of this thesis. You are an excellent supervisor and this thesis could not have been finished without your help. I would also like to thank the members of my defence committee, Benno van den Berg, Floris Roelofsen and Paul Dekker, for taking the time to read my thesis, attend my defence and provide feedback.

Thank you to Augie, Fergus, Grace, Ion, Jason, Lukas, Martin, Max, Nikki, Oana, Pepijn, Rachel and Sam for being good friends and making Amsterdam a fun place to live. I would also like to thank Austin, Hesam, Jeremy, Julian, Kevin, Kien, Markus, Ryan and Sam for keeping in touch despite the time differences. Thanks to Ryan and Louis for making sure that I would arrive perfectly on time for my defence.

Thank you to my Aunt Kitty for hosting my for some of my time in Amsterdam. You helped make Amsterdam feel more like home. Thank you to Jasper for always responding to whatever nonsense I sent you. You are a good brother and friend.

Finally, thank you to my parents for supporting me throughout my studies and allowing me to pursue my interests wherever they may take me.

Chapter 1

Introduction

Dependence is a pervasive concept that plays a role in both science and everyday life. Natural sciences like chemistry and physics examine what dependencies hold in nature. For example, in physics, the study of motion consists of discovering what factors determine how an object changes position over time. If we are examining the orbit of the Earth around the sun, we can attempt to discover what factors the speed of Earth's orbit depends on. In chemistry, we can study what factors determine what type of chain reaction occurs in a given scenario. Thus, in both chemistry and physics dependence plays a central role.

Another area where dependencies are important is in database theory. A database is an organized collection of information. Database store information by means of entries. Each entry may have certain attributes which represents the information about the entry. Dependency plays a role in this domain as there are often certain constraints that we want a database to satisfy. For example, if we are making a database for COVID-19 vaccination information, we would want to have certain constraints for each entry. In this case, we would want an ID for each individual. This ID can determine a number of things such as someone's personal information. Additionally, we would want the date of the second vaccination appointment to depend on the date of the first vaccination appointment. Since dependence is so crucial in maintaining a proper database, the topic has been heavily analysed by the database community.

Dependence also plays a role in everyday life. During a conversation, people often use dependence statements in response to questions asked of them. Consider the following exchange:

(1.1) Alice: Will you go to the beach tomorrow?

(1.2) Bob: It depends on the weather. If the weather is nice, I'll go to the beach.
If it isn't, I won't.

In the dialogue, this answer seems like a reasonable answer. However, Bob's response does not literally answer the question that Alice asked as he never explicitly states whether he will go to the beach tomorrow. Instead, Bob establishes a dependency between the weather and whether he will go to the beach. Often dependence statements are made in ordinary conversation. Thus, studying dependence statements is useful for understanding what is meant by certain phrases in ordinary conversation. Further, understanding why this constitutes a coherent dialogue and why the response is an appropriate answer to the question is one of the goals of pragmatics. Thus, the study of dependence statements are important in the field of linguistics as well.

In this thesis, we will focus on the study of dependence statements in the field of linguistics. However, instead of analysing the pragmatics of dependence statements we will focus on analysing dependence statements from a formal semantics perspective. We will present numerous puzzling features of dependence statements that must be explained for a suitable compositional analysis of dependence verbs. We will analyse dependence statements in a typed inquisitive semantics system (Theiler, 2014) (Ciardelli et al., 2017) and provide several ways in which inquisitive semantics can be extended to account for the puzzling cases that we have presented. We will make use of both conceptual covers and dynamic semantics to represent these cases.

The thesis will be structured as follows. The rest of the first chapter will provide an introduction to the concept of dependence. This will be done by presenting various analyses of dependence in inquisitive logic (Ciardelli, 2016), (Ciardelli, 2018), (Theiler et al., 2019) and dependence logic (Väänänen, 2007), (Yang and Väänänen, 2016). The second chapter will present various linguistic puzzles associated with compositionality and the verb *depend(ing) on*. In the last two chapters, we will present novel variations to existing inquisitive semantics frameworks that will allow for a better representation of dependence statements. The third chapter will cover how conceptual covers and concealed questions (Aloni and Roelofsen, 2011) can be incorporated into a typed inquisitive semantics. This will allow for the representation of dependence statements with concealed questions and identity questions. The fourth chapter will cover how modifications to dynamic inquisitive semantics (Dotlačil and Roelofsen, 2019) can allow dependence statements with propositional anaphora to be represented. This system will allow for the representation of anaphoric expressions which refer to inquisitive propositions.

1.1 Dependence

In this section, we will outline the specific notion of dependence that we seek to represent and introduce several formal representations of this type of dependence. We will begin by presenting a scenario which will demonstrate the specific relationship of dependence that we seek to analyse. In the following sections, we will provide an introduction to both dependence logic and inquisitive semantics. Then, we will present an analysis of dependence by Ciardelli (2016) where he argues that the notion of dependence can be characterized in inquisitive semantics as question entailment. In doing so, we will also present his argument that the analysis of dependence as a relationship between questions affords several benefits over analyzing dependence as a relation between variables. We will, then, examine further representations of dependence in inquisitive semantics by presenting Ciardelli (2018) where he argues that the original dependence logic and inquisitive logic formulations of dependence are not the correct way to formalize dependence statements. Following this, we will present an analysis of dependence statements in a typed semantics by Theiler et al. (2019) that builds on the analysis by Ciardelli (2018). Finally, we will present several arguments for why the analysis by Theiler et al. (2019) successfully captures several features of dependence statements.

1.2 An example of dependence

Example 1. Suppose Amy plans to go to the beach, but Amy will only go provided that it is sunny and it is the weekend. Given Amy's considerations about the weather we can conclude whether she will go to the beach depends on two things: (1) whether it is sunny and (2) whether it is the weekend. The relationship between Amy's decision to go the beach and the weather and time of the week is what we seek to examine in this thesis. We consider this relationship to be a form of dependence at the propositional level. This relationship can also be expressed by means of questions and answers. If someone has the answers to the questions about the weather and the time of the week, then they will have the answer to the question about Amy going to the beach. Usually in natural language, this relationship would be expressed by the statement in (1.3a).

(1.3) a. Whether Amy will go the beach depends on whether it is the weekend and whether it is sunny.

b. Whether Amy will go the beach depends on whether it is the weekend.

c. Whether Amy will go the beach depends on whether it is sunny.

Sentences with the form of these natural language expressions of this relationship will be the focus of this thesis. From the descirbed scenario, we can infer that niether (1.3b) nor (1.3c) are true. This is the case as it is impossible to determine in the scenario described whether Amy will go to the beach simply from knowing the answer to one of the questions in the object position of (1.3a).

There are two things to note. First, in this scenario it is true that if one knows either that it is not the weekend or it is not sunny, one can conclude that Amy will not go to the beach. We will call this relationship partial dependency, which we will not cover in this thesis. The second thing to note about this natural language representation of the example above is that less information is given in (1.3a) than in the example described. This is because in our example, we specified the exact way in which Amy's decision to go to the beach depended on the weather and the time of the week. By making a dependence statement, one does not specify how the subject depends on the object but simply that this relationship holds. Thus, the notion of dependence we seek to analyse is dependence between questions.

1.3 Dependence logic

In this section, we will briefly present dependence logic and show how the previous example can be represented in this system. Dependence logic emerged from the observation by Henkin that certain patterns of quantification over individuals were not possible in first-order logic. More specifically, expressing that certain values are determined by other values is not possible in first-order logic. Thus, logics were designed to represent this type of quantification. We will begin by presenting a dependence logic by Väänänen (2007). This logic is enriched with a new type of atomic formula $=(x_1, ..., x_n, y)$ which expresses the fact that the value of y is determined by the values of $x_1, ..., x_n$. In the context of a standard model and assignment, there is no way of making sense of values of variables being determined by other variables as there is only a singular assignment which gives each variable a specific value. Thus, formulas in dependence logic are interpreted with respect to a model and a set of assignments, called a team. The value of y is determined by the values of $x_1, ..., x_n$ relative to a team X in the case that the value that an assignment $g \in X$ assigns to y is determined by the values it assigns to $x_1, ..., x_n$. Thus, the semantic clause for dependence atoms is the following:

(1.4) $$M \vDash_X = (x_1, ..., x_n, y) \text{ iff } \forall g, g' \in X, \text{ if } g(x_i) = g'(x_i) \text{ for all } i, \text{ then } g(y) = g'(y)$$

This type of relationship between variables can also be expressed for the valuation of propositions. This variation of dependence logic is called Propositional Dependence Logic. Propositional Dependence Logic features atomic formulas which express that the value of a proposition is determined by the values of a sequence of propositions. Thus, one can represent with propositional dependence logic that the truth or falsity of some proposition is dependent on the truth of one or more propositions. Instead of teams of assignment functions for entities, Propositional Dependence Logic uses sets of valuation functions from propositions to truth values. Thus, a dependence atom in Propositional Dependence Logic has the following form: $=(p_1, ..., p_n, q)$ and is interpreted relative to a set of valuations s and expresses that the value of q is determined by the truth values of $p_1, ..., p_n$.

Using this, one can easily represent the natural language expression in the example presented in the last section as $=(W, S, B)$ where W represents the proposition 'it is the weekend', S represents 'it is sunny' and B represents 'Amy goes to the beach'.

1.4 Inquisitive semantics

We will now present the basic inquisitive semantics framework which can be used to analyse dependence statements. Inquisitive semantics is a semantic framework designed to analyse linguistic information exchange (Ciardelli et al., 2018). From the perspective of inquisitive semantics, information exchange is seen as a process of raising and resolving issues. Thus, one of the goals of inquisitive semantics is to provide a uniform analysis of declarative and interrogative propositions. Inquisitive propositions in a possible world semantics are defined as follows:

Definition 1.4.1 (Inquisitive propositions)**.**

- An information state is a set of worlds.
- Downward closure: A set S is downward closed iff for every $P \in S$ every $P' \subseteq P$, $P' \in S$.
- An inquisitive proposition is a downward closed set of states.

Inquisitive semantics, thus, evaluates propositions as sets of sets of worlds rather than sets of worlds as is the case with classical propositions. The notable difference from other question semantics frameworks like Hamblin semantics is the requirement of downward closure.

Questions are often characterized by the set of propositions that resolve them. Thus, downward closure is motivated from the fact that if a proposition P resolves a question ϕ, any proposition P' that is stronger than P should also resolve ϕ. Since we want questions to be characterized by all the propositions which resolve them, downward closure is a suitable property. In this thesis, we will refer to propositions in an inquisitive semantics framework as inq-propositions, declaratives sentences as non-inquisitive inq-propositions and interrogative sentences as inquisitive inq-propositions. We will now define the language $\mathcal{L}$ of Inq_B and the models which we will use to interpret this language. Our presentation of inquisitive semantics will be based on the version presented by Ciardelli et al. (2018).

Definition 1.4.2 (The language $\mathcal{L}$ of first-order Inq_B)**.**

$\phi ::= R(t_1, ..., t_n)|\neg\phi|\phi \wedge \phi|\phi \vee \phi|\phi \to \phi|\exists x\phi(x)|\forall x\phi(x)$, where

- R is an n-ary relational symbol
- $t_1, .., t_n$ are terms from a set of constants C
- x is a variable

To simplify the notation of the language, Ciardelli et al. (2018) assume that for any $d \in D$, the language $\mathcal{L}$ contains an individual constant d' such that $I_w(d') = d$ for all $w \in W$: If this is not the case the language is extended with new constants and we extend I to accommodate this. Thus, this version of the system does not contain formulas with free variables and therefore there are no variable assignments altogether. This move is not necessary but it is used to simplify the notation.

Definition 1.4.3 (Rigid interpretation model for first-order Inq_B)**.** A rigid Information model for first-order Inq_B is a triple $\langle W, D, I\rangle$ where:

- W is a set of possible worlds;
- D is a non-empty set of individuals
- I is a interpretation function s.t. for every world $w \in W$:
  - for every constant $c, I(c) \in D$;
  - for every n-ary relational symbol $R, I(w)(R) \subseteq D^n$

Definition 1.4.4 (Semantics of first-order Inq_B)**.**

Given a Rigid interpretation model $M = \langle W, D, I\rangle$

- $[\![R(t_1, ..., t_n)]\!] := \{s \subseteq W|\forall w \in s : \langle I(t_1), ..., I(t_n)\rangle \in I(R)\}$;
- $[\![\neg\phi]\!] := \{s \subseteq W|s \cap t = \emptyset \text{ for all } t \in [\![\phi]\!]\}$
- $[\![\phi \wedge \psi]\!] := [\![\phi]\!] \cap [\![\psi]\!]$
- $[\![\phi \vee \psi]\!] := [\![\phi]\!] \cup [\![\psi]\!]$
- $[\![\phi \to \psi]\!] := \{s \subseteq W| \text{ for every t } \subseteq s, \text{ if } t \in [\![\phi]\!] \text{ then } t \in [\![\psi]\!]\}$
- $[\![\forall x.\phi(x)]\!] := \cap_{d \in D}[\![\phi(d')]\!]$

- $[\![\exists x.\phi(x)]\!] := \cup_{d \in D}[\![\phi(d')]\!]$

We will now present several features of propositions in inquisitive semantics.

Definition 1.4.5 (Alternatives)**.**

- The maximal elements of a inq-proposition P are called the alternatives in P.
- For any inq-proposition P, alt(P) is the set of alternatives in P.

Definition 1.4.6 (Informative Content)**.** For any proposition P, info(P):= $\bigcup P$

Definition 1.4.7 (Inquisitive and informative propositions)**.**

- A proposition P is informative iff info$(P) \neq W$
- A proposition P is inquisitive iff alt$(P) > 1$

Definition 1.4.8 (Truth set)**.** For any $\phi \in \mathcal{L}$, the set of worlds where ϕ is classically true is called the truth set of ϕ and denoted as $|\phi|$

Through the definition of truth-sets we can also recursively define each sentence of our language as an inquisitive proposition $[\phi]$. These definition will be helpful for when we introduce the semantics of inq-propositions in a typed system. Before we provide these definitions we will define Absolute and Relative Pseudo-complements.

Definition 1.4.9 (Absolute and Relative pseudo-complements.)**.**

For any inq-propositions P, Q, the pseudo-complement of P relative to Q amounts to:

- $P \Rightarrow Q := \{s|$ for every $t \subseteq s, t \in P$ then $t \in Q\}$

For any inq-proposition P, the absolute pseudo complement of P amounts to:

- $P^* := \{s|s \cap t = \emptyset$ for all $t \in P\}$

Definition 1.4.10 (Propositional semantics for Inq$_B$)**.**

1. $[\phi] := \wp(|\phi|)$
2. $[\neg\phi] := [\phi]^*$
3. $[\phi \wedge \psi] := [\phi] \cap [\psi]$
4. $[\phi \vee \psi] := [\phi] \cup [\psi]$
5. $[\phi \rightarrow \psi] := [\phi] \Rightarrow [\psi]$
6. $[\forall x.\phi(x)] := \cap_{d \in D}[\phi(d')]$
7. $[\exists x.\phi(x)] := \cup_{d \in D}[\phi(d')]$

Another way of characterizing the semantics of propositions in inquisitive semantics is through support conditions.

Definition 1.4.11 (Support Conditions)**.**

- $s \vDash R(t_1, ..., t_n)$ iff $s \subseteq |R(t_1, ..., t_n)|$

- $s \vDash \neg\phi$ iff $\forall t \subseteq s$: if $t \neq \emptyset$ then $t \nvDash \phi$
- $s \vDash \phi \wedge \psi$ iff $s \vDash \phi$ and $s \vDash \psi$
- $s \vDash \phi \vee \psi$ iff $s \vDash \phi$ or $s \vDash \psi$
- $s \vDash \phi \rightarrow \psi$ iff $\forall t \subseteq s$: if $t \vDash \phi$ then $t \vDash \psi$
- $s \vDash \forall x \phi(x)$ iff $s \vDash \phi(d')$ for all $d \in D$
- $s \vDash \exists x \phi(x)$ iff $s \vDash \phi(d')$ for some $d \in D$

With this notion of support conditions we can also define entailment and entailment within a context which will be important for when we examine dependence in inquisitive semantics.

Definition 1.4.12 (Entailment and Entailment within a context)**.**

- $\Phi \vDash \psi \iff$ for every information state s, if $s \vDash \phi$ for all $\phi \in \Phi$, then $s \vDash \psi$
- $\Phi \vDash_s \psi \iff$ for every information state $t \subseteq s$, if $t \vDash \phi$ for all $\phi \in \Phi$, then $t \vDash \psi$

Finally, we will present two projection operators ! and ? that are commonly used in inquisitive semantics.

Definition 1.4.13 (Projection Operators)**.**

- The issue-cancelling operator ! :$!\phi := \wp(info(\phi))$
- The info-cancelling operator ?: $?\phi := \phi \cup \neg\phi$

The issue-cancelling operator ! turns a proposition ϕ into a non-inquisitive operator. Thus, it becomes a proposition with a single alternative $info(\phi)$. The ? operators turns a proposition ϕ into a non-informative proposition $\phi \vee \neg\phi$.

For a more detailed outline of Inq_B see Ciardelli et al. (2018).

Now that we have presented the basics of inquisitive semantics, we will present how example 1 in section 1.2 can be represented using inquisitive semantics according to Ciardelli (2016). This analysis argues that in inquisitive semantics the dependence relation is simply entailment within a context. Recall earlier we presented an example about Amy going to the beach where we said the following statement was true:

(1.5) Whether Amy will go the beach depends on whether it is the weekend and whether it is sunny.

Let B represent the proposition 'Amy will go to the beach', W the proposition 'It is the weekend' and S the proposition 'it is sunny'. Additionally, let s represent the context we described regarding Amy's considerations about going to the beach. Possible worlds will be represented using a sequence of digits to denote the truth values of the relevant propositions, B, W and S. Given the context we have described there are only 4 worlds: $101, 010, 011, 000$. The maximal sub-states of s which settle the questions in Figure 1.1 are represented using a shaded rectangle. As dependence in inquisitive semantics is considered to be entailment within a context, we will say that (1.5) is true in the context s when $?W \wedge ?S \vDash_s ?B$. We will also show that neither $?W$ nor $?S$ entail $?B$ in s as Amy going to the beach is not dependent on the weekend or the sun independently. Recall earlier our definition of entailment in a context.

$$\Phi \vDash_s \psi \iff \text{ for every information state } t \subseteq s, \text{ if } t \vDash \phi \text{ for all } \phi \in \Phi, \text{ then } t \vDash \psi$$

As we can see in the diagrams below a state settles the question $?B$ just in case all the worlds in the state share the same truth value for B. As the context only has one world which evaluates B as true, any question that implies $?B$ must contain $\{101\}$ as a maximal state. Since neither $?S$ nor $?W$ satisfy this requirement they do not entail $?B$. This can be seen in Figures 3.4(b,c) as there are maximal substates that include 101 and other worlds. However, we can see that $?W \wedge ?S$ does entail $?B$ as each maximal substate of $?W \wedge ?S$ settles $?B$.

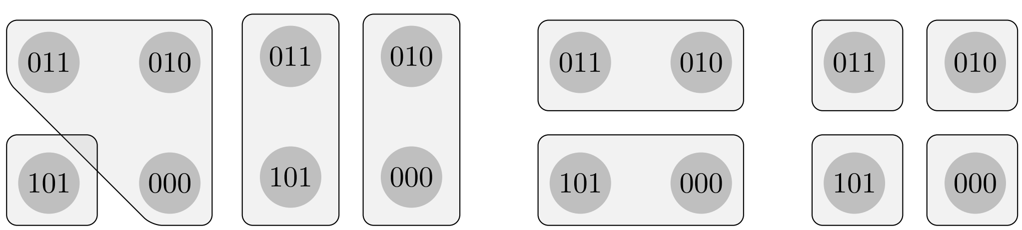
From this we can conclude that $s \vDash ?W \wedge ?S \rightarrow ?B$ and $s \nvDash ?W \rightarrow ?B$ and $s \nvDash ?S \rightarrow ?B$. This is because the context that entailment is relativized to is the same kind of object as the objects which formulas are evaluated on in a support based semantics. Further, the notion of implication in a support-based semantics is exactly the same as entailment within a context.

(1.6) $$s \vDash \phi \rightarrow \psi \iff \phi \vDash_s \psi$$

The relationship between entailment within a context $\vDash_s$ and implication $\rightarrow$ can be seen more clearly when we explicitly state the definitions. The definition of implication in a support based semantics exactly mirrors the definition of entailment within a context.

(1.7) $$s \vDash \phi \rightarrow \psi \iff \text{ for any state } t \subseteq s : \text{if } t \vDash \phi \text{ then } t \vDash \psi$$
$$\Phi \vDash_s \psi \iff \text{ for any state } t \subseteq s, \text{ if } t \vDash \phi \text{ for all } \phi \in \Phi, \text{ then } t \vDash \psi$$

Thus, we have managed to capture the relationship in our initial example using entailment within a context and implication in inquisitive semantics.



(a) ?B (b) ?S (c) ?W (d) $?W \wedge ?S$

Figure 1.1: The meaning of the questions in the given context. For readability, we only represent the maximal states of each question.

1.5 Dependence between variables or questions?

We will now compare the approaches to dependence in dependence logic and inquisitive semantics. Dependence logic analyses dependence as a relationship between variables while inquisitive semantics regards it as a relationship between questions. We have shown that given our example, both systems are able to represent the relationship in a satisfactory way.

Given the example that we have presented, variables in dependence logic simply correspond to questions in inquisitive logic and vice versa. However, it has been noted by Ciardelli (2016) that not all questions can be translated into variables in dependence logic. This is the case for mention-some questions such as: When can I get vaccinated? Where can I buy a sandwich? These questions can be answered by mentioning a singular instance and thus do not have a unique true answer. These questions cannot be represented as variables in dependence logic as variables are bound to have a unique value at a world. It was noted by van Rooij (2004) that questions which can have a mention-some reading never get this reading in dependence statements. However, questions which do not contain a unique true answer can be featured in a dependence statement. Consider the following dependence statement:

(1.8) Whether I go to the party depends on whether one of John and Mary go to the party.

The object question in this dependence statement does not contain a unique true answer at the world where both John and Mary are going to the party as answering either 'John' or 'Mary' constitutes a true answer. This type of question was shown to be unrepresentable in dependence logic by Ciardelli (2016). This is because dependence atoms represent dependencies between atomic polar questions. However, in this case the question cannot be represented by polar questions. Thus, inquisitive semantics affords benefits when representing dependencies as dependencies between questions with unique true answers can be represented. These same arguments apply to the newer version of dependence logic developed by Baltag and van Benthem (2021).

Another benefit of representing dependency as a relationship between questions is that it better reflects natural language as dependence verbs embed questions and inquisitive semantics is better at representing questions. Further, there is a typed version of inquisitive semantics which will allow us to analyse dependence statements compositionally. Because of these reasons, we will focus on using an inquisitive semantics analysis of dependence in this thesis. We will not discuss further the relationship between these two logics. For more information about the relationship between inquisitive semantics and propositional dependence logic see Yang and Väänänen (2016) and Ciardelli (2016).

1.6 Dependence as strict conditionals

In the previous section, we represented dependence in inquisitive semantics as entailment within a context. As shown by Ciardelli (2018) this analysis of dependence leads to several problems when trying to represent dependence statements such as the ones in (1.9). In this section, we will show the problems with an analysis of dependence as an inquisitive conditional and present a new analysis of dependence by Ciardelli (2018) which provides a better account of dependence in inquisitive semantics.

(1.9)
a. Whether p depends on whether q.
b. Whether p does not depend on whether q.
c. Whether p depends on whether q, or it is depends on r.
d. Alice knows that whether q depends on whether p.
e. Is it the case that whether q depends on whether p?

Consider (1.9a) which can be intuitively false given certain circumstances. E.g. it is intuitively false in the actual world if we are talking about two completely unrelated things such as 'the weather in Amsterdam' and 'my hair colour being black.' Let p represent the former and q the latter. Intuitively this is false in the actual world given that I do not change my hair colour given different weather in Amsterdam. However, $?q \to ?p$ at the actual world @ is true at every model as relative to $\{@\}$ p depends on q trivially. This will be the case for any singleton set of worlds no matter what truth values are assigned to p and q. Thus, dependence is not properly represented in this example.

Negated dependence statements are also not properly represented with an inquisitive conditional. Consider $\neg(?q \to ?p)$ which is the natural translation of (1.9b). This statement is a contradiction in inquisitive semantics. However, (1.9b) can be true given certain circumstances. Thus, it seems that the standard inquisitive conditional does not properly represent dependence. Here, we will not examine why (1.9c-e) are not represented properly by the inquisitive conditional. Further reasons for why the inquisitive conditional is not sufficient are provided by Ciardelli (2018).

We will now move onto representing dependence as a strict conditional $\Rightarrow$. To interpret this conditional in inquisitive semantics we need to equip our worlds with a binary relation R which captures relative possibility. This binary relation defines a set of worlds which are relevant to the dependence statement. Thus, we will now evaluate statements on a first-order Kripke Model: $M := \langle W, D, R, I \rangle$ where W is a set of worlds, D is a domain of individuals, R is a binary relation on W and I is an interpretation function. In the context of dependence statements R can mean many things. For example, in the scenario where Amy is deciding to go to the beach, R defines the possible ways that the weather could be like. The support clause for $\Rightarrow$ is defined as follows:

Definition 1.6.1 (Support for $\Rightarrow$). $M, s \vDash \phi \Rightarrow \psi$ iff $\forall w \in s : \phi \vDash_{R[w]} \psi$

Analyzing dependence statements as strict conditionals allows the proper analysis of the statements in (1.9). Representing (1.9a) as $?p \Rightarrow ?q$ will not lead to the same problem outlined above as the proposition will be evaluated based on the set of worlds defined by R given the actual world @. Thus, we will not run into the case where dependence statements are trivially true despite being intuitively false. Further, this representation prevents (1.9b) from always being a contradiction as it will be true if $?p \Rightarrow ?q$ is false. Thus, representing dependence statements as modal statements better illustrates the relationship of dependence that we want to capture.

Ciardelli (2018) showed that dependence can also be defined in functional terms. The definition in (1.10) expresses dependence between polar questions. However, it can also be the case that the alternatives of either the subject or object question is greater than 2, in this case this definition is inaccurate. Instead, dependence requires the existence of a function from the alternatives of the object question to the alternatives of the subject question.

(1.10) $M, w \vDash ?p \Rightarrow ?q$ iff
$\exists f : \{0,1\} \to \{0,1\}\ s.t.\ \forall v \in R[w] : V(v,q) = f(V(v,p))$

(1.11) $M, w \vDash ?p \Rightarrow ?q$ iff
$\exists f : alt(?q) \to alt(?p)\ s.t.\ \forall p \subseteq R[w] : \forall \alpha \in alt(?q)(p \subseteq \alpha \to p \subseteq f(\alpha))$

We will now move to a compositional analysis of dependence statements based on the analysis of dependence as a strict conditional presented by Theiler et al. (2019).

1.7 Compositional dependency

Theiler et al. (2019) analyses dependence verbs as question embedding verbs in a typed framework. This analysis of dependence relies on the functional definition of dependence as defined in Ciardelli (2018). Additionally, further semantics for dependence are specified which allows Theiler et al. to explain why *depends on* does not allow for declarative complements and why dependence statements are not true given cases of trivial dependence. In this section, we will present typed inquisitive semantics (Theiler (2014)) which we will refer to as Inq_B^λ for short and define the semantics of the verb *depends on* as provided by Theiler et al. (2019). Finally, we will present several arguments for this analysis of dependence.

In Inq_B^λ, English sentences will be translated into a type-theoretic language. Their corresponding expressions will, then, receive a model-theoretic interpretation. Similarly to in Inq_B, our type theoretic language will evaluate formulas based on a triple $\langle W, D, R, I \rangle$ where W is a set of worlds D is a domain of objects, R is a binary relation of W and I is an interpretation function. All worlds in the model share domain D. We will provide the syntax and semantics of the theory of types before defining a semantics in Inq_B^λ for a small fragment of English.

1.7.1 Typed inquisitive semantics

Definition 1.7.1 (Types)**.** The set T of types in type theory is the smallest set such that

- $e, t, s \in \mathrm{T}$
- if $a, b \in \mathrm{T}$, then $\langle a, b \rangle \in \mathrm{T}$

The vocabulary of $\mathcal{L}$ consists of the following symbols:

1. for every type σ, an infinite set of VAR_σ of type σ
2. for every type σ a possibly empty set CON_σ^L of constants of type σ
3. the connectives $\wedge, \vee, \to, \neg$
4. the quantifiers $\exists, \forall$
5. the identity symbol $=$

6. the brackets (and)

The syntax for this language is as follows:

1. If $\alpha \in VAR_\sigma^\mathcal{L}$, or $\alpha \in CON_\sigma^\mathcal{L}$, then $\alpha \in WE_\sigma^\mathcal{L}$
2. If $\alpha \in WE_{\langle\sigma,\tau\rangle}^\mathcal{L}$ and $\beta \in WE_\sigma^\mathcal{L}$, then $(\alpha(\beta)) \in WE_\tau^\mathcal{L}$
3. If $\phi, \psi \in WE_t^\mathcal{L}$, then $\neg\phi, \phi \vee \psi, \phi \wedge \psi, \phi \rightarrow \psi \in WE_t^\mathcal{L}$
4. If $\phi \in WE_t^\mathcal{L}$ and $v \in VAR_\sigma^\mathcal{L}$ then $\exists v\phi, \forall v\phi \in WE_t^\mathcal{L}$
5. If $\alpha, \beta \in WE_\sigma^\mathcal{L}$, then $(\alpha = \beta) \in WE_t^\mathcal{L}$
6. If $\alpha \in WE_\sigma^\mathcal{L}$ and $v \in VAR_\tau^\mathcal{L}$ then $\lambda v\alpha \in WE_{\langle\sigma,\tau\rangle}^\mathcal{L}$
7. For any σ all elements of $WE_\sigma^\mathcal{L}$ are constructed using (1-6) in a finite number of steps.

We will now specify the domain of interpretations for the different types in type theory. A domain for a type σ is defined based on a set of worlds W and a domaing of individuals D

Definition 1.7.2 (Domain of Types)**.**

- $D_{e,D,W} = D$
- $D_{s,D,W} = W$
- $D_{t,D,W} = \{0, 1\}$
- $D_{\langle\sigma\tau\rangle,D,W} = \{f | f : D_{\sigma,D,W} \rightarrow D_{\tau,D,W}\} = D_{D_{\tau,D,W}}^{D_{\sigma,D,W}}$

As mentioned earlier, a model M for a type theoretic language $\mathcal{L}$ is a tuple (D, W, R, I) where D is a domain of individuals, W is a set of worlds and I is an interpretation function (a function mapping expressions from $CON_\sigma^\mathcal{L}$ to objects in D_σ.) The extension of an expression α is $[\![\alpha]\!]_{M,w,g}$ and is defined relative to a model M a world $w \in W$ and an assignment function g (a function from $VAR_\sigma^\mathcal{L}$ to objects in D_σ). All worlds in our model will share the same domain.

Definition 1.7.3 (Extension of expressions)**.**

1. If $\alpha \in CON_\sigma^\mathcal{L}$, then $[\![\alpha]\!]_{M,g} = I(\alpha)$

   If $\alpha \in VAR_\sigma^\mathcal{L}$, then $[\![\alpha]\!]_{M,g} = g(\alpha)$
2. If $\alpha \in WE_{\langle\sigma,\tau\rangle}^\mathcal{L}$ and $\beta \in WE_\sigma^\mathcal{L}$, then$[\![\alpha(\beta)]\!]_{M,g} = [\![\alpha]\!]_{M,g}([\![\beta]\!]_{M,g})$
3. If $\phi, \psi \in WE_t^\mathcal{L}$, then:
   - $[\![\neg\phi]\!]_{M,g} = 1$ iff $[\![\phi]\!]_{M,g} = 0$
   - $[\![\phi \wedge \psi]\!]_{M,g} = 1$ iff $[\![\phi]\!]_{M,g} = [\![\psi]\!]_{M,g} = 1$
   - $[\![\phi \vee \psi]\!]_{M,g} = 1$ iff $[\![\phi]\!]_{M,g} = 1$ or $[\![\psi]\!]_{M,g} = 1$
   - $[\![\phi \rightarrow \psi]\!]_{M,g} = 1$ iff $[\![\phi]\!]_{M,g} = 0$ or $[\![\psi]\!]_{M,g} = 1$
4. If $\phi \in WE_t^\mathcal{L}$ and $v \in VAR_\sigma^\mathcal{L}$, then:

- $[\![\forall v\phi]\!]_{M,g} = 1$ iff for all $d \in D_\sigma : [\![\phi]\!]_{M,g[v/d]} = 1$
- $[\![\exists v\phi]\!]_{M,g} = 1$ iff for some $d \in D_\sigma : [\![\phi]\!]_{M,g[v/d]} = 1$

5. If $\alpha \in WE^{\mathcal{L}}_{\sigma}$ and $v \in VAR^{\mathcal{L}}_{\tau}$, then $[\![\lambda v\alpha]\!]_{M,g}$ is that function $h \in D_\sigma^{D_\tau}$ such that for all $d \in D_\tau : h(d) = [\![\alpha]\!]_{M,g[v/d]}$
6. If $\alpha, \beta \in WE^{\mathcal{L}}_{\sigma}$, then $[\![\alpha = \beta]\!] = 1$ iff $[\![\alpha]\!]_{M,g} = [\![\beta]\!]_{M,g}$

These semantic clauses provide us the worlds where a proposition is classically true. The denotation of a sentence in typed inquisitive semantics will mirror the definition of a inq-proposition in Inq_B. Recall earlier that we defined the denotation of $R(t_1, ..., t_n)$ as $\wp(|R(t_1, ..., t_n)|)$ which is a set which contains all the sub-states where $R(t_1, ..., t_n)$ is true. Thus, for an atomic proposition, we will want to create a λ-expression which is the set of states where that atomic proposition is supported. As an inq-proposition is a set of information states, propositions in typed inquisitive semantics will be of type: $\langle\langle s, t\rangle, t\rangle$. Thus, inq-propositions will be of the form $\lambda p_{\langle s,t\rangle}.\phi$. This will provide us all the states where ϕ is supported. We will show how this provides us with the correct denotation by means of an example. Suppose $\phi = \forall w \in p : walks(w)(j)$. $\lambda p.\phi$ will denote the set of states such that for all worlds in each state John walks. Thus, it is an inq-proposition as it is a set of states that is downward closed.

1.7.2 Dependence in typed inquisitive semantics

Now that we have defined the syntax and semantics for an inquisitive type theory, we will move on to our goal of representing dependence verbs in a compositional way. We will first present several features of dependence statements. Then, we will present the definition of *depends on* in Theiler et al. and several improvements that this analysis has over the analysis given in Ciardelli (2018).

Depends on is a question embedding verb. This means that under a compositional analysis the meaning of *depends on* should not allow for it to combine with declarative arguments. Thus, the meaning of a dependence verb should allow for sentences like (1.12a) where *depends on* only takes inquisitive arguments and not (1.12b-d) where *depends on* takes declarative arguments.

(1.12)
a. Whether John will go to the beach depends on what the weather is like.
b. *That John will go to the beach depends on what the weather is like.
c. *Whether John will go to the beach depends on that the weather is good.
d. *That John will go to the beach depends on that the weather is good.

As we are working in an inquisitive semantics framework, both declaratives and interrogative propositions will be given the same type. Additionally, declarative complements are never inquisitive which means that they only contain one alternative. Conversely, interrogative complements are never informative and thus, will completely cover the set of all possible worlds. For example, if the domain of discourse only includes John and the set of worlds contains two worlds: w_j where John will go to the beach and $w_\emptyset$ where John won't go to the beach. The alternatives of 'whether John goes to the beach' are $\{w_j\}$ and $\{w_\emptyset\}$.

As shown in the previous section, dependence verbs are best represented as modal statements. In this formulation of dependence verbs we will evaluate dependence relative to a specific set of possible worlds σ_w i.e. a modal base. This modal base is occasionally given by context as in (4.11a) or implied by context as in (4.11b). In (4.11b) the modality would be something like 'according to the schedules for the bus and the boat'.

(1.13) a. According to Canadian law, one's right to vote depends on one's age.
b. You may take the bus or the boat. It depends on the time of day.

Thus, the definition given by Theiler et al. (2019) will evaluate dependence statements relative to a modal base. The following is the analysis the dependence operator DEP_w.

(1.14) $[\![DEP_w(P_T, P'_T)]\!]_{M.g} = 1$ iff $\exists f \in alt(P)^{alt(P')}$ such that:
(i) $\forall p \subseteq \sigma_w . \forall \alpha \in alt(P').(p \subseteq \alpha \rightarrow p \subseteq f(\alpha))$ and
(ii) $\exists \alpha, \alpha' \in alt(P').\alpha \cap \sigma_w \neq \emptyset \wedge \alpha' \cap \sigma_w \neq \emptyset \wedge f(\alpha) \neq f(\alpha'))$

Clause (i) is the same notion of dependency given by Ciardelli (2018). The second clause (ii) prevents dependence verbs from combining with non-inquisitive propositions and prevents cases of trivial dependency. Consider sentence (1.12b) where the subject of *depends on* is P and the object is P'. Recall earlier that declarative complements only contain a single alternative. P only contains a single alternative as it is a declarative complement. Since P only contains a single alternative (1.12b) will come out as a contradiction as condition (ii) will not be satisfied because there is no function f which does not map every element of alt(P') onto the same element of alt(P). Now consider sentence (1.12c) where the non-inquisitive proposition is in the object position. This sentence will come out as false as there is only one alternative in P' and thus, again there is no function f which does not map every element of alt(P') onto the same element of alt(P). That non-inquisitive arguments lead to contradictions explains why depends on can only take inquisitive arguments.

Another motivation for this definition of dependence is its ability to predict trivial cases of dependency as infelicitous. These are cases where the function f maps every alternative in alt(P') compatible with the modal base σ_w to the same alternative in alt(P). We will show that this case should be ruled out by means of a scenario. Consider the dependence statement in (1.14a) and consider the scenario in which Canadian law allows everyone to vote. Thus, no matter what one's age is they are still able to vote. In this scenario, we would not say that (1.14a) is true as one's right to vote does not at all depend on one's age as everyone has the right to vote. Since everyone can vote, one's age has no influence on one's ability to vote and we would not want to state that one thing depends on the other. Thus, this definition has further motivation as it does not allow for cases of trivial dependency.

In the following chapters, we will use the analysis of Theiler et al. (2019) as a foundation for our analysis of dependence statements as it has several improvements over the analysis of dependence provided by Ciardelli (2018).

Chapter 2

Dependence statements: Puzzles and Solutions

In the following section, we will present various examples of dependence statements. These dependence statements will feature dependence verbs that do not seem to have questions as arguments. At first glance, these examples suggest that dependence verbs cannot straightforwardly be analyzed as entailment between questions as suggested by Ciardelli (2018) and Theiler et al. (2019). However, we will show upon further analysis that an inquisitive semantics framework with the addition of several tools from formal semantics can properly represent these cases.

2.1 Sentential arguments

As discussed in the previous chapter, *depends on* is a question embedding verb. It can associate interrogative complements in both subject and object position. Additionally, declarative complements create ungrammatical sentences when featured in either position.

(2.1) a. Whether we go to the beach depends on what the weather is like.
b. *That I will go to the beach depends on what the weather is like.
c. *Whether I will go to the beach depends on that the weather is good.
c. *Whether I will go to the beach depends on that the weather is good or bad.

In the previous chapter, we showed that the analysis by Theiler et al. (2019) predicts the inability for declarative complements to combine with *depends on* felicitously. Additionally, in the rest of this thesis, we will assume that 'that' operates similarly to the issue cancelling operator in inquisitive semantics. As the definition of dependence by Theiler et al. (2019) precludes the possibility of propositions which only contain one alternative, complements that begin with a 'that' will never be suitable arguments as they are not inquisitive.

2.2 Anaphoric arguments

Dependence verbs are also able to take anaphoric expressions such as 'it' or 'that' in both subject and object position. These anaphoric expressions are able to refer to various types of interrogative propositions. The fact that dependence verbs can take interrogative anaphora is predicted given that dependence verbs can take interrogative complements. We will begin by presenting dependence statements which have 'it' in the subject position which are preceded by a question.

(2.2) a. When are we going to the beach? It depends on the weather.

b. Would you like tea or coffee? It depends on what type of tea you have.

c. Do you like coffee? It depends on my mood.

d. Where are you? *It depends on what the weather is like.

e. Do you have tea or coffee? *It depends on what the weather is like.

f. Did John steal? *It depends on his mood.

g. Do you have tea or coffee? It depends on whether I have some in this box.

h. Did you steal? It depends on what you consider to be stealing.

i. Did John steal? According to the law, it depends on his mood.

In these examples, you can see that wh-questions, alternative questions and polar questions can provide suitable antecedents for anaphoric pronouns. Questions which ask about the future or one's preferences such as the one's in (2.2a-c) seem to almost always be suitable antecedents for anaphoric expressions. Factual questions about the present when taken as antecedents seem to create infelicitous dependence statements as seen in (2.2d-f). However, not all factual questions about the present create infelicitous dependence statements as shown in (2.2g-i). The difference between the felicity of these statements can be explained by means of a modal base. The modal base is important as the truth or falsity of a dependence statement is often determined by the relevant modal base. In the case of the infelicitous dependence statements, the default modal base specified seems to only include worlds where the true answer to the subject question holds. This is the case as the object questions in (2.2d-f) are unrelated to the subject questions and thus don't invoke any hypothetical worlds where the true answer to the factual subject question differs. Thus, the contextually relevant worlds (i.e. modal base) of these dependence statements will only include worlds which are consistent with the true answer to the subject question. Because of this, these statements denote trivial dependencies which we showed in the previous chapter are false under the interpretation of dependence by Theiler et al. Thus, these statements are deemed infelicitous as they denote trivial dependencies.

The reason why statements (2.2g-i) are felicitous despite having the same questions is that the modal base of these dependence statements can include worlds which are inconsistent with the true answer to the subject question. For example, in (2.2g), the modal base will include hypothetical worlds which have tea or coffee in the box. This differs from the modal base of (2.2e) which should not include these worlds by default as the object question does not invoke any relevant hypothetical worlds. How the variation of the modal base affects the felicity of a dependence statement is more evident in (2.2i) where the modal base being explicitly stated allows the dependence statement to be felicitous. By specifying the phrase 'according to the law', the specified modal base will include worlds where the answer to the question 'Did John steal?' varies based on his mood. Thus, we will not have a trivial dependency as there will be worlds in the modal base that differ on the answer to the question: 'Did John steal?'.

Another feature of dependence verbs is that both arguments are suitable antecedents for anaphoric pronouns. Consider the following dialogue:

(2.3) A: Whether we go to the beach depends on the weather.
B: It also depends on whether the bus is running.
A: It depends on that too.

In this example, both A and B use 'it' to refer to the phrase 'whether we go to the beach'. Additionally, A uses 'that' to refer to 'whether the bus is running'. This is not too surprising given our analysis of dependence verbs as many other question embedding verbs such as 'know' also allow for their arguments to be antecedents for anaphoric expressions when the question embedding verb is repeated as seen in (2.4).

(2.4) A: I know what the weather is like.
B: I know that too.

Whole dependence statements can also be taken as suitable antecedents for anaphoric expressions. Consider the following dialogue:

(2.5)
a. A: Whether we go to the beach depends on the weather.
b. B: That is true.
c. B: *I asked that.

Here 'that' in (2.5b) refers to the entire dependence statement in (2.5a). Additionally, in this example, neither the subject nor the object are available antecedents for 'that' as seem in (2.5c). This is unlike the previous example where both the subject and object question are suitable antecedents for both 'it' and 'that'. It seems that the arguments of a dependence clause are only suitable antecedents for anaphoric expressions when a dependence verb is repeated. Additionally, if we use a verb that only takes questions as arguments, no suitable antecedent can be found for 'that' as neither question is available as an antecedent and dependence statements are not questions.

Additionally, certain declaratives can can also act as antecedents for pronominal subjects or objects in dependence statements. As seen in (2.6a), (2.6b), and (2.6c) various modal declaratives, disjunctions and existential statements are acceptable antecedents for anaphoric pronouns. The modals which can be taken as antecedents by anaphoric pronouns seem to be those that introduce possibilities. Another thing to note is that although these declarative modalities are available as antecedents for anaphoric subjects or objects of dependence statements, they cannot be featured directly as subject or object in a dependence statement as seen in (2.6g). Further, not all modals can be taken as declaratives as seen in (2.6d). The modals that are not permissible seem to be those that introduce necessity or temporal universality. Additionally, plain declaratives are not permissible as anaphora as seen in

(2.6e).

(2.6) a. We could go to the beach. It depends on the weather.

b. You may go to the beach or the theater. It depends on what the weather is like.

c. Sometimes I like to go to the beach. It depends on what the weather is like.

d. *We must go to the beach. It depends on what the weather is like.

e. *We are going to the beach. It depends on what the weather is like.

f. *I always like to go to the beach. It depends on what the weather is like.

g. *That we could go to the beach depends on what the weather is like.

This data seems to go against the view that dependence verbs are question embedding verbs as *depends on* seems to be taking declarative arguments. However, in inquisitive semantics, disjunctions and existentials are interpreted as raising multiple alternatives. Thus, inquisitive semantics makes the correct predictions here as in (2.6a-c) the antecedent for the anaphoric expression is considered to be inquisitive. Thus, given the analysis given by Theiler et al. (2019) and a suitable account of question anaphora (2.6b-c) are predicted to be felicitous despite their status as declaratives. Further, the first sentences in (2.6d-f) do not raise alternatives in inquisitive semantics. This is the case because the interpretation of these sentences do not involve disjunctions or existential statements. In (2.6f), the adverb 'always' makes the dependence statement infelicitous as it forces an interpretation which quantifies over all possible times. Thus, the subject proposition does not depend on the object question as it will only contain one alternative. In this thesis, we will not discuss the interaction between dependence clauses and modalized statements further.

As inquisitive semantics provides a uniform account of propositions, we can assume that propositional anaphora like 'that' and 'it' are the same type regardless of whether they refer to questions or declaratives. Thus, we can have a uniform account for anaphora which take questions and declaratives as antecedents. This is another benefit that inquisitive semantics has over other systems. Systems which provide consider interrogatives and declaratives to be a different type must represent anaphora which refer to them separately. For example, Snider (2017) provides an account where propositional anaphora can only refer to declaratives. This is the case as he considers questions to be interpreted as sets of propositions. The treatment of questions as propositions provides a superior analysis for propositional anaphora. Consider the following example:

(2.7) a. Who went to the beach? Alice didn't reveal that to me.

b. I went to the beach. Alice didn't reveal that to me.

If we do not consider interrogatives to be propositions, the verb 'reveal' in (2.7a) and (2.7b) must either be different types or we must suppose some sort of type shifting operator on 'that' which allows for it to semantically compose with 'reveal'. This representation of propositional anaphora leads to a more complex interpretation of verbs which can feature both interrogative and declarative arguments. Thus, the uniform account of propositions in inquisitive semantics provides a benefit when representing these types of anaphoric expressions.

In this section, we have explained how dependence statements featuring anaphoric expressions are suitable given an inquisitive semantics framework. We have also explained how the infelicity of certain dependence statements can be explained by means of a modal base. However, we have not shown how to formally represent anaphoric expressions which refer to questions or dependence statements. In chapter 4, we will provide an account for how anaphora which reference questions can be represented in a dynamic inquisitive framework.

2.3 Nominals

In this section we will see that dependence verbs can associate various types of nominals in either subject or object position. The type of nominals ranges from determiner phrases, quantified noun phrases and names.

(2.8)
a. Whether we can order pizza depends on John.
b. Whether we can go to the beach depends on the weather.
c. Whether we can go to the beach depends on someone.
d. John's mood depends on the weather.
e. My decision to go to the store depends on whether we can avoid the rain.
f. Whether we can take the train depends on a few things.
g. Whether we can take the train depends on two things.

Note that these nominals don't appear to be interpreted referentially as is the case in (2.9) which features a different reading of the *depends on*. That reading of *depends on* relates to needing support which does not seem to be the same reading as the occurrences of *depends on* in (2.8).

(2.9) John depends on Mary for financial assistance.

Instead, the reading of *depends on* in (2.8a-g) seems to be the relationship of dependence between questions that we have previously been analysing. This is the case as in some of the examples (2.8), a question still appears in subject or object position. Additionally, the occurrences of nominals in these statements seem to behave like questions. For example, a natural paraphrase for (2.8a) seems to be: 'Whether we can order pizza depends on whether John wants to order pizza'. Thus, these cases seem to be examples of concealed questions (CQ). A concealed question (CQ) is a determiner phrase which is interpreted as a question. This phenomenon occurs when a noun phrase is the complement of a question-embedding verb. This phenomenon has been studied by Heim (1979) and Frana (2006) about several different verbs such as 'knows' and 'found out'. The fact that these nominals are interpreted as questions helps motivate the inquisitive semantics analysis of *depends on* and dependence statements. In chapter 3, we will discuss the phenomena of concealed questions further and how they can be incorporated into inquisitive semantics.

Not all the nominals that we have presented can be analysed as concealed questions. Consider the objects in (2.8f-g) which do not seem to have natural paraphrase as questions. If we analyse the object of (2.8g) as a concealed question we would paraphrase it as something 'Whether we can take the train depends on what two things are'. Interpreting the object of (2.8g) as a question about 'two things' does not seem suitable. Instead, it seems that 'two things' is a cataphor which allows for the introduction of two questions that are bound to 'two things'. For example, after the utterance of (2.8g), one could state 'the weather and the time of day' which are the two concealed questions that 'two things' is referencing. Thus, when someone makes the utterance in (2.8g), they are stating that there are two questions which taking the train depends on. The expression 'two things' can be analysed through a dynamic lens as introducing a discourse referent. In chapter 4, we will present a dynamic inquisitive framework which will show formally how to represent expressions like 'two things'.

2.4 Dependence and free choice

Another peculiar feature of dependence statements is its relationship to the free choice inference. The free choice inference is a natural language phenomenon where a disjunction which contains a modal operator receives a conjunctive interpretation. The following example illustrates a case of the free choice inference.

(2.10)
a. You may go to the beach or to the cinema.
b. $\rightsquigarrow$ You may go to the beach and to the cinema.
c. $\Diamond(P \vee Q) \rightsquigarrow \Diamond P \wedge \Diamond Q$

Typically, free choice statements which feature disjunctions like (2.10a) allow one to infer a conjoined free choice statement like (2.10b). Whenever a dependence statement follows a free choice disjunction, the free choice inference is impermissible.

(2.11)
a. You may go to the beach or the cinema. It depends on the weather
b. $\not\rightsquigarrow$ You may go to the beach and you may go to the cinema.

Our analysis of dependence predicts that the free choice inference is not permissible in cases where a dependence statements features an anaphoric expression which takes a disjunctive free choice statement as an antecedent. This is the case as *depends on* can only take inquisitive propositions as arguments. Thus, we must assume that the disjunction in (2.11a) takes a wide-scope configuration otherwise it would not be able to be a suitable antecedent for the 'it' in the dependence statement. This causes the free choice inference to fail as a wide-scope inquisitive disjunction of modals does not allow for the free-choice inference. This position coincides with accounts by Fusco (2019) and Aloni (2007) that argue that, in most cases, given a wide-scope configuration of free choice the inference is cancel-able or not permissible.

The relationship between free choice and dependence has also been studied by Kaufmann (2016). She argues that free choice is dependence on preferences. Since free choice for Kaufmann is a form of dependence, specifying a further dependence on other things blocks the free choice inference as it is no longer only one's preference that matters given a certain choice. Her analysis of dependence mirrors that of Ciardelli (2018) and Theiler et al. (2019) with a few differences as her analysis is not in an inquisitive semantics framework. Like the analyses we have provided, she views dependence as a relationship between questions and their alternatives. Thus, her theory makes many of the same predictions regarding dependence statements. The main point regarding dependence that she makes is that free choice should be interpreted as a form of dependence on preferences. This means that to have free choice to do something means that one's choice to do that thing depends on one's preference to do it. Thus, the inference in (2.10) is permissible according to Kaufmann as the referent of 'you' has permission to do either disjunct as they prefer. The paraphrase of free choice sentences as dependence is given in (2.12).

(2.12)

If you have an effective preference for going to the cinema, you may go to the cinema, and if you have an effective preference for going to the beach, you may go to the beach.

This way of representing free choice is problematic. It does not seem that in all cases of free choice, one's preference determines their ability to do that thing. Consider the example in (2.13a) which would be paraphrased as (2.13b) given that free choice is a form of dependence on preference. Given this interpretation of (2.13a), one would not conclude that the person may do what they do not prefer as they do not have a preference for doing so. However, this would mean that the free choice inference is not permissible in this case. This seems strange as the individual giving permission to the agent seems to be giving permission regardless of the preference of the agent. Thus, it seem that free choice as dependence is problematic.

(2.13)
a. You may do what you prefer or what you do not prefer.
b. If you have a preference for doing what you prefer, you may do what you prefer and if you have a preference for doing what you do not prefer, you may do what you do not prefer.

Thus, we will adopt a position that free choice is not a form of dependence. Instead, we will argue that free choice cancellation in the case of dependence statements is caused by an inquisitive wide-scope configuration of the free choice statement which does not allow for the free choice inference. This analysis of free choice makes less assumptions regarding its relationship with dependence while still predicting free choice cancellation in cases where a dependence statement follows a free choice statement.

Chapter 3

Dependence and Concealed Questions

In this chapter, we will cover how to incorporate concealed questions in an inquisitive semantics framework which will allow us to represent dependence statements that contain concealed questions. We will begin by presenting several phenomena which need to be properly represented in an account of concealed questions. Then, we will present an analysis of concealed questions by Aloni and Roelofsen (2011) that makes use of conceptual covers. We will motivate the need for conceptual covers by providing several examples for its use in question semantics and dependence statements. Finally, we will present a typed inquisitive semantics with conceptual covers. Using this framework, we will define a semantics for a small fragment of the English language that will allow us to represent dependence statements that feature concealed questions.

3.1 Concealed Questions

Concealed questions (CQ) are determiner phrases (DP) that can be interpreted as questions. This phenomena occurs when a noun phrase (NP) is the complement of a question-embedding verb such as 'knows'. The italicized NP in the following example is a case of a CQ.

(3.1) Alice knows *the capital of Canada*
$\approx$ Alice knows what the capital of Canada is

In (3.1), we see that 'the capital of Canada' can be naturally paraphrased as an identity question as shown in (3.1). Additionally, 'the capital of Canada' seems to behave semantically as a question as the verb 'knows' can receive its question embedding reading. Other types of DPs can also be concealed questions. It has been shown by Frana (2006) and Heim (1979) that indefinite and quantified NPs can also be CQs.

(3.2) John knows *a doctor who could help you* (Frana 2006)

(3.3) John knows *every phone number* (Heim 1979)

In an earlier chapter, we showed that dependence statements may also feature CQs. It is not surprising that DPs in dependence statements behave as CQs as dependence verbs are question embedding verbs which often feature CQs. However, one key difference is that dependence statements allow for proper nouns such as 'John' to also be interpreted as questions. This does not seem to be the case for other question embedding verbs. Usually for question embedding verbs, occurrences of a proper noun as an argument forces a different reading of the verb as shown in (3.4a). These readings evaluate the proper nouns as referential and block the interpretation of the object noun as a CQ.

(3.4) a. John knows Fred

b. John depends on Mary

However, unlike other question embedding verbs, *depends on* can feature proper nouns that are interpreted as CQs. As we noted in an earlier chapter, *depends on* also has a referential reading. This reading is given when both the subject and object arguments are interpreted referentially as in (3.4b). For this reason, it could be possible that, like other question embedding verbs that have different readings, *depends on* could only be interpreted referentially when taking a proper noun. However, this is not the case as *depends on* can take proper nouns as CQs. This is most likely because both arguments of *depends on*, given the question dependence reading of the verb, must be questions which is unlike verbs like 'know' which feature at least one nominal argument. Thus, when one of the arguments of *depends on* is clearly a question, *depends on* is interpreted as a question embedding verb and interprets proper nouns as CQs. This can be seen in the example in (3.5) where the question 'Whether we can order pizza' forces the question dependency reading of *depends on* and thus 'John' is interpreted as a concealed question. It can even be the case that the subject and object of *depends on* are noun phrases and yet the verb is interpreted as question embedding as is shown in (3.6a). However, this seems to occur only when a referential reading of *depends on* is not possible. In (3.6a) the referential reading is not possible as a decision to go the store cannot depend on anything given the referential reading of *depends on*.

(3.5) a. Whether we can order pizza depends on John

$\approx$ b. Whether we can order pizza depends on whether John wants to order pizza

(3.6) a. My decision to go the store depends on John

$\approx$ b. What my decision to go the store is depends on whether John wants to go

Another thing to note is that many types of questions can be generated by proper nouns like 'John'. In (3.5a), 'John' is interpreted as a question about a contextually relevant property about 'John'. In this case, as ordering pizza is contextually relevant the CQ could be paraphrased as being about his desire to order pizza. However, one can also imagine a scenario where the ordering of pizza depends on whether John arrives on time. In this case, the paraphrase of (3.5a) would not be (3.5b). Instead, 'John' should be interpreted as a question about arriving on time. Thus, the interpretation of CQs in dependence statements seems to be contextually determined. This means that a system for CQs would need to be able to treat NPs as questions about the entity with regards to some contextually relevant property.

This ambiguity with CQs in dependence statements is unsurprising as other ambiguities with CQs have been presented in the past. It has been noted by Heim (1979) that quantified CQs contain an ambiguity between what is called a predicational reading and a specificational reading. Consider the example in (3.3) repeated here as (3.7). The predicational reading of (3.7) is (3.8a) and the specificational reading is (3.8b).

(3.7) John knows *every phone number* (Heim 1979)

(3.8) a. John knows of every phone number whether it is a phone number

b. John knows that Alice's phone number is 12, that Bill's phone number is 43, etc.

This ambiguity can also occur for CQs in dependence statements. Consider the example given in (3.9) where the predicational reading is given in (3.10a) and the specificational reading is given in (3.10b). Thus, concealed questions in dependence statements may also feature this ambiguity.

(3.9) What we can order depends on everyone here

(3.10) $\approx$ a. What we can order depends on what everyone here wants

$\approx$ b. What we can order depends on who everyone here is

Another ambiguity observed by Heim (1979) concerns CQs containing CQs (CCQs). Statements containing CCQs are ambiguous between two readings: Reading A and Reading B. Consider the following sentence:

(3.11) John knows the capital that Fred knows

This sentence has the following two interpretations.

(3.12)

Reading A: There is exactly one country x such that Fred can name x's capital;

and John can name x's capital as well.

(3.13)

Reading B: John knows which country x is such that Fred can name the capital of x.

(although John may not be able to name the capital of x)

If we assume that Fred knows that the capital of Canada is Ottawa, then, under Reading A, it follows from (3.11) that John also knows that the capital of Canada is Ottawa. Under Reading B, no such entailment follows from (3.11). Instead, we can only conclude that John knows Canada as the country which Fred knows the capital of. CCQs can also be featured in dependence statements. Since dependence verbs can embed questions, they can embed questions about CCQs such as in (3.14).

(3.14) It depends on whether John knows the capital that Fred knows

The sentence in (3.14) is also ambiguous between Reading A and Reading B. Under Reading A, 'it' depends on a polar question about Reading A of (3.11). Under Reading B, the dependence relation is between 'it' and a polar questions about Reading B. These two questions differ in that they may have differing alternatives given a certain context. This is the case as in some worlds, Reading A of (3.11) may hold but not Reading B. Thus, to represent dependence statements, we require an account of CQs which allows us to interpret this ambiguity.

There are further ambiguities related to concealed questions. Some of these ambiguities will be presented later. For now, we will move onto presenting an overview of the different approaches to representing concealed questions in the literature.

3.2 Concealed Questions: Overview

There are many approaches on how to represent CQs. According to Aloni and Roelofsen (2011), these approaches tend to differ along two dimensions: *type* and *perspective*. Theories that differ along the *type* dimension assign differing semantic types to CQs. Some of these theories represent CQs formally as propositions while other represent CQs as properties, individual concepts or sets of propositions. Theories that differ in the *perspective* dimension either argue that the interpretation of a CQ depends on a specific perspective on the individuals of the domain of discourse or not.

We will assume an approach to CQs by Aloni and Roelofsen (2011) that assumes that the type of a CQ is that of a proposition and that the interpretation of a CQ depends on a specific perspective. However, since we are adopting an inquisitive approach to questions, we will assume in our framework that CQs denote the type of sets of propositions which are simply inq-propositions. This change will not cause too many differences in the analysis of CQs as inquisitive semantics analyses both declarative and interrogative propositions as the exact same type of semantic object.

We will now briefly present the benefit associated with assigning CQs the same type as propositions. Then, we will show that an inquisitive semantics analysis also provides the same benefits due to the uniform analysis of propositions. CQs can be coordinated with both interrogative and declarative complement clauses as in (3.15) and (3.16a-b).

(3.15) They revealed the winner of the contest and that the president of the association would hand the prize in person. (Aloni and Roelofsen, 2011)

(3.16) a. Whether John is coming and the time of arrival depends on the weather.

(3.17) b. Whether John is coming depends on whether Mary is coming and the weather.

Interpreting CQs as propositions allows us to easily account for these coordination facts. In (3.15), we can easily see that the treatment of CQs as propositions allows for proper coordination as declarative complements are typically considered to represent propositions. Interrogative complements can also be taken to denote propositions as done by Groenendijk and Stokhof (1984) and Aloni (2008). Thus, the coordination between the two clauses can be easily accounted for. This serves as one of the main benefits associated with analyzing CQs as propositions. Given this type of analysis no extra stipulations must be made regarding the coordination of concealed questions with declarative and interrogative propositions.

As we are working in an inquisitive semantics framework, the type of interrogative and declarative clauses are the same, but instead of denoting the type of propositions they are lifted to inq-propositions. Thus, in an inquisitive semantics framework CQs will be interpreted as inq-propositions. Again, this does not lead to any difficulties with coordination as declarative and interrogative clauses will be of the same type as concealed questions. Other theories on concealed questions that assign concealed questions the type of sets of propositions must stipulate type shifting operators to account for coordination facts. This argument is provided against the analysis of CQs as sets of propositions, however it is clear that this same argument does not hold weight against an inquisitive interpretation of propositions due to the uniform analysis of propositions.

The benefit associated with adopting a perspective based approach is that it can better represent concealed questions which are semantically identity questions. We will show this by means of an example.

Example 2. Consider the following scenario: There are two cards face down on the table the Ace of Hearts and the Ace of Spades. John is playing a game where you pick a card on the table and the dealer tells you if you picked the winning card. The winning card changes between rounds and can either be the Ace of Hearts or the Ace of Spades. The winning card is also revealed after the card is picked. John plays one round and picks the card on the left and Mary, who is watching, asks the question: Did John win?. The dealer responds with the following statement:

(3.18) Whether John won depends on the winning card.

The question is whether this sentence is true in the given scenario. On the one hand, the statement is true as whether John won depends on whether or not the card on the left is the winning card. On the other hand, the statement is false as whether John won does not depend on whether the Ace of Spades or the Ace of Hearts is the winning card. This is because simply from knowing that the Ace of Spades or the Ace of Hearts is the winning card, one cannot determine whether or not John won as one still needs to know whether that card was picked by John. As CQs can feature this type of ambiguity in dependence statements a perspectival approach is beneficial as it can account for such ambiguities. Therefore, we will adopt an approach that evaluates concealed questions based on a certain perspective. Later, we will also show that a perspectival approach to questions also allows inquisitive semantics to represent certain types of identity questions.

3.3 Concealed Questions using conceptual covers

Aloni and Roelofsen (2011) presented a framework which was able to account for the ambiguities listed in the previous section. As CQs in dependence statements behave in the same way that was described by Aloni and Roelofsen (2011), we will assume the basis of their analysis of CQs. However, instead of using partition theory, we will implement their analysis of CQs in an inquisitive semantics framework. Thus, we will adopt an approach that evaluates questions from a certain perspective. Their system makes use of conceptual covers to evaluate CQs based on a perspective. Thus, to adopt their approach to CQs we will have to incorporate conceptual covers into a typed inquisitive semantics. In this section, we will begin by introducing the theory of conceptual covers (Aloni, 2001) and present several arguments for why conceptual covers are useful for dependence statements and questions in inquisitive semantics. Then, we will present their theory of concealed questions which makes use of conceptual covers.

A conceptual cover is a way of formalizing ways of identifying objects using concepts. More formally, a conceptual cover is a set of individual concepts which satisfy certain constraints.

An individual concept c is a total function from a set of worlds W to a set of individuals D. An individual concept represents a way of identifying an object given a certain perspective. The full definition of a conceptual cover is given below:

Definition 3.3.1 (Conceptual Cover)**.** A conceptual cover CC based on a $\langle W, D\rangle$ is a set of functions $W \to D$ such that:

$$\forall w \in W : \forall d \in D : \exists! c \in CC : c(w) = d$$

We will illustrate the use of a conceptual cover using the previous example about cards. To formalize this scenario with conceptual covers we will use a model with four worlds $\{w_1, w_2, w_3, w_4\}$ and a domain consisting of two individuals: d_1 and d_2. In w_1, d_1 is the card on the left, the Ace of Hearts, and the winning card while d_2 is the card on the right and the Ace of Hearts and the losing card. In w_2, d_1 is the card on the left, the Ace of Hearts and the losing card while d_2 is the card on the right, the Ace of Spades and the winning card. In w_3, d_1 is the card on the left, the Ace of Spades, and the winning card while d_2 is the card on the right and the Ace of Hearts and the losing card. In w_4, d_1 is the card on the left, the Ace of Spades and the losing card while d_2 is the card on the right, the Ace of Hearts and the winning card. We will define three conceptual covers over this model: A which identifies cards by their position, B which identifies cards by their suit, and C which identifies cards as the winning card or the losing card. Visually these concepts are represented below. (The plus denotes the winning card)

$$\begin{array}{lll} w_1 \mapsto \heartsuit^+ & \spadesuit \\ w_2 \mapsto \heartsuit & \spadesuit^+ \\ w_3 \mapsto \spadesuit^+ & \heartsuit \\ w_4 \mapsto \spadesuit & \heartsuit^+ \end{array}$$

$A =$ {The card on the left, the card on the right}

$B =$ {Ace of Spades, Ace of Hearts}

$C =$ {the winning card, the losing card}

$D =$ {Ace of Spades, The card on the left}

Sets $A - C$ of individual concepts constitute proper conceptual covers as for each world the individual concepts in the cover denote a single unique object. However, D does not constitute a proper cover as in w_3 and w_4 both the Ace of Spades and the card on the left denote the same object d_1. Thus, there is not a unique concept in D that denotes d_1. Later when we have more of the framework worked out, we will represent how the dependence statement 'Whether John won depends on the winning card' comes out true based on a certain perspective and false based on another perspective. For now, we will move onto Aloni and Roelofsen's account of concealed questions.

Now that we have provided a basic account of conceptual covers, we will present the analysis of concealed questions by Aloni and Roelofsen (2011). We begin with a first-order logic enriched with an ι operator and a question operator '?'. Additionally, the language includes a set N of conceptual cover indices. A model in this system is a quadruple (W, D, I, C)

where W is a set of possible world, D is a set of individuals, I an interpretation function that is world dependent, and C a set of conceptual covers based on (W, D). Expressions are evaluated relative to a model, a world in that model, an assignment function and a contextual perspective which is a function from N to C. In Aloni and Roelofsen's framework, variables do not range over individuals, instead they range over the elements of some pragmatically determined conceptual cover. Additionally, variables in this language are indexed by elements of N[1]. The conceptual cover that a variable x_n ranges over is determined by a contextual perspective $\wp$.

Definition 3.3.2 (Contextual perspective)**.** Let $\mathcal{M} = (W, D, I, C)$ be a model and N be the set of indices in $\mathcal{L}$. A contextual perspective $\wp$ in $\mathcal{M}$ is a function from N to C.

Thus, a variable x_n is assigned a value in two steps, first $\wp(n)$ provides a conceptual cover. Then, g maps x to some element of $\wp(n)$. Thus, the extension of a variable is:

(3.19) $$[\![x_n]\!]_{M,w,g_\wp} = g_\wp(x_n)(w)$$

Variables denote individuals non-rigidly. This means that a variable can denote different individuals at different worlds. Aloni and Roelofsen's analysis of concealed questions also makes use of an ι operator which is defined as follows:

Definition 3.3.3 (The ι operator)**.**

- If there is a unique $c \in \wp(n)$ such that $[\![\phi]\!]_{M,w,g_\wp} = 1$, then $[\![\iota x \phi]\!]_{M,w,g_\wp} = c(w)$
- Otherwise $[\![\iota x \phi]\!]_{M,w,g_\wp}$ is undefined.

Another important definition for their analysis of concealed questions is the definition of a question under cover. They analyze questions as their possible exhaustive answers as in Groenendijk and Stokhof (1984). However, the key difference is that questions involve quantification over concepts rather than individuals. Thus, the denotation of a question under cover is given in the following:

(3.20) $$[\![?x_n\phi]\!]_{M,w,g_\wp} = \{v | \forall c \in \wp(n) : [\![\phi]\!]_{M,w,g_\wp[x/c]} = [\![\phi]\!]_{M,w,g_\wp[x/c]}\}$$

Additionally, the language is extended with a knowledge operator K_a for every agent a. The model in our extended language is a quintuple (W, D, I, C, E) where W, D, I and C are the same as above, and E is a function mapping individual world pairs (a, w) into subsets of W. $E(a, w)$ is meant to represent the epistemic state of a at w. We can now provide the meaning of $K_a(?x_n\phi)$.

Definition 3.3.4 (Knowledge Ascription)**.**

$$[\![K_a(?x_n\phi)]\!]_{M,w,g_\wp} = 1 \text{ iff } E(a, w) \subseteq [\![?x_n\phi]\!]_{M,w,g_\wp}$$

The exact details of how each of these operators work can be found in Aloni and Roelofsen (2011). We will now present the analysis of concealed questions in this framework. The basis

[1]The original paper by Aloni and Roelofsen (2011) implements conceptual covers without indices. We have chosen to choose an indexed implementation as it is more perspicuous.

of their analysis is a type-shifting operator $\uparrow$ which transforms an entity α into the question $?x_n.P(\alpha)$.

(3.21) $$\uparrow_{(n,P)} \alpha := ?x_n.P(\alpha)$$

This type shifting operation has two parameters: n which is a contextual cover index which when evaluated by a perspective $\wp$ provides a conceptual cover and P is a contextually determined property. These two parameters allow this analysis to account for the numerous ambiguities that we have presented.

We will now show how this operator allows for the representation of the Heim ambiguity. Recall earlier the ambiguity between Reading A and Reading B of (3.22). This ambiguity is analysed as a de-re/de-dicto ambiguity by Aloni and Roelofsen (2011). In this example the P parameter in (P, n) is set to the identity property (i.e $\lambda x.x = y_n$).

(3.22) John knows the capital that Fred knows.

$\exists x(x = \iota y.(capital(y) \wedge K_f(\uparrow_{(1,P)} y)) \wedge K_j(\uparrow_{(2,P)} x))$ (Reading A)

$K_j(\uparrow_{(3,P)} \iota y.(capital(y) \wedge K_f(\uparrow_{(4,P)} y))$ (Reading B)

The intended readings are given when the following covers are assumed for the variables in (3.22).

(3.23) a. $x, y, 3 \rightarrow$ {the capital of Germany, the capital of Italy,...}

(3.24) a. $1, 2, 4 \rightarrow$ {Berlin, Rome,...}

Reading A of the example says that there is a unique capital that Fred knows by name, and John knows that same capital by name. Reading B of the example says that John knows the country which Fred knows the capital of and knows that Fred knows this capital. There are other variations of this ambiguity in the literature which can also be handled by this analysis.

The P parameter allows Aloni and Roelofsen's account to deal with a number of ambiguities. The first ambiguity that it can account for is the specificational and predicational readings of QCQs. Dependence statements which feature proper nouns can also be accounted for as the variation of P accounts for the different contextually relevant property about the proper noun. We will show how this is the case by means of an example. Earlier we stated that (3.25) can be paraphrased as either (3.25a) or (3.25b). The variation between interpretations will vary based on the P parameter. In (3.25a), the type shifting operator will be indexed with P_1: the property of 'wanting to order a pizza'. In (3.25b), the type shifting operator will be indexed with P_2: the property of 'arriving on time'.

(3.25) Whether we can order pizza depends on John.

$\approx$ a. Whether we can order pizza depends on whether John wants to order pizza.

$\approx$ b. Whether we can order pizza depends on whether John arrives on time.

As we can see here Aloni and Roelofsen's account can deal with many of the ambiguities relating to concealed questions. There are other ambiguities associated with concealed questions. However we will not address these ambiguities as they are not within the scope of this paper. A further benefit with adopting Aloni's account of conceptual covers is that it is extremely useful in the context of questions as it can deal with several problems associated with questions which involve rigid designators. For example, if someone asks the question in (3.26a). In inquisitive semantics, this question is usually represented as either $\exists xx = j$ or $?\exists xx = j$. These questions however expresses a trivial issue over every information state as $j = j$ in every possible world. However, this question can be represented given that we can refer to individual concepts. Further, questions like (3.26b) also cannot be represented in a standard inquisitive semantics. Thus, as shown by Maden (2019), conceptual covers in inquisitive semantics allows for the representation of more questions.

(3.26)
a. Who is John?
b. Who is who?

We will now move onto the analysis of concealed questions in a typed inquisitive system.

3.4 Concealed Questions in typed inquisitive semantics

In this section we will cover how to incorporate concealed questions into a typed inquisitive semantics. We will do this in three steps. First, we will enrich typed inquisitive semantics with conceptual covers. We will then define a small fragment of English which will show how a typed inquisitive semantics with conceptual covers can account for a variety of questions and dependence statements. Then, we will add a type shifting operator that changes entities into questions. Finally, we will provide several examples of concealed questions in dependence statements.

We will now extend a typed inquisitive semantics with conceptual covers. This has been done earlier in a first-order setting by Maden (2019). Consider a language $\mathcal{L}$ of intensional type theory. Recall that in chapter 1.8 that we provided the syntax and semantics of an intensional type theory. $\mathcal{L}_{CC}$ is derived from $\mathcal{L}$ by making several changes to the syntax and language of $\mathcal{L}$. First we add a set of conceptual covers indices $N = \{0, 1, 2, ...\}$ to $\mathcal{L}_{CC}$. Then, we will make several changes to the vocabulary. Recall earlier that in $\mathcal{L}$ the following was the clause for variables:

- 1. For every type σ, an infinite set of VAR_σ of type σ.

In $\mathcal{L}_{CC}$, the following clause is made for variables

- 1a. For every type $\sigma \neq \langle s, e \rangle$, an infinite set of VAR_σ of type σ.
- 1b. For type $\langle s, e \rangle$ and set N of conceptual cover indices an infinite set of $VAR_{\langle s,e \rangle}$ of variables of type $\langle s, e \rangle$ indexed by $n \in N$.

Additionally, in the syntax of $\mathcal{L}$ we replace the following clause for quantification:

- 4. If $\phi \in WE_t^{\mathcal{L}}$ and $v \in VAR_\sigma^{\mathcal{L}}$ then $\exists v\phi, \forall v\phi \in WE_t^{\mathcal{L}}$

with the following clauses in $\mathcal{L}_{CC}$:

- 4a. If $\phi \in WE_t^{\mathcal{L}}$ and $v \in VAR_\sigma^{\mathcal{L}}$ where $\sigma \neq \langle s, e \rangle$ then $\exists v\phi, \forall v\phi \in WE_t^{\mathcal{L}}$
- 4b. If $\phi \in WE_t^{\mathcal{L}}$ and $v \in VAR_{\langle s,e \rangle}^{\mathcal{L}}$ and $n \in N$ then $\exists v_n\phi, \forall v_n\phi \in WE_t^{\mathcal{L}}$

A model $\mathcal{M}$ for typed inquisitive semantics with conceptual covers is a tuple (D, W, I, R, C) where D is a domain of individuals, W is a set of worlds, I is an interpretation function, R is a relation on W and C is a set of conceptual covers. Recall that the definition for a conceptual cover is as follows:

Definition 3.4.1 (Conceptual Cover)**.** A conceptual cover CC based on a $\langle W, D\rangle$ is a set of functions $W \to D$ such that:

$$\forall w \in W : \forall d \in D : \exists! c \in CC : c(w) = d$$

We can now build an inquisitive semantics for $\mathcal{L}_{CC}$. Expressions in the language will now also be interpreted with respect to a contextual perspective.

Definition 3.4.2 (Contextual perspective)**.** Let $\mathcal{M} = (W, D, I, R, C)$ be a model for Inq^{λ}_{CC} and N be the set of indices in $\mathcal{L}_{CC}$. A contextual perspective $\wp$ in $\mathcal{M}$ is a function from N to C.

The denotation $[\![\alpha]\!]_{\mathcal{M},g_\wp}$ of an expression α is defined relative to a model $\mathcal{M}$, a perspective based variable assignment $g_\wp$. $g_\wp$ evaluates the contextual cover index of a variable which thus provides a conceptual cover and then maps that variable to some element in that conceptual cover. The semantics remain the same for all expressions other than for quantifiers and variables. In the case of quantifiers, we keep the definitions for quantification over variables of type σ where $\sigma \neq \langle s, e\rangle$. For quantifiers with variables of type $\langle s, e\rangle$ we now add the following clauses:

- If $\phi \in WE^{\mathcal{L}}_t$ and $v_n \in VAR^{\mathcal{L}}_{\langle s,e\rangle}$ then:
  - $[\![\exists v_n \phi]\!]_{\mathcal{M},g_\wp} = 1$ iff for some $c \in \wp(n) : [\![\phi]\!]_{\mathcal{M},g_\wp[x_n/c]}$
  - $[\![\forall v_n \phi]\!]_{\mathcal{M},g_\wp} = 1$ iff for all $c \in \wp(n) : [\![\phi]\!]_{\mathcal{M},g_\wp[x_n/c]}$

For the interpretation of variables we add the following clause:

- If $\alpha \in VAR^{\mathcal{L}}_\sigma$, then $[\![\alpha]\!]_{\mathcal{M},g_\wp} = g_\wp(\alpha)$

This clause will allow the assignment function $g_\wp$ to evaluate the conceptual cover index of a variable. Thus, variables which feature conceptual cover indices will be sent to an individual concept in the correct conceptual cover.

To make the translations we have made more legible, we will employ the use of several inquisitive operators and connectives. The first operator we will introduce is $\boxed{\neg}$ which will represent inquisitive negation in our system.

(3.27) $$\boxed{\neg} = \lambda P_T \lambda p_{\langle s,t\rangle} \forall w \in p : \neg\exists q \in P : q(w)$$

The second operator we will introduce is the interrogative projection operator $\boxed{?}$. This operator is the type-theoretic counterpart of the information-cancelling operator in Inq_B. This operator will allow us to represent polar questions in our system.

(3.28) $$\boxed{?} = \lambda P_T \lambda p_{\langle s,t\rangle} P(p) \vee \boxed{\neg} P(p)$$

The third operator we will introduce is the declarative projection operator $\boxed{!}$. This operator is the type-theoretic counterpart of the issue-cancelling operator in Inq_B.

(3.29)
$$\boxed{!} = \lambda P_T \lambda p_{\langle s,t\rangle} \forall w \in p : \exists q \in P : q(w)$$

As we have only changed the clauses for quantification over individual concepts, we can define these operators in the same way as is done in Theiler (2014) without any problems. This is because these definitions only exploit the use of quantification over worlds and Inq-propositions. For more details on how these operators work see Theiler (2014) where she provides an account of embedded questions.

We derivation of sentences in our language will be done by means of two rules: Functional application, and Predication abstraction.

Definition 3.4.3 (Functional application)**.** If α is a branching node and $\{\beta\gamma\}$ is the set of its daughters, then $Tr(\alpha)$ is defined if $Tr(\beta)$ and $Tr(\gamma)$ are defined and $Tr(\beta)$ is of type $\langle\sigma,\tau\rangle$ and $Tr(\gamma)$ is of type σ. In this case $Tr(\alpha) = Tr(\beta)(Tr(\gamma))$

Definition 3.4.4 (Predicate abstraction)**.** If α is a branching node whose daughters are the movement index λ_i and β, then $Tr(\alpha)$ is defined if $Tr(\beta)$ is defined. In this case, $Tr(\alpha) = \lambda x_i.Tr(\beta)$.

Functional application represents the normal case of semantics composition. It allows us to combine terms together. Predicate abstraction allows us to represent more complex binding relations.

We will now define a small fragment of the English language to show the use of conceptual covers in a typed inquisitive semantics to represent identity questions.

Translations and their Types			
Cat	α	$Tr(\alpha)$	Type
PN	John	$\lambda w_s.j(w)$	$\langle s,e\rangle$
DP	He_n/t_n	$\lambda w.x_n(w)$	$\langle s,e\rangle$
DP	who	$\lambda P_{\langle\langle s,e,\rangle,T\rangle}\lambda p_{\langle s,t\rangle}\boxed{?}\exists x_n P(x_n)(p)$	$\langle\langle\langle s,e\rangle,T\rangle,T\rangle$
TV	is	$\lambda x_{\langle s,e\rangle}\lambda y_{\langle s,e\rangle}\lambda p_{\langle s,t\rangle}\forall w \in p : x(w) = y(w)$	$\langle\langle s,e,\rangle\langle\langle s,e\rangle,T\rangle\rangle$
IV	walks	$\lambda x_{\langle s,e\rangle}\lambda p_{\langle s,t\rangle}\forall w \in p : walks(w)(x_w)$	$\langle\langle s,e\rangle,T\rangle$
TV_T	depends on	$\lambda P'_T\lambda P_T.\lambda p.\forall w \in p : DEP_w(P',P)$	$\langle T,\langle T,T\rangle\rangle$
Neg	not	$\lambda X_T\lambda p_{\langle s,t\rangle}\boxed{\neg}X(p)$	$\langle T,T\rangle$

We will now represent a scenario involving an identity question in our typed system. Consider the following scenario: Mary arrives at a restaurant to have a blind date with John. The host tells her that John has already arrived and to take a seat. There are two tables open, one with John and one with Bill. One of the tables is on the right and one is on the left. Mary asks the host: Who is John?.

We represent this scenario with the following simplified model $\mathcal{M}$. $W = \{w_1, w_2\}$, $D = \{j, b\}, C = \{A, B\}$, $A = \{l, r\}, B = \{j, b\}$. l is the function associated with the concept *the one on the left* and r is the function associated with the concept *the one on the right.* In w_1, John is on the left and in w_2, John is on the right. We get the following translation of the question: 'Who is John?'

$Tr(\text{Who is John?}) = \lambda p.\exists x_n \forall w \in p : j(w) = x_n(w)$

When we evaluate this translation on our model M where the perspective $\wp$ assigns A to n, we get the following interpretation of $\lambda p.\exists x_n \forall w \in p : j(w) = x_n(w)$.

$$[\![\lambda p.\exists x_n \forall w \in p : j(w) = x_n(w)]\!]_{\mathcal{M},g_\wp} = \{\{w_1\}, \{w_2\}\}^{\downarrow}$$

Thus, in our model there are two alternatives in $Tr(\text{Who is John})$. $\{w_1\}$ represents the answer 'John is on the left' and $\{w_2\}$ represents the answer 'John is on the right'.

We will now explain how to represent concealed questions in a typed inquisitive system. The type shifting operator in our system is very similar to the one used by Aloni and Roelofsen. There are, however, two changes that need to be made to properly represent concealed questions. The first change we need to make is to the parameter P. Since we are now dealing with inq-propositions, we need to make sure that the concealed question is the correct type. Thus, the type shifting operator now shifts entities of type $\langle s, e\rangle$ to inquisitive propositions of type T. This is done by changing the type of the parameter P from type $\langle\langle s, e\rangle, t\rangle$ to type $\langle\langle s, e\rangle, T\rangle$. The second change that needs to be made is to the interpretation of a question under cover. This change needs to be made as the previous definition of a question under cover was made with an analysis of questions by Groenendijk and Stokhof (1984) which analysed questions as propositions. As we have moved into a typed inquisitive framework, we must make changes to the definition to accommodate for the differences between how these frameworks interpret questions. Thus, the definition for the type shifting operator and questions under covers is as follows:

(3.30) a. $\uparrow_{(n,P)} \alpha := \boxed{?}(\lambda p.\exists x_n.P(\alpha)(p))$

We will now add the concealed questions rule which will allows us to derive sentences with concealed questions in our language.

Definition 3.4.5 (Concealed Questions rule)**.** If α is a branching node whose daughters are β of type $\langle T, \sigma\rangle$ and γ of type $\langle s, e\rangle$, then $Tr(\alpha)$ is defined if $Tr(\beta)$ and $Tr(\gamma)$ are defined. In this case, $Tr(\alpha) = Tr(\beta)(\uparrow_{(n,P)} Tr(\gamma)) = \boxed{?}(\lambda p.\exists x_n.P(Tr(\gamma)(p))$

We will now repeat the dependence operator defined in chapter 1.

(3.31) $[\![DEP_w(P, P')]\!]_{\mathcal{M},g_\wp} = 1$ iff $\exists f \in alt(P)^{alt(P')}$ such that
(i) $\forall p \subseteq \sigma_w.\forall \alpha \in alt(P').(p \subseteq \alpha \to p \subseteq f(\alpha))$ and
(ii) $\exists \alpha, \alpha' \in alt(P').\alpha \cap \sigma_w \neq \emptyset \wedge \alpha' \cap \sigma_w \neq \emptyset \wedge f(\alpha) \neq f(\alpha'))$

We can now represent a dependence statement which features a concealed question. Consider the following sentence:

(3.32) Whether John walks depends on John

We will first break down the question in the subject position of the dependence statement. 'Whether John walks' is the embedded form of 'Does John walk?'. In our system, we want the question 'Does John walk?' to be represented as a polar question in inquisitive semantics. Thus, we want the interpretation of the sentence to be a set of propositions such that there are two alternatives: the largest classical proposition such that John walks and the largest classical proposition such that John does not walk. The verb 'walk' in our system denotes a function which maps an entity x of type e to the set of all classical propositions where in all worlds of the proposition x walks. Thus, when combined with 'John' to create 'John walks', we get the set of all propositions where John walks. The phrase 'Does John walk?' is given when we combine the polar question making operator with 'John walks'. Thus, the representation of 'Does John walk?' and its transformation 'Whether John walks' is given in (3.33a) which equivalently can be written as (3.33b). For legibility, we will typically use the simplified form in (3.33b).

(3.33) a. $\lambda p.\forall w \in p : walks(w)(j_w) \vee \forall w \in p : (\neg\exists q(q(w) \wedge \forall v \in q : walks(v)(j_w)))$

b. $\boxed{?}(\lambda p.\forall w \in p : walks(w)(j_w))$

We will evaluate this question on a simple model $\mathcal{M}$. $W = \{w_1, w_2, w_3, w_4\}$, $D = \{d_1\}$, $j(w_1) = j(w_2) = j(w_3) = j(w_4) = d_1$

$[\![walks(w_1)(j_w)]\!]_{\mathcal{M},g_\wp} = [\![walks(w_2)(j_w)]\!]_{\mathcal{M},g_\wp} = 1$

As we can see in (3.34), the denotation of 'whether John walks' provides us with an inq-proposition that has two alternatives: one which contains all the states where John is walking and one which contains all the states where John is not walking. This is the exact same inq-proposition represented by $?Wj$, the polar question about John walking, in a first-order system. Thus, we have the desired denotation of 'whether John walks'.

(3.34) $$[\![\boxed{?}(\lambda p.\forall w \in p : walks(w)(j_w))]\!]_{\mathcal{M},g_\wp} = \{\{w_1, w_2\}, \{w_3, w_4\}\}^{\downarrow}$$

Now we will represent the question in the object position of the dependence statement. We do this by applying the type-shifting operator to the individual concept $\lambda w j(w)$. The desired paraphrase of the concealed question 'John' is the polar question 'whether John slept'. Before we can apply the type shifting operator we have to set the n and P parameters. Let's assume that in this context that the P parameter denotes the inq-predicate $\lambda x \lambda p.\forall w \in p : S(w)(x)$ which denotes the property of having slept. As the concealed question is not an identity question the n parameter is unimportant. Thus, $Tr(\uparrow_{n,P} \lambda w.j(w))$ amounts to the following:

(3.35) $$\lambda p.\exists x_n \forall w \in p : sleeps(w)(j_w) \vee \forall w \in p : (\neg\exists q(q(w) \wedge \exists x_n \forall v \in q : sleeps(v)(j_w)))$$

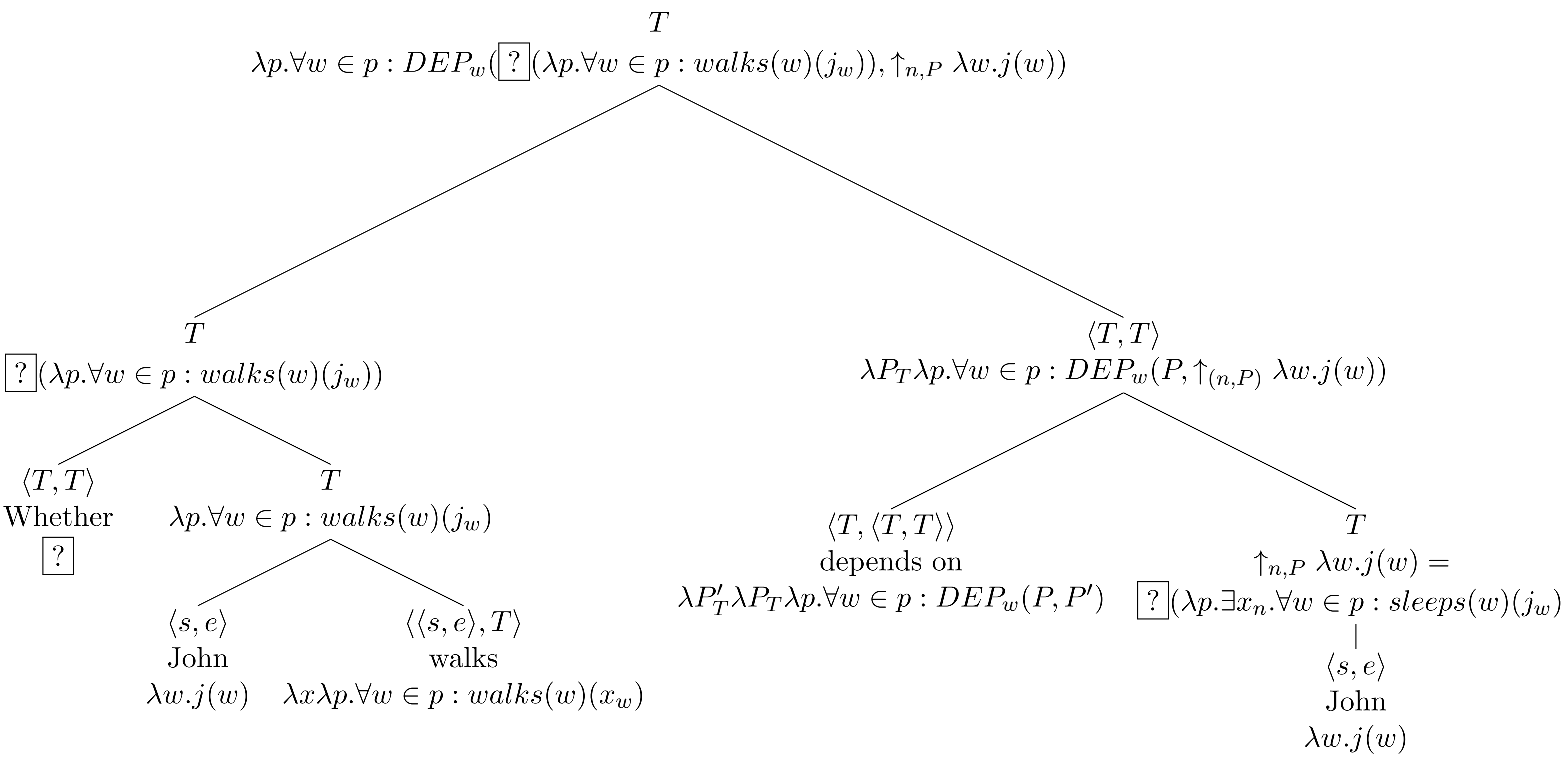
Since there are no occurrences of x_n the existential quantification over x_n can be ignored. This will be the case in any situation where the inquisitive property P does not denote an identity question. Thus, when interpreting (3.35) on the previous model we will get the following:

(3.36) $$[\![\boxed{?}(\lambda p.\forall w \in p : slept(w)(j))]\!]_{\mathcal{M},w,g_\wp} = \{\{w_1, w_3\}, \{w_2, w_4\}\}^{\downarrow}$$

This is the correct interpretation of the concealed question 'John' as it provides an inq-proposition that has two alternatives: one where in all the worlds in the alternative John had slept, and one where in all the worlds in the alternative John didn't sleep. Let α be the statement 'Whether John walks depends on John'. The translation of the dependence statement is as follows:

(3.37) $$Tr(\alpha) = \lambda p.\forall w \in p : DEP_w(\boxed{?}(\lambda p.\forall w \in p : walks(w)(j_w)), \uparrow_{n,P} \lambda w.j(w))$$

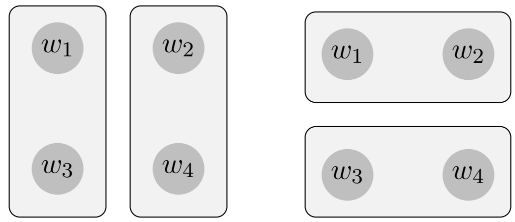
We will show how we get this representation by means of a tree diagram.



To evaluate the dependence statement above, we need to specify a modal base and a world in which we will evaluate the proposition. Let $\sigma_{w_1} = \{w_1, w_4\}$ and $\sigma_{w_2} = \{w_1, w_2, w_3, w_4\}$. Let ϕ be the subject question and ψ be the object question of the dependence statement. Let w_1 and w_2 be the worlds where John walks and let w_1 and w_3 be the worlds where John sleeps. We know that $alt(\phi) = \{\{w_1, w_2\}\{w_3, w_4\}\}$ and $alt(\psi) = \{\{w_1, w_3\}\{w_2, w_4\}\}$. In this example, the dependence statement comes out true when we evaluate based on w_1. This is because we can find a function f that satisfies the dependence conditions. The f in question is a function which maps $\{w_1, w_3\}$ to $\{w_1, w_2\}$ and $\{w_2, w_4\}$ to $\{w_3, w_4\}$. Since the modal base only includes w_1 and w_4, the first condition holds. However, we can see that for the modal base of w_2 it is false as if we define the same function, we will not be able to satisfy the first condition. This is because no matter the function we define there will be a state that is a subset of the alternative of the object question that is not a subset of the function applied to that same alternative. This can be seen given the two diagrams below which highlight the alternatives of ϕ and ψ.

We will now represent the scenario in 3.2 where whether someone won depended on the winning card. Recall that in the scenario, the following phrase was uttered:

(3.38) Whether John won depends on the winning card.



(a) ψ (b) ϕ

The representation of the question in the subject position will be pretty much the same as the representation of 'whether John walks' with the slight modification of replacing 'walks' with 'won'. Thus the representation of the sentence is as follows:

(3.39)
$$\boxed{?}(\lambda p.\forall w \in p : won(w)(j_w))$$

Now we represent the question 'the winning card' in the object position. We have decided to represent the winning card as a individual concept rather than a definite description for simplicity. As it is a concealed question, we apply the type shifting operator for concealed questions to the individual concept $\lambda w.c(w)$. To represent the concealed question from the perspective of location we will set the parameter n such that $\wp(n) = A$ where $A = \{\text{the card on the left, the card on the right}\}$. As we are dealing with an identity question we will set the P parameter to $\lambda y \lambda p \forall w \in p : y(w) = x_n(w)$ the property of being equal to the variable in question. Thus $\uparrow_{n,P} \lambda w.c(w)$ will provide us with the question in (3.40). This question is identical to the identity question 'What is the winning card' over the cover A given that the perspective maps n to A.

(3.40)
$$\begin{aligned}&\lambda p.\exists x_n \forall w \in p : c(w) = x_n(w) \vee \\ &\forall w \in p : (\neg \exists q(q(w) \wedge \exists x_n \forall v \in q : c(w) = x_n(w)))\end{aligned}$$

Now we will formalize the scenario described with a model $\mathcal{M}$. $W = \{w_1, w_2, w_3, w_4\}$ $D = \{d_1, d_2\}$ $C = \{A, B\}$ $A = \{l, r\}$ $B = \{h, s\}$. l is the individual concept associated with *the card on the left*. r is the individual concept associated with *the card on the right*. h is the individual concept associated with the *Ace of Hearts*. s is the individual concept associated with the *Ace of Spades*. The concept associated with each card is demonstrated in the image below: (the plus indicates that the card is the winning card).

$$\begin{array}{lll} w_1 \mapsto & \heartsuit^+ & \spadesuit \\ w_2 \mapsto & \heartsuit & \spadesuit^+ \\ w_3 \mapsto & \spadesuit^+ & \heartsuit \\ w_4 \mapsto & \spadesuit & \heartsuit^+ \end{array}$$

As John has picked the left card, we see that the question: 'Did John win?' has two alternatives $\{w_1, w_3\}$ the worlds where he won and $\{w_2, w_4\}$ the worlds where he did not win. Similarly, the identity question 'what is the winning card?' under the cover of location A has the same two alternatives. Thus, the dependence statements holds given the modal base $\sigma_{w_1} = \{w_1, w_2, w_3, w_4\}$ as there is a function that satisfies both dependence conditions namely the function that maps each alternative to itself.

Now consider the same sentence except $\wp(n) = B$ the cover denoting the suit of the card. In this case we see that the identity question 'what is the winning card' will give us a different partition of worlds. Instead of dividing the worlds based on the position of the card it will divide them based on whether a specific suit is the winning card. Thus, the alternatives of the question will be $\{w_1, w_4\}$ and $\{w_2, w_3\}$

We have shown in this chapter that concealed questions can be represented in dependence statements with the help of conceptual covers and a type-shifting operator. Further, we have shown how to compositionally represent dependence statements that feature concealed questions. We did this by incorporating conceptual covers into a typed inquisitive semantics framework. This allowed us to represent several of the numerous ambiguities associated with concealed questions.

Chapter 4

Propositional anaphora in Dynamic Inquisitive Semantics

The previous chapter covered a typed inquisitive semantics that provided an analysis of dependence statements. This system allowed for the interpretation of dependence statements that included certain nominals. However, this system was unable to represent anaphora and nominals such as ‘two things’. In this chapter, we will present a system that aims to represent this phenomena dynamically.

4.1 Introduction to dynamic semantics

In dynamic semantics (Dekker, 2012), propositional meanings are characterized by their context change potential. The semantic value of sentences in this system will be modeled as functions from input contexts to output contexts. The motivation for this approach comes from the desire to store and retrieve discourse information. The ability to store and retrieve discourse information allows dynamic semantics to represent a number of different linguistic phenomena including anaphoric pronouns, definite noun phrases and presuppositions. Consider the following:

(4.1)
a. A man is walking. He is smiling.
b. ?He is smiling. A man is walking.

In (4.1a), the pronoun ‘he’ is well-behaved as it is preceded by a noun phrase. However, when the pronoun ‘he’ precedes the noun phrase it is odd. In a standard first-order logic, there is no difference between these sentences as conjunction is commutative. However, as the meaning of sentences is characterized by their potential to change contexts, dynamic semantics does not evaluate conjunction as commutative. Instead, the ordering of sentences matters as each sentence has a potential to change contexts. In this particular sentence, ‘A man’ introduces a discourse referent which can thus be referred to by the pronoun ‘he’. In (4.1a), ‘he’ can successfully refer back to the discourse referent introduced by ‘A man’. This is not the case for (4.1b). We can also see in (4.2) that such ordering matters in the case of dependence statements.

(4.2) a. Will you go to the store? It depends on the weather.
b. ?It depends on the weather. Will you go to the store?

Dynamic semantics has also shown to be useful in the analysis of questions. Originally, dynamic systems that dealt with questions relied on the idea that questions induce a partition on the set of worlds. More recently, Dotlačil and Roelofsen (2019); Dotlacil and Roelofsen (2021), and Zhao (2019) have worked on a dynamic inquisitive semantics. This system is the result of incorporating several insights from inquisitive semantics into a dynamic framework. We will make use of a dynamic inquisitive semantics framework to interpret dependence statements with anaphora. However, we will have to make several modifications. This is because, originally, discourse referents in dynamic semantics were used to track entities. As dependence statements are question embedding verbs, the propositional anaphora featured in a dependence statement are not considered entities. Instead, they should be considered questions. Thus, discourse referents in our system will refer to inq-propositions instead of entities. The system we will present will allows us to properly represent expressions which introduce propositional discourse referents. This will allow us to represent dependence statements that feature propositional anaphora.

4.2 Basics of Inq$_{DP}$

Contexts in Dynamic inquisitive semantics for propositional anaphora (Inq$_{DP}$) are a combination of static inquisitive semantics Inq$_B$ and the notion of possibilities. A possibility in Inq$_{DP}$ is a pair $\langle w, g\rangle$ where w is a possible world and g is a function from discourse referents to static inq-propositions. This will allow us to represent anaphoric reference to propositions and the introduction of discourse referents which refer to propositions. A context in Inq$_{DP}$ will be a set of sets of possibilities. We will further define these notions in the coming section.

One thing to note is that the function g maps discourse referents to static inq-propositions rather than dynamic inq-propositions. We have done this because a dynamic inq-proposition contains states which contain possibilities that include g. If we define g as a function whose range is the set of dynamic inq-propositions, we run into the problem of circularity. This is the case as the range of g will be the set of dynamic inq-propositions, however the definition of a dynamic inq-proposition includes g. Thus, we have chosen to map discourse referents to static inq-propositions to avoid this problem.

First, we will begin by providing the layout of the type system that Inq$_{DP}$ operates on. We start with the basic types from Ty2 namely the types of individuals e, worlds s, and truth values t, and a new basic type of discourse referents r. The set of all types is the smallest set containing the basic types and such that for any two types $\sigma, \tau \in$ **Types**, there is also $\langle\sigma, \tau\rangle \in$ **Types** and $(\sigma \times \tau) \in$ **Types**.

A frame is a set of domains such that:

(i) D_e, D_s, D_t, D_r are pairwise disjoint.

(ii) D_e is a domain of entities.

(iii) D_s is a set of possible worlds.

(iv) $D_t = \{0, 1\}$.

(v) D_r is a set of discourse referents and sequences of discourse referents.

(vi) for any $\langle \sigma, \tau \rangle \in$ **Types**, $D_{\langle \sigma, \tau \rangle}$ is the set of all functions from D_σ to D_τ.

(vii) for any $\langle \sigma \times \tau \rangle \in$ **Types**, $D_{\langle \sigma, \tau \rangle}$ is the set of all pairs from D_σ to D_τ.

A model is defined as a frame paired with an interpretation function I over constants of each type and a variable assignment function θ over variables of each type. One thing that differs from dynamic inquisitive semantics is that the domain of the discourse referents contains both discourse referents and sequences of discourse referents. This feature will be explained later in the chapter.

We will begin with a step by step introduction of the objects that make up a context in Inq_{DP}.

Definition 4.2.1 (Static Inq-propositions)**.**

- Downward closure: A set S is downward closed iff for every $P \in S$ every $P' \subseteq P$, $P' \in S$.
- A static Inq-proposition is a downward closed set of sets of worlds; we abbreviate the type $\langle\langle s, t\rangle t\rangle$ as T.

Definition 4.2.2 (dref Assignment Functions)**.** Let F be an Inq_{DP} frame and $\delta \subseteq D_r$ a set of discourse referents in F. A dref assignment function is a partial function $g \in D_{\langle r,T\rangle}$ with $dom(g) := \delta \subseteq D_r$. We abbreviate the type $\langle r, T\rangle$ as m.

Definition 4.2.3 (Possibilities)**.** For any set of discourse referents δ, a possibility with domain δ is a pair $\langle w, g\rangle \in D_{\langle s \times m\rangle}$ where w is a possible world and g is a dref assignment function with domain δ.

Definition 4.2.4 (Information states)**.** An information state is a set of possibilities that share the same domain, we abbreviate the type $\langle\langle s \times m\rangle, t\rangle$ of information states as i.

Definition 4.2.5 (Contexts)**.** A context is a non-empty, downward closed set of information states, thus of type $\langle i, t\rangle$, abbreviated as k.

We list the abbreviations to these notions in the following table:

Object	Type	Abbreviation
Static inq-proposition	$\langle\langle s,t\rangle t\rangle$	T
dref assignment function	$\langle r,T\rangle$	m
Possibility	$\langle s\times m\rangle$	
Information state	$\langle\langle s\times m\rangle t\rangle$	i
Context	$\langle i,t\rangle$	k
Context Update function	$\langle k,k\rangle$	P

Types and Abbreviation conventions

4.3 Context update functions

The denotation of a sentence in inquisitive dynamic semantics is represented as a context update function of type P. These functions take a context as input and output another context. Thus, the meaning of a sentence is characterized by how they change a given context. Context updates in dynamic semantics can either be *constructive* or *eliminative*. A *constructive* update is an update which introduces new discourse referents and thus new possibilities. These possibilities will assign these discourse referents to static propositions. An eliminative update removes information states from the context. Eliminative updates behave similarly to propositions in InqB. They can either provide information or raise issues.

Here we will outline some of the properties of update functions. These definitions will be reminiscent of the properties of propositions in static InqB. This is the case as these properties are the dynamic counterparts. As we have already explained these notions in chapter 1, we will simply outline their formal definitions without explanation.

Definition 4.3.1 (Informative content)**.** For any context c, its Informative content $info(c) := \cup c$.

Definition 4.3.2 (Informativeness)**.** A context c with domain δ is informative iff there is a possibility $\langle w,g\rangle$ with domain δ such that $\langle w,g\rangle \notin info(c)$. Otherwise, c is uninformative.

Definition 4.3.3 (Alternatives)**.** The set of alternatives of a context c, $alt(c) := \{s \in c|\text{there is no } t \in c \text{ such that } t \supset s\}$.

Definition 4.3.4 (Inquisitiveness)**.** A context c is inquisitive iff $|alt(c)| > 1$.

4.4 Context extension and subsistence

In dynamic semantics, constructive updates extend contexts with discourse referents which are encoded into the domain of the dref assignment functions of the context. In this section, we will provide the definition of context extension and subsistence (Dekker et al., 1993). The extension of a context c should satisfy the following conditions: (i) it contains the same world information as c, and possibly additional compatible information. (ii) it contains the same discourse information as c and possibly additional discourse information. We will now specify what it means to be an extension of a dref assignment function.

Definition 4.4.1 (Extension of a dref assignment function)**.** A dref assignment function g' is an extension of another dref assignment function g, written as $g' \geq g$, iff $dom(g) \subseteq dom(g')$, and for all $u' \in dom(g)\ \{u\}, g(u) = g'(u)$.

The extension of a dref assignment function maintains the same discourse information that is already established. It also potentially creates new discourse information. Using this definition, we can define the extensions of possibilities, information states and contexts.

Definition 4.4.2 (Extension of a Possibility)**.** A possibility $\langle w', g' \rangle$ is an extension of another possibility $\langle w, g \rangle$, written as $\langle w', g' \rangle \geq \langle w, g \rangle$, iff $w = w'$ and $g' \geq g$.

Definition 4.4.3 (Extension of an Information State)**.** A an information state s' is an extension of another information state s, written as $s' \geq s$, iff for every possibility $\langle w', g' \rangle \in s'$, there is $\langle w, g \rangle \in s$ such that $\langle w', g' \rangle \geq \langle w, g \rangle$.

Definition 4.4.4 (Extension of a Context)**.** A context c' is an extension of another context c, written as $c' \geq c$, iff for every information state $s' \in c$, there is $s \in c$ such that $s' \geq s$.

The definitions for extensions of information states and contexts differ in form from the definitions for dref assignment functions. Extensions of a dref assignment function requires each element in the original set to be extended. However, for information states and contexts, the extension only needs to have a counterpart in the original set. This difference amounts to the fact that the extension of a context can have eliminated world information. We will also define a specific kind of context extension called subsistence that only involves the addition of discourse information.

Definition 4.4.5 (Subsistence of an information state)**.** An information state s subsists in another information state s', written as $s \preccurlyeq s'$ iff $s' \geq s$ and for every possibility $\langle w, g \rangle \in s$ there is $\langle w', g' \in s'$ such that $\langle w', g' \rangle \geq \langle w, g \rangle$.

Definition 4.4.6 (Subsistence of an information state in a Context)**.** An information state s subsists in a context c, written as $s \preccurlyeq c$ iff there is one or more $s' \in c$ such that $s \preccurlyeq s'$. This s' is called a descendant of s in c.

Definition 4.4.7 (Subsistence of a Context)**.** A context c subsists in another context c', written as $c \preccurlyeq c'$ iff $c' \geq c$ and for every $s \in c, s \preccurlyeq c'$.

We will now move onto defining several context update functions.

4.5 Constructive and eliminative context updates

The first update function that we will introduce is $[u]$. This function is constructive and introduces a dref indexed by u. Since $[u]$ is a context update function, it is of type P. It is a function that takes an input context c and outputs a context c' such that $c \preccurlyeq c'$ and c' is enriched from c with a new dref u. To define this update function, we will have to introduce several projection functions and some sentences in our logical vocabulary.

π_1, π_2 are projection functions such that for any possibility $p = \langle w, g \rangle$, $\pi_1(p) = w, \pi_2(p) = g$. We will write $\pi_1(p)$ as w_p and $\pi_2(p)$ as g_p.

a. $g[u]g' := [dom(g') = dom(g) \cup \{u\}] \wedge \forall v \in dom(g') \cap dom(g) : g(v) = g'(v)$ where g and g' are dref assignment functions.

b. $p[u]p' := w_p = w_{p'} \wedge g_p[u]g_{p'}$

(a.) represents the introduction of u on the level of dref assignment functions and (b.) represents the introduction of u on the level of possibilities. Now that we have these definitions in place, we can define $[u]$.

(4.3) $$[u] := \lambda c_k \lambda s'_i . \exists s \in c\, [\forall p' \in s' : \exists p \in s . p[u]p'] \wedge [\forall p \in s : \exists p \in s' . p[u]p']$$

We will illustrate the effect of $[u]$ on a context c with only two possibilities and an empty domain. The input context c is illustrated in Fig(1.a). An application of $[u]$ yields Fig(1.b). In these diagrams, black dots represent possibilities. The world component of the possibility is specified above and fixed for each column. The associated dref assignment function is specified on the left and fixed for every row. The rectangle in the diagram represents the alternatives of the context. This means that all the subsets of the information state represented by the rectangle are part of the context. The resulting context is extended from c with values for u. As there are 5 static propositions that can be formed given a static context with 2 worlds, only 5 rows of possibilities are added.

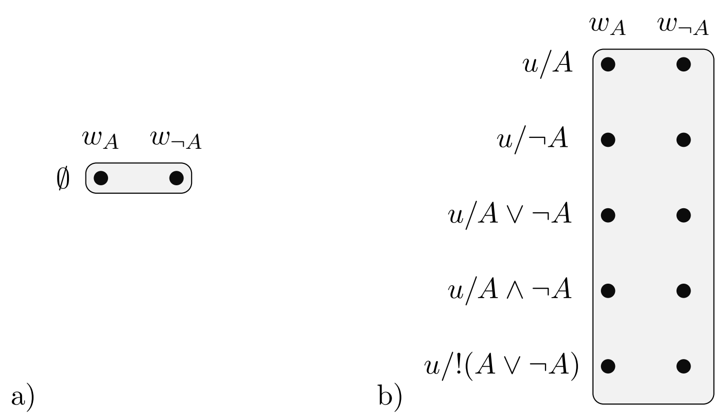


Figure 4.1: Application of the dref introduction operator $[u]$

We will now introduce several eliminative context update functions which will either provide information or raise issues. We will start with the most basic eliminative update function which will be for atomic propositions. These functions will eliminate worlds where a relation R does not hold for the specified entities of type e. Let $R(d_1, ..., d_n)$ be an update function. The semantic interpretation of $R(d_1, ..., d_n)$ is as follows:

(4.4) $R\{d_1, ..., d_n\} := \lambda c_k \lambda s_i . s \in c \wedge \forall p \in s : R(d_1, ..., d_n)(w_p)$

(4.5) Given an Inq_{DP} frame F, and interpretation function $I : R(d_1, ..., d_n)(w) = 1$ iff $\langle I(d_1), ..., I(d_n) \rangle \in I(R)$ at w.

The update function defined in (4.4) will keep all states s in the input context c where all possibilities $p \in s$ have the $n-$tuple $d_1, ..., d_n$ in the extension of R at w_p. Note that this definition does not make use of the dref assignment function as it does not deal with propositional anaphora.

We illustrate the effect of an atomic proposition $R(a)$ on a context c with only two possibilities and a single entity a. The input context is illustrate in Fig(2.a). And the application of $R(a)$ yields Fig(2.b). Since there is only one possibility in which R holds for a, the alternative of the context represented in Fig(2.b) only contains that world.

Figure 4.2: Application of R(a)

We will now introduce several ways of combining context update functions. The first method of combining update functions will be by use of conjunction. Conjunctions will be represented by a merging operation ';', which is represented as follows:

$$A; B := \lambda c_k.B(A(c)) \tag{4.6}$$

The conjunction of two update functions on a context c will be done by sequentially applying the first and second function on c. We will illustrate the effect of updating a context c with only two possibilities and a single entity a with $[u]; R(a)$. The result of updating c with $[u]; R(a)$ is represented in Fig(4.3c). It is the result of first updating with $[u]$ which is represented in Fig(4.3b) and then with $R(a)$. Since $R(a)$ selects only possibilities where R holds for a only those possibilities are in the resulting context.

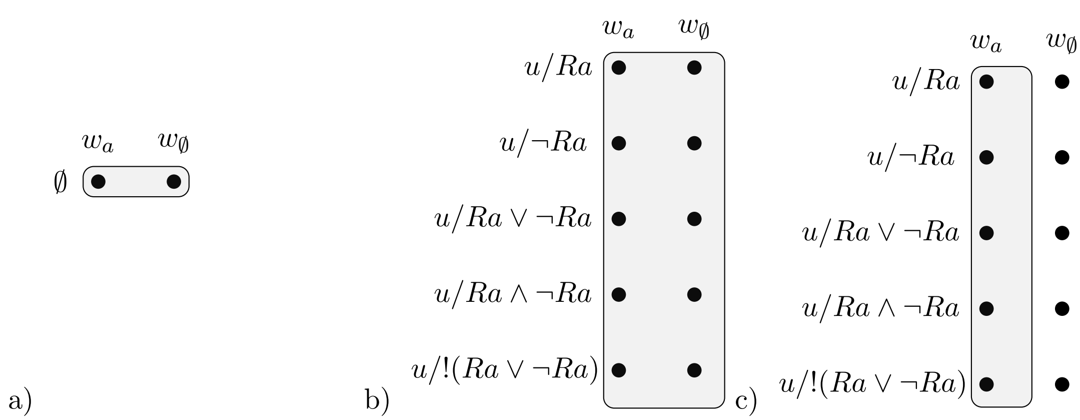


Figure 4.3: Application of the dref introduction operator $[u]; R(a)$

The second method of combining update functions is through disjunction. The update of a disjunction is obtained by taking the union of the result of separately updating the input context with each disjunct. Thus disjunction will be represented as follows:

$$A_P \sqcup B_P := \lambda c_k A(c) \cup B(c) \tag{4.7}$$

We will illustrate the effect of updating a context c with only two possibilities and a single entity a with $[u]; R(a) \sqcup \neg R(a)$ to illustrate the effect of a disjunction. The context is first updated with $[u]$ which yields Fig(4.4b). Then, it is updated with $R(a) \sqcup \neg R(a)$ which yields Fig(4.4c). We see that disjunction functions in the same way in a dynamic context as in a normal context. In Inq$_B$, the disjunction $Ra \vee \neg Ra$ provides two alternatives based on the truth and falsity of Ra. The same effect occurs in the dynamic setting.

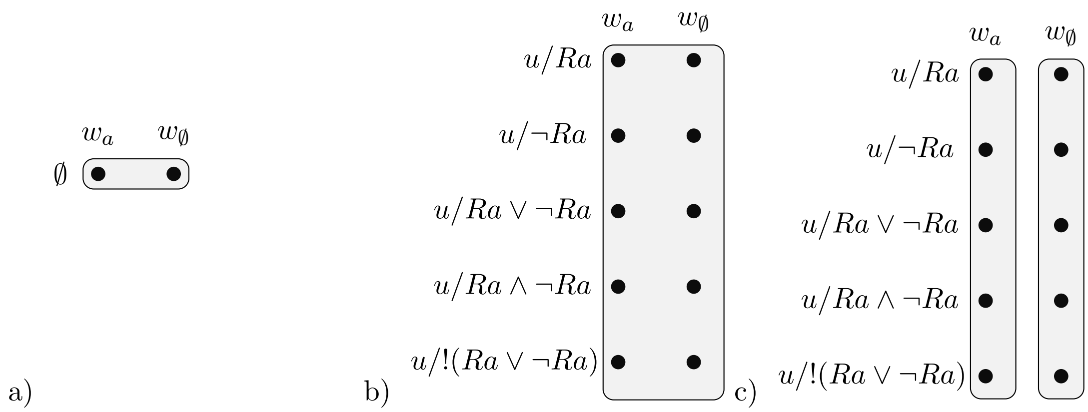


Figure 4.4: Application of the dref introduction operator $[u]; R(a) \sqcup \neg R(a)$

We will now introduce an eliminative update function which assigns a value to an already introduced discourse referent. Since discourse referents are assigned to static propositions, we will first have to define several functions and operators that we will allow for the conversion of dynamic propositions into their static counterparts.

We will now define four projection functions. The first function π_3 takes a dynamic information state and returns its dref assignment-free counterpart. The second function π_4 takes a dynamic information state and returns its world-free counterpart. The third function π_5 takes a dynamic context and returns its dref assignment-free counterpart. The fourth function takes a dynamic context and returns its world-free counterpart.

π_3, π_4 are projection functions such that for any $i = \{p_1, ..., p_n\}$, $\pi_3(i) = \{w_{p_1}, ..., w_{p_n}\}$ and $\pi_4(i) = \{g_{p_1}, ..., g_{p_n}\}$. We will occasionally write $\pi_3(i)$ as i_w and $\pi_4(i)$ as i_g.

π_5, π_6 are projection functions such that for any $c = \{i_1, ..., i_n\}, \pi_5(c) = \{\pi_3(i_1), ..., \pi_3(i_n)\}$ and $\pi_6(c) = \{\pi_4(i_1), ..., \pi_4(i_n)\}$. We will occasionally write $\pi_5(c)$ as c_w and $\pi_6(c)$ as c_g.

We will also need to define the base context c_0. The base context will help us to convert context update functions into static inq-propositions.

$$c_0 = \lambda s. \forall p \in s. g_p = \emptyset \tag{4.8}$$

Now that we have the base context and the two projection functions, we can now define an operator which takes context update functions and converts them into static inq-propositions.

This definition will be revised later on.

$$\downarrow \phi := \lambda p_{(st)}.p \in \pi_5(\phi(c_0)) \tag{4.9}$$

Now that we have a method of converting dynamic propositions to their static counterparts, we can define an eliminative update function $[u, A]$ that assigns an active discourse referent $[u]$ to the static version of a proposition A. This update function when applied to u and A will return a context where all information states only include possibilities which assign u to A.

$$[u, A] := \lambda c_k \lambda s_i s \in c \wedge \forall p \in s : g_p(u) =\downarrow A \tag{4.10}$$

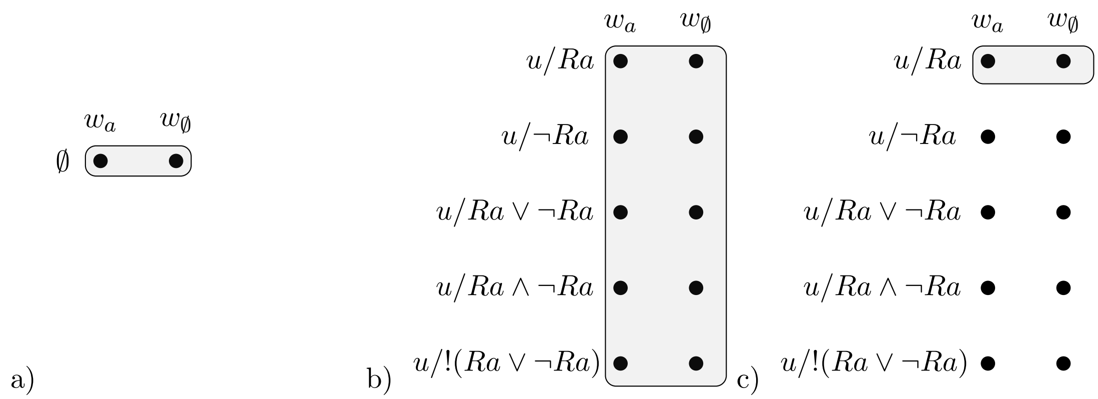


Figure 4.5: Application of the dref introduction operator $[u]; [u, Ra]$

4.6 Dependence in Inq$_{DP}$

We will now define the semantics for dependence which will be an adaptation of the definition by Theiler et al. (2019). This adaptation will evaluate dependence based on the modal base of the possibilities within the states of a context rather than a single world or possibility. We will make the assumption that the possibilities in the context share the same modal base relative to some contextually determined modality. Recall as was the case with the earlier definition, σ_p is the modal base of p. As discussed in an earlier chapter this modality is sometimes determined because it is expressed explicitly as in (4.11a) or implied by context as in (4.11b). In (4.11b) the modality would be something like 'according to the schedules for the bus and the boat'.

(4.11) a. According to Canadian law, one's ability to vote depends on one's age.

a. You may take the bus or the boat. It depends on the time of day.

The translation and semantics for dependence are as follows:

Definition 4.6.1 (Dependence in Inq$_{DP}$)**.** $Tr(\text{Depends on}) = \lambda \vec{v}_{(r)} \lambda u_{(r)} depend\{u, \vec{v}\}$

- $depend\{u, \vec{v}\} := \lambda c_k \lambda s_i.s \in c \wedge \forall p \in s : DEP_{w_p}(g_p(u), g_p(v_1) \cap ... \cap g_p(v_n))$

- $[\![DEP_{w_p}(X,Y)]\!]_{M,g} = 1$ iff $\exists f \in alt(X)^{alt(Y)}$ such that
  1. $\forall W \subseteq \sigma_{w_p}.\forall \alpha \in alt(Y).(W \subseteq \alpha \rightarrow W \subseteq f(\alpha))$ and
  2. $\exists \alpha, \alpha' \in alt(Y).\alpha \cap \sigma_{w_p} \neq \emptyset \wedge \alpha' \cap \sigma_{w_p} \neq \emptyset \wedge f(\alpha) \neq f(\alpha'))$

This definition is the dynamic version of the analysis of dependence from Theiler et al. (2019). Since we have moved to a dynamic framework propositions are not evaluated at the level of worlds but instead at the level of contexts. In Theiler's analysis dependence was truth-conditional. In this analysis, dependence functions as an eliminative update. If all possibilities satisfy the dependence statement the update will function as a test. However, if any possibility does not satisfy the dependence statement, the states which include those possibilities will be removed. This will occur when possibilities assign different static propositions to the discourse referents. Notably any possibility which assigns a static non-inquisitive proposition to a discourse referent will always be removed given the second condition.

The definition of dependence as taking a sequence of discourse referents in the object position will be more clear once we examine the case which involves 'two things' and the case of 'It depends'. As discussed in the chapter on anaphora, both subject and object position of a dependence statement are suitable antecedents for future anaphora. Thus, we will need to define an operator that allows for propositions which have not been assigned to discourse referents to be introduced during dependence statements.

Because of how we defined *Depends on* a problem has arisen from how we have originally defined the base context c_0 and thus the operator $\downarrow$ which converts dynamic propositions into their static counterparts. This is because context update functions can sometimes contain unintroduced or free variables. This will be the case whenever dependence statements feature propositional anaphora that have not already been introduced. If this is the case, the context update function will be undefined on the base context c_0. To solve this problem, we will have to define what it means for a discourse referent to be unintroduced given a context update function ϕ. We will, then, define c_ϕ which is a context variable indexed with a context update function. This context will replace c_0 and will allow unintroduced discourse referents to be evaluated by the $\downarrow$ operator. This will allow us to assign context update functions with unintroduced discourse to discourse referents.

Our definitions for unintroduced variables will be reminiscent of the notion of free variables and active quantifiers from Dynamic Predicate Logic (DPL) (Groenendijk and Stokhof, 1991).

$iv(\phi)$ = the set of introduced discourse referents in ϕ.

$uv(\phi)$ = the set of unintroduced discourse referents in ϕ.

For a context update ϕ, an introduced discourse referent u is any discourse referent that has been introduced by a context update function $[u]$. An unintroduced discourse referent u is a discourse referent that has not been previously introduced by the context update function $[u]$.

1a. $iv(R\{u_1, ...u_n\}) = \emptyset$

1b. $uv(R\{u_1, ...u_n\}) = \{u_1, ...u_n\}$

2a. $iv([u]) = u$

2b. $uv([u]) = \emptyset$

3a. $iv([u, A]) = \emptyset$

3b. $uv([u, A]) = \{u\}$

4a. $iv(\phi; \psi) = iv(\phi) \cup iv(\psi)$

4b. $uv(\phi; \psi) = uv(\phi) \cup \{u \in uv(\psi) | u \notin iv(\phi)\}$

5a. $iv(\phi \sqcup \psi) = iv(\phi) \cup iv(\psi)$

5b. $uv(\phi \sqcup \psi) = (uv(\phi) \cup uv(\psi)) - (iv(\phi) \cup iv(\psi))$

Now that we have defined introduced and unintroduced variables, we can define c_ϕ which is a context which contains all the states with possibilities that assign the free variables of ϕ the same value as a possibility in one of the states of the context c. This context has discourse information relevant to ϕ in the current context without being informative or inquisitive. Now that we have this definition in place we can redefine the operator $\downarrow$ and $[u, A]$

$$c_\phi := \lambda s.\forall p \in s.g_p \in \bigcup c_g \upharpoonright_{uv(\phi)} \tag{4.12}$$

$$\downarrow_c \phi := \lambda p_{(st)}.p \in \pi_5(\phi(c_\phi)) \tag{4.13}$$

$$[u, A] := \lambda c_k \lambda s_i s \in c \wedge \forall p \in s : g_p(u) = \downarrow_c A \tag{4.14}$$

We will not be able to illustrate the effect of a dependence statement on a given context as the models that are required to show a case of dependence are too large to represent visually. This is the case as the discourse referent introduction function applied on a context with more than two worlds returns an unwieldy amount of possibilities. This is because the amount of inq-propositions that can be made increases exponentially when the number of worlds is increased. Thus, we will not be able to represent the effect visually.

4.7 Compositional Inq$_{DP}$

Now we will illustrate how dependence statements in this system can be formed compositionally. We will first aim to represent a sentence of the form: ‘A or B. It depends on C’. Then, we will aim to represent dependence statements which feature ‘two things’.

We will begin by defining the rules that will allow us to derive sentences in our language. As was done in a previous chapter we will do this by means of Functional application and Predicate abstraction. Which we will repeat below.

Definition 4.7.1 (Functional application)**.** If α is a branching node and $\{\beta\gamma\}$ is the set of its daughters, then $Tr(\alpha)$ is defined if $Tr(\beta)$ and $Tr(\gamma)$ are defined and $Tr(\beta)$ is of type $\langle\sigma, \tau\rangle$ and $Tr(\gamma)$ is of type σ. In this case $Tr(\alpha) = Tr(\beta)(Tr(\gamma))$.

Definition 4.7.2 (Predicate abstraction)**.** If α is a branching node whose daughters are the movement index λ_i and β, then $Tr(\alpha)$ is defined if $Tr(\beta)$ is defined. In this case, $Tr(\alpha) = \lambda x_i.Tr(\beta)$.

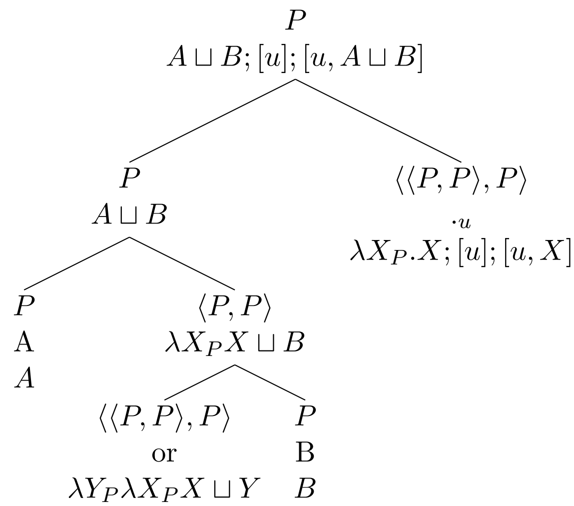
We will now define the translations for several English expressions. To keep things simple, we will use variables of type P to denote sentences. For example, when we seek to represent sentences of the form 'You may walk or you may run. It depends on the weather.', we will represent the first sentence as $A \sqcup B$. This has been done to make this section more readable. Thus, we will not describe the compositionality of the questions which make up these dependence statements. However, as is the case in typed-inquisitive semantics, the compositionality of a question can be represented in dynamic inquisitive semantics.

Translations and their Types			
Cat	α	$Tr(\alpha)$	Type
C	or	$\lambda Y_P \lambda X_P . X \sqcup Y$	$\langle\langle P, P\rangle, P\rangle$
C	and	$\lambda Y_P \lambda X_P X; Y$	$\langle\langle P, P\rangle, P\rangle$
C	$._u$	$\lambda X_P . X; [u]; [u, X]$	$\langle\langle P, P\rangle, P\rangle$
C	(space)	$\lambda X_P \lambda Y_P X; Y$	$\langle\langle P, P\rangle, P\rangle$
S	A	A	P
IV_T	depends on	$\lambda \vec{v}_{(r)} \lambda u_{(r)} DEP\{u, \vec{v}\}$	$\langle\langle r, r\rangle, P\rangle$
dref	$\text{It}_u/\text{That}_u$	u	r

We will begin by describing the translations for connectives. The connectives 'and' and 'or' have been explained earlier and are simply merging and union respectively. '$._u$' in our system represents the ending of a sentence. As propositions are suitable antecedents for anaphoric reference, we will assume that ending a sentence introduces, as a discourse referent, the static inq-proposition expressed by that sentence. This is represented in our system by assuming that the period at the end of a sentence introduces a discourse referent and assigns it to the static proposition associated with the sentence. Thus, sentences of the form: 'A or B.' will be represented as is shown in (4.15a.). We will also occasionally write $[A \sqcup B]_u$ as an abbreviation as seen in (4.15b.). '(space)' in our system represents the conjunction of two separate sentences.

(4.15) a. $Tr(\text{A or B}._u) = A \sqcup B; [u]; [u, A \sqcup B]$
b. $[A \sqcup B]_u := A \sqcup B; [u]; [u, A \sqcup B]$

We will now illustrate in a tree structure how to generate 'A or B.'



We will now provide some information about the anaphoric expressions 'it' and 'that'. When discourse referents are not defined at any possibility, context update functions which feature discourse referents are undefined. This is the case as context update functions are not defined on every context. For example, if a context function A refers to a discourse referent u that is not in the domain of a context c, then $A(c)$ will be undefined. This is desired as it is unclear how a context is updated when someone makes the utterance 'It depends on that.' without introducing the referents from 'it or that' beforehand. Recall earlier that we presented the following two statements and said that (4.16b) is odd. Our system predicts this oddity as the dependence statement will come out as undefined as 'it' has no referent.

(4.16) a. Will you go to the store? It depends on the weather.
b. ?It depends on the weather. Will you go to the store?

As *depends on* is of type $\langle\langle r, r\rangle, P\rangle$ it combines with two discourse referents to become a context update function, we will have to define a type shifting operator which will allow context update functions to combine with *depends on*. Additionally, this operator will allow arguments of depends on to be introduced as discourse referents. This will allow other dependence statements to refer to arguments of previous dependence statements. How exactly this lets context update functions combine with depends on will be explained later.

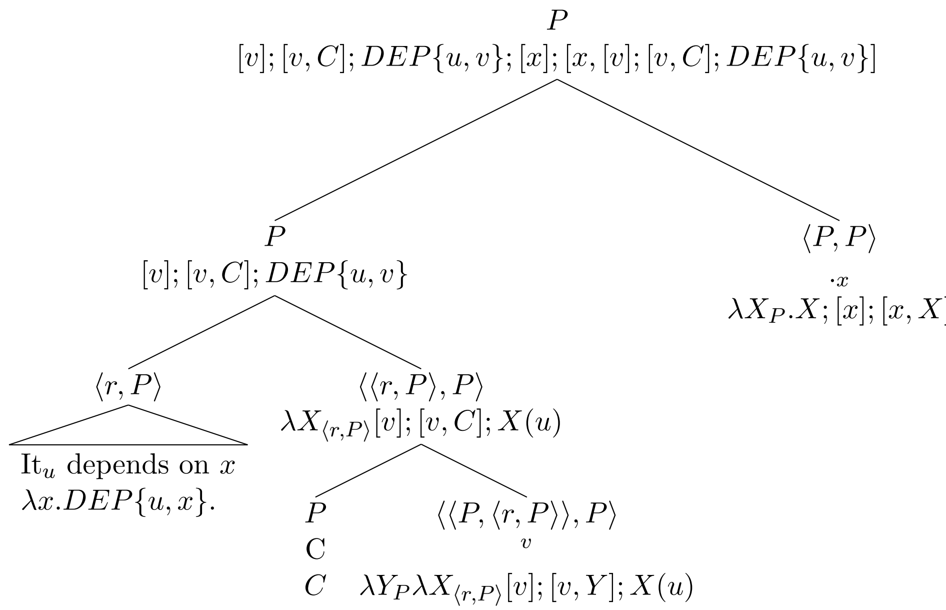
(4.17) $$^{u} := \lambda Y_P \lambda X_{\langle r, P\rangle}[u]; [u, Y]; X(u)$$

We finally have all the tools necessary to define the semantics for 'It depends on C' and 'A or B. It depends on C'. As the meaning of these propositions is quite complex, we will first show the representation before showing how the terms combine compositionally by means of a tree.

(4.18) $Tr(\text{It}_u \text{ depends on } C^v{\cdot}_x) = [v]; [v, C]; DEP\{u, v\}; [x]; [x, [v]; [v, C]; DEP\{u, v\}]$

(4.19) $Tr(\text{A or B}{\cdot}_u \text{ It}_u \text{ depends on } C^v{\cdot}_x) =$
$[A \sqcup B]_u; [v]; [v, C]; DEP\{u, v\}; [x]; [x, [v]; [v, C]; DEP\{u, v\}]$

We will now illustrate with a tree how the sentence 'It$_u$ depends on C$^v{\cdot}_x$' combines compositionally.



The previous tree illustrates how depends on can combine with type-shifted context update functions to create dependence statements. The expression 'It$_u$ depends on x' is created by means of predicate abstraction on the expression 'It$_u$ depends on that$_x$'. This allows it to combine with the type shifted context update function C^v. To compose the statement: 'A or B. It depends on C', we simply combine the previous two trees by means of the '(space)' conjunction. This allows us to generate the desired sentence.

We will now represent dependence statements which involve the noun phrase 'two things'. As discussed in the section on nominals, 'two things' seems to function as a cataphor which refers to two questions that can be listed later. Thus, we want utterances of 'two things' in the system to introduce two discourse referents. Additionally, we want the discourse referents introduced to be assignable to static propositions. We have already defined an update function that introduces a discourse referent and allows it to be assigned to a static proposition. We now need to define an update which takes two discourse referents and removes all possibilities where they are assigned to the same static proposition.

(4.20) $$[u \neq v] := \lambda c_k \lambda s_i.s \in c \wedge \forall p \in s : g_p(u) \neq g_p(v)$$

This context update function will remove all possibilities from the context which map u and v to the same static proposition. With this definition we can now define the meaning of 'two things' and the sentence 'It depends on two things'.

(4.21) $$Tr(\text{two things}_{u,v}) := \lambda X_{\langle r,P\rangle}.[u];[v];[u \neq v];X(u,v)$$

(4.22) $$Tr(\text{It}_u \text{ depends on two things}_{v_1,v_2}) = [v_1];[v_2];[v_1 \neq v_2];DEP\{(u),(v_1,v_2)\}$$

Given that the 'it' in (4.22) has a suitable antecedent in the context. The update will eliminate any states such that the possibilities in that state assign the discourse referents to questions that 'it' does not depend on. Thus, it will function as an eliminative update which removes possibilities with assignment functions that send v_1, v_2 to inq-propositions that the subject question does not depend upon.

We will now illustrate by means of a tree how this sentence combines compositionally.

P
$[v_1]; [v_2]; [v_1 \neq v_2]; DEP\{u, (v_1, v_2)\}$

$\langle r, P\rangle$
It$_u$ depends on x
$\lambda x.DEP\{u, x\}.$

$\langle\langle r, P\rangle, P\rangle$
two things$_{v_1,v_2}$
$\lambda X_{\langle r,P\rangle}.[v_1]; [v_2]; [v_1 \neq v_2]; X(v_1, v_2)$

Finally, we will show how to represent the listing of questions that can follow a dependence statement with the expression 'two things'. In Chapter 2, we presented data where someone made a dependence statement about two things and then listed the two things that it depended upon. We can see the form of the statement in (4.23a.). We want it to be such that on any context (4.23a) and (4.23b) have the same effect. In the case where A does not depend on B and C, we want the resulting context to be empty. In the case where A does depend on B and C, we want the context to only include possibilities that assign the introduced discourse referents to the correct questions. These two desiderata can be achieved by representing the listing as the merging of the eliminative update function $[u, A]$. Thus, if v_1 and v_2 are the discourse referents introduced by 'two things', we will represent the listing 'B and C' as $[v_1, B]; [v_2, C]$. This update will remove any state which includes a possibility that assigns other inq-propositions to the discourse referents. This will give us the desired result in both cases.

(4.23)
a. A depends on two things. B and C.
b. A depends on B and C.

In this chapter, we presented a system designed to represent propositional anaphora in dependence statements. We did this by means of a dynamic inquisitive system which assigned discourse referents to static inq-propositions. This system was able to represent a number of dependence statements which featured propositional anaphora. Although this system was designed to represent dependence statements which feature propositional anaphora, the system can likely also represent other statements which contain embedded inq-propositions. Thus, if one wanted to analyse other predicates which embed inq-propositions in a dynamic system, they simply need to provide a proper semantics for the predicate in question.

Chapter 5

Conclusion

The goal of this thesis was to represent the formal structure of dependence statements. The purpose of this was to provide a better understanding of the concept of dependence and the various arguments they can take. We began by presenting the concept of dependence in both dependence logic and inquisitive semantics. We then presented a bunch of puzzling linguistic data about dependence statements and analysed them from the perspective of inquisitive semantics. We discovered that the analysis of *depends on* as a question embedding verb was not problematic despite the numerous seemingly non-inquisitive arguments that it could take. This was due to our use of inquisitive semantics which analysed these seemingly non-inquisitive arguments as inquisitive.

We then, extended a typed semantics with the notion of conceptual covers. This allowed us to represent dependence statements that featured concealed questions. We also demonstrated that concealed questions in dependence statements should be evaluated from a perspective on the individuals of the domain of discourse. This was shown by presenting an example where a concealed question in a dependence statement was ambiguous between two perspectives. Thus, we provided further support for a perspectival analysis of concealed questions. Additionally, we showed that dependence statements which contained concealed questions shared similar ambiguities to other cases of concealed questions. Thus, the analysis by Aloni and Roelofsen (2011) was also able to represent these ambiguities.

Finally, we presented a variation of dynamic inquisitive semantics which was able to represent propositional anaphora. This system allowed us to represent anaphoric expressions such as 'it' and 'that' in dependence statements. It also allowed us to represent the expression 'two things' which when featured in a dependence statement serves as a cataphor for two questions which can be listed later by the speaker. Although, the system that we presented was designed for the representation of dependence statements, it is easy to see that such a system can model propositional anaphora more generally.

We have chosen to analyse concealed questions and propositional anaphora separately. However, we believe that a system can be designed that can represent dependence statements that feature both concealed questions and propositional anaphora. We have not explored this system in this thesis however, such a system is likely possible.

Bibliography

Aloni, M. (2001). *Quantification under Conceptual Covers.* PhD thesis.

Aloni, M. (2007). Free choice, modals, and imperatives. *Natural Language Semantics*, 15(1):65–94.

Aloni, M. (2008). Concealed questions under cover. *Grazer Philosophische Studien*, 77:191–216.

Aloni, M. and Roelofsen, F. (2011). Interpreting concealed questions. *Linguistics and Philosophy*, 34(5):443–478.

Baltag, A. and van Benthem, J. (2021). A simple logic of functional dependence. *Journal of Philosophical Logic*, pages 1–67.

Ciardelli, I. (2016). *Dependency as Question Entailment*, pages 129–181. Springer International Publishing, Cham.

Ciardelli, I. (2018). Dependence statements are strict conditionals. In *Advances in Modal Logic*, pages 123–142.

Ciardelli, I., Groenendijk, J., and Roelofsen, F. (2018). *Inquisitive Semantics.* Oxford Surveys in Semantics. Oxford University Press.

Ciardelli, I., Roelofsen, F., and Theiler, N. (2017). Composing alternatives. *Linguistics and Philosophy*, 40(1):1–36.

Dekker, P. (2012). Dynamic semantics, vol. 91 of studies in linguistics and philosophy.

Dekker, P. et al. (1993). Transsentential meditations: Ups and downs in dynamic semantics. *Linguistics*, 14:65–117.

Dotlačil, J. and Roelofsen, F. (2019). Dynamic inquisitive semantics: Anaphora and questions. In *Proceedings of Sinn und Bedeutung*, volume 23, pages 365–382.

Dotlacil, J. and Roelofsen, F. (2021). A dynamic semantics of single-wh and multiple-wh questions. In *Semantics and Linguistic Theory*, volume 30, pages 376–395.

Frana, I. (2006). The de-re analysis of concealed questions: a unified approach to definite and indefinite concealed questions. In *Semantics and Linguistic Theory*, volume 16, pages 17–34.

Fusco, M. (2019). Sluicing on free choice. *Semantics and Pragmatics*, 12:1–20.

Groenendijk, J. and Stokhof, M. (1984). *Studies on the Semantics of Questions and the Pragmatics of Answers*. PhD thesis.

Groenendijk, J. and Stokhof, M. (1991). Dynamic predicate logic. *Linguistics and philosophy*, pages 39–100.

Heim, I. (1979). Concealed questions. In Bäuerle, R., Egli, U., and von Stechow, A., editors, *Semantics From Different Points of View*, pages 51–60. Springer Verlag.

Kaufmann, M. (2016). Free choice is a form of dependence. *Natural Language Semantics*, 24.

Maden, R. (2019). Inquisitive semantics under conceptual covers. *Logic and Conversation 2019 Course Paper*.

Snider, T. N. (2017). *Anaphoric reference to propositions*. PhD thesis.

Theiler, N. (2014). A multitude of answers: Embedded questions in typed inquisitive semantics. *Master of Logic Thesis*.

Theiler, N., Roelofsen, F., and Aloni, M. (2019). Picky predicates: Why believe doesn't like interrogative complements, and other puzzles. *Natural Language Semantics*, 27(2):95–134.

Väänänen, J. (2007). *Dependence logic: A new approach to independence friendly logic*, volume 70. Cambridge University Press.

van Rooij, R. (2004). Utility of mention-some questions. *Research on Language and Computation*, 2(3):401–416.

Yang, F. and Väänänen, J. (2016). Propositional logics of dependence. *Annals of Pure and Applied Logic*, 167:557–589.

Zhao, Z. (2019). Varieties of distributivity: From mandarin dou to plurality, free choice and scalarity. *Master of Logic Thesis*.