1 *Introduction*

This book is about the building of syntactic structures and the dependencies established within them, with particular emphasis on the question of whether structures are erected from lexical heads up to potentially quite large functional extended projections of these lexical heads or the other way around. It is a book, therefore, about dependency and directionality.

1.1 *Dependency*

In the title of the book, 'dependency' is a cover for all the kinds of dependencies that can be established in a syntactic structure. The dependency relation between a predicate and its arguments is one example. Another is the dependency between an element up the tree and a gap somewhere downstream – a 'filler–gap dependency', in the terminology of Fodor (1978). Such dependencies arise in constructions in which one or more constituents of the structure find themselves in positions in which they cannot be fully interpreted – for instance, when a *wh*-constituent finds itself at the left edge of a matrix clause but serves as one of the arguments of a predicate in a lower clause embedded inside that matrix, as in (1a). In such sentences, a single element (here, *which claim*) seems to belong in two places in the structure: it belongs in the matrix SpecCP position (where it is pronounced) by virtue of the fact that it is a *wh*-phrase in a constituent question, scoping over the entire complex sentence; but it also belongs to the predicate *false*, which it serves as its subject. Filler–gap dependencies are characterised exactly by this 'two for one' signature: two positions for one element. For such dependencies, it is customary in the generative approach to syntax to postulate a silent element in the lower of the two positions associated to the filler upstairs, as depicted in (1b).

1

2   *Introduction*

(1)   a. which claim do you think that the book proves false?
        b. *which claim* do you think that the book proves   *<gap>*   false?
                                                        t
                                                  which claim
                                                        pro

    The gap has taken on a variety of different identities in the history of generative syntax: a trace (notation: *t*), a silent copy (marked with strike-through), or a silent pronominal element (here called '*pro*'). We will look into these options in more detail. For now, what matters is that there is a cogent reason for thinking that there is indeed a gap inside the lower clause linked to the *wh*-filler in the matrix: without the postulation of the gap, it would be impossible to establish a predicate–argument dependency between *false* and *which claim* in keeping with what we know about such dependencies. One thing we know with certainty is that predicate–argument dependencies are strictly local: we cannot understand *false* in *the book proves this claim false* as being predicated of anything other than *this claim*, its sister constituent; it is impossible to interpret this sentence in such a way that the claim is proved and the book is false, with *the book* serving as the subject for the predicate *false*. Likewise, in (1a) we cannot understand *false* as being predicated of anything other than *which claim*. To capture this, we want *which claim* in (1a) to be structurally represented close to *false*. The postulation of the gap in (1b) takes care of this: regardless of the exact identity of the element in the position of the gap, as long as this element is linked to the filler in the matrix clause, *which claim* is represented in the structure in a position strictly local to the predicate to which it belongs.

1.2   *Directionality*

Predicate–argument dependencies, such as the one between *false* and *which claim* in (1a), are at the very core of syntactic structures. Every clause has at least one predicate–argument dependency established inside it. Because such dependencies are so fundamental to syntax, it is reasonable to think that the construction of syntactic phrase structures is anchored to them: we start out with the predicate head, saturate its argument structure by merging elements into the structure to which the predicate can assign argument roles, and then proceed to erect additional syntactic structure atop the basic predicate–argument structure, as needed. This is the view of syntactic structure building pervading the generative syntactic literature, called the 'bottom-up'

approach as it is customary to place heads of phrase structures at the bottom of
the tree and to project these heads and the functional projections outside their
projections upwards, from the bottom up.[1]

In the bottom-up approach, saturation of predicate–argument dependencies
is the syntax's first order of business. For constructions involving one or more
filler–gap dependencies, this line of thinking typically gives rise to a treatment
of such dependencies in which the filler 'moves up the tree': in (1b), *which
claim* starts out in the position of the gap in the lower clause, and eventually
makes it to the position in the matrix clause where it is pronounced. Such
movement must leave something behind in the position in which *which claim*
first entered the tree: otherwise, we would forfeit the local link between it and
its predicate *false*. For some time, generative syntacticians called the element
left behind a 'trace', taking the movement metaphor quite literally: when you
move, you leave traces of yourself behind. Traces are usually thought to emerge
in the syntactic structure entirely as a by-product of movement: they were not
there before. If so, traces are 'intruders' (more fancily put, 'extraneous sym-
bols'); they pop up ex nihilo.

In more recent developments of the mainstream generative theory of
syntactic analysis, the bottom-up approach has been taken more seriously
than ever before. Everything is fed to syntax from below. We cannot start
building complex syntactic structures unless we first acquire a collection of
building blocks ('morphemes'). Once we have acquired a decent collection (a
'lexicon'), we can take a subset of morphemes from it and put them in a bag,
forming the selection ('numeration', 'lexical array') out of which we can
build our structure. We build by combining one morpheme with another
('selection' under 'sisterhood'), thus forming a complex structure that will
bear the label of one of the two constituent parts ('projection'). We can then
merge the structure thus created with (a substructure projected from) another

---

[1] Syntactic trees thus look very much like genealogical trees, where the patriarch is usually at the
top of the diagram and the most distant descendants at the bottom. The idea that syntactic trees
grow upwards effectively means that they grow 'towards the root'. In the natural world, trees of
course do not grow from the leaves to the branches to the root: real trees grow the other way
round, from the root out and up. The 'bottom-up' approach *seems* to draw a parallel between
syntactic tree-growing ('bottom-up') and biological tree-growing (likewise 'bottom-up'). But
the parallel is illusory: in generative syntax, where the root of the tree is standardly located at the
top of the diagram, 'bottom-up' means *towards* the root, whereas in the natural world (where
trees are not upside down) 'bottom-up' means '*out from* the root'. Of course, my point here is not
that parallels between syntactic tree-growing and biological tree-growing should be expected or
deemed desirable. My point is merely that it would be a mistake to think that 'bottom-up'
structure building in syntax is 'natural' or 'right' because that is how trees grow in general.

morpheme. Our structure keeps getting larger and larger. We stop once we have placed the last remaining item from our selection into the structure. But at any point before we put this last item from the selection into the structure, we may change our minds, uncoupling a substructure from the structure built up to that point, and reattaching it somewhere higher up the structure. The result of this will leave a gaping hole in the structure: a gap, linked to the reattached bit of structure higher up. This is 'movement' – or 'Internal Merge', as it is usually called today.

For our example in (1a), the numeration will contain (among other things, of course) *false, which*, and *claim*. By taking *which* and *claim* out of the numeration and putting them together we build the complex noun phrase *which claim*. This noun phrase we can subsequently put together with *false*, which gives *which claim* its argument role, and thereby build a well-formed minimal predication [*which claim false*]. Here, *which claim* is a so-called *wh*-phrase serving as the subject of the predicate *false*. In languages such as English, argumental *wh*-phrases often do not occur in argument positions on the surface: they must typically occur in a position commensurate to their status as operators – one usually says *which claim does the book prove false?*, not *the book proves which claim false?* (though there are specific registers, such as 'quiz master questions', in which the latter does frequently occur; and in *wh*-questions containing multiple *wh*-phrases, all except one must be pronounced in situ in English, as in *which book proves which claim false?*). So if we build [*which claim false*] at an early point in the syntactic structure-building process, we will eventually want to reattach *which claim* to a larger development of the structure to arrive at (1a). This reattachment gets *which claim* to the position in which it is pronounced in (1a). The position in the structure where we had initially merged *which claim* cannot be left totally empty. We must put something there linked to *which claim*. One way to ensure continued occupancy of the initial merge site after movement is to make a copy of *which claim*, so that we have two tokens, and to leave one of these tokens in the initial merge site. This is the so-called copy theory. It is the optimal solution to the problem posed by movement if the numeration for (1a) does not contain anything besides the physical elements *which* and *claim* as the subconstituents of the subject of *false*.

The alternative to multiple copies is to assume that the numeration also includes, besides *which* and *claim*, a silent element that can be merged in the position next to *false*, and be linked in the final representation to *which claim*, merged directly in the position in which it is pronounced. Such a silent element could in principle be precisely what the early principles-and-parameters

literature referred to as a trace – that is, trace theory is not in and of itself irreconcilable with the Inclusiveness Condition of Chomsky (1995), according to which everything present in a syntactic structure must have been included in the array out of which the structure was built: if we are willing to espouse the presence of a trace in the lexica of languages that allow displacement, the trace-based approach would be compatible with inclusiveness. One may not like the idea of including traces in the lexicon, but in the original incarnation of principles-and-parameters syntax (often referred to as 'government-and-binding' or 'GB' theory; Chomsky 1981), traces fit perfectly well into the feature-based matrix of nominal empty categories: NP-traces (traces left behind by movement to an A-position) occupy the [+anaphoric, –pronominal] cell of the typology of nominal types, and Ā-bound traces or variables occupy the [–anaphoric, –pronominal] cell. The fact that traces can represent whole phrases, not just heads, should be treatable in the same way that the fact that overt proforms can represent whole phrases (e.g., in *he lives here*, where the proforms *he* and *here* stand for entire phrases: a DP and PP, respectively) is dealt with.

Alongside overt proforms such as *he*, it seems inevitable to believe that silent pronouns (*pro*'s) must also exist: in languages allowing 'argument drop' (i.e., the non-realisation of one or more arguments of a predicate in the form of some overt nominal element), such silent pronouns ensure that the argument structure requirements imposed by the predicate will always be met. The distribution of *pro* will need to be carefully understood if we are to entertain the possibility that even in non-pro-drop languages the gap linked to a filler could, under certain circumstances, be *pro*. But there can be no doubt that for languages that allow pro-drop, the gap associated with a *wh*-filler can be pronominal. We will see evidence for this in Chapter 4.

So it seems that, all things considered, all three perspectives on the identity of the gap in (1b) – trace, silent copy, and *pro* – are viable in principle. If we take the word 'trace' literally, the use of traces will bias the syntactic derivation towards a bottom-up approach featuring movement: traces are 'left behind'. On the other hand, identifying the gap in (1b) as a silent pronoun is more compatible with an approach to filler–gap dependencies in which the filler is introduced directly in its pronunciation site and linked to the gap via a binding relation. Some binding relations have been reworked in terms of movement by some syntacticians in recent years (see Kayne 2002, Zwart 2002, and references therein); in a similar vein, some of the relations represented in earlier approaches in terms of 'control' (a representational relationship similar in many ways to binding) have been reconceived as movement

dependencies (see Hornstein's 1999 movement theory of control). It seems unlikely, however, that *all* binding and control dependencies can be remodelled in movement terms. Proponents of the movement theory of control have been quite explicit that 'non-obligatory control' cannot consistently be subsumed under movement. One could eliminate reference to 'control' here as well, by representing 'non-obligatory control' as pronominal binding. But if so, and if the argument against treating non-obligatory control in movement terms is valid, then not all binding can be reduced to movement. As we will see in the course of the discussion in Chapter 4, Ā-dependencies footed in a pronominal element ('*pro*' in (1b)) also behave demonstrably differently from Ā-dependencies in which the element downstairs is not pronominal. Not all Ā-dependencies are created equal: we will want to have the pronominal strategy at our disposal. The pronominal strategy does not presuppose a bottom-up approach to syntactic structure building – it may in fact work better in a top-down alternative.

The copy theory of filler–gap dependencies popular in present-day syntactic theorising was conceived as part of the bottom-up approach. It is important to realise, however, that it is intrinsically neutral with respect to the question of how the syntactic derivation unfolds. All that the copy theory says is that there are two or more tokens of a particular constituent present in the structure – in the particular case of (1b), there are two tokens of *which claim*: one in the operator position in the matrix clause, and one in the θ-position local to the predicate *false*. The copy theory is not itself a theory of movement: it is a theory (to the extent that it deserves that epithet) of the creation of multiple tokens of the same constituent. The idea that one of the copies arrives in a different position from the one it started out life in as a result of movement through the tree is extrinsic to the idea that there are two copies of the displaced constituent; what matters, if copying is the right metaphor, is that there are (at least) two tokens of *which claim* in the structure for (1a), and that the one at the left edge of the matrix clause is the one that is pronounced. If we believe that the syntactic derivation proceeds from the bottom of the tree upwards, and hence constructs [*which claim false*] long before we ever get to the auxiliary *do* in the upstairs clause, then a copy of *which claim* must be made that can eventually be merged in the position to the left of *do*. If this copy is made after *which claim* was first merged into the structure, next to *false*, then one of the two tokens of *which claim* must be transported up the tree, via 'movement' or 'Internal Merge'. But it is only if we assume that the copying is done after *which claim* has been merged in its argument position that we visit copy displacement or 'movement' upon the syntactic derivation. We can readily think of at least

two plausible alternatives that do not require the movement metaphor and for
which bottom-up syntactic derivation is not needed.

   The first alternative is to assume that the copy is made before *which claim* is
even merged into a larger structure. The noun phrase *which claim* must be
merged as a chunk with whatever it belongs to – we cannot merge *claim* with
false 'directly out of the numeration', so to speak, because predicates such as
false cannot actually select bare count singulars such as *claim* (at least, not in
English): **the book proves claim false* is ungrammatical (except in telegraphic
language, which has a separate grammar). So we must first construct *which
claim*, a complex noun phrase, before we can have it saturate *false*'s argument
structure requirements. Once we are done building *which claim*, we can make it
input to the construction of the predication structure featuring *false* right away,
and create a copy of it later, as in the standard 'copy theory of movement'. But
in principle we could just as well produce a copy of *which claim* before this
merge operation is performed, and subsequently merge one of the copies with
the projection of *false* while keeping the other on hold. Each of the two tokens
of *which claim* could then be 'Externally Merged' (or 'base-generated') in the
syntactic structure independently of one another, in two different positions –
one the θ-position, the other the position in which *which claim* is pronounced.
Thinking of the copy theory in these terms makes this approach to filler–gap
dependencies completely independent of the overall perspective on the direc-
tionality of syntactic structure building.[2]

   The other alternative to the familiar bottom-up movement-based approach to
copying is to Externally Merge the filler not in its θ-position, but in the position
where it is pronounced, and to perform the copy operation on that high token.
A suitable place for the copy will then have to be found further down the tree –
in the case of *which claim* in (1a), in the θ-position next to *false*. If the search for
a suitable place for the copy were couched in 'movement' terms, the
copy would have to descend down the tree. But we can also take the copy of

----

[2] Creating a copy of *which claim* before the noun phrase is even merged into a larger syntactic
structure is difficult to 'trigger' in a bottom-up grammar given that at the point at which we have
completed *which claim*, we do not know whether this phrase is displaced or not: we could be
dealing with a quiz master question or a multiple *wh*-question, in which case *which claim* is
pronounced next to its predicate *false*. It is important to bear in mind, however, that in the bottom-
up standard, the familiar copy-and-move approach cannot make the creation of a copy of the *wh*-
phrase locally deterministic either: the phase locally dominating *which claim* (the embedded *v*P)
is not a *wh*-question; postulating an uninterpretable [WH] or generalised 'EPP' feature on *v* is ad
hoc and ultimately ineffective (see Section 2.3.4 for discussion). It is more efficacious to assume
that copying is free in principle; derivations with too few/many copies of a particular constituent
lose out to the derivation with exactly as many copies as needed.

the *wh*-phrase to be stored temporarily and to be associated to the θ-position as soon as this position presents itself in the course of the structure-building process, which, on this conception, proceeds 'top-down', from the root node down to the lower branches and leaves.

1.3   The Objectives of the Book

In this book, I will be plumping for such a 'top-down' approach to syntactic structure building and the construction of filler–gap dependencies – an approach in which the filler is first put in the tree in the position in which it is pronounced and/or takes scope, and the grammar initiates a search operation whose mission can be characterised by the title of one of the book's key chapters: Find the Gap (Chapter 3). This, then, is one way of reading this title.

But there is a broader sense in which this title applies to the book as a whole. As the discussion in the previous section has already intimated, generative syntactic theorising is often marred by gaps in the argumentation. Often, the idea that a silent element is 'left behind' by 'movement' is merely taken for granted, not argued for explicitly; and the nature of the silent element (a trace, a null copy, or a null proform) is seldom the object of explicit scrutiny, leaving specific proposals vulnerable to alternative construals that might very well undermine the key conclusions for which support was sought. It is this methodological sense of 'find the gap (in the argument)' that will figure prominently throughout the book as well, in an effort to clearly define the hypothesis space and narrow it down in an explicit and reasoned manner.

The book thus has two main objectives: (a) to lay out and scrutinise syntactic approaches to structure building and the filler–gap dependencies established within them, and (b) to present and support one particular approach, proceeding from the root of the tree towards the bottom, with particular reference to the locality restrictions imposed on filler–gap dependencies. The book pits the discussion of the top-down approach against classic and more recent arguments for bottom-up structure building in the generative approach. By documenting and reviewing the arguments for the standard bottom-up approach and its attributes (including the phase) and for the top-down alternative, the book aims to instil in its readers a conscious command of the role of the directionality of the building of syntactic structures and the dependencies inside them.

1.4   The Structure of the Book

In the process of doing so, the book presents detailed illustrations of the workings of a specific top-down approach as well as extensive empirical and analytical material in the form of several case studies. To be able to achieve the book's objectives in the optimal way, and to make the book both accessible and of interest to audiences with varying backgrounds and degrees of expertise, I have structured the book into four substantive chapters, whose contents are summarised in the following subsections.

1.4.1   Chapter 2: The Directionality of Structure Building

Chapter 2 reviews arguments for a particular direction in which to build syntactic tree structures. It approaches the question of how trees grow (from the bottom up or from the top down) by asking what makes them grow in the first place. Every meaningful structure is ultimately rooted in the projection of a predicate–argument structure. The standard approach in generative theory has always been to take this predicate–argument structure to be projected first, and to grow the rest of the tree from that predicational core through the projection of functional structure facilitating the licensing of the various ingredients of the core. It is often assumed without argument that in a minimalist theory of syntax, structure building *must* proceed in a bottom-to-top fashion, from the minimal predicate–argument structure to the highest node of the functional structure dominating it. Chapter 2 shows that when we closely examine the way clauses are made, it turns out to be by no means infeasible to start at the top and to make one's way down to the predicational core; in fact, from the perspective of the transformational cycle and the computational efficiency of the syntactic derivation, proceeding from top to bottom appears superior to the bottom-up standard.

In the context of the discussion, the chapter puts a special spotlight on Chomsky's (1965) original argument for bottom-up syntactic derivation, based on the cycle, and confronts it with Chomsky's (1957) approach to clausal recursion. This part of the discussion gives rise to a pronounced perspective on the structure of transitive clauses, the syntax of the 'object of' relation, and the syntax/semantics mapping for presuppositionality – one that will serve us well in later chapters, too.

In the second part of the chapter, the focus shifts to two extended arguments in the literature for top-down syntactic derivation. Phillips' (1996, 2003) argument is based on the alleged syntactic reality of the 'temporary constituents' made available by top-down structure building. His arguments are

reviewed meticulously, and (rather unfortunately, in the light of the general thrust of this book) found not to be successful. Chesi's (2007, 2015) arguments for a top-down model of syntax concentrate on long-distance dependencies, successive cyclicity, and parasitic gaps. Though those arguments are shown to have major weaknesses as they stand, Chesi's general outlook on filler–gap dependencies, employing pushdown automata, will be put to good use in the ensuing chapters of this book.

1.4.2   *Chapter 3: Find the Gap*

The mission of Chapter 3 is to design fully explicit accounts for the two types of locality restriction on filler–gap dependencies – absolute (or 'strong') and intervention (or 'weak') islands. The analyses of the two island effects are maximally inclusive, and fall out directly from the general theory of top-down structure building presented throughout the later chapters of the book.

The crux of the top-down analysis of filler–gap dependencies is the idea (familiar from psycholinguistic and computational approaches to the processing of such dependencies, and also exploited in Chesi's [2007, 2015] syntax-centred work) that a filler encountered in a position in which it cannot be fully interpreted is copied, with its copy uploaded onto a pushdown stack or memory buffer; the copy is subsequently downloaded from the stack onto the edge of the first predication structure constructed in the course of the downward structure-building process, at which point the copy is either interpreted in situ or, if it cannot, starts a search for a gap farther down the tree. This search for a gap is interfered with by two types of obstacles: absolute (or 'strong') islands, and intervention (or 'weak') islands.

The central ingredient of the analysis of absolute/strong islands is the hypothesis (ultimately going back to Chomsky 1986, but first proposed within the minimalist programme by Rackowski and Richards 2005) that domains that are in an Agree relationship with an asymmetrically c-commanding head are transparent for the establishment of filler–gap dependencies across their boundaries, whereas domains that are not in such an Agree relation are opaque. Detailed discussion in the realm of absolute/strong islands is provided for the Subject Condition, the Complex Noun Phrase Constraint, the Specificity Condition, and factive islands. Particular attention is paid to the Adjunct Condition, with focus on the ways in which adjuncts can engage in Agree relations and the role played by such relations in regulating extraction from them.

For weak/intervention islands, the key analytical ingredient is the hypothesis that the search for a trace is halted by an intervener of the same type as the filler, but that *argumental* fillers that bump into a weak island can be re-uploaded on the back of an intervener that needs to be uploaded onto a pushdown stack for independent reasons. This hypothesis is shown to provide simple and empirically highly accurate analyses of *wh*-islands as well as the complete inventory of 'Beck effects' (Beck 1996) and 'inner islands' (Ross 1984) – all cases of harmful intervention of a scope-taking element between a non-argumental filler and its gap.

1.4.3   Chapter 4: A Syntactic Typology of Long Ā-Dependencies

Chapter 4 looks in detail at the strategies available to the language user for the formation of Ā-dependencies, with particular emphasis on long-distance *wh*-fronting constructions – that is, *wh*-dependencies spanning more than a single clause. For cross-clausal Ā-dependencies (such as the one that *which claim* engages in in the example in (1a)) it has long been standard in the generative literature to treat them in terms of a succession of short movement steps from the filler's base position up to its spell-out or scope position. This familiar approach will be confronted with the rich tableau of long-distance Ā-dependency types that natural language has at its disposal (with particular focus on the facts of Hungarian), including *wh*-scope marking (aka partial *wh*-movement), *wh*-prolepsis-cum-resumption, *wh*-topic drop, *wh*-copying, and *wh*-control.

For all *wh*-dependency types reviewed in Chapter 4, a top-down perspective on syntactic structure building yields the optimal results. The survey of the typology of long Ā-dependencies thus informs the way trees are built. The chapter also shows that of the various strategies available to the grammar for the formation of long Ā-dependencies, none involves successive-cyclic movement – although one particular strategy does mimic familiar successive cyclicity closely, it is immune to *wh*-islands and exhibits an information-structural signature that is incompatible with a successive-cyclic movement analysis. The chapter closes on a brief and pointed critique of the most detailed successive-cyclic approach to long-distance Ā-dependencies available in the minimalist literature: Van Urk and Richards' (2015) analysis of the facts of Dinka.

1.4.4   Chapter 5: The Trouble with Subjects

Filler–gap dependencies involving the subject are the central focus of Chapter 5. These are singled out for chapter-length scrutiny because the

generative approach to date has had very little success in understanding the complex and multifarious restrictions imposed on the subject and the dependencies it entertains in an integrated way. It is the purpose of this chapter to arrive at such an understanding.

In the first part of this chapter, the central players in the realm of the structural subject and its vagaries are introduced, starting with early principles-and-parameters theory (Chomsky 1981), and examining ways to unify and derive the two key principles (the 'Extended Projection Principle' [EPP] and the 'Empty Category Principle' [ECP]) in the minimalist programme, against the background of the main conceptual shifts involved in the transition from GB to minimalism. The second part will show that the top-down theory of syntactic structure building and the creation of filler–gap dependencies presented earlier in the book affords us a new, rigorous, and fully integrated account of the restrictions imposed on the dependencies involving the subject.

The proposal advanced in this chapter unifies the Case Filter and the EPP (the requirement that the structural subject position be filled), integrates the ECP and the EPP, and along the way presents refined perspectives on agreement (Agree) and the distribution of PRO. It does so on the basis of an overarching principle governing the licensing of specifiers, which has specialised incarnations for A–specifiers and traces in specifier positions. The discussion includes, among many other things, in-depth accounts of the ungrammaticality of long subject dependencies in biclausal constructions whose matrix clause contains a verb from the *convince* class (**who did they convince him (that) should do it?*), and the circumstances under which highest-subject questions and relative clauses permit the construction of filler–gap dependencies across their boundaries (*this is a puzzle that we need to find someone who could solve*).

1.4.5   Chapter 6: Conclusion

The brief final chapter recapitulates the main findings and the central lessons learnt from the discussion in the four substantive chapters of the book. It distils the overall perspective on syntax emerging from the book, and gives an incentive for further research.

1.5   *A Note to the Reader*

Large parts of each of the four main chapters of this book are based on the voluminous body of literature produced within the generative approach on the subject of structure building and filler–gap dependencies in syntax,

fundamental to syntactic theorising. The text throughout will make a concerted effort to introduce and critically assess the central issues emerging from this literature, in an endeavour to make the book a complete and self-sufficient whole.

Although a lot of fundamental ingredients of syntactic theory can be picked up and internalised through a perusal of this book, this is not an introduction to syntax. The book was written so as to make it accessible to readers familiar with the basic concepts and technicalities of generative syntax. A proper knowledge of these will be presupposed. The more advanced case studies included in each of the chapters present the fruits of original research based on the foundations laid earlier, and serve the general plotline of the book: an extended argument for a top-down theory of syntax. There is a lot that advanced undergraduate students and beginning graduate students can pick up from those case studies, and I would strongly encourage all readers to make their way through the book linearly. But because of their degree of detail and technicality, some of these case studies will likely be appreciated primarily by more seasoned scholars, including advanced graduate students. I will occasionally include a *caveat lector* at the outset of particularly specialised discussions, giving those whose interests lie primarily elsewhere a pointer towards the bottom line of those discussions.

Readers who confine themselves to the background sections and the critical reviews of the literature should take away from this book a solid familiarity with the issues in the domain of structure building and filler–gap dependencies in generative syntactic theory, and with influential proposals in the literature and their pros and cons. It is my hope that readers who make their way through the book as a whole, including the in-depth case studies, will derive from it a solid sense of how the problems with the extant bottom-up approaches can be tackled by a rigorous and technically parsimonious top-down grammar.