

1. Record Nr.	UNINA9910971502003321
Titolo	Exploring crash-proof grammars // edited by Michael T. Putnam
Pubbl/distr/stampa	Amsterdam ; ; Philadelphia, : John Benjamins Pub. Company, 2010
ISBN	9786612775079 9781282775077 1282775073 9789027288011 9027288011
Edizione	[1st ed.]
Descrizione fisica	1 online resource (315 p.)
Collana	Language faculty and beyond ; ; v. 3
Altri autori (Persone)	PutnamMichael T
Disciplina	415
Soggetti	Grammar, Comparative and general - Syntax Generative grammar Minimalist theory (Linguistics)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Bibliographic Level Mode of Issuance: Monograph
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Exploring Crash-Proof Grammars -- Editorial page -- Title page -- LCC data -- Dedication page -- Table of contents -- Preface & -- acknowledgments -- List of contributors -- Exploring Crash-proof grammars -- 1. Introduction -- 2. Defining crash(es) -- 3. Scope and content of this volume -- Works cited -- Part I. Applications of crash-proof grammar -- Computation efficiency and feature inheritance in crash-proof syntax -- 1. Introduction -- 2. Feature inheritance -- 3. Subject-verb agreement -- 4. Subject extraction and Anti-Agreement Effects -- 5. Long distance extraction and agreement -- 6. DONATE, KEEP and SHARE application in crash-proof syntax -- 7. Conclusion -- Implications of grammatical gender for the theory of uninterpretable features -- 1. Introduction -- 1.1 Theoretical overview -- 1.2 Structure of the paper -- 2. Gender and interpretability -- 2.1 Romance gender -- 2.2 Bantu noun class -- 3. Gender agreement in Bantu and Romance -- 4. Why Bantu agreement is independent of case -- 4.1 The proposal: Gender is never deactivated -- 4.2 Against an Agree-with-Agreement approach -- support from semitic -- 5. Activity: A closer look -- 5.1 Strengthening the Activity Requirement -- 6. A problem for Feature

Inheritance -- 7. Deriving Goal Deactivation -- 8. Conclusion --
References -- The Empty Left Edge Condition (ELEC) -- 1. Introduction
-- 2. A uniform approach to null-arguments -- 3. Germanic argument
drop and the ELEC -- 4. More cases of left edge sensitive argument
drop -- 5. The emptiness conditions are operative in PF -- 6.
Concluding remarks -- References -- Part II. The crash-proof debate
-- Grammaticality, interfaces, and UG -- 1. Linguistics as the study of
I-language -- 2. Acceptability and grammaticality -- 3. Selection and
structure-building -- 4. Prospects for an unprincipled syntax -- A tale
of two minimalisms.
1. Introductory remarks -- 2. The distinguishing feature between the
two minimalisms -- 2.1 The crash-proof route -- 2.2 The alternative
route -- 2.3 A concise comparison, and why Merge has an edge -- 3.
On Agree -- 4. Conclusion -- Uninterpretable features -- 1. Unclearities
regarding the distinction between crash vs. convergent gibberish -- 2.
A pervasive empirical problem for the valuation-transfer analysis -- 3.
Designing a perfect system "primarily" for CI and "secondarily" for SM
-- 4. A crash-proof system and a remaining question -- References --
Syntactic relations in Survive-minimalism -- 1. Introduction -- 2.
"Phrase structure rules" à la the Survive Principle -- 3. Theta Roles in
Survive-minimalism -- 4. Cleaning-up crashes -- 5. Consequences and
conclusions -- References -- Toward a strongly derivational syntax --
1. Introduction -- 2. Labeling and First Merge -- 2.1 C-selection is not
feature checking -- 2.2 C-selection has no role in labeling -- 2.3
Labeling at First Merge: Agree -- 2.4 Collins' Locus and First Merge --
2.5 Crash-proof derivation vs. immediate filtering -- 3. The issues
facing First and Second Merge -- 4. Toward a strongly derivational
syntax -- 4.1 Eliminating the First Merge/Second Merge dichotomy --
4.2 Eliminating First Merge -- 4.3 Asymmetry in narrow syntax and at
the interfaces -- 4.4 Eliminating Merge -- 4.5 Consequences of
Eliminating Merge -- 4.6 Transfer and feature checking -- 4.7
Complex specifiers -- 5. Concluding remarks -- On the mathematical
foundations of crash-proof grammars -- 1. Rainbow, language, theory
-- 2. The concept of crash-proof syntax -- 3. Mechanisms of crash-
proof syntax -- 4. Elements, contexts, and formal Systems -- 5.
Peano's axioms -- 6. The language-number correspondence -- 7.
Conclusions -- Crash-proof syntax and filters -- 1. Introduction -- 2.
OT-syntax as a theory of filters.
3. Crash-proof syntax does not void the need for filters -- 3.1
Movement -- 3.2 Negative sentences -- 3.3 Other differences -- 3.4
The universal generator -- 3.5 Conclusions -- 4. Why developing a
crash-proof syntax may be desirable -- 5. Conclusion -- References --
Crash-free syntax and crash phenomena in model-theoretic grammar
-- 1. Introduction -- 2. Grammar as system of declarative constraints
rather than a system of production operations -- 3. When derivations
crash (in performance) -- 4. Head-Driven Phrase Structure Grammar --
5. Conclusion -- References -- Index -- the Language Faculty and
Beyond series.

Sommario/riassunto

The Minimalist Program has advanced a research program that builds the design of human language from conceptual necessity. Seminal proposals by Frampton & Gutmann (1999, 2000, 2002) introduced the notion that an ideal syntactic theory should be 'crash-proof'. Such a version of the Minimalist Program (or any other linguistic theory) would not permit syntactic operations to produce structures that 'crash'. There have, however, been some recent developments in Minimalism - especially those that approach linguistic theory from a biolinguistic perspective (cf. Chomsky 2005 et seq.) - that have called the pursuit of a 'crash-proof grammar' into serious question. The papers in this

volume take on the daunting challenge of defining exactly what a 'crash' is and what a 'crash-proof grammar' would look like, and of investigating whether or not the pursuit of a 'crash-proof grammar' is biolinguistically appealing.
