

Is Syntax Crash-Proof?

Cedric Boeckx, Harvard University

As is often pointed out, minimalism is a program, not a theory, and there exist quite a few ways of articulating basic minimalist guidelines. One possibility is to insist on the crash-proof character of the computational system known as narrow syntax. I take it that the family of approaches falling under the rubric of crash-proof syntax is a natural development of Chomsky 1993/1995, where the notions of feature checking, Greed, and feature-driven movement were central. In this talk I argue in favor of a different conception of syntactic derivations, one that relies less on feature-specifications and their checking needs, and more on the symmetry between external merge and internal merge, and the free application of these (Chomsky 2004). By becoming less dependent on lexical specifications, syntactic analyses can offer a conceptually superior account of instances of agreement dissociated from movement ('long-distance agreement'), of successive cyclicity, and of the EPP. It also offers a better way of thinking about the syntax-semantics mapping, as syntax acquires a more 'Neo-Davidsonian' (as opposed to a 'Fregean') character. For all these reasons, I argue that, despite initial plausibility, syntax is not crash-proof.