

Information Models

Art Griesser

Which conveys more information?

- Colorless green ideas sleep furiously
- “You don’t have no more troubles, Roscoe,”
I tell him, “you and me is just become
business partners.”

Colorless green ideas sleep furiously

- Grammatically correct
- Meaningless

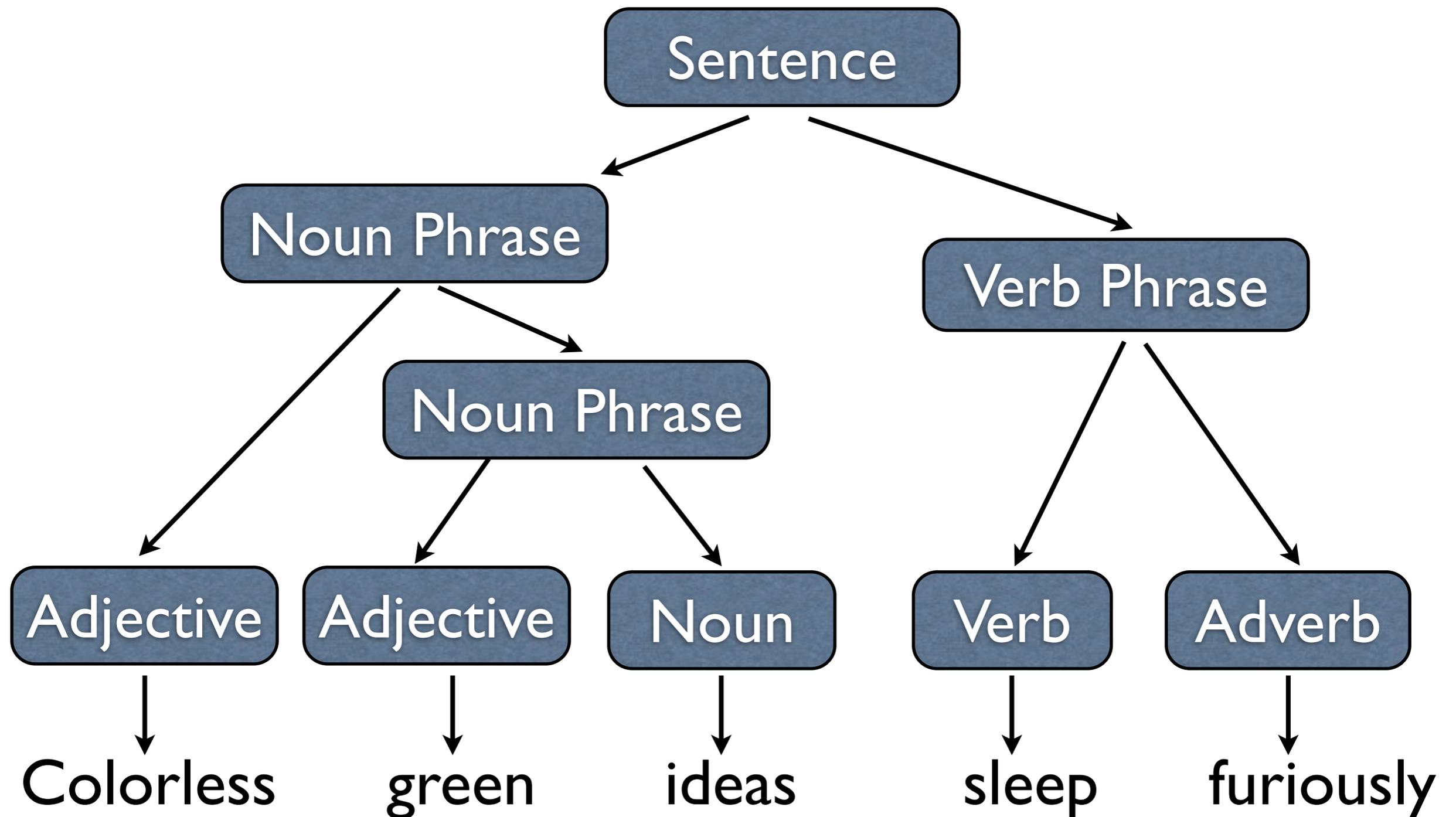
Notorious example from *Syntactic Structures* by Noam Chomsky (1957)

“You don’t have no more troubles, Roscoe,” I tell him, “you and me is just become business partners.”

- Grammatically atrocious
- Meaningful

From the story “Hog-Belly Honey” in the anthology *Nine Hundred Grandmothers* by R.A. Lafferty (1970)

Grammatically Correct



Attempts at meaning

- Colorless  nondescript
- Green  immature or newly formed
- Sleep  mental or verbal dormancy

- “Newly formed bland ideas are inexpressible in an infuriating way”

- “Nondescript immature ideas have violent nightmares”

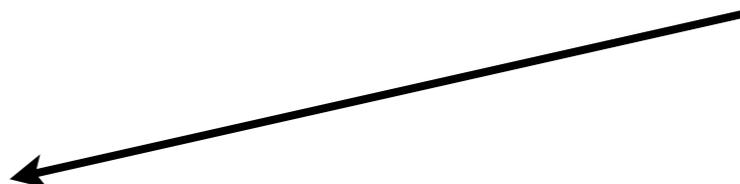
Given only one, which is better to have?

- Grammar (syntax)
- Meaning (semantics)

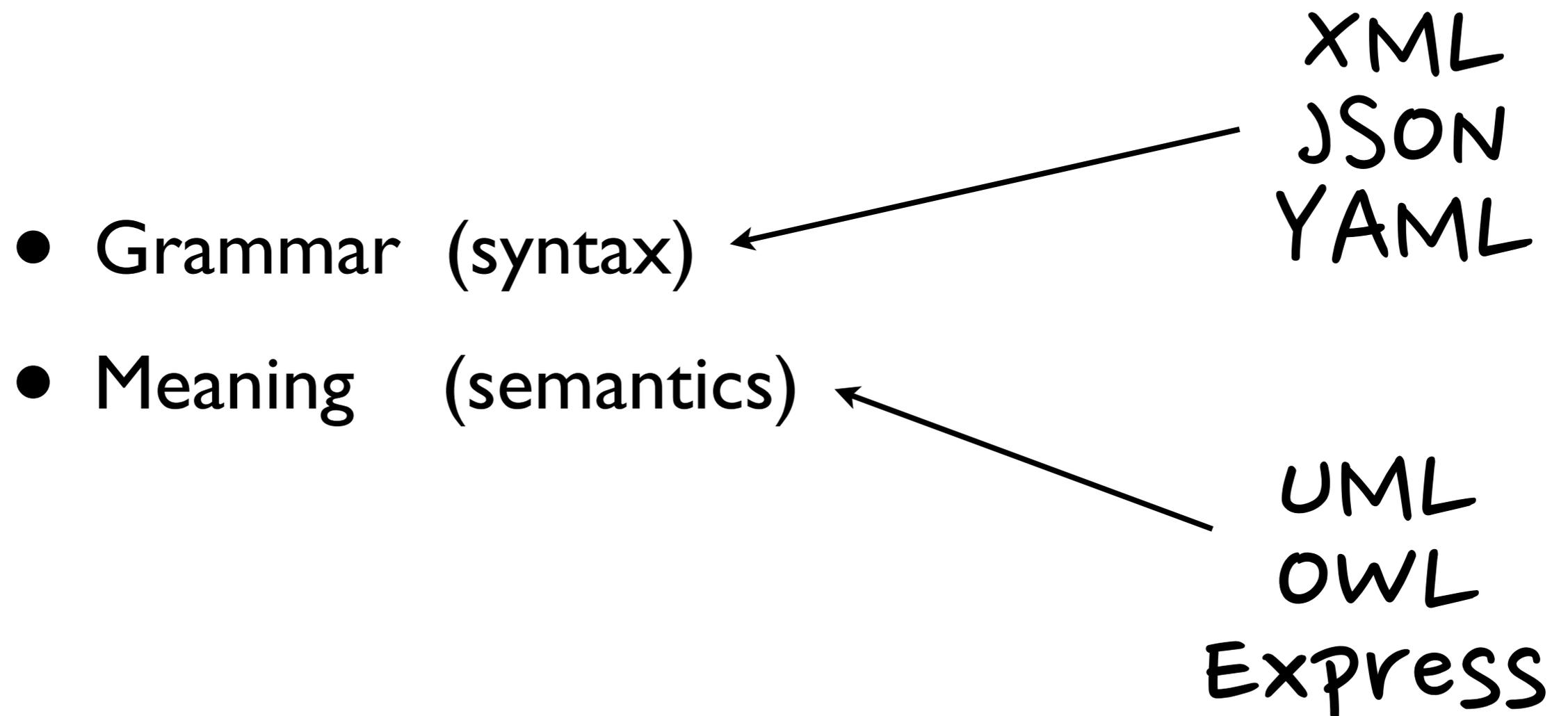
Given only one, which is better to have?

- Grammar (syntax)
- Meaning (semantics)

XML
JSON
YAML



Given only one, which is better to have?



Why XML Schema is Syntactic

Grammar



XML Schema

specifies when a
sequence of

specifies when a
sequence of

words



XML elements

forms a valid

forms a valid

sentence



document

Why UML is Semantic

- Describes relationships between information
- Does not define serialization

An companion to UML (XMI) does define serialization (of both models and instance data). XMI is not widely used for instance data.

Who cares, we've got
them both, right?

Who cares, we've got
them both, right?

Or do we...

Things Like This Happen

An XML schema whose purpose to convey to a bank everything necessary to record a collection of new mortgages, but had no connection between mortgage holder and property

(Details changed to protect the guilty)

Things Like This Happen

If a message is syntactically perfect...
does it have meaning if it omits vital
information?

Can we have them
both?

Yes...

Can we have them
both?

Yes... if the semantics come first

Why do Semantics Need to Come First?

Bike Shed Effect

“Parkinson's Law of Triviality, also known as bikeshedding or the bicycle-shed example, is C. Northcote Parkinson's 1957 argument that organisations give disproportionate weight to trivial issues. Parkinson demonstrated this by contrasting the triviality of a bike shed to a nuclear reactor.”

Wadler's Law

“In the context of programming language design, one encounters Wadler's law, named for computer scientist Philip Wadler. This principle asserts that the bulk of discussion on programming language design centers around syntax (which, for purposes of the argument is considered a solved problem), as opposed to semantics.”

Why do Semantics Need to Come First?

- Because syntax soaks up resources
 - Without addressing deeper problems
- Focus on the semantics delivers more value