Ash tek durbatulûk
ash tek gimbatul,
ash tek thrakatulûk
agh SISO-ishi krimpatul

Making the Case for Alignment of
Standards by Andreas Tolk, Ph.D.

What is a Language?

- What do we mean when we talk about language?
  - English, Spanish, French, German, …
  - UML, XML, DAML, WSDL, …
  - C-BML, MSDL, …
- What do we mean when we talk about language between systems?
 - What sentences do we need?
  - What vocabulary and grammar do we need?

Understanding in Systems

Observing System

Observed System

①
Perception

②
Meta-Models

③
Relationship Mapping
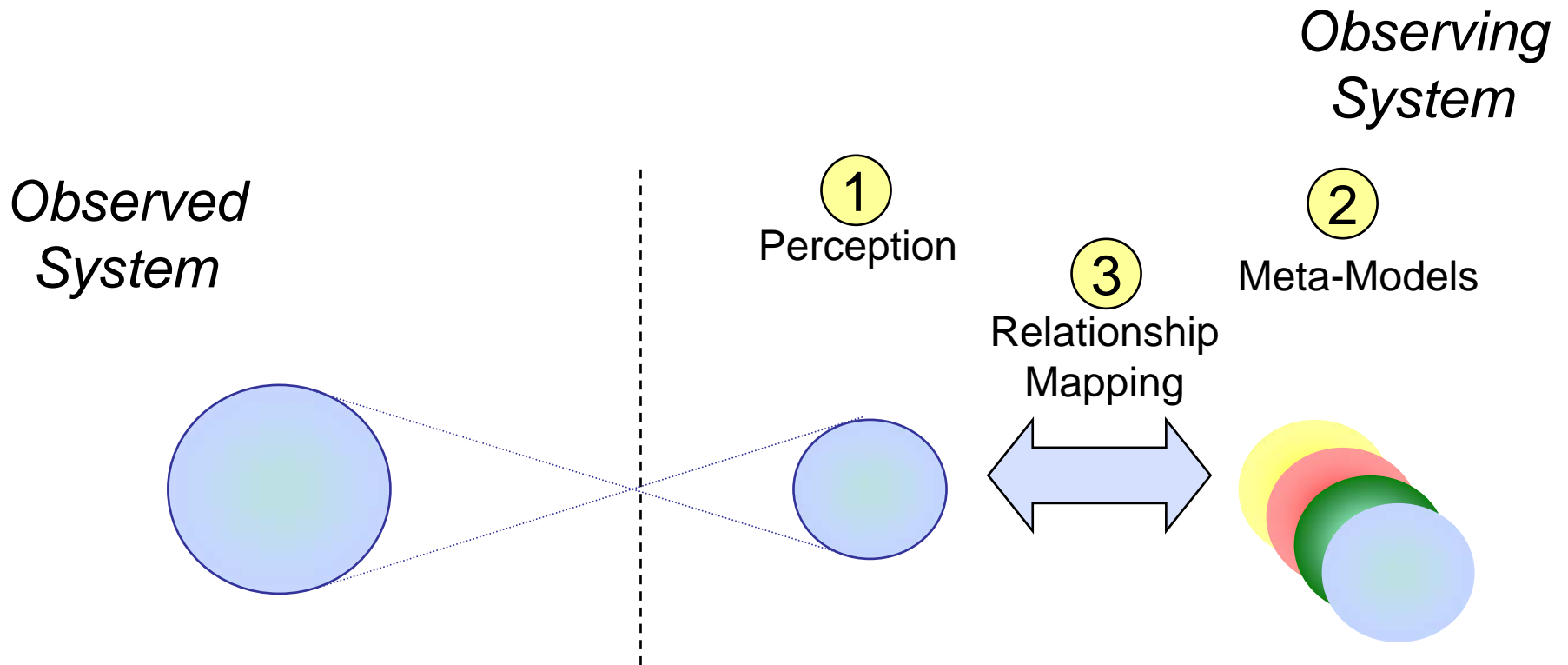
The Three Premises for Understanding
Zeigler B.P. *Toward a Simulation Methodology for Variable Structure Modeling,* In Elzas/Oren/Zeigler (Eds.) Modeling and Simulation Methodology in the Artificial Intelligence Era, North Holland, 1986

One Model – Many Views
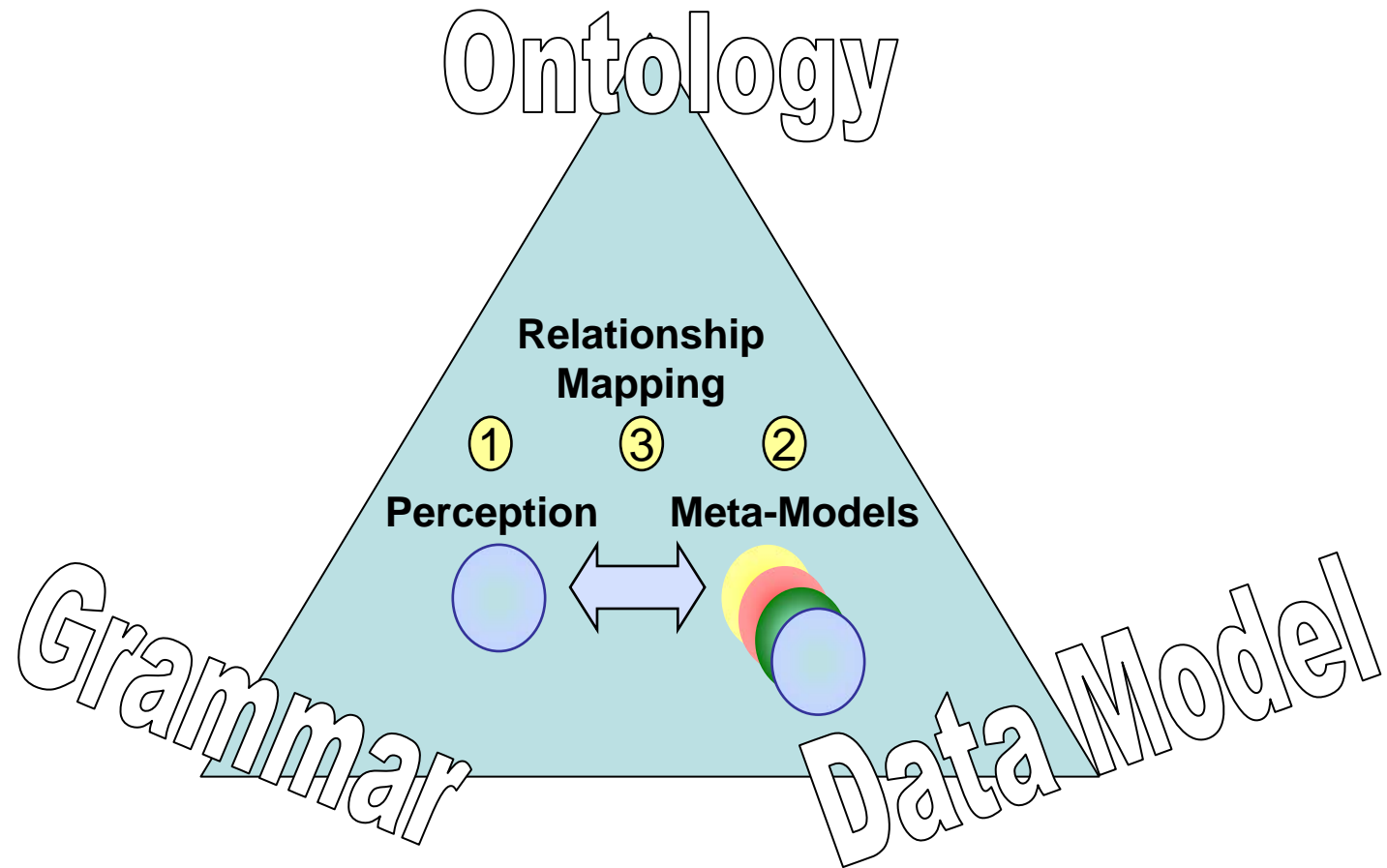
What is a System

- When talking about IT systems, we are talking about *Finite Automata*
  - Exchange is mapping of regular expressions (scope, resolution, structure)
  - Construction techniques can be used to define what finite automata can understand
    - Defined Sets
      - Enumeration of possibilities as a set
      - Data Modeling with relations
      - XML Schemas
    - Grammars
      - Production Rules
      - Vocabulary

J.L. Hein: *Discrete Structures, Logic, and Computability*. 2nd Edition. Jones & Bartlett, 2002

What does this mean for MSDL

- If we want to be consistent,
  - We need one common model from which all views can be derived
  - We need semantically loss-free translations between the views
- The challenge
  - An XML schema defines a finite automata
  - A Grammar defines a finite automata
  - If we combine both, we must make sure that the finite automata is free of contradictions

  We cannot simply combine independently developed XML schema, databases, and grammars and apply ad-hoc mediations between them without introducing inconsistencies

Recommendation

- Facts to consider
  - <u>XML Schemas</u> define an underlying data model
  - <u>Data models</u> capture the building blocks for languages unambiguously
  - <u>Regular grammars</u> introduce production rules to produce sentences that can be accepted by such a data model
  - ***Grammars and Data Models are equivalent regarding regular expressions***
  - <u>Non-regular grammars introduce</u> first helps to capture business rules (like the *constituent and functional structures* in the LFG*)*
  - <u>Ontologies</u> axiomize all business rules (as done in a prototype for JC3IEDM by IDA)
- Conclusion:
  - Decide on your underlying data model
  - Extend the other topics from there gradually

C2 Organization
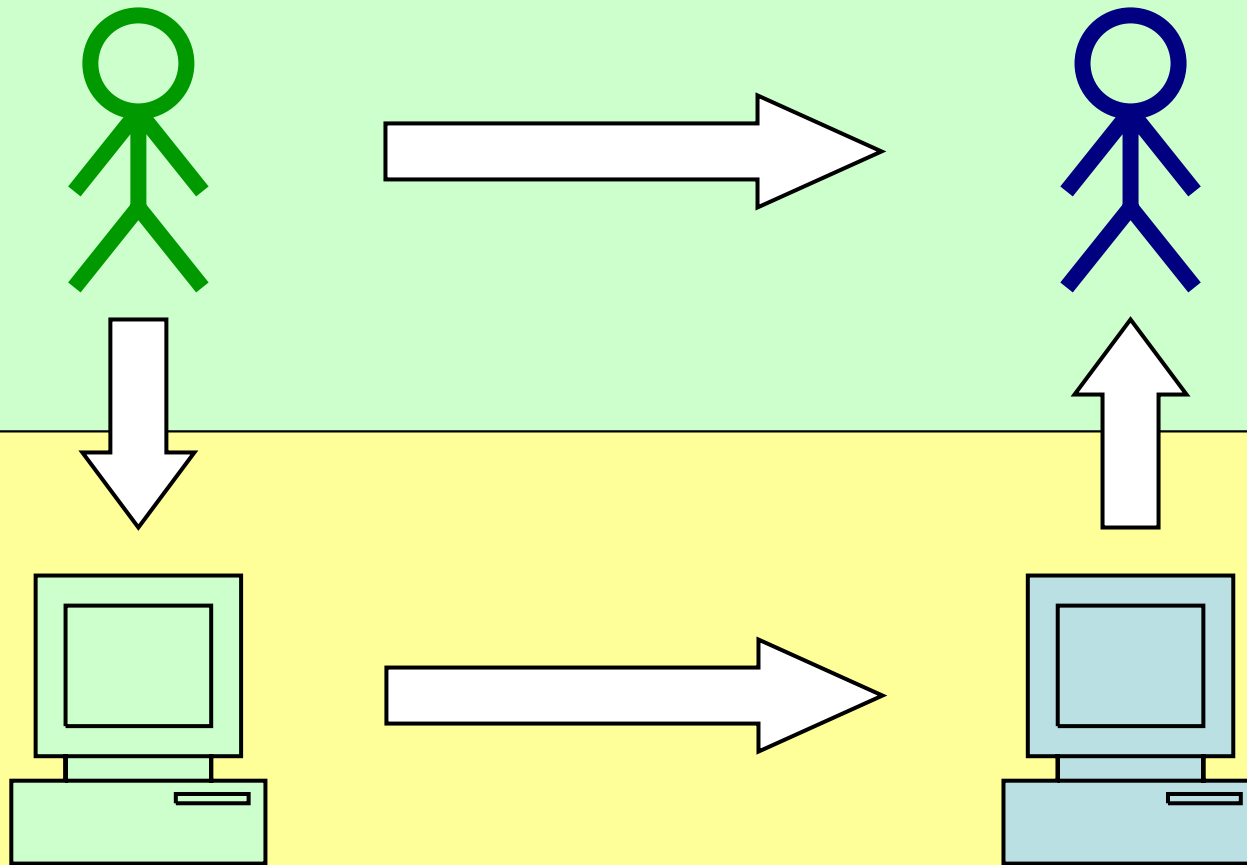
C2/IT Systems

Ash tek durbatulûk
ash tek gimbatul,
ash tek thrakatulûk
agh SISO-ishi krimpatul

One model to rule them all,
one model to find them,
one model to bring them all
and in SISO bind them