Theory Graphs and Meta-Logical/Grammatical Frameworks: MMT as a Logic/Language/World-Workbench

Michael Kohlhase & the KWARC Group

Professur für Wissensrepräsentation und -verarbeitung
Informatik, FAU Erlangen-Nürnberg
http://kwarc.info

14. Jan. 2019 – Logik-Kolloquium, Konstanz

1 Introduction & Motivation

About Humans and Computers in Mathematics

- Computers and Humans have complementary strengths.
  - Computers can handle large data and computations flawlessly at enormous speeds.
  - Humans can sense the environment, react to unforeseen circumstances and use their intuitions to guide them through only partially understood situations.

About Humans and Computers in Mathematics

- Computers and Humans have complementary strengths.
  - Computers can handle large data and computations flawlessly at enormous speeds.
  - Humans can sense the environment, react to unforeseen circumstances and use their intuitions to guide them through only partially understood situations.

- In mathematics: we exploit this, we
  - let humans explore mathematical theories and come up with novel insights/proofs,
  - delegate symbolic/numeric computation and typesetting of documents to computers.
  - (sometimes) delegate proof checking and search for trivial proofs to computers

About Humans and Computers in Mathematics

- Computers and Humans have complementary strengths.
  - Computers can handle large data and computations flawlessly at enormous speeds.
  - Humans can sense the environment, react to unforeseen circumstances and use their intuitions to guide them through only partially understood situations.
- In mathematics: we exploit this, we
  - let humans explore mathematical theories and come up with novel insights/proofs,
  - delegate symbolic/numeric computation and typesetting of documents to computers.
  - (sometimes) delegate proof checking and search for trivial proofs to computers
- Overlooked Opportunity: management of existing mathematical knowledge
  - cataloguing, retrieval, refactoring, plausibilization, change propagation and in some cases even application do not require (human) insights and intuition
  - can even be automated in the near future given suitable representation formats and algorithms.

About Humans and Computers in Mathematics

- Computers and Humans have complementary strengths.
  - Computers can handle large data and computations flawlessly at enormous speeds.
  - Humans can sense the environment, react to unforeseen circumstances and use their intuitions to guide them through only partially understood situations.
- In mathematics: we exploit this, we
  - let humans explore mathematical theories and come up with novel insights/proofs,
  - delegate symbolic/numeric computation and typesetting of documents to computers.
  - (sometimes) delegate proof checking and search for trivial proofs to computers
- Overlooked Opportunity: management of existing mathematical knowledge
  - cataloguing, retrieval, refactoring, plausibilization, change propagation and in some cases even application do not require (human) insights and intuition
  - can even be automated in the near future given suitable representation formats and algorithms.

  Math. Knowledge Management (MKM): is the discipline that studies this.
- Application: Scaling Math beyond the One-Brain-Barrier

The One-Brain-Barrier

- ***Observation 1.1.*** More than 10^5 math articles published annually in Math.
- ***Observation 1.2.*** The libraries of Mizar, Coq, Isabelle,... have $\sim 10^5$ statements+proofs each. (but are mutually incompatible)
- Consequence: humans lack overview over – let alone working knowledge in – all of math/formalizations. (Leonardo da Vinci was said to be the last who had)
- Dire Consequences: duplication of work and missed opportunities for the application of mathematical/formal results.

The One-Brain-Barrier

- ***Observation 1.1.*** More than 10^5 math articles published annually in Math.
- ***Observation 1.2.*** The libraries of Mizar, Coq, Isabelle,... have $\sim 10^5$ statements+proofs each. (but are mutually incompatible)
- Consequence: humans lack overview over – let alone working knowledge in – all of math/formalizations. (Leonardo da Vinci was said to be the last who had)
- Dire Consequences: duplication of work and missed opportunities for the application of mathematical/formal results.
- Problem: Math Information systems like `arXiv.org`, Zentralblatt Math, MathSciNet, etc. do not help (only make documents available)
- Fundamenal Problem: the One-Brain Barrier (OBB)
  - To become productive, math must pass through a brain
  - Human brains have limited capacity (compared to knowledge available online)

The One-Brain-Barrier

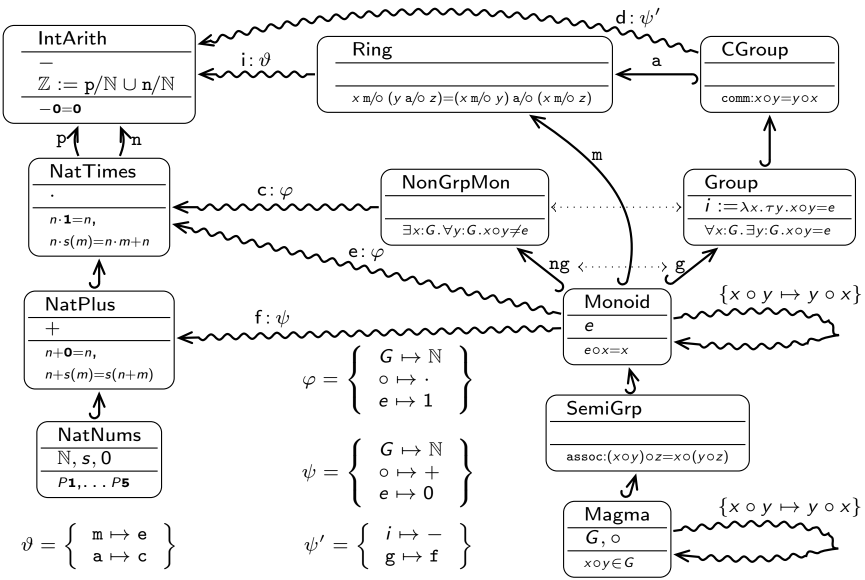
- ***Observation 1.1.*** More than 10^5 math articles published annually in Math.
- ***Observation 1.2.*** The libraries of Mizar, Coq, Isabelle,... have $\sim 10^5$ statements+proofs each. (but are mutually incompatible)
- Consequence: humans lack overview over – let alone working knowledge in – all of math/formalizations. (Leonardo da Vinci was said to be the last who had)
- Dire Consequences: duplication of work and missed opportunities for the application of mathematical/formal results.
- Problem: Math Information systems like `arXiv.org`, Zentralblatt Math, MathSciNet, etc. do not help (only make documents available)
- Fundamenal Problem: the One-Brain Barrier (OBB)
  - To become productive, math must pass through a brain
  - Human brains have limited capacity (compared to knowledge available online)
- Idea: enlist computers (large is what they are good at)
- Prerequisite: make math knowledge machine-actionable & foundation-independent (use MKM)

2 Modular Representation of Mathematics

Modular Representation of Math (Theory Graph)

- Idea: Follow mathematical practice of generalizing and framing
  - framing: If we can view an object a as an instance of concept B, we can inherit all of B properties (almost for free.)
  - state all assertions about properties as general as possible (to maximize inheritance)
  - examples and applications are just special framings.
- Modern expositions of Mathematics follow this rule (radically e.g. in Bourbaki)
- formalized in the theory graph paradigm (little/tiny theory doctrine)
  - theories as collections of symbol declarations and axioms (model assumptions)
  - theory morphisms as mappings that translate axioms into theorems
- **Example 2.1 (MMT: Modular Mathematical Theories).** MMT is a foundation-indepent theory graph formalism with advanced theory morphisms.
- Problem: With a proliferation of abstract (tiny) theories readability and accessibility suffers (one reason why the Bourbaki books fell out of favor)

Modular Representation of Math (MMT Example)



$$\varphi = \left\{ \begin{array}{l} G \mapsto \mathbb{N} \\ \circ \mapsto \cdot \\ e \mapsto 1 \end{array} \right\}$$

$$\psi = \left\{ \begin{array}{l} G \mapsto \mathbb{N} \\ \circ \mapsto + \\ e \mapsto 0 \end{array} \right\}$$

$$\vartheta = \left\{ \begin{array}{l} \mathtt{m} \mapsto \mathtt{e} \\ \mathtt{a} \mapsto \mathtt{c} \end{array} \right\}$$

$$\psi' = \left\{ \begin{array}{l} i \mapsto - \\ \mathtt{g} \mapsto \mathtt{f} \end{array} \right\}$$

Concrete MMT Syntax

- **Example 2.2 (A Theory and Type for Unital Magmas).**

```
theory Unital : base:?Logic =
    include ?Magma ❙

    theory unital_theory : base:?Logic =
        include ?Magma/magma_theory ❙
        unit : U | # e prec -1 ❙
        axiom_leftUnital : ⊦ prop_leftUnital op e ❙
        axiom_rightUnital : ⊦ prop_rightUnital op e ❙
    ❚
    unital = Mod unital_theory  ❙

    unitOf  : {G: unital} dom G | # %I1 e prec 5 | = [G] (G.unit) ❙
❚
```

where the following is imported with ?Magma

```
prop_leftUnital : {U : type} (U ⟶ U ⟶ U) ⟶ U ⟶ prop |
    = [U,op,e] ∀[x] op e x ≐ x | # prop_leftUnital 2 3 ❙
prop_rightUnital : {U : type} (U ⟶ U ⟶ U) ⟶ U ⟶ prop |
    = [U,op,e] ∀[x] op x e ≐ x | # prop_rightUnital 2 3 ❙
```

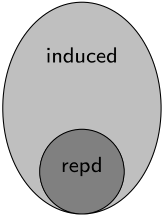
The MMT Module System

- Central notion: theory graph with theory nodes and theory morphisms as edges
- **Definition 2.3.** In MMT, a theory is a sequence of constant declarations – optionally with type declarations and definitions
- MMT employs the Curry/Howard isomorphism and treats
  - axioms/conjectures as typed symbol declarations (propositions-as-types)
  - inference rules as function types (proof transformers)
  - theorems as definitions (proof terms for conjectures)
- **Definition 2.4.** MMT had two kinds of theory morphisms
  - structures instantiate theories in a new context (also called: definitional link, import) they import of theory S into theory T induces theory morphism $S \to T$
  - views translate between existing theories (also called: postulated link, theorem link) views transport theorems from source to target (framing)
- together, structures and views allow a very high degree of re-use
- **Definition 2.5.** We call a statement t induced in a theory T, iff there is
  - a path of theory morphisms from a theory S to T with (joint) assignment σ,
  - such that $t = \sigma(s)$ for some statement s in S.
- In MMT, all induced statements have a canonical name, the MMT URI.

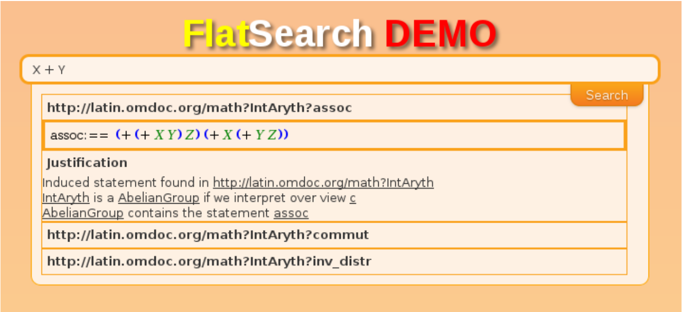
♭search on the LATIN Logic Atlas

- Flattening the LATIN Atlas (once):

type	modular	flat	factor
declarations	2310	58847	25.4
library size	23.9 MB	1.8 GB	14.8
math sub-library	2.3 MB	79 MB	34.3
`MathWebSearch` harvests	25.2 MB	539.0 MB	21.3



- simple ♭search frontend at `http://cds.omdoc.org:8181/search.html`



Applications for Theories in Physics

- Theory Morphisms allow to "view" source theory in terms of target theory.
- Theory Morphisms occur in Physics all the time.

Theory	Temp. in Kelvin	Temp. in Celsius	Temp. in Fahrenheit
Signature	$^\circ$K	$^\circ$C	$^\circ$F
Axiom:	absolute zero at 0°K	Water freezes at 0°C	cold winter night: 0°F
Axiom:	$\delta(^\circ K1) = \delta(^\circ C1)$	Water boils at 100°C	domestic pig: 100°F
Theorem:	Water freezes at 271.3°K	domestic pig: 38°C	Water boils at 170°F
Theorem:	cold winter night: 240°K	absolute zero at -271.3°C	absolute zero at -460°F

Views: $^\circ C \stackrel{+271.3^\circ}{\longrightarrow}$ K, $^\circ C \stackrel{-32/2^\circ}{\longrightarrow}$ F, and $^\circ F \stackrel{+240/2^\circ}{\longrightarrow}$ K, inverses.

- Other Examples: Coordinate Transformations,
- Application: Unit Conversion: apply view morphism (flatten) and simplify with UOM. (For new units, just add theories and views.)
- Application: MathWebSearch on flattened theory (Explain view path)

3 Foundational Pluralism (the Meta-Meta Level)

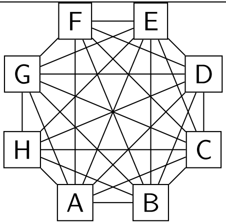
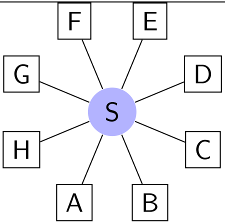
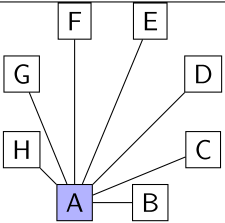
Assembling a Global Knowlege Resource (Problems)

- Problems: encountered in practice
  - Different systems have different, mutually incompatible logical/mathematical foundations (hundreds, optimize different aspects)
  - the respective communities are largely disjoint
  - have built large, incompatible, but mathematically overlapping libraries
  - all tools lack crucial features (cannot afford to develop)
  - new logics/foundations/systems seldom get off the ground (too expensive)
- **Definition 3.1.** A foundation (of mathematics) consists of
  - a foundational language (e.g. first-order logic or the calculus of constructions)
  - a foundational theory (e.g. axiomatic set theory)

  Observation: need a system that can deal with multiple foundations $\rightsquigarrow$ foundational pluralism

Realizing Foundational Pluralism

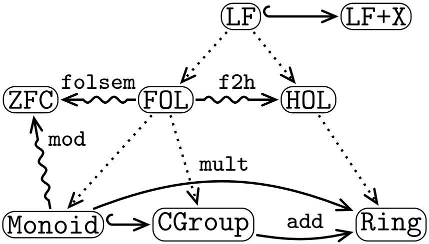
- Towards Integration at the Foundation Level:

peer to peer	open standard	industry standard
		
$n^2/2$ translations symmetric	$2n$ translations symmetric	$2n - 2$ translations asymmetric

- Problem: So far So Obvious! But what should be in the middle?
- Idea (reused): A modular representation of foundations (logics/theories)
  Bring-Your-Own-Foundation $\leadsto$ foundation independent systems/tools

Representing Logics and Foundations as Theories

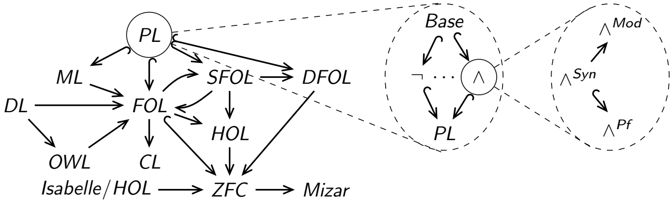
- **Example 3.2.** Logics and foundations represented as MMT theories



- **Definition 3.3.** Meta-relation between theories – special case of inclusion
- Uniform Meaning Space: morphisms between formalizations in different logics become possible via meta-morphisms.
- **Remark 3.4.** Semantics of logics as views into foundations, e.g., `folsem`.
- **Remark 3.5.** Models represented as views into foundations (e.g. ZFC)
- **Example 3.6.** $\mathtt{mod} := \{G \mapsto \mathbb{Z}, \circ \mapsto +, e \mapsto 0\}$ interprets Monoid in ZFC.

The LATIN Logic Atlas

- **Definition 3.7.** The LATIN project (Logic Atlas and Integrator)
- Idea: Provide a standardized, well-documented set of theories for logical languages, logic morphisms as theory morphisms.



- Technically: Use MMT as a representation language logics-as-theories
- Integrate logic-based software systems via views.
- State: ∼ 1000 modules (theories and morphisms) written in MMT/LF [RS09]

MMT a Module System for Mathematical Content

- MMT: Universal representation language for formal mathematical/logical content
- Implementation: MMT API with generic
  - module system for math libraries, logics, foundations
  - parsing + type reconstruction + simplification
  - IDEs (web server + JEdit+IntelliJ)
  - change management
- Continuous development since 2007 (> 30000 lines of Scala code)
- Close relatives:
  - LF, Isabelle, Dedukti: but flexible choice of logical framework
  - Hets: but declarative logic definitions

Concrete MMT Syntax: Propositional Logic

- **Example 3.8 (Propositional Logic (Syntax)).**

```
theory PropLogSyntax : ur:?LF =
    prop : type | # bool ❙

    and     : bool ⟶ bool ⟶ bool      | # 1 ∧ 2    prec 45 ❙ /T jwedge ❙
    not     : bool ⟶ bool             | # ¬ 1      prec 50 ❙ /T jneg   ❙

    or      : bool ⟶ bool ⟶ bool | # 1 ∨ 2   prec 40 |
            = [a,b] ¬ (¬ a ∧ ¬ b) ❙ /T jvee ❙
    implies : bool ⟶ bool ⟶  bool | # 1 ⇒ 2 prec 35 |
            = [a,b] ¬ a ∨ b ❙ /T jrA     ❙

    iff : bool ⟶ bool ⟶ bool | # 1 ⇔ 2 prec 40 | = [a,b] (a ⇒ b) ∧ (b ⇒ a) ❙

    true : bool | # ⊤ ❙ /T jtop ❙
    false : bool | = ¬ ⊤ | # ⊥ ❙ /T jbot    ❙
❚
```

Concrete MMT Syntax: Propositional Natural Deduction

- **Example 3.9 (Propositional Logic (Natural Deduction)).**

```
theory PropLogNatDed : ur:?LF =
    include ?PropLogSyntax ❙

    ded : bool ⟶ type | # ⊦ 1 prec 1 ❙ /T jvdash  ❙

    andEl   : {A,B} ⊦ A ∧ B ⟶ ⊦ A | # andEl 3 ❙
    andEr   : {A,B} ⊦ A ∧ B ⟶ ⊦ B | # andEr 3 ❙
    andI    : {A,B} ⊦ A ⟶ ⊦ B ⟶ ⊦ A ∧ B | # andI 3 4 ❙

    implI   : {A,B} (⊦ A ⟶ ⊦ B) ⟶ ⊦ A ⇒ B | # implI 3 ❙
    implE   : {A,B} ⊦ A ⇒ B ⟶ ⊦ A ⟶ ⊦ B | # implE 3 4 ❙

    orIl        : {A,B} ⊦ A ⟶ ⊦ A ∨ B | # orIl 3 ❙
    orIr        : {A,B} ⊦ B ⟶ ⊦ A ∨ B | # orIr 3 ❙
    orE     : {A,B,C} ⊦ A ∨ B ⟶ (⊦ A ⟶ ⊦ C) ⟶ (⊦ B ⟶ ⊦ C) ⟶ ⊦ C | # orE 4 5 6❙

    notI        : {A} (⊦ A ⟶ ⊦ ⊥) ⟶ ⊦ ¬A | # notI 2 ❙
    notE    : {A} ⊦ ¬¬A ⟶ ⊦ A | # notE 2 ❙

❚
```

- **Example 3.10 (Propositional Logic (Natural Deduction)).**

```
theory Proofs : ?PropLogNatDed =
    conjComm     : {A,B} ⊦ A ∧ B ⇒ B ∧ A |
                           = [A,B] implI ([ab] andI (andEr ab) (andEl ab)) ❙
❚
```

Concrete MMT Syntax: First-Order Logic (Syntax)

- **Example 3.11 (First-Order Logic (Syntax)).**

```
theory FOLSyntax : ur:?LF =
    include ?PropLogSyntax ❙

    ind : type | # ι ❙ /T jiota ❙

    forall  : (ι ⟶ bool) ⟶ bool | # ∀ 1 prec 55 ❙
    exists  : (ι ⟶ bool) ⟶ bool | # ∃ 1 prec 60 |
                  = [P] ¬ ∀ [x] ¬ (P x)❙ /T jexists   ❙

    // existsUnique : ??? | = ??? | # ∃! 1 prec 65 ❙
❚

theory FOLEQSyntax : ur:?LF =
  include ?FOLSyntax ❙
    equality : ι ⟶ ι ⟶ bool | # 1 ≐ 2 prec 65 ❙
❚
```

Concrete MMT Syntax: First-Order Natural Deduction

- **Example 3.12 (First-Order Logic (Natural Deduction)).**

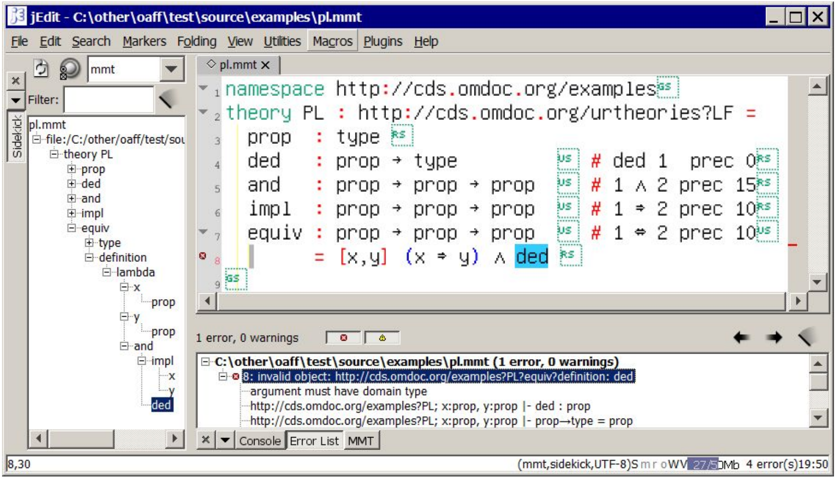
```
theory FOLNatDed : ur:?LF =
    include ?FOLSyntax ❙
    include ?PropLogNatDed ❙

    forallI : {P} ({y : ι} ⊦ P y) ⟶ ⊦ ∀ [x] P x | # forallI 2 ❙
    forallE : {P,B} ⊦ (∀ [x] P x) ⟶ ⊦ P B | # forallE 3 ❙
    /T Everytime you write $∀ P$, somewhere a unicorn cries ❙

    existsI : {P,c} ⊦ (P c) ⟶ ⊦ ∃ [x] P x | # existsI 3 ❙
    existsE : {P,B} ⊦ (∃ [x] P x) ⟶ ({c} ⊦ P c ⟶ ⊦ B) ⟶ ⊦ B | # existsE 3 4 ❙
❚
```

4 MMT Software Eosystem

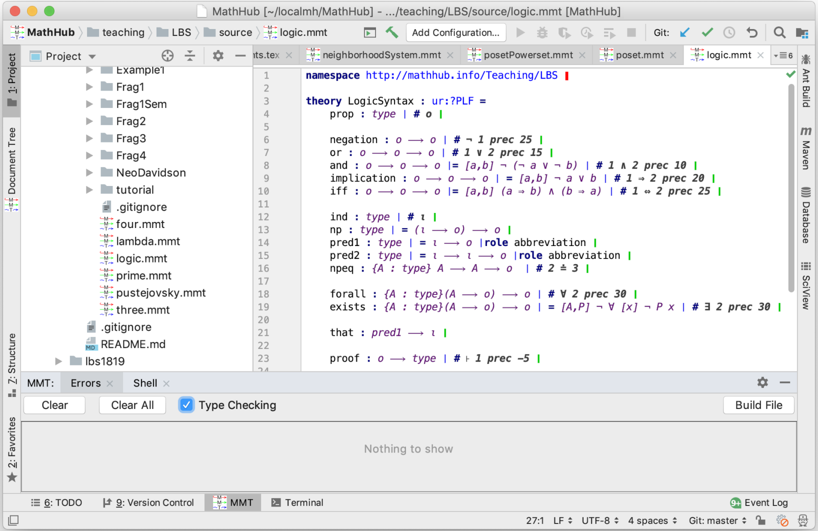
MMT a Module System for Mathematical Content

- MMT: Universal representation language for formal mathematical/logical content
- Implementation: MMT API with generic
  - module system for math libraries, logics, foundations
  - parsing + type reconstruction + simplification
  - IDEs (web server + JEdit+IntelliJ)
  - change management
- Continuous development since 2007 (> 30000 lines of Scala code)
- Close relatives:
  - LF, Isabelle, Dedukti: but flexible choice of logical framework
  - Hets: but declarative logic definitions

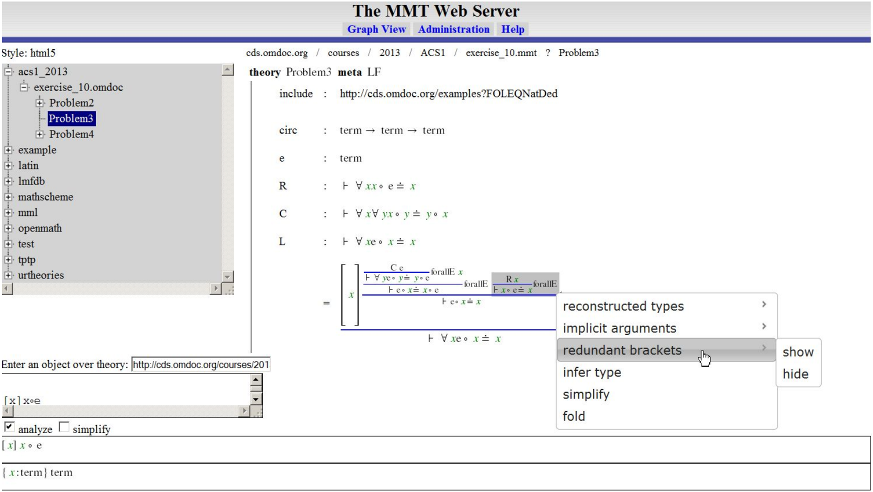
MMT API JEdit Integration (IDE)



MMT API IntelliJ (IDE)



MMT API Browser Integration

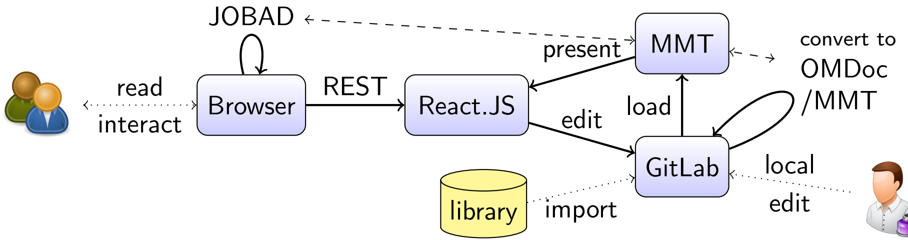


MathHub: A Portal and Archive of Flexiformal Maths

- Idea: learn from the open source community, offer a code repository with management support that acts as a hub for publication/development projects.
- MathHub: a collaborative development/hosting/publishing system of open-source, formal/informal math. (See http://mathhub.info)

MathHub: A Portal and Archive of Flexiformal Maths

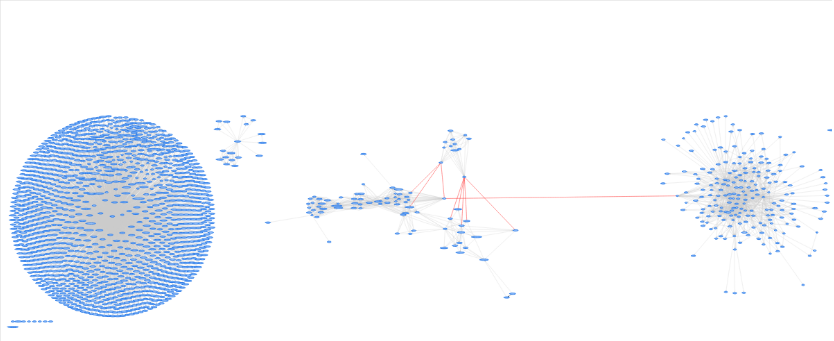
- Idea: learn from the open source community, offer a code repository with management support that acts as a hub for publication/development projects.
- MathHub: a collaborative development/hosting/publishing system of open-source, formal/informal math. (See `http://mathhub.info`)
- MathHub Architeture: Three core components (meet requirements above)
  - Representation: *OMDoc*/MMT mechanized by the MMT system.
  - Repositories: GitLab (git-based public/private repositories)
  - Front-End: React.JS (all content served by MMT)



TGView/TGView3D: Flexible Interaction with Theory Graphs

- **Definition 4.1.** TGView is a flexible facility for viewing and interacting with (theory) graphs in MathHub.
  - TGView gives access to MathHub libraries (user selects graph)
  - MMT API generates JSON graph representation
  - TGView draws graph to Browser canvas (via the vis.js library)

  TGView3D is a VR version for the Oculus Rift.
- **Example 4.2 (CAS Interfaces, MitM Ontology, and Alignments).**



5 MMT+GF as a Natural Language Semantics Workbench

Meaning of Natural Language; e.g. Machine Translation

- Idee: Machine Translation is very simple! (we have good lexica)
- **Example 5.1.** *Peter liebt Maria.* ⇝ *Peter loves Mary.*
- this only works for simple examples
- **Example 5.2.** *Wirf der Kuh das Heu über den Zaun.* ⇝̸ *Throw the cow the hay over the fence.* (differing grammar; Google Translate)
- **Example 5.3.** Grammar is not the only problem
  - *Der Geist ist willig, aber das Fleisch ist schwach!*
  - *Der Schnaps ist gut, aber der Braten ist verkocht!*
- We have to understand the meaning!

Language and Information

- Observation: Humans use words (sentences, texts) in natural languages to represent and communicate information.
- But: what really counts is not the words themselves, but the meaning information they carry.

Language and Information

- Observation: Humans use words (sentences, texts) in natural languages to represent and communicate information.
- But: what really counts is not the words themselves, but the meaning information they carry.
- **Example 5.4.**

  Zeitung $\leadsto$ 

  The New York Times

  NEW YORK, THURSDAY, NOVEMBER 11, 2004

  RECITAL REVIEW

  Offering Beethoven With Crisp Lines, but Softened by 'Moonlight'

  By ALLAN KOZINN

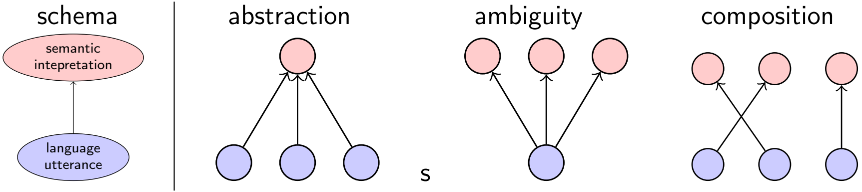
- for questions/answers, it would be very useful to find out what words (sentences/texts) mean.

Language and Information

- Observation: Humans use words (sentences, texts) in natural languages to represent and communicate information.
- But: what really counts is not the words themselves, but the meaning information they carry.
- **Example 5.4.**

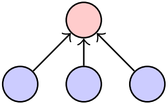
  Zeitung ⇝ 


- for questions/answers, it would be very useful to find out what words (sentences/texts) mean.
- Interpretation of natural language utterances: three problems



Language and Information (Examples)

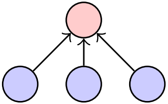
- **Example 5.5 (Abstraction).**



car and *automobile* have the same meaning

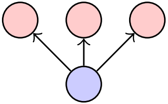
Language and Information (Examples)

- **Example 5.5 (Abstraction).**



car and *automobile* have the same meaning

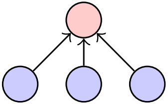
- **Example 5.6 (Ambiguity).**



a *bank* can be a financial institution or a geographical feature

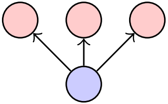
Language and Information (Examples)

- **Example 5.5 (Abstraction).**



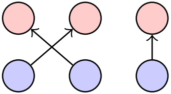
car and *automobile* have the same meaning

- **Example 5.6 (Ambiguity).**



a *bank* can be a financial institution or a geographical feature

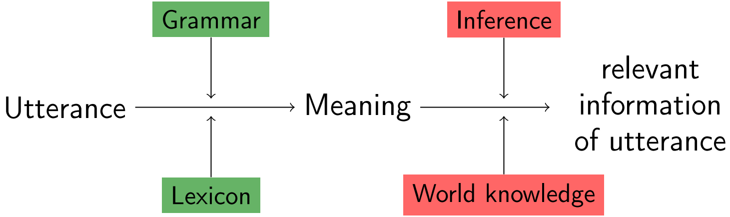
- **Example 5.7 (Composition).**



Every student sleeps $\leadsto \forall x.student(x) \Rightarrow sleep(x)$

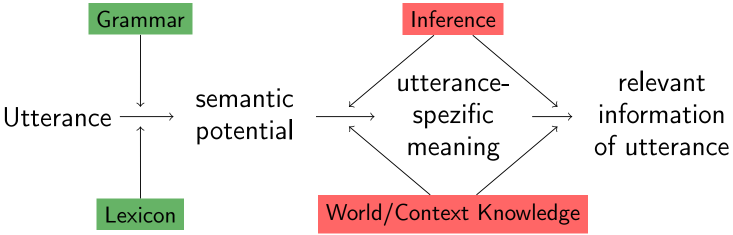
Context Contributes to the Meaning of NL Utterances

- Observation: Not all information conveyed is linguistically realized in an utterance.
- **Example 5.8.** *The lecture begins at 11:00 am.* What lecture? Today?
- **Definition 5.9.** We call a piece i of information linguistically realized in an utterance U, iff, we can trace i to a fragment of U.
- Possible Mechanism: Inference



Context Contributes to the Meaning of NL Utterances

- **Example 5.10.** *It starts at eleven.* What starts?
- Before we can resolve the time, we need to resolve the anaphor *it*.
- Possible Mechanism: More Inference!



What is the State of the Art In NLU?

- Two avenues of attack for the problem: knowledge-based and statistical techniques (they are complementary)

Deep	Knowledge-based We are here	Not there yet cooperation?
Shallow	no-one wants this	Statistical Methods applications
Analysis ↑ vs. Coverage →	narrow	wide

- We will cover foundational methods of deep processing in the course and a mixture of deep and shallow ones in the lab.

Environmental Niches for both Approaches to NLU

- There are two kinds of applications/tasks in NLU
  - consumer-grade applications have tasks that must be fully generic, and wide coverage (e.g. machine translation $\leadsto$ Google Translate)
  - producer-grade applications must be high-precision, but domain-adapted (multilingual documentation, voice-control, ambulance translation)

Precision

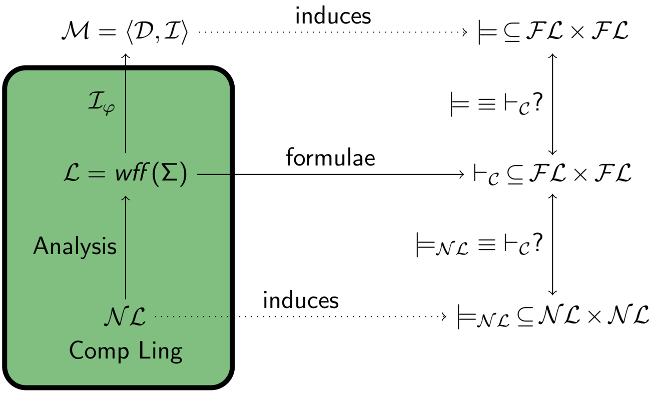
100% Producer Tasks

50% Consumer Tasks

$10^{3\pm1}$ Concepts $10^{6\pm1}$ Concepts **Coverage**

- A producer domain I am interested in: Mathematical/Technical documents

Natural Language Semantics?



Structural Grammar Rules

- **Definition 5.11.** Fragment 1 knows the following eight syntactical categories

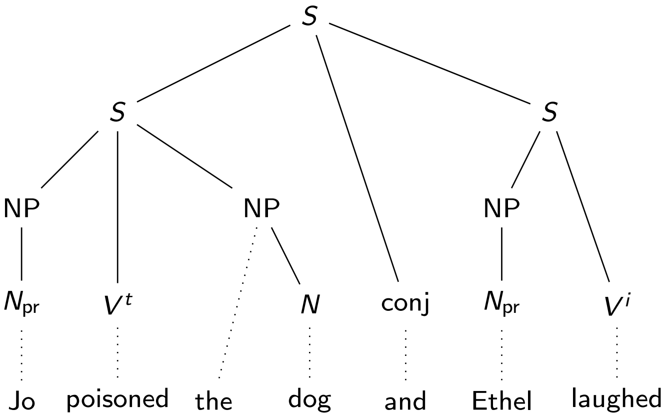
S	sentence	NP	noun phrase
N	noun	N_{pr}	proper name
V^i	intransitive verb	V^t	transitive verb
conj	connective	Adj	adjective

- **Definition 5.12.** We have the following grammar rules in fragment 1.

S1.	S	$\rightarrow$	NP V^i
S2.	S	$\rightarrow$	NP V^t NP
N1.	NP	$\rightarrow$	N_{pr}
N2.	NP	$\rightarrow$	the$N$
S3.	S	$\rightarrow$	It is not the case that S
S4.	S	$\rightarrow$	S conj S
S5.	S	$\rightarrow$	NP is NP
S6.	S	$\rightarrow$	NP is Adj.

Syntax Example: *Jo poisoned the dog and Ethel laughed*

- ***Observation 5.13.*** *Jo poisoned the dog and Ethel laughed* is a sentence of fragment 1
- We can construct a syntax tree for it!



Concrete MMT Syntax: Propositional Logic

- **Example 5.14 (Propositional Logic (Syntax)).**

```
theory PropLogSyntax : ur:?LF =
    prop : type | # bool ❙

    and     : bool ⟶ bool ⟶ bool     | # 1 ∧ 2    prec 45 ❙ /T jwedge ❙
    not     : bool ⟶ bool            | # ¬ 1      prec 50 ❙ /T jneg   ❙

    or      : bool ⟶ bool ⟶ bool | # 1 ∨ 2   prec 40 |
            = [a,b] ¬ (¬ a ∧ ¬ b) ❙ /T jvee ❙
    implies : bool ⟶ bool ⟶  bool | # 1 ⇒ 2 prec 35 |
            = [a,b] ¬ a ∨ b ❙ /T jrA     ❙

    iff : bool ⟶ bool ⟶ bool | # 1 ⇔ 2 prec 40 | = [a,b] (a ⇒ b) ∧ (b ⇒ a) ❙

    true : bool | # ⊤ ❙ /T jtop ❙
    false : bool | = ¬ ⊤ | # ⊥ ❙ /T jbot    ❙
❚
```

Domain Theories for Fragment 1 (Lexicon)

- A "lexicon theory" (only selected constants here)

```
theory frag1Lex : ?plngd =
      meta ?gfmeta?correspondsTo `frag1Lex.pgf ❙
      Ethel_NP : ι ❙
      book_N : pred1 ❙
      sing_V : pred1 ❙
      read_V2 : pred2 ❙
      happy_A : pred1 ❙
❚
```

  declares one logical constant for each from abstract GF grammar (automation?)
- Extend by axioms that encode background knowledge about the domain
- **Example 5.15 (What makes you sing).**

```
      happy_sing : ⊦ ∀[x] happy x ⇒ sing x ❙
      read_happy : ⊦ ∀[x] (∃[y] book y ∧ read x y) ⇒ happy x ❙
```

Hello World Example for GF (Syntactic)

- **Example 5.16 (A Hello World Grammar).**

abstract zero = {
  flags startcat=O;
  cat
    S ; NP ; V2 ;
  fun
    spo : V2 −> NP −> NP −> S ;
    John, Mary : NP ;
    Love : V2 ;
}
	concrete zeroEng **of** zero = {
	  lincat
	    S, NP, V2 = Str ;
	  lin
	    spo vp s o = s ++ vp ++ o;
	    John = *"John"* ;
	    Mary = *"Mary"* ;
	    Love = *"loves"* ;
	}

- parse a sentence in gf: parse "John loves Mary" $\leadsto$ Love John Mary

Hello World Example for GF (Syntactic)

- **Example 5.16 (A Hello World Grammar).**

<pre>**abstract** zero = {
  flags startcat=O;
  cat
    S ; NP ; V2 ;
  fun
    spo : V2 −> NP −> NP −> S ;
    John, Mary : NP ;
    Love : V2 ;
}</pre>
	<pre>**concrete** zeroEng **of** zero = {
	  lincat
	    S, NP, V2 = Str ;
	  lin
	    spo vp s o = s ++ vp ++ o;
	    John = *"John"* ;
	    Mary = *"Mary"* ;
	    Love = *"loves"* ;
	}</pre>

- Make a French grammar with John="Jean"; Mary="Marie"; Love="aime";
- parse a sentence in gf: parse "John loves Mary" $\leadsto$ Love John Mary

Hello World Example for GF (Syntactic)

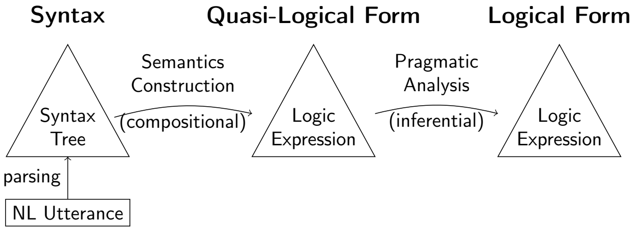
- **Example 5.16 (A Hello World Grammar).**

abstract zero = {
  flags startcat=O;
  cat
    S ; NP ; V2 ;
  fun
    spo : V2 −> NP −> NP −> S ;
    John, Mary : NP ;
    Love : V2 ;
}
	concrete zeroEng **of** zero = {
	  lincat
	    S, NP, V2 = Str ;
	  lin
	    spo vp s o = s ++ vp ++ o;
	    John = *"John"* ;
	    Mary = *"Mary"* ;
	    Love = *"loves"* ;
	}

- Make a French grammar with John="Jean"; Mary="Marie"; Love="aime";
- parse a sentence in gf: parse "John loves Mary" ⇝ Love John Mary
- linearize in gf: linearize Love John Mary ⇝ John loves Mary
- translate in in gf: parse −lang=Eng "John Loves Mary"| linearize −lang=Fre
- generate random sentences to test:
  generate_random −number=10 | linearize −lang=Fre ⇝ Jean aime Marie

Embedding GF into MMT

- Observation: GF provides Java bindings and MMT is programed in Scala, which compiles into the Java virtual machine.
- Idea: Use GF as a sophisticated NL-parser/generator for MMT
  - ⇝ MMT with a natural language front-end.
  - ⇝ GF with a multi-logic back-end
- **Definition 5.17.** The GF/MMT integration mapping interprets GF abstract syntax trees as MMT terms.
- Observation: This fits very well with our interpretation process in LBS



- Implementation: transform GF (Java) data structures to MMT (Scala) ones

Correspondence between GF Grammars and MMT Theories

- Idea: We can make the GF/MMT integration mapping essentially the identity.
- Prerequisite: MMT theory isomorphic to GF grammar (declarations aligned)
- Mechanism: use the MMT metadata mechanism
  - symbol `correspondsTo` in metadata theory `gfmeta` specifies relation
  - import `?gfmeta` into domain theories
  - meta keyword for "metadata relation whose subject is this theory".
  - object is MMT string literal ‘grammar.pgf.

```
theory gfmeta : ur:?LF = correspondsTo ❙ ❚

theory plnqd : ur:?LF =
include ?gfmeta
  meta ?gfmeta?correspondsTo `grammar.pgf ❙
```

- Observation: GF grammars and MMT theories best when organized modularly.
- Best Practice: align "grammar modules" and "little theories" modularly.

6 OMDoc/MMT in Argumentation Theory

6.1 Introduction: Argumentation Theory [adapted from Sarah Gaggl]

Argumentation is Ubiquitous

- Observation: We exchange arguments in politics, in court, when making decisions, and in science

Argumentation is Ubiquitous

- Observation: We exchange arguments in politics, in court, when making decisions, and in science
- Questions: But what is argumentation? Can we model/decide arguments?

Argumentation is Ubiquitous

- Observation: We exchange arguments in politics, in court, when making decisions, and in science
- Questions: But what is argumentation? Can we model/decide arguments?
- **Example 6.1.** Is this Argumentation?

Background: SPP 1999 RATIO & Project ALMANAC

- DFG Schwerpunktprogramm (SPP) 1999 (established 2017)
  - RATIO: Robust Argumentation Machines (2018-20; 2021-23)
  - Going from mere facts to coherent argumentative structures as information units for decision-making
  - Areas involved: semantic web, computational linguistics, information retrieval, Logic, human/computer interaction.

Background: SPP 1999 RATIO & Project ALMANAC

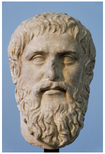
- DFG Schwerpunktprogramm (SPP) 1999 (established 2017)
  - RATIO: Robust Argumentation Machines (2018-20; 2021-23)
  - Going from mere facts to coherent argumentative structures as information units for decision-making
  - Areas involved: semantic web, computational linguistics, information retrieval, Logic, human/computer interaction.
- ca. 12 projects, (see http://spp-ratio.de)
- method interoperability by joint data sets and use case (Hackathons)

Background: SPP 1999 RATIO & Project ALMANAC

- DFG Schwerpunktprogramm (SPP) 1999 (established 2017)
  - RATIO: Robust Argumentation Machines (2018-20; 2021-23)
  - Going from mere facts to coherent argumentative structures as information units for decision-making
  - Areas involved: semantic web, computational linguistics, information retrieval, Logic, human/computer interaction.
- ca. 12 projects, (see `http://spp-ratio.de`)
- method interoperability by joint data sets and use case (Hackathons)
- ALMANAC: Argumentation Logics Manager & Argument Context Graph,
  - WA1: Atlas of Argumentation Logics (representing/organizing logics in LF)
  - WP2: Context Graphs for Argumentation (Theory Graphs for Multi-Agent-Logic)
  - WP3: Archiving & Managing Argumentation Logis (MathHub.info)

Argumentation in History

- **Definition 6.2 (Plato's Dialectic).**
  The dialectical method is discourse between two or more people holding different points of view about a subject, who wish to establish the truth of the matter guided by reasoned arguments. (The Republic (Plato), 348b)

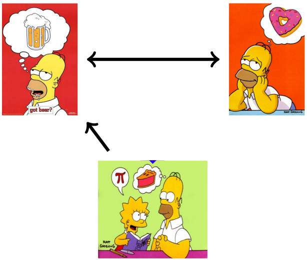


- **Definition 6.3 (Leibniz' Dream).**
  The only way to rectify our reasonings is to make them as tangible as those of the Mathematicians, so that we can find our error at a glance, and when there are disputes among persons, we can simply say: Let us calculate [Calculemus!], without further ado, to see who is right. (Leibniz, Gottfried Wilhelm, The Art of Discovery 1685, Wiener 51)

Abstract Argumentation Systems

- Abstract Argumentation [Dung, 1995]:
  - In abstract argumentation frameworks (AAFs) statements (called arguments) are formulated together with a relation (attack) between them.
  - Abstraction from the internal structure of the arguments.
  - The conflicts between the arguments are resolved on the semantical level.
- **Example 6.4.**

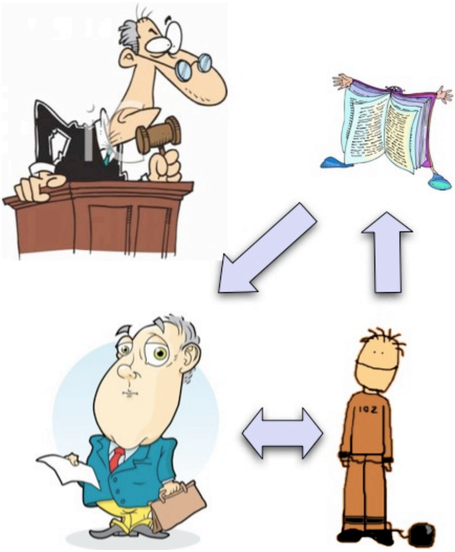
Situation



Result

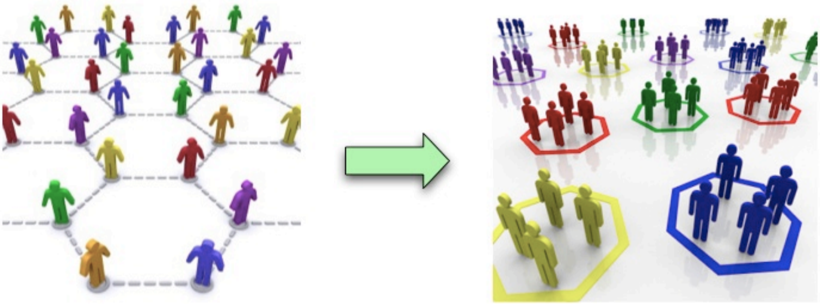
;

Legal Reasoning



Decision Support

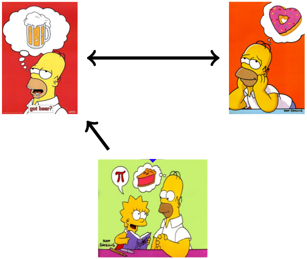
Social Networks



The Problem with Abstract Argumentation Systems

- Abstract Argumentation [Dung, 1995]:
  - In abstract argumentation frameworks (AAFs) statements (called arguments) are formulated together with a relation (attack) between them.
  - Abstraction from the internal structure of the arguments.
  - The conflicts between the arguments are resolved on the semantical level.
- **Example 6.5.**

Situation



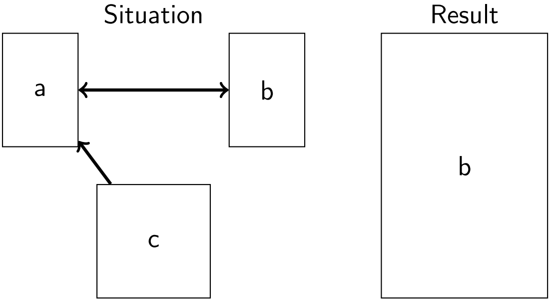
Result

;

The Problem with Abstract Argumentation Systems

- Abstract Argumentation [Dung, 1995]:
  - In abstract argumentation frameworks (AAFs) statements (called arguments) are formulated together with a relation (attack) between them.
  - Abstraction from the internal structure of the arguments.
  - The conflicts between the arguments are resolved on the semantical level.

Example 6.6.



Robust Representation of Individual Inference

- Idea: To represent arguments, we need to represent everyday reasoning.
- *There is a logic for that*! (actually many many of them)

Robust Representation of Individual Inference

- Idea: To represent arguments, we need to represent everyday reasoning.
- *There is a logic for that*! (actually many many of them)
- Robust Representation of Individual Inference (usually "philosophical logics")
  - *(multi-)modal logics* extend classical logic by notions of *possibility* and *necessity*.
  - *Preference logic* allows for stating sentences of the form "A is better/worse than B". [Han02]
  - *Relevance logic* restricts the classical (i.e. material) implication to protect from implications between seemingly disconnected premises and conclusions,[DR02].
  - other *paraconsistent logics*, which try to deal with inconsistency in a non-fatal manner by systematically avoiding *ex falso quodlibet*.
  - *Temporal logics* allow for reasoning about time (e.g. "X is true at time t_0"), [Bur84],
  - *probabilistic logics* about probabilities. [Nil86].
  - *Dynamic Logics* to model all kinds of anaphora

Robust Representation of Individual Inference

- Idea: To represent arguments, we need to represent everyday reasoning.
- *There is a logic for that*! (actually many many of them)
- Robust Representation of Individual Inference (usually "philosophical logics")
  - *(multi-)modal logics* extend classical logic by notions of *possibility* and *necessity*.
  - *Preference logic* allows for stating sentences of the form "A is better/worse than B". [Han02]
  - *Relevance logic* restricts the classical (i.e. material) implication to protect from implications between seemingly disconnected premises and conclusions,[DR02].
  - other *paraconsistent logics*, which try to deal with inconsistency in a non-fatal manner by systematically avoiding *ex falso quodlibet*.
  - *Temporal logics* allow for reasoning about time (e.g. "X is true at time t_0"), [Bur84],
  - *probabilistic logics* about probabilities. [Nil86].
  - *Dynamic Logics* to model all kinds of anaphora
- Proof Theory: Most logics have a natural-deduction-style calculus, some even machine-oriented calculi.
- Model Theory: mostly modal $\leadsto$ possible worlds semantics

Robust Representation of Individual Inference

- Idea: To represent arguments, we need to represent everyday reasoning.
- *There is a logic for that*! (actually many many of them)
- Robust Representation of Individual Inference (usually "philosophical logics")
  - *(multi-)modal logics* extend classical logic by notions of *possibility* and *necessity*.
  - *Preference logic* allows for stating sentences of the form "A is better/worse than B". [Han02]
  - *Relevance logic* restricts the classical (i.e. material) implication to protect from implications between seemingly disconnected premises and conclusions,[DR02].
  - other *paraconsistent logics*, which try to deal with inconsistency in a non-fatal manner by systematically avoiding *ex falso quodlibet*.
  - *Temporal logics* allow for reasoning about time (e.g. "X is true at time t_0"), [Bur84],
  - *probabilistic logics* about probabilities. [Nil86].
  - *Dynamic Logics* to model all kinds of anaphora
- Proof Theory: Most logics have a natural-deduction-style calculus, some even machine-oriented calculi.
- Model Theory: mostly modal $\leadsto$ possible worlds semantics
- Interoperability Problem: Most logics are "formally unrelated", incomparable (evaluation?, duplicated work)

6.2 Work Area 2: Context Graphs for Argumentation

Deep Modeling of Argumentation in STEM Settings

- Observation: Much of the wealth and prospects of central European Countries are based on STEM knowledge. (laid down in technical documents)
- STEM documents often have a non-trivial argumentation structure

Deep Modeling of Argumentation in STEM Settings

- **Observation**: Much of the wealth and prospects of central European Countries are based on STEM knowledge. (laid down in technical documents)
- STEM documents often have a non-trivial argumentation structure
- **Example 6.7.** Short excerpt of Coffey's and Sondow's rebuttal [CS12] of Kowalenko's paper [Kow10].

> The irrationality of Euler's constant γ *[...]* has long been conjectured. *[...]* In 2010 Kowalenko claimed that simple arguments suffice to settle this matter [4]. *[...]* we *[...]* describe the flaws in his very limited approach.
> *[...]*
> Kowalenko derives the following formula for Euler's constant in equation (65) of [4, p. 428]: *[...]*
> *[...]*
> Here he claims that the sum of a series of positive rational numbers cannot be equal to $C - \pi^2/6$. But, for example, decimal expansion does give such a series: *[...]*

- **Observation**: Often the aim of STEM argumentation is *uncovering the truth* (and reputation/grant money gain)

Deep Modeling of Argumentation in STEM Settings

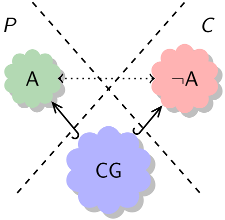
- Observation: Much of the wealth and prospects of central European Countries are based on STEM knowledge. (laid down in technical documents)
- STEM documents often have a non-trivial argumentation structure
- **Example 6.7.** Short excerpt of Coffey's and Sondow's rebuttal [CS12] of Kowalenko's paper [Kow10].

> The irrationality of Euler's constant γ *[...]* has long been conjectured. *[...]* In 2010 Kowalenko claimed that simple arguments suffice to settle this matter [4]. *[...]* we *[...]* describe the flaws in his very limited approach.
> *[...]*
> Kowalenko derives the following formula for Euler's constant in equation (65) of [4, p. 428]: *[...]*
> *[...]*
> Here he claims that the sum of a series of positive rational numbers cannot be equal to $C - \pi^2/6$. But, for example, decimal expansion does give such a series: *[...]*

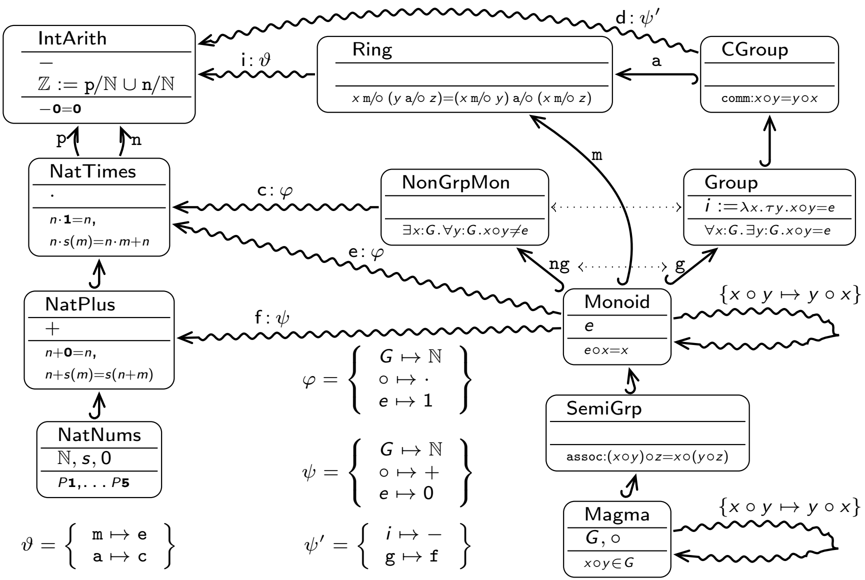
- Observation: Often the aim of STEM argumentation is *uncovering the truth* (and reputation/grant money gain)
- Idea: RATIO on technical/scientific documents (needs deep modeling)

Deep Modeling of Argumentation in STEM Settings

- Observation: often the ultimate source of differing opinions in STEM lies in differing assumptions.
- **Example 6.8 (Example).** various models in physics that make differing predictions, e.g. heliocentric vs. geocentric universe.
- Scientific Method: Explore the inferential closure of the model assumptions, contrast to others/experiments, argue for your model.
- Idea: Meta-model differing model assumptions as OMDoc/MMT theory graph
  - recast the *support*, *refutation* or *undercut* relations via theory morphisms + ϵ.
  - theory morphisms incorporate inferential closure and renaming/framing.
  - concept-minimal graphs explicitly manage common ground.
  - Extend theory graph algorithms for that.



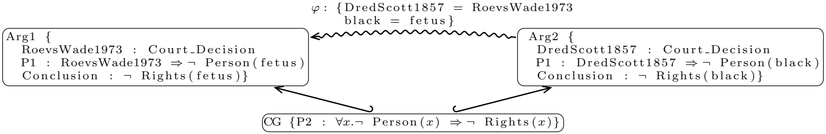
Modular Representation of Math (MMT Example)



Framing in Arguments

- Definition: In a nutshell, framing means that a *concept mapping* between *argumentation/knowledge contexts* (a **frame**) is established and the facts and assumptions underlying the argument are mapped along the frame.
- Observation: This happens often in counter-arguments by framing the original argument in terms of an obviously wrong argument.
- **Example 6.9 (Roe vs. Wade).** from `www.truthmapping.com/map/647/`
  - The 1973 Roe vs. Wade decision denied fetus' rights on the basis of personhood.
  - The 1857 Dred Scott decision denied Black Americans rights on the basis of personhood.
  - Personhood for Black Americans has been denied purely on the basis of cultural consensus.
  - Therefore the denial of personhood for fetuses could also be purely on the basis of cultural consensus.

  Model in a theory graph using a frames as morphisms approach

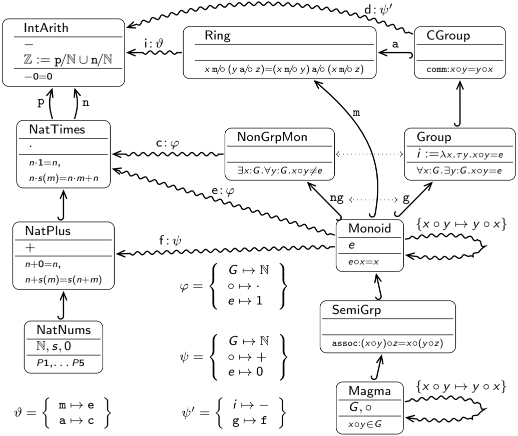


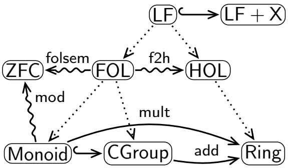
Work Area 2: Work Plan

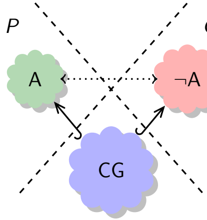
- WP2.1: Annotated Corpus of Technical Documents
  1. Subcorpus Identification
  2. Argumentation/Context Annotation
  3. Distribution
- WP2.2: Context Graph via Argumentation Relations
- WP2.3: Extending the MMT system with Context Graph Relations
- WP2.4: Framing in Arguments
  1. Modelling (work through lots of examples)
  2. Automation (use the OMDoc/MMT view finder to discover possible frames)

Visual Conclusion (please ask questions)

- Summary: Understanding/Supporting Logic-Based Deep Modeling of Arg.
- Contribution: develop and manage the targets of semantics extraction!







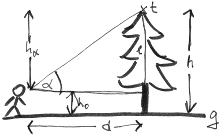
7 Application: Serious Games

Framing for Problem Solving (The FrameIT Method)

- **Example 7.1 (Problem 0.8.15).**
  How can you measure the height of a tree you cannot climb, when you only have a protactor and a tape measure at hand.

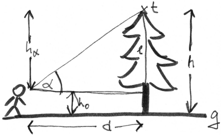
Framing for Problem Solving (The FrameIT Method)

- **Example 7.1 (Problem 0.8.15).**
  How can you measure the height of a tree you cannot climb, when you only have a protactor and a tape measure at hand.

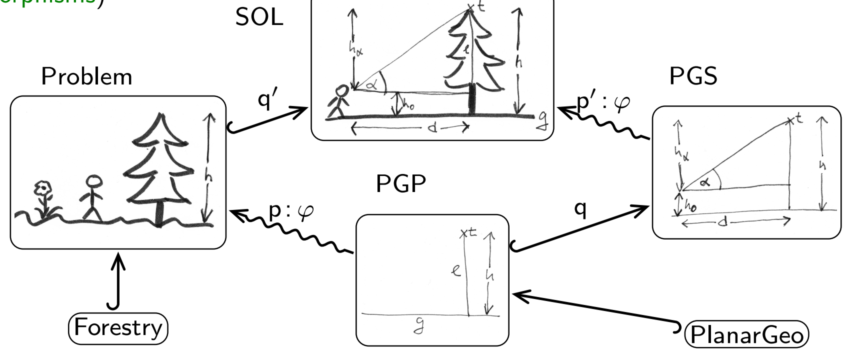


Framing for Problem Solving (The FrameIT Method)

- **Example 7.1 (Problem 0.8.15).**
  How can you measure the height of a tree you cannot climb, when you only have a protactor and a tape measure at hand.

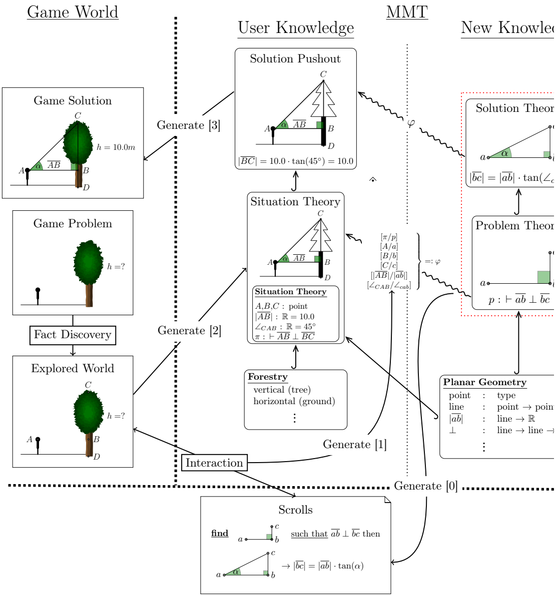


- Framing: view the problem as one that is already understood (using theory morphisms)



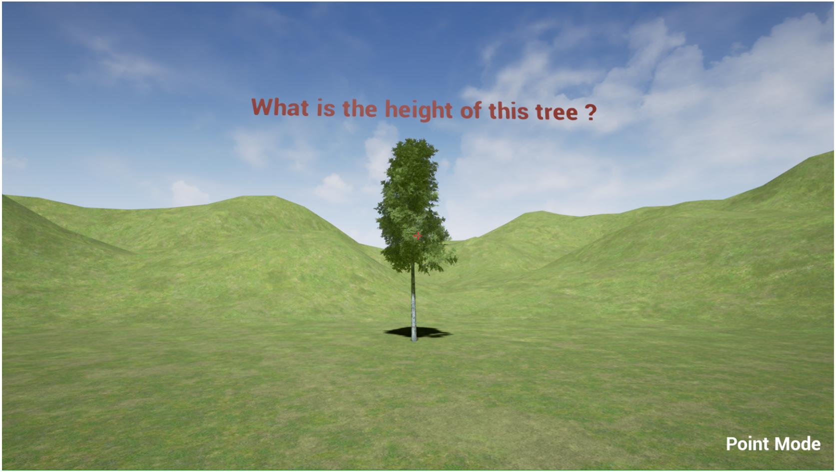
- squiggly (framing) morphisms guaranteed by metatheory of theories!

Example Learning Object Graph



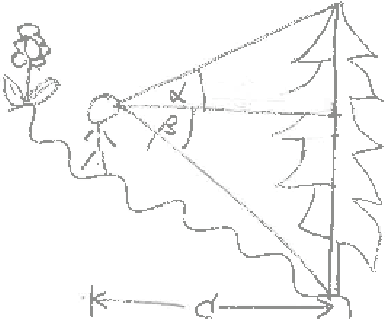
FrameIT Method: Problem

- Problem Representation in the game world (what the student should see)



- Student can interact with the environment via gadgets so solve problems
- "Scrolls" of mathematical knowledge give hints.

Combining Problem/Solution Pairs



- We can use the same mechanism for combining P/S pairs
- create more complex P/S pairs (e.g. for trees on slopes)

Overview: KWARC Research and Projects

Applications: eMath 3.0, Active Documents, Semantic Spreadsheets, Semantic CAD/CAM, Change Mangagement, Global Digital Math Library, Math Search Systems, SMGloM: Semantic Multilingual Math Glossary, Serious Games, ...		
Foundations of Math: ▶ MathML, *OpenMath* ▶ advanced Type Theories ▶ MMT: Meta Meta Theory ▶ Logic Morphisms/Atlas ▶ Theorem Prover/CAS Interoperability ▶ Mathematical Models/Simulation	**KM & Interaction**: ▶ Semantic Interpretation (aka. Framing) ▶ math-literate interaction ▶ MathHub: math archives & active docs ▶ Semantic Alliance: embedded semantic services	**Semantization**: ▶ LaTeXML: LaTeX → XML ▶ sTeX: Semantic LaTeX ▶ invasive editors ▶ Context-Aware IDEs ▶ Mathematical Corpora ▶ Linguistics of Math ▶ ML for Math Semantics Extraction
Foundations: Computational Logic, Web Technologies, *OMDoc*/MMT		

References I

John P. Burgess. "Basic Tense Logic". In: *Handbook of Philosophical Logic: Volume II: Extensions of Classical Logic*. Ed. by D. Gabbay and F. Guenthner. Dordrecht: Springer Netherlands, 1984, pp. 89–133. ISBN: 978-94-009-6259-0. DOI: 10.1007/978-94-009-6259-0_2.

Mark W. Coffey and Jonathan Sondow. "Rebuttal of Kowalenko's paper as concerns the irrationality of Euler's constant". In: (2012). eprint: arXiv:1202.3093.

J. Michael Dunn and Greg Restall. "Relevance Logic". In: *Handbook of Philosophical Logic*. Ed. by Dov M. Gabbay and F. Guenthner. Dordrecht: Springer Netherlands, 2002, pp. 1–128. ISBN: 978-94-017-0460-1. DOI: 10.1007/978-94-017-0460-1_1.

Sven Ove Hansson. "Preference Logic". In: *Handbook of Philosophical Logic*. Ed. by D. M. Gabbay and F. Guenthner. Dordrecht: Springer Netherlands, 2002, pp. 319–393. ISBN: 978-94-017-0456-4. DOI: 10.1007/978-94-017-0456-4_4.

References II

- Arif Jinha. “Article 50 million: an estimate of the number of scholarly articles in existence”. In: *Learned Publishing* 23.3 (2010), pp. 258–263. DOI: 10.1087/20100308.
- Andrea Kohlhase and Michael Kohlhase. “Communities of Practice in MKM: An Extensional Model”. In: *Mathematical Knowledge Management (MKM)*. Ed. by Jon Borwein and William M. Farmer. LNAI 4108. Springer Verlag, 2006, pp. 179–193. URL: http://kwarc.info/kohlhase/papers/mkm06cp.pdf.
- Victor Kowalenko. “Properties and applications of the reciprocal logarithm numbers”. In: *Acta applicandae mathematicae* 109.2 (2010), pp. 413–437.
- Peder Olesen Larsen and Markus von Ins. “The rate of growth in scientific publication and the decline in coverage provided by Science Citation Index”. In: *Scientometrics* 84.3 (2010), pp. 575–603. DOI: 10.1007/s11192-010-0202-z.

References III

Nils J. Nilsson. “Probabilistic Logic”. In: *Artif. Intell.* 28.1 (Feb. 1986), pp. 71–88. ISSN: 0004-3702. DOI: 10.1016/0004-3702(86)90031-7.

Florian Rabe and C. Schürmann. “A Practical Module System for LF”. In: *Proceedings of the Workshop on Logical Frameworks: Meta-Theory and Practice (LFMTP)*. Ed. by J. Cheney and A. Felty. Vol. LFMTP’09. ACM International Conference Proceeding Series. ACM Press, 2009, pp. 40–48.