A Structured Set of Higher-Order Problems

Christoph E. Benzmüller and Chad E. Brown

Computer Science

Saarland University

Saarbrücken, Germany

TPHOLS'05, 22 August 2005, Oxford, UK

UNIVERSITÄT DES SAARLANDES

Test Problems for Theorem Provers

- Test problems for FOL theorem provers

Test Problems for Theorem Provers

- Test problems for FOL theorem provers
  - [McCharenOverbeekWos76], [WilsonMinker79], [Pelletier86], etc.

Test Problems for Theorem Provers

- Test problems for FOL theorem provers
  - [McCharenOverbeekWos76], [WilsonMinker79], [Pelletier86], etc.
  - TPTP [PelletierSutcliffeSuttner02]

Test Problems for Theorem Provers

- Test problems for FOL theorem provers
  - [McCharenOverbeekWos76], [WilsonMinker79], [Pelletier86], etc.
  - TPTP [PelletierSutcliffeSuttner02]
  - significantly fostered the development of FOL ATPs

Test Problems for Theorem Provers

- Test problems for FOL theorem provers
  - [McCharenOverbeekWos76], [WilsonMinker79], [Pelletier86], etc.
  - TPTP [PelletierSutcliffeSuttner02]
  - significantly fostered the development of FOL ATPs
- Test problems for HOL theorem provers

Test Problems for Theorem Provers

- Test problems for FOL theorem provers
  - [McCharenOverbeekWos76], [WilsonMinker79], [Pelletier86], etc.
  - TPTP [PelletierSutcliffeSuttner02]
  - significantly fostered the development of FOL ATPs
- Test problems for HOL theorem provers
  - common library missing

Test Problems for Theorem Provers

- Test problems for FOL theorem provers
  - [McCharenOverbeekWos76], [WilsonMinker79], [Pelletier86], etc.
  - TPTP [PelletierSutcliffeSuttner02]
  - significantly fostered the development of FOL ATPs
- Test problems for HOL theorem provers
  - common library missing
- Is our paper proposing challenging HOL benchmark problems?

Test Problems for Theorem Provers

- Test problems for FOL theorem provers
  - [McCharenOverbeekWos76], [WilsonMinker79], [Pelletier86], etc.
  - TPTP [PelletierSutcliffeSuttner02]
  - significantly fostered the development of FOL ATPs
- Test problems for HOL theorem provers
  - common library missing
- Is our paper proposing challenging HOL benchmark problems?
  - No!!!

Test Problems for Theorem Provers

- Test problems for FOL theorem provers
  - [McCharenOverbeekWos76], [WilsonMinker79], [Pelletier86], etc.
  - TPTP [PelletierSutcliffeSuttner02]
  - significantly fostered the development of FOL ATPs
- Test problems for HOL theorem provers
  - common library missing
- Is our paper proposing challenging HOL benchmark problems?
  - No!!!
  - But see the ESHOL'05-WS at LPAR'05 in Jamaica `http://www.ags.uni-sb.de/~chris/ESHOL-05`

UNIVERSITÄT DES SAARLANDES

The Examples in this Paper

- Examples are simple

The Examples in this Paper

- Examples are simple
  - highlight the essence of some semantical or technical point

The Examples in this Paper

- Examples are simple
  - highlight the essence of some semantical or technical point
  - easy to understand and easy to encode

The Examples in this Paper

- Examples are simple
  - highlight the essence of some semantical or technical point
  - easy to understand and easy to encode
  - relevant for both: automated and interactive TP

The Examples in this Paper

- Examples are simple
  - highlight the essence of some semantical or technical point
  - easy to understand and easy to encode
  - relevant for both: automated and interactive TP
- Examples are structured

The Examples in this Paper

- Examples are simple
  - highlight the essence of some semantical or technical point
  - easy to understand and easy to encode
  - relevant for both: automated and interactive TP
- Examples are structured
  - quick indicators for completeness and soundness wrt to HOL model classes from [Benzm.BrownKohlhase-JSL-04]

The Examples in this Paper

- Examples are simple
  - highlight the essence of some semantical or technical point
  - easy to understand and easy to encode
  - relevant for both: automated and interactive TP
- Examples are structured
  - quick indicators for completeness and soundness wrt to HOL model classes from [Benzm.BrownKohlhase-JSL-04]
  - shall precede formal soundness / completeness analysis

The Examples in this Paper

- Examples are simple
  - highlight the essence of some semantical or technical point
  - easy to understand and easy to encode
  - relevant for both: automated and interactive TP
- Examples are structured
  - quick indicators for completeness and soundness wrt to HOL model classes from [Benzm.BrownKohlhase-JSL-04]
  - shall precede formal soundness / completeness analysis
  - many are collected from experience with LEO and TPS

UNIVERSITÄT DES SAARLANDES

The Examples in this Paper

- Examples are simple
  - highlight the essence of some semantical or technical point
  - easy to understand and easy to encode
  - relevant for both: automated and interactive TP
- Examples are structured
  - quick indicators for completeness and soundness wrt to HOL model classes from [Benzm.BrownKohlhase-JSL-04]
  - shall precede formal soundness / completeness analysis
  - many are collected from experience with LEO and TPS
- (Some more challenging examples are also added)

Outline of Talk

- HOL (notion and syntax)

Outline of Talk

- HOL (notion and syntax)
- HOL-CUBE: different model classes for HOL (semantics)

Outline of Talk

- HOL (notion and syntax)
- HOL-CUBE: different model classes for HOL (semantics)
  - Boolean extensionality
  - functional extensionality
    - η-equality
    - ξ-equality

Outline of Talk

- HOL (notion and syntax)
- HOL-CUBE: different model classes for HOL (semantics)
  - Boolean extensionality
  - functional extensionality
    - η-equality
    - ξ-equality
- HOL-Problems: simple problems structured along HOL-CUBE

Outline of Talk

- HOL (notion and syntax)
- HOL-CUBE: different model classes for HOL (semantics)
  - Boolean extensionality
  - functional extensionality
    - η-equality
    - ξ-equality
- HOL-Problems: simple problems structured along HOL-CUBE
- (Some more challenging problems)

Outline of Talk

- HOL (notion and syntax)
- HOL-CUBE: different model classes for HOL (semantics)
  - Boolean extensionality
  - functional extensionality
    - η-equality
    - ξ-equality
- HOL-Problems: simple problems structured along HOL-CUBE
- (Some more challenging problems)
- Conclusion

HOL: Simple Types

Simple Types $\mathcal{T}$:	o	(truth values)
	ι	(individuals)
	$(\alpha\beta)$	(functions from β to α)

HOL: Simple Types

Simple Types $\mathcal{T}$:	o	(truth values)
	ι	(individuals)
	$(\alpha\beta)$	(functions from β to α)

$(\alpha\beta\gamma)$ abbreviates $((\alpha\beta)\gamma)$

HOL: Simply Typed λ-Terms

Terms:	X_α	Variables ($\mathcal{V}$)
	a_α	Parameters ($\mathcal{P}$)
	c_α	Logical constants ($\mathcal{S}$)
	$[\mathbf{F}_{\alpha\beta}\,\mathbf{B}_\beta]_\alpha$	Application
	$[\lambda \mathsf{Y}_\beta\,\mathbf{A}_\alpha]_{\alpha\beta}$	λ-abstraction

HOL: Simply Typed λ-Terms

Terms:

X_α	Variables ($\mathcal{V}$)
a_α	Parameters ($\mathcal{P}$)
c_α	Logical constants ($\mathcal{S}$)
$[\mathbf{F}_{\alpha\beta}\,\mathbf{B}_\beta]_\alpha$	Application
$[\lambda \mathsf{Y}_\beta\,\mathbf{A}_\alpha]_{\alpha\beta}$	λ-abstraction

Equality of terms: $\alpha\beta\eta$

HOL: Simply Typed λ-Terms

Terms:	X_α	Variables ($\mathcal{V}$)
	a_α	Parameters ($\mathcal{P}$)
	c_α	Logical constants ($\mathcal{S}$)
	$[\mathbf{F}_{\alpha\beta}\,\mathbf{B}_\beta]_\alpha$	Application
	$[\lambda \mathsf{Y}_\beta\,\mathbf{A}_\alpha]_{\alpha\beta}$	λ-abstraction

Equality of terms: $\alpha\beta\eta$

α-conversion	Changing Bound Variables	
β-reduction	$[[\lambda \mathsf{Y}_\beta\,\mathbf{A}_\alpha]\,\mathbf{B}] \xrightarrow{\beta} [\mathbf{B}/\mathsf{Y}]\mathbf{A}$	
η-reduction	$[\lambda \mathsf{Y}_\beta\,[\mathbf{F}_{\alpha\beta}\,\mathsf{Y}]] \xrightarrow{\eta} \mathbf{F}$	$(\mathsf{Y}_\beta \notin \mathsf{Free}(\mathbf{F}))$

HOL: Simply Typed λ-Terms

Terms:	X_α	Variables ($\mathcal{V}$)
	a_α	Parameters ($\mathcal{P}$)
	c_α	Logical constants ($\mathcal{S}$)
	$[\mathbf{F}_{\alpha\beta}\,\mathbf{B}_\beta]_\alpha$	Application
	$[\lambda \mathsf{Y}_\beta\,\mathbf{A}_\alpha]_{\alpha\beta}$	λ-abstraction

Equality of terms: $\alpha\beta\eta$

Every term has a unique $\beta\eta$-normal form (up to α-conversion).

HOL: Logical Constants

Some logical constants which may be in $\mathcal{S}$:

- $\top_o$ – true
- $\bot_o$ – false
- $\neg_{oo}$ – negation
- $\vee_{ooo}$ – disjunction
- $\wedge_{ooo}$ – conjunction
- $\Rightarrow_{ooo}$ – implication
- $\Leftrightarrow_{ooo}$ – equivalence

HOL: Logical Constants

Some logical constants which may be in $\mathcal{S}$:

- $=^{\alpha}_{o\alpha\alpha}$ – equality at type α
- $\Pi^{\alpha}_{o(o\alpha)}$ – universal quantification over type α
- $\Sigma^{\alpha}_{o(o\alpha)}$ – existential quantification over type α

Intuition:

- $[\Sigma^{\alpha} \mathbin{\blacksquare} \lambda X_{\alpha}\, \mathbf{C}_{o}]$ is true iff $\{X_{\alpha} \mid \mathbf{C}\}$ is nonempty.

HOL: Abbreviations for Logical Operators

- $[\mathbf{A}_o \lor \mathbf{B}_o]$ means $[\lor_{ooo}\, \mathbf{A}_o\, \mathbf{B}_o]$
- $[\mathbf{A}_o \land \mathbf{B}_o]$ means $[\land_{ooo}\, \mathbf{A}_o\, \mathbf{B}_o]$
- $[\mathbf{A}_o \Rightarrow \mathbf{B}_o]$ means $[\Rightarrow_{ooo}\, \mathbf{A}_o\, \mathbf{B}_o]$
- $[\mathbf{A}_o \Leftrightarrow \mathbf{B}_o]$ means $[\Leftrightarrow_{ooo}\, \mathbf{A}_o\, \mathbf{B}_o]$
- $[\mathbf{A}_\alpha =^\alpha \mathbf{B}_\alpha]$ means $[=^\alpha_{o\alpha\alpha}\, \mathbf{A}_\alpha\, \mathbf{B}_\alpha]$
- $[\forall X_\alpha\, \mathbf{A}_o]$ means $[\Pi^\alpha_{o(o\alpha)} \blacksquare \lambda X_\alpha\, \mathbf{A}_o]$
- $[\exists X_\alpha\, \mathbf{A}_o]$ means $[\Sigma^\alpha_{o(o\alpha)} \blacksquare \lambda X_\alpha\, \mathbf{A}_o]$

HOL: Equality and Extensionality

Some important conventions and reminders for the talk:

- $=$ denotes primitive equality

HOL: Equality and Extensionality

Some important conventions and reminders for the talk:

- $=$ denotes primitive equality
- $\doteq$ denotes Leibniz equality: $\mathbf{A}_\alpha \doteq^\alpha \mathbf{B}_\alpha := \forall \mathsf{P}_{o\alpha}\text{.}\ (\mathsf{P}\mathbf{A}) \Rightarrow (\mathsf{P}\mathbf{B})$

HOL: Equality and Extensionality

Some important conventions and reminders for the talk:

- $=$ denotes primitive equality
- $\doteq$ denotes Leibniz equality: $\mathbf{A}_\alpha \doteq^\alpha \mathbf{B}_\alpha := \forall \mathsf{P}_{o\alpha}\textbf{.}\, (\mathsf{P}\mathbf{A}) \Rightarrow (\mathsf{P}\mathbf{B})$
- $\ddot{=}$ …other definition of equality (e.g., see [Andrews02])

HOL: Equality and Extensionality

Some important conventions and reminders for the talk:

- $=$ denotes primitive equality
- $\doteq$ denotes Leibniz equality: $\mathbf{A}_\alpha \doteq^\alpha \mathbf{B}_\alpha := \forall \mathsf{P}_{o\alpha}\textbf{.}\ (\mathsf{P}\mathbf{A}) \Rightarrow (\mathsf{P}\mathbf{B})$
- $\ddot{=}$ ...other definition of equality (e.g., see [Andrews02])

We use $\overset{*}{=}$ in the following to refer to any of the above

HOL: Church's Type Theory

Church's Type Theory:

- Simply typed λ-calculus with the signature

  $$\mathcal{S} := \{\neg, \vee\} \cup \{\Pi^\alpha \mid \alpha \in \mathcal{T}\}$$

  (and perhaps a description or choice operator).

HOL: Church's Type Theory

Church's Type Theory:

- Simply typed λ-calculus with the signature

  $\mathcal{S} := \{\neg, \vee\} \cup \{\Pi^{\alpha} \mid \alpha \in \mathcal{T}\}$

  (and perhaps a description or choice operator).
- Axioms of extensionality

HOL: Church's Type Theory

Church's Type Theory:

- Simply typed λ-calculus with the signature

  $\mathcal{S} := \{\neg, \vee\} \cup \{\Pi^\alpha \mid \alpha \in \mathcal{T}\}$

  (and perhaps a description or choice operator).
- Axioms of extensionality
- Axiom of description or choice

Church’s Type Theory:

- Simply typed λ-calculus with the signature

  $$\mathcal{S} := \{\neg, \vee\} \cup \{\Pi^{\alpha} \mid \alpha \in \mathcal{T}\}$$

  (and perhaps a description or choice operator).
- Axioms of extensionality
- Axiom of description or choice
- Axiom of infinity

HOL: Elementary and Ext. Type Theory

$\mathcal{S}$ Fragment of Elementary Type Theory:

$\mathcal{S}$ Fragment of Extensional Type Theory:

HOL: Elementary and Ext. Type Theory

$\mathcal{S}$ Fragment of Elementary Type Theory:

- Simply typed λ-calculus with the signature $\mathcal{S}$

$\mathcal{S}$ Fragment of Extensional Type Theory:

- Simply typed λ-calculus with the signature $\mathcal{S}$

HOL: Elementary and Ext. Type Theory

$\mathcal{S}$ Fragment of Elementary Type Theory:

- Simply typed λ-calculus with the signature $\mathcal{S}$
- No extensionality

$\mathcal{S}$ Fragment of Extensional Type Theory:

- Simply typed λ-calculus with the signature $\mathcal{S}$
- Extensionality

HOL: Elementary and Ext. Type Theory

$\mathcal{S}$ Fragment of Elementary Type Theory:

- Simply typed λ-calculus with the signature $\mathcal{S}$
- No extensionality
- No axiom of description or choice

$\mathcal{S}$ Fragment of Extensional Type Theory:

- Simply typed λ-calculus with the signature $\mathcal{S}$
- Extensionality
- No axiom of description or choice

HOL: Elementary and Ext. Type Theory

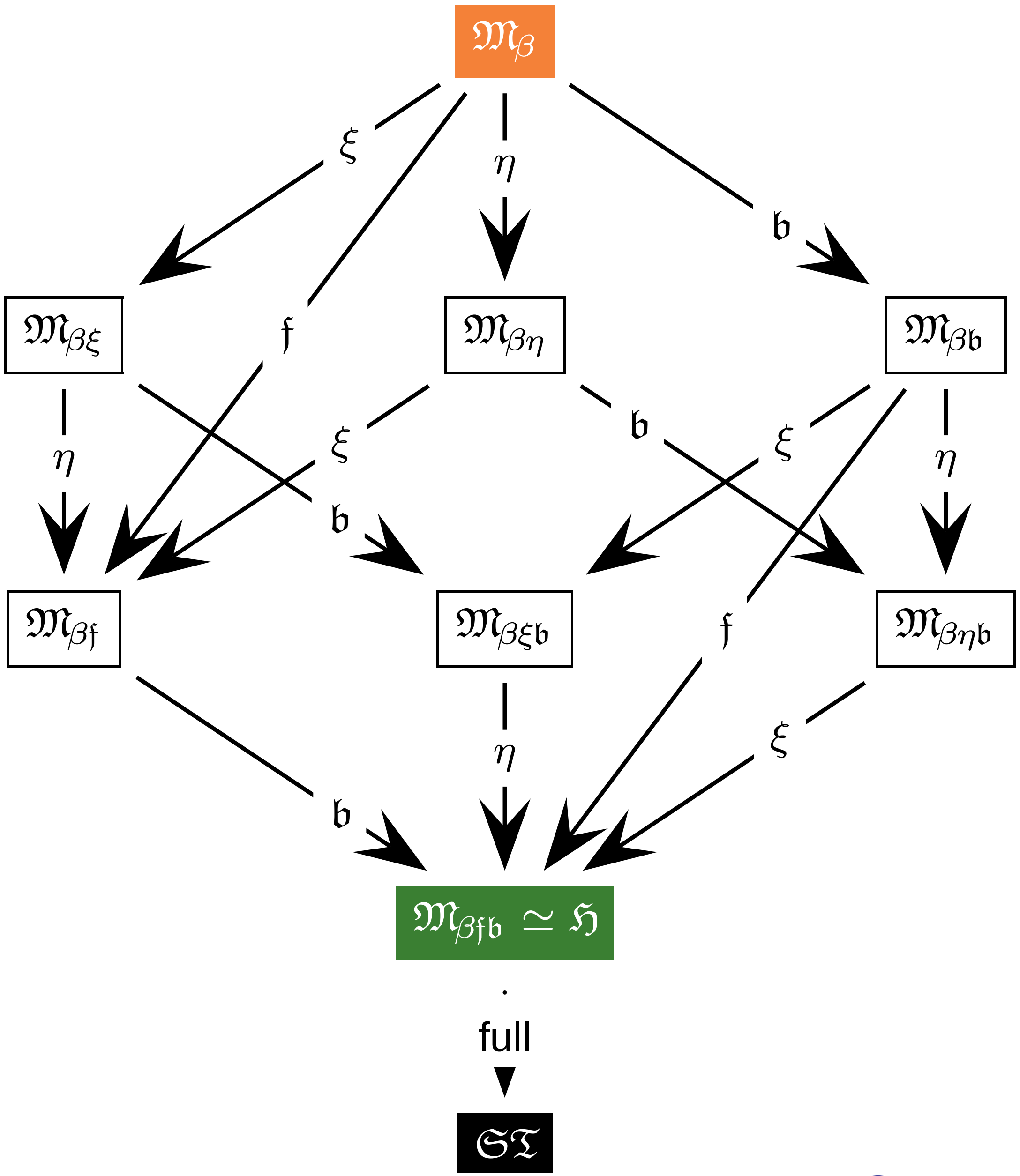
$\mathcal{S}$ Fragment of Elementary Type Theory:

- Simply typed λ-calculus with the signature $\mathcal{S}$
- No extensionality
- No axiom of description or choice
- No axiom of infinity

$\mathcal{S}$ Fragment of Extensional Type Theory:

- Simply typed λ-calculus with the signature $\mathcal{S}$
- Extensionality
- No axiom of description or choice
- No axiom of infinity

Semantics: HOL-CUBE



Landscape of HOL model classes

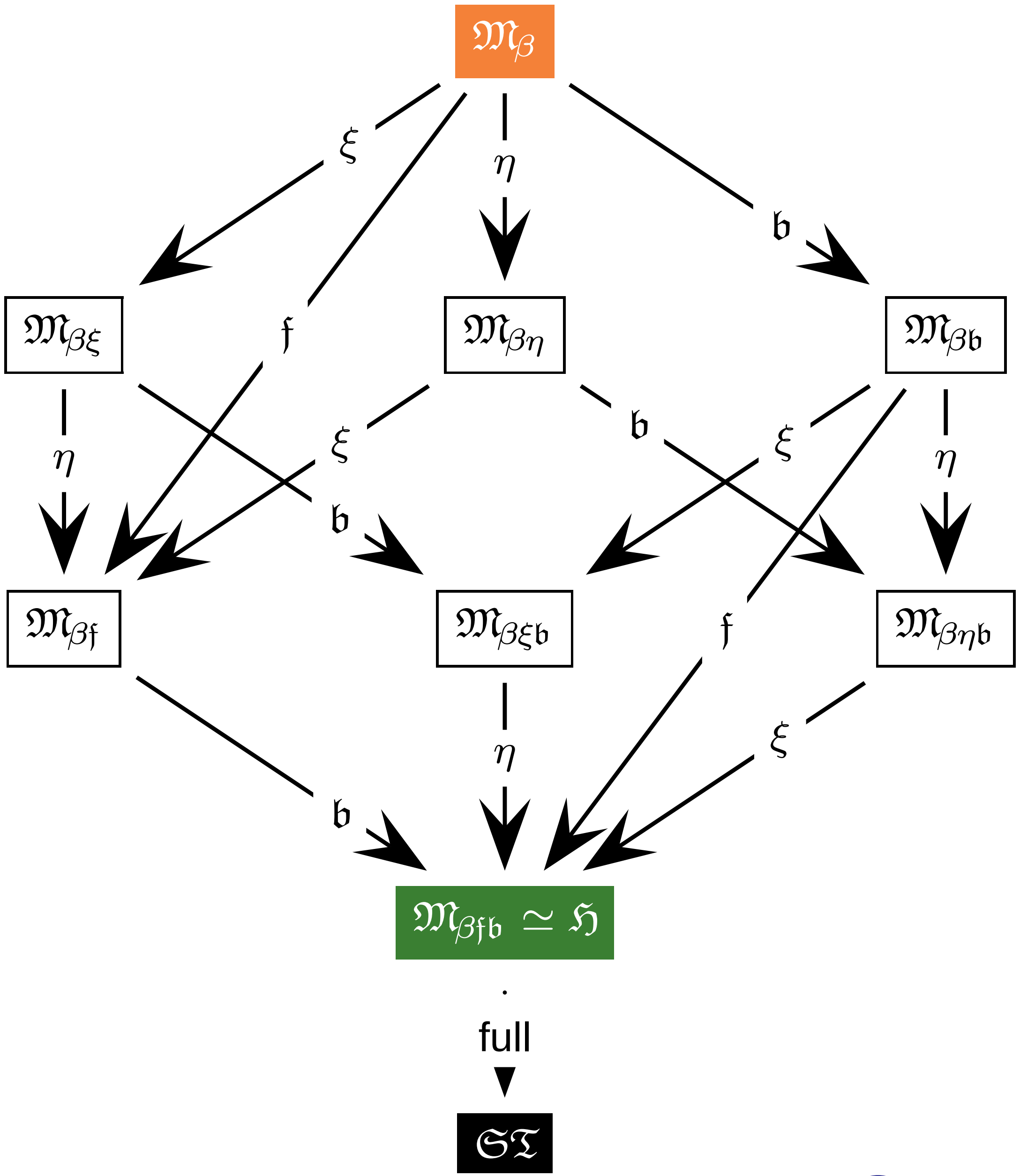
[Kohlhase-PhD-94]

[Benzmüller-PhD-99]

[Brown-PhD-04]

[Benzm.BrownKohlhase-JSL-04]

Semantics: HOL-CUBE



Landscape of HOL model classes

[Kohlhase-PhD-94]

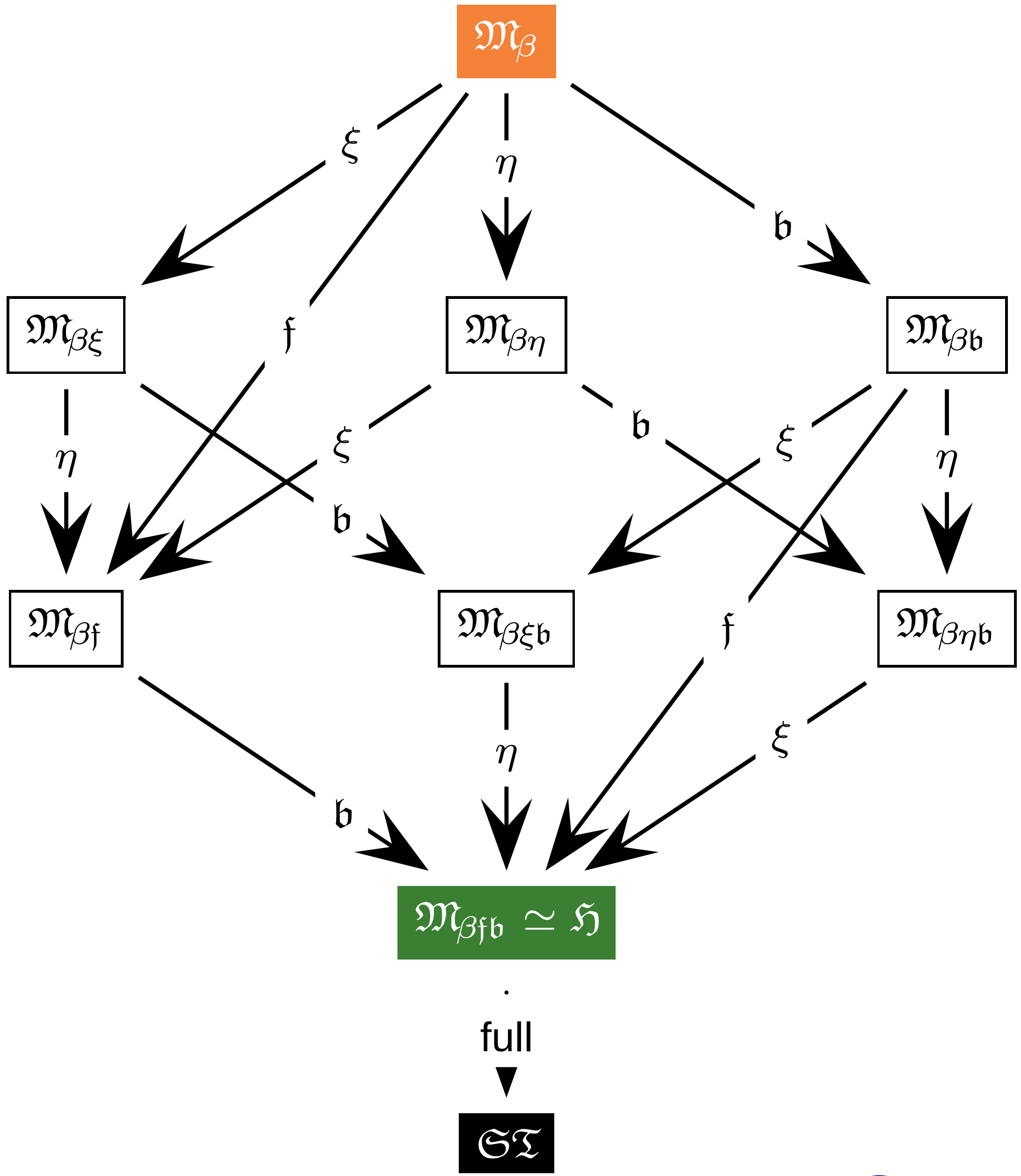
[Benzmüller-PhD-99]

[Brown-PhD-04]

[Benzm.BrownKohlhase-JSL-04]

$\mathfrak{M}_{\beta}$ model class for $\mathcal{S}$ fragment of elementary type theory

Semantics: HOL-CUBE



Landscape of HOL model classes

[Kohlhase-PhD-94]

[Benzmüller-PhD-99]

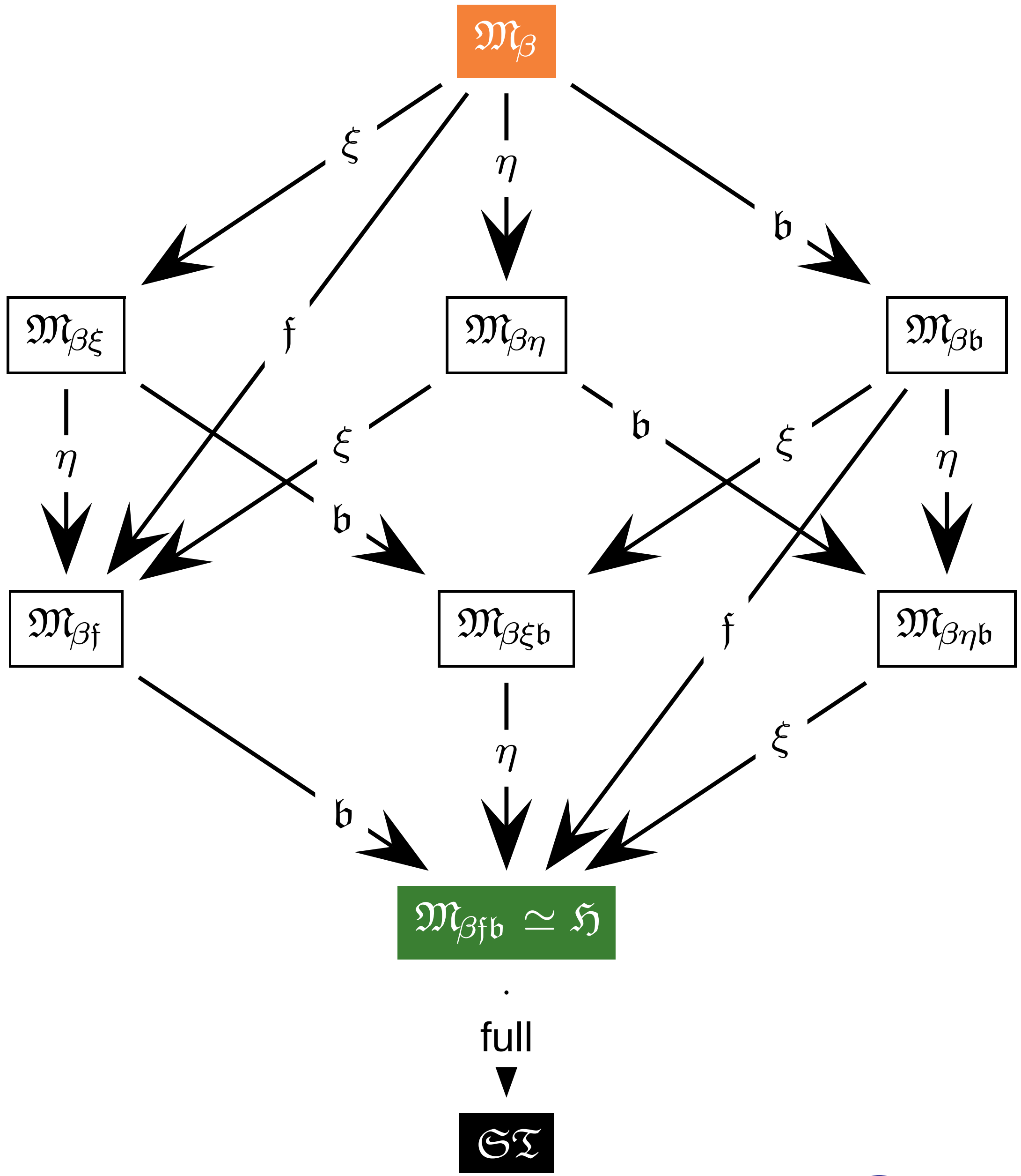
[Brown-PhD-04]

[Benzm.BrownKohlhase-JSL-04]

$\mathfrak{M}_\beta$ model class for $\mathcal{S}$ fragment of elementary type theory

$\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$ model class for $\mathcal{S}$ fragment of extensional type theory (Henkin models)

Semantics: HOL-CUBE



Landscape of HOL model classes

[Kohlhase-PhD-94]

[Benzmüller-PhD-99]

[Brown-PhD-04]

[Benzm.BrownKohlhase-JSL-04]

$\mathfrak{M}_\beta$ model class for $\mathcal{S}$ fragment of elementary type theory

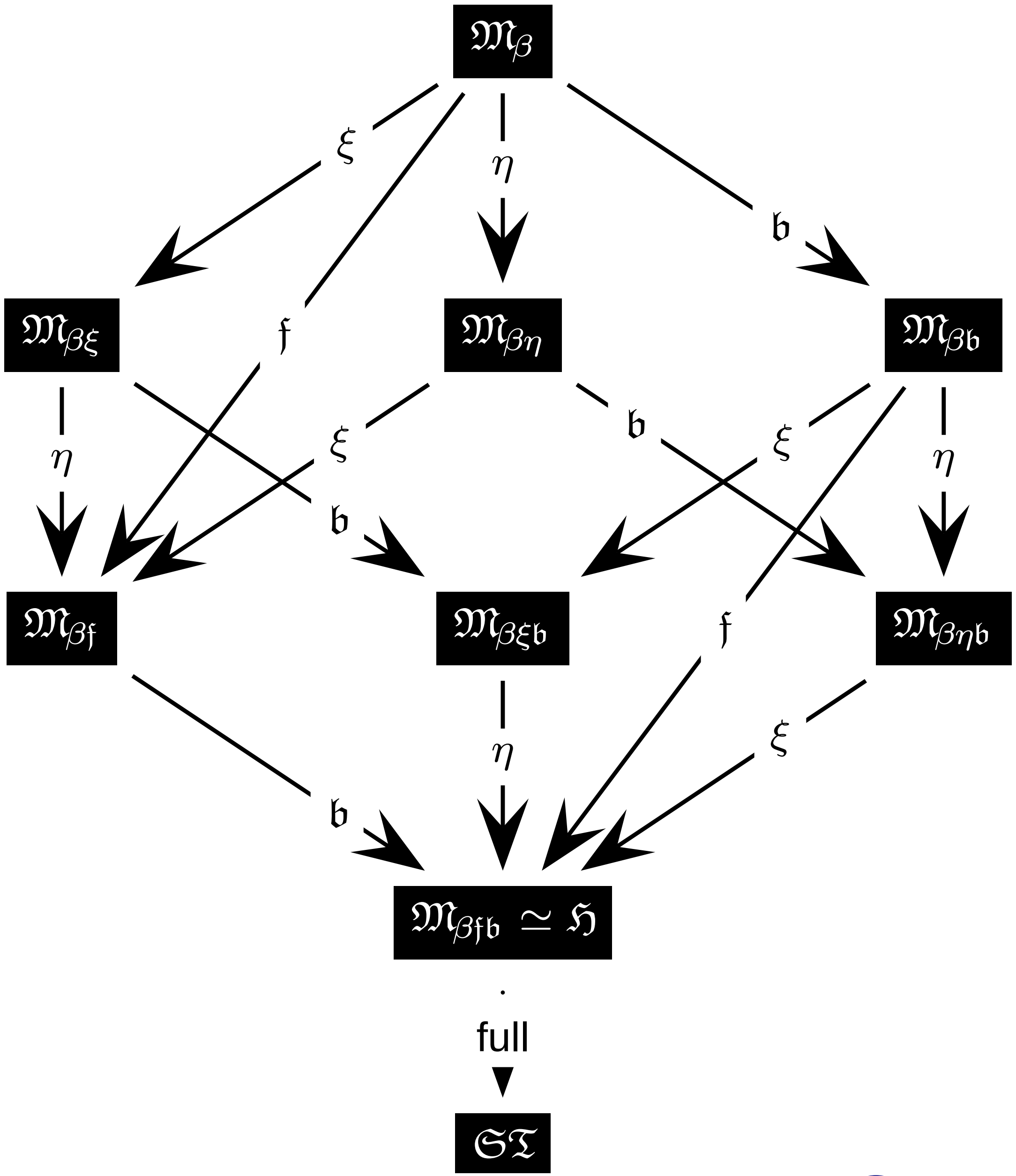
$\mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}}$ model class for $\mathcal{S}$ fragment of extensional type theory (Henkin models)

Signature $\mathcal{S}$ defined as

$$\{\top, \bot, \neg, \wedge, \vee, \Rightarrow, \Leftrightarrow\} \cup \{\Pi^\alpha, \Sigma^\alpha, =^\alpha\}$$

(less logical connectives are possible)

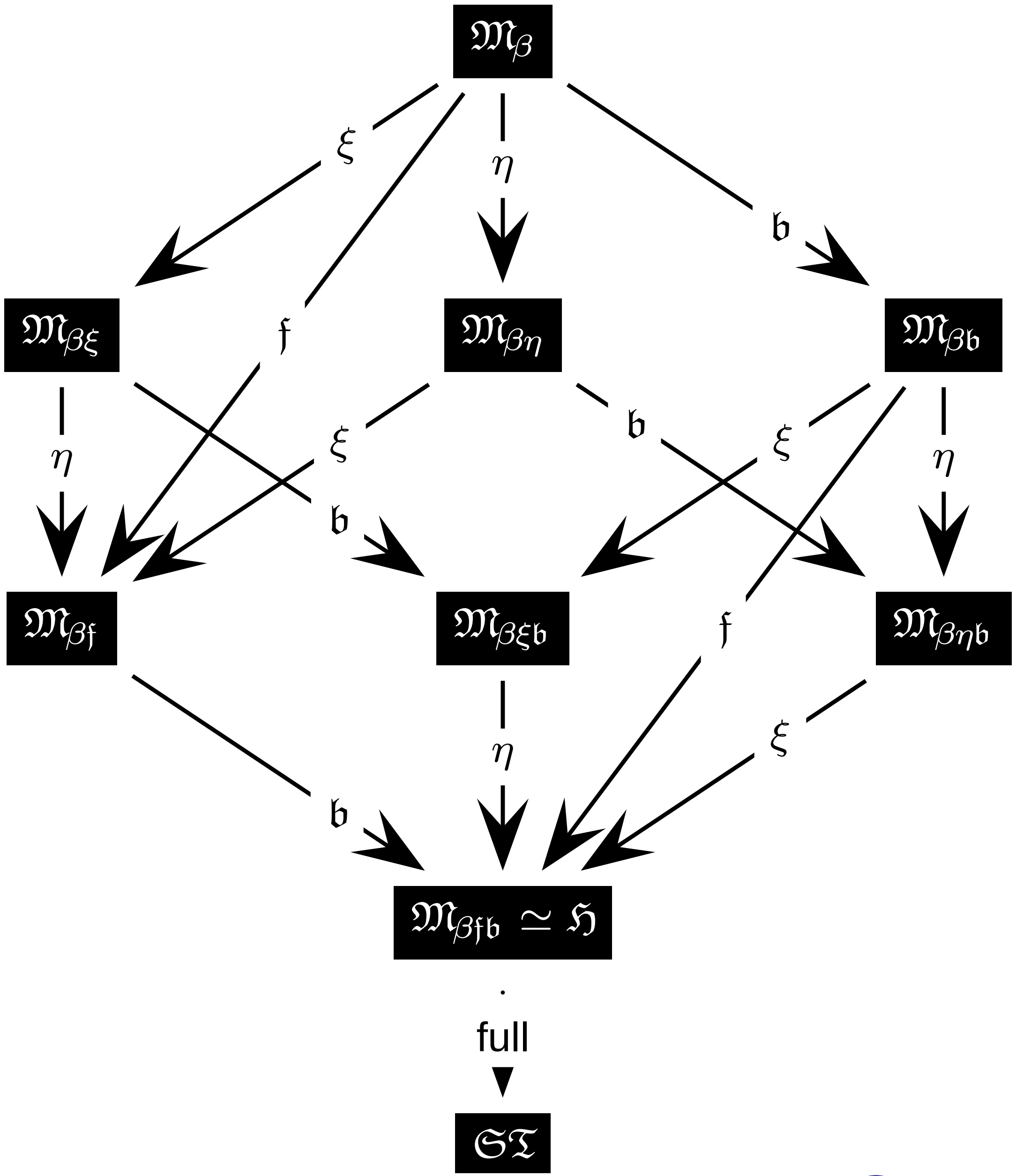
Semantics: HOL-CUBE



$\mathfrak{q}$: models provide identity relations

$$\forall\alpha : \mathrm{id} \in \mathcal{D}_{\alpha\to\alpha\to o}$$

Semantics: HOL-CUBE



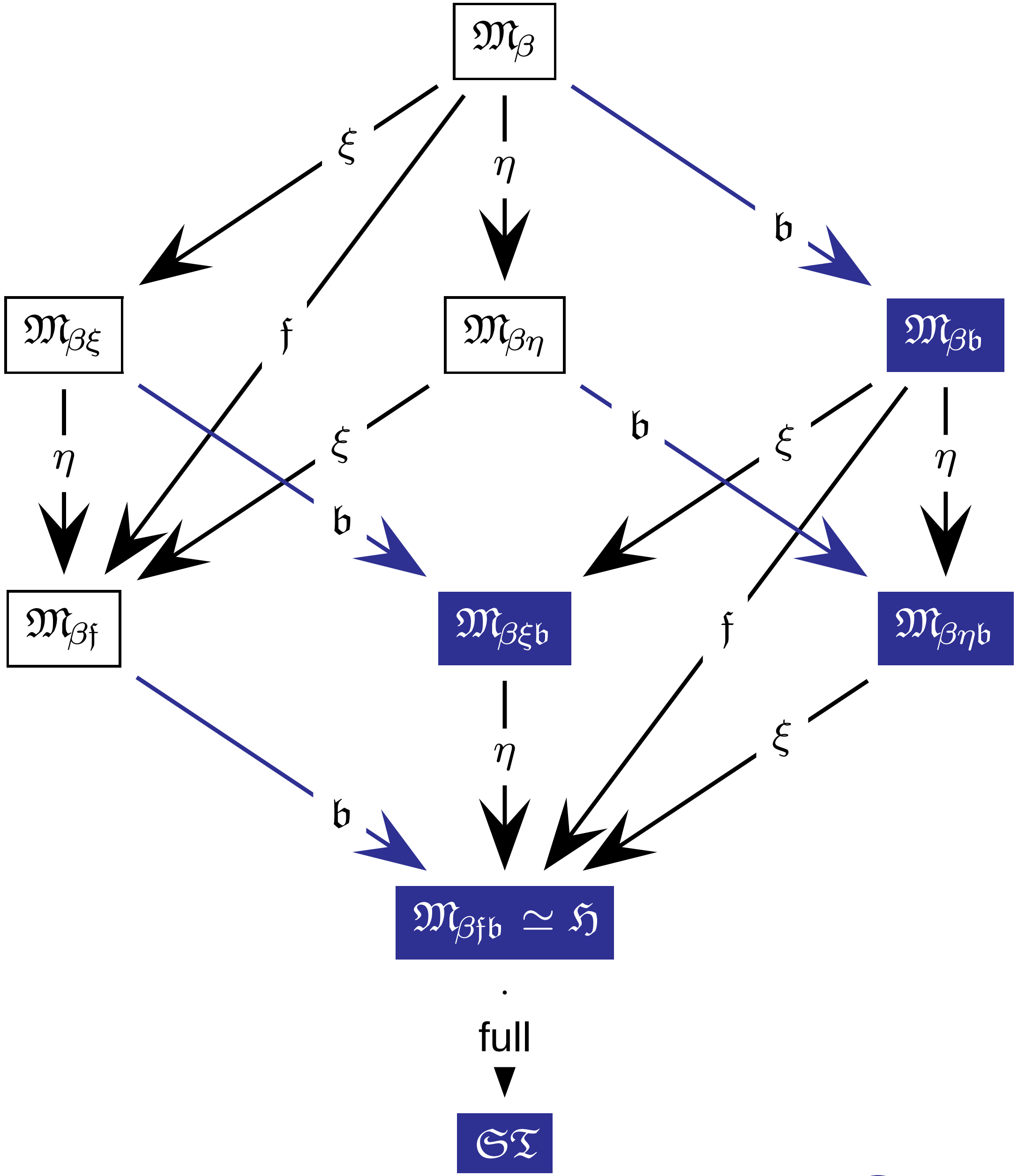
$\mathfrak{q}$: models provide identity relations

$$\forall\alpha : \mathsf{id} \in \mathcal{D}_{\alpha\to\alpha\to o}$$

Relationship of $\dot{=}$ and $=$

- [Andrews72]: without property $\mathfrak{q}$ Leibniz equality $\dot{=}$ not necessarily evaluates to identity relation
- $(\mathsf{a}_\alpha \doteq^\alpha \mathsf{b}_\alpha) \Rightarrow (\mathsf{a} =^\alpha \mathsf{b})$ (property $\mathfrak{q}$ corresponds to requiring this to be a theorem)

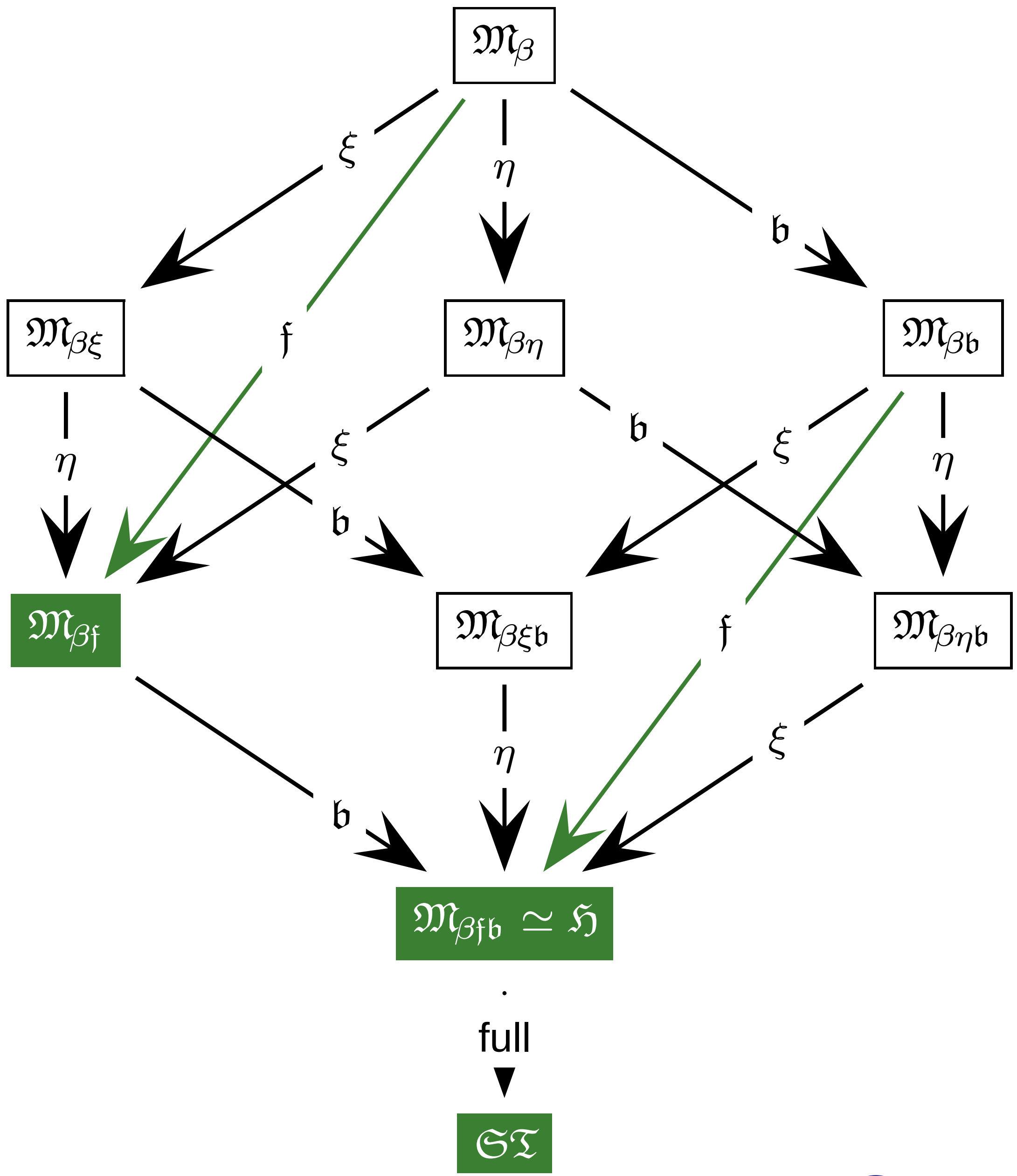
Semantics: HOL-CUBE



$\mathfrak{b}$: models are Boolean extensional

$$\mathcal{D}_o \equiv \{\bot, \top\}$$

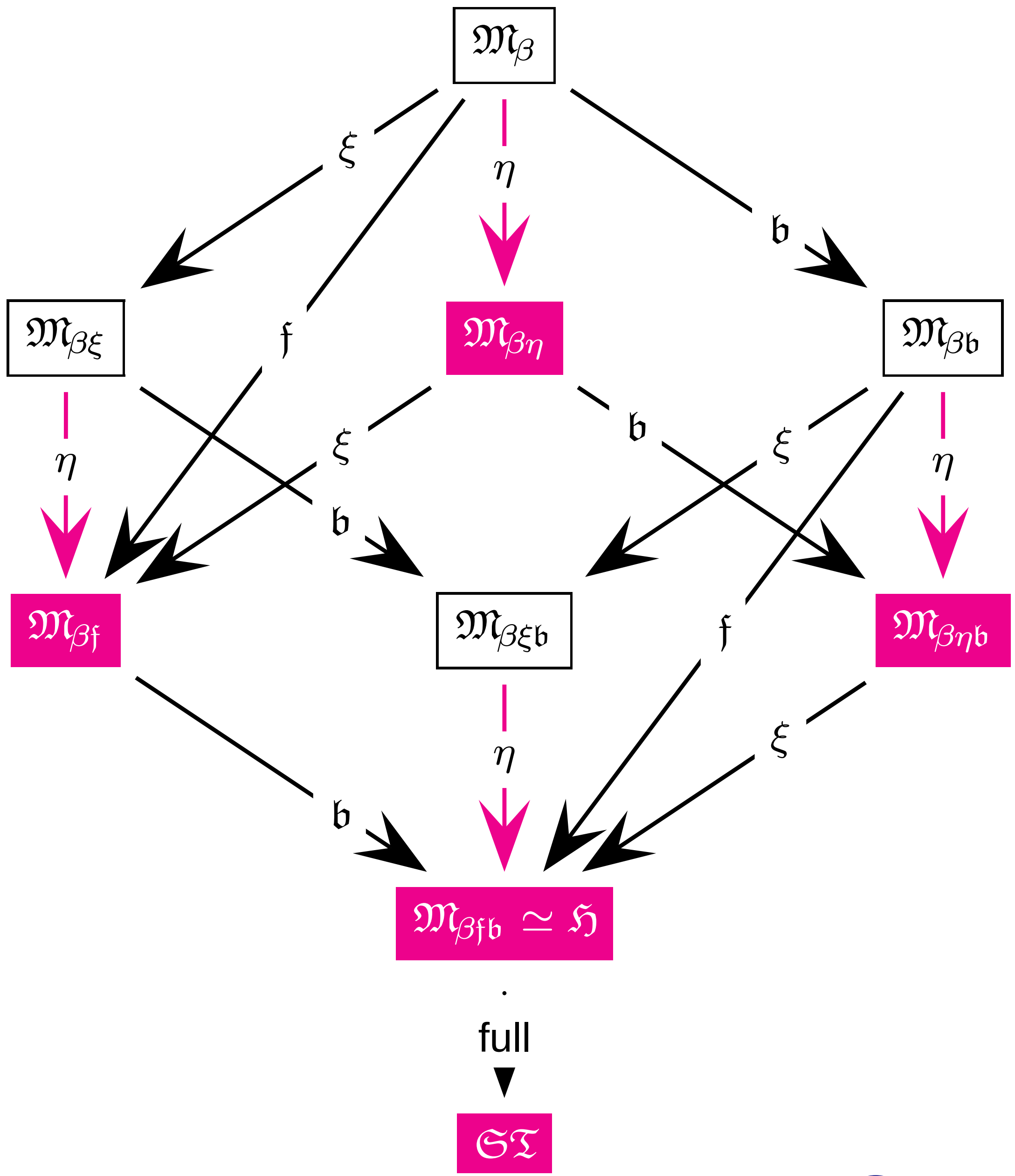
Semantics: HOL-CUBE



$\mathfrak{f}$: models are functional

$$\forall f, g \in \mathcal{D}_{\beta\alpha} :$$
$$f \equiv g \text{ iff } f@a \equiv g@a \; (\forall a \in \mathcal{D}_\alpha)$$

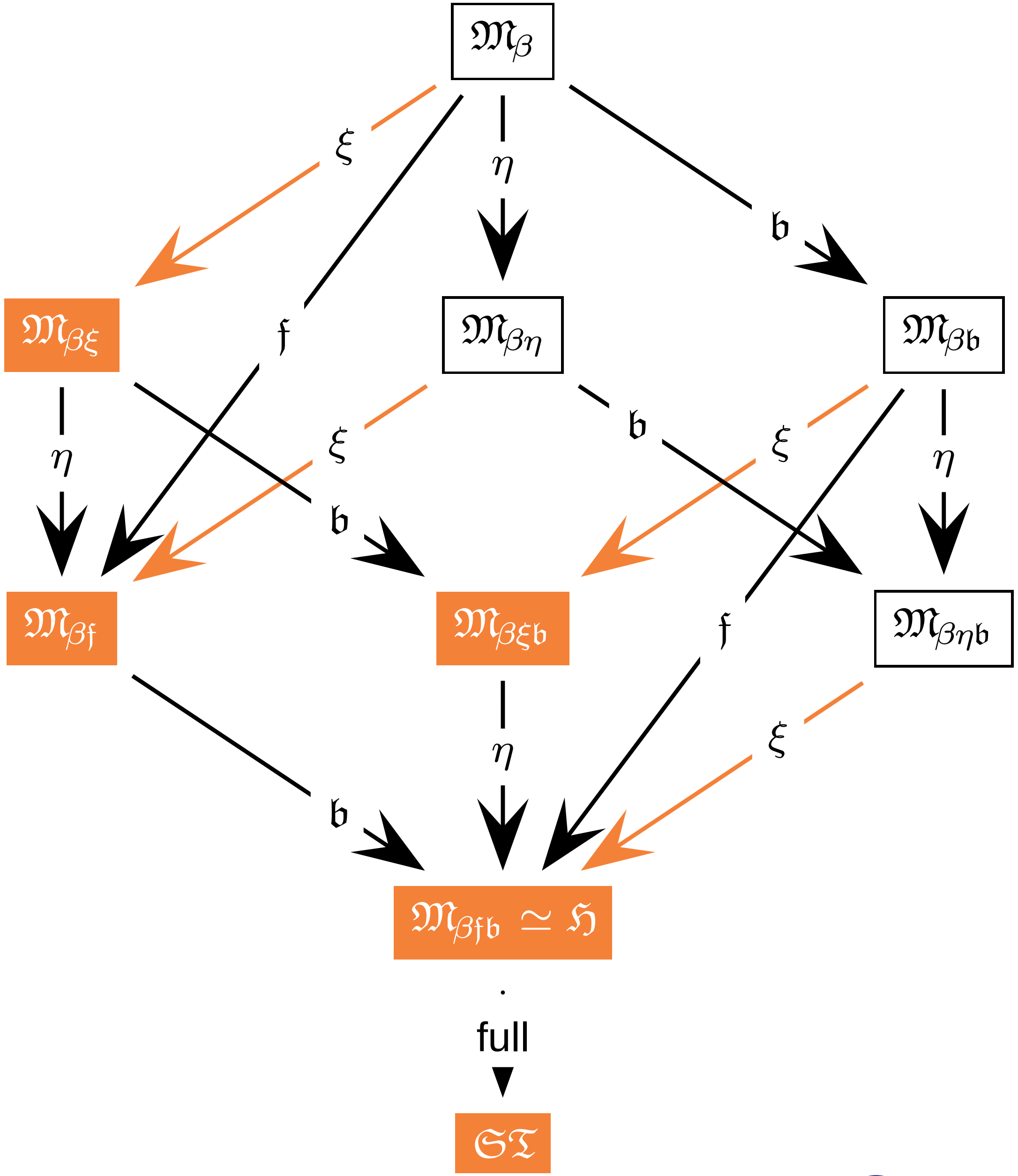
Semantics: HOL-CUBE



η: models are η-functional

$$\mathcal{E}_{\varphi}(\mathbf{A}) \equiv \mathcal{E}_{\varphi}(\mathbf{A}\downarrow_{\beta\eta})$$

Semantics: HOL-CUBE



ξ: models are ξ-functional

$$\mathcal{E}_\varphi(\lambda X_\alpha.\, \mathbf{M}_\beta) \equiv \mathcal{E}_\varphi(\lambda X_\alpha.\, \mathbf{N}_\beta) \text{ iff}$$
$$\mathcal{E}_{\varphi,[a/X]}(\mathbf{M}) \equiv \mathcal{E}_{\varphi,[a/X]}(\mathbf{N}) \ (\forall a \in \mathcal{D}_\alpha)$$

HOL-CUBE: Abstract Consistency

- Abstract consistency properties are provided in [Benzm.BrownKohlhase-JSL-04]

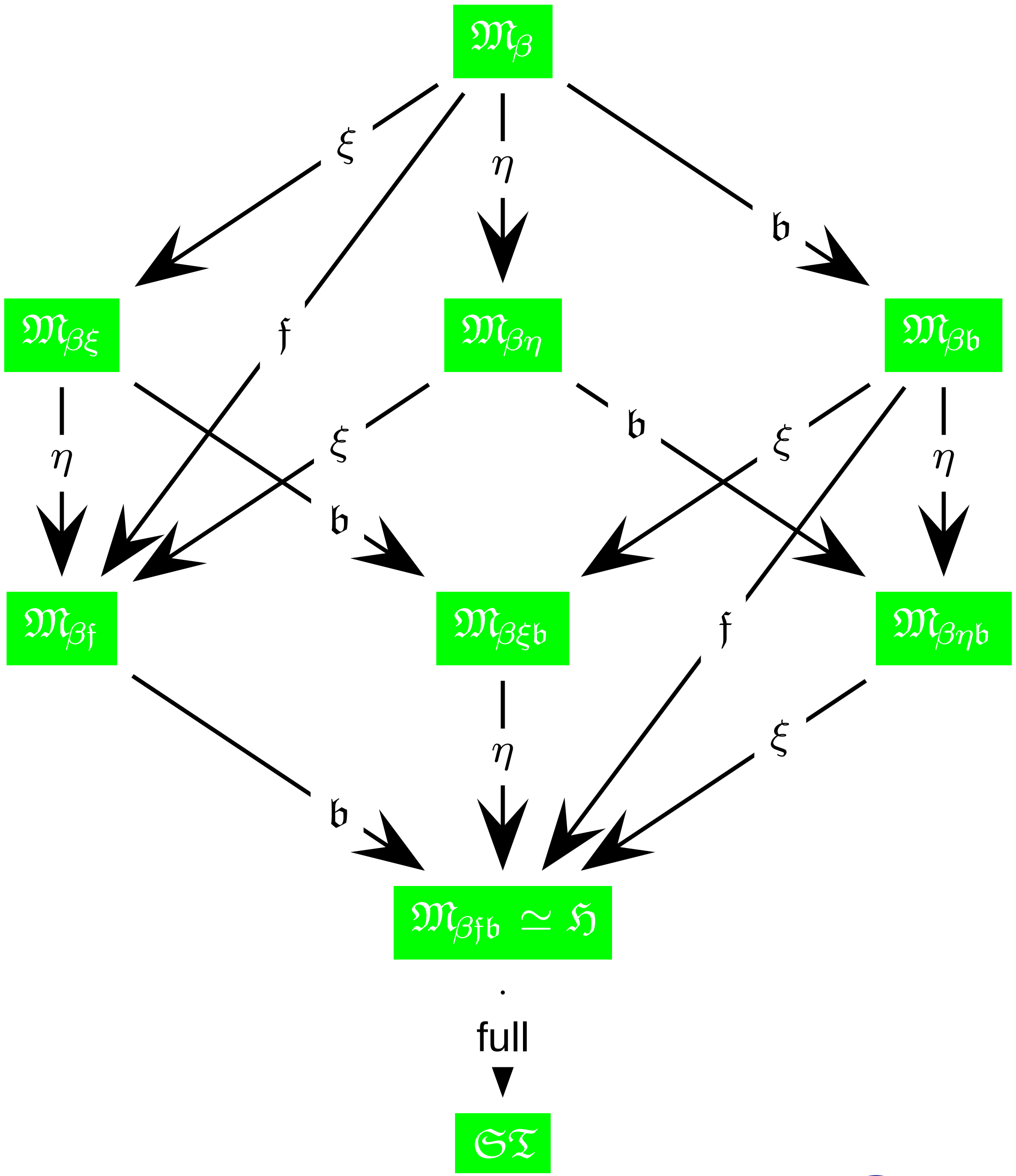
HOL-CUBE: Abstract Consistency

- Abstract consistency properties are provided in [Benzm.BrownKohlhase-JSL-04]
- They support completeness and soundness analysis of calculi by syntactical means for the HOL-CUBE

HOL-CUBE: Abstract Consistency

- Abstract consistency properties are provided in [Benzm.BrownKohlhase-JSL-04]
- They support completeness and soundness analysis of calculi by syntactical means for the HOL-CUBE
- Proposal:
  use the examples of this paper before trying a formal analysis

HOL-Problems requiring β



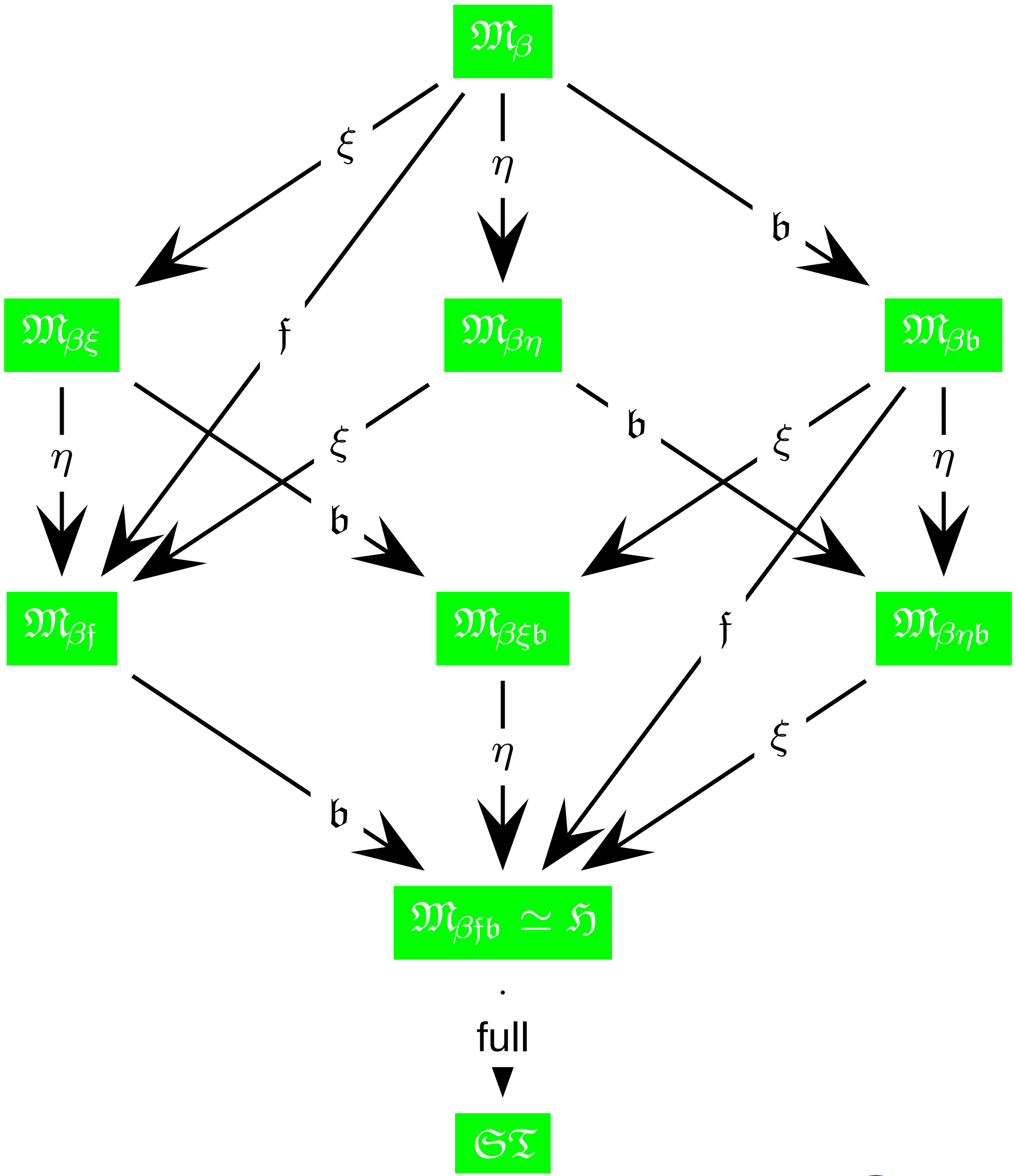
Church numerals:

$\overline{\mathsf{n}}^\alpha := (\lambda F_{\alpha\alpha}\lambda Y_\alpha\text{.}\ (F^n\ Y))$

$\overline{+} := \lambda M\lambda N\lambda F\lambda Y\text{.}\ MF(NFY)$

$\overline{\times} := \lambda M\lambda N\lambda F\lambda Z\text{.}\ N(MF)Z$

HOL-Problems requiring β



Church numerals:

$\overline{\mathsf{n}}^{\alpha} := (\lambda F_{\alpha\alpha} \lambda Y_{\alpha}.\ (F^{n}\ Y))$

$\overline{+} := \lambda M \lambda N \lambda F \lambda Y.\ MF(NFY)$

$\overline{\times} := \lambda M \lambda N \lambda F \lambda Z.\ N(MF)Z$

Efficiency of β-conversion:

- $\overline{3}\overline{\times}\overline{4} \stackrel{*}{\doteq} \overline{5}\overline{+}\overline{7}$
- $(\overline{10}\overline{\times}\overline{10})\overline{\times}\overline{10} \stackrel{*}{\doteq} ((\overline{10}\overline{\times}\overline{5})\overline{+}(\overline{5}\overline{\times}\overline{10}))\overline{\times}\overline{10}$

UNIVERSITÄT DES SAARLANDES

HOL-Problems requiring β

$$
\begin{tikzcd}
 & & \mathfrak{M}_\beta \arrow[dll, "\xi"'] \arrow[d, "\eta"] \arrow[drr, "\mathfrak{b}"] \arrow[ddll, "\mathfrak{f}"] & & \\
\mathfrak{M}_{\beta\xi} \arrow[d, "\eta"] \arrow[drr, "\mathfrak{b}"] & & \mathfrak{M}_{\beta\eta} \arrow[dll, "\xi"'] \arrow[drr, "\mathfrak{b}"] & & \mathfrak{M}_{\beta\mathfrak{b}} \arrow[dll, "\xi"'] \arrow[d, "\eta"] \arrow[ddll, "\mathfrak{f}"] \\
\mathfrak{M}_{\beta\mathfrak{f}} \arrow[drr, "\mathfrak{b}"] & & \mathfrak{M}_{\beta\xi\mathfrak{b}} \arrow[d, "\eta"] & & \mathfrak{M}_{\beta\eta\mathfrak{b}} \arrow[dll, "\xi"'] \\
 & & \mathfrak{M}_{\beta\mathfrak{f}\mathfrak{b}} \simeq \mathfrak{H} \arrow[d, dotted, "\text{full}"] & & \\
 & & \mathfrak{ST} & &
\end{tikzcd}
$$

Church numerals:

$\overline{n}^\alpha := (\lambda F_{\alpha\alpha}\lambda Y_\alpha.\ (F^n\ Y))$

$\overline{+} := \lambda M\lambda N\lambda F\lambda Y.\ MF(NFY)$

$\overline{\times} := \lambda M\lambda N\lambda F\lambda Z.\ N(MF)Z$

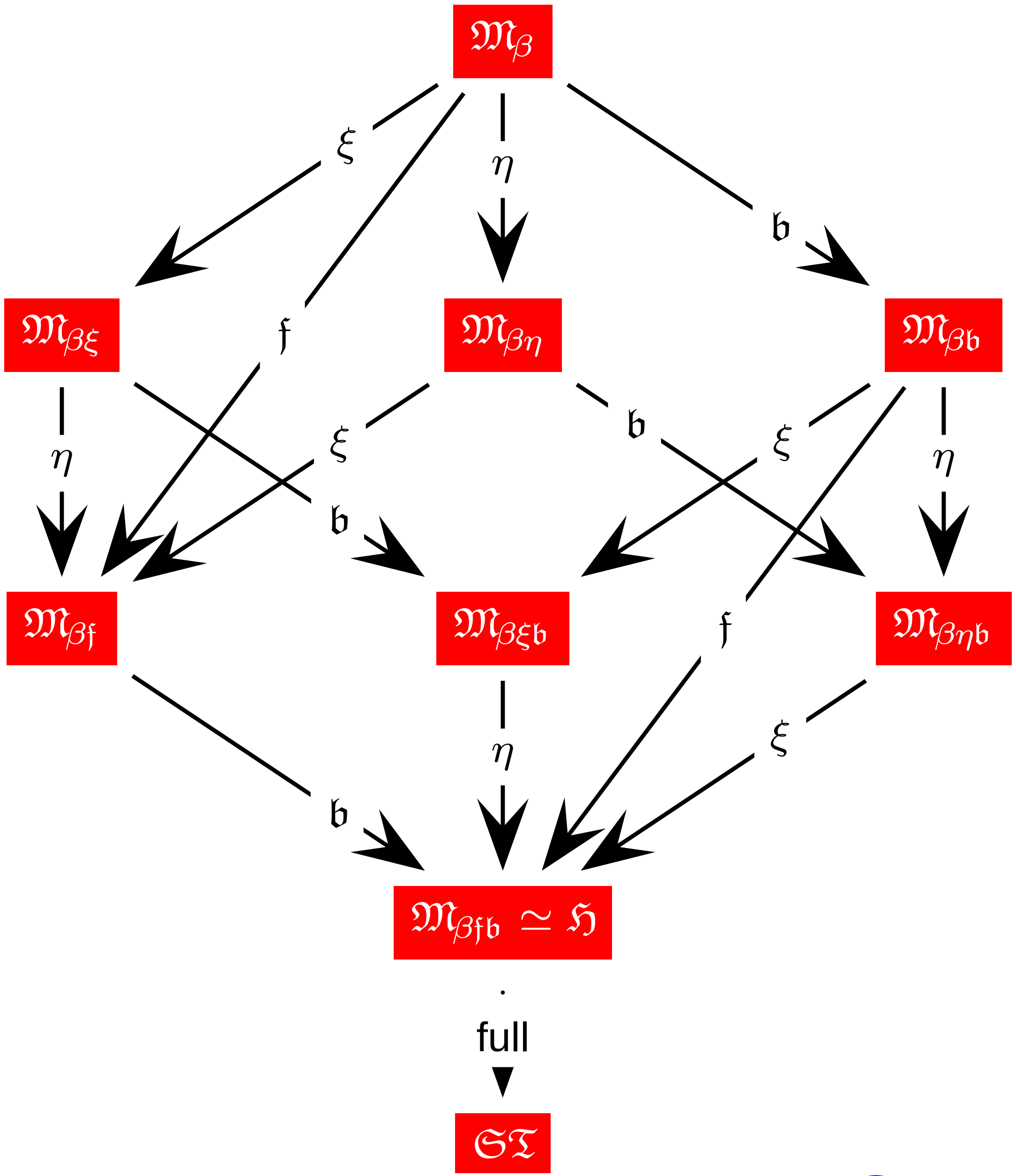
Efficiency of β-conversion:

- $\overline{3}\overline{\times}\overline{4} \stackrel{*}{\doteq} \overline{5}\overline{+}\overline{7}$
- $(\overline{10}\overline{\times}\overline{10})\overline{\times}\overline{10} \stackrel{*}{\doteq} ((\overline{10}\overline{\times}\overline{5})\overline{+}(\overline{5}\overline{\times}\overline{10}))\overline{\times}\overline{10}$

Pre-unification with β-conversion:

- $\exists N_{(\iota\to\iota)\to\iota\to\iota}.\ ((N\overline{\times}\overline{1}) \stackrel{*}{\doteq} \overline{1})$
  (two solutions if only β;
  one solution if $\beta\eta$)
- $\exists N.\ N\overline{\times}\overline{4} \stackrel{*}{\doteq} \overline{5}\overline{+}\overline{7}$
- $\exists H.\ ((H\ \overline{2})\overline{3}) \stackrel{*}{\doteq} \overline{6} \wedge ((H\ \overline{1})\overline{2}) \stackrel{*}{\doteq} \overline{2}$
- $\exists N, M.\ N\overline{\times}\overline{4} \stackrel{*}{\doteq} \overline{5}\overline{+}M$
  (infinitely many solutions!)

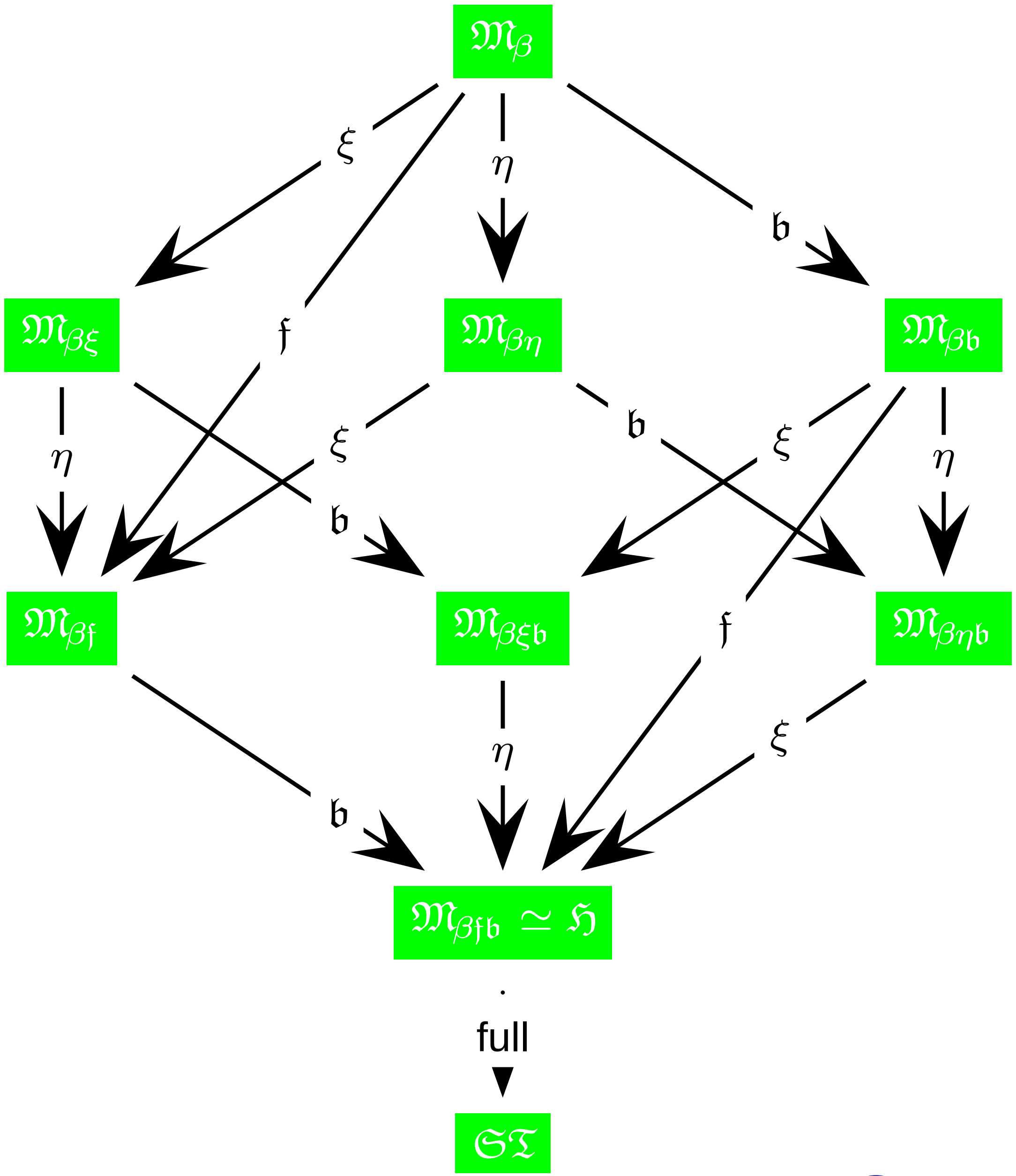
HOL-Non-Problems



Some non-theorems:

- essentially FOL
- apply to all model classes
- address
  - Skolemization
  - axiom of choice

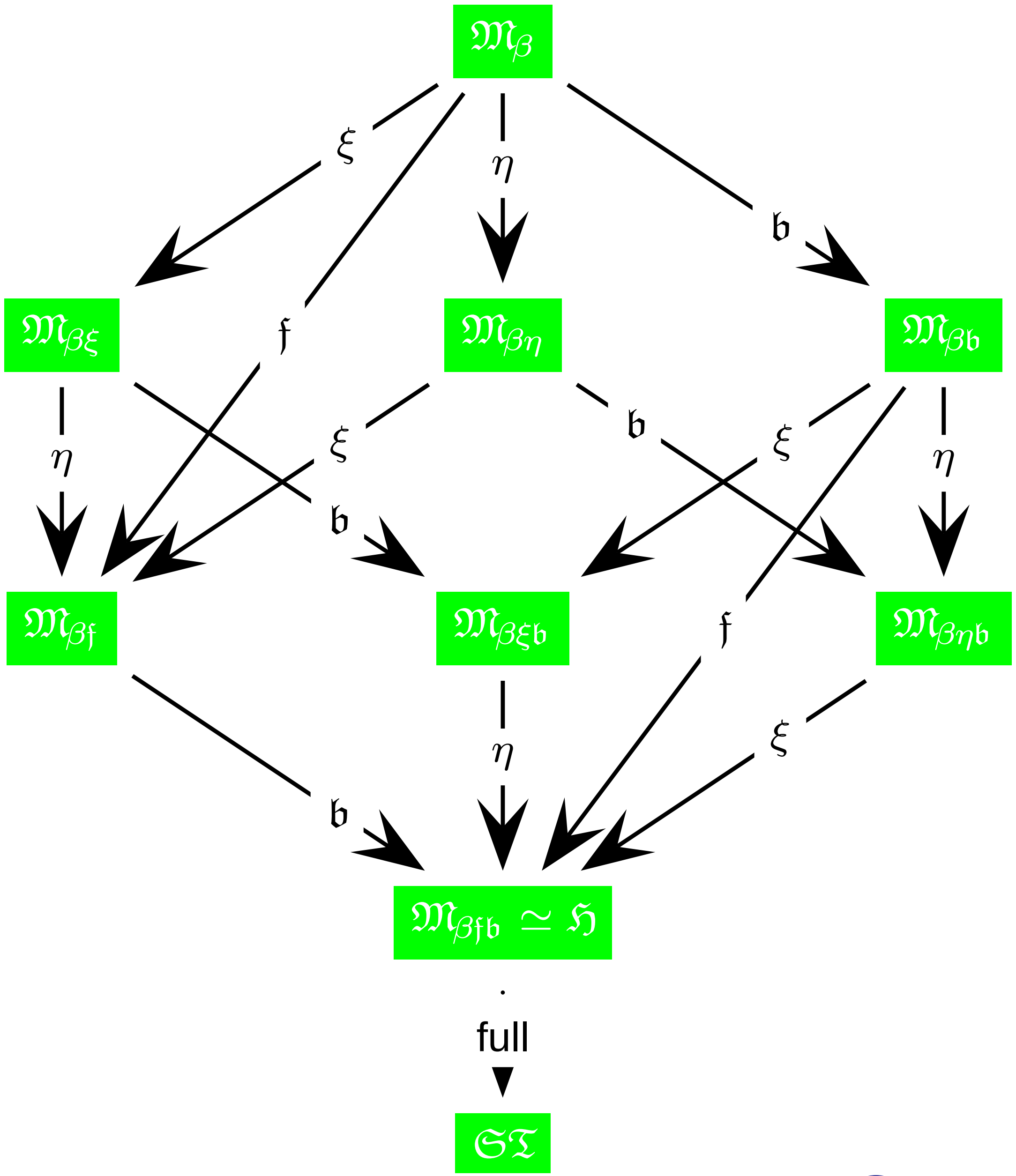
HOL-Problems requiring β



$\stackrel{*}{\doteq}$ is equivalence relation

- $\forall X_\alpha \centerdot X \stackrel{*}{\doteq} X$
- $\forall X_\alpha, Y_\alpha \centerdot X \stackrel{*}{\doteq} Y \Rightarrow Y \stackrel{*}{\doteq} X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha \centerdot (X \stackrel{*}{\doteq} Y \wedge Y \stackrel{*}{\doteq} Z) \Rightarrow X \stackrel{*}{\doteq} Z$

HOL-Problems requiring β



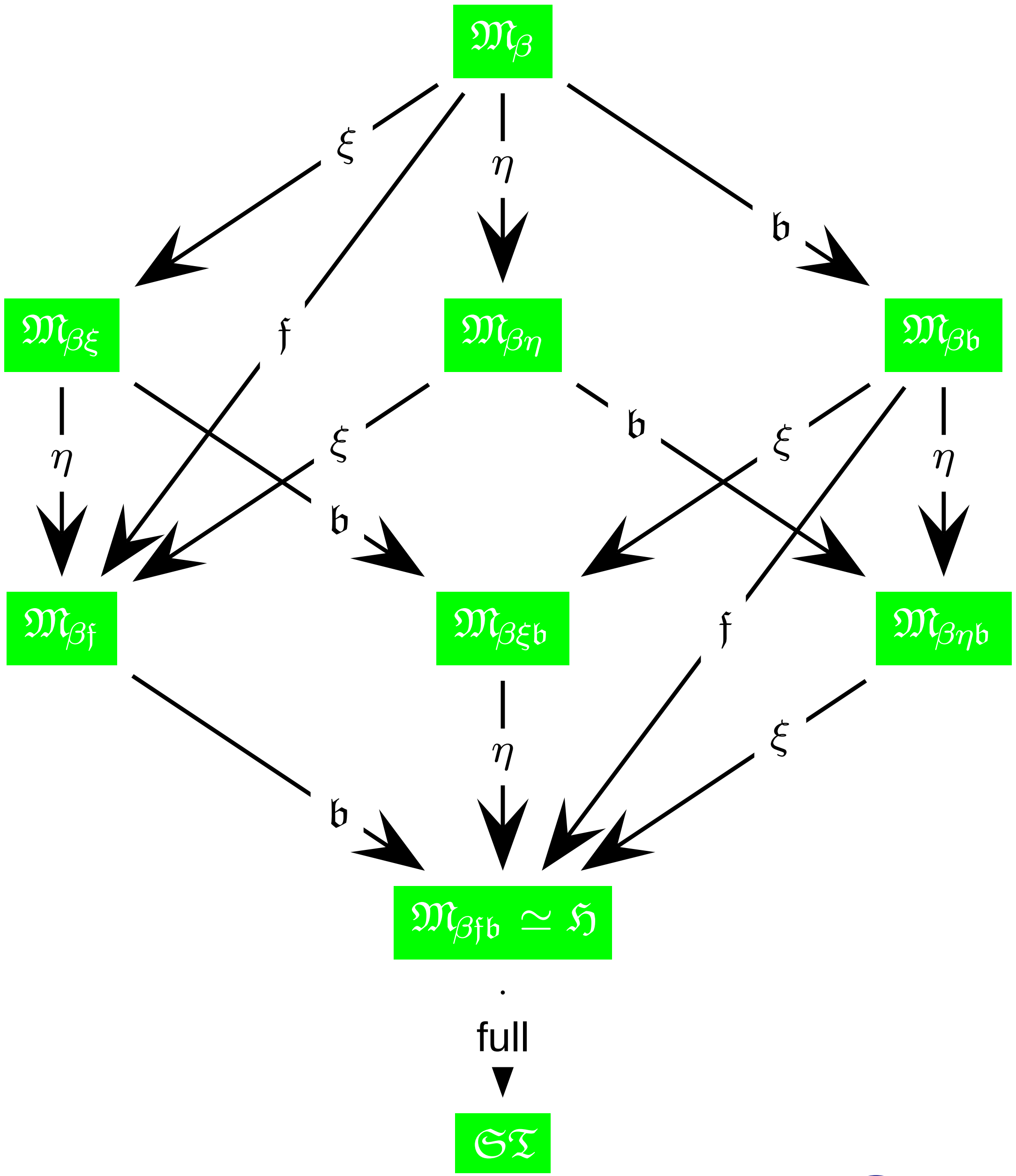
$\stackrel{*}{\doteq}$ is equivalence relation

- $\forall X_\alpha.\, X \stackrel{*}{\doteq} X$
- $\forall X_\alpha, Y_\alpha.\, X \stackrel{*}{\doteq} Y \Rightarrow Y \stackrel{*}{\doteq} X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha.\, (X \stackrel{*}{\doteq} Y \wedge Y \stackrel{*}{\doteq} Z) \Rightarrow X \stackrel{*}{\doteq} Z$

$\stackrel{*}{\doteq}$ is congruence relation

- $\forall X_\alpha, Y_\alpha, F_{\alpha\alpha}.\, X \stackrel{*}{\doteq} Y \Rightarrow (FX) \stackrel{*}{\doteq} (FY)$
- $\forall X_\alpha, Y_\alpha, P_{o\alpha}.\, X \stackrel{*}{\doteq} Y \wedge (PX) \Rightarrow (PY)$

HOL-Problems requiring β



$\stackrel{*}{\doteq}$ is equivalence relation

- $\forall X_\alpha.\ X \stackrel{*}{\doteq} X$
- $\forall X_\alpha, Y_\alpha.\ X \stackrel{*}{\doteq} Y \Rightarrow Y \stackrel{*}{\doteq} X$
- $\forall X_\alpha, Y_\alpha, Z_\alpha.\ (X \stackrel{*}{\doteq} Y \wedge Y \stackrel{*}{\doteq} Z) \Rightarrow X \stackrel{*}{\doteq} Z$

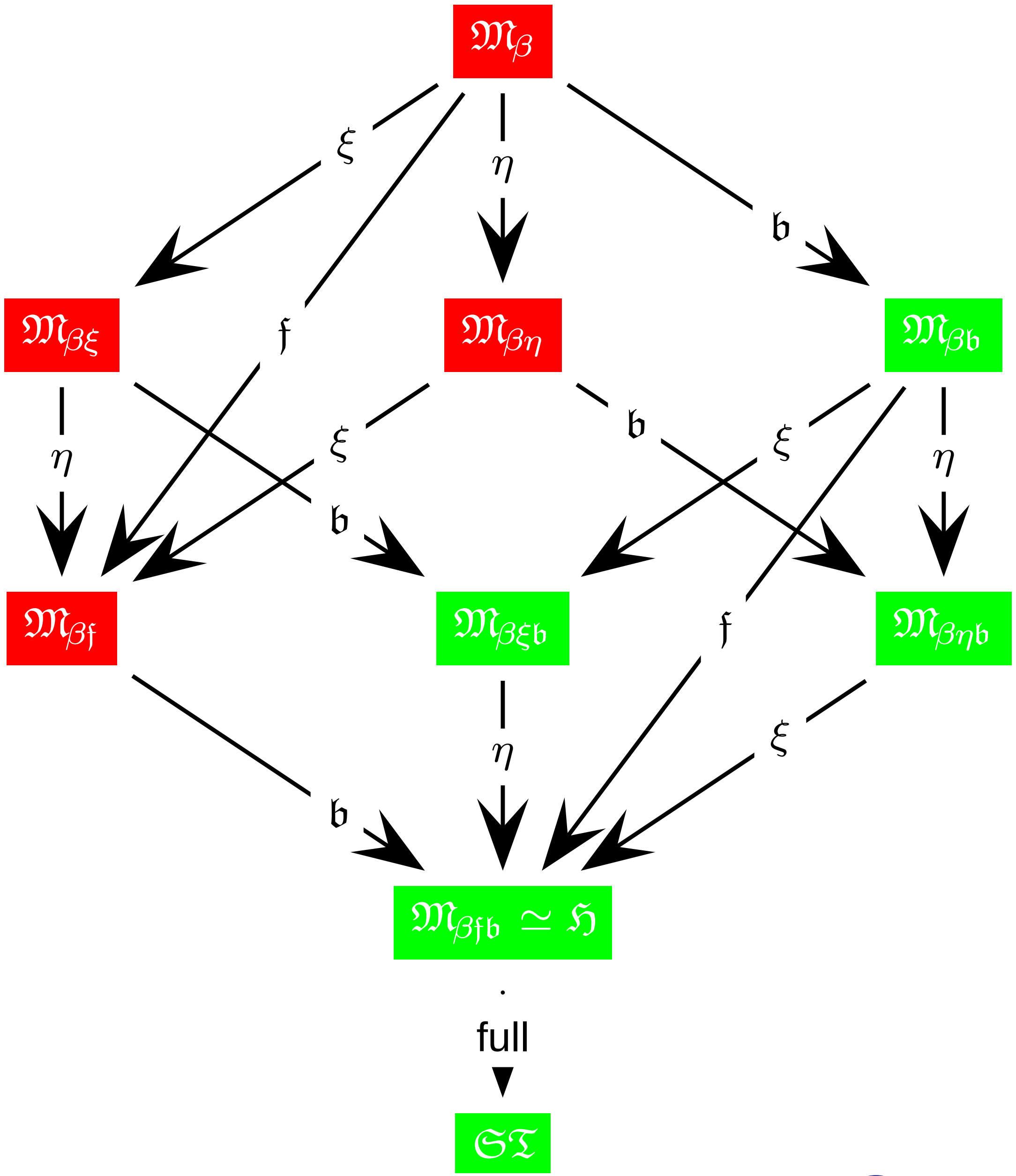
$\stackrel{*}{\doteq}$ is congruence relation

- $\forall X_\alpha, Y_\alpha, F_{\alpha\alpha}.\ X \stackrel{*}{\doteq} Y \Rightarrow (FX) \stackrel{*}{\doteq} (FY)$
- $\forall X_\alpha, Y_\alpha, P_{o\alpha}.\ X \stackrel{*}{\doteq} Y \wedge (PX) \Rightarrow (PY)$

Trivial directions of Boolean and functional extensionality

- $\forall A_o, B_o.\ A \stackrel{*}{\doteq} B \Rightarrow (A \Leftrightarrow B)$
- $\forall F_{\beta\alpha}, G_{\beta\alpha}.\ F \stackrel{*}{\doteq} G \Rightarrow (\forall X_\alpha.\ FX \stackrel{*}{\doteq} GX)$

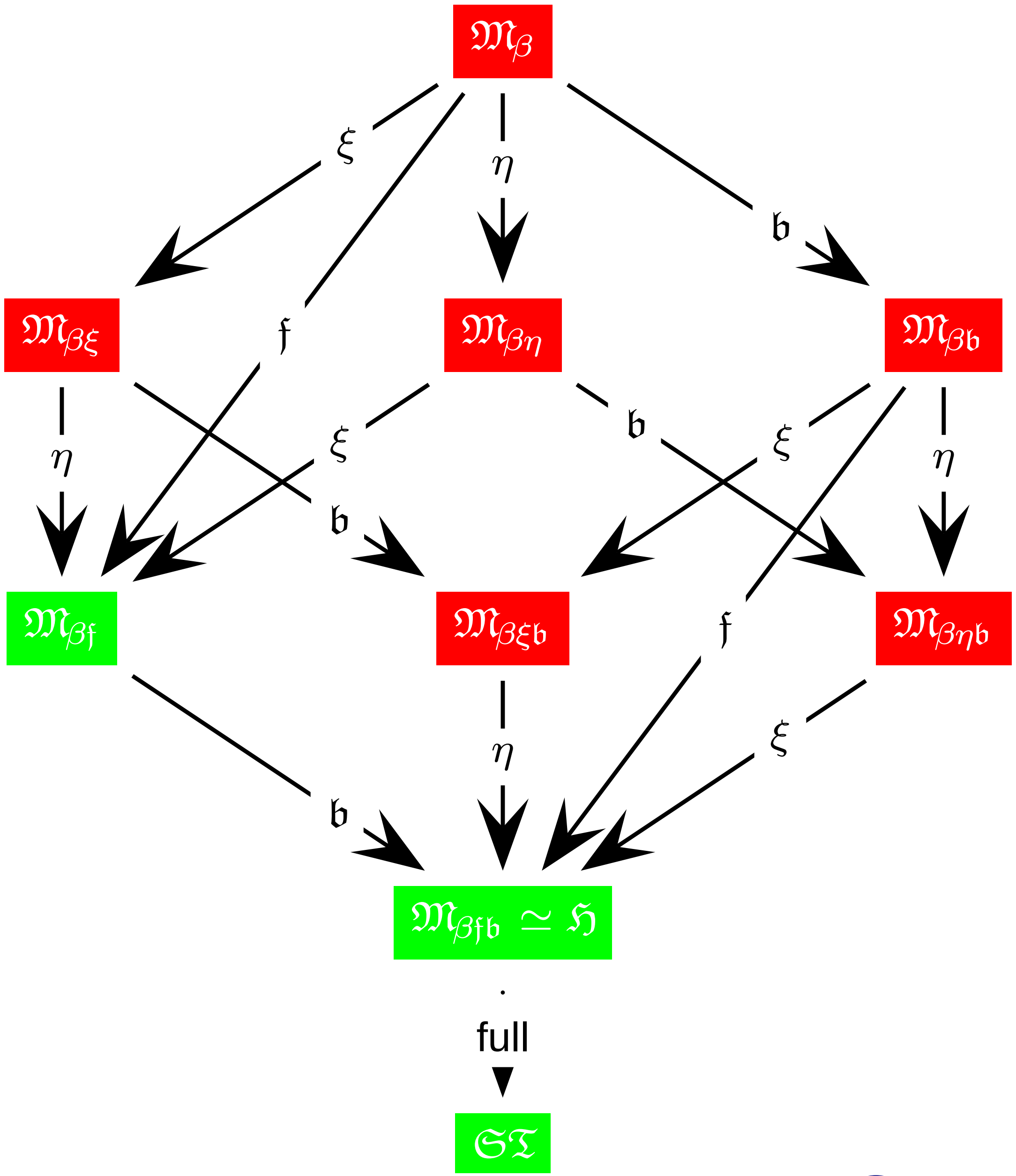
HOL-Problems requiring $\mathfrak{b}$



Non-trivial direction of Boolean extensionality

- $\forall A_o, B_o.\ (A \Leftrightarrow B) \Rightarrow A \stackrel{*}{=} B$

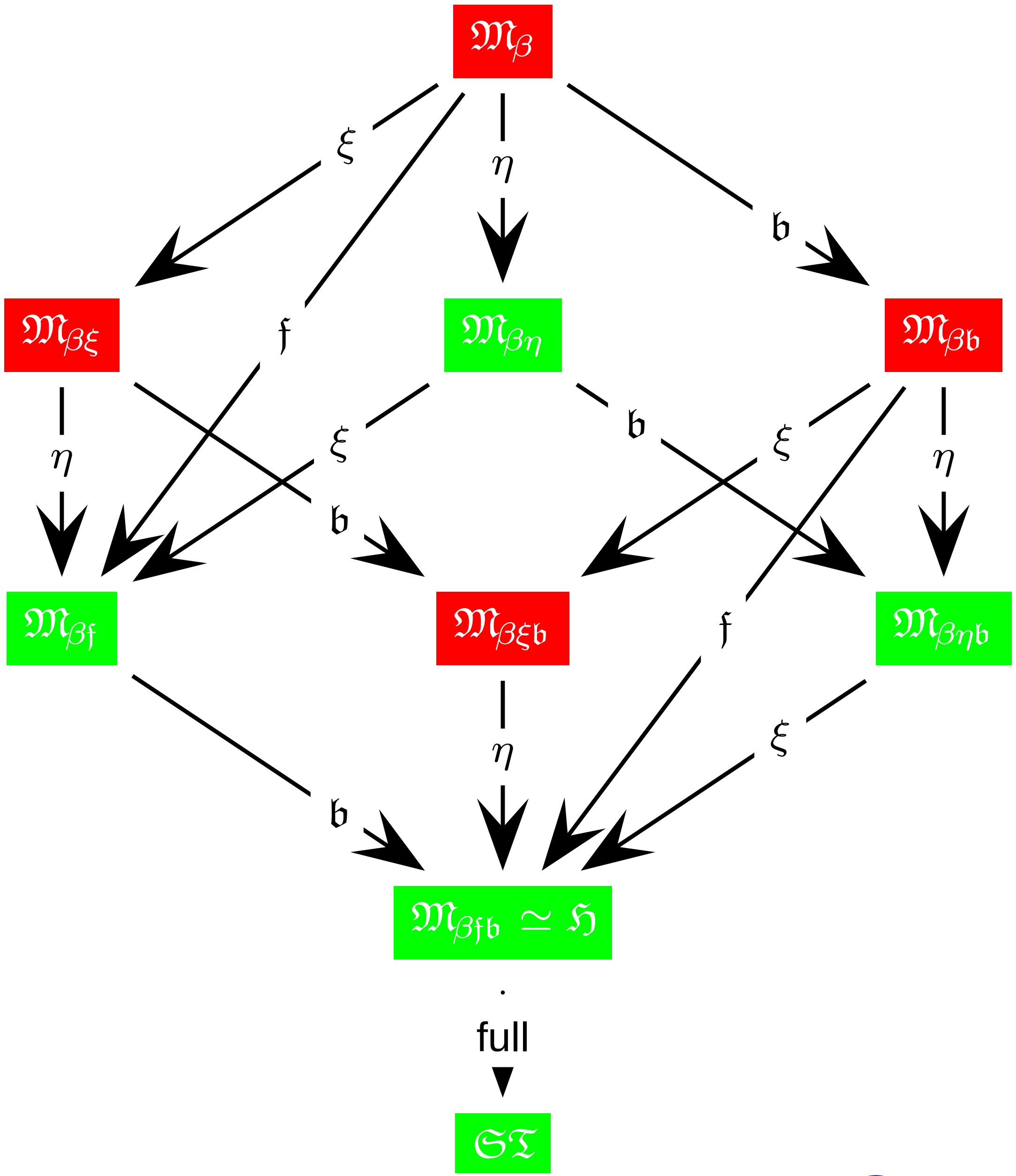
HOL-Problems requiring $\mathfrak{f}$



Non-trivial direct. of functional extensionality

- $\forall F_{\beta\alpha}, G_{\beta\alpha}. (\forall X_\alpha. FX \stackrel{*}{=} GX) \Rightarrow F \stackrel{*}{=} G$

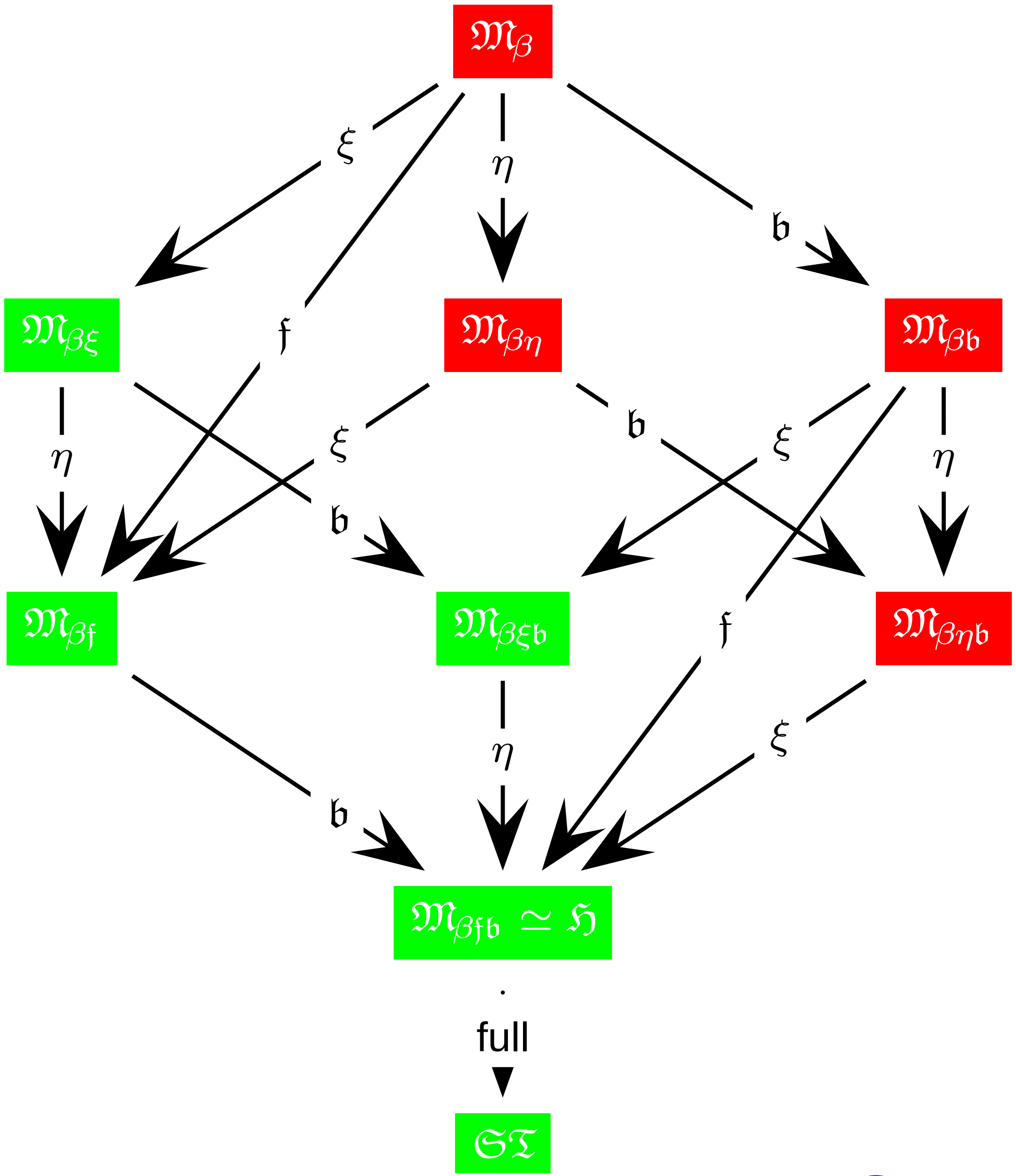
HOL-Problems requiring η



Example requiring property η

- $(\mathsf{p}_{o(\iota\iota)}(\lambda X_\iota.\, \mathsf{f}_{\iota\iota} X)) \Rightarrow (\mathsf{p}\ \mathsf{f})$

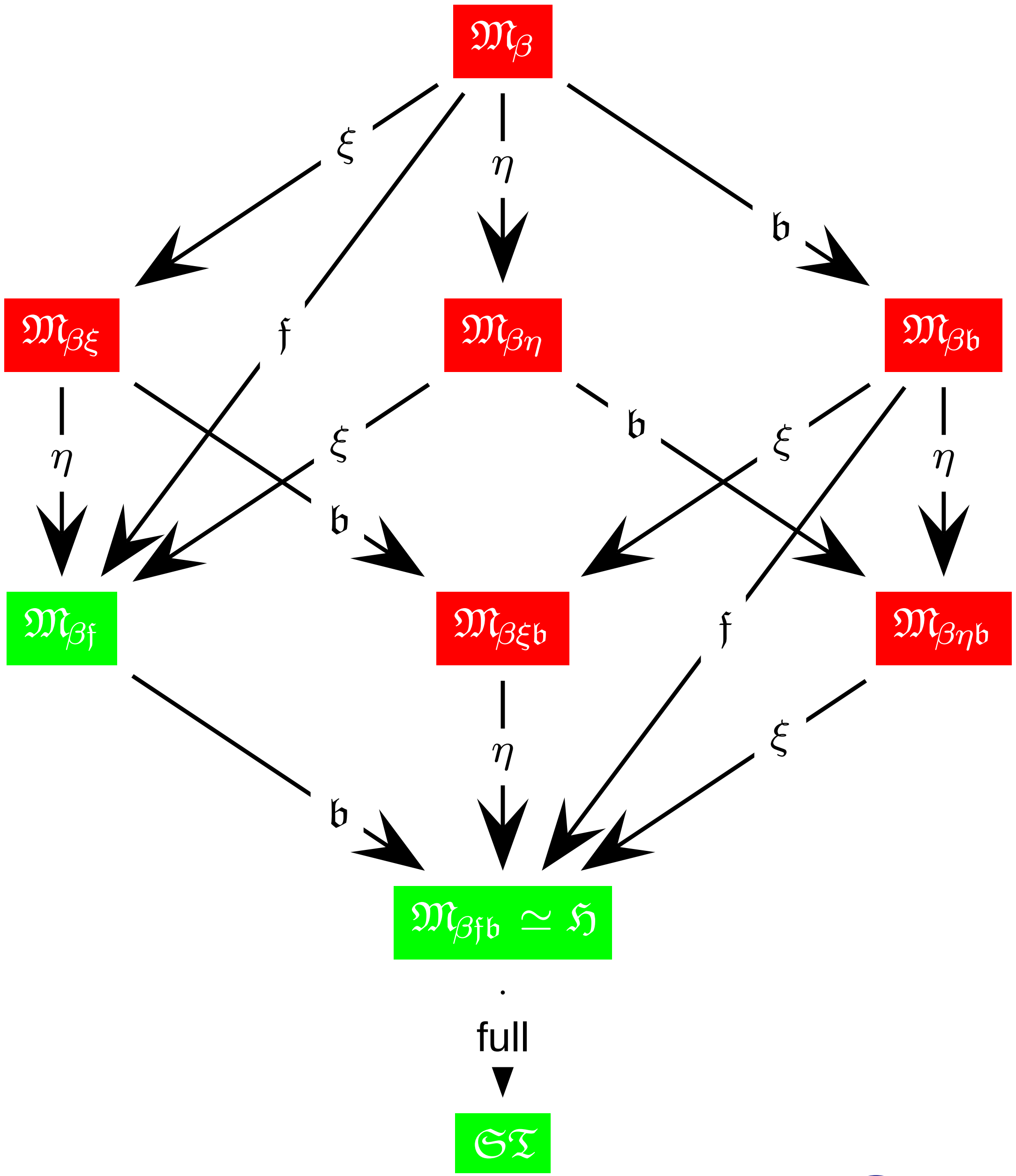
HOL-Problems requiring ξ



Example requiring property ξ (and $\mathfrak{q}$!)

- $(\forall X_\iota.\ (f_{\iota\iota} X) \stackrel{*}{\doteq} X) \wedge p_{o(\iota\iota)}(\lambda X_\iota.\ X)$
  $\Rightarrow p(\lambda X_\iota.\ fX)$

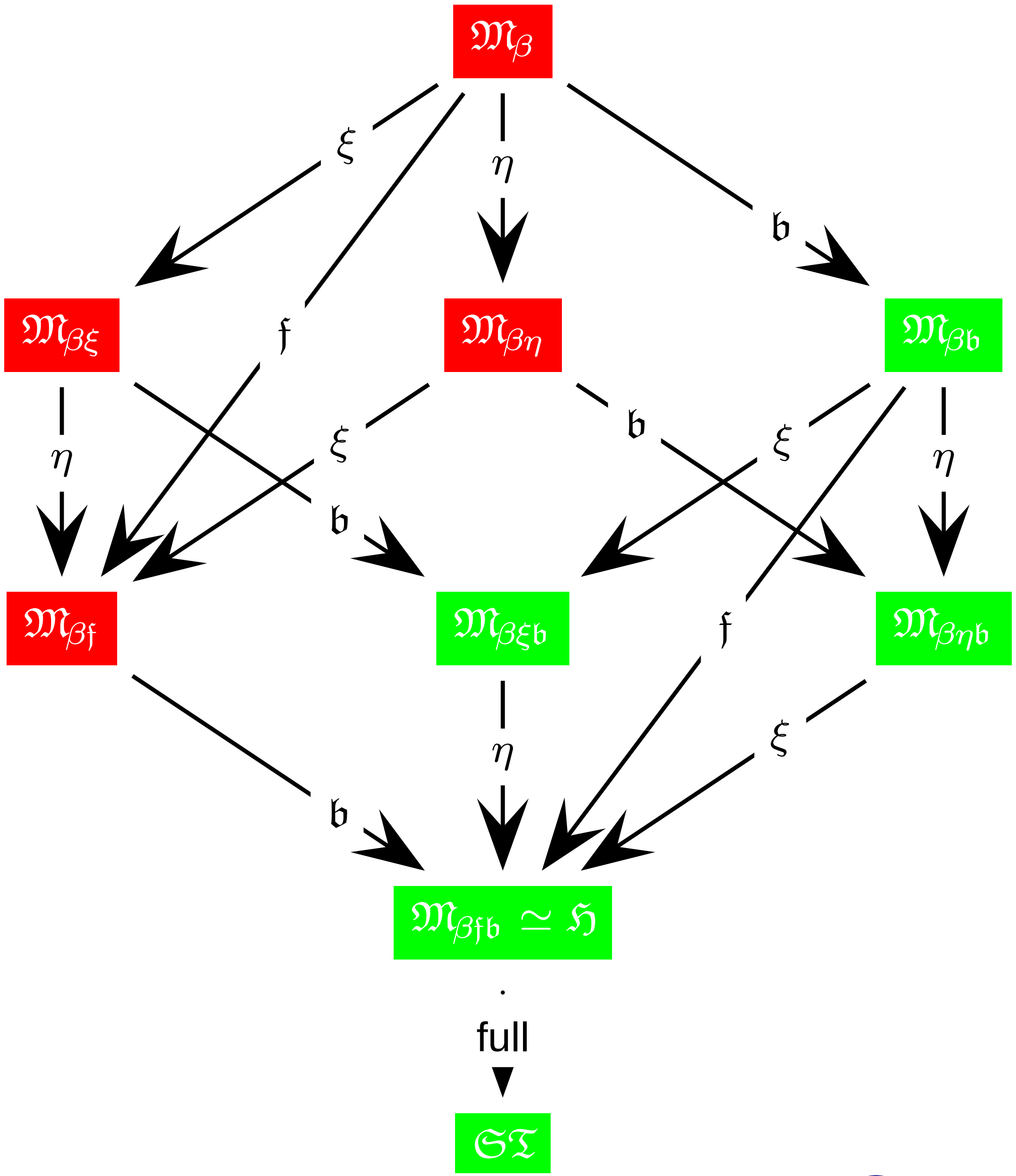
HOL-Problems requiring $\mathfrak{f}$



Example requiring property $\mathfrak{f}$ (and $\mathfrak{q}$!)

- $(\forall X_\iota.\ (f_{\iota\iota} X) \stackrel{*}{=} X) \wedge p_{o(\iota\iota)}(\lambda X_\iota.\ X)$
  $\Rightarrow (p\ f)$

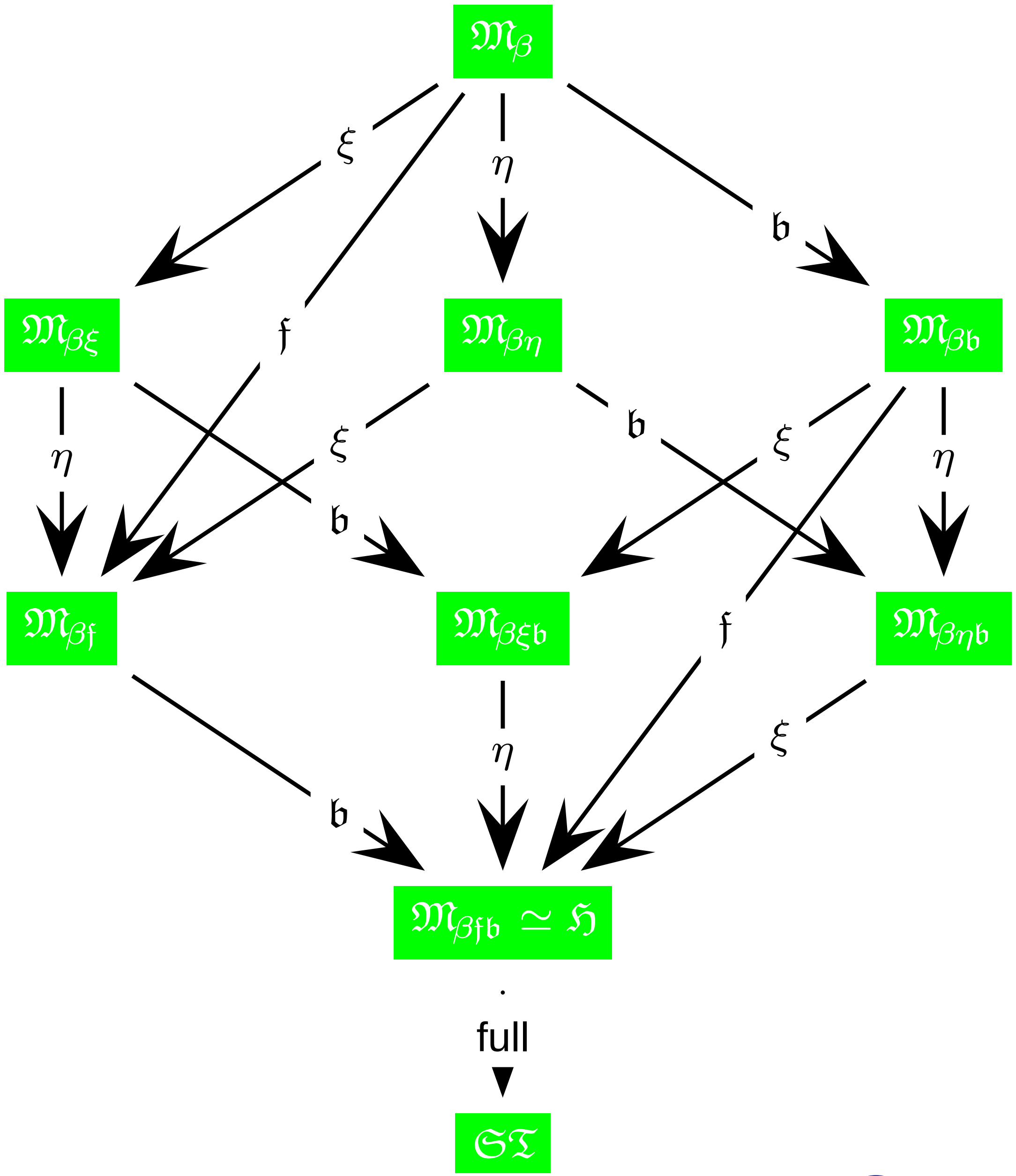
HOL-Problems requiring $\mathfrak{b}$



Examples requiring property $\mathfrak{b}$

- $(p_{oo}\ a_o) \wedge (p\ b_o) \Rightarrow (p\ (a \wedge b))$
- $\neg(a \stackrel{*}{=} \neg a)$ (in particular $\neg(a = \neg a)$)
- $(h_{\iota o}((h\top) \stackrel{*}{=} (h\bot))) \stackrel{*}{=} (h\bot)$

HOL-Problems: DeMorgan's Law



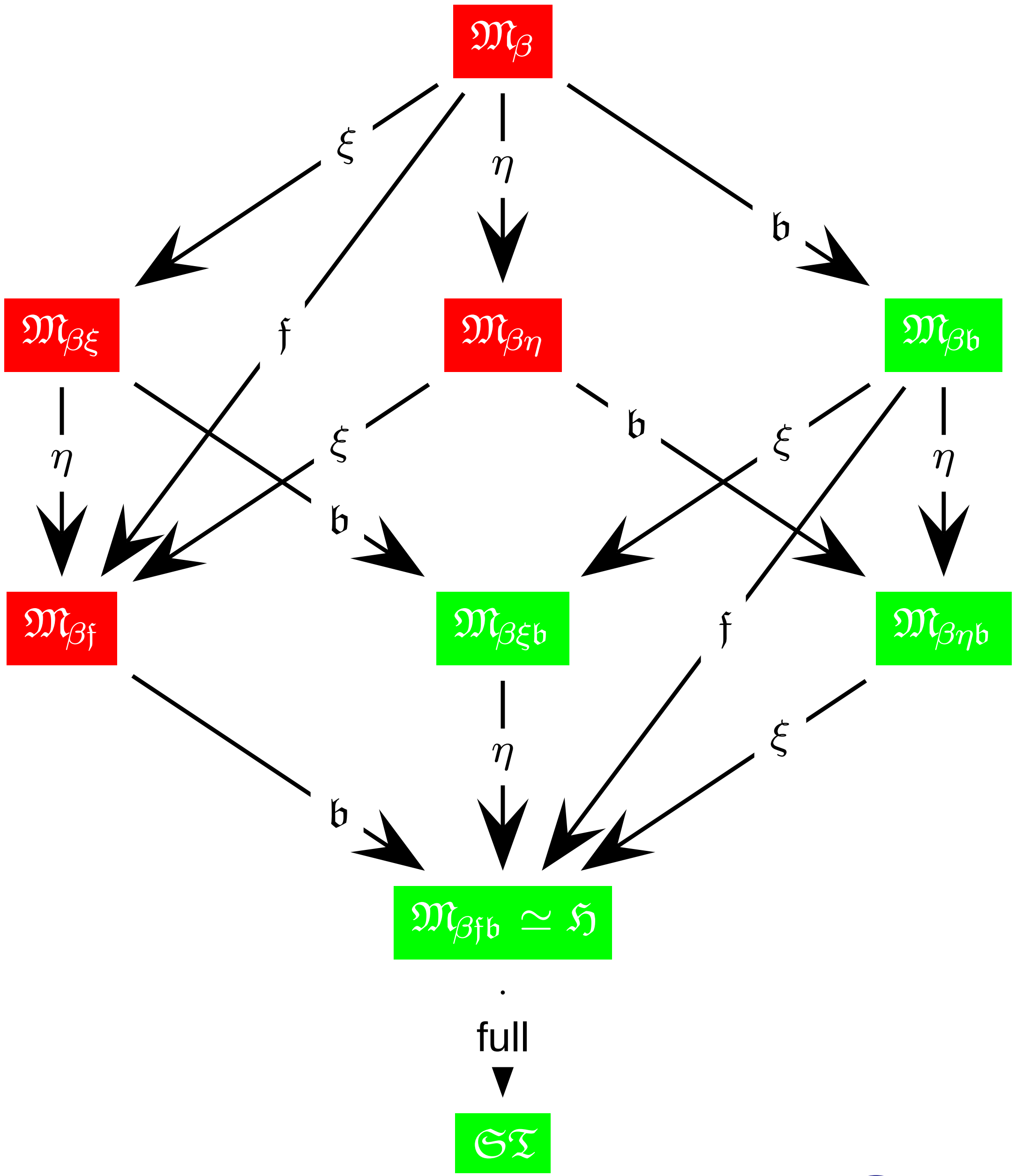
Playing with DeMorgan's Law:

- $\forall X, Y. X \wedge Y \Leftrightarrow \neg(\neg X \vee \neg Y)$

'Ok' for all model classes

UNIVERSITÄT DES SAARLANDES

HOL-Problems: DeMorgan's Law

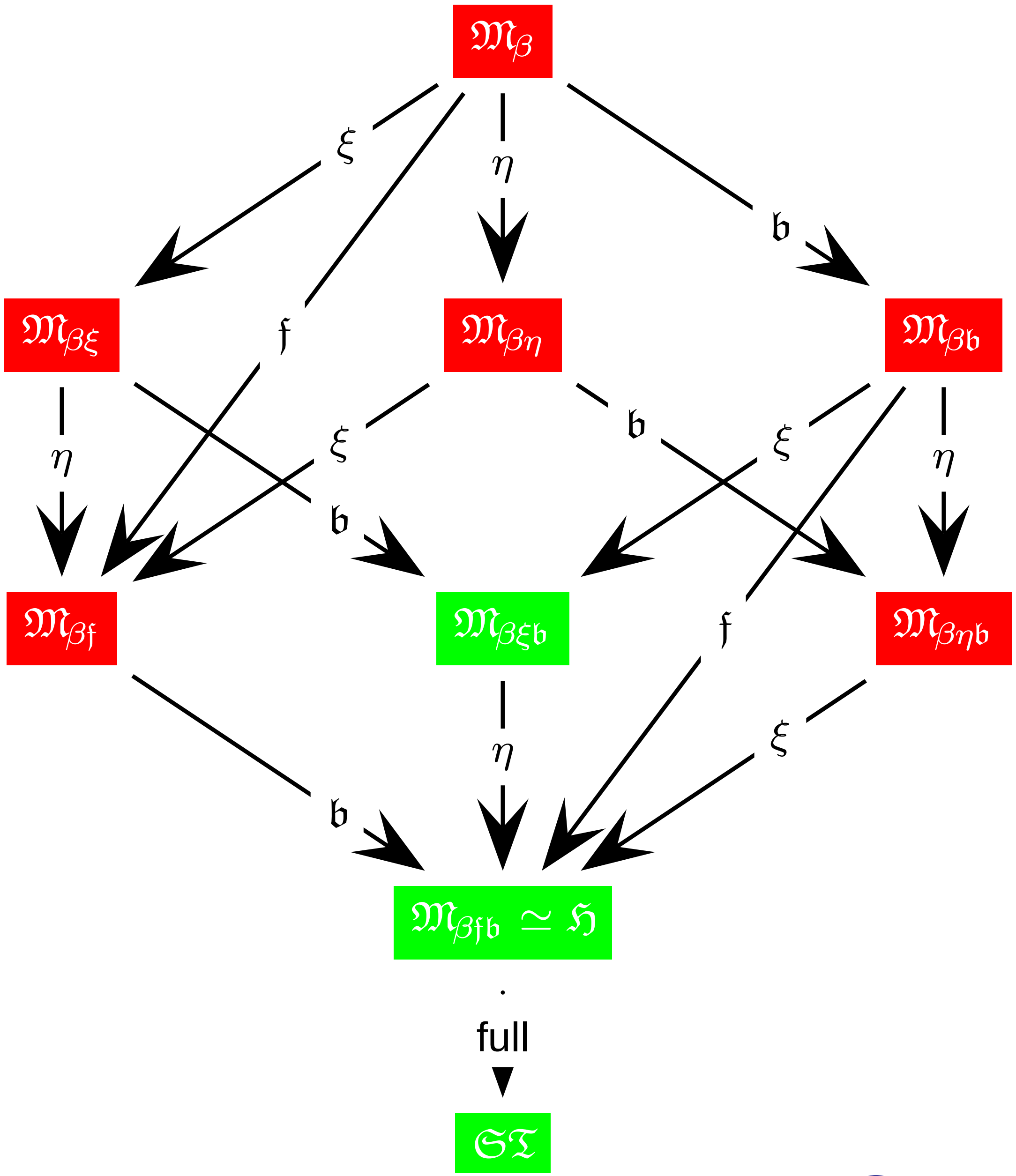


Playing with DeMorgan's Law:

- $\forall X, Y.\ X \wedge Y \Leftrightarrow \neg(\neg X \vee \neg Y)$
- $\forall X, Y.\ X \wedge Y \stackrel{*}{\doteq} \neg(\neg X \vee \neg Y)$

requires $\mathfrak{b}$

HOL-Problems: DeMorgan’s Law

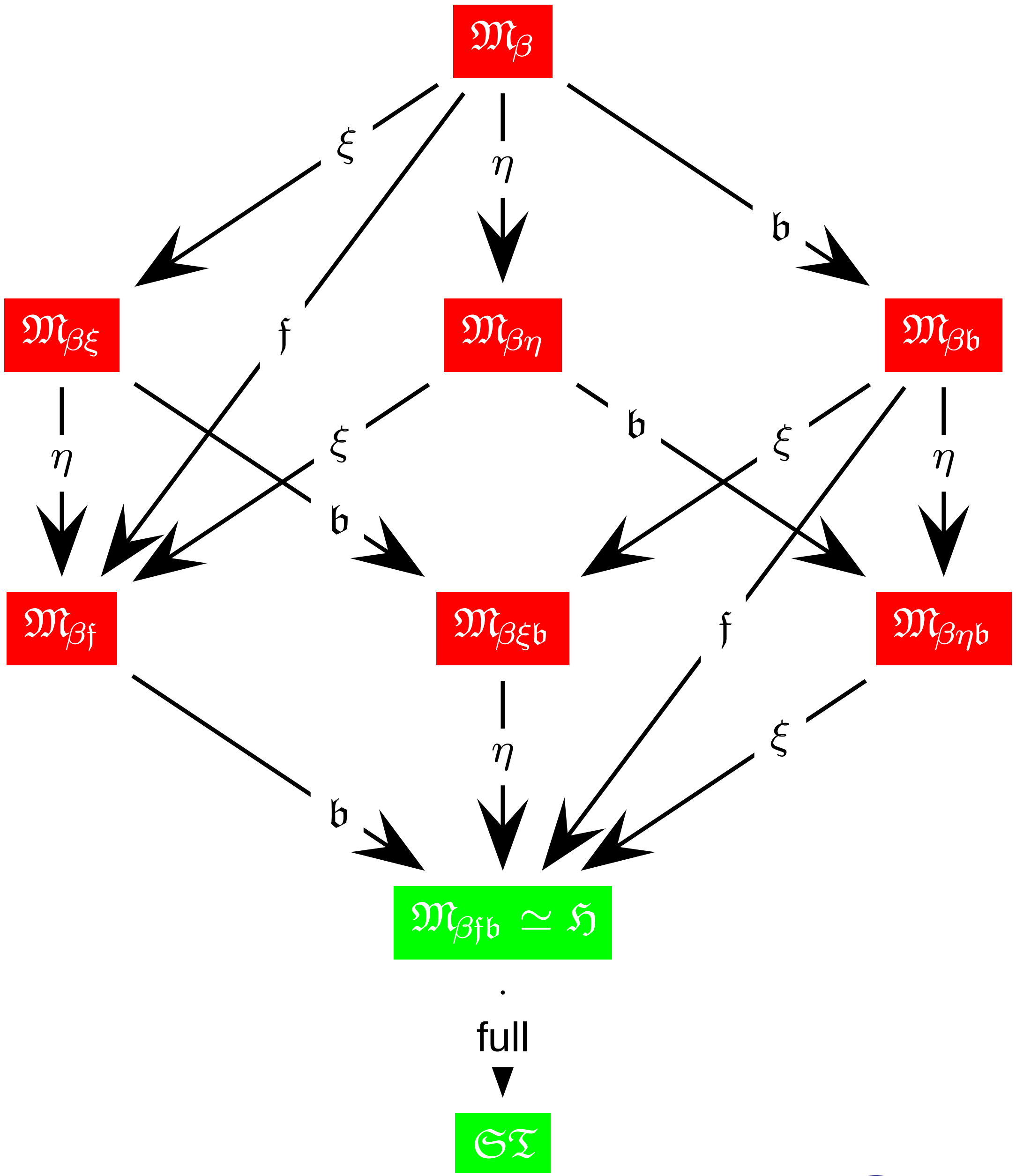


Playing with DeMorgan’s Law:

- $\forall X, Y.\ X \land Y \Leftrightarrow \neg(\neg X \lor \neg Y)$
- $\forall X, Y.\ X \land Y \stackrel{*}{\doteq} \neg(\neg X \lor \neg Y)$
- $(\lambda U \lambda V.\ U \land V) \stackrel{*}{\doteq} (\lambda X \lambda Y.\ \neg(\neg X \lor \neg Y))$

requires $\mathfrak{b}$ and ξ

HOL-Problems: DeMorgan's Law

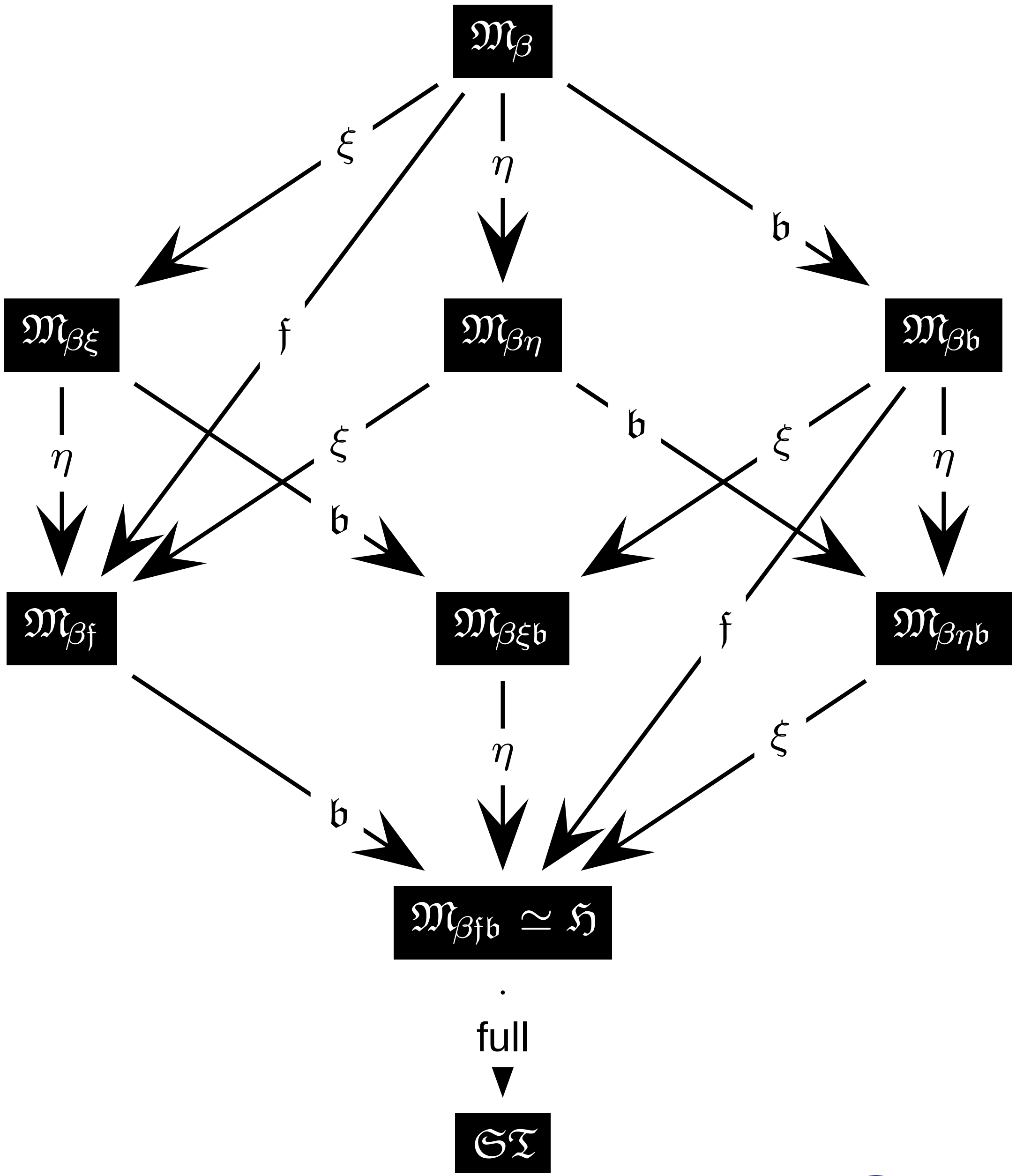


Playing with DeMorgan's Law:

- $\forall X, Y.\ X \wedge Y \Leftrightarrow \neg(\neg X \vee \neg Y)$
- $\forall X, Y.\ X \wedge Y \stackrel{*}{\doteq} \neg(\neg X \vee \neg Y)$
- $(\lambda U \lambda V.\ U \wedge V) \stackrel{*}{\doteq} (\lambda X \lambda Y.\ \neg(\neg X \vee \neg Y))$
- $\wedge \stackrel{*}{\doteq} (\lambda X \lambda Y.\ \neg(\neg X \vee \neg Y))$

requires $\mathfrak{b}$ and $\mathfrak{f}$

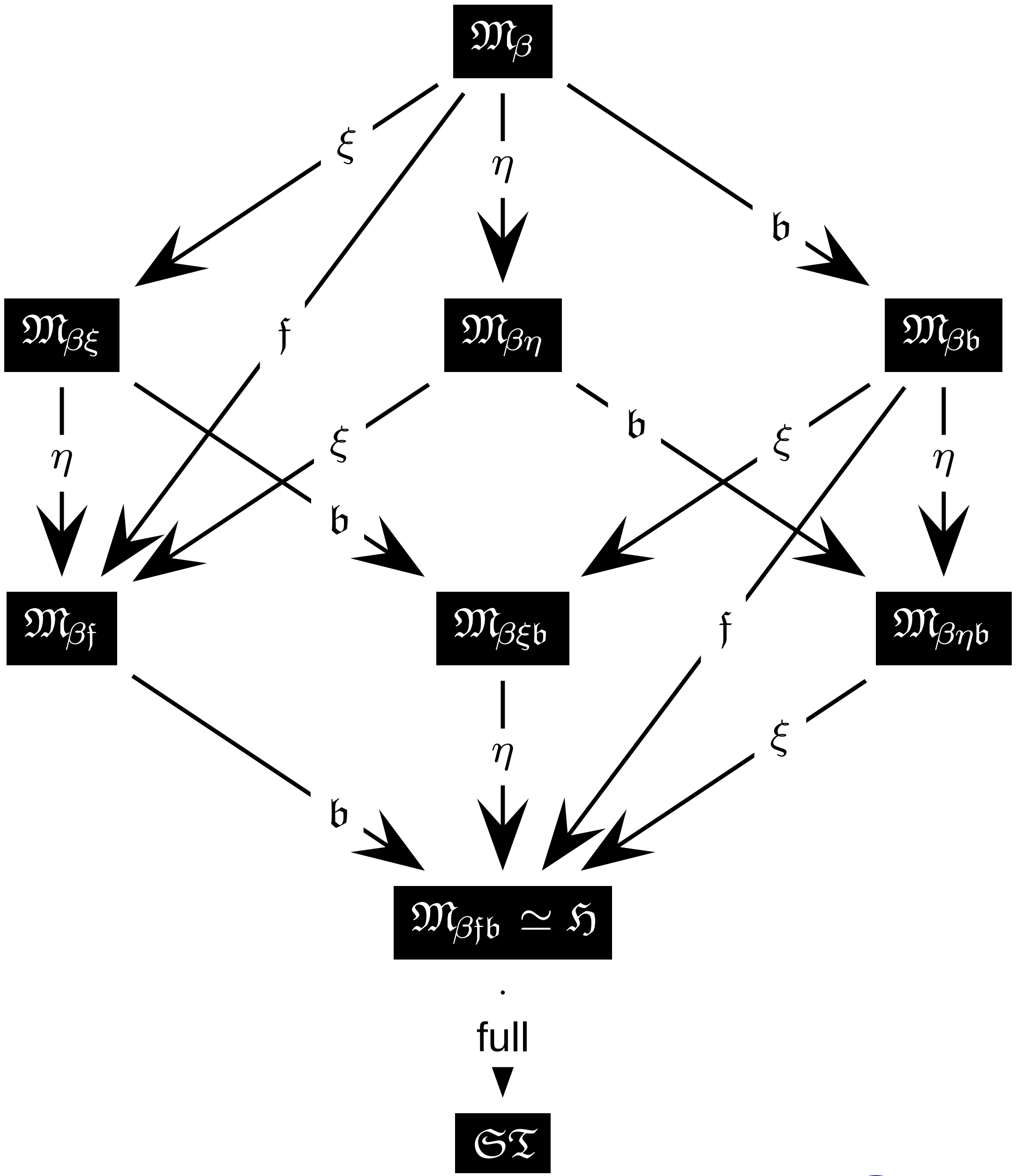
HOL-Problems: Set Comprehension



Set comprehension

- generating set instantiations is a challenge (for automation)
- [Benzm.BrownKohlhase-Draft-05] set instantiations can be used to simulate cut-rule if one of the following axioms is given: comprehension, induction, extensionality, choice, description
- possible set instantiations depend on logical constants in $\mathcal{S}$

HOL-Problems: Set Comprehension



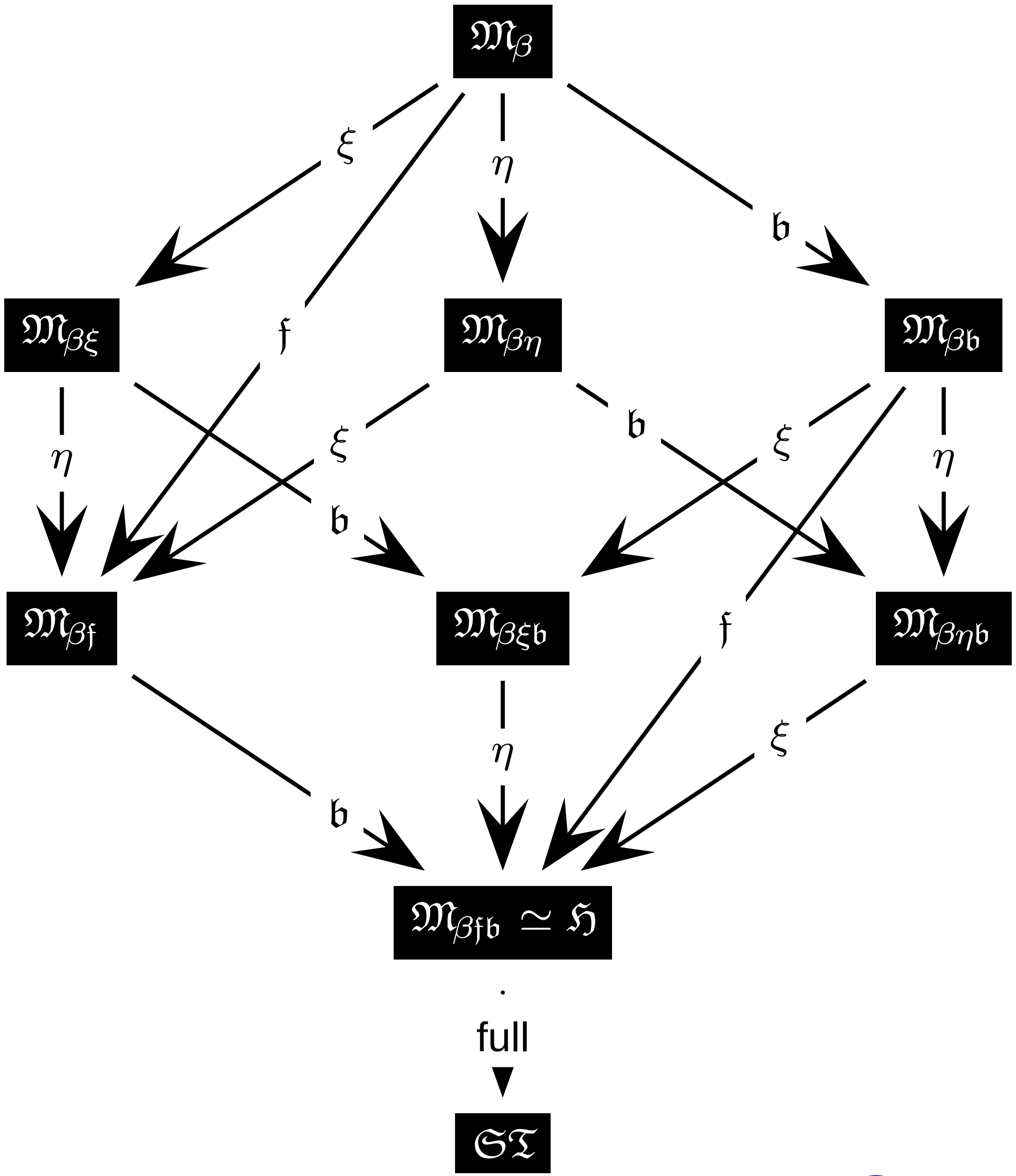
Set comprehension

- generating set instantiations is a challenge (for automation)
- [Benzm.BrownKohlhase-Draft-05] set instantiations can be used to simulate cut-rule if one of the following axioms is given: comprehension, induction, extensionality, choice, description
- possible set instantiations depend on logical constants in $\mathcal{S}$

In the remainder

- signature $\mathcal{S}$ varying
- no property $\mathfrak{q}$ assumed

HOL-Problems: Set Comprehension



Set comprehension

- generating set instantiations is a challenge (for automation)
- [Benzm.BrownKohlhase-Draft-05] set instantiations can be used to simulate cut-rule if one of the following axioms is given: comprehension, induction, extensionality, choice, description
- possible set instantiations depend on logical constants in $\mathcal{S}$

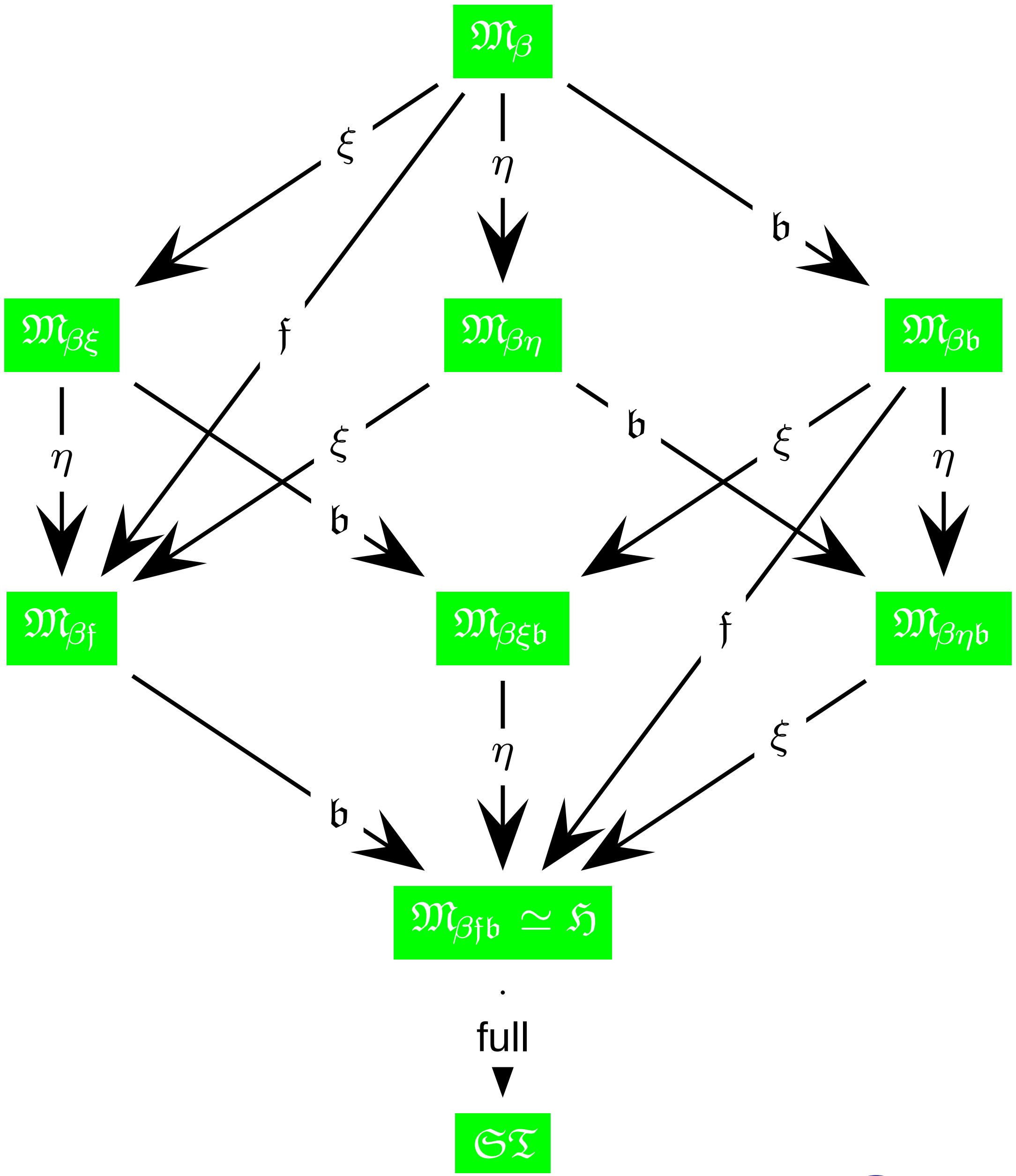
In the remainder

- signature $\mathcal{S}$ varying
- no property $\mathfrak{q}$ assumed

External vs. internal logical constants

- if $\neg \notin \mathcal{S}$ formula $(\neg\mathbf{A})$ still makes sense:
  $\neg$ refers to external symbol and $\mathcal{M} \models \neg\mathbf{A}$ means $\mathcal{M} \not\models \mathbf{A}$

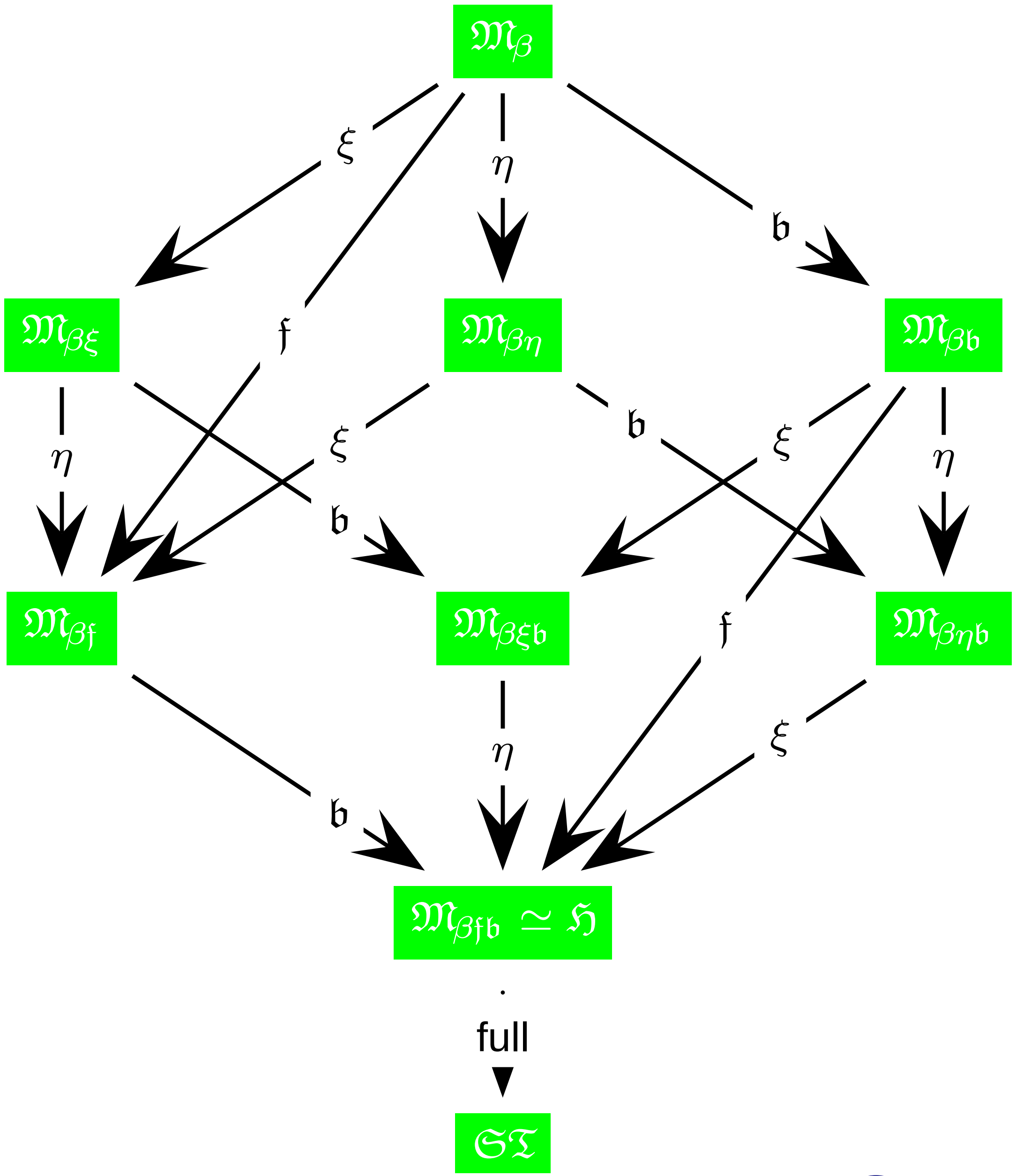
HOL-Problems: Set Comprehension



Set comprehension

- $\exists P.\, P$
  - if $\top \in \mathcal{S}$ or
  - if $\neg \in \mathcal{S}$

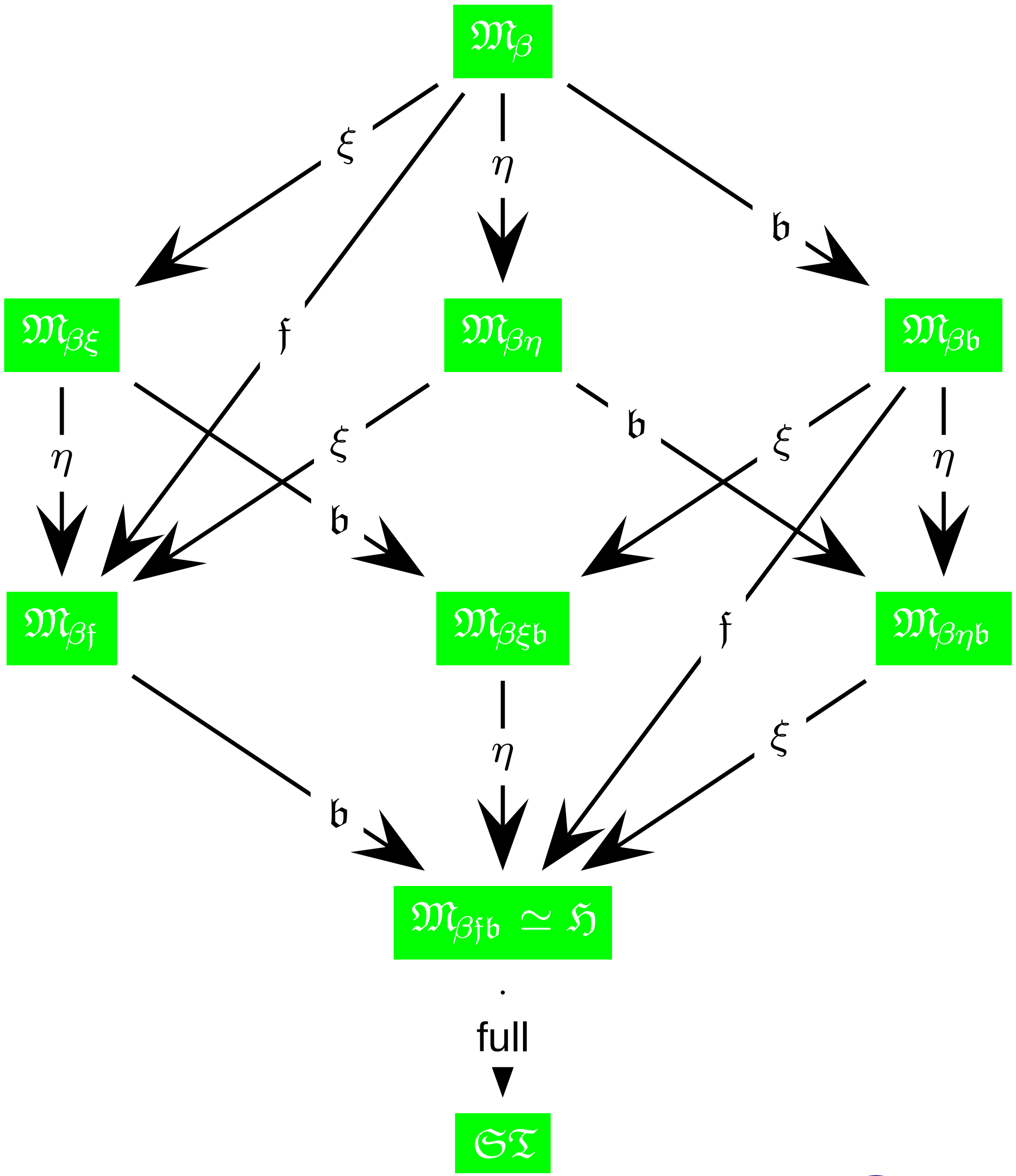
HOL-Problems: Set Comprehension



Set comprehension

- $\exists P.\ P$
  - if $\top \in \mathcal{S}$ or
  - if $\neg \in \mathcal{S}$
- $\exists N_{oo} \forall P_o.\ NP \Leftrightarrow \neg P$
  - if $\neg \in \mathcal{S}$ or
  - if $\{\bot, \Rightarrow\} \subseteq \mathcal{S}$ or $\{\bot, \Leftrightarrow\} \subseteq \mathcal{S}$

HOL-Problems: Set Comprehension



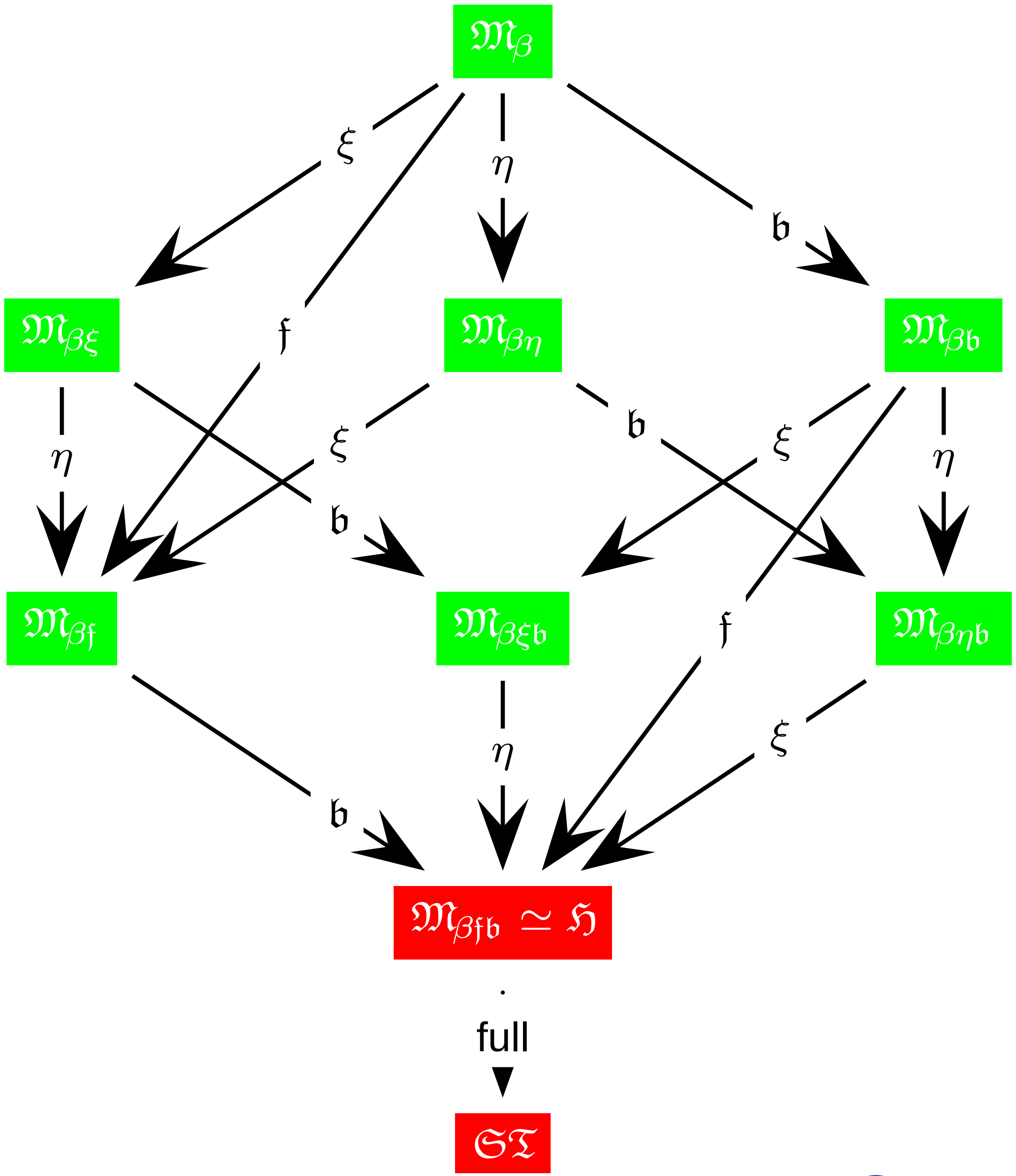
Set comprehension

- $\exists P. P$
  - if $\top \in \mathcal{S}$ or
  - if $\neg \in \mathcal{S}$
- $\exists N_{oo} \forall P_o. NP \Leftrightarrow \neg P$
  - if $\neg \in \mathcal{S}$ or
  - if $\{\bot, \Rightarrow\} \subseteq \mathcal{S}$ or $\{\bot, \Leftrightarrow\} \subseteq \mathcal{S}$

Other examples from [Brown-PhD-04]

- Surjective Cantor Theorem
- Injective Cantor Theorem

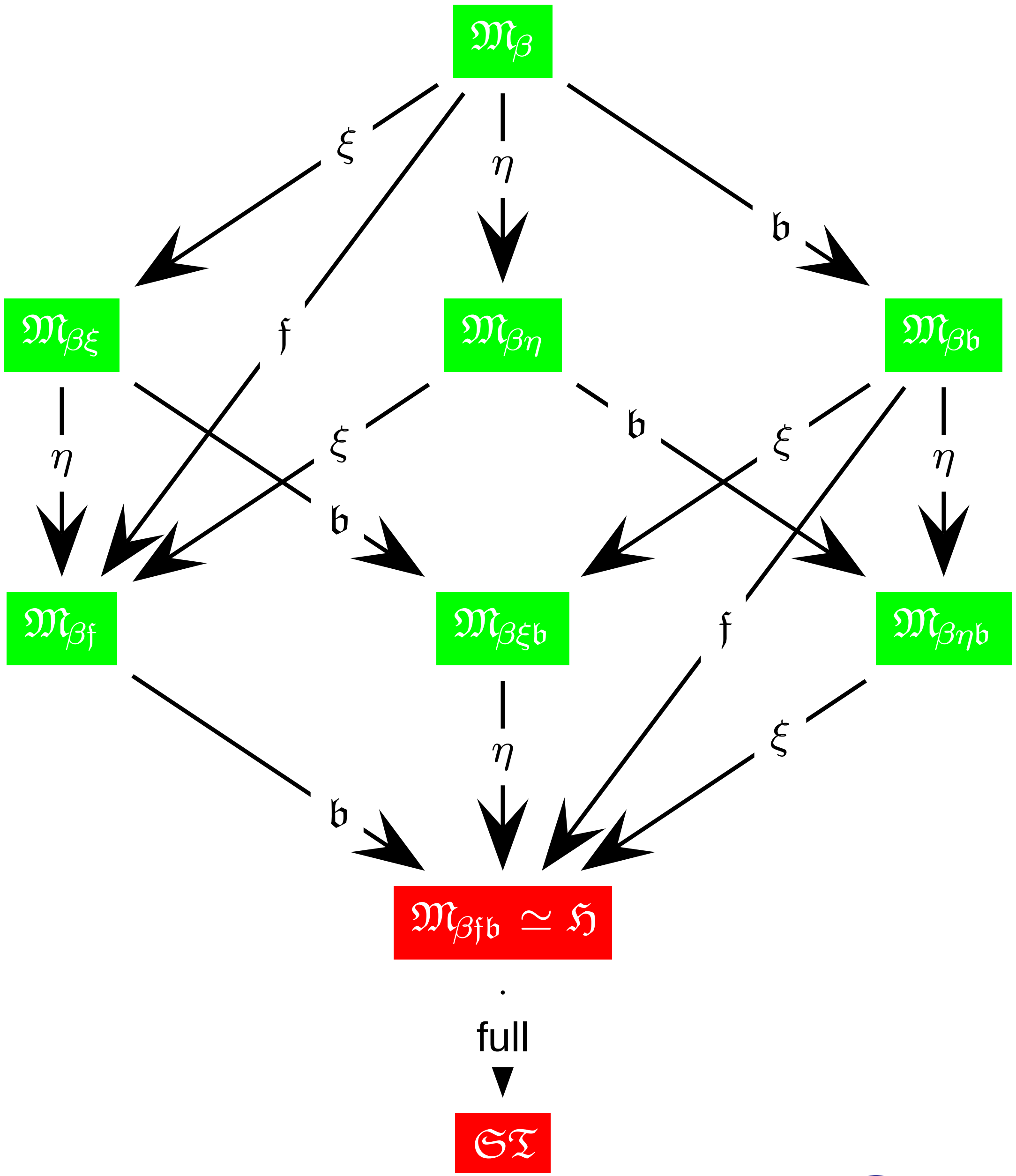
HOL-Problems: Set Comprehension



Set comprehension

- $\exists P.\, P$
  - if $\top \notin \mathcal{S}$ and $\neg \notin \mathcal{S}$

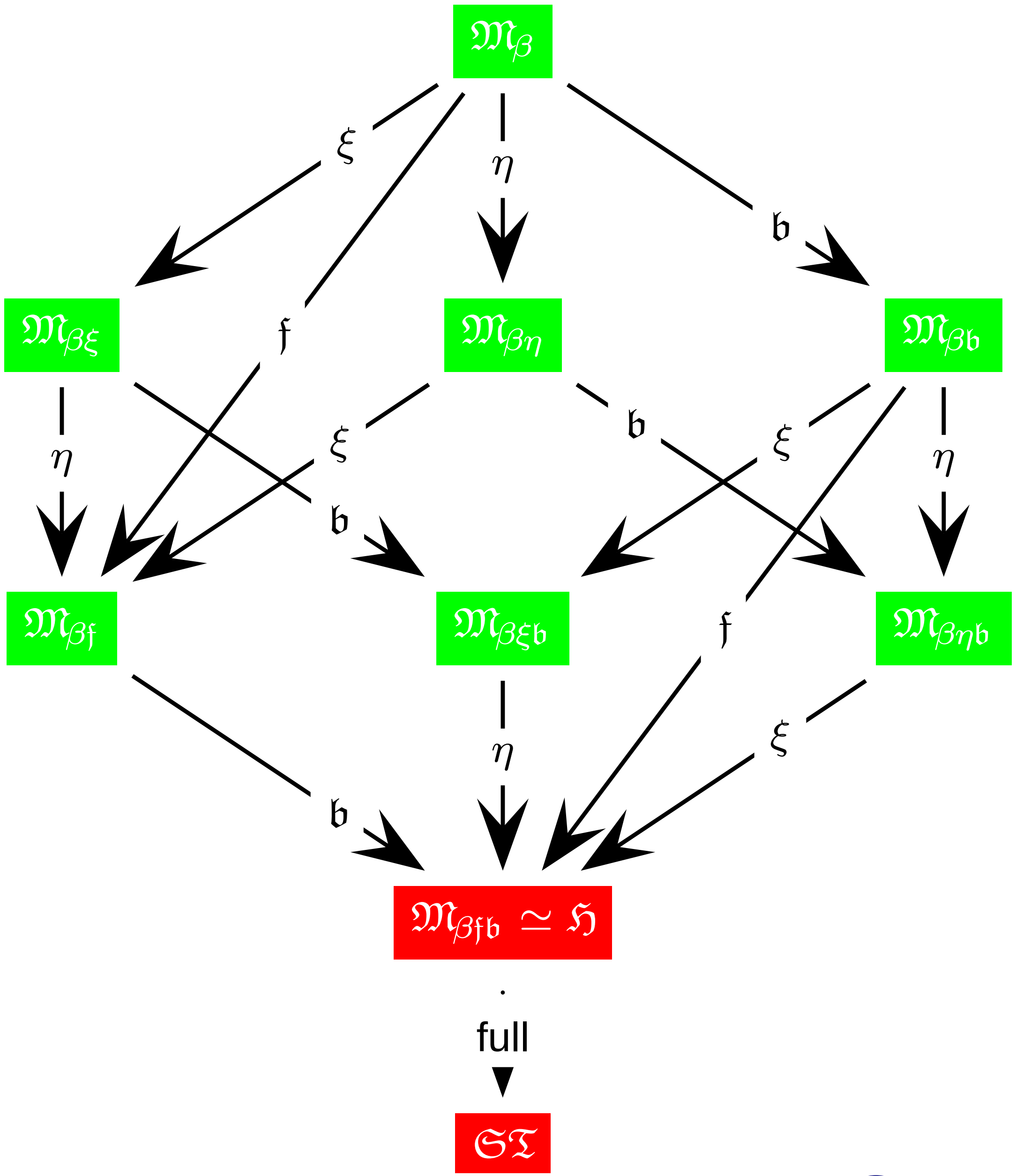
HOL-Problems: Set Comprehension



Set comprehension

- $\exists P_{\bullet}\ P$
  - if $\top \notin \mathcal{S}$ and $\neg \notin \mathcal{S}$
- $\exists N_{oo} \forall P_{o \bullet}\ NP \Leftrightarrow \neg P$
  - if $\neg \notin \mathcal{S}$ and $\{\bot, \Rightarrow\} \not\subseteq \mathcal{S}$ or $\{\bot, \Leftrightarrow\} \not\subseteq \mathcal{S}$

HOL-Problems: Set Comprehension



Set comprehension

- $\exists P.\ P$
  - if $\top \notin \mathcal{S}$ and $\neg \notin \mathcal{S}$
- $\exists N_{oo} \forall P_o.\ NP \Leftrightarrow \neg P$
  - if $\neg \notin \mathcal{S}$ and $\{\bot, \Rightarrow\} \not\subseteq \mathcal{S}$ or $\{\bot, \Leftrightarrow\} \not\subseteq \mathcal{S}$

Other examples from [Brown-PhD-04]

- Surjective Cantor Theorem
- Injective Cantor Theorem

Conclusion

- Presented simple examples

Conclusion

- Presented simple examples
  - highlight some semantical or technical point

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode
  - relevant for automated and interactive TP

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode
  - relevant for automated and interactive TP
  - shall precede formal soundness / completeness analysis

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode
  - relevant for automated and interactive TP
  - shall precede formal soundness / completeness analysis
- Further examples needed

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode
  - relevant for automated and interactive TP
  - shall precede formal soundness / completeness analysis
- Further examples needed
  - to illustrate properties of different forms of equality

UNIVERSITÄT DES SAARLANDES

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode
  - relevant for automated and interactive TP
  - shall precede formal soundness / completeness analysis
- Further examples needed
  - to illustrate properties of different forms of equality
  - mediocre and challenge ones to built up a HOL TPTP

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode
  - relevant for automated and interactive TP
  - shall precede formal soundness / completeness analysis
- Further examples needed
  - to illustrate properties of different forms of equality
  - mediocre and challenge ones to built up a HOL TPTP
- Outlook

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode
  - relevant for automated and interactive TP
  - shall precede formal soundness / completeness analysis
- Further examples needed
  - to illustrate properties of different forms of equality
  - mediocre and challenge ones to built up a HOL TPTP
- Outlook
  - Refine model classes for: description, choice, etc.

Conclusion

- Presented simple examples
  - highlight some semantical or technical point
  - easy to understand / encode
  - relevant for automated and interactive TP
  - shall precede formal soundness / completeness analysis
- Further examples needed
  - to illustrate properties of different forms of equality
  - mediocre and challenge ones to built up a HOL TPTP
- Outlook
  - Refine model classes for: description, choice, etc.
  - See you at ESHOL-05 in Jamaica:
    `http://www.ags.uni-sb.de/~chris/ESHOL-05`