Constructing the Object of Bialgebras in a 2-Category with PIE-Limits

David Forsman

Université catholique de Louvain, Louvain-la-Neuve, Belgium
david.forsman@uclouvain.be

TACL 2026 | Kraków, Poland
July 27, 2026

Supported by a Doctoral Fellowship from the F.R.S.-FNRS (Belgium)
ArXiv: A Syntactic Approach to Ulmer's Bialgebras

Outline of the Talk

1. Background & Motivation

- PIE-limits: Products, Inserters, and Equifiers.
- Ulmer's semantic definition of bialgebras and the missing syntax.

Outline of the Talk

1. Background & Motivation

- PIE-limits: Products, Inserters, and Equifiers.
- Ulmer's semantic definition of bialgebras and the missing syntax.

2. Formal Syntax & 2-Categorical Semantics

- Two-level grammar: 2-signatures Σ, relative 1-signatures σ, and theories T.
- Constructing the object of operations M^σ and bialgebras M^T via PIE limits.
- Syntactic and semantic examples: Monoids and Fields.

Outline of the Talk

1. Background & Motivation

- PIE-limits: Products, Inserters, and Equifiers.
- Ulmer's semantic definition of bialgebras and the missing syntax.

2. Formal Syntax & 2-Categorical Semantics

- Two-level grammar: 2-signatures Σ, relative 1-signatures σ, and theories T.
- Constructing the object of operations M^σ and bialgebras M^T via PIE limits.
- Syntactic and semantic examples: Monoids and Fields.

3. The Property Liftings

- **Lax/Oplax/Strong Functors:** The Induced Functor of Algebras Theorem.
- **Limits & Colimits:** Strict creation of right and left Kan extensions.
- **Structures:** Accessibility, Local Presentability, Factorization Systems, Regularity, and Exactness.

Motivation: General Algebraic Structures

The Goal

Provide a **unified syntax and semantics** for 1-dimensional bialgebraic structures (groups, monoids, Hopf algebras, monad algebras) in general categories, and prove general structural property-lifting theorems.

Motivation: General Algebraic Structures

The Goal

Provide a **unified syntax and semantics** for 1-dimensional bialgebraic structures (groups, monoids, Hopf algebras, monad algebras) in general categories, and prove general structural property-lifting theorems.

The 2-Categorical Setting

While we give explicit definitions in **CAT** for geometric intuition, all constructions and theorems are formulated strictly within an arbitrary **PIE-complete** 2-**category** $\mathcal{K}$.

Motivation: General Algebraic Structures

The Goal

Provide a **unified syntax and semantics** for 1-dimensional bialgebraic structures (groups, monoids, Hopf algebras, monad algebras) in general categories, and prove general structural property-lifting theorems.

The 2-Categorical Setting

While we give explicit definitions in **CAT** for geometric intuition, all constructions and theorems are formulated strictly within an arbitrary **PIE-complete** 2-**category** $\mathcal{K}$.

Building Blocks

The cornerstone of our construction relies on the following 2-categorical limits:

Products · **Inserters** · **Equifiers** (PIE Limits)

PIE Limits I: Products

PIE Limits I: Products

Product Category $\prod_{i \in I} \mathcal{C}_i$

For a family of categories $(\mathcal{C}_i)_{i \in I}$:

- **Objects:** Tuples $(c_i)_{i \in I}$ with c_i and object of $\mathcal{C}_i$ for each $i \in I$.
- **Morphisms:** Tuples $(f_i)_i \colon (c_i)_i \to (d_i)_i$ where $f_i \colon c_i \to d_i$ is a morphism in $\mathcal{C}_i$ for each $i \in I$.

PIE Limits II: Inserters

Let $F, G\colon \mathcal{C} \to \mathcal{D}$ be parallel functors.

PIE Limits II: Inserters

Let $F, G\colon \mathcal{C} \to \mathcal{D}$ be parallel functors.

Inserter Category $\mathbf{Ins}(F, G)$

- **Objects:** Pairs $(c, Fc \xrightarrow{\alpha} Gc)$ where c is an object in $\mathcal{C}$ and α is a morphism in $\mathcal{D}$.

PIE Limits II: Inserters

Let $F, G \colon \mathcal{C} \to \mathcal{D}$ be parallel functors.

Inserter Category $\mathbf{Ins}(F, G)$

- **Objects:** Pairs $(c, Fc \xrightarrow{\alpha} Gc)$ where c is an object in $\mathcal{C}$ and α is a morphism in $\mathcal{D}$.
- **Morphisms:** $f \colon (c, \alpha) \to (d, \beta)$ given by a morphism $f \colon c \to d$ in $\mathcal{C}$ making the diagram

$$\begin{tikzcd} Fc \arrow[r, "\alpha"] \arrow[d, "Ff"'] & Gc \arrow[d, "Gf"] \\ Fd \arrow[r, "\beta"'] & Gd \end{tikzcd}$$

commute.

PIE-limits II: Inserters

Universal Diagram

The forgetful functor $U\colon \mathbf{Ins}(F,G) \to \mathcal{C}$ and universal transformation $\phi\colon F \circ U \Rightarrow G \circ U$ form the universal inserter square:

$$\begin{tikzcd}
\mathbf{Ins}(F,G) \arrow[r, "U"] \arrow[d, "U"'] & \mathcal{C} \arrow[d, "G"] \\
\mathcal{C} \arrow[r, "F"'] \arrow[ur, Rightarrow, "\phi" description] & \mathcal{D}
\end{tikzcd}$$

Here $\phi_{(c,\alpha)} := \alpha\colon Fc \to Gc$ for each object $(c, \alpha\colon Fc \to Gc)$ in $\mathbf{Ins}(F,G)$.

PIE Limits III: Equifiers

PIE Limits III: Equifiers

Let $\eta, \theta \colon F \Rightarrow G \colon \mathcal{C} \to \mathcal{D}$ be parallel natural transformations.

Equifier Category $\mathbf{Eq}(\eta, \theta)$

PIE Limits III: Equifiers

Let $\eta, \theta \colon F \Rightarrow G \colon \mathcal{C} \to \mathcal{D}$ be parallel natural transformations.

Equifier Category $\mathbf{Eq}(\eta, \theta)$

- **Definition:** The **full subcategory** of $\mathcal{C}$ consisting of those objects c in $\mathcal{C}$ for which all components agree:

$$\eta_c = \theta_c.$$

PIE Limits III: Equifiers

Let $\eta, \theta \colon F \Rightarrow G \colon \mathcal{C} \to \mathcal{D}$ be parallel natural transformations.

Equifier Category $\mathbf{Eq}(\eta, \theta)$

- **Definition:** The **full subcategory** of $\mathcal{C}$ consisting of those objects c in $\mathcal{C}$ for which all components agree:

$$\eta_c = \theta_c.$$

- **Universal Inclusion:** Fully faithful inclusion functor $E \colon \mathbf{Eq}(\eta, \theta) \hookrightarrow \mathcal{C}$ such that $\eta * E = \theta * E$.

PIE Limits III: Equifiers

Let $\eta, \theta \colon F \Rightarrow G \colon \mathcal{C} \to \mathcal{D}$ be parallel natural transformations.

Equifier Category $\mathbf{Eq}(\eta, \theta)$

- **Definition:** The **full subcategory** of $\mathcal{C}$ consisting of those objects c in $\mathcal{C}$ for which all components agree:

$$\eta_c = \theta_c.$$

- **Universal Inclusion:** Fully faithful inclusion functor $E \colon \mathbf{Eq}(\eta, \theta) \hookrightarrow \mathcal{C}$ such that $\eta * E = \theta * E$.

Universal Equifier Diagram

$$\begin{tikzcd}
\mathbf{Eq}(\eta, \theta) \arrow[r, "E"] & \mathcal{C} \arrow[r, bend left=50, "F"{name=U}] \arrow[r, bend right=50, "G"'{name=D}] & \mathcal{D}
\arrow[from=U, to=D, Rightarrow, shift right=2, "\theta"']
\arrow[from=U, to=D, Rightarrow, shift left=2, "\eta"]
\end{tikzcd}$$

Ulmer's Semantic Bialgebras (1977)

Ulmer's Semantic Bialgebras (1977)

Using PIE limits, Ulmer defined a general category of bialgebras:

Ulmer's Two-Step Construction

Ulmer's Semantic Bialgebras (1977)

Using PIE limits, Ulmer defined a general category of bialgebras:

Ulmer's Two-Step Construction

Given a category $\underline{A}$, support functors $F_d, F_c \colon \underline{A} \to X$, and parallel pair of natural transformations $\alpha, \beta \colon G_d V \Rightarrow G_c V \colon \mathbf{Ins}(F_d, F_c) \xrightarrow{V} \underline{A} \xrightarrow{G_d, G_c} Y$:

Ulmer's Semantic Bialgebras (1977)

Using PIE limits, Ulmer defined a general category of bialgebras:

Ulmer's Two-Step Construction

Given a category $\underline{A}$, support functors $F_d, F_c \colon \underline{A} \to X$, and parallel pair of natural transformations $\alpha, \beta \colon G_d V \Rightarrow G_c V \colon \mathbf{Ins}(F_d, F_c) \xrightarrow{V} \underline{A} \xrightarrow{G_d, G_c} Y$:

1. **Pre-bialgebras (Operations): P-Bialg**$(\underline{A}) \coloneqq \mathbf{Ins}(F_d, F_c)$

Ulmer's Semantic Bialgebras (1977)

Using PIE limits, Ulmer defined a general category of bialgebras:

Ulmer's Two-Step Construction

Given a category $\underline{A}$, support functors $F_d, F_c \colon \underline{A} \to X$, and parallel pair of natural transformations $\alpha, \beta \colon G_d V \Rightarrow G_c V \colon \mathbf{Ins}(F_d, F_c) \xrightarrow{V} \underline{A} \xrightarrow{G_d, G_c} Y$:

1. **Pre-bialgebras (Operations):** $\mathbf{P\text{-}Bialg}(\underline{A}) \coloneqq \mathbf{Ins}(F_d, F_c)$
2. **Bialgebras (Equations):** $\mathbf{Bialg}(\underline{A}) \coloneqq \mathbf{Eq}(\alpha, \beta)$

Ulmer's Semantic Bialgebras (1977)

Using PIE limits, Ulmer defined a general category of bialgebras:

Ulmer's Two-Step Construction

Given a category $\underline{A}$, support functors $F_d, F_c \colon \underline{A} \to X$, and parallel pair of natural transformations $\alpha, \beta \colon G_d V \Rightarrow G_c V \colon \mathbf{Ins}(F_d, F_c) \xrightarrow{V} \underline{A} \xrightarrow{G_d, G_c} Y$:

1. **Pre-bialgebras (Operations):** $\mathbf{P\text{-}Bialg}(\underline{A}) \coloneqq \mathbf{Ins}(F_d, F_c)$
2. **Bialgebras (Equations):** $\mathbf{Bialg}(\underline{A}) \coloneqq \mathbf{Eq}(\alpha, \beta)$

The Limitations of Ulmer's Semantics

Ulmer's Semantic Bialgebras (1977)

Using PIE limits, Ulmer defined a general category of bialgebras:

Ulmer's Two-Step Construction

Given a category $\underline{A}$, support functors $F_d, F_c \colon \underline{A} \to X$, and parallel pair of natural transformations $\alpha, \beta \colon G_d V \Rightarrow G_c V \colon \mathbf{Ins}(F_d, F_c) \xrightarrow{V} \underline{A} \xrightarrow{G_d, G_c} Y$:

1. **Pre-bialgebras (Operations):** $\textbf{P-Bialg}(\underline{A}) \coloneqq \mathbf{Ins}(F_d, F_c)$
2. **Bialgebras (Equations):** $\mathbf{Bialg}(\underline{A}) \coloneqq \mathbf{Eq}(\alpha, \beta)$

The Limitations of Ulmer's Semantics

Ulmer's definition is **semantic**:

- Support functors (e.g., $x \mapsto x \otimes x$) and transformations must be **specified by hand**.

Ulmer's Semantic Bialgebras (1977)

Using PIE limits, Ulmer defined a general category of bialgebras:

Ulmer's Two-Step Construction

Given a category $\underline{A}$, support functors $F_d, F_c \colon \underline{A} \to X$, and parallel pair of natural transformations $\alpha, \beta \colon G_d V \Rightarrow G_c V \colon \mathbf{Ins}(F_d, F_c) \xrightarrow{V} \underline{A} \xrightarrow{G_d, G_c} Y$:

1. **Pre-bialgebras (Operations): P-Bialg**$(\underline{A}) \coloneqq \mathbf{Ins}(F_d, F_c)$
2. **Bialgebras (Equations): Bialg**$(\underline{A}) \coloneqq \mathbf{Eq}(\alpha, \beta)$

The Limitations of Ulmer's Semantics

Ulmer's definition is **semantic**:

- Support functors (e.g., $x \mapsto x \otimes x$) and transformations must be **specified by hand**.
- No formal mechanism explains how background structures (like monoidal tensor products) generate these support functors.

Ulmer's Semantic Bialgebras (1977)

Using PIE limits, Ulmer defined a general category of bialgebras:

Ulmer's Two-Step Construction

Given a category $\underline{A}$, support functors $F_d, F_c \colon \underline{A} \to X$, and parallel pair of natural transformations $\alpha, \beta \colon G_d V \Rightarrow G_c V \colon \mathbf{Ins}(F_d, F_c) \xrightarrow{V} \underline{A} \xrightarrow{G_d, G_c} Y$:

1. **Pre-bialgebras (Operations): P-Bialg**$(\underline{A}) \coloneqq \mathbf{Ins}(F_d, F_c)$
2. **Bialgebras (Equations): Bialg**$(\underline{A}) \coloneqq \mathbf{Eq}(\alpha, \beta)$

The Limitations of Ulmer's Semantics

Ulmer's definition is **semantic**:

- Support functors (e.g., $x \mapsto x \otimes x$) and transformations must be **specified by hand**.
- No formal mechanism explains how background structures (like monoidal tensor products) generate these support functors.

Our Solution: A formal **syntactic grammar** that allows an automatic building of these support functors and parallel pairs of natural transformations.

Syntax I: Signature Pairs (Σ, σ)

We formalize a two-level syntactic grammar:

Syntax I: Signature Pairs (Σ, σ)

We formalize a two-level syntactic grammar:

1. Background 2-Signature $\Sigma = (C, \mathcal{F}, \mathcal{T})$

Fixes the background categorical structure (monoidal, cartesian, monadic, etc.):

- **Category symbols:** A set $C = \{c, d, \dots\}$.

Syntax I: Signature Pairs (Σ, σ)

We formalize a two-level syntactic grammar:

1. Background 2-Signature $\Sigma = (C, \mathcal{F}, \mathcal{T})$

Fixes the background categorical structure (monoidal, cartesian, monadic, etc.):

- **Category symbols:** A set $C = \{c, d, \ldots\}$.
- **Functor symbols:** A typed set $\mathcal{F} \to C^* \times C$ of functor symbols $B\colon c_1 \cdots c_n \to d \in \mathcal{F}$. Generates the typed set $\mathrm{Term}_V^\Sigma \to C$, where $V \to C$ is a typed set of variable symbols.

Syntax I: Signature Pairs (Σ, σ)

We formalize a two-level syntactic grammar:

1. Background 2-Signature $\Sigma = (C, \mathcal{F}, \mathcal{T})$

Fixes the background categorical structure (monoidal, cartesian, monadic, etc.):

- **Category symbols:** A set $C = \{c, d, \dots\}$.
- **Functor symbols:** A typed set $\mathcal{F} \to C^* \times C$ of functor symbols $B\colon c_1 \cdots c_n \to d \in \mathcal{F}$. Generates the typed set $\mathrm{Term}_V^\Sigma \to C$, where $V \to C$ is a typed set of variable symbols.
- **Transformation symbols:** $\alpha^{\bar{x}}\colon t_1 \to t_2 \in \mathcal{T}$ in context $\bar{x}$ with $t_1, t_2 \in \mathrm{Term}_V^\Sigma$ of the same type.

Syntax I: Signature Pairs (Σ, σ)

We formalize a two-level syntactic grammar:

1. Background 2-Signature $\Sigma = (C, \mathcal{F}, \mathcal{T})$

Fixes the background categorical structure (monoidal, cartesian, monadic, etc.):

- **Category symbols:** A set $C = \{c, d, \ldots\}$.
- **Functor symbols:** A typed set $\mathcal{F} \to C^* \times C$ of functor symbols $B\colon c_1 \cdots c_n \to d \in \mathcal{F}$. Generates the typed set $\mathrm{Term}_V^\Sigma \to C$, where $V \to C$ is a typed set of variable symbols.
- **Transformation symbols:** $\alpha^{\bar{x}}\colon t_1 \to t_2 \in \mathcal{T}$ in context $\bar{x}$ with $t_1, t_2 \in \mathrm{Term}_V^\Sigma$ of the same type.

2. Relative 1-Signature $\sigma = (S, F)$

Syntax I: Signature Pairs (Σ, σ)

We formalize a two-level syntactic grammar:

1. Background 2-Signature $\Sigma = (C, \mathcal{F}, \mathcal{T})$

Fixes the background categorical structure (monoidal, cartesian, monadic, etc.):

- **Category symbols:** A set $C = \{c, d, \dots\}$.
- **Functor symbols:** A typed set $\mathcal{F} \to C^* \times C$ of functor symbols $B \colon c_1 \cdots c_n \to d \in \mathcal{F}$. Generates the typed set $\mathrm{Term}_V^\Sigma \to C$, where $V \to C$ is a typed set of variable symbols.
- **Transformation symbols:** $\alpha^{\bar{x}} \colon t_1 \to t_2 \in \mathcal{T}$ in context $\bar{x}$ with $t_1, t_2 \in \mathrm{Term}_V^\Sigma$ of the same type.

2. Relative 1-Signature $\sigma = (S, F)$

Defines the internal algebraic operations over Σ:

- **Sorts:** A typed set $S \to C$ of sorts.

Syntax I: Signature Pairs (Σ, σ)

We formalize a two-level syntactic grammar:

1. Background 2-Signature $\Sigma = (C, \mathcal{F}, \mathcal{T})$

Fixes the background categorical structure (monoidal, cartesian, monadic, etc.):

- **Category symbols:** A set $C = \{c, d, \ldots\}$.
- **Functor symbols:** A typed set $\mathcal{F} \to C^* \times C$ of functor symbols $B \colon c_1 \cdots c_n \to d \in \mathcal{F}$. Generates the typed set $\mathrm{Term}_V^\Sigma \to C$, where $V \to C$ is a typed set of variable symbols.
- **Transformation symbols:** $\alpha^{\bar{x}} \colon t_1 \to t_2 \in \mathcal{T}$ in context $\bar{x}$ with $t_1, t_2 \in \mathrm{Term}_V^\Sigma$ of the same type.

2. Relative 1-Signature $\sigma = (S, F)$

Defines the internal algebraic operations over Σ:

- **Sorts:** A typed set $S \to C$ of sorts.
- **Function symbols:** As typed set $F \to \mathrm{Term}_S^\Sigma \times \mathrm{Term}_S^\Sigma$ of function symbols $f \colon t_1 \to t_2$, where the **function arity** term t_1 and the **function coarity term** t_2 are typed the same.

Syntax II: Term Graph

Fix a signature pair (Σ, σ).

Syntax II: Term Graph

Fix a signature pair (Σ, σ).

The Term Graph G_σ

We define the vertices of the multigraph G_σ as the set Term_S^Σ of terms and the arrows are constructed recursively as follows:

Syntax II: Term Graph

Fix a signature pair (Σ, σ).

The Term Graph G_σ

We define the vertices of the multigraph G_σ as the set Term_S^Σ of terms and the arrows are constructed recursively as follows:

1. Each function symbol $f : t_1 \to t_2$ forms an arrow $t_1 \to t_2$ in G_σ.

Syntax II: Term Graph

Fix a signature pair (Σ, σ).

The Term Graph G_σ

We define the vertices of the multigraph G_σ as the set Term_S^Σ of terms and the arrows are constructed recursively as follows:

1. Each function symbol $f \colon t_1 \to t_2$ forms an arrow $t_1 \to t_2$ in G_σ.
2. For a transformation symbol $\alpha^{x_1 \cdots x_n} \colon t_1 \to t_2$, we set

$$\alpha_{t'_1, \ldots, t'_n} \colon t_1^{\bar{x}}(t'_1, \ldots, t'_n) \to t_2^{\bar{x}}(t'_1, \ldots, t'_n)$$

   as an arrow in G_σ for any sequence $t_1 \colon c_{x_1}, \ldots, t_n \colon c_{x_n} \in \mathrm{Term}_S^\Sigma$.

Syntax II: Term Graph

Fix a signature pair (Σ, σ).

The Term Graph G_σ

We define the vertices of the multigraph G_σ as the set Term_S^Σ of terms and the arrows are constructed recursively as follows:

1. Each function symbol $f\colon t_1 \to t_2$ forms an arrow $t_1 \to t_2$ in G_σ.
2. For a transformation symbol $\alpha^{x_1 \cdots x_n}\colon t_1 \to t_2$, we set

$$\alpha_{t'_1,\ldots,t'_n}\colon t_1^{\bar{x}}(t'_1,\ldots,t'_n) \to t_2^{\bar{x}}(t'_1,\ldots,t'_n)$$

   as an arrow in G_σ for any sequence $t_1\colon c_{x_1},\ldots,t_n\colon c_{x_n} \in \mathrm{Term}_S^\Sigma$.
3. For any Σ-functor symbol $B\colon c_0 \cdots c_n \to d$ and an arrow $p\colon t_i \to t'_i$ in G_σ we set

$$B(t_0,\ldots,t_{i-1},p,t_{i+1},\ldots,t_n)\colon B(t_1,\ldots,t_n) \to B(t_0,\ldots,t_{i-1},t'_i,t_{i+1},\ldots,t_n)$$

   as an arrow of G_σ.

Syntax III: Bialgebraic Theories

Syntax III: Bialgebraic Theories

Bialgebraic σ-Theory T

A bialgebraic σ-**theory** T is a set of parallel paths $(p, q\colon t_1 \to t_2\colon c)$ in G_σ (commutative diagrams).

Syntax III: Bialgebraic Theories

Bialgebraic σ-Theory T

A bialgebraic σ-**theory** T is a set of parallel paths $(p, q \colon t_1 \to t_2 \colon c)$ in G_σ (commutative diagrams).

- T is **algebraic** if all **equational coarity terms** t_2 and functional coarity terms are sorts ($t_2 \in S$).

Syntax III: Bialgebraic Theories

Bialgebraic σ-Theory T

A bialgebraic σ-**theory** T is a set of parallel paths $(p, q\colon t_1 \to t_2\colon c)$ in G_σ (commutative diagrams).

- T is **algebraic** if all **equational coarity terms** t_2 and functional coarity terms are sorts $(t_2 \in S)$.
- T is **coalgebraic** if all **equational arity terms** t_1 and functional arity terms are sorts.

Syntactic Example: The Theory of Monoids T_{Mon}

Signature Pair $(\Sigma_{\mathsf{Mon}}, \sigma_{\mathsf{Mon}})$ and the theory T_{Mon} of monoids.

- **Background** 2-**Signature** Σ_{Mon}:

$$c, \quad \Box\colon cc \to c, \quad I\colon () \to c, \quad (x\Box y)\Box z \xrightarrow{\alpha^{xyz}} x\Box(y\Box z), \quad x\Box I \xrightarrow{\rho^{x}} x \xleftarrow{\lambda^{x}} I\Box x$$

Syntactic Example: The Theory of Monoids T_{Mon}

Signature Pair $(\Sigma_{\mathrm{Mon}}, \sigma_{\mathrm{Mon}})$ and the theory T_{Mon} of monoids.

- **Background** 2-**Signature** Σ_{Mon}:

$$c, \quad \square \colon cc \to c, \quad I \colon () \to c, \quad (x \square y) \square z \xrightarrow{\alpha^{xyz}} x \square (y \square z), \quad x \square I \xrightarrow{\rho^{x}} x \xleftarrow{\lambda^{x}} I \square x$$

- **Relative** 1-**Signature** σ_{Mon}:

$$s \colon c, \quad \mu \colon s \square s \to s, \quad u \colon I \to s$$

Syntactic Example: The Theory of Monoids T_{Mon}

Signature Pair $(\Sigma_{\mathrm{Mon}}, \sigma_{\mathrm{Mon}})$ and the theory T_{Mon} of monoids.

- **Background** 2-**Signature** Σ_{Mon}:

$$c, \quad \square \colon cc \to c, \quad I \colon () \to c, \quad (x \square y) \square z \xrightarrow{\alpha^{xyz}} x \square (y \square z), \quad x \square I \xrightarrow{\rho^{x}} x \xleftarrow{\lambda^{x}} I \square x$$

- **Relative** 1-**Signature** σ_{Mon}:

$$s \colon c, \quad \mu \colon s \square s \to s, \quad u \colon I \to s$$

- **Theory of monoids** T_{Mon}:

$$\begin{tikzcd}
(s \square s) \square s \arrow[r, "\alpha_{s,s,s}"] \arrow[d, "\mu \square s"'] & s \square (s \square s) \arrow[r, "s \square \mu"] & s \square s \arrow[d, "\mu"] \\
s \square s \arrow[rr, "\mu"'] & & s
\end{tikzcd}
\qquad
\begin{tikzcd}
s \square I \arrow[r, "s \square u"] \arrow[dr, "\rho_s"'] & s \square s \arrow[d, "\mu"] \\
& s
\end{tikzcd}
\qquad
\begin{tikzcd}
s \square s \arrow[d, "\mu"'] & I \square s \arrow[l, "u \square s"'] \arrow[dl, "\lambda_s"] \\
s &
\end{tikzcd}$$

Syntactic Example: The Theory of Fields T_{Field}

Signature Pair (Σ, σ) and the theory T_{Field} of fields.

- **Background 2-Signature Σ:**

$$c, \quad \Box \colon cc \to c, \quad + \colon cc \to c \quad I \colon () \to c$$

$$x \xrightarrow{\delta^x} x \Box x, \quad x \xleftarrow{p_1^{xy}} x \Box y \xrightarrow{p_2^{xy}} y, \quad x + x \xrightarrow{\nabla^x} x, \quad x \xrightarrow{\iota_1^{xy}} x + y \xleftarrow{\iota_2^{x,y}} y, \quad x \xrightarrow{!^x} I$$

- **Relative 1-Signature σ:**

$$s \colon c, \quad m \colon s \Box s \to s, \quad i \colon s \to s, \quad u \colon I \to s$$

$$m_1, m_2 \colon (s + I) \Box (s + I) \to (s + I), \quad \nu \colon (s + I) \to (s + I)$$

- **Theory of Fields $T_{\textbf{Field}}$** (Equations in G_σ):
  - (s, m, i, u) satisfies the equations of a **commutative group** (for k^*).
  - $(s + I, m_1, \nu, \iota_2, m_2, \iota_1 \circ u)$ satisfies the equations of a **commutative ring** (for $k^* + 1$).
  - $\iota_1 \colon s \to s + I$ is a morphism of magmas from (s, m) to $(s + I, m_2)$.

Σ-Models and Term Interpretation

Let $\Sigma = (\mathcal{C}, \mathcal{F}, \mathcal{T})$ be a 2-signature and $\mathcal{K}$ a 2-category with finite products.

Σ-Models and Term Interpretation

Let $\Sigma = (C, \mathcal{F}, \mathcal{T})$ be a 2-signature and $\mathcal{K}$ a 2-category with finite products.

Definition (Σ-Model M)

A Σ-**model** M in $\mathcal{K}$ assigns:

- An object $M_c \in \mathcal{K}$ for $c \in C$. For a sequence of typed symbols $\bar{x} = x_1 \ldots x_n$, set $M_{\bar{x}} \coloneqq \prod_{i=1}^{n} M_{c_{x_i}}$.
- A 1-cell $M(B)\colon M_{\bar{c}} \to M_d$ for each functor symbol $B\colon \bar{c} \to d \in \mathcal{F}$.
- A 2-cell $M(\alpha)\colon M_{\bar{x}}(t_1) \Rightarrow M_{\bar{x}}(t_2)$ for each transformation symbol $\alpha^{\bar{x}}\colon t_1 \to t_2$.

Σ-Models and Term Interpretation

Let $\Sigma = (\mathcal{C}, \mathcal{F}, \mathcal{T})$ be a 2-signature and $\mathcal{K}$ a 2-category with finite products.

Definition (Σ-Model M)

A **Σ-model** M in $\mathcal{K}$ assigns:

- An object $M_c \in \mathcal{K}$ for $c \in \mathcal{C}$. For a sequence of typed symbols $\bar{x} = x_1 \ldots x_n$, set $M_{\bar{x}} \coloneqq \prod_{i=1}^{n} M_{c_{x_i}}$.
- A 1-cell $M(B) \colon M_{\bar{c}} \to M_d$ for each functor symbol $B \colon \bar{c} \to d \in \mathcal{F}$.
- A 2-cell $M(\alpha) \colon M_{\bar{x}}(t_1) \Rightarrow M_{\bar{x}}(t_2)$ for each transformation symbol $\alpha^{\bar{x}} \colon t_1 \to t_2$.

- **Terms in Context:** For $t \in \mathbf{Term}_V^{\Sigma}$ in context $\bar{x}$, we recursively construct a 1-cell:

$$M_{\bar{x}}(t) \colon M_{\bar{x}} \longrightarrow M_c$$

- **Sort Terms:** For a term $t' \in \mathbf{Term}_S^{\Sigma}$ over sorts S, we set $M_0 \coloneqq \prod_{s \in S} M_{c_s}$ and construct:

$$M(t') \colon M_0 \longrightarrow M_c$$

The Object of Operations M^σ and Bialgebras M^T

Let (Σ, σ) be a signature pair, T a theory, and M a Σ-model in a PIE-complete 2-category $\mathcal{K}$.

The Object of Operations M^σ and Bialgebras M^T

Let (Σ, σ) be a signature pair, T a theory, and M a Σ-model in a PIE-complete 2-category $\mathcal{K}$.

- **1. Object of Operations M^σ:** Interprets σ-function symbols $f : t_1 \to t_2 : c_f$ via the inserter:

$$\begin{tikzcd}
M^\sigma \arrow[r, "u_M"] \arrow[d, "u_M"'] & M_0 \arrow[d, "(M(t_2))_f"] \\
M_0 \arrow[r, "(M(t_1))_f"'] \arrow[ur, Rightarrow, "\phi_M"] & \prod_{f \colon t_1 \to t_2} M_{c_f}
\end{tikzcd}$$

The Object of Operations M^σ and Bialgebras M^T

Let (Σ, σ) be a signature pair, T a theory, and M a Σ-model in a PIE-complete 2-category $\mathcal{K}$.

- **1. Object of Operations M^σ:** Interprets σ-function symbols $f : t_1 \to t_2 : c_f$ via the inserter:

$$\begin{tikzcd}
M^\sigma \arrow[r, "u_M"] \arrow[d, "u_M"'] & M_0 \arrow[d, "(M(t_2))_f"] \\
M_0 \arrow[r, "(M(t_1))_f"'] \arrow[ur, Rightarrow, "\phi_M"] & \prod_{f \colon t_1 \to t_2} M_{c_f}
\end{tikzcd}$$

- **2. Object of T-Bialgebras M^T:**

The Object of Operations M^σ and Bialgebras M^T

Let (Σ, σ) be a signature pair, T a theory, and M a Σ-model in a PIE-complete 2-category $\mathcal{K}$.

- **1. Object of Operations M^σ:** Interprets σ-function symbols $f: t_1 \to t_2 : c_f$ via the inserter:

$$\begin{tikzcd}
M^\sigma \arrow[r, "u_M"] \arrow[d, "u_M"'] & M_0 \arrow[d, "(M(t_2))_f"] \\
M_0 \arrow[ur, Rightarrow, "\phi_M"] \arrow[r, "(M(t_1))_f"'] & \prod_{f:\, t_1 \to t_2} M_{c_f}
\end{tikzcd}$$

- **2. Object of T-Bialgebras M^T:**
  A path $p: t_1 \to t_2 : c$ in G_σ evaluates to a 2-cell
  $\overline{p}_M : M^\sigma(t_1) \Rightarrow M^\sigma(t_2)(= M(t_2) \circ u_M) : M^\sigma \to M_c$.

The Object of Operations M^σ and Bialgebras M^T

Let (Σ, σ) be a signature pair, T a theory, and M a Σ-model in a PIE-complete 2-category $\mathcal{K}$.

- **1. Object of Operations M^σ:** Interprets σ-function symbols $f : t_1 \to t_2 : c_f$ via the inserter:

$$\begin{tikzcd} M^\sigma \arrow[r, "u_M"] \arrow[d, "u_M"'] & M_0 \arrow[d, "(M(t_2))_f"] \\ M_0 \arrow[ur, Rightarrow, "\phi_M"] \arrow[r, "(M(t_1))_f"'] & \prod_{f\colon t_1 \to t_2} M_{c_f} \end{tikzcd}$$

- **2. Object of T-Bialgebras M^T:**
A path $p\colon t_1 \to t_2 \colon c$ in G_σ evaluates to a 2-cell $\overline{p}_M \colon M^\sigma(t_1) \Rightarrow M^\sigma(t_2)(= M(t_2) \circ u_M) \colon M^\sigma \to M_c$. M^T enforces equations $(p, q \colon t_1 \to t_2 \colon c) \in T$ via the equifier:

$$\begin{tikzcd} M^T \arrow[r, "e_M"] & M^\sigma \arrow[r, bend left, "(M^\sigma(t_1))"] \arrow[r, bend right, "(M^\sigma(t_2))"'] \arrow[r, phantom, "(\overline{p}_M)\Downarrow \quad \Downarrow(\overline{q}_M)"] & \prod_{(p,q)\in T} M_c \end{tikzcd}$$

Example: The Theory of Monoids

Consider a monoidal category $M = (\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ as a Σ_{Mon}-model in **CAT**.

Example: The Theory of Monoids

Consider a monoidal category $M = (\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ as a Σ_{Mon}-model in **CAT**.

1. Operations Object M^σ (Pointed Magmas)

For $\sigma_{\mathsf{Mon}} = (s\colon c, \mu\colon s\square s \to s, u\colon I \to s)$, the arity/coarity terms yield support functors:

$$M(s\square s)(x) = x \otimes x, \quad M(I)(x) = I, \quad M(s)(x) = x.$$

Example: The Theory of Monoids

Consider a monoidal category $M = (\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ as a Σ_{Mon}-model in **CAT**.

1. Operations Object M^σ (Pointed Magmas)

For $\sigma_{\mathrm{Mon}} = (s\colon c, \mu\colon s\Box s \to s, u\colon I \to s)$, the arity/coarity terms yield support functors:

$$M(s\Box s)(x) = x \otimes x, \quad M(I)(x) = I, \quad M(s)(x) = x.$$

The inserter $M^\sigma = \mathbf{Ins}\,((M(s\Box s), M(I)), (M(s), M(s)))$ consists of **pointed magmas** $(x, m\colon x \otimes x \to x, e\colon I \to x)$.

Example: The Theory of Monoids

Consider a monoidal category $M = (\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ as a Σ_{Mon}-model in **CAT**.

1. Operations Object M^σ (Pointed Magmas)

For $\sigma_{\mathrm{Mon}} = (s\colon c, \mu\colon s\Box s \to s, u\colon I \to s)$, the arity/coarity terms yield support functors:

$$M(s\Box s)(x) = x \otimes x, \quad M(I)(x) = I, \quad M(s)(x) = x.$$

The inserter $M^\sigma = \mathbf{Ins}\,((M(s\Box s), M(I)), (M(s), M(s)))$ consists of **pointed magmas** $(x, m\colon x \otimes x \to x, e\colon I \to x)$.

2. Bialgebra Object M^T (Internal Monoids)

The equifier M^T chooses those objects (x, m, e) in M^σ that satsify the equations:

$$m \circ (m \otimes \mathrm{id}_x) = m \circ (\mathrm{id}_x \otimes m) \circ \alpha_{x,x,x}$$
$$m \circ (\mathrm{id}_x \otimes e) = \rho_x, \quad m \circ (e \otimes \mathrm{id}_x) = \lambda_x$$

Thus, $M^T \cong \mathbf{Mon}(M)$, the category of **internal monoids** in M.

Morphisms & The Induced Functor Theorem

Morphisms of Σ-Models

A morphism $(F, \eta) \colon M \to N$ assigns 1-cells $F_c \colon M_c \to N_c$ equipped with structural comparison 2-cells $\eta^B \colon N(B) \circ F_{\bar{c}} \Rightarrow F_d \circ M(B)$ for each Σ-functor symbol $B \colon \bar{c} \to d$ and satisfying a coherence axiom for each Σ-transformation symbol $\alpha^{\bar{x}} \colon t_1 \to t_2$.

- There are three notion of morphisms: **lax** ($\Rightarrow$), **oplax** ($\Leftarrow$), or **strong** ($\cong$).

Morphisms & The Induced Functor Theorem

Morphisms of Σ-Models

A morphism $(F, \eta)\colon M \to N$ assigns 1-cells $F_c\colon M_c \to N_c$ equipped with structural comparison 2-cells $\eta^B\colon N(B) \circ F_{\bar{c}} \Rightarrow F_d \circ M(B)$ for each Σ-functor symbol $B\colon \bar{c} \to d$ and satisfying a coherence axiom for each Σ-transformation symbol $\alpha^{\bar{x}}\colon t_1 \to t_2$.

- There are three notion of morphisms: **lax** ($\Rightarrow$), **oplax** ($\Leftarrow$), or **strong** ($\cong$).

Corollaries (Unifying Classical Lifting Theorems)

- If F is **lax** and T is **algebraic** $\implies$ F induces $F^T\colon M^T \to N^T$ *(e.g., classical lifting of lax monoidal functors to categories of internal monoids)*.

Morphisms & The Induced Functor Theorem

Morphisms of Σ-Models

A morphism $(F, \eta)\colon M \to N$ assigns 1-cells $F_c\colon M_c \to N_c$ equipped with structural comparison 2-cells $\eta^B\colon N(B) \circ F_{\bar{c}} \Rightarrow F_d \circ M(B)$ for each Σ-functor symbol $B\colon \bar{c} \to d$ and satisfying a coherence axiom for each Σ-transformation symbol $\alpha^{\bar{x}}\colon t_1 \to t_2$.

- There are three notion of morphisms: **lax** ($\Rightarrow$), **oplax** ($\Leftarrow$), or **strong** ($\cong$).

Corollaries (Unifying Classical Lifting Theorems)

- If F is **lax** and T is **algebraic** $\implies$ F induces $F^T\colon M^T \to N^T$
  (e.g., classical lifting of lax monoidal functors to categories of internal monoids).
- If F is **oplax** and T is **coalgebraic** $\implies$ F induces $F^T\colon M^T \to N^T$.

Morphisms & The Induced Functor Theorem

Morphisms of Σ-Models

A morphism $(F, \eta)\colon M \to N$ assigns 1-cells $F_c\colon M_c \to N_c$ equipped with structural comparison 2-cells $\eta^B\colon N(B) \circ F_{\bar{c}} \Rightarrow F_d \circ M(B)$ for each Σ-functor symbol $B\colon \bar{c} \to d$ and satisfying a coherence axiom for each Σ-transformation symbol $\alpha^{\bar{x}}\colon t_1 \to t_2$.

- There are three notion of morphisms: **lax** ($\Rightarrow$), **oplax** ($\Leftarrow$), or **strong** ($\cong$).

Corollaries (Unifying Classical Lifting Theorems)

- If F is **lax** and T is **algebraic** $\implies$ F induces $F^T\colon M^T \to N^T$ *(e.g., classical lifting of lax monoidal functors to categories of internal monoids)*.
- If F is **oplax** and T is **coalgebraic** $\implies$ F induces $F^T\colon M^T \to N^T$.
- If F is **strong** $\implies$ F induces $F^T\colon M^T \to N^T$ for **any** theory T.

Structural Lifting I: Internal Limits & Kan Extensions

Structural Lifting I: Internal Limits & Kan Extensions

Note: Kan Extensions as 2-Dimensional Limits

In 2-categories, left and right Kan extensions generalize colimits and limits.

Structural Lifting I: Internal Limits & Kan Extensions

Note: Kan Extensions as 2-Dimensional Limits

In 2-categories, left and right Kan extensions generalize colimits and limits.

Theorem (Lifting Kan Extensions Along PIE Limits)

Structural Lifting I: Internal Limits & Kan Extensions

Note: Kan Extensions as 2-Dimensional Limits

In 2-categories, left and right Kan extensions generalize colimits and limits.

Theorem (Lifting Kan Extensions Along PIE Limits)

Let $\mathcal{K}$ be a 2-category.

- **Products ($\prod_{i \in I} c_i$):** The projection 1-cells $(\pi_i \colon \prod_i c_i \to c_i)_{i \in I}$ **preserve and jointly strictly create** all left and right Kan extensions.

Structural Lifting I: Internal Limits & Kan Extensions

Note: Kan Extensions as 2-Dimensional Limits

In 2-categories, left and right Kan extensions generalize colimits and limits.

Theorem (Lifting Kan Extensions Along PIE Limits)

Let $\mathcal{K}$ be a 2-category.

- **Products ($\prod_{i \in I} c_i$):** The projection 1-cells $(\pi_i \colon \prod_i c_i \to c_i)_{i \in I}$ **preserve and jointly strictly create** all left and right Kan extensions.
- **Inserters ($\mathbf{Ins}(f, g) \xrightarrow{u} a$):** The canonical 1-cell u **strictly creates** left Kan extensions preserved by f, and right Kan extensions preserved by g.

Structural Lifting I: Internal Limits & Kan Extensions

Note: Kan Extensions as 2-Dimensional Limits

In 2-categories, left and right Kan extensions generalize colimits and limits.

Theorem (Lifting Kan Extensions Along PIE Limits)

Let $\mathcal{K}$ be a 2-category.

- **Products ($\prod_{i\in I} c_i$):** The projection 1-cells $(\pi_i \colon \prod_i c_i \to c_i)_{i\in I}$ **preserve and jointly strictly create** all left and right Kan extensions.
- **Inserters ($\mathbf{Ins}(f,g) \xrightarrow{u} a$):** The canonical 1-cell u **strictly creates** left Kan extensions preserved by f, and right Kan extensions preserved by g.
- **Equifiers ($\mathbf{Eq}(\eta,\theta) \xrightarrow{u} c$):** The canonical 1-cell u **strictly creates** left (right) Kan extensions that f (g) preserves.

Structural Lifting I: Internal Limits & Kan Extensions II

Corollary (Creation of Limits/Colimits for Bialgebras)

Let M be a Σ-model in a PIE-complete 2-category $\mathcal{K}$. Then the 1-cell $u\colon M^T \to M_0$ **strictly creates**:

Structural Lifting I: Internal Limits & Kan Extensions II

Corollary (Creation of Limits/Colimits for Bialgebras)

Let M be a Σ-model in a PIE-complete 2-category $\mathcal{K}$. Then the 1-cell $u\colon M^T \to M_0$ **strictly creates**:

- **Left Kan extensions (colimits)** preserved by all arity supports $M(t_1)$.

Structural Lifting I: Internal Limits & Kan Extensions II

Corollary (Creation of Limits/Colimits for Bialgebras)

Let M be a Σ-model in a PIE-complete 2-category $\mathcal{K}$. Then the 1-cell $u\colon M^T \to M_0$ **strictly creates**:

- **Left Kan extensions (colimits)** preserved by all arity supports $M(t_1)$.
- **Right Kan extensions (limits)** preserved by all coarity supports $M(t_2)$.

Structural Lifting I: Internal Limits & Kan Extensions II

Corollary (Creation of Limits/Colimits for Bialgebras)

Let M be a Σ-model in a PIE-complete 2-category $\mathcal{K}$. Then the 1-cell $u \colon M^T \to M_0$ **strictly creates**:

- **Left Kan extensions (colimits)** preserved by all arity supports $M(t_1)$.
- **Right Kan extensions (limits)** preserved by all coarity supports $M(t_2)$.
- **All right Kan extension** if T is algebraic, and **all left Kan extensions** if T is coalgebraic.

Structural Lifting II: Accessibility & Local Presentability

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed).

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed). If each $\mathcal{V}$-category M_c is $\mathcal{V}$-accessible

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed). If each $\mathcal{V}$-category M_c is $\mathcal{V}$-accessible and each $M(B)$ is $\mathcal{V}$-accessible, then:

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed). If each $\mathcal{V}$-category M_c is $\mathcal{V}$-accessible and each $M(B)$ is $\mathcal{V}$-accessible, then:

The $\mathcal{V}$-category M^T and the $\mathcal{V}$-functor $M^T \to M_0$ are $\mathcal{V}$-accessible.

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed). If each $\mathcal{V}$-category M_c is $\mathcal{V}$-accessible and each $M(B)$ is $\mathcal{V}$-accessible, then:

The $\mathcal{V}$-category M^T and the $\mathcal{V}$-functor $M^T \to M_0$ are $\mathcal{V}$-accessible.

Proof: Lack & Tendas (2023) showed $\mathcal{V}$-**ACC** is closed under PIE-limits in $\mathbf{CAT}_{\mathcal{V}}$.

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed). If each $\mathcal{V}$-category M_c is $\mathcal{V}$-accessible and each $M(B)$ is $\mathcal{V}$-accessible, then:

The $\mathcal{V}$-category M^T and the $\mathcal{V}$-functor $M^T \to M_0$ are $\mathcal{V}$-accessible.

Proof: Lack & Tendas (2023) showed $\mathcal{V}$-**ACC** is closed under PIE-limits in $\mathbf{CAT}_{\mathcal{V}}$.

Theorem ($\mathcal{V}$-Enriched Local Presentability)

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed). If each $\mathcal{V}$-category M_c is $\mathcal{V}$-accessible and each $M(B)$ is $\mathcal{V}$-accessible, then:

The $\mathcal{V}$-category M^T and the $\mathcal{V}$-functor $M^T \to M_0$ are $\mathcal{V}$-accessible.

Proof: Lack & Tendas (2023) showed $\mathcal{V}$-**ACC** is closed under PIE-limits in $\mathbf{CAT}_{\mathcal{V}}$.

Theorem ($\mathcal{V}$-Enriched Local Presentability)

Assume each M_c is $\mathcal{V}$-locally presentable and $M(B)$ is $\mathcal{V}$-accessible.

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed). If each $\mathcal{V}$-category M_c is $\mathcal{V}$-accessible and each $M(B)$ is $\mathcal{V}$-accessible, then:

The $\mathcal{V}$-category M^T and the $\mathcal{V}$-functor $M^T \to M_0$ are $\mathcal{V}$-accessible.

Proof: Lack & Tendas (2023) showed $\mathcal{V}$-**ACC** is closed under PIE-limits in $\mathbf{CAT}_{\mathcal{V}}$.

Theorem ($\mathcal{V}$-Enriched Local Presentability)

Assume each M_c is $\mathcal{V}$-locally presentable and $M(B)$ is $\mathcal{V}$-accessible.

1. **Algebraic Theories:** If coarity supports are continuous, then $U\colon M^T \to M_0$ is **monadic** and M^T is **locally presentable**.

Structural Lifting II: Accessibility & Local Presentability

Theorem ($\mathcal{V}$-Enriched Accessibility)

Let M be a Σ-model in $\mathbf{CAT}_{\mathcal{V}}$ ($\mathcal{V}$ locally presentable symmetric monoidal closed). If each $\mathcal{V}$-category M_c is $\mathcal{V}$-accessible and each $M(B)$ is $\mathcal{V}$-accessible, then:

The $\mathcal{V}$-category M^T and the $\mathcal{V}$-functor $M^T \to M_0$ are $\mathcal{V}$-accessible.

Proof: Lack & Tendas (2023) showed $\mathcal{V}$-$\mathbf{ACC}$ is closed under PIE-limits in $\mathbf{CAT}_{\mathcal{V}}$.

Theorem ($\mathcal{V}$-Enriched Local Presentability)

Assume each M_c is $\mathcal{V}$-locally presentable and $M(B)$ is $\mathcal{V}$-accessible.

1. **Algebraic Theories:** If coarity supports are continuous, then $U \colon M^T \to M_0$ is **monadic** and M^T is **locally presentable**.
2. **Coalgebraic Theories:** If arity supports are cocontinuous, then $U \colon M^T \to M_0$ is **comonadic** and M^T is $\mathcal{V}$-**locally presentable**.

Structural Lifting III: Orthogonal Factorization Systems (OFS)

Structural Lifting III: Orthogonal Factorization Systems (OFS)

Theorem (Lifting OFS Along PIE Limits)

Structural Lifting III: Orthogonal Factorization Systems (OFS)

Theorem (Lifting OFS Along PIE Limits)

- **Products ($\prod_i \mathcal{C}_i$):** If each $\mathcal{C}_i$ has an OFS, then $\prod_i \mathcal{C}_i$ inherits a unique OFS preserved by the projections.

Structural Lifting III: Orthogonal Factorization Systems (OFS)

Theorem (Lifting OFS Along PIE Limits)

- **Products ($\prod_i \mathcal{C}_i$):** If each $\mathcal{C}_i$ has an OFS, then $\prod_i \mathcal{C}_i$ inherits a unique OFS preserved by the projections.
- **Inserters ($\mathbf{Ins}(F, G)$):** If $\mathcal{C}, \mathcal{D}$ have OFSs, $F, G \colon \mathcal{C} \to \mathcal{D}$ preserve the left and the right classes, respectively, then $\mathbf{Ins}(F, G)$ inherits a unique OFS preserved by $U \colon \mathbf{Ins}(F, G) \to \mathcal{C}$.

Structural Lifting III: Orthogonal Factorization Systems (OFS)

Theorem (Lifting OFS Along PIE Limits)

- **Products ($\prod_i \mathcal{C}_i$):** If each $\mathcal{C}_i$ has an OFS, then $\prod_i \mathcal{C}_i$ inherits a unique OFS preserved by the projections.
- **Inserters ($\mathbf{Ins}(F, G)$):** If $\mathcal{C}, \mathcal{D}$ have OFSs, $F, G \colon \mathcal{C} \to \mathcal{D}$ preserve the left and the right classes, respectively, then $\mathbf{Ins}(F, G)$ inherits a unique OFS preserved by $U \colon \mathbf{Ins}(F, G) \to \mathcal{C}$.
- **Equifiers ($\mathbf{Eq}(\eta, \theta)$):** If $\mathcal{C}, \mathcal{D}$ have OFSs, where $F, G \colon \mathcal{C} \to \mathcal{D}$ preserve the left and right classes, respectively, then $\mathbf{Eq}(\eta, \theta)$ inherits a unique OFS preserved by $U \colon \mathbf{Eq}(\eta, \theta) \to \mathcal{C}$ for any parallel pair of natural transformations $\eta, \theta \colon F \Rightarrow G$.

Structural Lifting III: Orthogonal Factorization Systems (OFS)

Theorem (Lifting OFS Along PIE Limits)

- **Products ($\prod_i \mathcal{C}_i$):** If each $\mathcal{C}_i$ has an OFS, then $\prod_i \mathcal{C}_i$ inherits a unique OFS preserved by the projections.
- **Inserters ($\mathbf{Ins}(F, G)$):** If $\mathcal{C}, \mathcal{D}$ have OFSs, $F, G \colon \mathcal{C} \to \mathcal{D}$ preserve the left and the right classes, respectively, then $\mathbf{Ins}(F, G)$ inherits a unique OFS preserved by $U \colon \mathbf{Ins}(F, G) \to \mathcal{C}$.
- **Equifiers ($\mathbf{Eq}(\eta, \theta)$):** If $\mathcal{C}, \mathcal{D}$ have OFSs, where $F, G \colon \mathcal{C} \to \mathcal{D}$ preserve the left and right classes, respectively, then $\mathbf{Eq}(\eta, \theta)$ inherits a unique OFS preserved by $U \colon \mathbf{Eq}(\eta, \theta) \to \mathcal{C}$ for any parallel pair of natural transformations $\eta, \theta \colon F \Rightarrow G$.

Generalization to Bialgebras (M^T)

Let M be a categorical Σ-model where each M_c has an OFS (E^c, M^c). If arity supports $M(t_1)$ preserve E and coarity supports $M(t_2)$ preserve M, then:

M^T **inherits a unique OFS preserved by the forgetful functor** $U \colon M^T \to M_0$.

(In particular, holds if each $M(B)$ preserves left and right classes componentwise).

Structural Lifting IV: Regularity & Exactness

Theorem (Lifting Regularity & Exactness Along PIE Limits)

Let $\mathcal{C}_i, \mathcal{C}, \mathcal{D}$ be regular categories (or exact). Let $F, G \colon \mathcal{C} \to \mathcal{D}$ be functors.

- **Regularity:** $\prod_i \mathcal{C}_i$, $\mathbf{Ins}(F, G)$, and $\mathbf{Eq}(\eta, \theta)$ inherit regularity provided F preserves regular epimorphisms and G preserves finite limits.
- **Exactness:** Inherited if F additionally preserves coequalizers of kernel pairs (or reflexive coequalizers).

Structural Lifting IV: Regularity & Exactness

Theorem (Lifting Regularity & Exactness Along PIE Limits)

Let $\mathcal{C}_i, \mathcal{C}, \mathcal{D}$ be regular categories (or exact). Let $F, G \colon \mathcal{C} \to \mathcal{D}$ be functors.

- **Regularity:** $\prod_i \mathcal{C}_i$, $\mathbf{Ins}(F, G)$, and $\mathbf{Eq}(\eta, \theta)$ inherit regularity provided F preserves regular epimorphisms and G preserves finite limits.
- **Exactness:** Inherited if F additionally preserves coequalizers of kernel pairs (or reflexive coequalizers).

Application to Bialgebras (M^T)

Let M be a categorical Σ-model where each M_c is regular (exact). Then:

$$M^T \textbf{ is regular (exact)}$$

whenever each coarity support $M(t_2)$ preserves finite limits and each $M(B)$ preserves regular epimorphisms (or reflexive coequalizers) componentwise for each functor symbol B.

Summary & Takeaways

Key Contributions

Summary & Takeaways

Key Contributions

1. **Syntactic structure:** Signature pairs (Σ, σ) and theories T generate Ulmer's support functors automatically from formal syntax.

Summary & Takeaways

Key Contributions

1. **Syntactic structure:** Signature pairs (Σ, σ) and theories T generate Ulmer's support functors automatically from formal syntax.
2. 2-**Categorical Internalization:** Internal bialgebras M^T are constructed via strict PIE limits in any PIE-complete 2-category $\mathcal{K}$.

Summary & Takeaways

Key Contributions

1. **Syntactic structure:** Signature pairs (Σ, σ) and theories T generate Ulmer's support functors automatically from formal syntax.
2. 2-**Categorical Internalization:** Internal bialgebras M^T are constructed via strict PIE limits in any PIE-complete 2-category $\mathcal{K}$.
3. **Liftings:** Provides uniform proofs for lifting:

Summary & Takeaways

Key Contributions

1. **Syntactic structure:** Signature pairs (Σ, σ) and theories T generate Ulmer's support functors automatically from formal syntax.
2. 2-**Categorical Internalization:** Internal bialgebras M^T are constructed via strict PIE limits in any PIE-complete 2-category $\mathcal{K}$.
3. **Liftings:** Provides uniform proofs for lifting:
   - Model morphisms $F\colon M \to N \mapsto F^T\colon M^T \to N^T$ (Induced Functor of Algebras Theorem).

Summary & Takeaways

Key Contributions

1. **Syntactic structure:** Signature pairs (Σ, σ) and theories T generate Ulmer's support functors automatically from formal syntax.
2. 2-**Categorical Internalization:** Internal bialgebras M^T are constructed via strict PIE limits in any PIE-complete 2-category $\mathcal{K}$.
3. **Liftings:** Provides uniform proofs for lifting:
   - Model morphisms $F\colon M \to N \mapsto F^T\colon M^T \to N^T$ (Induced Functor of Algebras Theorem).
   - Limits, colimits (Kan extensions).

Summary & Takeaways

Key Contributions

1. **Syntactic structure:** Signature pairs (Σ, σ) and theories T generate Ulmer's support functors automatically from formal syntax.
2. 2-**Categorical Internalization:** Internal bialgebras M^T are constructed via strict PIE limits in any PIE-complete 2-category $\mathcal{K}$.
3. **Liftings:** Provides uniform proofs for lifting:
   - Model morphisms $F\colon M \to N \mapsto F^T\colon M^T \to N^T$ (Induced Functor of Algebras Theorem).
   - Limits, colimits (Kan extensions).
   - Accessibility and Local Presentability.

Summary & Takeaways

Key Contributions

1. **Syntactic structure:** Signature pairs (Σ, σ) and theories T generate Ulmer's support functors automatically from formal syntax.
2. 2-**Categorical Internalization:** Internal bialgebras M^T are constructed via strict PIE limits in any PIE-complete 2-category $\mathcal{K}$.
3. **Liftings:** Provides uniform proofs for lifting:
   - Model morphisms $F\colon M \to N \mapsto F^T\colon M^T \to N^T$ (Induced Functor of Algebras Theorem).
   - Limits, colimits (Kan extensions).
   - Accessibility and Local Presentability.
   - Factorization systems, regularity, and exactness.

References

J. Adámek and J. Rosický.
Locally Presentable and Accessible Categories.
Cambridge Univ. Press, 1994.

E. J. Dubuc.
Kan Extensions in Enriched Category Theory
Springer-Verlag, Berlin-New York, 1970.

D. Forsman.
A Syntactic Approach to Ulmer's Bialgebras.
preprint, 2026. arXiv:2607.19587.

G. M. Kelly.
Basic Concepts of Enriched Category Theory.
Cambridge Univ. Press, 1982.

S. Lack and G. Tendas.
Virtual concepts in the theory of accessible categories.
J. Pure Appl. Algebra, 227(2):107196, 2023.

F. Ulmer.
Bialgebras in locally presentable categories.
Preprint, Univ. of Wuppertal, 1977.

Thank you for your attention!