Teoría de Categorías en la enseñanza de la lógica.

Cecilia Chávez Aguilera

Instituto de Investiaciones Filosóficas

U.N.A.M.

2 de abril de 2009

Agenda

Agenda

Motivaciones

- Tarski *On some fundamental concepts of metamathematics*, 1928.
- Dado un conjunto A considérense las funciones:
  Cn: $\wp(A) \longrightarrow \wp(A)$
  tales que satisfacen para cada $X \subseteq A$,

  1. $X \subseteq Cn(X)$
  2. $Cn(Cn(X)) = Cn(X)$
  3. $Cn(X) = \bigcup\{Cn(Y) : Y \subseteq X, Y finito\}$
     La condición 3 es llamada la condición finitaria, e implica la propiedad débil de monotonía de C, es decir:
  4. Si $X \subseteq Y$, entonces $Cn(X) \subseteq Cn(Y)$

- El operador de consecuencia no necesita definirse sobre un conjunto contable y es cualquier función $Cn : \wp(A) \longrightarrow \wp(A)$ que satisface las condiciones 1 y 2. Si además satisface 3 el operador de consecuencia se denomina finitario.
- Łos y Susko *Remarks on sentential logics* (1958)
- Una lógica, en un lenguaje $\mathcal{L}$ es un par $\mathcal{S} = \langle Fm, \vdash_{\mathcal{S}} \rangle$ donde $\vdash_{\mathcal{S}}$ es una relación: $\vdash \subseteq \wp(Fm) \times Fm$ que satisface:
  1. Si $a \in X$, entonces $X \vdash_{\mathcal{S}} a$
  2. Si $Y \vdash_{\mathcal{S}} a$ para toda $a \in X$, y $X \vdash_{\mathcal{S}} b$, entonces $Y \vdash_{\mathcal{S}} b$
  3. Si $X \vdash_{\mathcal{S}} a$ y $X \subseteq Y$, entonces $Y \vdash_{\mathcal{S}} a$
  4. Si $\Gamma \vdash_{\mathcal{S}} \varphi$, entonces para toda sustitución σ, $\sigma[\Gamma] \vdash_{\mathcal{S}} \sigma(\varphi)$

- El operador de consecuencia no necesita definirse sobre un conjunto contable y es cualquier función $Cn : \wp(A) \longrightarrow \wp(A)$ que satisface las condiciones 1 y 2. Si además satisface 3 el operador de consecuencia se denomina finitario.
- Łos y Susko *Remarks on sentential logics* (1958)
- Una lógica, en un lenguaje $\mathcal{L}$ es un par $\mathcal{S} = \langle Fm, \vdash_{\mathcal{S}} \rangle$ donde $\vdash_{\mathcal{S}}$ es una relación: $\vdash \subseteq \wp(Fm) \times Fm$ que satisface:
  1. Si $a \in X$, entonces $X \vdash_{\mathcal{S}} a$
  2. Si $Y \vdash_{\mathcal{S}} a$ para toda $a \in X$, y $X \vdash_{\mathcal{S}} b$, entonces $Y \vdash_{\mathcal{S}} b$
  3. Si $X \vdash_{\mathcal{S}} a$ y $X \subseteq Y$, entonces $Y \vdash_{\mathcal{S}} a$
  4. Si $\Gamma \vdash_{\mathcal{S}} \varphi$, entonces para toda sustitución σ, $\sigma[\Gamma] \vdash_{\mathcal{S}} \sigma(\varphi)$

- El operador de consecuencia no necesita definirse sobre un conjunto contable y es cualquier función $Cn : \wp(A) \longrightarrow \wp(A)$ que satisface las condiciones 1 y 2. Si además satisface 3 el operador de consecuencia se denomina finitario.
- Łos y Susko *Remarks on sentential logics* (1958)
- Una lógica, en un lenguaje $\mathcal{L}$ es un par $\mathcal{S} = \langle Fm, \vdash_{\mathcal{S}} \rangle$ donde $\vdash_{\mathcal{S}}$ es una relación: $\vdash \subseteq \wp(Fm) \times Fm$ que satisface:
  1. Si $a \in X$, entonces $X \vdash_{\mathcal{S}} a$
  2. Si $Y \vdash_{\mathcal{S}} a$ para toda $a \in X$, y $X \vdash_{\mathcal{S}} b$, entonces $Y \vdash_{\mathcal{S}} b$
  3. Si $X \vdash_{\mathcal{S}} a$ y $X \subseteq Y$, entonces $Y \vdash_{\mathcal{S}} a$
  4. Si $\Gamma \vdash_{\mathcal{S}} \varphi$, entonces para toda sustitución σ, $\sigma[\Gamma] \vdash_{\mathcal{S}} \sigma(\varphi)$

- El operador de consecuencia no necesita definirse sobre un conjunto contable y es cualquier función $Cn : \wp(A) \longrightarrow \wp(A)$ que satisface las condiciones 1 y 2. Si además satisface 3 el operador de consecuencia se denomina finitario.
- Łos y Susko *Remarks on sentential logics* (1958)
- Una lógica, en un lenguaje $\mathcal{L}$ es un par $\mathcal{S} = \langle Fm, \vdash_{\mathcal{S}} \rangle$ donde $\vdash_{\mathcal{S}}$ es una relación: $\vdash \subseteq \wp(Fm) \times Fm$ que satisface:

1. Si $a \in X$, entonces $X \vdash_{\mathcal{S}} a$
2. Si $Y \vdash_{\mathcal{S}} a$ para toda $a \in X$, y $X \vdash_{\mathcal{S}} b$, entonces $Y \vdash_{\mathcal{S}} b$
3. Si $X \vdash_{\mathcal{S}} a$ y $X \subseteq Y$, entonces $Y \vdash_{\mathcal{S}} a$
4. Si $\Gamma \vdash_{\mathcal{S}} \varphi$, entonces para toda sustitución $\sigma, \sigma[\Gamma] \vdash_{\mathcal{S}} \sigma(\varphi)$

- El operador de consecuencia no necesita definirse sobre un conjunto contable y es cualquier función $Cn : \wp(A) \longrightarrow \wp(A)$ que satisface las condiciones 1 y 2. Si además satisface 3 el operador de consecuencia se denomina finitario.
- Łos y Susko *Remarks on sentential logics* (1958)
- Una lógica, en un lenguaje $\mathcal{L}$ es un par $\mathcal{S} = \langle Fm, \vdash_{\mathcal{S}} \rangle$ donde $\vdash_{\mathcal{S}}$ es una relación: $\vdash \subseteq \wp(Fm) \times Fm$ que satisface:
  1. Si $a \in X$, entonces $X \vdash_{\mathcal{S}} a$
  2. Si $Y \vdash_{\mathcal{S}} a$ para toda $a \in X$, y $X \vdash_{\mathcal{S}} b$, entonces $Y \vdash_{\mathcal{S}} b$
  3. Si $X \vdash_{\mathcal{S}} a$ y $X \subseteq Y$, entonces $Y \vdash_{\mathcal{S}} a$
  4. Si $\Gamma \vdash_{\mathcal{S}} \varphi$, entonces para toda sustitución σ, $\sigma[\Gamma] \vdash_{\mathcal{S}} \sigma(\varphi)$

- El operador de consecuencia no necesita definirse sobre un conjunto contable y es cualquier función $Cn : \wp(A) \longrightarrow \wp(A)$ que satisface las condiciones 1 y 2. Si además satisface 3 el operador de consecuencia se denomina finitario.
- Łos y Susko *Remarks on sentential logics* (1958)
- Una lógica, en un lenguaje $\mathcal{L}$ es un par $\mathcal{S} = \langle Fm, \vdash_{\mathcal{S}} \rangle$ donde $\vdash_{\mathcal{S}}$ es una relación: $\vdash \subseteq \wp(Fm) \times Fm$ que satisface:

  1. Si $a \in X$, entonces $X \vdash_{\mathcal{S}} a$
  2. Si $Y \vdash_{\mathcal{S}} a$ para toda $a \in X$, y $X \vdash_{\mathcal{S}} b$, entonces $Y \vdash_{\mathcal{S}} b$
  3. Si $X \vdash_{\mathcal{S}} a$ y $X \subseteq Y$, entonces $Y \vdash_{\mathcal{S}} a$
  4. Si $\Gamma \vdash_{\mathcal{S}} \varphi$, entonces para toda sustitución σ, $\sigma[\Gamma] \vdash_{\mathcal{S}} \sigma(\varphi)$

- El operador de consecuencia no necesita definirse sobre un conjunto contable y es cualquier función $Cn : \wp(A) \longrightarrow \wp(A)$ que satisface las condiciones 1 y 2. Si además satisface 3 el operador de consecuencia se denomina finitario.
- Łos y Susko *Remarks on sentential logics* (1958)
- Una lógica, en un lenguaje $\mathcal{L}$ es un par $\mathcal{S} = \langle Fm, \vdash_{\mathcal{S}} \rangle$ donde $\vdash_{\mathcal{S}}$ es una relación: $\vdash \subseteq \wp(Fm) \times Fm$ que satisface:
  1. Si $a \in X$, entonces $X \vdash_{\mathcal{S}} a$
  2. Si $Y \vdash_{\mathcal{S}} a$ para toda $a \in X$, y $X \vdash_{\mathcal{S}} b$, entonces $Y \vdash_{\mathcal{S}} b$
  3. Si $X \vdash_{\mathcal{S}} a$ y $X \subseteq Y$, entonces $Y \vdash_{\mathcal{S}} a$
  4. Si $\Gamma \vdash_{\mathcal{S}} \varphi$, entonces para toda sustitución σ, $\sigma[\Gamma] \vdash_{\mathcal{S}} \sigma(\varphi)$

- Fundacional
  - Barwise *Axioms for abstract model theory* 1974
  - Gabbay *What is a logic?* 1994
- Computacional
  - Goguen y Burstall *Introducing institutions* 1984
  - Fiadeiro y Sernadas *Structuring Theories on Consequence* 1988
  - Meseguer *General Logics* 1989

- Fundacional
    - Barwise *Axioms for abstract model theory* 1974
    - Gabbay *What is a logic?* 1994
- Computacional
    - Goguen y Burstall *Introducing institutions* 1984
    - Fiadeiro y Sernadas *Structuring Theories on Consequence* 1988
    - Meseguer *General Logics* 1989

- Fundacional
  - Barwise *Axioms for abstract model theory* 1974
  - Gabbay *What is a logic?* 1994
- Computacional
  - Goguen y Burstall *Introducing institutions* 1984
  - Fiadeiro y Sernadas *Structuring Theories on Consequence* 1988
  - Meseguer *General Logics* 1989

- Fundacional
  - Barwise *Axioms for abstract model theory* 1974
  - Gabbay *What is a logic?* 1994
- Computacional
  - Goguen y Burstall *Introducing institutions* 1984
  - Fiadeiro y Sernadas *Structuring Theories on Consequence* 1988
  - Meseguer *General Logics* 1989

- Fundacional
  - Barwise *Axioms for abstract model theory* 1974
  - Gabbay *What is a logic?* 1994
- Computacional
  - Goguen y Burstall *Introducing institutions* 1984
  - Fiadeiro y Sernadas *Structuring Theories on Consequence* 1988
  - Meseguer *General Logics* 1989

- Fundacional
  - Barwise *Axioms for abstract model theory* 1974
  - Gabbay *What is a logic?* 1994
- Computacional
  - Goguen y Burstall *Introducing institutions* 1984
  - Fiadeiro y Sernadas *Structuring Theories on Consequence* 1988
  - Meseguer *General Logics* 1989

- Fundacional
  - Barwise *Axioms for abstract model theory* 1974
  - Gabbay *What is a logic?* 1994
- Computacional
  - Goguen y Burstall *Introducing institutions* 1984
  - Fiadeiro y Sernadas *Structuring Theories on Consequence* 1988
  - Meseguer *General Logics* 1989

Agenda

Un sistema lógico es:Semántica + Sintaxis + Cálculo de Pruebas Informalmente, una institución consiste de: **Sign**, dos funtores SEN y MOD que dan, respectivamente, para cada objeto Σ en Ob(**Sign**), un conjunto de Σ-fórmulas y una categoría de $\Sigma-$modelos. Para cada objeto Σ de las signaturas, fórmulas y modelos son relacionados vía una Σ-relación de satisfacción. Sus principales axiomas formalizan la propiedad de que la verdad es invariable bajo cambio de notación o traducción.La noción de $\pi - institución$ que es la contraparte natural de la noción de institución para la relación de derivabilidad, involucra una relación $\vdash$

Un sistema lógico es:Semántica + Sintaxis + Cálculo de PruebasInformalmente, una institución consiste de: **Sign**, dos funtores SEN y MOD que dan, respectivamente, para cada objeto Σ en Ob(**Sign**), un conjunto de Σ-fórmulas y una categoría de $\Sigma-$modelos. Para cada objeto Σ de las signaturas, fórmulas y modelos son relacionados vía una Σ-relación de satisfacción. Sus principales axiomas formalizan la propiedad de que la verdad es invariable bajo cambio de notación o traducción.La noción de $\pi - institución$ que es la contraparte natural de la noción de institución para la relación de derivabilidad, involucra una relación $\vdash$

Un sistema lógico es:Semántica + Sintaxis + Cálculo de PruebasInformalmente, una institución consiste de: **Sign**, dos funtores SEN y MOD que dan, respectivamente, para cada objeto Σ en Ob(**Sign**), un conjunto de Σ-fórmulas y una categoría de $\Sigma-$modelos. Para cada objeto Σ de las signaturas, fórmulas y modelos son relacionados vía una Σ-relación de satisfacción. Sus principales axiomas formalizan la propiedad de que la verdad es invariable bajo cambio de notación o traducción.La noción de $\pi - \mathit{institución}$ que es la contraparte natural de la noción de institución para la relación de derivabilidad, involucra una relación $\vdash$

- La categoría **Sign**:
  - $Ob(\textbf{Sign}) = \{\Sigma | \Sigma = \langle S_0, \ldots, S_n \rangle\}$
  - $Mor(\textbf{Sign}) = H : \Sigma \to \Sigma'$ Morfismos de signaturas
  - Multiplicación e identidad usuales
- Lógica Proposicional Clásica
  $\Sigma = \langle F, P \rangle$ donde,
  $F = \{f_1^\alpha, \ldots f_n^\kappa\}$ con f_i^λ un símbolo de función de aridad λ
  $P = \{p_1^\alpha, \ldots, p_m^\kappa\}$ donde cada p_i^λ es un símbolo de predicado de aridad λ
  Los morfismos son funciones que preservan aridad

- La categoría **Sign**:
    - $Ob(\mathbf{Sign}) = \{\Sigma | \Sigma = \langle S_0, \ldots, S_n \rangle\}$
    - $Mor(\mathbf{Sign}) = H : \Sigma \to \Sigma'$ Morfismos de signaturas
    - Multiplicación e identidad usuales
- Lógica Proposicional Clásica
  $\Sigma = \langle F, P \rangle$ donde,
  $F = \{f_1^{\alpha}, \ldots f_n^{\kappa}\}$ con f_i^{λ} un símbolo de función de aridad λ
  $P = \{p_1^{\alpha}, \ldots, p_m^{\kappa}\}$ donde cada p_i^{λ} es un símbolo de predicado de aridad λ
  Los morfismos son funciones que preservan aridad

- La categoría **Sign**:
  - $Ob(\mathbf{Sign}) = \{\Sigma | \Sigma = \langle S_0, \ldots, S_n \rangle\}$
  - $Mor(\mathbf{Sign}) = H : \Sigma \to \Sigma'$ Morfismos de signaturas
  - Multiplicación e identidad usuales
- Lógica Proposicional Clásica
  $\Sigma = \langle F, P \rangle$ donde,
  $F = \{f_1^{\alpha}, \ldots f_n^{\kappa}\}$ con f_i^{λ} un símbolo de función de aridad λ
  $P = \{p_1^{\alpha}, \ldots, p_m^{\kappa}\}$ donde cada p_i^{λ} es un símbolo de predicado de aridad λ
  Los morfismos son funciones que preservan aridad

- La categoría **Sign**:
  - $Ob(\mathbf{Sign}) = \{\Sigma | \Sigma = \langle S_0, \ldots, S_n \rangle\}$
  - $Mor(\mathbf{Sign}) = H : \Sigma \to \Sigma'$ Morfismos de signaturas
  - Multiplicación e identidad usuales
- Lógica Proposicional Clásica
  $\Sigma = \langle F, P \rangle$ donde,
  $F = \{f_1^\alpha, \ldots f_n^\kappa\}$ con f_i^λ un símbolo de función de aridad λ
  $P = \{p_1^\alpha, \ldots, p_m^\kappa\}$ donde cada p_i^λ es un símbolo de predicado de aridad λ
  Los morfismos son funciones que preservan aridad

- La categoría **Sign**:
  - $Ob(\mathbf{Sign}) = \{\Sigma | \Sigma = \langle S_0, \ldots, S_n \rangle\}$
  - $Mor(\mathbf{Sign}) = H : \Sigma \rightarrow \Sigma'$ Morfismos de signaturas
  - Multiplicación e identidad usuales
- Lógica Proposicional Clásica
  $\Sigma = \langle F, P \rangle$ donde,
  $F = \{f_1^{\alpha}, \ldots f_n^{\kappa}\}$ con f_i^{λ} un símbolo de función de aridad λ
  $P = \{p_1^{\alpha}, \ldots, p_m^{\kappa}\}$ donde cada p_i^{λ} es un símbolo de predicado de aridad λ
  Los morfismos son funciones que preservan aridad

- Denotamos $SEN(\Sigma)$ el conunto de todas las fórmulas de una signatura Σ
- Para cada $s \in SEN(\Sigma)$ dado un morfismo $H : \Sigma \to \Sigma'$, $H(s) \in SEN(\Sigma')$, y $H(s)$ se obtiene $s' \in SEN(\Sigma')$
- Una π-institución es una terna $\xi = \langle \mathbf{Sign}, SEN, \vdash \rangle$ donde:
  - **Sign** es una categoría cuyos objetos son signaturas y sus
  - Un funtor SEN: $\mathbf{Sign} \to \mathbf{Set}$, llamado el funtor de fórmulas, que asocia a cada signatura Σ un conjunto llamado el conjunto de Σ-fórmulas, cuyos elementos son fórmulas sobre esa signatura.
  - Una función $\vdash$: $Ob(\mathbf{Sign}) \to \wp(\wp(SEN(\Sigma)) \times SEN(\Sigma))$, que asocia a cada $\Sigma \in Ob(\mathbf{Sign})$ una relación binaria $\vdash_\Sigma \subseteq \wp(SEN(\Sigma)) \times SEN(\Sigma)$ que satisface las siguientes propiedades:

- Denotamos $SEN(\Sigma)$ el conunto de todas las fórmulas de una signatura Σ
- Para cada $s \in SEN(\Sigma)$ dado un morfismo $H : \Sigma \to \Sigma'$, $H(s) \in SEN(\Sigma')$, y $H(s)$ se obtiene $s' \in SEN(\Sigma')$
- Una π-institución es una terna $\xi = \langle \mathbf{Sign}, SEN, \vdash \rangle$ donde:
  - **Sign** es una categoría cuyos objetos son signaturas y sus morfismos son morfismos de signaturas.
  - Un funtor SEN: $\mathbf{Sign} \to \mathbf{Set}$, llamado el funtor de fórmulas, que asocia a cada signatura Σ un conjunto llamado el conjunto de Σ-fórmulas, cuyos elementos son fórmulas sobre esa signatura.
  - Una función $\vdash$: $Ob(\mathbf{Sign}) \to \wp(\wp(SEN(\Sigma)) \times SEN(\Sigma))$, que asocia a cada $\Sigma \in Ob(\mathbf{Sign})$ una relación binaria $\vdash_\Sigma \subseteq \wp(SEN(\Sigma)) \times SEN(\Sigma)$ que satisface las siguientes propiedades:

- Denotamos $SEN(\Sigma)$ el conunto de todas las fórmulas de una signatura Σ
- Para cada $s \in SEN(\Sigma)$ dado un morfismo $H : \Sigma \rightarrow \Sigma'$, $H(s) \in SEN(\Sigma')$, y $H(s)$ se obtiene $s' \in SEN(\Sigma')$
- Una π-institución es una terna $\xi = \langle \mathbf{Sign}, SEN, \vdash \rangle$ donde:
  - **Sign** es una categoría cuyos objetos son signaturas y sus morfismos son morfismos de signaturas.
  - Un funtor SEN: **Sign** $\rightarrow$ **Set**, llamado el funtor de fórmulas, que asocia a cada signatura Σ un conjunto llamado el conjunto de Σ-fórmulas, cuyos elementos son fórmulas sobre esa signatura.
  - Una función $\vdash$: $Ob(\mathbf{Sign}) \rightarrow \wp(\wp(SEN(\Sigma)) \times SEN(\Sigma))$, que asocia a cada $\Sigma \in Ob(\mathbf{Sign})$ una relación binaria $\vdash_\Sigma \subseteq \wp(SEN(\Sigma)) \times SEN(\Sigma)$ que satisface las siguientes propiedades:

- Denotamos $SEN(\Sigma)$ el conunto de todas las fórmulas de una signatura Σ
- Para cada $s \in SEN(\Sigma)$ dado un morfismo $H : \Sigma \to \Sigma'$, $H(s) \in SEN(\Sigma')$, y $H(s)$ se obtiene $s' \in SEN(\Sigma')$
- Una π-institución es una terna $\xi = \langle \mathbf{Sign}, SEN, \vdash \rangle$ donde:
    - **Sign** es una categoría cuyos objetos son signaturas y sus morfismos son morfismos de signaturas.
    - Un funtor SEN: $\mathbf{Sign} \to \mathbf{Set}$, llamado el funtor de fórmulas, que asocia a cada signatura Σ un conjunto llamado el conjunto de Σ-fórmulas, cuyos elementos son fórmulas sobre esa signatura.
    - Una función $\vdash$: $Ob(\mathbf{Sign}) \to \wp(\wp(SEN(\Sigma)) \times SEN(\Sigma))$, que asocia a cada $\Sigma \in Ob(\mathbf{Sign})$ una relación binaria $\vdash_\Sigma \subseteq \wp(SEN(\Sigma)) \times SEN(\Sigma)$ que satisface las siguientes propiedades:

- Denotamos $SEN(\Sigma)$ el conunto de todas las fórmulas de una signatura Σ
- Para cada $s \in SEN(\Sigma)$ dado un morfismo $H : \Sigma \to \Sigma'$, $H(s) \in SEN(\Sigma')$, y $H(s)$ se obtiene $s' \in SEN(\Sigma')$
- Una π-institución es una terna $\xi = \langle \mathbf{Sign}, SEN, \vdash \rangle$ donde:
    - **Sign** es una categoría cuyos objetos son signaturas y sus morfismos son morfismos de signaturas.
    - Un funtor SEN: **Sign** $\to$ **Set**, llamado el funtor de fórmulas, que asocia a cada signatura Σ un conjunto llamado el conjunto de Σ-fórmulas, cuyos elementos son fórmulas sobre esa signatura.
    - Una función $\vdash$: $Ob(\mathbf{Sign}) \to \wp(\wp(SEN(\Sigma)) \times SEN(\Sigma))$, que asocia a cada $\Sigma \in Ob(\mathbf{Sign})$ una relación binaria $\vdash_\Sigma \subseteq \wp(SEN(\Sigma)) \times SEN(\Sigma)$ que satisface las siguientes propiedades:

- Denotamos $SEN(\Sigma)$ el conunto de todas las fórmulas de una signatura Σ
- Para cada $s \in SEN(\Sigma)$ dado un morfismo $H : \Sigma \to \Sigma'$, $H(s) \in SEN(\Sigma')$, y $H(s)$ se obtiene $s' \in SEN(\Sigma')$
- Una π-institución es una terna $\xi = \langle \mathbf{Sign}, SEN, \vdash \rangle$ donde:
    - **Sign** es una categoría cuyos objetos son signaturas y sus morfismos son morfismos de signaturas.
    - Un funtor SEN: $\mathbf{Sign} \to \mathbf{Set}$, llamado el funtor de fórmulas, que asocia a cada signatura Σ un conjunto llamado el conjunto de Σ-fórmulas, cuyos elementos son fórmulas sobre esa signatura.
    - Una función $\vdash$: $Ob(\mathbf{Sign}) \to \wp(\wp(SEN(\Sigma)) \times SEN(\Sigma))$, que asocia a cada $\Sigma \in Ob(\mathbf{Sign})$ una relación binaria $\vdash_\Sigma \subseteq \wp(SEN(\Sigma)) \times SEN(\Sigma)$ que satisface las siguientes propiedades:

1. Reflexividad: Para toda $\varphi \in SEN(\Sigma), \;\; \{\varphi\} \vdash_\Sigma \varphi$.
2. Monotonía: Si $\Gamma \vdash_\Sigma \varphi$ y $\Gamma \subseteq \Gamma'$, *entonces* $\Gamma' \vdash_\Sigma \varphi$.
3. Transitividad: Si $\Gamma \vdash_\Sigma \varphi_i, i \in I$, y $\Gamma \cup \{\varphi_i | i \in I\} \vdash_\Sigma \psi$, *entonces* $\Gamma \vdash_\Sigma \psi$.
4. $\vdash$-traducibilidad: Si $\Gamma \vdash_\Sigma \varphi$ entonces para cualquier $H \in Mor(\mathbf{Sign}), H : \Sigma \to \Sigma'$:
$$H(\Gamma) \vdash_{\Sigma'} H(\varphi).$$

1. Reflexividad: Para toda $\varphi \in SEN(\Sigma)$, $\{\varphi\} \vdash_\Sigma \varphi$.
2. Monotonía: Si $\Gamma \vdash_\Sigma \varphi$ *y* $\Gamma \subseteq \Gamma'$, *entonces* $\Gamma' \vdash_\Sigma \varphi$.
3. Transitividad: Si $\Gamma \vdash_\Sigma \varphi_i, i \in I$, y $\Gamma \cup \{\varphi_i | i \in I\} \vdash_\Sigma \psi$, *entonces* $\Gamma \vdash_\Sigma \psi$.
4. $\vdash$-traducibilidad: Si $\Gamma \vdash_\Sigma \varphi$ entonces para cualquier $H \in Mor(\mathbf{Sign})$, $H : \Sigma \to \Sigma'$:

$$H(\Gamma) \vdash_{\Sigma'} H(\varphi).$$

1. Reflexividad: Para toda $\varphi \in SEN(\Sigma)$, $\{\varphi\} \vdash_\Sigma \varphi$.
2. Monotonía: Si $\Gamma \vdash_\Sigma \varphi$ *y* $\Gamma \subseteq \Gamma'$, *entonces* $\Gamma' \vdash_\Sigma \varphi$.
3. Transitividad: Si $\Gamma \vdash_\Sigma \varphi_i, i \in I$, y $\Gamma \cup \{\varphi_i | i \in I\} \vdash_\Sigma \psi$, *entonces* $\Gamma \vdash_\Sigma \psi$.
4. $\vdash$-traducibilidad: Si $\Gamma \vdash_\Sigma \varphi$ entonces para cualquier $H \in Mor(\mathbf{Sign})$, $H : \Sigma \to \Sigma'$:

$$H(\Gamma) \vdash_{\Sigma'} H(\varphi).$$

1. Reflexividad: Para toda $\varphi \in SEN(\Sigma)$, $\{\varphi\} \vdash_\Sigma \varphi$.
2. Monotonía: Si $\Gamma \vdash_\Sigma \varphi$ *y* $\Gamma \subseteq \Gamma'$, *entonces* $\Gamma' \vdash_\Sigma \varphi$.
3. Transitividad: Si $\Gamma \vdash_\Sigma \varphi_i, i \in I$, y $\Gamma \cup \{\varphi_i | i \in I\} \vdash_\Sigma \psi$, *entonces* $\Gamma \vdash_\Sigma \psi$.
4. $\vdash$-traducibilidad: Si $\Gamma \vdash_\Sigma \varphi$ entonces para cualquier $H \in Mor(\mathbf{Sign}), H : \Sigma \to \Sigma'$:

$$H(\Gamma) \vdash_{\Sigma'} H(\varphi).$$

- Definimos **Th** y **Mod**
  Dada una π-institución ξ, podemos asociar a ella la categoría **Th**
- $Ob(\mathbf{Th}) = \{T | T = \langle \Sigma, \Gamma \rangle\}$. con Σ una signatura y $\Gamma \subseteq SEN(\Sigma)$,
- Mor(**Th**)$= \{H : \langle \Sigma, \Gamma \rangle \rightarrow \langle \Sigma', \Gamma' \rangle\}$ tales que:

$$Si\ \varphi \in \Gamma\ entonces\ \Gamma' \vdash_{\Sigma'} H(\varphi)$$

- Multiplicación e Identidad usuales

- Definimos **Th** y **Mod**
  Dada una π-institución ξ, podemos asociar a ella la categoría **Th**
- $Ob(\mathbf{Th}) = \{T | T = \langle \Sigma, \Gamma \rangle\}$. con Σ una signatura y $\Gamma \subseteq SEN(\Sigma)$,
- Mor(**Th**)= $\{H : \langle \Sigma, \Gamma \rangle \rightarrow \langle \Sigma', \Gamma' \rangle\}$ tales que:

$$Si\ \varphi \in \Gamma\ entonces\ \Gamma' \vdash_{\Sigma'} H(\varphi)$$

- Multiplicación e Identidad usuales

- Definimos **Th** y **Mod**
  Dada una π-institución ξ, podemos asociar a ella la categoría **Th**
- $Ob(\mathbf{Th}) = \{T | T = \langle \Sigma, \Gamma \rangle\}$. con Σ una signatura y $\Gamma \subseteq SEN(\Sigma)$,
- Mor(**Th**)= $\{H : \langle \Sigma, \Gamma \rangle \rightarrow \langle \Sigma', \Gamma' \rangle\}$ tales que:

$$Si\ \varphi \in \Gamma\ entonces\ \Gamma' \vdash_{\Sigma'} H(\varphi)$$

- Multiplicación e Identidad usuales

- Definimos **Th** y **Mod**
  Dada una π-institución ξ, podemos asociar a ella la categoría **Th**
- $Ob(\mathbf{Th}) = \{T | T = \langle \Sigma, \Gamma \rangle\}$. con Σ una signatura y $\Gamma \subseteq SEN(\Sigma)$,
- Mor(**Th**)= $\{H : \langle \Sigma, \Gamma \rangle \to \langle \Sigma', \Gamma' \rangle\}$ tales que:

$$Si\ \varphi \in \Gamma\ entonces\ \Gamma' \vdash_{\Sigma'} H(\varphi)$$

- Multiplicación e Identidad usuales

- Σ asociamos a ella la categoría $\mathbf{Mod}(\Sigma)$
- $Ob(\mathbf{Mod}(\Sigma)) = \{\mathcal{M} | \mathcal{M} = \langle \Sigma, f \rangle\}$
- $Mor(\mathbf{Mod}(\Sigma)) = H : \langle \Sigma, f \rangle \to \langle \Sigma', f' \rangle$ tales que preservan operaciones y predicados.
- Multiplicación e Identidad usual
- Esto se puede representar dando el funtor MOD:
  MOD: $\boldsymbol{Sign} \to \boldsymbol{Cat}^{op}$
  utilizando la notación $MOD(H) = H^b$
  H permite ver un Σ'-modelo M' como un Σ-modelo $H^b(M')$ con:

$$H^b : MOD(\Sigma') \to MOD(\Sigma).$$

- Σ asociamos a ella la categoría $\mathbf{Mod}(\Sigma)$
- $Ob(\mathbf{Mod}(\Sigma)) = \{\mathcal{M} | \mathcal{M} = \langle \Sigma, f \rangle\}$
- $Mor(\mathbf{Mod}(\Sigma)) = H : \langle \Sigma, f \rangle \to \langle \Sigma', f' \rangle$ tales que preservan operaciones y predicados.
- Multiplicación e Identidad usual
- Esto se puede representar dando el funtor MOD:
  MOD: $\textbf{\textit{Sign}} \to \textbf{\textit{Cat}}^{op}$
  utilizando la notación $MOD(H) = H^b$
  H permite ver un Σ'-modelo M' como un Σ-modelo $H^b(M')$ con:

$$H^b : MOD(\Sigma') \to MOD(\Sigma).$$

- Σ asociamos a ella la categoría $\mathbf{Mod}(\Sigma)$
- $Ob(\mathbf{Mod}(\Sigma)) = \{\mathcal{M} | \mathcal{M} = \langle \Sigma, f \rangle\}$
- $Mor(\mathbf{Mod}(\Sigma)) = H : \langle \Sigma, f \rangle \rightarrow \langle \Sigma', f' \rangle$ tales que preservan operaciones y predicados.
- Multiplicación e Identidad usual
- Esto se puede representar dando el funtor MOD:
  MOD: $\boldsymbol{Sign} \rightarrow \boldsymbol{Cat}^{op}$
  utilizando la notación $MOD(H) = H^b$
  H permite ver un Σ'-modelo M' como un Σ-modelo $H^b(M')$ con:

$$H^b : MOD(\Sigma') \rightarrow MOD(\Sigma).$$

- Σ asociamos a ella la categoría $\mathbf{Mod}(\Sigma)$
- $Ob(\mathbf{Mod}(\Sigma)) = \{\mathcal{M}|\mathcal{M} = \langle\Sigma, f\rangle\}$
- $Mor(\mathbf{Mod}(\Sigma)) = H : \langle\Sigma, f\rangle \rightarrow \langle\Sigma', f'\rangle$ tales que preservan operaciones y predicados.
- Multiplicación e Identidad usual
- Esto se puede representar dando el funtor MOD:
  MOD: $\boldsymbol{Sign} \rightarrow \boldsymbol{Cat}^{op}$
  utilizando la notación $MOD(H) = H^b$
  H permite ver un Σ'-modelo M' como un Σ-modelo $H^b(M')$ con:

$$H^b : MOD(\Sigma') \rightarrow MOD(\Sigma).$$

- Σ asociamos a ella la categoría $\mathbf{Mod}(\Sigma)$
- $Ob(\mathbf{Mod}(\Sigma)) = \{\mathcal{M}|\mathcal{M} = \langle\Sigma, f\rangle\}$
- $Mor(\mathbf{Mod}(\Sigma)) = H : \langle\Sigma, f\rangle \to \langle\Sigma', f'\rangle$ tales que preservan operaciones y predicados.
- Multiplicación e Identidad usual
- Esto se puede representar dando el funtor MOD:
  MOD: ***Sign*** $\to$ ***Cat***^{op}
  utilizando la notación $MOD(H) = H^b$
  H permite ver un Σ'-modelo M' como un Σ-modelo $H^b(M')$ con:
  $$H^b : MOD(\Sigma') \to MOD(\Sigma).$$

Una institución es una cuádrupla $I = \langle \mathbf{Sign}, \ SEN, \ MOD, \ \models \rangle$, donde:

- **Sign** es una categoría de signaturas.
- SEN: **Sign**$\rightarrow$ **Set** un funtor de fórmulas.
- Una función
  $\models : Ob(\mathbf{Sign}) \rightarrow \wp(Ob(\mathbf{MOD}(\Sigma)) \times SEN(\Sigma))$
  que asocia a una signatura Σ una relación de satisfacción
  $\models_\Sigma \subseteq Ob(MOD(\Sigma)) \times SEN(\Sigma)$
  tal que para cada
  $M' \in MOD(\Sigma), \ H : \Sigma \rightarrow \Sigma'$ y $\varphi \in SEN(\Sigma)$
  se cumple que:

$$H^b(M') \models_\Sigma \varphi \text{ si y sólo si } M' \models_{\Sigma'} H(\varphi)$$

Agenda

Una lógica es una quíntupla
$L = \langle \mathbf{Sign}, SEN, MOD \vdash, \models \rangle$
donde:

- $\langle \mathbf{Sign}, SEN, \vdash \rangle$ es una π-institución,
- $\langle \mathbf{Sign}, SEN, MOD, \models \rangle$ es una institución
  y para cada
  $\Sigma \in Ob(Sign)$, $\Gamma \subseteq SEN(\Sigma)$ y $\varphi \in SEN(\Sigma)$:
  se satisface que:

$$\textit{Si } \Gamma \vdash_\Sigma \varphi \textit{ entonces } \Gamma \models_\Sigma \varphi$$

que es una condición de correctud. Una lógica es completa si adicionalmente

$$\Gamma \vdash_\Sigma \varphi \to \Gamma \models_\Sigma \varphi$$

Cálculo de Pruebas

- Idea Básica relacionar T con una estructura teórico probatoria $P(T)$
- Abstracción: Para un cálculo de pruebas dado existe una categoría **Str** que tiene como bojetos tales estructuras y un funtor $P : \mathit{Th}_0 \to \mathit{Str}$ que asigna a cada teoría T su estructura de pruebas $P(T)$
- Una multicategoría consiste de un conjunto $\mathcal{O}$ de objetos básicos junto con una categoría **C** cuyos objetos son cadenas finitas $\Gamma = A_1, \ldots, A_n$ de elementos de $\mathcal{O}$.

Cálculo de Pruebas

- Idea Básica relacionar T con una estructura teórico probatoria $P(T)$
- Abstracción: Para un cálculo de pruebas dado existe una categoría **Str** que tiene como bojetos tales estructuras y un funtor $P : \boldsymbol{Th_0} \to \boldsymbol{Str}$ que asigna a cada teoría T su estructura de pruebas $P(T)$
- Una multicategoría consiste de un conjunto $\mathcal{O}$ de objetos básicos junto con una categoría $\mathbf{C}$ cuyos objetos son cadenas finitas $\Gamma = A_1, \ldots, A_n$ de elementos de $\mathcal{O}$.

Cálculo de Pruebas

- Idea Básica relacionar T con una estructura teórico probatoria $P(T)$
- Abstracción: Para un cálculo de pruebas dado existe una categoría **Str** que tiene como bojetos tales estructuras y un funtor $P : \boldsymbol{Th_0} \to \boldsymbol{Str}$ que asigna a cada teoría T su estructura de pruebas $P(T)$
- Una multicategoría consiste de un conjunto $\mathcal{O}$ de objetos básicos junto con una categoría **C** cuyos objetos son cadenas finitas
  $\Gamma = A_1, \ldots, A_n$ de elementos de $\mathcal{O}$.

- Los morfismos de **C** tienen una estructura de monoide con la multiplicación de dos morfismos:
  $\alpha : \Gamma_1 \to \Delta_1$ y $\beta : \Gamma_2 \to \Delta_2$
  denotada α, β y definida de la forma:
  $\alpha, \beta : \Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2$
- La multiplicación α, β es un funtor:
  $(-, -) : \boldsymbol{C}^2 \to \boldsymbol{C}$,
  que preserva identidades y composición. Un homomorfismo entre multicategorías es un funtor estrictamente monoidal

- Los morfismos de **C** tienen una estructura de monoide con la multiplicación de dos morfismos:
  $\alpha : \Gamma_1 \to \Delta_1$ y $\beta : \Gamma_2 \to \Delta_2$
  denotada α, β y definida de la forma:
  $\alpha, \beta : \Gamma_1, \Gamma_2 \to \Delta_1, \Delta_2$
- La multiplicación α, β es un funtor:
  $(-, -) : \boldsymbol{C^2} \to \boldsymbol{C}$,
  que preserva identidades y composición. Un homomorfismo entre multicategorías es un funtor estrictamente monoidal

Ejemplo

- Lógica clásica proposicional con deducción natural como cálculo de pruebas:
- Dada una teoría $T = (\Sigma, \Delta)$ en $\mathcal{L}_{CP}$, asociamos a ella una multicategoría $P(T)$ con $SEN(\Sigma)$ como sus objetos básicos y sus morfismos:
  $\alpha : A_1, \ldots, A_n \to B_1, \ldots, B_m$
  consisten de secuencias $\alpha = \alpha_1, \ldots, \alpha_m$ con α_i un árbol de prueba en deducción natural de B_i cuyas hojas sólo tienen fórmulas contenidas en Δ.
- La identidad

$$id_{A_1 \ldots A_n} : A_1, \ldots, A_n \to A_1, \ldots, A_n$$

Ejemplo

- Lógica clásica proposicional con deducción natural como cálculo de pruebas:
- Dada una teoría $T = (\Sigma, \Delta)$ en $\mathcal{L}_{CP}$, asociamos a ella una multicategoría $P(T)$ con $SEN(\Sigma)$ como sus objetos básicos y sus morfismos:
  $\alpha : A_1, \ldots, A_n \to B_1, \ldots, B_m$
  consisten de secuencias $\alpha = \alpha_1, \ldots, \alpha_m$ con α_i un árbol de prueba en deducción natural de B_i cuyas hojas sólo tienen fórmulas contenidas en Δ.
- La identidad

$$id_{A_1 \ldots A_n} : A_1, \ldots, A_n \to A_1, \ldots, A_n$$

Ejemplo

- Lógica clásica proposicional con deducción natural como cálculo de pruebas:
- Dada una teoría $T = (\Sigma, \Delta)$ en $\mathcal{L}_{CP}$, asociamos a ella una multicategoría $P(T)$ con $SEN(\Sigma)$ como sus objetos básicos y sus morfismos:
  $\alpha : A_1, \ldots, A_n \to B_1, \ldots, B_m$
  consisten de secuencias $\alpha = \alpha_1, \ldots, \alpha_m$ con α_i un árbol de prueba en deducción natural de B_i cuyas hojas sólo tienen fórmulas contenidas en Δ.
- La identidad

$$id_{A_1 \ldots A_n} : \; A_1, \ldots, A_n \to A_1, \ldots, A_n$$

donde $A_1 \ldots A_n$ es vista como una secuencia de árboles de prueba.La composición, dados:

- $\alpha : A_1 \ldots A_n \to B_1 \ldots B_m$ y $\beta : B_1 \ldots B_m \to C_1 \ldots C_k$

es una secuencia:

$$\gamma = \gamma_1 \ldots \gamma_k$$

con γ_i el árbol de prueba obtenido del árbol β_i pegando el árbol α_j en cada presencia de B_j en las hojas.

- Se puede extraer de $P(T)$ el conjunto de todas las pruebas de los teoremas de T:
  $proofs(T) = \{\alpha : \varnothing \rightarrow \varphi \; en \; P(T) | \varphi \in SEN(T)\}$
  como el conjunto $Pr(P(T))$, donde Pr es un funtor,
  $Pr : \boldsymbol{MultiCat} \rightarrow \boldsymbol{Set}$
  que manda cada multicategoría $(\mathcal{O}, \boldsymbol{C})$ al conjunto:
  $Pr(\mathcal{O}, \boldsymbol{C}) = \{\alpha : \varnothing \rightarrow A \; en \; C | A \in \mathcal{O}\}$.
- El modo en que un cálculo de prueba y una π-institución se relacionan está dado por la función :
  $\pi_T : proofs(T) \rightarrow SEN(T)$
  que manda cada prueba $\alpha : \varnothing \rightarrow \varphi$ a su correspondiente teorema φ

- Se puede extraer de $P(T)$ el conjunto de todas las pruebas de los teoremas de T:
  $proofs(T) = \{\alpha : \varnothing \to \varphi \textit{ en } P(T) | \varphi \in SEN(T)\}$
  como el conjunto $Pr(P(T))$, donde Pr es un funtor,
  $Pr : \textbf{\textit{MultiCat}} \to \textbf{\textit{Set}}$
  que manda cada multicategoría $(\mathcal{O}, \boldsymbol{C})$ al conjunto:
  $Pr(\mathcal{O}, \boldsymbol{C}) = \{\alpha : \varnothing \to A \textit{ en } C | A \in \mathcal{O}\}$.

- El modo en que un cálculo de prueba y una π-institución se relacionan está dado por la función :
  $\pi_T : proofs(T) \to SEN(T)$
  que manda cada prueba $\alpha : \varnothing \to \varphi$ a su correspondiente teorema φ

Cálculo de pruebas

Un cálculo de pruebas es una séxtupla $P = \langle \textbf{\textit{Sign}}, SEN, \vdash, P, Pr, \pi \rangle$ con:

- $\langle \textbf{\textit{Sign}}, SEN, \vdash \rangle$ una π- institución,
- Un funtor $P : \textbf{\textit{Th}}_0 \to \textbf{\textit{Str}}$;
- $Pr : \textbf{\textit{Str}} \to \textbf{\textit{Set}}$ un funtor; para cada teoría T el conjunto $Pr(P(T))$ es llamado el conjunto de pruebas. Se denota por *proofs* el funtor compuesto $Pr \circ P : \textbf{\textit{Th}}_0 \to \textbf{\textit{Set}}$,
- $\pi : proofs \Rightarrow SEN$ una transformación natural tal que por cada teoría $T = (\Sigma, \Gamma)$ la imagen de $\pi_T : proofs(T) \to SEN(T)$ es el conjunto $Cn(\Gamma)$.

Lógica General

- Un sistema lógico es una óctupla;
  $S = \langle \mathbf{Sign}, SEN, MOD, \vdash, \models, P, Pr, \pi \rangle$, tal que:
  1. $\langle \mathbf{Sign}, SEN, MOD, \vdash, \models \rangle$ es una lógica,
  2. $\langle \mathbf{Sign}, SEN, \vdash, P, Pr, \pi \rangle$ es un cálculo de pruebas.

Lógica General

- Un sistema lógico es una óctupla;
  $S = \langle \mathbf{Sign}, SEN, MOD, \vdash, \models, P, Pr, \pi \rangle$, tal que:

  1. $\langle \mathbf{Sign}, SEN, MOD, \vdash, \models \rangle$ es una lógica,
  2. $\langle \mathbf{Sign}, SEN, \vdash, P, Pr, \pi \rangle$ es un cálculo de pruebas.

Lógica General

- Un sistema lógico es una óctupla;
  $S = \langle \mathbf{Sign}, SEN, MOD, \vdash, \models, P, Pr, \pi \rangle$, tal que:
  1. $\langle \mathbf{Sign}, SEN, MOD, \vdash, \models \rangle$ es una lógica,
  2. $\langle \mathbf{Sign}, SEN, \vdash, P, Pr, \pi \rangle$ es un cálculo de pruebas.

- Logic Framework
- Goguen y Brustall *Instituciones generalizadas*, con valores no booleanos para la relación de satisfacción
- *Instituciones contextuadas*, que manejan contextos de variables y sustituciones
- *Instituciones fibradas* o *de Grothendieck*, que combinan múltiples instituciones en una sola

- Logic Framework
- Goguen y Brustall *Instituciones generalizadas*, con valores no booleanos para la relación de satisfacción
- *Instituciones contextuadas*, que manejan contextos de variables y sustituciones
- *Instituciones fibradas* o *de Grothendieck*, que combinan múltiples instituciones en una sola

- Logic Framework
- Goguen y Brustall *Instituciones generalizadas*, con valores no booleanos para la relación de satisfacción
- *Instituciones contextuadas*, que manejan contextos de variables y sustituciones
- *Instituciones fibradas* o *de Grothendieck*, que combinan múltiples instituciones en una sola

- Logic Framework
- Goguen y Brustall *Instituciones generalizadas*, con valores no booleanos para la relación de satisfacción
- *Instituciones contextuadas*, que manejan contextos de variables y sustituciones
- *Instituciones fibradas* o *de Grothendieck*, que combinan múltiples instituciones en una sola