

ОПИСАНИЕ СЕМАНТИКИ ФОРМАЛЬНЫХ ЯЗЫКОВ МЕТОДОМ СЕМАНТИЧЕСКОЙ ИНДУКЦИИ

Выхованец В.С.

Институт проблем управления РАН

<http://valery.vykhovanets.ru>

План доклада

- 1 Синтаксис и семантика
- 2 Формальные языки
- 3 Методы описания семантики
- 4 Семантическое замыкание
- 5 Семантическая индукция
- 6 Примеры
- 7 Выводы

1 Семантика

- **Семантика** – отношение между языковыми выражениями (знаками) и миром, реальным или воображаемым (Энциклопедия Кругосвет)
- **Формальная семантика** – отношения между выражениями формальных языков и их интерпретацией в некоторой модели мира (Р. Монтегю)
- **Логическая семантика** – раздел металогики, изучающий интерпретации логических исчислений (Словарь общенаучных терминов)
- **Семантика формального языка** устанавливается путем присваивания значений его примитивным конструкциям (ГОСТ 34.320-96)

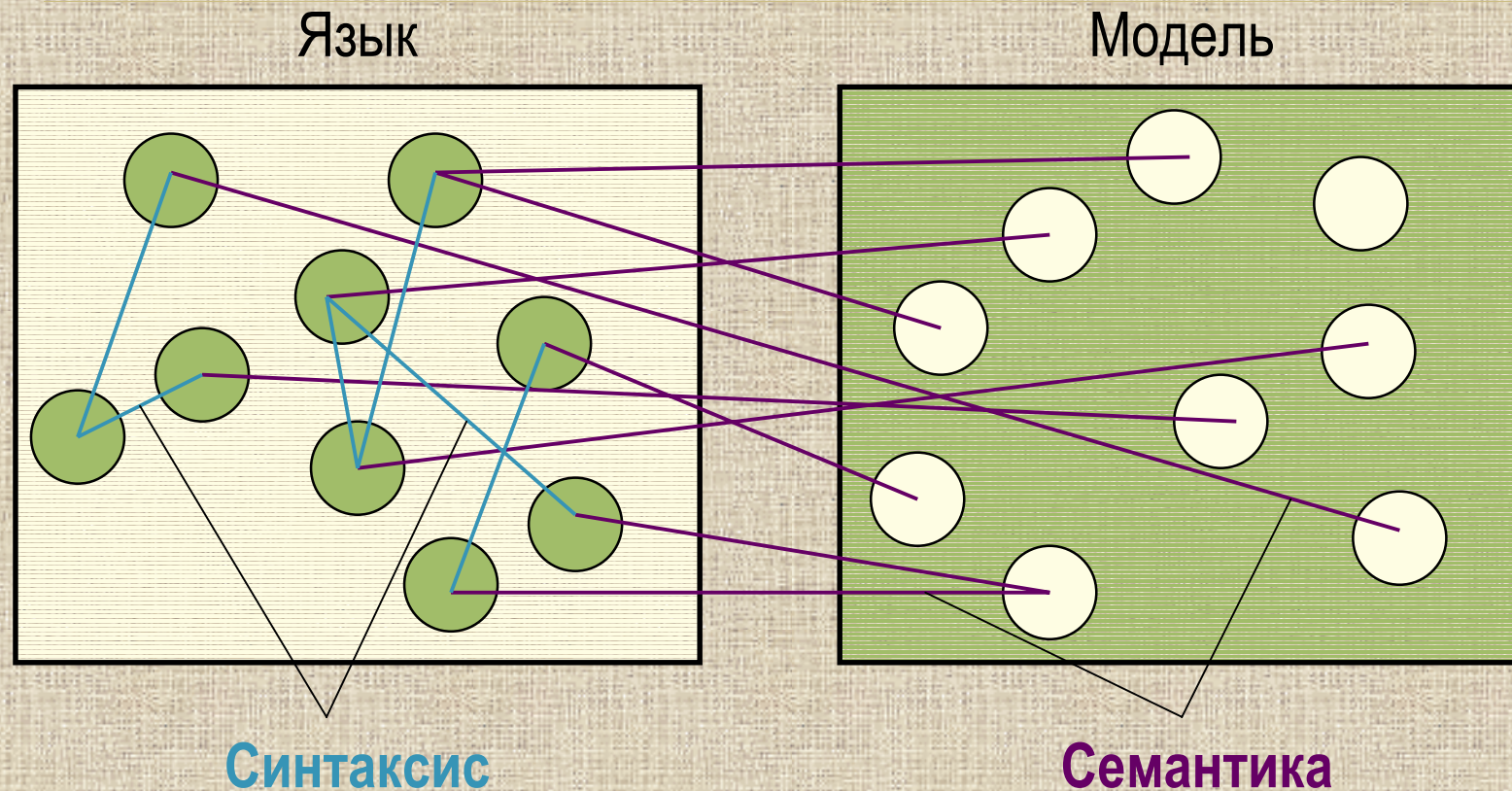
1 Принцип композиционности

- **Сложный знак** (текст) может быть понят на основе значения его частей и знания его структуры
- **Семантика целого** определяется через семантику частей, т.е. между семантическими правилами интерпретации и синтаксическими правилами образования выражений языка может быть установлено соответствие (Г. Фреге)
- **Примеры:**
 - «Семиотика – наука о знаках и знаковых системах»
 - «Мал золотник, да дорог»

1 Синтаксис

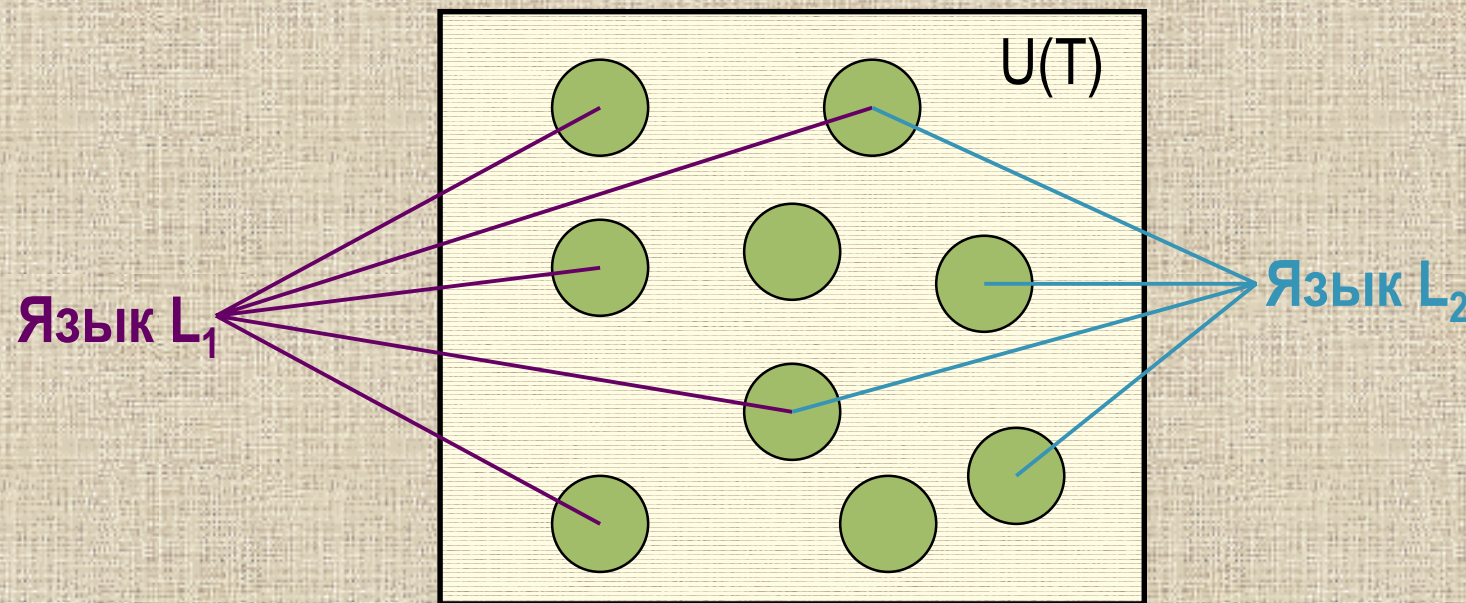
- **Синтаксис в языкознании** – совокупность правил языка, относящихся к построению единиц, более протяженных, чем слово (Энциклопедия Кругосвет)
- **Синтаксис в семиотике** – совокупность отношений между знаками языка (Ч. Моррис)
- **Логический синтаксис** – раздел металогики, изучающий структуру и свойства неинтерпретированных исчислений (Словарь общенаучных терминов)
- **Синтаксис формального языка** определяется посредством введения грамматических понятий... (ГОСТ 34.320-96)

1 Синтаксис и семантика



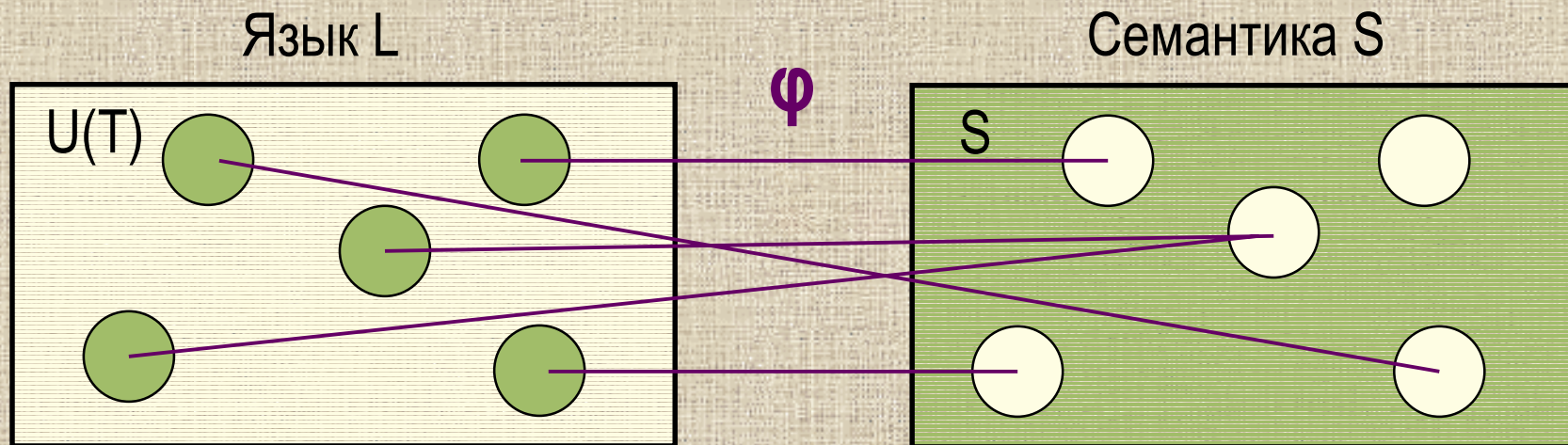
2 Формальный язык

- **Формальным языком** L над алфавитом T называется произвольное подмножество универсального множества U строк конечной длины в алфавите T (Н. Хомский, Дж. Миллер)



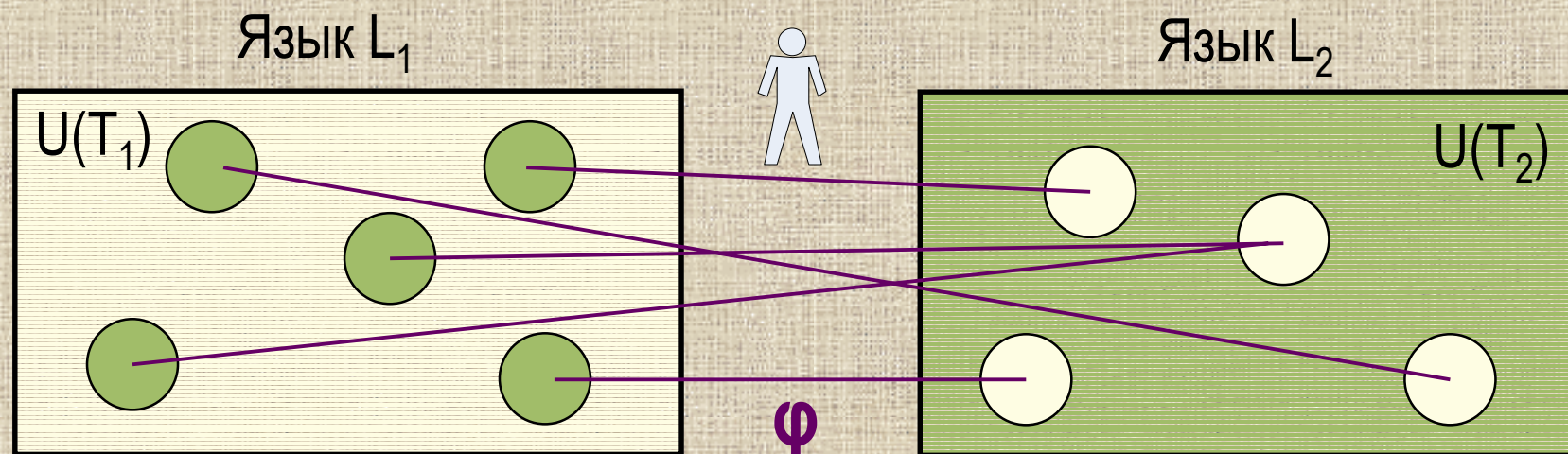
2 Семантика формального языка

- Множество S называется **формальной семантикой** формального языка L , если задано интерпретирующее отображение φ языка L в S ; тогда для строки $\alpha \in L$ элемент $\varphi(\alpha) \in S$ будет ее смыслом (О.П. Кузнецов)



2 Интерпретация

- **Интерпретация** – это сопоставление системе знаков объектов реального мира посредством мыслительного аппарата (Словарь терминов логики)
- **Интерпретация формальной теории** – это распространение положений этой теории на объекты реального мира или другой теории (Ж.-Л. Лорьер)



2 Интерпретация и семантика

- Чтобы описать семантику формального языка достаточно задать его **интерпретацию** в объекты реального мира или в другой язык
- Для описания семантики одного формального языка (**предметного языка**) необходимо определить отображение его строк в строки другого формального языка (**семантического языка**)
- **Принцип композиционности** позволяет упростить описание семантики, т.к. становится возможным использование конструктивных средств для сопоставления строк предметного языка и семантического языка

2 Синтаксис формальных языков

- **Формальные грамматики** – конечная форма задания формальных языков и описания их синтаксиса

$$G(L) = \langle T, N, P, I \rangle,$$

T – терминальный алфавит (элементарные знаки);

N – нетерминальный алфавит (множество понятий);

P – правила вывода (продукции) вида $\alpha \rightarrow \beta$;

I – аксиома (начальный нетерминальный знак), $I \in N$.

- **Пример:** язык двоичных чисел

$$G(L) = \langle \{0,1\}, \{I\}, P, I \rangle, \quad \left| \begin{array}{l} I \rightarrow 0, I \rightarrow 1I \rightarrow 10, \\ I \rightarrow 1I \rightarrow 10I \rightarrow 101, \dots \end{array} \right.$$
$$P = \{I \rightarrow 0 \mid 1 \mid 1I \mid 10I\}$$

3 Методы описания семантики

- **Перечисление** – сопоставление строк двух языков
- **Аксиоматический** – сопоставление продукциям грамматики формул аксиоматической теории, задаваемой аксиомами и правилами вывода
- **Алгебраический** – сопоставление продукциям грамматики формул алгебраической теории, задаваемой множеством и операциями на нем
- **Денотационный** – сопоставление продукциям грамматики функциональных денотатов (сложных функций)
- **Операционный** – сопоставление продукциям грамматики последовательности команд (операций)

3 Прямое перечисление

Семантический язык – **натуральные числа** в десятичной системе

$$N = \{0,1,2,3,\dots\}$$

$$0 \quad \{S \leftarrow 0\}$$

$$1 \quad \{S \leftarrow 1\}$$

$$10 \quad \{S \leftarrow 2\}$$

$$11 \quad \{S \leftarrow 3\}$$

$$100 \quad \{S \leftarrow 4\}$$

$$101 \quad \{S \leftarrow 5\}$$

$$110 \quad \{S \leftarrow 6\}$$

$$111 \quad \{S \leftarrow 7\}$$

...

3 Аксиоматическая семантика

Семантический язык – **формальная арифметика**

$\exists 0$

$\forall a \exists a'$

$\forall a (\neg(a' = 0))$

$\forall a (a + 0 = a)$

$\forall a (a * 0 = 0)$

$\forall a \forall b (a + b' = (a + b)')$

$\forall a \forall b (a * b' = a * b + a)$

$\forall a \forall b (a = b \rightarrow a' = b')$

$\forall a \forall b (a' = b' \rightarrow a = b)$

$\forall a \forall b \forall c (a = b \rightarrow (a = c \rightarrow b = c))$

$\varphi(0) \wedge \forall a (\varphi(a) \rightarrow \varphi(a')) \rightarrow \forall b (\varphi(b))$

$G(L) = \langle \{0,1\}, \{I\}, P, I \rangle, P = \{I \rightarrow 0 \mid 1 \mid I0 \mid I1\}$

$I \rightarrow 0 \quad \{S \leftarrow 0\}$

$I \rightarrow 1 \quad \{S \leftarrow 0'\}$

$I \rightarrow I0 \quad \{S \leftarrow (S) * (0' + 0')\}$

$I \rightarrow I1 \quad \{S \leftarrow (S) * (0' + 0') + 0'\}$

$I \rightarrow I1 \rightarrow 11$

$S \leftarrow 0'$

$S \leftarrow (0') * (0' + 0') + 0'$

$S \leftarrow 0' * 0' + 0' * 0' + 0'$

3 Алгебраическая семантика

Семантический язык – **алгебра натуральных чисел**

$$A = \langle N, +, * \rangle$$

$$G(L) = \langle \{0,1\}, \{I\}, P, I \rangle, P = \{I \rightarrow 0 \mid 1 \mid I0 \mid I1\}$$

$$I \rightarrow 0 \quad \{S \leftarrow 0\}$$

$$I \rightarrow 1 \quad \{S \leftarrow 1\}$$

$$I \rightarrow I0 \quad \{S \leftarrow (S) * 2\}$$

$$I \rightarrow I1 \quad \{S \leftarrow (S) * 2 + 1\}$$

$$I \rightarrow I1 \rightarrow 11$$

$$S \leftarrow 1$$

$$S \leftarrow (1) * 2 + 1$$

$$S \leftarrow 3$$

3 Денотационная семантика

Семантический язык – λ -исчисление Черча

$$F = \langle f, g \rangle, \quad gx = x * 2, \quad fx = x + 1$$

$$G(L) = \langle \{0,1\}, \{I\}, P, I \rangle, \quad P = \{I \rightarrow 0 \mid 1 \mid I0 \mid I1\}$$

$$I \rightarrow 0 \quad \{S \leftarrow 0\}$$

$$I \rightarrow 1 \quad \{S \leftarrow 1\}$$

$$I \rightarrow I0 \quad \{S \leftarrow \lambda S.g(S)\}$$

$$I \rightarrow I1 \quad \{S \leftarrow \lambda S.fg(S)\}$$

$$I \rightarrow I1 \rightarrow 11$$

$$S \leftarrow 1$$

$$S \leftarrow \lambda S.fgS$$

$$S \leftarrow 3$$

3 Операционная семантика

Семантический язык – **команды процессора**

$$K = \{mov\ A, B; shl\ A, B; inc\ A\}$$

$$G(L) = \langle \{0,1\}, \{I\}, P, I \rangle, P = \{I \rightarrow 0 \mid 1 \mid I0 \mid I1\}$$

$$I \rightarrow 0 \quad \{S \leftarrow mov\ V, 0\}$$

$$I \rightarrow 1 \quad \{S \leftarrow mov\ V, 1\}$$

$$I \rightarrow I0 \quad \{S \leftarrow S; shl\ V, 1\}$$

$$I \rightarrow I1 \quad \{S \leftarrow S; shl\ V, 1; inc\ V\}$$

$$I \rightarrow I1 \rightarrow 11$$

$$S \leftarrow mov\ V, 1$$

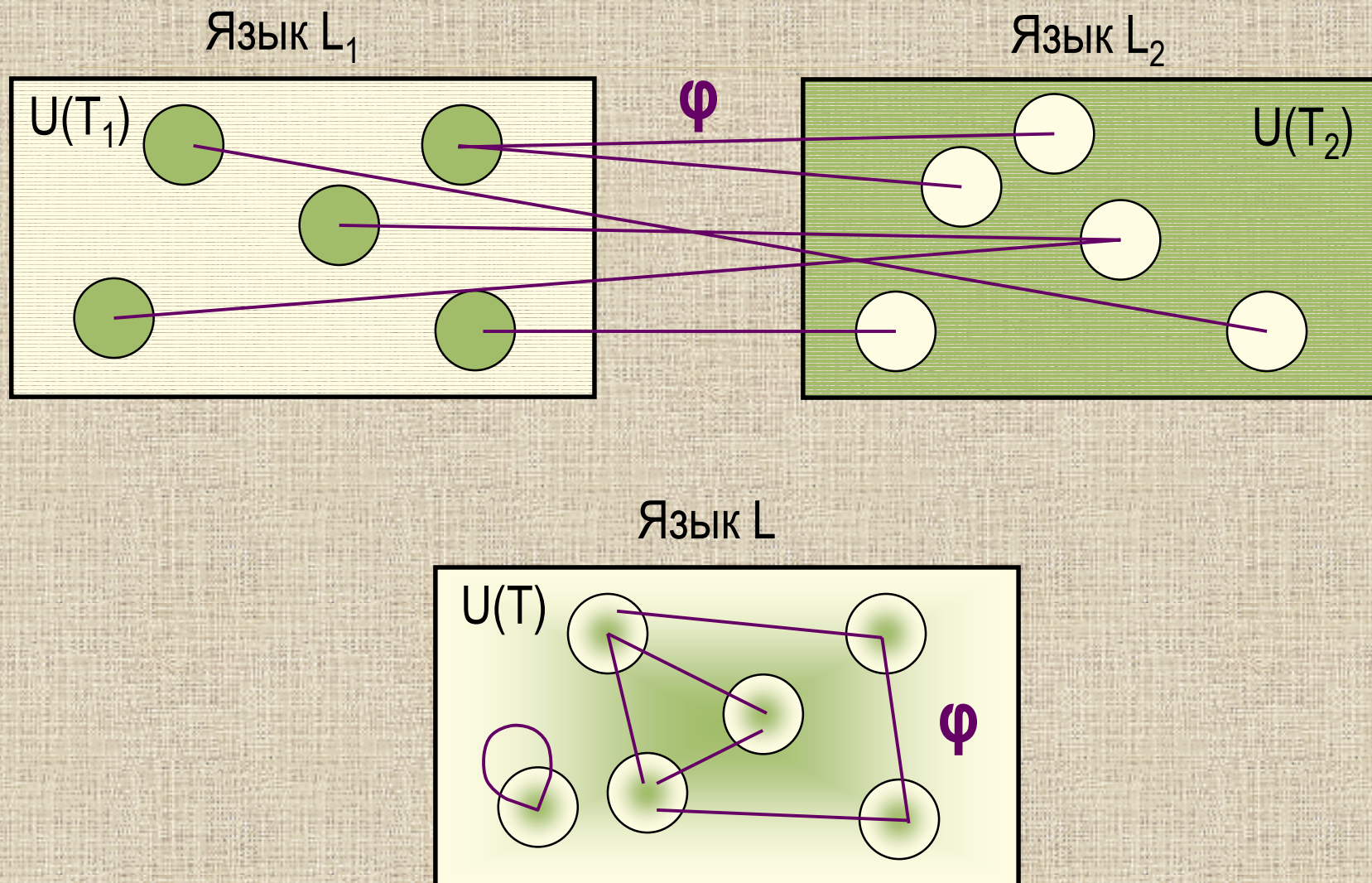
$$S \leftarrow mov\ V, 1; shl\ V, 1; inc\ V$$

$$V = 3$$

3 Особенности описания семантики

- Семантика формального языка – это **интерпретация** его строк с помощью другого языка
- Принцип **композиционности** реализуется путем использования аппарата формальных грамматик
- Фиксируется **конечное множество** семантических категорий, через которые описывается семантика
- Описание семантики осуществляется **конструктивными** (формальными) средствами
- Определяется только **одна семантика** (задается только одна интерпретация)

4 Суть подхода



4 Семантическое замыкание



- **Метаязык** – язык для выражения суждений о предметном языке (БСЭ)



4 Протоязык

- **Протоязык** – формальный язык с фиксированными синтаксисом и семантикой, предназначенный для определения синтаксиса проблемных языков

$$G(L) = \langle T, N, P, I \rangle$$

model	→	essences [model]
essences	→	'(' ')' notion '(' ')' [intension]
intension	→	sentence [intension]
sentence	→	syntax '{' '}'
syntax	→	item [syntax]
item	→	notion term
term	→	'" [terms] "'

$$P = \{ I \rightarrow 0 \mid 1 \mid I0 \mid I1 \}$$

() / ()

'0' {}

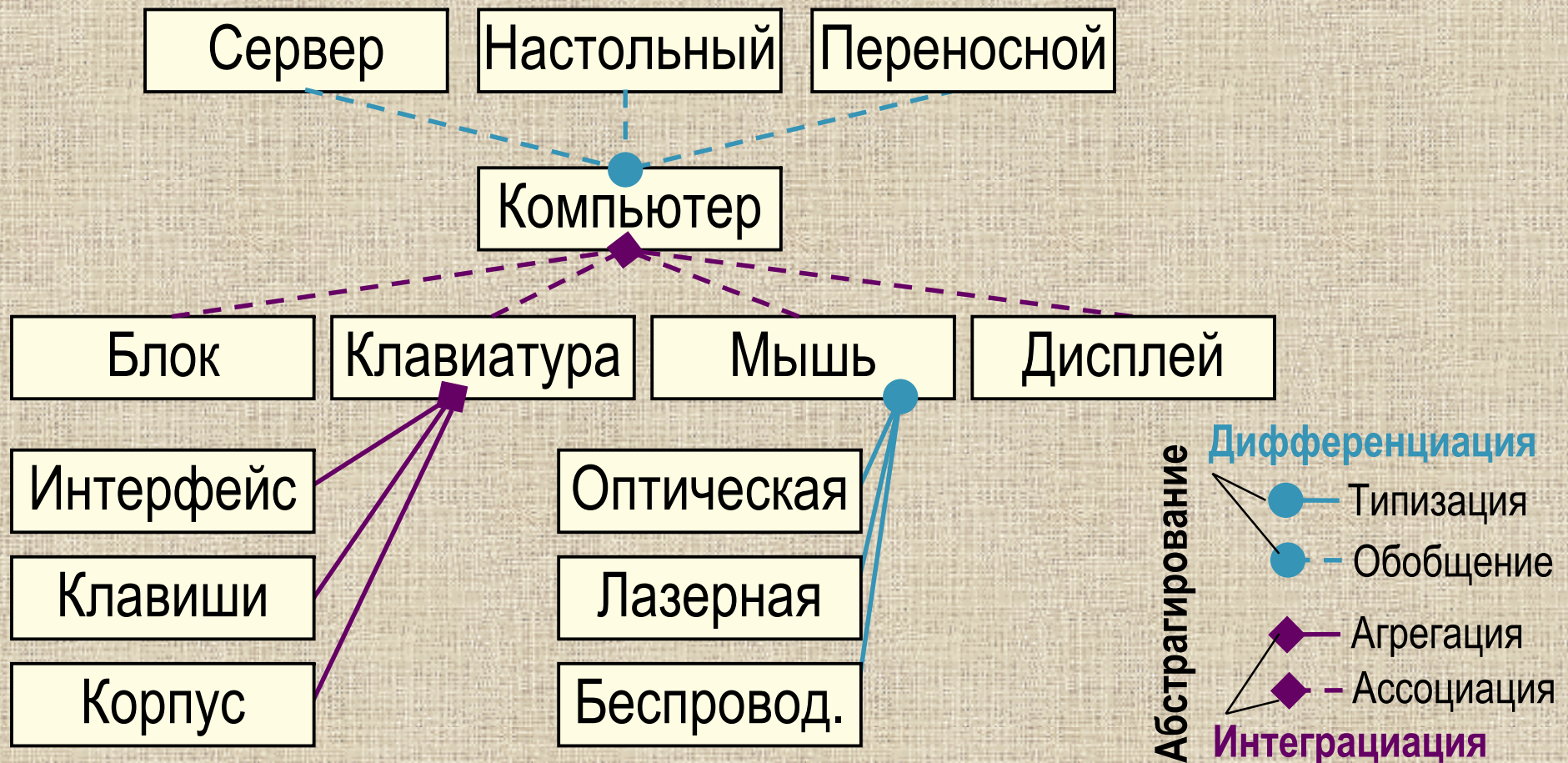
'1' {}

| '0' {}

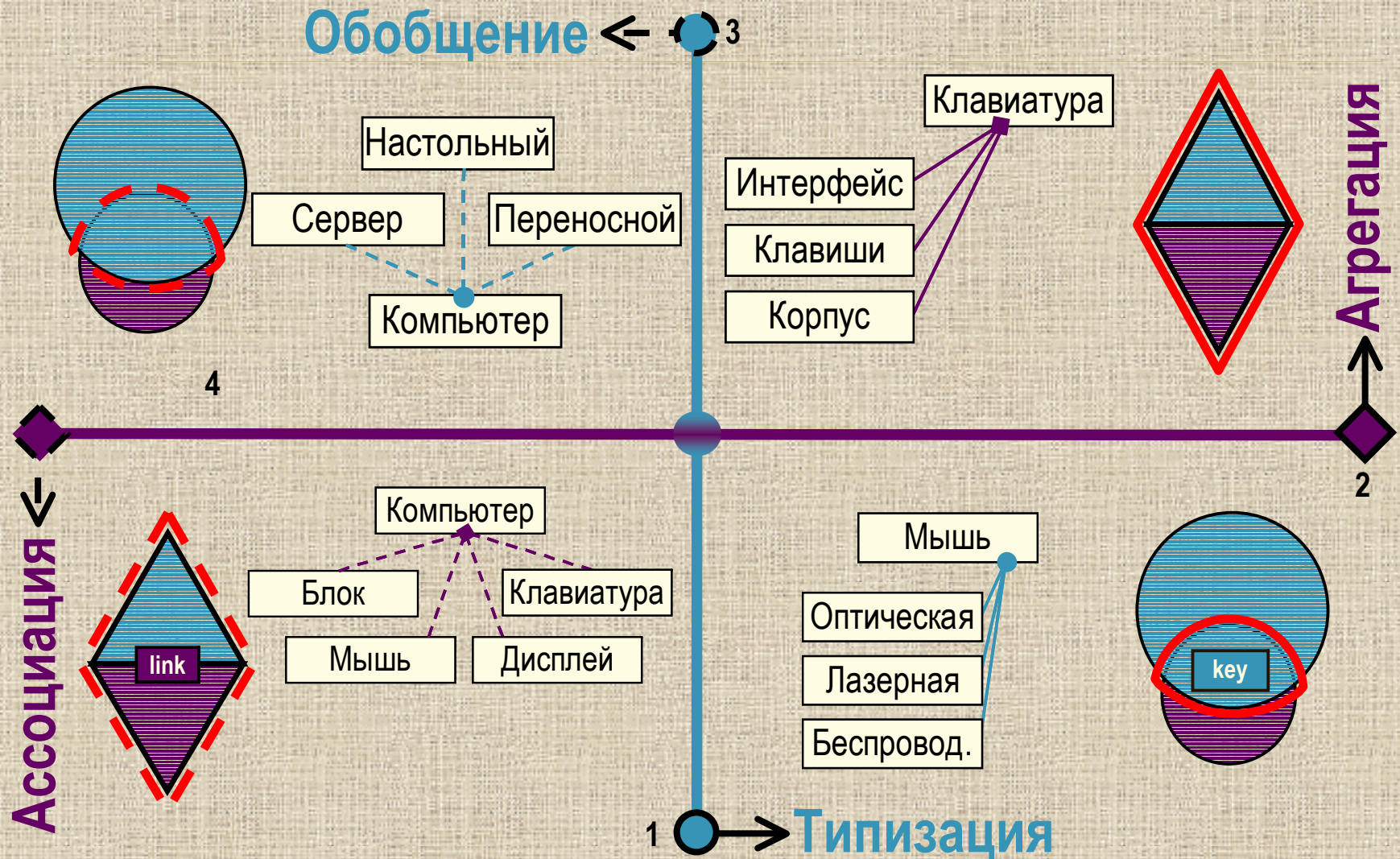
| '1' {}

4 Понятийная структура

- **Понятийная структура** – структура, описывающая понятия и их способы абстрагирования

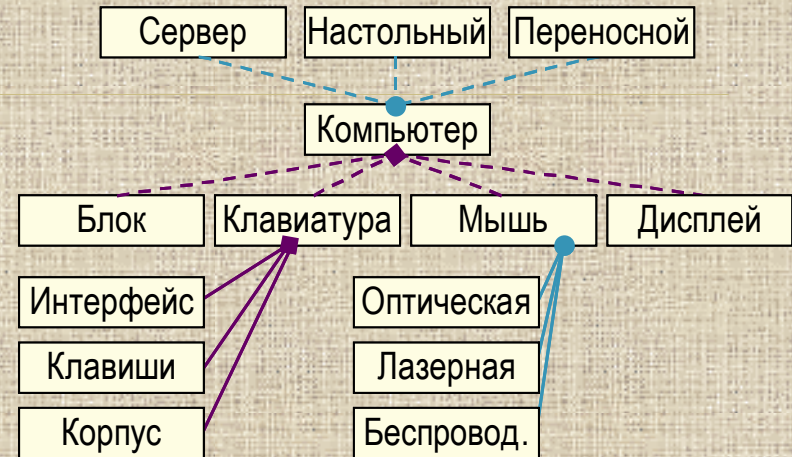


4 Абстрагирование



4 Онтологический язык

model → essences [model]
 essences → differentiation *notion*
 integration [intension]
 differentiation → '(' [notions] ')'
 integration → '(' [notions] ')'
 notions → **notion** [notions]



() *Сервер* () () *Настольный* () () *Переносной* ()
 () *Интерфейс* () () *Корпус* () () *Клавиши* ()
 () *Оптическая* () () *Лазерная* () () *Беспроводная* ()
 () *Блок* () () *Дисплей* ()
 (Оптическая Лазерная Беспроводная) *Мышь* ()
 () *Клавиатура* (Интерфейс Клавиши Корпус)
 (Сервер Настольный Переносной) *Компьютер*
 (Блок Клавиатура Мышь Дисплей)

4 Проблемный язык

- **Проблемный язык** – формальный язык, предназначенный для описания своей семантики, предметной области и решаемых в ней задач

cognition → model [situation] [cognition]

model → essences [model]

essences → differentiation *notion*
integration [intension]

intension → sentence [intension]

sentence → syntax semantic

syntax → item [syntax]

item → **notion** | '"' [*terms*] '"'

semantic → pragmatic [semantic]

pragmatic → [*aspect*] '{' [text] '}'

text → phrase [text]

phrase → **terms** | [**aspect**] '{' text '}'

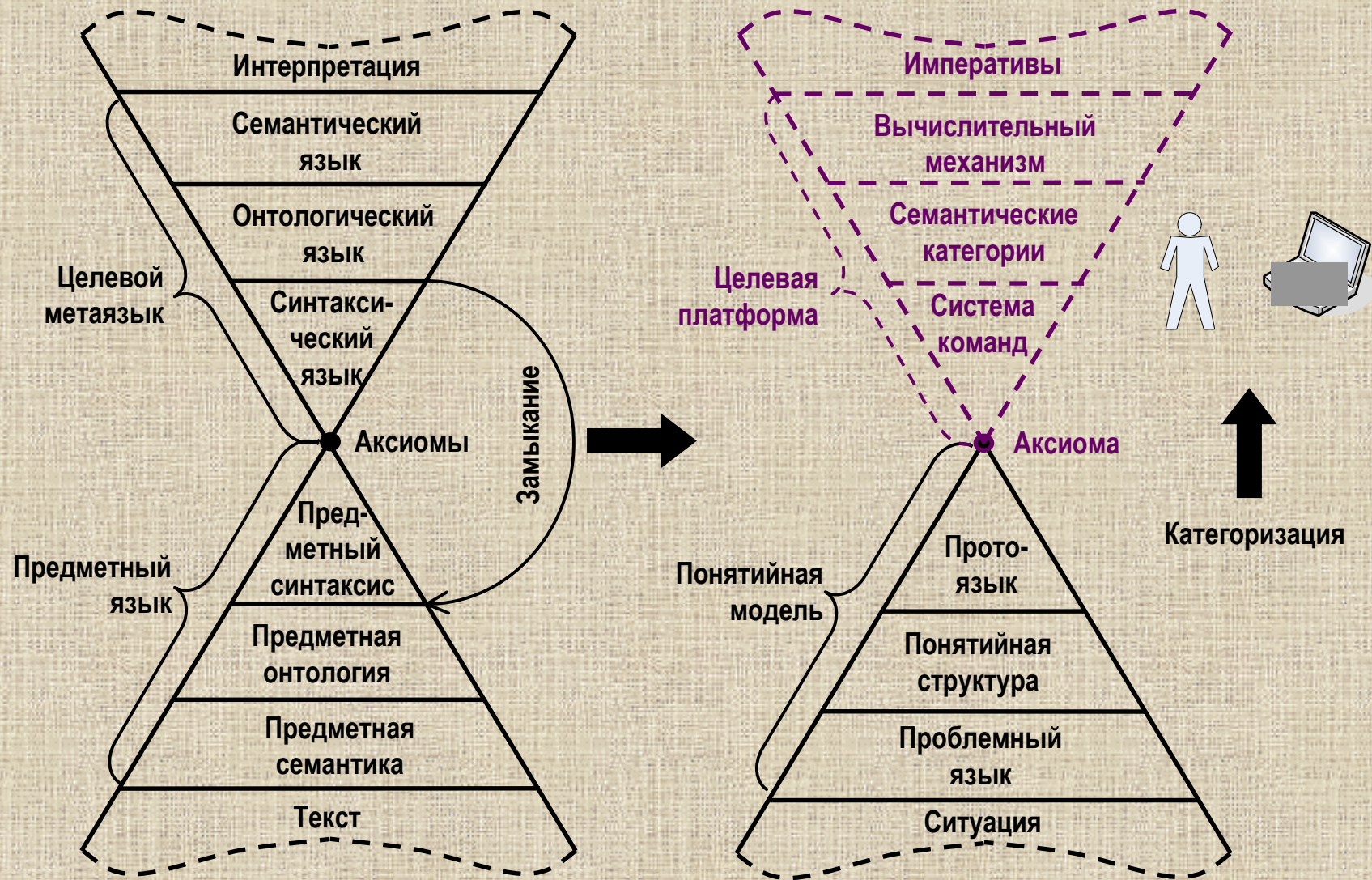
situation → [**aspect**] '<' [text] '>'

```
() Boolean ()  
'false' {...}  
'true' {...}  
"[A-Za-z][A-Za-z0-9]*" {...}  
'( Boolean )' {}  
'not' Boolean {...}  
Boolean 'and' Boolean {...}  
Boolean [a] 'or' Boolean [b]  
    { not (not a and not b) }  
< (not x or y) and z >  
fuzzy < not (x or y) and z >
```

5 Семантическая индукция

- **Семантическая индукция** – определение семантических категорий по мере необходимости, в процессе определения проблемного языка и средствами этого языка:
 - **база индукции** – первичные категории, которые непосредственно реализуются целевой вычислительной платформой и декларируются перед использованием;
 - **предположение индукции** – все ранее определенные семантические категории, выраженные на проблемном языке;
 - **индуктивный переход** – описание семантики нового или уже существующего предложения на уже определенном до этого проблемном подязыке;
 - **заключение индукции** – определение новой или доопределение существующей семантической категории.

5 Категоризация



5 Аксиома

- **Аксиома** – «пустые» квадратные скобки реализуют запись в область императива того значения, которое задается текущим элементом предложения

```
() ()  
'#' "[0-9A-F][0-9A-F]" [] {}  
() Boolean ()  
'not' Boolean  
{ #58 #F7 #D0 #50 }  
Boolean 'or' Boolean  
{ #58 #5A #0B #C2 #50 }  
Boolean [a] 'imp' [b] Boolean  
{ not a or b }
```

```
() ()  
"' "[0-9A-Za-z\s,]+" [] {}  
() Boolean ()  
'not' Boolean  
{ `pop eax` `not eax`  
  `push eax` }  
Boolean 'or' Boolean  
{ `pop eax` `pop edx`  
  `or eax, edx` `push eax` }  
Boolean [a] 'imp' [b] Boolean  
{ not a or b }
```

6 Синтаксически управляемый перевод

() *Expression* (String)

"[0-9]+" [n] { n } dif { '0' }

'x' { 'x' } dif { '1' }

(' *Expression* ') [e]

{ '(' & e & ')' } dif { '(' & dif { e } & ')' }

'sin' (' *Expression* [e] ')

{ 'sin(' & e & ')' } dif { 'cos(' & e & ') * (' & dif { e } & ')' }

'cos' (' *Expression* [e] ')

{ 'cos(' & e & ')' } dif { '-sin(' & e & ') * (' & dif { e } & ')' }

'-' *Expression* [e]

{ '-' & e } dif { '-' & dif { e } }

Expression [e1] '*' *Expression* [e2]

{ e1 & '*' & e2 }

dif { '(' & e1 & '*' & dif { e2 } & '+' & dif { e1 } & '*' & e2 & ')' }

Expression [e1] "+|-"_o *Expression* [e2]

{ e1 & o & e2 } dif { '(' & dif { e1 } & o & dif { e2 } & ')' }

dif < sin(5*cos(x)) - x*x >

(cos(5*cos(x))*(5* - sin(x)*(1) + 0*cos(x)*(1)))-(x*1 + 1*x)

6 Атрибутные грамматики

(Number) *Integer* ()

"[0-9]" [d]

{ d }

Integer [i] "[0-9]" [d]

{ $i * 10 + d$ }

(Float) *Fraction* ()

"[0-9]" [d]

{ d }

"[0-9]" [d] *Fraction* [f]

{ $d + f / 10$ }

() *Fixed* (Float)

Integer [i]

{ i }

Integer [i] '.' *Fraction* [f]

{ $i + f / 10$ }

() *Expr* (Number)

(Number) *Const* ()

"[0-9]+"

{ c }

(Const) *Prim* ()

('(' *Expr* [e] ')')

{ e }

(Prim) *Mul* ()

Prim [p1] '*' *Prim* [p2]

{ $p1 * p2$ }

(Mul) *Add* ()

Mul [p1] '+' *Mul* [p2]

{ $p1 + p2$ }

() *Expr* ()

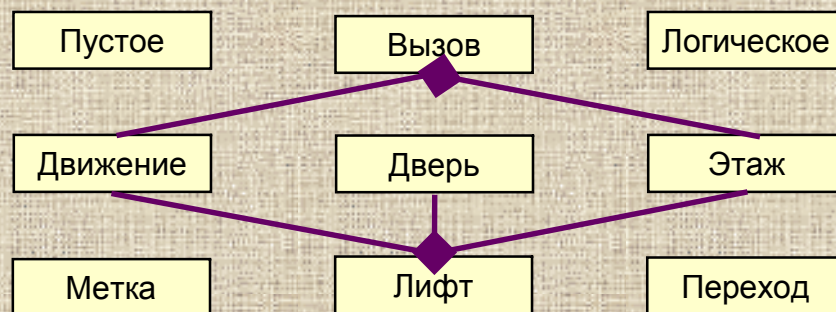
Add [a]

{ a }

6 Лифт (задача)

Дано многоэтажное здание с одним лифтом. На каждом этаже – кнопки для вызова лифта на движение вверх и вниз. В кабине имеется панель с кнопками для перемещения на один из девяти этажей.

Разработать систему управления лифтом.



1. Ожидание вызова

Если вызовов нет, то ожидать вызов (1). Если вызов со второго этажа, то перейти к открытию дверей (2), иначе – на принятие решения о движении (4).

2. Открытие дверей

Открыть дверь. Если дверь открылась, то перейти на принятие решения о движении (4), иначе повторить открытие двери (2).

3. Закрытие дверей

Закреть дверь. Если дверь не закрылась, то повторить закрытие двери (3), иначе перейти к определению направления движения (4).

4. Направление движения

Продолжение движения. Если лифт двигался вверх (вниз) и имеются вызовы на движение вверх (вниз), то начать движение вверх (5) (вниз 6). **Изменение направления.** Если вызовов на движение вверх (вниз) нет, но есть вызовы на движение вниз (вверх), то начать движение вниз (6) (вверх 5). **Возврат в начало.** Если вызовов нет и при этом лифт выше (ниже) второго этажа, то начать движение вниз (6) (вверх 5), иначе перейти в состояние ожидания (1).

5. Движение вверх

Начать движение вверх. Если при проходе этажа имеется вызов на движение вверх, то остановить лифт и открыть дверь (2), иначе продолжить движение вверх (5).

6. Движение вниз

Начать движение вниз. Если при проходе этажа имеется вызов для движения вниз, то остановить лифт и открыть дверь (2), иначе продолжить движение вниз (6).

6 Лифт (решение)

() **Движение** ()
 "[Дд]вижение" {...}

() **Дверь** ()
 "[Дд]верь" {...}

() **Этаж** ()
 'этаж' "[0-9]" {...}
 'этаж' 'вызова' "|вверх|вниз" {...}
 'этаж' 'лифта' {...}

() **Вызов (Этаж Движение)**
 'вызов' {...}

() **Лифт (Этаж Движение Дверь)**
 'лифт' {...}

() **Метка** ()
 "[А-Яа-я][А-Яа-я0-9]*" {...}

() **Переход** ()
 Метка {...}

() **Логическое** ()
 Этаж "выше|ниже|равен" **Этаж** {...
 Вызов "вверх|вниз|нет" {...}
 Дверь "открыта|закрыта" {...}
 'не' **Логическое** {...}

() ()
 '.' {...}
 Метка ':' {...}
 Движение "вверх|вниз|останов" {...}
 Дверь "открыть|закрыть" {...}
 'Если' **Логическое** ',' 'то' **Переход** {...}
 'Если' **Логическое** ',' 'то' **Переход** ','
 'иначе' **Переход** {...}

Ожидание:

Если вызов нет, то Ожидание.
Если этаж вызова равен этаж 2,
то Открыть, иначе Решение.

Открыть:

Дверь открыть.
Если дверь открыта, то Решение,
иначе Открыть.

Закрыть:

Дверь закрыть.
Если дверь закрыта, то Решение,
иначе Закрыть.

Решение:

Если лифт вверх и вызов вверх, то Вверх.
Если лифт вниз и Вызов вниз, то Вниз.
Если не вызов вверх и вызов вниз, то Вниз.
Если не вызов вниз и вызов вверх, то Вверх.
Если вызов нет и этаж лифта выше этаж 2,
то Вниз.
Если вызов нет и этаж лифта ниже этаж 2,
то Вверх, иначе Ожидание.

Вверх:

Движение вверх.
Если этаж лифта равен этаж вызова вверх,
то Открыть, иначе Вверх.

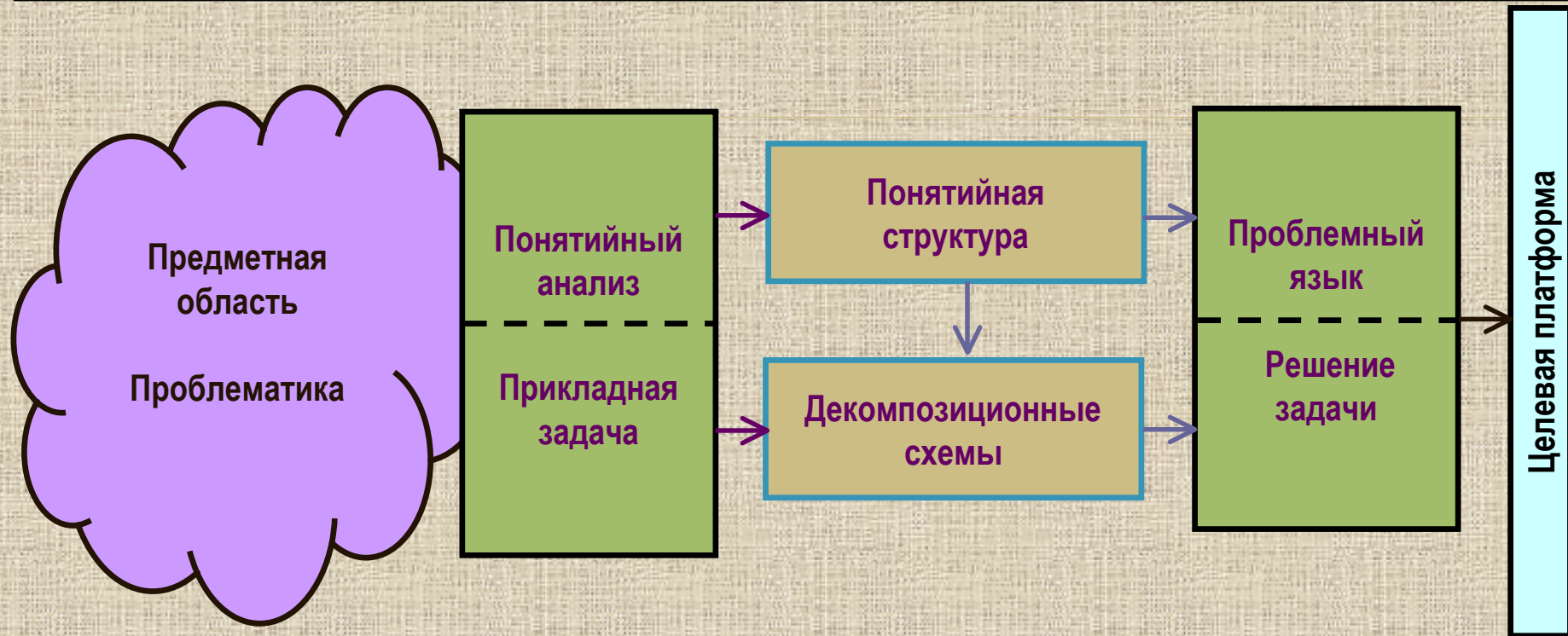
Вниз:

Движение вниз.
Если этаж лифта равен этаж вызова вниз,
то Открыть, иначе Вниз.

7 Выводы

- Предложена **методология** понятийного анализа, позволяющая получать семантически инвариантные понятийные структуры предметной области.
- Разработана **технология** представления и обработки знаний, сокращающая семантический разрыв между содержательными представлениями относительно предметной области и средствами их формальной спецификации.
- Решена **задача** определения семантики формальных языков на основе метода семантической индукции.

Методология*



Понятийный анализ – построение понятийной структуры предметной области: выявление признаков, сопоставление сущностей, образование понятий, описание структуры, уточнение абстракций, вычисление схем, верификация

* Слайд не демонстрировался

Понятие*

- Понятие**
 (сложный смысл)

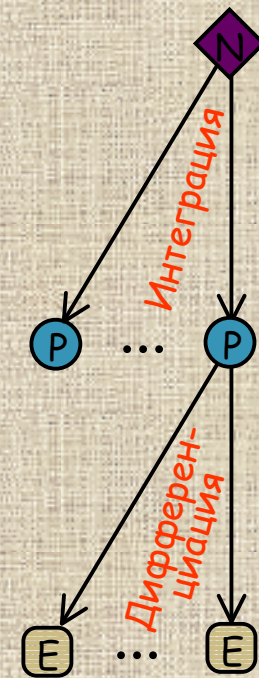
$$N = \begin{cases} \text{shm } N = (P^0, P^1, \dots, P^{n-1}); \\ \text{int } N = \{(P_j^0, P_j^1, \dots, P_j^{n-1}) \mid j = 0, 1, \dots\}; \\ \text{ext } N = \{N_0, N_1, \dots, N_{m-1}\}. \end{cases}$$

- Признак**
 (составной смысл)

$$P = \begin{cases} \text{shm } P = (P); \\ \text{int } P = \{(P_0), (P_1), \dots, (P_{u-1})\}; \\ \text{ext } P = \{E_0, E_1, \dots, E_{u-1}\}. \end{cases}$$

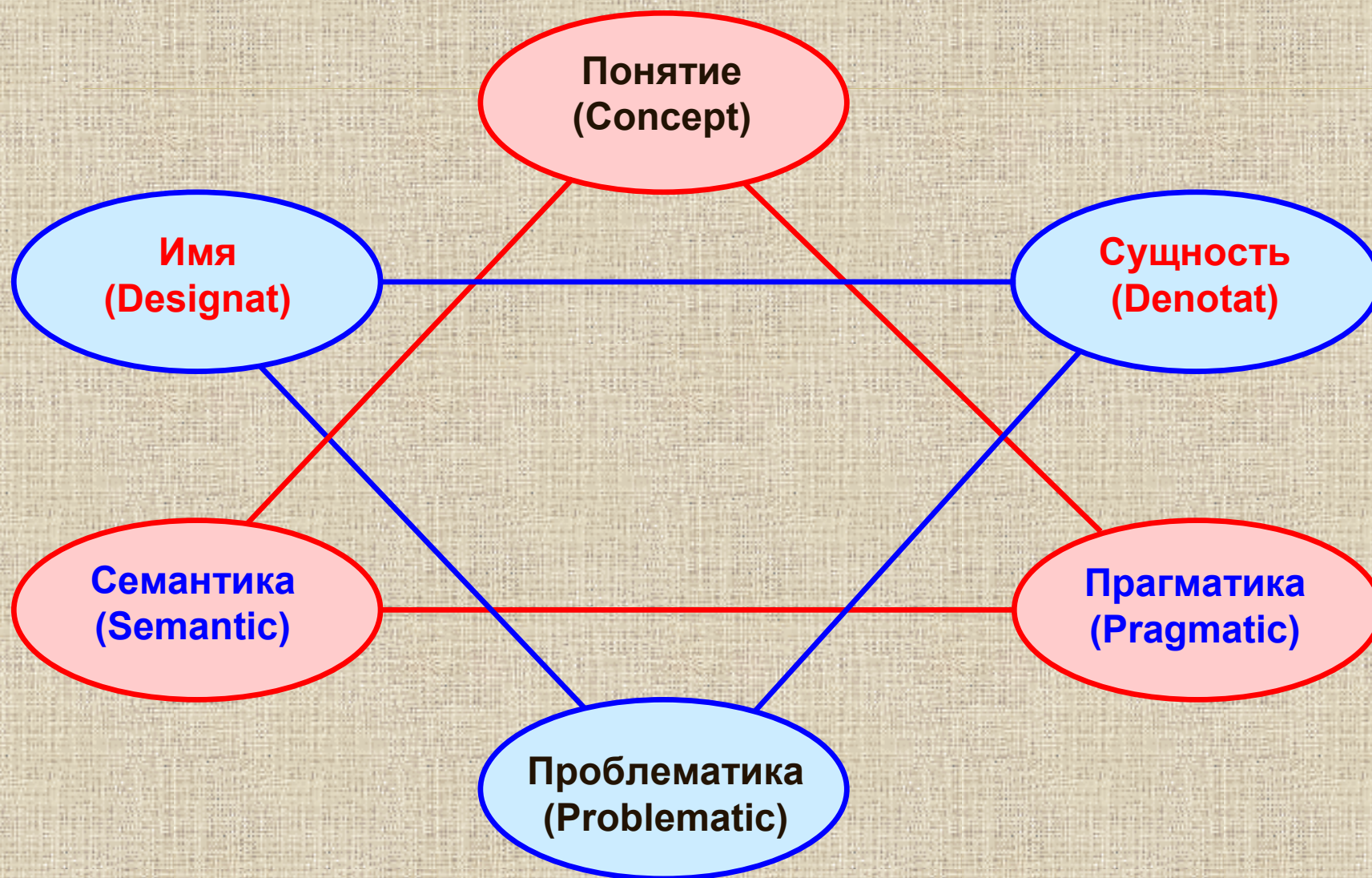
- Сущность**
 (простой смысл)

$$E = \begin{cases} \text{shm } E = (E); \\ \text{int } E = \{(E)\}; \\ \text{ext } E = \{E\}. \end{cases}$$



* Слайд не демонстрировался

Модель понятия*



* Слайд не демонстрировался

10 апреля 2008 г.

Семантика формальных языков

36

Методика*



Программа = Модель + Решения задачи

Модель = Структура + Синтаксис + Семантика

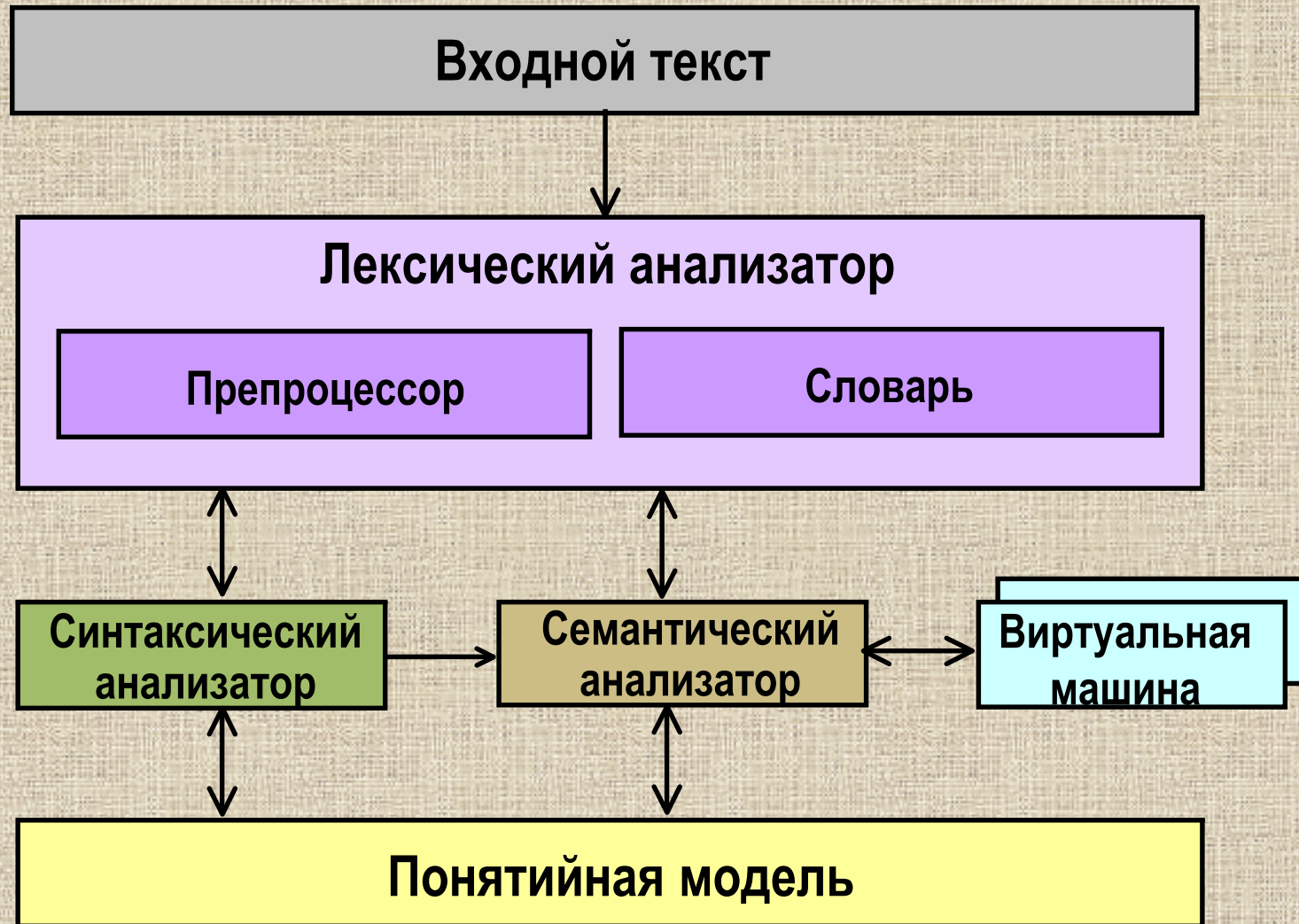
* Слайд не демонстрировался

10 апреля 2008 г.

Семантика формальных языков

37

Технология*



* Слайд не демонстрировался

10 апреля 2008 г.

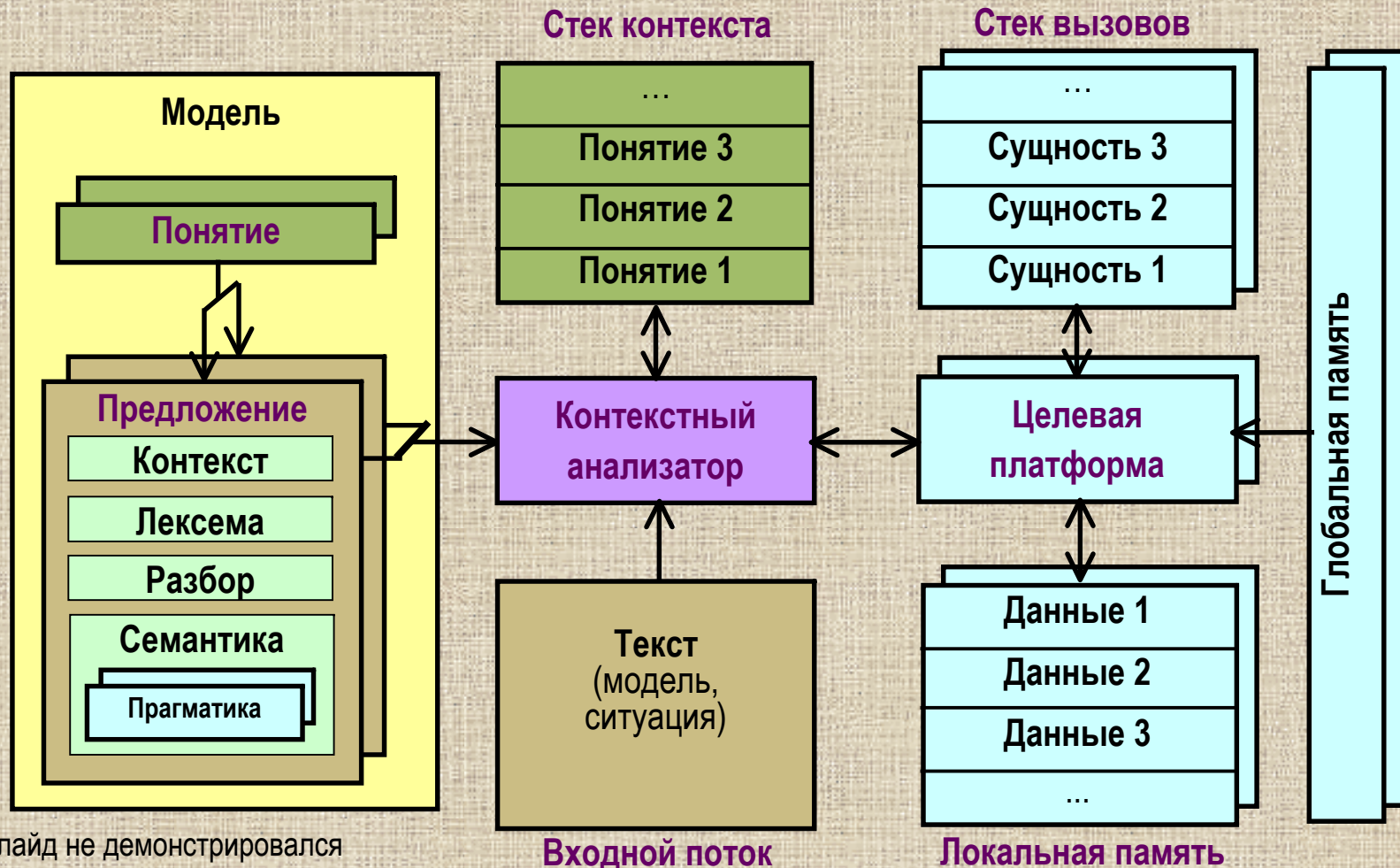
Семантика формальных языков

38

Разнесенный разбор*

Boolean [a] *'imp'* [b] **Boolean** { not a or b }

Контекст Лексема Разбор Императив



* Слайд не демонстрировался

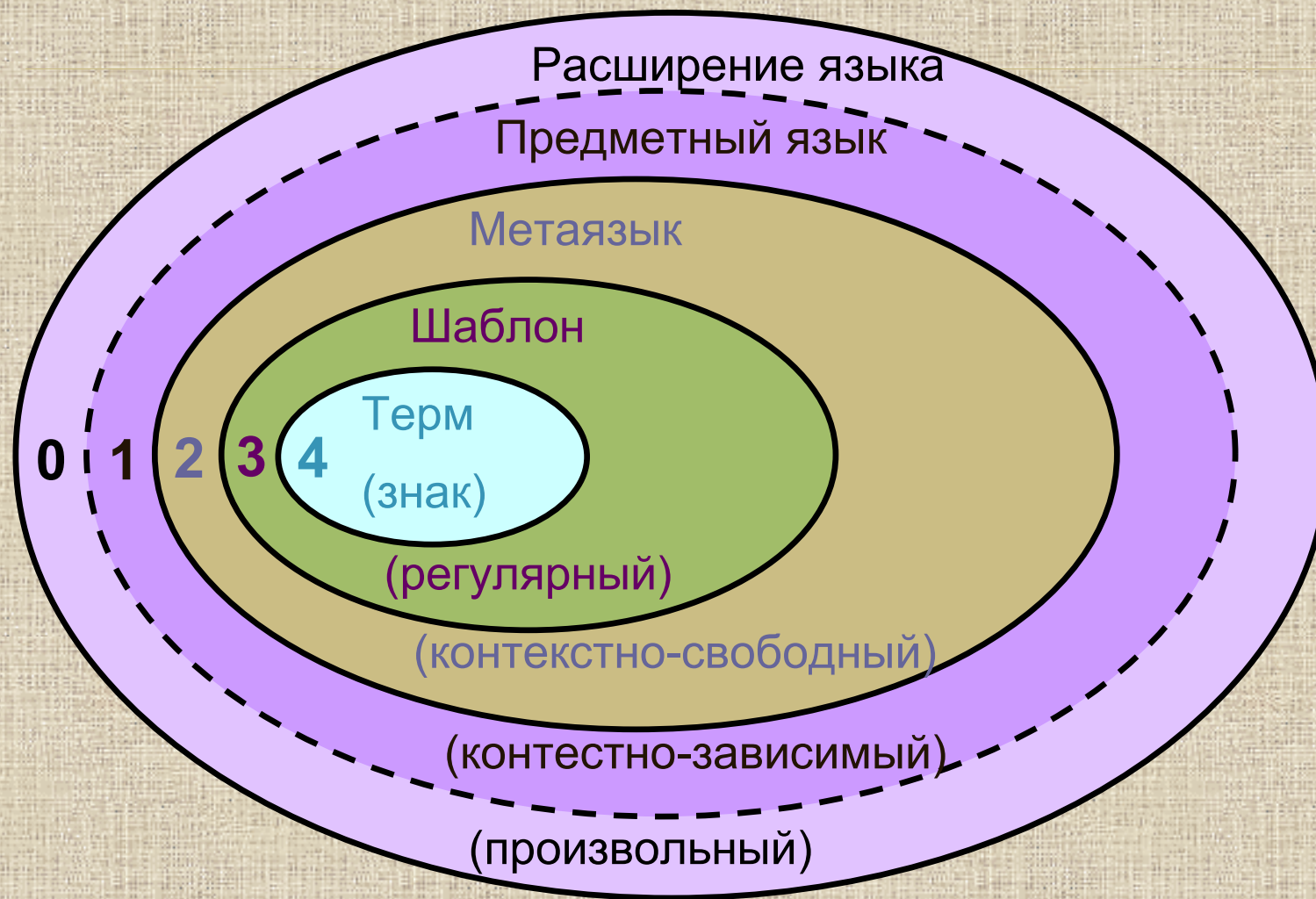
10 апреля 2008 г.

Входной поток

Семантика формальных языков

Локальная память

Иерархия языков*

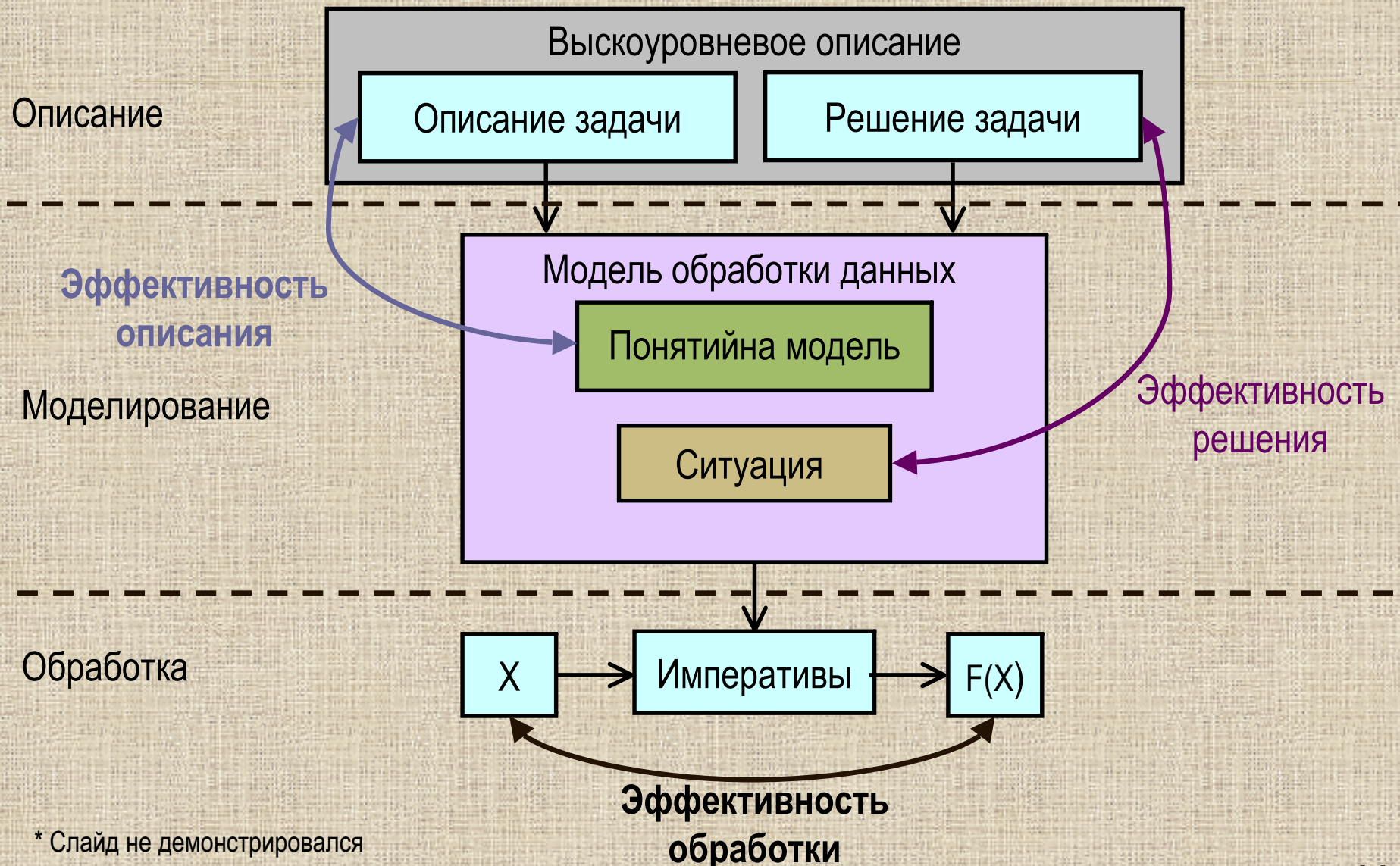


* Слайд не демонстрировался

10 апреля 2008 г.

Семантика формальных языков

Эффективность*



* Слайд не демонстрировался

10 апреля 2008 г.

Семантика формальных языков

Лифт (эффективность)*

Длина исходного текста, написанного экспертом на естественном языке, равна **1545** знакам, причем этот текст не содержит определение своего синтаксиса и не раскрывает семантику терминов и понятий, которые в нем использованы.

Длина понятийной модели и ситуационного описания на квазиестественном языке равна **1606** знакам, причем понятийная модель содержит описание синтаксиса, но не содержит описания семантики.

Предварительная эффективность реализации системы управления лифтом – **$1606/1545=1,04$** .

Итоговая эффективность. Если описание семантики будет реализовано с той же эффективностью, что и общее описание, то эффективность всего решения существенно не изменится.

Вывод: семантический разрыв между текстом исходного описания и текстом программы практически отсутствует.

* Слайд не демонстрировался