

Понятийный анализ и контекстная технология программирования

Выхованец Валерий Святославович, к.т.н. (ИРУ РАН)

1. Постановка задачи
2. Основные определения
3. Представление понятий
4. Понятийный анализ
5. Сравнение с объектно ориентированным анализом
6. Понятийная модель
7. Метаязык контекстной технологии
8. Система контекстного программирования

Постановка задачи

Конечной целью исследований в данной работе является сокращение семантического (смыслового) разрыва между содержательными представлениями о предметной области информационной системы и теми средствами, которые заложены в языке программирования, используемом для ее реализации.

Практическая значимость рассматриваемой проблемы обусловлена тем, что

– современные информационные технологии характеризуются возрастающей сложностью создаваемых на их основе информационных систем;

– проектирование информационных систем выполняется в основном на интуитивном уровне с применением неформализованных методов, основанных на искусстве проектировщиков, их практическом опыте, экспертных оценках и дорогостоящих экспериментальных проверках получаемых результатов;

– недостатком современных технологий программирования является несоответствие решаемых прикладных задач тем средствам, которые используются для их решения.

Научная новизна доклада состоит в том, что для сокращения семантического разрыва средствами контекстной технологии программирования для каждого класса решаемых задач создается специализированный язык, отражающий понятийную структуру предметной области.

Такой подход основывается на допущении, что уже в процессе изучения предметной области формируется система понятий, наиболее приспособленная для постановки и решения прикладных задач. Понятийный анализ определяется как методика построения понятийной структуры. Выявленная в результате понятийного анализа система понятий ложится в основу создаваемого языка программирования.

При использовании специализированного языка, создаваемого по контекстной технологии, ожидается получение более качественных и надежных программ.

Основные определения

Проблемная область – совокупность предметной области и решаемых на ней задач (проблем).

Предметная область – фрагмент реальной (виртуальной) действительности, представляемый совокупностью принадлежащих ему сущностей.

Сущность – уникальное устойчивое представление о выделенной части предметной области. Сущность воспринимается в виде своих признаков.

Признак – именованная сущность, характеризующаяся множеством своих проявлений (значений) и имеющая проблемную интерпретацию (семантическую роль).

Понятие – именованное множество сущностей, объединенных на основе общности своих признаков. Понятие задается схемой, интенционалом и экстенционалом.

Схема понятия – множество признаков, на которых определено понятие.

Интенционал или содержание понятия – множество значений взаимосвязанных признаков, позволяющих отличать сущности, принадлежащие понятию от других сущностей предметной области.

Экстенционал или объем понятия – множество сущностей, принадлежащих понятию.

Представление понятий, признаков и сущностей

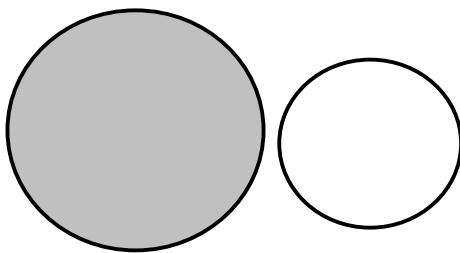
Понятие = Имя N + Схема Shm + Интенционал Int + Экстенционал Ext

$$N = \begin{cases} Shm(N) = \{P_1, P_2, \dots, P_S\}; \\ Int(N) = \{(V_1^i, V_2^i, \dots, V_S^i) \mid i = \overline{1, I}\}; V_j^i - \text{перечислимы или разрешимы} \\ Ext(N) = \{E_1, E_2, \dots, E_S\}. \end{cases}$$

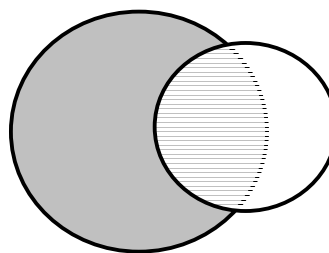
Признак – специализированное понятие, схема которого состоит из одного элемента, экстенционал (домен) задает синтаксические единицы, а интенционал (семантическая роль) определяет семантические единицы предметной области.

Сущность – понятие, имеющее единичный объем (терминальное понятие).

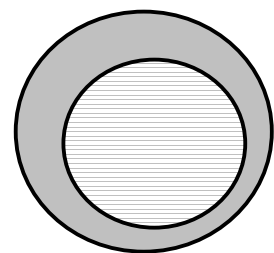
Пространство признаков



а) независимость



б) дифференциация



в) интеграция

Понятийный анализ

Понятийный анализ – методика построения понятийной структуры предметной области, основанная на абстрагировании понятий.

Абстрагирование – форма мышления, при которой образуются новые понятия на основе отражения общих и существенных признаков абстрагируемых понятий.

Виды абстракции – обобщение (специализация), агрегация (декомпозиция), типизация (конкретизация) и ассоциация (индивидуализация).

Обобщение – порождение понятия-обобщения на основе пересечения признаков обобщаемых понятий и объединения их экстенсионалов. При **специализации** понятия-обобщения выделяются обобщаемые им понятия.

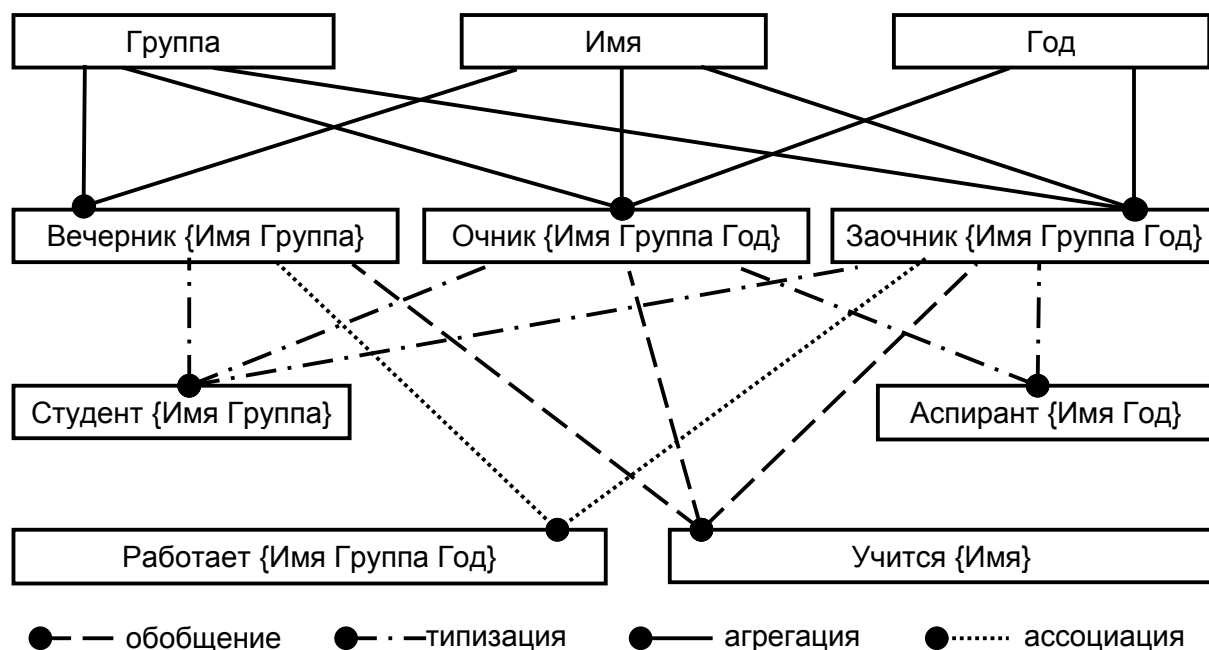
Агрегация – порождение понятия-агрегата на основе объединения признаков агрегируемых понятий и декартового произведения их экстенсионалов. При **декомпозиции** понятия-агрегата выделяются входящие в него агрегированные понятия.

Типизация – порождение понятия-типа на основе ограниченного пересечения признаков типизируемых понятий и объединения их экстенсионалов. При **конкретизации** понятия-типа выделяются типизируемые понятия, для чего используется общее подмножество признаков, называемое **ключом**.

Ассоциация – порождение понятия-ассоциации на основе ограниченного объединения признаков ассоциируемых понятий и декартового произведения их экстенсионалов. При **индивидуализации** понятия-ассоциации выделяются ассоциируемые с ним понятия, для чего используются подмножества признаков, называемые **связями**.

Понятийная структура

Понятийная структура предметной области задается множеством понятий, на котором определены отображения обобщения, агрегации, типизации и ассоциации.



Понятийный и объектный анализ

Понятийный анализ	Объектный анализ
Сущность:	Объект:
<ul style="list-style-type: none"> – обладает уникальностью (имя); – имеет признаки (синтаксис); – может быть описана (семантика). 	<ul style="list-style-type: none"> – обладает идентичностью; – имеет состояние; – проявляет поведение.
Связи между сущностями:	Связи между объектами:
<ul style="list-style-type: none"> – дифференциация признаков – интеграция признаков 	<ul style="list-style-type: none"> – связи (физич. или концепт. соединение) – агрегация (отношение часть-целое)
Понятие	Класс
Множество сущностей, объединенных на основе существенных признаков	Множество объектов, имеющих общую структуру и общее поведение
Структура понятия:	Спецификация класса:
<ul style="list-style-type: none"> – имя (уникальность), схема (признаки); – интенционал (содержание) – экстенционал (сущности) 	<ul style="list-style-type: none"> – имя (идентичность) – свойства (состояние) – методы (поведение)
Признак (элементарное понятие):	Свойство (атрибут)
<ul style="list-style-type: none"> – имя; – семантическая роль (интенционал); – домен (экстенционал). 	состояние объекта характеризуется перечнем всех свойств и текущими значениями каждого из этих свойств.
Взаимосвязь понятий:	Взаимосвязь классов:
<ul style="list-style-type: none"> – обобщение (есть некоторый) – агрегация (есть часть) – ассоциация (есть связь) – типизация (есть экземпляр) 	<ul style="list-style-type: none"> – общее и частное (is-a, наследование) – целое и часть (part-of, агрегация) – зависимость (association, ассоциация)
Обобщение (пересечение признаков):	Наследование:
понятие-обобщение содержит общие признаки обобщаемых понятий	один класс повторяет структуру и поведение другого класса
Агрегация (объединение признаков):	Агрегация:
понятие-агрегат включает все признаки агрегируемых понятий	отношения целого и части, приводящие к иерархии объектов
Ассоциация (ограничение агрегации):	Ассоциация:
понятие-ассоциация включает часть признаков ассоциируемых понятий	зависимость между классами, обеспечивающая переход от объектов одного класса к объектам другого
Типизация (ограничение обобщения):	
понятие-тип содержит часть общих признаков типизируемых понятий	
Понятийная структура:	Диаграммы классов:
структура, заданная на множестве понятий с отображениями:	– иерархия классов с отношениями наследования;
<ul style="list-style-type: none"> – обобщения; – агрегации; 	<ul style="list-style-type: none"> – типизации; – ассоциации.
	– иерархия объектов с отношениями агрегации.

Понятийная модель

Программа = Понятийная модель + Решение задачи (Ситуация)

Понятийная модель = Структура + Синтаксис + Семантика + Компиляция

Метаязык контекстной технологии	Специализированный язык
Описание понятийной структуры	Описание компиляции
Описание синтаксиса	Описание семантики
Привязка к целевой платформе, проверка контекстных условий	

Понятийная структура

Структура = (Дифференциация) *Имя* (Интеграция) [Содержание]

- 1 () Имя ()
- 2 () Год ()
- 3 () Группа ()
- 4 () Вечерник (Имя Группа)
- 5 () Очник (Имя Группа Год)
- 6 () Заочник (Имя Группа Год)
- 7 (Вечерник Очник Заочник) Студент ()
- 8 (Очник Заочник) Аспирант ()
- 9 (Вечерник Очник Заочник) Учится ()
- 10 () Работает (Вечерник Очник Заочник)

Синтаксис и семантика

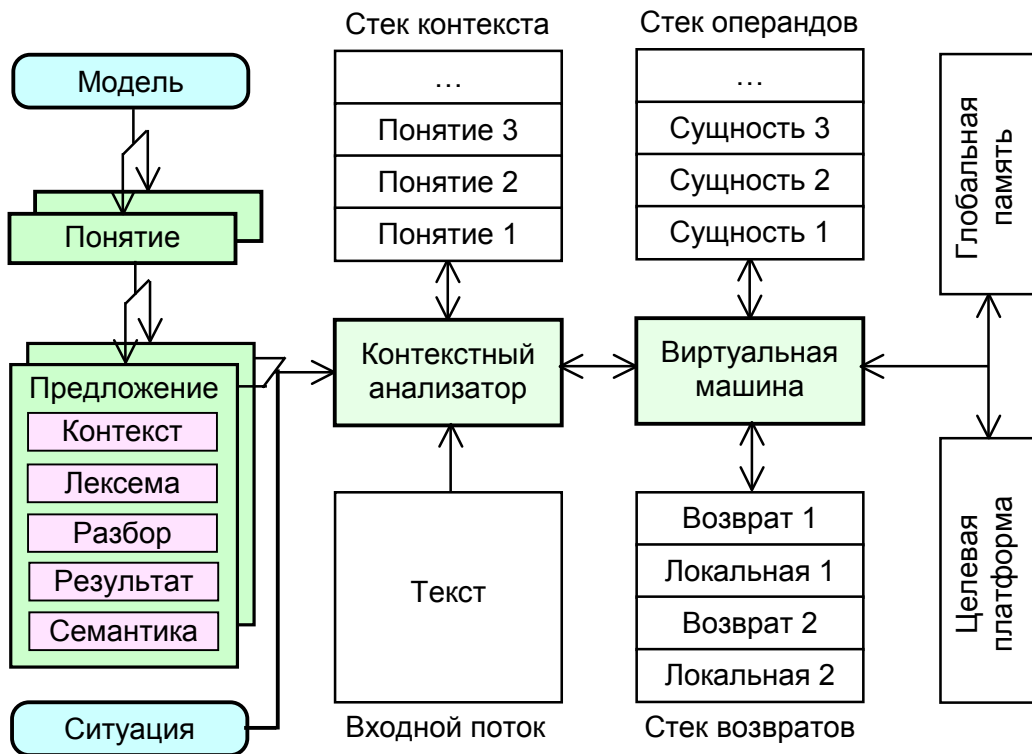
Синтаксис = Понятие | Лексема | Понятие [Лексема] Синтаксис

- 1 () Variable
- 2 "[A-Za-z][A-Za-z0-9]*" { ... }
- 3 (Variable) Boolean ()
- 4 'false' { mov eax, 0; push eax }
- 5 'true' { mov eax, -1; push eax }
- 6 Variable { pop ebx; mov eax, [ebx]; push eax }
- 7 '(' Boolean ')' { }
- 8 'not' Boolean { pop eax; not eax; push eax }
- 9 Boolean 'and' Boolean { pop eax; pop edx; and eax, edx; push eax }
- 10 Boolean 'or' Boolean { pop eax; pop edx; or eax, edx; push eax }
- 11 Boolean 'imp' Boolean { not %1 or %2 }

Метаязык КОНТЕКСТНОЙ ТЕХНОЛОГИИ

1	program	→	model [situation] [program]
2	model	→	essences essences model
3	essences	→	differentiation <i>notion</i> integration [description]
4	differentiation	→	'(' notions ')'
5	integration	→	'(' notions ')'
6	notions	→	[notion] notion [compile] notions
7	description	→	sentence sentence description
8	sentence	→	syntax semantic
9	syntax	→	notions [lexeme [items]]
10	lexeme	→	term pattern
11	term	→	" <i>terminals</i> "
12	pattern	→	"" <i>terminals</i> ""
13	items	→	item item items
14	item	→	notion [compile] lexeme [compile]
15	semantic	→	pragmatic pragmatic semantic
16	pragmatic	→	[<i>aspect</i>] '{' text '}'
17	text	→	phrase phrase text
18	phrase	→	terminals [<i>aspect</i>] '{' text '}'
19	compile	→	'[' text ']'
20	situation	→	'<' text '>'

Система контекстного программирования



- 1 **() Variable ()**
- 2 **"[A-Za-z][A-Za-z0-9]*" {}**
- 3 **() Constant ()**
- 4 **"false|true" {}**
- 5 **(Variable) Logic ()**
- 6 **Variable {}**
- 7 **(' Boolean ') {}**
- 8 **(Constant Logic) Negation ()**
- 9 **'not' Logic {}**
- 10 **(Negation) Conjunction ()**
- 11 **Negation 'and' Negation {}**
- 12 **(Conjunction) Disjunction ()**
- 13 **Conjunction 'or' Conjunction {}**

