

Smart-контракты в проектировании образовательной программы

Кубеков Б.С., Амиргалиев Е.Н., Утегенова А.У., Науменко В.В., Аленова Р.А.
Республика Казахстан, Университет Туран, Институт информационных и вычислительных технологий МОН РК, turpost_09@mail.ru (Кубеков Булат Сальмуханович, b.kubekov@mail.ru, +77017179800)

Постановка задачи, описание проблемы и т.п.

Формализация представления знаний в формате модели онтологии, в виде *знаниевых компонент*. Применение концепции smart-контракта для конфигурирования знаниевых компонент, в соответствии с сценарием обучения, и проектирование знаниевой архитектуры образовательной программы, удовлетворяющей индивидуальным предпочтениям и профессиональным потребностям обучающегося.

Ключевые определения:

- 1) Формализмы отображения знаний – *модель онтологии* $O_m = \langle C, R, F \rangle$;
- 2) Формализмы спецификации понятий онтологии: *отношения и свойства понятий онтологии*.
- 3) *Выражение знания*: $C_1 \leq *C_{1.1} (+C_1 \sim +C_2 * C_3 * C_4 + C_5) * C_{1.2} (*C_1 * C_2 * C_3) * C_{1.3}$;
- 4) *Знаниевый компонент КС*: композиция выражений знаний.
- 5) *Проектно-ориентированная технология обучения – CDIO*.
- 6) *Компетентностная модель этапа CDIO*: композиция профессиональных, базовых и дополнительных компетентностей.
- 7) *Выражение компетентностей*: $es.PrC.Conceive \leq *PrC_1 (*P_1 * P_2) * PrC_2 * PrC_3$;
- 8) *Smart-контракт* знаниевого компонента КС: $\{P\} < KC, Col, Lev > \{Q\}$

Достигнутые результаты

Параметры **smart-контракта** знаниевого компонента **КС**, позволяют рассматривать знаниевый компонент как элемент конфигурационного управления процессом проектирования *знаниевой архитектуры образовательной программы*.

Концептуально, знаниевая архитектура образовательной программы представляется в виде *взаимодействия* знаниевых компонент, реализация каждого из которых задается параметрами smart-контракта: {P}< КС, Col, Lev>{Q}

Введенная концепция smart-контракта позволяет:

- *во-первых*, использовать smart-контракт для моделирования знаниевых компонент прототипа образовательной программы;
- *во-вторых*, проводить уточнение и верификацию знаниевой архитектуры, на предмет соответствия требованиям обучающегося;
- *в - третьих*, производить оценку выражений знаний репозитория интерактивной образовательной Smart - среды;
- *в-четвертых*, производить мониторинг процесса обучения и успеваемости обучающегося;
- *в-пятых*, проводить анализ характеристик знаниевой архитектуры с целью оценки ее пригодности для обучения;
- *в-шестых*, находить проектные решения по формированию знаниевых компонент *повторного использования*.

Методы, приёмы, советы, практика

Разработана методика формирования знаниевых компонент, основанная на онтологическом инжиниринге и компетентностном подходе. Компетентностная модель, на примере этапа **Design** проектно-ориентированной технологии обучения – **CDIO**, представляется следующими **выражениями профессиональных, базовых и дополнительных** компетентностей:

$$ec.PrC.Design \leq *PrC_1 * PrC_2 * PrC_3;$$

$$ec.BaC.Design \leq *BaC_1 + BaC_2 * BaC_3;$$

$$ec.AdC.Design \leq *AdC_1 \sim *AdC_2 (*A_1 + A_2) \sim *AdC_3;$$

где, например, $PrC_1 \leq *C_1 (*C_1 \sim *C_2 * C_3) * C_2 * C_3 (*C_1 * C_2)$; - *опорные понятия, освоение которых гарантирует приобретение профессиональной компетентности PrC₁*, а для каждого из опорных понятий PrC₁ строится выражение знаний:

$C_1 \leq *C_{1.1} * C_{1.2} (+C_1 * C_2) \sim +C_{1.3} * C_{1.4}$; - *выражение знаний опорного понятия C₁ профессиональной компетентности PrC₁, включающего понятия предметной области. Для проекта "Банковская система типа клиент-сервер", выделено 54 выражений знаний, определяющих необходимый и достаточный уровень знаний, необходимых для реализации работ и процессов жизненного цикла разработки распределенных приложений и систем реального времени. На базе выражений знаний сформированы знаниевые компоненты, которые использованы для проектирования **знаниевой архитектуры** дисциплины "Технология разработки распределенных приложений", специальности "Вычислительная техника и программное обеспечение".*

Возникшие трудности, способы их преодоления

В качестве основополагающих концепций процесса формирования знаниевых компонент приняты: **проектно-ориентированная технология обучения и стандарты CDIO; компетентностная модель этапа CDIO; SWEBOOK - области знаний инженерии программного обеспечения; онтологическая модель отображения знаний.**

Для преподавателей разработан парсер, позволяющий формировать и редактировать знаниевые компоненты репозитория образовательной среды.

Основные трудности возникли с инженерией образовательной области - деятельностью, связанной с анализом источников образовательных ресурсов домена обучения, формализацией семантического контекста выделенных ресурсов в виде онтологий опорных понятий, и их спецификацией выражениями знаний, поскольку разработанная методика требует:

- во-первых, проведение анализа структуры и организации семантического контекста домена обучения, используемого для проектирования знаниевых компонент дисциплин учебных планов и образовательных программ;
- во-вторых, создание визуально-мыслительных образов, с помощью ассоциативного связывания понятий и структурных элементов **семантического контекста домена обучения**, необходимых для формирования целостной системы знаний, в рамках отдельной дисциплины, курса, или специальности.

ФОРМАЛИЗМЫ МОДЕЛИРОВАНИЯ ЗНАНИЙ

Модель онтологии

$O_m = \langle C, R, F \rangle$, где C – множество понятий (терминов) предметной области; R – множество отношений между понятиями; F – множество функций интерпретации, определения которых заданы на отношениях между понятиями в онтологии.

В наших исследованиях модель онтологии представляется иерархической структурой понятий, корневой вершиной которой является **опорное понятие**, второй уровень – *идентифицирующие* понятия опорного понятия, третий уровень – *конкретизирующие* понятия каждого из идентифицирующих понятий онтологии.

Для спецификации модели онтологии разработан язык, позволяющий описывать **онтологию опорного понятия** в виде **выражения знания**. Для этих целей используются знаки отношений: ‘*’, ‘+’ и ‘~’, которые на понятиях онтологии определяют соответственно отношения «композиция», «агрегация» и «альтернативный выбор», и с помощью которых отображаются свойства **общности** и **изменчивости** понятий онтологии.

Введенные **типы понятий** и отношения на понятиях, покажем на примере области знания программной инженерии **SWEBOK: "Требования к программному обеспечению"**, декларативные знания которой представим онтологиями семи опорных понятий. Рассмотрим спецификацию онтологии для одного из опорных понятий: **C_5 - Нефункциональные требования.**

*C_{5.1} - требования к продукту. Данное идентифицирующее понятие является семантическим контекстом второго уровня конкретизации, включающего следующие дочерние понятия: *C₁ - требования к эксплуатации, *C₂ - требования к эффективности, *C₃ - требования к надежности, +C₄ - требования к переносимости, *C₅ - обработка ошибок, *C₆ - интерфейсные требования.

*C_{5.2} - организационные требования. Данное идентифицирующее понятие является семантическим контекстом второго уровня конкретизации, включающего следующие дочерние понятия: *C₁ - выходные требования, +C₂ - требования на реализацию, *C₃ - требования на стандарты.

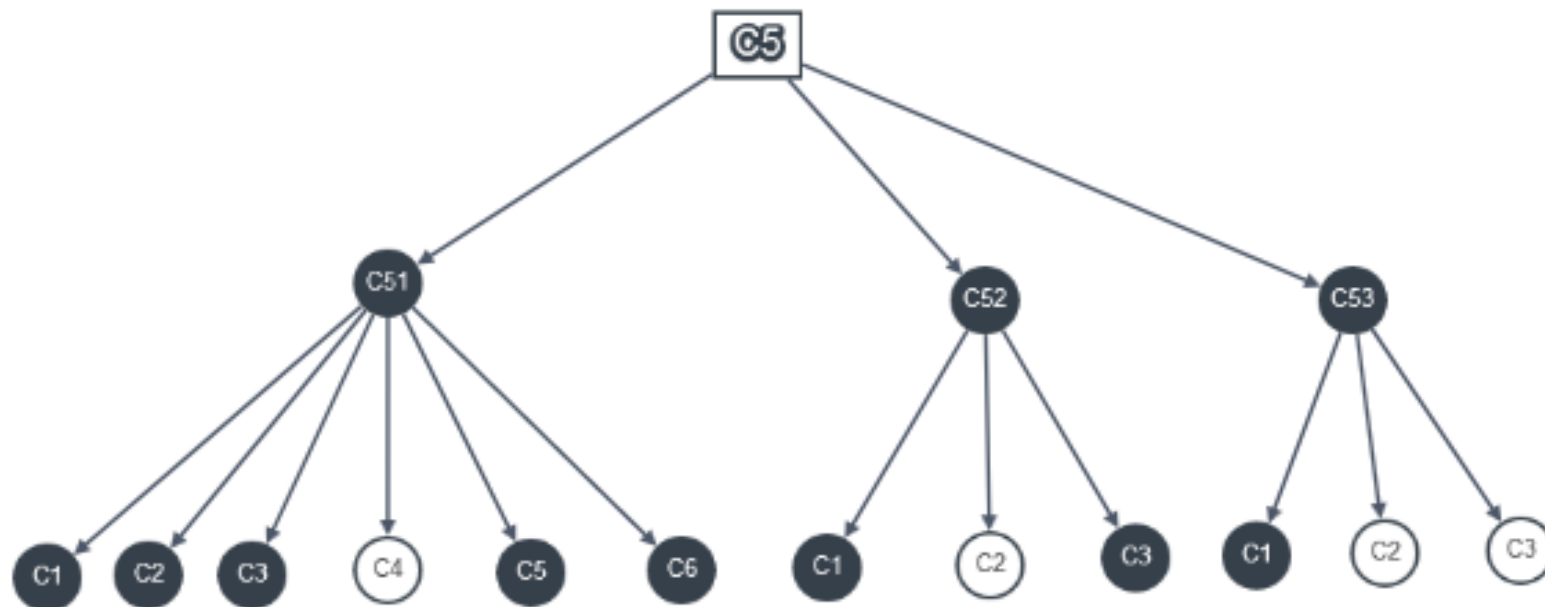
*C_{5.3} - внешние требования. Данное идентифицирующее понятие является семантическим контекстом второго уровня конкретизации, включающего следующие дочерние понятия: *C₁ - требования на взаимодействие, +C₂ - этические требования, +C₃ - юридические требования.

Выражение знания опорного понятия C₅ - «Нефункциональные требования» будет иметь следующий вид:

$$C_5 \leq *C_{5.1}(*C_1*C_2*C_3+C_4*C_5*C_6)*C_{5.2}(*C_1+C_2*C_3)*C_{5.3}(*C_1+C_2+C_3);$$

где понятия выражения знания, вместе с определенными над ними отношениями, **имплицитно истинность** заключения об опорном понятии C₅.

Онтология опорного понятия визуально представляется в виде реляционного графа, в котором закрашенные кружочки отображают понятия, имеющие *свойство общности*, а не закрашенные - имеющие *свойство изменчивости*.



ПРОЕКТНО – КОМПЕТЕНТНОСТНАЯ ПАРАДИГМА ПРЕДСТАВЛЕНИЯ ЗНАНИЙ

Задача образовательной среды заключается в том, чтобы в соответствии с требуемыми *профессиональными компетенциями* этапов инициативы **CDIO**, **сформировать знаниевые компоненты**, используемые для проектирования базовых и профилирующих дисциплин учебного плана специальности.

В качестве основного контента, была использована монография *Гома Хассан UML. Проектирование систем реального времени, параллельных и распределенных приложений: Пер. с англ.-М.: ДМК Пресс, 2014. -с.704.*

Согласно концепции инициативы **CDIO** любой технологический процесс выполнения проекта представляется последовательностью четырех обязательных этапов: **Conceive, Design, Implement и Operate**.

Для формирования знаниевых компонент рассматривался студенческий проект **"Банковская система типа клиент-сервер"**.

На примере этапа **Design**, представим онтологии опорных понятий, которые были использованы для проектирования знаниевого контента дисциплины **"Технология разработки распределенных приложений"**, специальности «Вычислительная техника и программное обеспечение».

Компетентностная модель этапа Design

Рассмотрим, например, этап **Design**, проекта "Банковская система типа клиент-сервер", для успешной разработки которого обучающимся необходимо освоить знания компетентностной модели этапа **Design**, представляемой композицией *профессиональных, базовых и дополнительных* компетенций.

Профессиональные компетенции этапа **Design** включают онтологии *шести* опорных понятий:

$$PrC = \{C_1, C_2, C_3, C_4, C_5, C_6\}, \text{ где}$$

C₁ - Консолидированная диаграмма кооперации клиентской подсистемы;

C₂ - Консолидированная диаграмма кооперации подсистемы сервера;

C₃ - Декомпозиция системы на подсистемы;

C₄ - Параллельные задачи подсистем;

C₅ - Архитектура задач проектной модели;

C₆ - Детальное проектирование программы.

Базовые компетенции этапа **Design** включают онтологии *восьми опорных понятий*:

$$\text{BaC} = \{C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8\},$$

где C_1 - *Архитектурное проектирование*;

C_2 - *Структурирование системы*;

C_3 - *Модели управления*;

C_4 - *Модульная декомпозиция системы*;

C_5 - *Архитектура распределенных приложений - параллельных и реального времени*;

C_6 - *Архитектура распределенных объектов*;

C_7 - *Проектирование классов, скрывающих информацию*;

C_8 - *COMET - метод архитектурного проектирования и моделирования параллельных объектов*.

Дополнительные компетенции этапа **Design** включают онтологии *пяти опорных понятий*:

$$\text{AdC} = \{C_1, C_2, C_3, C_4, C_5\}, \quad \text{где}$$

C_1 - *Объектно-ориентированное проектирование*;

C_2 - *Детальное проектирование*;

C_3 - *Типы архитектур*;

C_4 - *Проектирование по контракту*;

C_5 - *ICONIX - метод проектирования систем в архитектуре MDA*.

Выражения компетентностей (компетентностная модель этапа Design)

$ec.PrC.Design \leq *PrC_1 * PrC_2 * PrC_3;$ - выражение *профессиональных*
компетентностей

$ec.BaC.Design \leq *BaC_1 + BaC_2 * BaC_3;$ - выражение *базовых*
компетентностей

$ec.AdC.Design \leq *AdC_1 \sim *AdC_2 (*A_1 + A_2) \sim *AdC_3;$ - выражение
дополнительных компетентностей.

$PrC_1 \leq *C_1 (*C_1 \sim *C_2 * C_3) * C_2 * C_3 (*C_1 * C_2);$ - *опорные понятия*
профессиональной компетентности PrC_1

$C_1 \leq *C_{1.1} * C_{1.2} (+C_1 * C_2) \sim +C_{1.3} * C_{1.4};$ - *выражение знания* опорного
понятия C_1 профессиональной компетентности PrC_1

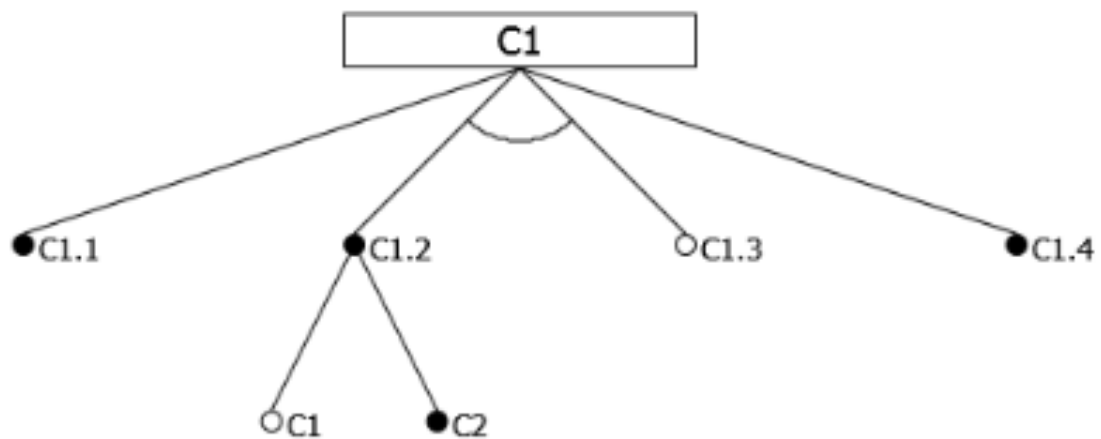
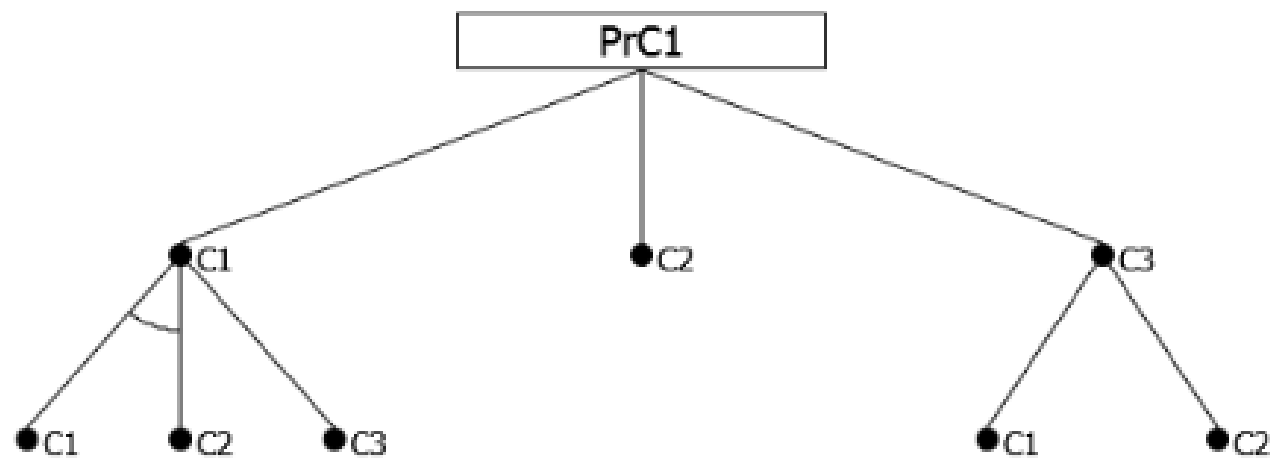
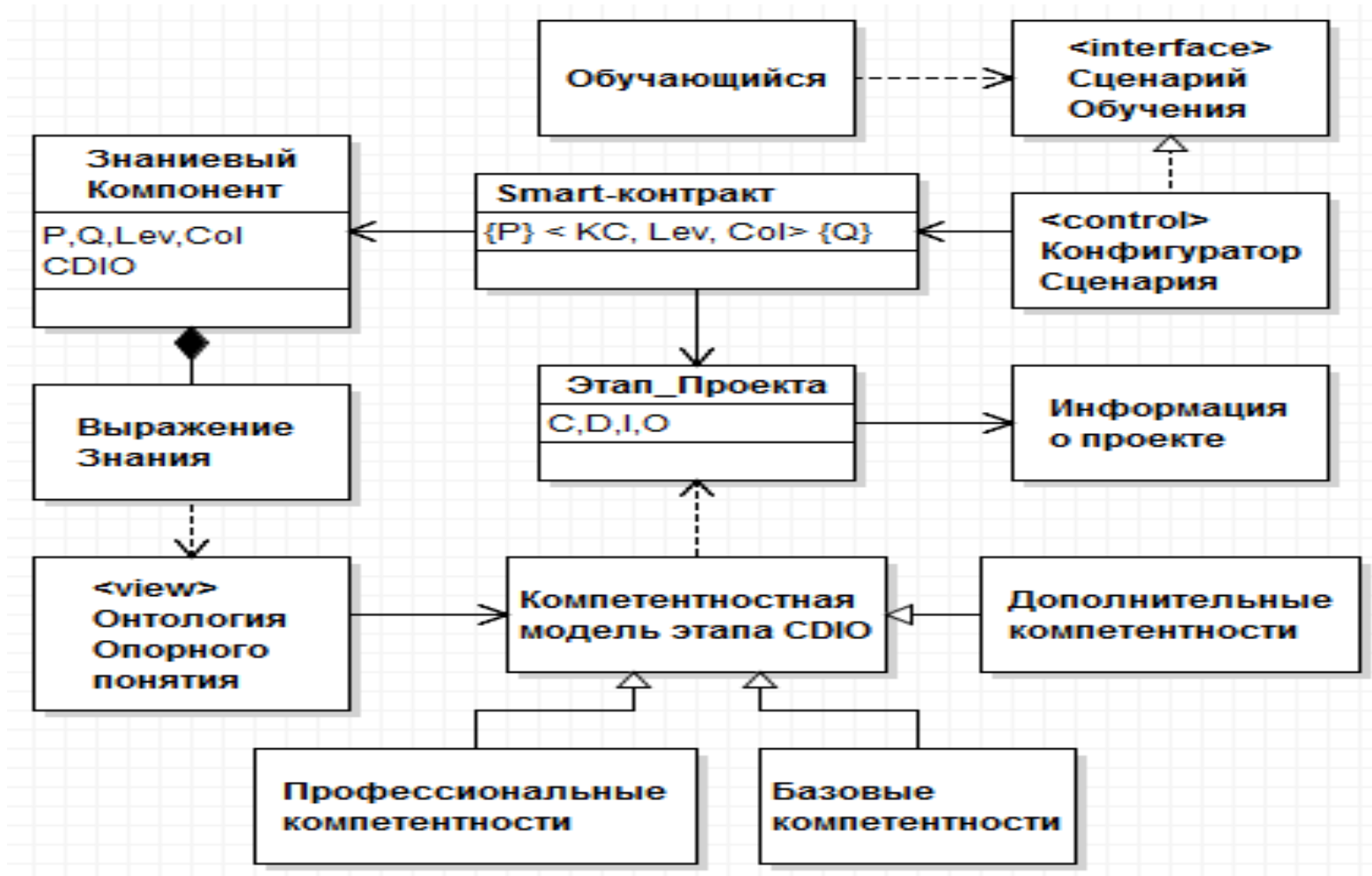


Диаграмма классов конфигуратора сценария знаниевой архитектуры



Smart-контракт знаниевого компонента

Принятая в исследовании *методика моделирования и отображения знаний* основана на концепциях: *компетентностного подхода, этапах проектно-ориентированной технологии обучения – CDIO и онтологическом инжиниринге.*

Данные концепции позволяют релевантно моделировать знания предметной области в виде онтологий опорных понятий и специфицировать их **выражениями знаний**.

Знаниевый компонент (КС, Knowledge Component) – композиция выражений знаний, представляющая собой структурный элемент среды и конфигурационного управления, и обладающий четко определенным *smart-контрактом*, с помощью которого описываются правила реализации знаниевого компонента в образовательной программе.

Другими словами, знаниевый компонент представляет собой выделенную область специальных знаний, требуемых для решения определенных практических задач и взаимодействующий с другими знаниевыми компонентами среды, посредством параметров smart-контракта:

{P}< КС, Col, Lev>{Q}

{P}< КС, Col, Lev>{Q}

Параметр **P** - *предусловие*, то есть утверждение в виде сигнатуры требуемых компетентностей, необходимых для успешности обучения, предусмотренных знаниевым компонентом КС. Предусловие всегда должно выполняться при обращении к знаниевому компоненту, иначе корректность результата обучения не гарантируется.

Параметр **Q** – *постусловие*, то есть утверждение в виде *сигнатуры компетентностей*, гарантируемых обучающемуся, при выполнении *предусловия* и успешном освоении знаний, предоставляемых знаниевым компонентом.

Параметры **Lev** и **Col**, определяют **инфраструктуру** компонентной среды, позволяющую учитывать индивидуальные особенности обучающегося, а знаниевым компонентам, взаимодействовать по определенным правилам.

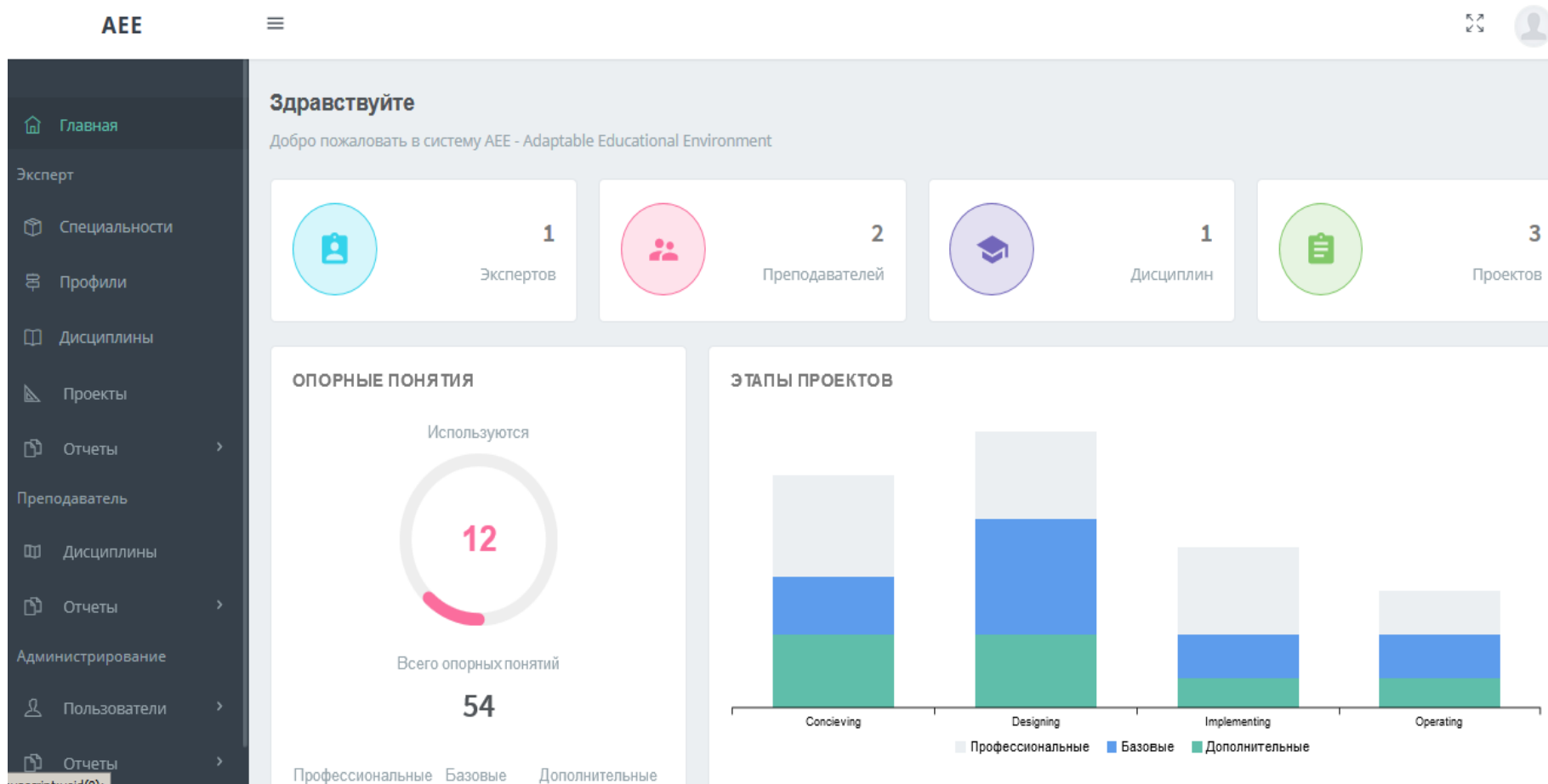
Таким образом, параметры **smart-контракта** знаниевого компонента, позволяют рассматривать знаниевый компонент как элемент конфигурационного управления процесса проектирования *знаниевой архитектуры образовательной программы*.

Концептуально, знаниевая архитектура образовательной программы представляется в виде *взаимодействия* знаниевых компонент, реализация каждого из которых задается параметрами smart-контракта - **{P}< КС, Col, Lev>{Q}**

Введенная концепция smart-контракта позволяет:

- *во-первых*, использовать smart-контракт для моделирования знаниевых компонент прототипа образовательной программы;
- *во-вторых*, проводить уточнение и верификацию знаниевой архитектуры, на предмет соответствия требованиям обучающегося;
- *в - третьих*, производить оценку выражений знаний репозитория интерактивной образовательной Smart - среды;
- *в-четвертых*, производить мониторинг процесса обучения и успеваемости обучающегося;
- *в-пятых*, проводить анализ характеристик знаниевой архитектуры с целью оценки ее пригодности для обучения;
- *в-шестых*, находить проектные решения по формированию знаниевых компонент *повторного использования*.

Интерактивная образовательная Smart - среда





- Главная
- Эксперт
- Специальности
- Профили
- Дисциплины
- Проекты
- Отчеты >
- Преподаватель
- Дисциплины
- Отчеты >
- Администрирование
- Пользователи >
- Отчеты >

CONCEIVING

DESIGNING

IMPLEMENTING

OPERATING

ЗАДУМКА

Список опорных понятий этапа Conceiving - Задумка.

[+ Добавить опорное понятие](#)

	Опорное понятие	Выражение знания
Профессиональная компетенция		
	Инженерия требований	$C1 \leftarrow *C11(+C1\rightarrow+C2*C3*C4+C5)*C12(*C1*C2*C3)*C13(*C1+C2\rightarrow+C3\rightarrow+C4)+C14+C15;$
	Моделирование предметной области	$C2 \leftarrow *C2.1(*C1*C2+C3*C4+C5)*C2.2(*C1*C2*C3*C4*C5*C6)*C23(*C1*C2*C3*C4)*C24(*C1*C2*C3+C4);$
	Разбиение на объекты	$C3 \leftarrow *C31(*C1+C2*C3+C4*C5)*C32*C33*C34(*C1*C2*C3);$
	Динамическое моделирование системы	$C4 \leftarrow *C41(*C1*C2)*C42*C43*C44(*C1*C2+C3);$
	Технологии параллельной и распределенной обработки	$C5 \leftarrow *C51(*C1*C2*C3)*C52(*C1*C2*C3)*C53(*C1*C2*C3*C4*C5)*C54(*C1*C2*C3*C4*C5 *C6);$
	Планирование задач	$C6 \leftarrow *C61*C62*C63(*C1*C2*C3);$
	Технологии создания клиент-серверных и распределенных приложений	$C7 \leftarrow *C71*C72*C73(*C1*C2*C3*C4*C5*C6)*C74(*C1*C2*C3)*C75(*C1*C2*C3*C4*C5)*C76(*C1*C2*C3)*C77(*C1*C2);$

[Главная](#)

Эксперт

[Специальности](#)[Профили](#)[Дисциплины](#)[Проекты](#)[Отчеты](#) >

Преподаватель

[Дисциплины](#)[Отчеты](#) >

Администрирование

[Пользователи](#) >[Отчеты](#) >

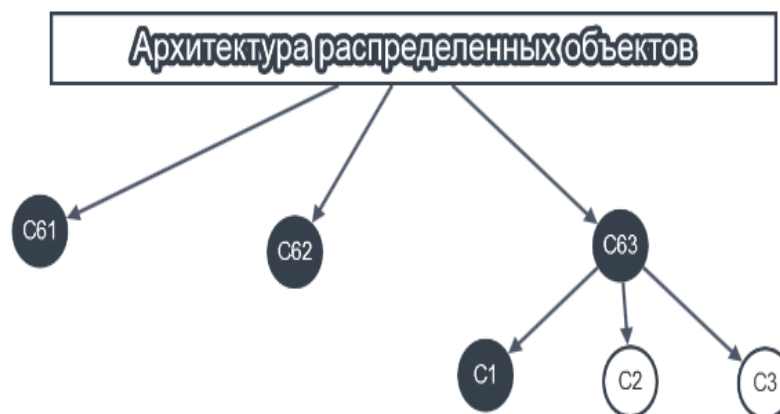
Архитектура распределенных объектов

[Главная](#) / [Проекты](#) / [Банковская система типа клиент сервер](#) / [Архитектура распределенных объектов](#)[Вернуться к списку](#)

Выражение знания

$$C6 \leftarrow *C61 * C62 * C63 (*C1 + C2 + C3);$$
[Построить онтологию](#)

МОДЕЛЬ ОНТОЛОГИИ

[Просмотреть печатную форму](#)

Исследования выполняются при поддержке гранта Министерства образования и науки Республики Казахстан - проект №АР05134973 «Исследование и разработка моделей и методики представления и организации знаний с применением онтологического подхода и инструментальных средств Smart-технологии, при реализации образовательных программ и процессов»



XVII открытая Всероссийская конференция
«Преподавание информационных технологий
в Российской Федерации»

N* Новосибирский
государственный
университет
*НАСТОЯЩАЯ НАУКА

БЛАГОДАРЮ ЗА ВНИМАНИЕ

Кубеков Булат Сальмуханович
Алматы, университет Туран

b.kubekov@mail.ru

+7 701 717 98 00



XVII открытая Всероссийская конференция
«Преподавание информационных технологий
в Российской Федерации»

N* Новосибирский
государственный
университет
*НАСТОЯЩАЯ НАУКА