Теоретико-категорный подход к проектированию программных систем

С. П. КОВАЛЁВ
Институт проблем управления им. В. А. Трапезникова РАН
e-mail: kovalyov@nm.ru

УДК 519.68

Ключевые слова: разработка, управляемая моделями, аспектно-ориентированное программирование, трассируемость, теория категорий, формальная технология проектирования, разделение ответственности.

Аннотация

Работа посвящена применению теории категорий для решения проблемы представления разнородных технологий программной инженерии в единой форме, удобной для их интеграции и координации в рамках общего цикла проектирования программных систем. Особое внимание уделяется современным технологиям, таким, как разработка, управляемая моделями, и аспектно-ориентированное программирование. Построены универсальные теоретико-категорные семантические модели этих технологий. Предложен новый способ разделения ответственности путём экспликации аспектной структуры формальных моделей программ. Построены и исследованы формальные технологии проектирования технологий, образующие математический фундамент разработки, управляемой моделями.

Abstract

S. P. Kovalyov, Category-theoretic approach to software systems design, Fundamentalnaya i prikladnaya matematika, vol. 19 (2014), no. 3, pp. 111—170.

Category theory is applied to the problem of representing heterogeneous software engineering technologies in a unified form suitable for their integration and coordination within the common software systems engineering cycle. Special attention is paid to modern technologies such as model-driven engineering and aspect-oriented programming. Universal category-theoretic semantic models of these technologies are constructed. A novel method of separation of concerns by explicating the aspectual structure of formal models of programs is proposed. We construct and analyze formal technologies (architecture schools) for designing technologies that comprise a mathematical basis for model-driven engineering.

Введение

Автоматизация процессов проектирования и производства технических изделий по праву считается главным способом повышения качества изделий и

Фундаментальная и прикладная математика, 2014, том 19, № 3, с. 111—170.

эффективности труда инженеров. В материальной промышленности изделия изготавливаются автоматическими станками и 3D-принтерами, задания для которых формируются на компьютере инструментами автоматизированного проектирования (CAD-средствами, Computer-Aided Design). Современные инструменты способны свести инженерную деятельность к созданию, анализу и преобразованию компьютерных моделей — упрощённых формальных представлений изделия, отражающих его крупноблочную структуру и основные свойства. Главную роль традиционно играют геометрические и мультифизические модели: они позволяют на этапе проектирования виртуально имитировать поведение изделия в любых условиях, сборку составного изделия из частей, разнообразные технологические процедуры.

Виртуальная имитация и автоматическое изготовление изделий могут быть очень полезны и в разработке программного обеспечения. Инструменты моделирования программ, анализа моделей, генерации текстов (CASE-средства, Computer Aided Software Engineering) активно разрабатываются уже более полувека. Однако эффект от их внедрения оказывается значительно меньшим, чем от внедрения CAD-средств [7]. Это во многом связано с особой нематериальной природой программного обеспечения: в отличие от технических изделий, оно не имеет ни геометрических, ни физических характеристик. Программные изделия характеризуются широким набором специфических показателей, зачастую нечисловых, трудно формализуемых, имеющих различный смысл в зависимости от предметной области задач, для решения которых предназначено изделие. Примерами таких показателей служат изменяемость, устойчивость, правильность [2]. Большое количество разнородных показателей приводит к появлению множества разноплановых языков и методов моделирования программ, препятствуя созданию универсальных CASE-средств. Среди моделей программ встречаются объектные, структурные, сценарные, автоматные, алгебраические, логические, денотационные и многие другие.

Чтобы эффективно поддерживать такое разнообразие моделей в технологическом цикле разработки программных систем, в середине 2000-х годов был предложен новый подход к автоматизации процессов программной инженерии — разработка, управляемая моделями (MDE, Model-Driven Engineering) [35]. Этот подход предписывает создавать для удобства моделирования программ разнообразные формализованные предметно-ориентированные языки. Языки строятся путём специализации на уровне метамоделей из языков общего назначения, таких, как язык объектно-ориентированного моделирования UML (Unified Modeling Language) [22], языки реального времени [12], языки логического вывода [1] и др. Предоставляются инструменты для создания средств автоматического формирования, анализа, сборки и трансформации моделей, в том числе генерации текстов программ. Большой набор таких инструментов и средств создан и опубликован в открытом исходном коде в рамках проекта «Eclipse Modeling Project» [16].

Однако технологии MDE оказались плохо масштабируемыми: затраты труда, сопряжённые с их применением, слишком быстро растут по мере увеличения размера и структурной сложности программных систем [29]. Особые трудности в ходе разработки системы вызывает согласование модификаций в многочисленных взаимосвязанных моделях, ответственных за разные части или аспекты системы и записанных на разных языках [26]. Условия согласованности выявляются путём трассирования задач — прослеживания их реализации в разнородных моделях и программах. Чем полнее и детальнее проведено трассирование, тем проще локализовать модификации, необходимые в ответ на очередное требование, и избежать побочных эффектов, способных вызвать утрату соответствия другим требованиям. Трассирование задач относится к числу самых трудоёмких операций в программной инженерии [9]. Главная проблема состоит в том, что разработчики программных систем не привыкли считать удобство трассирования обязательным критерием качества результатов своей деятельности [27]. Широко распространённые языки, технологии и стили проектирования и программирования разрешают неявно «подразумевать» назначение фрагментов моделей и программ, избавляя авторов от необходимости педантично записывать трассировочные взаимосвязи. Известное исключение составляет аспектно-ориентированное программирование (АОП) [28], позволяющее автоматически трассировать вспомогательные программно-технические задачи, такие, как ведение журнала функционирования системы, защита информации и т. д. Средства решения таких задач рассеиваются по единицам модульной архитектуры, «перемешиваясь» с программной реализацией основных задач, локализованной в модулях. Технологии АОП позволяют оформлять рассеянные задачи в виде так называемых аспектов — особых самостоятельных программных единиц, которые автоматически встраиваются в модули в явно задаваемых точках, обеспечивая трассируемость «по построению».

Большой интерес вызывает применение АОП в технологиях типа MDE, поскольку в ходе разработки моделей с привлечением аспектно-ориентированного подхода формируются детализированные трассировочные связи [23]. Однако такому применению препятствует недостаточная проработка концептуальной базы АОП, реализации которого специфичны для частных технологий составления и моделирования программ [37]. Неясно, как расширить произвольную технологию проектирования программных систем типа MDE эффективными приёмами формирования моделей аспектов и их встраивания в модели модулей.

Актуальна и более общая проблема представления разнообразных технологий программной инженерии в единой универсальной форме, удобной для их интеграции и координации в рамках общего цикла проектирования сложных гетерогенных систем. При решении этой проблемы важную роль играет выбор математического аппарата для моделирования и анализа самих процессов программной инженерии с достаточным уровнем строгости. Традиционный подход к математическому моделированию посредством дифференциальных уравнений и минимизируемых функционалов, выработанный в физике и других науках, здесь не помогает из-за отсутствия подходящих аналогов вариационных принципов, законов сохранения, статистических закономерностей и т. п. [7]. Существует альтернативный подход, основанный на наблюдении, что для большинства систем доступна (либо легко восстановима) история их сборки из некоторых первичных компонентов. Если известны математические модели компонентов и технологических операций разработки систем, то можно рассчитывать интегральные характеристики системы по формальным аналогам сборочных чертежей или «мегамоделям» — ориентированным графам (диаграммам), узлы которых помечены обозначениями компонентов, а рёбра помечены обозначениями операций. Здесь требуется формировать и обрабатывать большие графы, которые даже нельзя изобразить целиком, а можно только описать структурными ограничениями. Мощный аппарат для конструирования и анализа графов такого рода развит в рамках теории категорий — раздела высшей алгебры, который «начинается с наблюдения, что многие свойства математических систем можно представить просто и единообразно посредством диаграмм» [6, с. 12]. Системным единицам (компонентам, подсистемам, системам и т. д.) ставятся в соответствие объекты подходящих категорий, операциям — морфизмы, комплексным технологическим процедурам — диаграммные конструкции [14]. Для структурного согласования разнородных процедур, заданных в разных категориях такого рода, вводятся подходящие функторы между этими категориями.

Для разностороннего исследования процессов разработки программных систем средствами теории категорий была предложена особая общая конструкция — формальная технология проектирования [18]. В настоящей работе, в развитие результатов работы [4], конструкции такого рода рассматриваются как объекты категорий, морфизмы в которых описывают конструирование и преобразование технологий, в том числе создание предметно-ориентированных языков и средств моделирования. В частности, добавление приёмов трассирования и АОП в произвольную технологию формально описано как её преобразование, в результате которого модели получают дополнительную структуру — разметку классами задач на уровне интеграционных интерфейсов. Если удаётся «поднять» разметку на уровень самих моделей (эксплицировать), то путём трассирования классов задач можно извлечь из модели составляющие её аспекты в виде модульных единиц, т. е. произвести полное разделение ответственности. Такой подход к формализации MDE и АОП иллюстрируется примерами из разнообразных технологий проектирования программных систем. Конечно, в литературе описано множество других формальных подходов (см. [15] и др.), однако они представлены в терминах частных формализмов теоретического программирования (проверка на моделях, лямбда-исчисление и др.) и поэтому могут применяться только в рамках частных технологий.

Работа построена следующим образом. В разделе 1 вводится понятие формальной технологии. Раздел 2 посвящён трассированию задач и преобразованию модульных технологий в аспектно-ориентированные. В разделе 3 описаны конструкции аспектного связывания и экспликации аспектной структуры. В разделе 4 строятся и анализируются формальные технологии проектирования формальных технологий. В разделе 5 рассмотрено конструирование предметно-ориентированных технологий, предоставляющих широкие возможности для трассирования. В заключении подводятся итоги и намечаются пути дальнейших исследований.

1. Принципы теоретико-категорного описания процессов разработки программ

Определения теоретико-категорных понятий, используемых в настоящей работе, можно найти в [6,8]. В частности, в [8] объекты и морфизмы категории C кратко называются C-объектами и C-морфизмами соответственно, а функтор $\Delta\colon X \to C$ — C-диаграммой со схемой X (мы будем предполагать, что X — малая категория). Диаграмма рассматривается как граф категории X, вершины которого помечены C-объектами, а рёбра — C-морфизмами. Если из X удалить все нетождественные морфизмы, то получится дискретная поддиаграмма в Δ, которая обозначается через $|\Delta|$, а её схема — через $|X|$. Все C-диаграммы образуют категорию $\mathbf{D}C$ (ковариантная категория «сверхзапятой» [6] — конструкция Гротендика для функтора $C^{-}\colon X \mapsto C^{X}$), в которой морфизмом диаграммы $\Delta\colon X \to C$ в $\Xi\colon Y \to C$ служит любая пара вида $\langle \varepsilon, \mathrm{fd}\rangle$, состоящая из функтора $\mathrm{fd}\colon X \to Y$ и естественного преобразования $\varepsilon\colon \Delta \to \Xi \circ \mathrm{fd}$. Любой функтор $\mathrm{fun}\colon C \to D$ порождает функтор

$$\begin{aligned}\mathrm{fun} \circ -\colon\ & \mathbf{D}C \to \mathbf{D}D\\ :\ & \Delta \mapsto \mathrm{fun} \circ \Delta,\ \langle \varepsilon, \mathrm{fd}\rangle \mapsto \langle \mathrm{fun}(\varepsilon), \mathrm{fd}\rangle,\end{aligned}$$

так что имеется эндофунктор

$$\begin{aligned}\mathbf{D}\colon\ & \mathbf{CAT} \to \mathbf{CAT}\\ :\ & C \mapsto \mathbf{D}C,\ \mathrm{fun} \mapsto (\mathrm{fun} \circ -),\end{aligned}$$

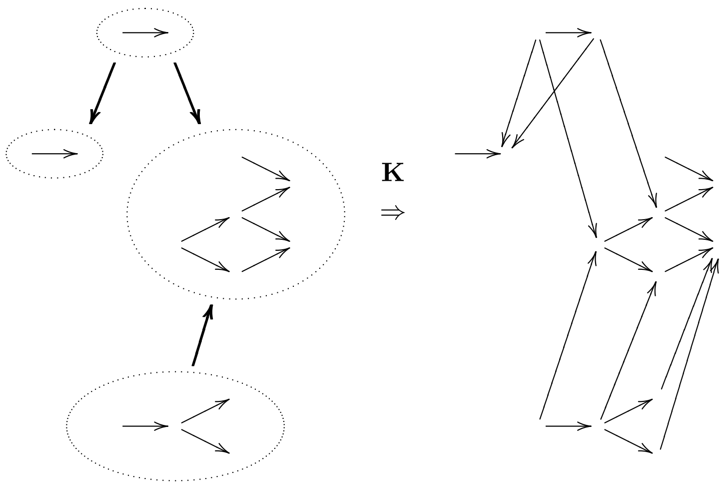
где $\mathbf{CAT}$ — категория всех категорий и всех функторов.

Обозначим через $\mathbf{\mathit{1}}$ терминальный $\mathbf{CAT}$-объект — категорию, состоящую из одного объекта 0 и одного морфизма 1_0. Имеется полное вложение $\ulcorner - \urcorner\colon C \hookrightarrow \mathbf{D}C$, ставящее в соответствие произвольному C-объекту S точку — диаграмму $\ulcorner S \urcorner\colon \mathbf{\mathit{1}} \to C\colon 0 \mapsto S$. Коконусом называется $\mathbf{D}C$-морфизм, имеющий точку в качестве кообласти. Копределом диаграммы Δ называется коконус $\mathrm{colim}\,\Delta\colon \Delta \to \ulcorner S \urcorner$, универсальный в том смысле, что для любых C-объекта T и коконуса $\delta\colon \Delta \to \ulcorner T \urcorner$ существует единственный C-морфизм $w\colon S \to T$ (стрелка копредела), такой что $\delta = \ulcorner w \urcorner \circ \mathrm{colim}\,\Delta$. Если копредел существует, то он определяется однозначно с точностью до изоморфизма: коконус δ является копределом диаграммы Δ тогда и только тогда, когда существует C-изоморфизм i, такой что $\delta = \ulcorner i \urcorner \circ \mathrm{colim}\,\Delta$.

Пусть Cdia — некоторый класс C-диаграмм, имеющих копределы. Можно рассматривать его как полную подкатегорию в $\mathbf{D}C$ и определить функтор копредела colim, действующий из него в C, ставя в соответствие каждой диаграмме из Cdia вершину некоторого её копредела, а каждому $\mathbf{D}C$-морфизму $\theta\colon \Delta \to \Xi$, где $\Delta, \Xi \in \mathrm{Cdia}$, — стрелку копредела $\mathrm{colim}(\theta)$, такую что $\mathrm{colim}\,\Xi \circ \theta = \ulcorner \mathrm{colim}(\theta) \urcorner \circ \mathrm{colim}\,\Delta$:

$$\begin{tikzcd}
\Delta \arrow[r, "\mathrm{colim}\,\Delta"] \arrow[d, "\theta"'] & \ulcorner \mathrm{colim}(\Delta) \urcorner \arrow[d, dotted, "\ulcorner \mathrm{colim}(\theta) \urcorner"] \\
\Xi \arrow[r, "\mathrm{colim}\,\Xi"] & \ulcorner \mathrm{colim}(\Xi) \urcorner
\end{tikzcd} \quad .$$

Для наглядности копределы и другие диаграммные конструкции, заданные в категории $\mathbf{D}C$, часто изображаются C-диаграммами. Например, произвольный коконус $\nu\colon \Delta \to \ulcorner P \urcorner$ можно изобразить, «пририсовав» стрелку — ребро, направленное в добавочную вершину P, к каждой вершине диаграммы-основания Δ. Этот приём можно формализовать и обобщить следующим образом. Коконус очевидным образом рассматривается как $\mathbf{D}C$-диаграмма со схемой $\mathit{2}$ (напомним, что символом $\mathit{2}$ обозначается категория, состоящая из двух объектов 0, 1, их тождественных морфизмов и одного нетождественного морфизма $0 \to 1$). В свою очередь, любую $\mathbf{D}C$-диаграмму можно перевести в C-диаграмму каноническим естественным способом — посредством функтора отрисовки $\mathbf{K}\colon \mathbf{DD}C \to \mathbf{D}C$. Действие функтора $\mathbf{K}$ на произвольную диаграмму $\Gamma\colon Z \to \mathbf{D}C$ заключается в замене каждой вершины A графа схемы Z графом C-диаграммы $\Gamma(A)$, а каждого ребра $f\colon A \to B$ совокупностью рёбер, по одному для каждой вершины I графа диаграммы $\Gamma(A)$, направленному из I в вершину $\mathrm{fd}(I)$ графа диаграммы $\Gamma(B)$ и помеченному C-морфизмом ε_I, где $\langle \varepsilon, \mathrm{fd} \rangle = \Gamma(f)$. В [24] дано строгое определение функтора отрисовки и доказано, что тройка $\langle \mathbf{D}, \ulcorner - \urcorner, \mathbf{K} \rangle$ образует монаду в $\mathbf{CAT}$, по своим свойствам аналогичную монаде степени $\langle 2^-, \{-\}, \cup \rangle$ в категории $\mathbf{Set}$ всех множеств и всех отображений.



Например, встраивание коконуса $\nu\colon \Delta \to \ulcorner P \urcorner$ в некоторую C-диаграмму Ξ, содержащую P среди вершин, можно отрисовать в виде C-диаграммы $\mathbf{K}\Pi$, где Π — $\mathbf{D}C$-диаграмма со схемой $\mathit{2}$, изображающая $\mathbf{D}C$-морфизм $\pi \circ \nu\colon \Delta \to \Xi$, а $\pi\colon \ulcorner P \urcorner \hookrightarrow \Xi$ — вложение. Примечательно, что если диаграмма Ξ обладает копределом, то встраивание коконуса не меняет его (более строго, существует $\mathbf{D}C$-морфизм $\rho\colon \mathbf{K}\Pi \to \Xi$, такой что $\operatorname{colim} \mathbf{K}\Pi = (\operatorname{colim} \Xi) \circ \rho$ и $\rho \circ \chi = 1_\Xi$, где $\chi\colon \Xi \hookrightarrow \mathbf{K}\Pi$ — каноническое вложение поддиаграммы). Ещё одна диаграммная конструкция, с которой мы встретимся в дальнейшем, называется сингулярной склейкой. Это отрисовка $\mathbf{D}C$-диаграммы, имеющей вид пары стрелок с общим началом $\varphi\colon \Phi \leftarrow \ulcorner \mathbf{1} \urcorner \to \Theta :\theta$, где $\mathbf{1}$ — терминальный C-объект. Копредел склейки имеет ту же вершину, что и копредел C-диаграммы $s\colon \operatorname{colim}(\Phi) \leftarrow \mathbf{1} \to \operatorname{colim}(\Theta) :t$, где $\ulcorner s \urcorner = (\operatorname{colim} \Phi) \circ \varphi$ и $\ulcorner t \urcorner = (\operatorname{colim} \Theta) \circ \theta$ (при условии, что все упомянутые копределы существуют).

Нам потребуется рассматривать функторы с точки зрения их воздействия на копределы. Следуя [8, разд. 13], говорим, что функтор $\operatorname{fun}\colon C \to D$

— сохраняет копределы диаграммы Δ, если для любого копредела $\delta\colon \Delta \to \ulcorner S \urcorner$ коконус $\operatorname{fun} \circ \delta\colon \operatorname{fun} \circ \Delta \to \ulcorner \operatorname{fun}(S) \urcorner$ является копределом диаграммы $\operatorname{fun} \circ \Delta$;

— поднимает копределы диаграммы Δ, если для любого копредела $\xi\colon \operatorname{fun} \circ \Delta \to \ulcorner T \urcorner$ существует копредел $\delta\colon \Delta \to \ulcorner S \urcorner$, такой что $\operatorname{fun} \circ \delta = \xi$.

Следуя [4], говорим, что функтор *детерминирует* копределы диаграммы Δ, имеющей копредел, если он сохраняет и поднимает их. Например, функтор-проекция произведения категорий на какую-либо компоненту детерминирует копределы всех диаграмм, обладающих ими. Такой же способностью детерминировать копределы обладает функтор-инъекция любой компоненты в сумму категорий.

Теоретико-категорный подход к формализации разработки программ развивается начиная с 1970-х годов (см., например, [19]). Основную роль играют категории, объектами в которых служат формальные модели программ, а морфизмы формально описывают всевозможные действия по их интеграции. Композиция морфизмов отвечает выполнению многошаговых действий (процессов), а тождественные морфизмы отвечают «ничегонеделанию». Мы будем обозначать категорию такого рода через c-DESC. На неё не накладывается никаких априорных ограничений, поскольку нет оснований исключать, что произвольная категория описывает некоторую (может быть, ещё не придуманную) технологию разработки систем.

Актам сборки систем отвечают копределы конфигураций (формальных «мегамоделей» [26]) — c-DESC-диаграмм, состоящих из компонентов и их интеграционных взаимосвязей. Например, рассмотрим *соединение* компонента P с системой S — приём сборки, состоящий в добавлении промежуточного компонента G, называемого «клеем» или связкой [10], который способен интегрироваться как с компонентом, так и с системой. Путём соединения строятся системы на базе промежуточного программного обеспечения, которое и служит связкой. Мегамодель соединения имеет вид пары морфизмов $f\colon P \leftarrow G \to S :g$, которую мы обозначим через Δ. Коконус над Δ — это коммутативный квадрат: он задаётся объектом-вершиной V и парой морфизмов-рёбер $p\colon P \to V \leftarrow S :s$, удовлетворяющей соотношению $p \circ f = s \circ g$. Коконус является копределом (и называется кодекартовым квадратом), если выполняется следующее условие универсальности: для любых объекта T и пары морфизмов $u\colon P \to T \leftarrow S :v$, образующих коконус $\delta\colon \Delta \to \ulcorner T \urcorner$ (т. е. удовлетворяющих соотношению $u \circ f = v \circ g$), существует единственный морфизм $w\colon V \to T$, такой что $w \circ p = u$ и $w \circ s = v$. Это частный случай условия $\delta = \ulcorner w \urcorner \circ \operatorname{colim} \Delta$ из определения копредела, порождающий треугольную $\mathbf{D}$c-DESC-диаграмму, отрисовка которой обычно изображается следующим образом:

$$\begin{tikzcd}
G \arrow[r, "g"] \arrow[d, "f"'] & S \arrow[d, dashed, "s"] \arrow[ddr, "v"] & \\
P \arrow[r, dashed, "p"] \arrow[drr, "u"'] & V \arrow[dr, dotted, "w"] & \\
& & T.
\end{tikzcd}$$

Ясно, что если такой кодекартов квадрат существует, то объект V действительно отвечает системе, которая собрана из S и P путём соединения посредством G (и не содержит ничего «лишнего»). Если же копредела не существует, то делается вывод, что клей G не способен соединить компонент P с S посредством действий f и g.

Чтобы проводить всесторонние исследования процессов разработки программных систем средствами теории категорий, формализуются понятия интеграционного интерфейса и трансформации [18]. Интерфейсом называется часть компонента, «видимая» системе при интеграции, например декларация веб-сервиса на языке WSDL. Трансформация — это шаг разработки компонента, например реализация спецификации веб-сервиса на алгоритмическом языке программирования. Формальные модели интерфейсов образуют категорию, обозначаемую через SIG (поскольку в программировании они часто называются сигнатурами), а операция выделения интерфейса формализуется как функтор $\mathrm{sig}\colon \text{c-DESC} \to \mathrm{SIG}$. Поскольку разные модели могут иметь один и тот же интерфейс, функтор sig не обязан быть инъективным на объектах. Однако sig-образы двух разных действий по интеграции одного и того же компонента в одну и ту же систему должны быть разными: иначе получится, что интерфейсы недостаточно детально описывают интеграционные возможности компонентов. Иными словами, функтор sig должен быть унивалентным, т. е. инъективным на каждом множестве $\mathrm{Mor}(A, B)$, $A, B \in \mathrm{Ob}\,\text{c-DESC}$. Кроме того, интерфейсы должны быть реализуемыми: должен существовать функтор дискретной реализации интерфейсов $\mathrm{sig}^*\colon \mathrm{SIG} \to \text{c-DESC}$, такой что $\mathrm{sig} \circ \mathrm{sig}^* = 1_{\mathrm{SIG}}$ и имеется биекция $\mathrm{sig}\colon \mathrm{Mor}(\mathrm{sig}^*(I), S) \cong \mathrm{Mor}\big(I, \mathrm{sig}(S)\big)$ для любых $I \in \mathrm{Ob}\,\mathrm{SIG}$, $S \in \mathrm{Ob}\,\text{c-DESC}$

(на языке теории категорий это требование компактно формулируется в терминах сопряжения функторов). Например, дискретная реализация WSDL-описания веб-сервиса состоит из «заглушек» — пустых процедур, так что она может автоматически генерироваться CASE-средствами и позволяет быстро собирать отладочные версии приложений.

Существует естественная взаимосвязь между интерфейсами и конфигурациями. Если две c-DESC-диаграммы имеют один и тот же $\mathbf{D}$sig-образ, то они обе должны либо принадлежать классу Conf, либо нет. Это условие задаёт своего рода логический закон непротиворечия для интерфейсов: корректные конфигурации систем должны быть однозначно распознаваемыми на уровне интерфейсов. Кроме того, выделение интерфейса должно быть естественным относительно сборки систем в том «сильном» смысле, что копредел $\mathbf{D}$sig-образа любой конфигурации должен быть $\mathbf{D}$sig-образом её копредела (т. е. функтор sig должен поднимать копределы конфигураций). Как мы увидим далее (предложение 2), из этого и других свойств функтора sig вытекает естественность и в «слабом» смысле, более привычном в теории категорий: функтор sig сохраняет копределы конфигураций (иными словами, $\mathbf{D}$sig-образ копредела любой конфигурации является копределом её $\mathbf{D}$sig-образа). Таким образом, функтор sig детерминирует копределы всех конфигураций.

Что касается трансформаций, то они описываются морфизмами подходящей категории, обозначаемой через r-DESC, которая в общем случае отличается от c-DESC, однако обладает таким же классом объектов и содержит все изоморфизмы из c-DESC. Условие естественности трансформаций относительно сборки систем налагается в следующей форме: любой набор трансформаций объектов некоторой конфигурации должен порождать трансформацию собираемой из них системы как целого. Это условие можно формализовать с привлечением естественных преобразований, состоящих из r-DESC-морфизмов, поскольку любая дискретная c-DESC-диаграмма является также r-DESC-диаграммой. Произвольный набор трансформаций объектов диаграммы $\Delta\colon X \to \text{c-DESC}$ представляет собой не что иное, как естественное преобразование ($\text{r-DESC}^{|X|}$-морфизм) дискретной диаграммы $|\Delta|$ в дискретную диаграмму Σ, состоящую из результатов всех трансформаций. Если $\Delta \in \mathrm{Conf}$, то диаграмма Σ должна входить в некоторую конфигурацию, вершина копредела которой может быть получена путём трансформации из $\mathrm{colim}(\Delta)$.

Изложенные соображения приводят к следующей сложной конструкции, в рамках которой можно формально исследовать процессы разработки программных систем с разных сторон [18].

Определение 1. Пусть задана категория c-DESC, объектами в которой служат формальные модели программ, а морфизмами — формальные описания действий по интеграции программных систем. *Формальной технологией проектирования* над c-DESC называется четвёрка $\langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig}, \text{r-DESC}\rangle$, где

— Conf — класс c-DESC-диаграмм, называющихся допустимыми конфигурациями систем,

— sig — функтор из c-DESC в категорию, обозначаемую через SIG, объекты которой называются интерфейсами или сигнатурами, а морфизмы называются формальными описаниями действий по интеграции интерфейсов,
— r-DESC — категория, объектами которой являются формальные модели программ, а морфизмы называются формальными описаниями трансформаций,

причём выполняются следующие условия.

(i) Любая диаграмма из класса Conf имеет копредел.
(ii) Функтор sig унивалентен.
(iii) Функтор sig обладает левым сопряжённым (обозначение: sig^*) с тождественной единицей сопряжения.
(iv) Функтор sig поднимает копределы всех диаграмм из класса Conf.
(v) Для любых c-DESC-диаграмм Δ, Ξ если $\mathrm{sig} \circ \Delta = \mathrm{sig} \circ \Xi$ и $\Delta \in \mathrm{Conf}$, то $\Xi \in \mathrm{Conf}$.
(vi) $\mathrm{Ob}\,\text{r-DESC} = \mathrm{Ob}\,\text{c-DESC}$.
(vii) Подкатегория в c-DESC, состоящая из всех c-DESC-объектов и всех изоморфизмов, является подкатегорией в r-DESC.
(viii) Для любых диаграммы $\Delta\colon X \to \text{c-DESC} \in \mathrm{Conf}$ и естественного преобразования вида $\varphi\colon |\Delta| \to \Sigma \in \mathrm{Mor}\,\text{r-DESC}^{|X|}$ существуют диаграмма $\Delta \oplus \varphi \in \mathrm{Conf}$, содержащая Σ в качестве поддиаграммы, и r-DESC-морфизм $r\colon \mathrm{colim}(\Delta) \to \mathrm{colim}(\Delta \oplus \varphi)$.

На базе этой конструкции мы строим две производных.

Определение 2. Тройка $\langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig} \rangle$, удовлетворяющая условиям (i)—(v), называется *формальной технологией специфицирования*, а пара $\langle \text{c-DESC}, \mathrm{Conf} \rangle$, удовлетворяющая условию (i) — *формальной технологией конфигурирования*.

Докажем высказанное выше утверждение, что выделение интерфейсов перестановочно со сборкой систем. Предварительно сформулируем и докажем одно техническое утверждение.

Предложение 1. *Функтор* fun *детерминирует копределы диаграммы* Δ *тогда и только тогда, когда он поднимает их и диаграмма* $\mathrm{fun} \circ \Delta$ *имеет копредел.*

Доказательство. Необходимость очевидна, докажем достаточность. Пусть δ — произвольный копредел диаграммы Δ, ξ — произвольный копредел диаграммы $\mathrm{fun} \circ \Delta$. По условию диаграмма Δ имеет копредел δ', такой что $\mathrm{fun} \circ \delta' = \xi$. Имеем, что $\delta = \ulcorner i \urcorner \circ \delta'$ для некоторого изоморфизма i, поэтому коконус $\mathrm{fun} \circ \delta = \ulcorner \mathrm{fun}(i) \urcorner \circ \xi$ является копределом диаграммы $\mathrm{fun} \circ \Delta$. Следовательно, функтор fun сохраняет копределы диаграммы Δ, а значит, детерминирует их. $\square$

Предложение 2. *В любой формальной технологии специфицирования либо проектирования функтор* sig *сохраняет копределы всех конфигураций.*

Доказательство. Пусть η и ε — единица и коединица сопряжения $\mathrm{sig}^* \dashv \mathrm{sig}$ из условия (iii) определения 1. Напомним, что η — естественное преобразование функтора 1_{SIG} в $\mathrm{sig} \circ \mathrm{sig}^* = 1_{\mathrm{SIG}}$, причём оно состоит из тождественных SIG-морфизмов, а ε — естественное преобразование функтора $\mathrm{sig}^* \circ \mathrm{sig}$ в $1_{\text{c-DESC}}$. Согласно стандартному определению коединицы через единицу [6, с. 99] имеем $\mathrm{sig}(\varepsilon_S) \circ \eta_{\mathrm{sig}(S)} = 1_{\mathrm{sig}(S)}$ для любого $S \in \mathrm{Ob}\,\text{c-DESC}$, поэтому $\mathrm{sig}(\varepsilon_S) = 1_{\mathrm{sig}(S)}$. А поскольку любой унивалентный функтор отражает мономорфизмы [8, предложение 7.37 (2)], из условия (ii) определения 1 вытекает, что ε_S — мономорфизм. (Одновременно ε_S является и эпиморфизмом, поскольку любой унивалентный функтор отражает эпиморфизмы [8, предложение 7.44].)

Выберем произвольную диаграмму $\Delta \in \mathrm{Conf}$, положим $\Delta^* = \mathrm{sig}^* \circ \mathrm{sig} \circ \Delta$. Поскольку $\mathrm{sig} \circ \Delta^* = \mathrm{sig} \circ \Delta$, по условию (v) определения 1 имеем, что $\Delta^* \in \mathrm{Conf}$, следовательно, ввиду условия (i) определения 1 существует копредел вида $\sigma\colon \Delta^* \to \ulcorner S \urcorner$. Рассмотрим коконус $\sigma^* = \mathrm{sig}^* \circ \mathrm{sig} \circ \sigma\colon \Delta^* \to \ulcorner S^* \urcorner$, где $S^* = \mathrm{sig}^*\bigl(\mathrm{sig}(S)\bigr)$. Пусть $u\colon S \to S^*$ — такая стрелка копредела, что $\sigma^* = \ulcorner u \urcorner \circ \sigma$. Поскольку $\sigma = \ulcorner \varepsilon_S \urcorner \circ \sigma^*$, имеем, что $\varepsilon_S \circ u = 1_S$, так что ε_S, будучи обратимым справа мономорфизмом, является изоморфизмом [8, предложение 7.36], следовательно, σ^* — копредел диаграммы Δ^*. Поскольку функтор sig^* задаёт полное вложение категории SIG в c-DESC, отсюда следует, что коконус $\mathrm{sig} \circ \sigma^* = \mathrm{sig} \circ \sigma$ является копределом SIG-диаграммы $\mathrm{sig} \circ \Delta^* = \mathrm{sig} \circ \Delta$. Таким образом, диаграмма $\mathrm{sig} \circ \Delta$ обладает копределом, поэтому в силу условия (iv) определения 1 и предложения 1 функтор sig сохраняет копределы диаграммы Δ. $\square$

Этот факт позволяет выявить все диаграммы, способные служить конфигурациями. Будем называть SIG-диаграмму Θ sig-*предконфигурацией*, если она имеет копредел и функтор sig поднимает копределы всех диаграмм из класса $\mathbf{D}\mathrm{sig}^{-1}(\{\Theta\}) = \{\Delta \mid \mathrm{sig} \circ \Delta = \Theta\}$. Легко убедиться, что класс Conf обязан иметь вид $\mathbf{D}\mathrm{sig}^{-1}(\mathrm{IC})$ для некоторого класса sig-предконфигураций IC [4, предложение 6].

Как указывалось во введении, формальные технологии сами выступают объектами категорий, конструкции в которых отвечают процедурам разработки сложных технологий. Вопросам инженерии технологий посвящён раздел 4; здесь в качестве подготовки к нему, следуя в общих чертах [4, разд. 6], мы вводим конструкцию морфизма формальных технологий.

Определение 3. *Морфизмом формальной технологии проектирования*

$$\langle \text{c-DESC}_1,\ \mathrm{Conf}_1,\ \mathrm{sig}_1\colon \text{c-DESC}_1 \to \mathrm{SIG}_1,\ \text{r-DESC}_1 \rangle$$

в

$$\langle \text{c-DESC}_2,\ \mathrm{Conf}_2,\ \mathrm{sig}_2\colon \text{c-DESC}_2 \to \mathrm{SIG}_2,\ \text{r-DESC}_2 \rangle$$

называется тройка функторов

$$\langle \mathrm{cm}\colon \text{c-DESC}_1 \to \text{c-DESC}_2,\ \mathrm{sm}\colon \mathrm{SIG}_1 \to \mathrm{SIG}_2,\ \mathrm{rm}\colon \text{r-DESC}_1 \to \text{r-DESC}_2 \rangle,$$

удовлетворяющая следующим условиям:

(i) $\mathrm{cm} \circ \mathrm{Conf}_1 \subseteq \mathrm{Conf}_2$;

(ii) cm сохраняет копределы всех диаграмм из Conf_1;
(iii) $\mathrm{sig}_2 \circ \mathrm{cm} = \mathrm{sm} \circ \mathrm{sig}_1$;
(iv) $\mathrm{rm}(i) = \mathrm{cm}(i)$ для любого $i \in \mathrm{Iso}\,\text{c-DESC}_1$.

Пара функторов $\langle \mathrm{cm}, \mathrm{sm} \rangle$, удовлетворяющая условиям (i)—(iii), называется *морфизмом формальной технологии специфицирования* $\langle \text{c-DESC}_1, \mathrm{Conf}_1, \mathrm{sig}_1 \rangle$ в $\langle \text{c-DESC}_2, \mathrm{Conf}_2, \mathrm{sig}_2 \rangle$. Функтор cm, удовлетворяющий условию условиям (i), (ii), называется *морфизмом формальной технологии конфигурирования* $\langle \text{c-DESC}_1, \mathrm{Conf}_1 \rangle$ в $\langle \text{c-DESC}_2, \mathrm{Conf}_2 \rangle$. Если все составляющие морфизма технологий являются вложениями подкатегорий, то область морфизма называется *подтехнологией* в его кообласти.

$$\begin{array}{ccccc}
\langle \text{c-DESC}_1, & \mathrm{Conf}_1, & \text{c-DESC}_1 \xrightarrow{\mathrm{sig}_1} & \mathrm{SIG}_1, & \text{r-DESC}_1 \rangle \\
{\scriptstyle \mathrm{cm}}\big\downarrow & \mathbf{D}\mathrm{cm} \mid\!\cap & {\scriptstyle \mathrm{cm}}\big\downarrow & \big\downarrow{\scriptstyle \mathrm{sm}} & \big\downarrow{\scriptstyle \mathrm{rm}} \\
\langle \text{c-DESC}_2, & \mathrm{Conf}_2, & \text{c-DESC}_2 \xrightarrow{\mathrm{sig}_2} & \mathrm{SIG}_2, & \text{r-DESC}_2 \rangle
\end{array}$$

Требования к функторам, составляющим морфизм технологий, обеспечивают естественность действия морфизма относительно сборки систем (условия (i), (ii)), выделения интерфейсов (условие (iii)) и трансформаций (условие (iv)). При этом естественность относительно покомпонентной трансформации конфигураций (условие (viii) определения 1) выполняется в следующем смысле. Для любых диаграммы $\Delta\colon X \to \text{c-DESC} \in \mathrm{Conf}_1$ и естественного преобразования вида $\varphi\colon |\Delta| \to \Sigma \in \mathrm{Mor}\,\text{r-DESC}_1^{|X|}$ трансформации из семейства $\mathrm{rm}(\varphi)$ действуют на объекты диаграммы $\mathrm{cm} \circ \Delta \in \mathrm{Conf}_2$, причём диаграмма $\mathrm{cm} \circ (\Delta \oplus \varphi)$ может служить результатом её трансформации (т. е. выступать в качестве $(\mathrm{cm} \circ \Delta) \oplus \mathrm{rm}(\varphi)$), а r-DESC_2-морфизм $\mathrm{rm}(r)$ — соответствующей трансформацией вершины её копредела. Действительно, имеем $\mathrm{rm}(\varphi)\colon |\mathrm{cm} \circ \Delta| \to \mathrm{cm} \circ \Sigma$. Диаграмма $\mathrm{cm} \circ (\Delta \oplus \varphi)$ содержит поддиаграмму $\mathrm{cm} \circ \Sigma$, а поскольку функтор cm сохраняет копределы диаграмм Δ и $\Delta \oplus \varphi$, имеем, что $\mathrm{rm}(r)\colon\ \mathrm{colim}(\mathrm{cm} \circ \Delta) \to \mathrm{colim}\big(\mathrm{cm} \circ (\Delta \oplus \varphi)\big)$.

Приведём несколько примеров формальных технологий. Произвольная категория C порождает тривиальную формальную технологию проектирования $\mathbf{triv}(C) = \langle C, \varnothing, 1_C, (\mathrm{Ob}\,C, \mathrm{Iso}\,C) \rangle$, причём отображение **triv** очевидным образом расширяется до функтора, действующего из **CAT** в категорию **ARCH**, состоящую из всех формальных технологий проектирования и всех их морфизмов. Среди нетривиальных формальных технологий выделим технологии «над» категорией **Set**, которые возникают при формализации многих известных методов моделирования программ. Моделями здесь служат множества, снабжённые некоторой структурой (алгебраические системы, графы и т. п.), а описаниями действий по интеграции — отображения множеств, совместимые со структурой. sig является каноническим функтором взятия основного множества, «забывающим» структуру, а левый сопряжённый к нему создаёт на произвольном множестве «минимальную» (дискретную) структуру. В качестве трансформаций часто выступают антифункции — отношения, задающие «расширение» точек основного множества моделей до подмножеств. Примером служит формальная технология событийного моделирования [3], имеющая вид

$$\mathrm{SM} = \langle \mathbf{Pos}, \mathrm{CPos}, |{-}|, \text{r-Pos} \rangle,$$

где:

— $\mathbf{Pos}$ — категория всех частично упорядоченных множеств и всех их монотонных отображений (моделью служит сценарий — множество событий, частично упорядоченное причинно-следственными связями);
— CPos — наименьший класс $\mathbf{Pos}$-диаграмм, содержащий все точки и замкнутый относительно встраивания коконусов, копроизведений и сингулярных склеек (сложные сценарии можно собирать путём попарной синхронизации составляющих не более чем в одном событии, чтобы не допустить нарушения причинности);
— $|{-}|\colon \mathbf{Pos} \to \mathbf{Set}\colon S \mapsto |S|$ — канонический функтор, «забывающий» порядок (внешний наблюдатель сценария «видит» множество составляющих его событий, но не видит их порядок, в частности, любая процедура интеграции сценариев может быть полностью задана сопоставлением событий);
— r-Pos — категория, объектами которой являются все частично упорядоченные множества, а морфизмом X в Y служит любое антифункциональное тотальное отношение $R \subseteq X \times Y$, такое что

$$\forall\, x, x' \in X \ \forall\, y, y' \in Y \ \big((xRy \wedge x'Ry' \wedge x \neq x') \Rightarrow (x \leqslant x' \Leftrightarrow y \leqslant y')\big)$$

(трансформация сценария заключается в расширении событий до подсценариев с полным наследованием порядка).

Левым сопряжённым к функтору выделения интерфейсов $|{-}|\colon \mathbf{Pos} \to \mathbf{Set}$ служит функтор дискретного упорядочения $\mathrm{dord}\colon \mathbf{Set} \to \mathbf{Pos}\colon I \mapsto \langle I, = \rangle$. На примере технологии SM можно показать «жёсткость» формальной конструкции интеграционного интерфейса: существует единственный (с точностью до эквивалентности категорий) нетривиальный функтор выделения интерфейса у сценария, удовлетворяющий условиям (ii), (iii) определения 1. Действительно, пусть psig — функтор с областью $\mathbf{Pos}$, удовлетворяющий указанным условиям, ω — коединица сопряжения $\mathrm{psig}^* \dashv \mathrm{psig}$. Пара

$$\Big(\mathrm{psig}^*\big(\mathrm{psig}(\mathrm{Ob}\,\mathbf{Pos})\big), \mathrm{psig}^*\big(\mathrm{psig}(\mathrm{Mor}\,\mathbf{Pos})\big)\Big)$$

является полной подкатегорией в $\mathbf{Pos}$, которую мы обозначим через PPos, она изоморфна кообласти функтора psig. Поскольку ω состоит из монотонных биекций, в PPos найдётся дискретно упорядоченное множество произвольной мощности. Предположим, что в PPos найдётся также некоторое множество X с отношением порядка, не совпадающим с равенством, и покажем, что в этом случае ω состоит из изоморфизмов. Существуют элементы $a, b \in X$, такие что $a < b$ (т. е. $a \neq b$ и $a \leqslant b$). Выберем произвольно $\mathbf{Pos}$-объект Y и элементы

$p, q \in \mathrm{psig}^*\big(\mathrm{psig}(Y)\big)$, удовлетворяющие соотношению $\omega_Y(p) \leqslant \omega_Y(q)$. Определим отображение $f\colon X \to Y$ следующим образом:

$$f(x) = \begin{cases} \omega_Y(p), & x \leqslant a, \\ \omega_Y(q) & \text{иначе.} \end{cases}$$

Ясно, что отображение f монотонно, поэтому по определению сопряжения существует монотонное отображение $f^*\colon X \to \mathrm{psig}^*\big(\mathrm{psig}(Y)\big)$, такое что $\omega_Y \circ f^* = f$, откуда получаем, что $p = f^*(a) \leqslant f^*(b) = q$. Ввиду произвольности выбора p и q отсюда следует, что ω_Y — изоморфизм. Следовательно, категория PPos либо состоит из дискретно упорядоченных множеств и эквивалентна категории $\mathbf{Set}$, либо эквивалентна категории $\mathbf{Pos}$.

В качестве более сложного примера приведём формальную технологию автоматизированной разработки алгебраических спецификаций и онтологий SPECWARE [36]. В ней спецификации записываются на языке классического многосортного исчисления первого порядка (расширенного некоторыми конструкциями высшего порядка). Любая его сигнатура представляет собой конечное множество, состоящее из имён сортов и профилей многосортных операций. Сигнатурным морфизмом называется отображение сигнатур $\phi\colon \sigma \to \sigma'$, согласованное с профилями операций в том смысле, что для любой операции $f\colon S_1, \ldots, S_k \to S_0 \in \sigma$ имеем $\phi(f)\colon \phi(S_1), \ldots, \phi(S_k) \to \phi(S_0)$. Все конечные многосортные сигнатуры вместе со всеми сигнатурными морфизмами образуют подкатегорию в $\mathbf{Set}$, которую мы обозначим через MSS.

Сигнатурные морфизмы «поднимаются» на уровень спецификаций следующим образом. Пусть SLS_σ — множество всех конечных множеств предложений языка сигнатуры σ (спецификаций). Обозначим через SLS категорию, класс объектов которой имеет вид $\bigcup\limits_{\sigma \in \mathrm{Ob}\,\mathrm{MSS}} \mathrm{SLS}_\sigma$,

$$\mathrm{Mor}(T, T') = \{\phi\colon \sigma(T) \to \sigma(T') \mid T' \vdash \phi T\}$$

для любых SLS-объектов T, T'. Стандартный закон композиции отображений действительно делает её категорией, поскольку если $T' \vdash \phi T$ и $T'' \vdash \vartheta T'$ для произвольных спецификаций T, T', T'' и сигнатурных морфизмов $\phi\colon \sigma(T) \to \sigma(T')$, $\vartheta\colon \sigma(T') \to \sigma(T'')$, то $T'' \vdash \vartheta\phi T$. Операция взятия сигнатуры спецификации расширяется до функтора $\sigma(-)\colon \mathrm{SLS} \to \mathrm{MSS}$. Этот функтор унивалентен и обладает левым сопряжённым, ставящим в соответствие любой сигнатуре пустую спецификацию этой сигнатуры, причём единица этого сопряжения тождественна.

Категория SLS конечно кополна. Действительно, категория MSS, рассматриваемая как подкатегория в $\mathbf{Set}$, замкнута относительно копределов конечных диаграмм, поскольку, как легко проверить, кодекартов квадрат, построенный в $\mathbf{Set}$ над любой парой MSS-морфизмов, содержится в MSS. В свою очередь, копредел любой конечной диаграммы $\Delta\colon X \to \mathrm{SLS}$ фактически вычисляется на уровне категории MSS: его вершина имеет вид $\ulcorner \bigcup\limits_{I \in \mathrm{Ob}\, X} \varphi_I \Delta(I) \urcorner$, где φ — семейство рёбер копредела MSS-диаграммы $\sigma \circ \Delta$. Иными словами, функтор $\sigma(-)$

поднимает копределы всех конечных SLS-диаграмм, совокупность которых мы обозначим через FinSLS.

Трансформацией SLS-объекта T в T' служит любая пара SLS-морфизмов с общим концом вида $\phi\colon T \to X \leftarrow T' :\phi'$, такая что ϕ' задаёт дефинициальное расширение теории T', т. е. морфизм ϕ' инъективен и в X имеются аксиомы, однозначно определяющие любой символ из $\sigma(X) \setminus \phi'\bigl(\sigma(T')\bigr)$ через символы из $\phi'\bigl(\sigma(T')\bigr)$. Грубо говоря, трансформация превосходит сигнатурный морфизм возможностью удалить из его кообласти производные символы. Тождественная трансформация — это пара тождественных SLS-морфизмов. Композиция трансформаций вычисляется при помощи кодекартова квадрата: если пара $\vartheta\colon T' \to Y \leftarrow T'' :\vartheta'$ также является трансформацией, то их композиция представляет собой пару $\vartheta''\phi\colon T \to Z \leftarrow T'' :\phi''\vartheta'$, где $\vartheta''\colon X \to Z \leftarrow Y :\phi''$ — рёбра кодекартова квадрата $\vartheta''\phi' = \phi''\vartheta$. В [36] доказано, что класс всех дефинициальных расширений замкнут относительно композиций и кодекартовых квадратов, поэтому все SLS-объекты и все их трансформации действительно образуют категорию, которую мы обозначим через r-SLS. Она содержит SLS в качестве (собственной) подкатегории: любому SLS-морфизму $\phi\colon T \to T'$ соответствует трансформация $\phi\colon T \to T' \leftarrow T' :1_{T'}$. Для любой конечной SLS-диаграммы Δ и любого семейства трансформаций её объектов φ в [36] описана процедура построения диаграммы типа $\Delta \oplus \varphi$ из условия (viii) определения 1 и соответствующей трансформации $r\colon \operatorname{colim}(\Delta) \to \operatorname{colim}(\Delta \oplus \varphi)$.

Таким образом, имеется формальная технология проектирования

$$\mathrm{SPECWARE} = \langle \mathrm{SLS},\ \mathrm{FinSLS},\ \sigma(-)\colon \mathrm{SLS} \to \mathrm{MSS},\ \text{r-SLS} \rangle.$$

2. Трассирование задач и аспектно-ориентированный подход

Построим из конструкций в формальных технологиях проектирования теоретико-категорную семантическую модель трассирования и АОП. Зафиксируем произвольную формальную технологию проектирования

$$\mathbf{AR} = \langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig}, \text{r-DESC} \rangle,$$

обозначим через ε коединицу сопряжения $\mathrm{sig}^* \dashv \mathrm{sig}$. Хорошо известно, что трассируемость легко нарушается при трансформациях (хрестоматийным примером является реализация алгебраической спецификации программы на алгоритмическом языке программирования). При интеграции, напротив, обычно обеспечивается хотя бы частичное трассирование (здесь примером служит прямая сумма индексированного семейства множеств в категории $\mathbf{Set}$, при построении которой элементы каждой компоненты снабжаются её индексом — «меткой»). Ввиду этого для произвольной трансформации — r-DESC-морфизма $r\colon S \to T$ — возникает следующее необходимое условие возможности трассировать вдоль неё результат к источнику [4]: обращение её направления, т. е. теоретико-категорная дуализация, должна превращать эту трансформацию в действие по интеграции — в c-DESC-морфизм $r^{\mathrm{op}}\colon T \to S$.

Поскольку в ходе разработки программных систем трансформация перемежается со сборкой систем, необходимо совместно трассировать трансформации и действия по интеграции. Проще всего провести такое совместное трассирование, если трасса r^{op} обратима справа [3]. Действительно, существование c-DESC-морфизма $s\colon S \to T$, такого что $r^{\mathrm{op}} \circ s = 1_S$, эквивалентно тому, что для любого c-DESC-морфизма $p\colon X \to S$, задающего интеграцию некоторого компонента X в систему S, существует действие по интеграции X в T, совместимое с трассированием трансформации r в том смысле, что композиция трассы r^{op} с этим действием даёт p. Таким действием служит $s \circ p$, поскольку $r^{\mathrm{op}} \circ (s \circ p) = p$.

$$\begin{tikzcd} T \\ S \arrow[u, "r"'] \end{tikzcd} \qquad\qquad \begin{tikzcd} & T \arrow[d, "r^{\mathrm{op}}"] \\ X \arrow[ur, dashed, "s\circ p"] \arrow[r, "p"] & S \end{tikzcd}$$

В свою очередь, для любого c-DESC-морфизма $q\colon T \to Y$ морфизм $q \circ s$ описывает действие по интеграции S в Y, совместимое с трассированием в несколько ином смысле: выбор $q = r^{\mathrm{op}}$ превращает его в тождественный морфизм. Заметим, что s является регулярным мономорфизмом — теоретико-категорным аналогом операции включения источника трансформации в результат, не разрушающей его структуру. На практике его построение может быть весьма трудоёмким, но он требуется не всегда, поскольку трассированию вдоль процессов сборки систем подлежат в первую очередь интеграционные требования, которые предъявляются к интерфейсам моделей. В таком случае достаточно потребовать обратимость справа не для трассы r^{op}, а лишь для SIG-морфизма $\mathrm{sig}(r^{\mathrm{op}})$, представляющего действие трассы на уровне интерфейсов и называемого разметкой [3]. Реализация обращения разметки обычно не требует значительных затрат, поскольку интерфейсы проектируются так, чтобы интегрировать их было «проще», чем сами модели. Получается частный случай известного подхода к снижению затрат на трассирование путём ограничения класса трассируемых требований согласно их значимости [17].

Согласно условию (vii) определения 1 тривиальным примером трассируемой трансформации является c-DESC-изоморфизм (напомним, что морфизм, двойственный к изоморфизму, отождествляется с обратным к нему и также является изоморфизмом [8]). Для выявления нетривиальных трассируемых трансформаций воспользуемся следующим приёмом. Рассмотрим совокупность всех общих подкатегорий в c-DESC и r-DESC$^{\mathrm{op}}$, содержащих все c-DESC-изоморфизмы. Она является полной нижней полурешёткой по включению. Обозначим через cr-DESC пересечение всех её максимальных

Определение 4. cr-DESC-морфизм t называется (sig-)*трассой*, если SIG-морфизм $\mathrm{sig}(t)$ является ретракцией (т. е. имеет правый обратный). sig-образ трассы называется (sig-)*разметкой*. r-DESC-морфизм, двойственный к трассе, называется *трассируемой трансформацией*. Трансформация называется *обратимой*, если она трассируема и двойственна к c-DESC-ретракции. Формальная технология $\mathbf{AR}$ *поддерживает трассирование*, если все трансформации в ней обратимы.

Обозначим через tr-DESC пару, состоящую из класса всех c-DESC-объектов и класса всех трасс.

Предложение 3. tr-DESC *является подкатегорией в* c-DESC, *обладающей следующими свойствами.*

(i) $\mathrm{Iso}\,\text{c-DESC} \subseteq \mathrm{Mor}\,\text{tr-DESC} \subseteq \mathrm{Epi}\,\text{c-DESC}$.

(ii) $\mathrm{Iso}\,\mathrm{SIG} \subseteq \mathrm{sig}(\mathrm{Mor}\,\text{tr-DESC}) \subseteq \mathrm{Retract}\,\mathrm{SIG}$.

Доказательство. Любая ретракция является эпиморфизмом [8, предложение 7.42], а любой унивалентный функтор отражает эпиморфизмы [8, предложение 7.44]. □

Трассируемые трансформации «хорошо» действуют на конфигурации. Рассмотрим произвольные диаграммы $\Delta\colon X \to \text{c-DESC} \in \mathrm{Conf}$ и $\Sigma\colon |X| \to \text{c-DESC}$, связанные $\mathbf{D}$c-DESC-морфизмом $\langle \tau, jx\rangle\colon \Sigma \to \Delta$, где τ — некоторое семейство c-DESC-морфизмов, $jx\colon |X| \hookrightarrow X$ — вложение категорий. Обозначим через $\tau \rightrightarrows \Delta$ отрисовку $\mathbf{D}$c-DESC-диаграммы со схемой *2*, изображающей указанный морфизм, и назовём её накачкой диаграммы Δ семейством τ [4, разд. 5]. Накачка не меняет копредела диаграммы Δ, так как сводится к многократному встраиванию коконусов вида $\ulcorner t \urcorner$, где t пробегает семейство τ. Поэтому трассируемые трансформации компонентов являются неразрушающими по отношению к сборке систем: если семейство τ состоит из трасс и накачка является конфигурацией, то условие (viii) определения 1 можно обеспечить, взяв её в качестве $\Delta \oplus \tau^{\mathrm{op}}$. Этот факт мотивирует введение следующего понятия.

Определение 5. Формальная технология $\mathbf{AR}$ называется (трансформационно) *коодногородной*, если $\text{r-DESC} \subseteq \text{c-DESC}^{\mathrm{op}}$ и класс Conf замкнут относительно накачек $\text{r-DESC}^{\mathrm{op}}$-морфизмами.

В формальной технологии над $\mathbf{Set}$ разметки являются сюръективными отображениями, поскольку в силу аксиомы выбора любой $\mathbf{Set}$-эпиморфизм является ретракцией. Действие трансформации $t^{\mathrm{op}}\colon X \to Y$, двойственной по отношению к сюръекции $t\colon |Y| \to |X|$, можно описать как раскрытие всех точек множества $|X|$ в их прообразы относительно t, образующие разбиение множества $|Y|$, с (частичным) переносом структуры объекта X на них. Точку $x \in |X|$ можно рассматривать как обозначение класса задач, реализуемого путём раскрытия, в соответствие с интуитивным пониманием трансформации. Примером служит трансформация сценариев в технологии событийного моделирования SM, разбивающая свой результат на подмножества, хорошо упорядоченные в том смысле, что для любых $x, y, z, u \in Y$, таких что $t(x) = t(y) \neq t(z) = t(u)$, условие $x \leqslant z$

влечёт $y \leqslant u$ (см. [3]). Класс всех разметок сценариев состоит из всех сюръективных отображений множеств. Технология SM поддерживает трассирование и является коодногородной. Более того, как показано в разделе 5, выбор трансформаций в этой технологии оптимален с точки зрения баланса между широтой возможных преобразований моделей и трудоёмкостью их трассирования.

Наиболее прямым и экономным способом обеспечения полной трассируемости является «запоминание» трасс трансформаций вместе с моделями программ, порождёнными ими из классов задач [9]. Примем во внимание, что в инженерии программных систем вообще, и в АОП в частности, интерес представляет в первую очередь влияние трансформаций на интеграционные возможности моделей. Ввиду этого достаточно присоединить к моделям действия трансформаций на уровне интерфейсов, т. е. разметки. (Как мы увидим в разделе 3, возможность «поднять» трансформацию на уровень моделей является каноническим условием разделения ответственности — оформления каждой составляющей задачи единицей модульной архитектуры.) Интеграция и трансформация таких обогащённых моделей должна согласованно выполняться на двух уровнях: модульных основ и аспектных структур. Как указано в [20, разд. 7], в теории категорий имеется специальная конструкция, предназначенная для естественного присоединения действий к объектам — категория запятой [6, разд. 2.6]. Рассмотрим категорию запятой $\mathrm{sig} \downarrow \mathrm{SIG}$. Её объектами являются все пары вида $\langle A, l\colon \mathrm{sig}(A) \to L\rangle$, где A — c-DESC-объект, l — SIG-морфизм. Морфизмом объекта $\langle A_1, l_1\colon \mathrm{sig}(A_1) \to L_1\rangle$ в $\langle A_2, l_2\colon \mathrm{sig}(A_2) \to L_2\rangle$ является любая пара $\langle f\colon A_1 \to A_2,\, b\colon L_1 \to L_2\rangle$, такая что $b \circ l_1 = l_2 \circ \mathrm{sig}(f)$.

$$\begin{tikzcd}
\langle A_1, \arrow[d, "f"'] & \mathrm{sig}(A_1) \arrow[r, "l_1"] \arrow[d, "\mathrm{sig}(f)"'] & L_1\rangle \arrow[d, "b"] \\
\langle A_2, & \mathrm{sig}(A_2) \arrow[r, "l_2"] & L_2\rangle
\end{tikzcd}$$

Определение 6. *Аспектно-ориентированной моделью* (АО-моделью) называется любой $(\mathrm{sig} \downarrow \mathrm{SIG})$-объект $\langle A, l\colon \mathrm{sig}(A) \to L\rangle$, такой что l является sig-разметкой. c-DESC-объект A называется (модульной) *основой* АО-модели, SIG-морфизм l — её (аспектной) *разметкой*, SIG-объект L — её *аспектной структурой*.

Будем обозначать через АО полную подкатегорию в $\mathrm{sig} \downarrow \mathrm{SIG}$, класс объектов которой состоит из всех АО-моделей. Поясним эту конструкцию на примере формальной технологии над **Set**. Напомним, что здесь разметка l — сюръективное отображение, ставящее в соответствие каждой точке множества $|A|$ элемент множества L, обозначающий класс задач, для решения которого она предназначена. В частности, аспектом естественно называть любую АО-модель, в которой L состоит из одного элемента. По существу (с точностью до изоморфизма), разметка является отношением эквивалентности на множестве $|A|$, классы эквивалентности которого отвечают отдельным аспектам. Любое отношение эквивалентности превращает A в корректную АО-модель, так что аспектная структура в общем случае никак не связана с модульной. АО-морфизмом является в точности любой c-DESC-морфизм, сохраняющий это отношение эквивалентности. Поскольку канонический забывающий функтор из категории **Equ** всех множеств с отношением эквивалентности и всех их гомоморфизмов в **Set** является топологическим [8, гл. 21], категория помеченных моделей во многих известных случаях эквивалентна топологической категории над c-DESC. В этих случаях она «наследует» приёмы сборки систем из исходных непомеченных моделей.

Например, в технологии событийного моделирования SM АО-моделью является в точности любое частично упорядоченное мультимножество аспектов. Все помеченные сценарии образуют категорию, эквивалентную топологической категории над **Pos**. Подобный подход к моделированию сценариев был предложен ещё в 1980-х годах [33], однако природа меток и способы их синтеза оставались неясными, поскольку они не рассматривались в контексте АОП. С разнообразными классами помеченных сценариев оперируют технологии инженерии процессов, предлагающие те или иные специализированные нотации: графические, гипертекстовые, сети Петри и др. Кроме того, помеченные сценарии лежат в основе одной из классических семантических моделей АОП, известной как трассовая семантика [15].

Вернёмся к рассмотрению произвольных формальных технологий. Нам потребуется категория морфизмов вида C^{2}, где C — произвольная категория. Напомним, что класс всех C^{2}-объектов состоит из всех C-морфизмов (диаграмм вида $\mathit{2} \to C$), а морфизмом из f в g является любая пара C-морфизмов $\langle u, v\rangle$, такая что $v \circ f = g \circ u$. Имеются функторы $\mathrm{dom}, \mathrm{codom}\colon C^{\mathit{2}} \to C$, переводящие любой C-морфизм в его область и кообласть соответственно.

Пусть LAB — полная подкатегория в $\mathrm{SIG}^{\mathit{2}}$, класс объектов которой состоит из всех sig-разметок, $\mathrm{il}\colon \mathrm{LAB} \hookrightarrow \mathrm{SIG}^{\mathit{2}}$ — вложение. Мы будем использовать следующие «забывающие» функторы, порождённые структурой категории $\mathrm{sig} \downarrow \mathrm{SIG}$:

$$\begin{aligned}
\mathrm{mod}\colon & \mathrm{AO} \to \text{c-DESC}\\
\colon & \langle A, l\rangle \mapsto A,\ \langle f, b\rangle \mapsto f,\\
\mathrm{asp}\colon & \mathrm{AO} \to \mathrm{LAB}\\
\colon & \langle A, l\rangle \mapsto l,\ \langle f, b\rangle \mapsto \langle \mathrm{sig}(f), b\rangle,\\
\mathrm{int} = \mathrm{sig} \circ \mathrm{mod} = \mathrm{dom} \circ \mathrm{il} \circ \mathrm{asp}\colon & \mathrm{AO} \to \mathrm{SIG}\\
\colon & \langle A, l\rangle \mapsto \mathrm{sig}(A),\ \langle f, b\rangle \mapsto \mathrm{sig}(f),\\
\mathrm{str} = \mathrm{codom} \circ \mathrm{il} \circ \mathrm{asp}\colon & \mathrm{AO} \to \mathrm{SIG}\\
\colon & \langle A, l\rangle \mapsto \mathrm{codom}\, l,\ \langle f, b\rangle \mapsto b.
\end{aligned}$$

Мы будем называть *фундаментальными* функторы 1_{AO}, mod, asp и int. С их помощью можно построить АО как универсальную конструкцию в категории **CAT** (ср. [6, с. 61]): непосредственно проверяется, что равенство $\mathrm{sig} \circ \mathrm{mod} = (\mathrm{dom} \circ \mathrm{il}) \circ \mathrm{asp}$, определяющее функтор int, задаёт декартов квадрат (предел) пары функторов $\mathrm{sig}\colon \text{c-DESC} \to \mathrm{SIG} \leftarrow \mathrm{LAB} :\! \mathrm{dom} \circ \mathrm{il}$, обладающий вершиной

АО. Отсюда вытекает, что обогащение моделей программ аспектными разметками можно выполнить единственным образом (с точностью до изоморфизма).

$$\begin{tikzcd} \mathrm{AO} \arrow[r, dashed, "\mathrm{asp}"] \arrow[d, dashed, "\mathrm{mod}"'] & \mathrm{LAB} \arrow[d, "\mathrm{dom} \circ \mathrm{il}"] \\ \text{c-DESC} \arrow[r, "\mathrm{sig}"] & \mathrm{SIG} \end{tikzcd}$$

Фундаментальные функторы имеют большое значение для формализации аспектно-ориентированного проектирования: ниже мы докажем (теорема 1), что они позволяют извлекать из АО-моделей различные интеграционные интерфейсы в смысле определения 1, т. е. порождают формальные технологии проектирования. Разные интерфейсы требуются для выработки разных видов проектных решений: модульный интерфейс, извлекаемый функтором mod, играет ключевую роль при модуляризации аспектов (см. раздел 3), аспектный, извлекаемый функтором asp — при конструировании разметок, исходный, извлекаемый функтором int — при специфицировании интеграционных требований к моделям без детализации их модульной или аспектной структуры. В общем случае возможны и другие виды интерфейсов, «уточняющие» исходные интерфейсы из категории SIG, т. е. естественным образом отображающиеся в них.

В то же время, как мы увидим далее (предложение 4), аспектная структура, выделяемая функтором str, не может служить интерфейсом для нетривиальных АО-моделей. Функтор str формально выражает «квинтэссенцию» аспектно-ориентированного расширения технологий проектирования систем, не сводимую к понятиям модульного подхода. Примечательно, что функтор str сюръективен, т. е. любая аспектная структура реализуется в подходящей АО-модели (следствие 1.4). Как показано в [30], для формального определения ключевых понятий аспекта и связывания в дополнение к конструкции категории АО требуется только этот функтор. Напомним, что связыванием называется сборка программы из аспектов, поскольку она выходит за рамки традиционной компоновки модулей В разделе 3 мы формально опишем связывание универсальной конструкцией в категории АО — копределом диаграммы соединения специального вида.

Зададим правила построения конфигураций и трансформаций АО-моделей из модульного «материала». Трансформации естественным образом получаются как двойственные к трассам (нетрассируемые трансформации технологии **AR** полностью игнорируются). Положим

$$\text{tr-AO} = \big(\mathrm{Ob}\,\mathrm{AO}, \mathrm{mod}^{-1}(\mathrm{Mor}\,\text{tr-DESC})\big).$$

В силу предложения 3 tr-AO является категорией.

В качестве конфигураций АО-моделей целесообразно выбрать АО-диаграммы, копределы которых согласованно вычисляются на уровне модульных основ и аспектных структур и которые расширяют модульные конфигурации. Дополнительно нужно обеспечить свойство кооднородности, поскольку оно гарантирует неразрушаемость при покомпонентных трансформациях систем.

Определение 7. АО-диаграмма Δ называется *аспектно детерминированной*, если функтор $\langle \mathrm{mod}, \mathrm{str}\rangle\colon \mathrm{AO} \to \text{c-DESC} \times \mathrm{SIG}$ детерминирует её копределы. Пусть ai — произвольный функтор, действующий из АО в произвольную категорию INT. Класс INT-диаграмм AIDia называется *аспектно замкнутым* (относительно ai), если класс АО-диаграмм $\mathbf{D}\mathrm{ai}^{-1}(\mathrm{AIDia})$ состоит из аспектно детерминированных диаграмм, содержится в классе $\mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf})$ и замкнут относительно накачек tr-АО-морфизмами.

Будем обозначать через $\mathrm{AOInt}_{\mathrm{ai}}$ объединение всех аспектно замкнутых классов INT-диаграмм. Ясно, что оно также является аспектно замкнутым классом.

Определение 8. Произвольный функтор ai с областью АО называется *порождающим аспектно-ориентированную формальную технологию* (АО-технологию) над $\mathbf{AR}$, если четвёрка

$$\mathrm{AO}_{\mathrm{ai}}(\mathbf{AR}) = \langle \mathrm{AO}, \mathbf{D}\mathrm{ai}^{-1}(\mathrm{AOInt}_{\mathrm{ai}}), \mathrm{ai}, \text{tr-AO}^{\mathrm{op}}\rangle$$

является формальной технологией проектирования и существует функтор si, такой что $\mathrm{si} \circ \mathrm{ai} = \mathrm{int}$.

Теорема 1. *Любой фундаментальный функтор порождает АО-технологию над* $\mathbf{AR}$.

Доказательство. Условия (i), (v)—(viii) определения 1 выполняются для четвёрки $\mathrm{AO}_{\mathrm{ai}}(\mathbf{AR})$ при любом выборе функтора ai с областью АО. Остаётся проверить условия (ii)—(iv) для функторов mod и asp, поскольку для функтора 1_{AO} они выполняются очевидным образом, а для функтора int — ввиду их устойчивости относительно композиции функторов.

Проверим условие (ii). Функтор $\mathrm{dom} \circ il$ унивалентен: для произвольных LAB-морфизмов $\langle f, b\rangle, \langle f, b'\rangle\colon l \to k$ выполняется равенство $b' \circ l = k \circ f = b \circ l$, откуда следует, что $b' = b$, поскольку l обратим справа. Поэтому функтор mod, будучи ребром декартова квадрата в $\mathbf{CAT}$, параллельным $\mathrm{dom} \circ \mathrm{il}$, унивалентен. Аналогично функтор asp, параллельный унивалентному функтору sig, унивалентен.

Проверим условие (iii). Функтор дискретной разметки

$$\begin{aligned} \mathrm{mod}^*\colon & \text{c-DESC} \to \mathrm{AO} \\ & \colon A \mapsto \langle A, 1_{\mathrm{sig}(A)}\rangle,\ f \mapsto \langle f, \mathrm{sig}(f)\rangle \end{aligned}$$

является левым сопряжённым к mod с тождественной единицей. Функтор

$$\begin{aligned} \mathrm{asp}^*\colon & \mathrm{LAB} \to \mathrm{AO} \\ & \colon l \mapsto \langle \mathrm{sig}^*(\mathrm{dom}\, l), l\rangle,\ \langle p, q\rangle \mapsto \langle \mathrm{sig}^*(p), q\rangle \end{aligned}$$

является левым сопряжённым к asp с тождественной единицей.

Проверим условие (iv). Согласно определению 7 функтор mod поднимает копределы конфигураций. Чтобы показать, что функтор asp также поднимает их, рассмотрим произвольную диаграмму $\Delta \in \mathbf{D}\mathrm{asp}^{-1}(\mathrm{AOInt}_{\mathrm{asp}})$ и копредел $\sigma\colon \mathrm{asp} \circ \Delta \to \ulcorner k \urcorner$. Выберем произвольно копредел $\delta\colon \Delta \to \langle A, l\rangle$. Ввиду определения 7 $\mathrm{str} \circ \delta$ — копредел диаграммы $\mathrm{str} \circ \Delta$ и $\mathrm{mod} \circ \delta$ — копредел диаграммы

$\mathrm{mod} \circ \Delta$, так что и $\mathrm{int} \circ \delta$ — копредел диаграммы $\mathrm{int} \circ \Delta$, поскольку ввиду предложения 2 функтор sig сохраняет копределы конфигураций. Поэтому непосредственно проверяется, что l — вершина копредела диаграммы $\mathrm{il} \circ \mathrm{asp} \circ \Delta$ в категории стрелок $\mathrm{SIG}^{\mathbf{2}}$, так что имеется копредел $\mathrm{asp} \circ \delta\colon \mathrm{asp} \circ \Delta \to \ulcorner l \urcorner$, поскольку l — sig-разметка. Поскольку копредел определён однозначно с точностью до изоморфизма, существует LAB-изоморфизм i, такой что $\sigma = \ulcorner i \urcorner \circ (\mathrm{asp} \circ \delta)$, поэтому коконус $\mathrm{dom} \circ \mathrm{il} \circ \sigma = \ulcorner \mathrm{dom}\big(\mathrm{il}(i)\big) \urcorner \circ (\mathrm{int} \circ \delta)$ является копределом диаграммы $\mathrm{int} \circ \Delta$. А поскольку функтор sig поднимает копределы конфигураций, существует копредел $\theta\colon \mathrm{mod} \circ \Delta \to \ulcorner B \urcorner$, такой что $\mathrm{sig} \circ \theta = \mathrm{dom} \circ \mathrm{il} \circ \sigma$. Тогда $\langle \theta, \mathrm{codom} \circ \mathrm{il} \circ \sigma \rangle\colon \Delta \to \ulcorner \langle B, k \rangle \urcorner$ — копредел. □

Следствие 1.1. *Произвольный функтор* $\mathrm{ai}\colon \mathrm{AO} \to \mathrm{INT}$ *порождает АО-технологию над* $\mathbf{AR}$ *тогда и только тогда, когда он удовлетворяет следующим условиям.*

(i) *Функтор* ai *обладает левым сопряжённым с тождественной единицей.*

(ii) $\mathrm{int} \circ \mathrm{ai}^* \circ \mathrm{ai} = \mathrm{int}$, *где* ai^* — *левый сопряжённый из условия* (i).

(iii) *Любой аспектно замкнутый класс* INT-*диаграмм состоит из* ai-*предконфигураций.*

Доказательство. Необходимость вытекает из определения 8 с учётом того, что

$$\mathrm{si} = \mathrm{si} \circ (\mathrm{ai} \circ \mathrm{ai}^*) = \mathrm{int} \circ \mathrm{ai}^*$$

для любого функтора si, удовлетворяющего условию $\mathrm{si} \circ \mathrm{ai} = \mathrm{int}$. Достаточность вытекает из того, что условия (i), (v)—(viii) определения 1 выполняются при любом выборе функтора ai (ср. с доказательством теоремы 1), а условие (ii) гарантируется унивалентностью функтора int, поскольку функтор ai является первой компонентой его разложения в композицию (см. [8, предложение 3.30 (2)]). □

Следствие 1.2. *Любая конфигурация технологии* $\mathrm{AO}_{\mathrm{int}}(\mathbf{AR})$ *является конфигурацией в любой АО-технологии над* $\mathbf{AR}$. *В свою очередь, любая конфигурация любой АО-технологии над* $\mathbf{AR}$ *является конфигурацией в* $\mathrm{AO}_{1_{\mathrm{AO}}}(\mathbf{AR})$.

Доказательство. Пусть $\mathrm{ai}\colon \mathrm{AO} \to \mathrm{INT}$ — произвольный функтор, порождающий АО-технологию над $\mathbf{AR}$, $\mathrm{si}\colon \mathrm{INT} \to \mathrm{SIG}$ — функтор, удовлетворяющий условию $\mathrm{si} \circ \mathrm{ai} = \mathrm{int}$. Выберем произвольно АО-диаграмму $\Delta \in \mathbf{D}\mathrm{int}^{-1}(\mathrm{AOInt}_{\mathrm{int}})$, положим $\Theta = \mathrm{int} \circ \Delta$, $\Xi = \mathrm{ai} \circ \Delta$. Любая АО-диаграмма $\Delta' \in \mathbf{D}\mathrm{ai}^{-1}(\{\Xi\})$ аспектно детерминирована и содержится в классе $\mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf})$, поскольку $\mathrm{int} \circ \Delta' = \mathrm{si} \circ \Xi = \Theta \in \mathrm{AOInt}_{\mathrm{int}}$. Обозначим через $\mathrm{ai} \rightrightarrows \Omega$ класс $\mathbf{D}$ai-образов всех накачек АО-диаграммы Ω tr-АО-морфизмами. Любая АО-диаграмма $\Delta'' \in \mathbf{D}\mathrm{ai}^{-1}(\mathrm{ai} \rightrightarrows \Delta')$ также аспектно детерминирована и содержится в $\mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf})$, поскольку

$$\mathrm{int} \circ \Delta'' \in \mathrm{si} \circ (\mathrm{ai} \rightrightarrows \Delta') \subseteq \mathrm{int} \rightrightarrows \Delta' \subseteq \mathrm{AOInt}_{\mathrm{int}}.$$

Рассуждая по индукции, определим последовательность классов INT-диаграмм PDia, полагая

$$\mathrm{PDia}_0 = \{\Xi\}, \quad \mathrm{PDia}_{n+1} = \bigcup_{\Omega \in \mathbf{D}\mathrm{ai}^{-1}(\mathrm{PDia}_n)} \mathrm{ai} \rightrightarrows \Omega.$$

Класс $\bigcup\limits_{n \geqslant 0} \mathrm{PDia}_n$ аспектно замкнут, а поскольку он содержит Ξ, имеем, что $\Delta \in \mathbf{D}\mathrm{ai}^{-1}(\mathrm{AOInt}_{\mathrm{ai}})$. Первое утверждение доказано.

Второе утверждение вытекает непосредственно из того, что класс $\mathbf{D}\mathrm{ai}^{-1}(\mathrm{AOInt}_{\mathrm{ai}})$ аспектно замкнут относительно функтора 1_{AO}. □

Следствие 1.3. *Класс* $\mathrm{Mor}\,\mathrm{tr}\text{-}\mathrm{AO}$ *состоит из всех трасс технологии* $\mathrm{AO}_{\mathrm{int}}(\mathbf{AR})$.

Следствие 1.4. *Функтор* str *обратим справа.*

Доказательство. Функтор $1_{-}\colon \mathrm{SIG} \hookrightarrow \mathrm{LAB}$: $A \mapsto 1_A$, $f \mapsto \langle f, f\rangle$ является правым обратным к функтору $\mathrm{codom} \circ \mathrm{il}$, а функтор asp обратим справа по теореме 1. □

Существуют технологии проектирования, аспектно-ориентированное расширение которых не привносит ничего существенно нового. Покажем формально, что они характеризуются отсутствием трассируемых трансформаций, способных порождать нетривиальную аспектную структуру: любая разметка в них оказывается изоморфизмом. Этот критерий эквивалентен ряду других, таких как способность функтора str служить для выделения интерфейсов AO-моделей.

Определение 9. Формальная технология проектирования $\mathbf{AR}$ называется *аспектно тривиальной*, если функтор mod является эквивалентностью категорий AO и c-DESC.

Предложение 4. *Следующие утверждения эквивалентны для любой формальной технологии* $\mathbf{AR}$.

(i) *Формальная технология* $\mathbf{AR}$ *аспектно тривиальна.*
(ii) *Любая разметка является изоморфизмом.*
(iii) *Функтор* int *естественно изоморфен функтору* str.
(iv) *Функтор* str *унивалентен.*

Доказательство. Мы будем пользоваться свойствами эквивалентности категорий, установленными в [6, разд. 4.4]. Обозначим через μ коединицу сопряжения $\mathrm{mod}^* \dashv \mathrm{mod}$. Зафиксируем произвольную sig-разметку $l\colon X \to L$.

Докажем импликацию (i) $\Longrightarrow$ (ii). Проверка условия (iii) в доказательстве теоремы 1 даёт, что $l = \mathrm{str}(\mu_{\mathrm{asp}^*(l)})$. По предположению μ состоит из изоморфизмов, следовательно, l — изоморфизм.

Докажем импликацию (ii) $\Longrightarrow$ (i). Согласно той же проверке пункта (iii) в доказательстве теоремы 1 из предположения вытекает, что μ состоит из изоморфизмов.

Докажем импликацию (ii) $\Longrightarrow$ (iii). Естественное преобразование функтора int в str, состоящее из разметок всех АО-моделей, по предположению состоит из изоморфизмов.

Докажем импликацию (iii) $\Longrightarrow$ (iv). Любой функтор, естественно изоморфный унивалентному, унивалентен.

Докажем импликацию (iv) $\Longrightarrow$ (ii). Рассмотрим АО-морфизм $l^* = \mathrm{asp}^*(\langle l, 1_L \rangle)$. Поскольку $\mathrm{str}(l^*) = 1_L$, по предположению $l^* \in \mathrm{Mono\,AO}$ (любой унивалентный функтор отражает мономорфизмы [8, предложение 7.37 (2)]). Тогда $l = \mathrm{int}(l^*) \in \mathrm{Mono\,SIG}$ (любой функтор, имеющий левый сопряжённый, сохраняет все пределы [6, разд. 5.5], в частности мономорфизмы). Поскольку l обратим справа, получаем, что $l \in \mathrm{Iso\,SIG}$. □

Например, легко проверить, что условие (ii) предложения 4 выполняется для технологии SPECWARE, так что она аспектно тривиальна: в алгебраическом подходе к проектированию аспектно-ориентированные приёмы традиционно не применяются.

3. Связывание, разделение ответственности и экспликация аспектной структуры

Опишем мегамодель процедуры связывания системы из аспектов как универсальную конструкцию в АО-технологии [30]. В классическом АОП связывание состоит в подключении программы аспекта, называемой советом, к базовой программе в заданных местах, называемых точками соединения [11]. Каждый раз, когда при исполнении базовой программы встречается точка соединения, вызывается совет. Поэтому совет обычно выглядит как блок программного кода, охраняемый условием, идентифицирующим точку соединения, а начало блока служит точкой вызова совета. Таким образом, инструмент связывания принимает на вход две спецификации:

— описание точек соединения в базовой программе, или срез, часто задаваемый в форме регулярного выражения над синтаксическими конструкциями языка, на котором написана базовая программа;
— описание точек вызова совета в точках соединения.

При связывании на первом шаге (виртуально) создаётся достаточное количество копий совета, по одной на каждую точку соединения, с маркировкой соответствующих им точек вызова. Далее на втором шаге эти точки «склеиваются» друг с другом так, чтобы не разрушить аспектную структуру базы и совета. Для формальной записи правил связывания привлекается модель C, называемая связкой [32], в дополнение к моделям базы B и совета W. Связка интегрируется с базой в точках соединения, а с советом — в точках вызова, порождая пару АО-морфизмов $j\colon B \leftarrow C \rightarrow W :e$. (Здесь наглядно проявляется отличие связывания от модульной компоновки, формализуемой одношаговым действием вида $l\colon M \to S$, где M — модуль, S — система.) Как легко убедиться на примере АО-технологии событийного моделирования, первому шагу связывания отвечает произведение $C \times W$, а второму — кодекартов квадрат (соединение) над парой морфизмов $j\colon B \leftarrow C \rightarrow C \times W :\langle 1_C, e\rangle$. Эти операции должны быть естественными относительно выделения модульной основы и аспектной структуры, поэтому накладывается требование типа аспектной детерминированности (определение 7): функтор $\langle \mathrm{mod}, \mathrm{str}\rangle\colon \mathrm{AO} \rightarrow \mathrm{c\text{-}DESC} \times \mathrm{SIG}$ должен детерминировать как произведение, так и кодекартов квадрат. При выполнении этого требования результатом связывания является вершина указанного кодекартова квадрата, обозначаемая через $j \bowtie e$.

$$\begin{tikzcd} C \arrow[r, "\langle 1_C, e\rangle"] \arrow[d, "j"'] & C \times W \arrow[d, dashed] \\ B \arrow[r, dashed] & j \bowtie e \end{tikzcd}$$

Ясно, что результат связывания, если он существует, определяется однозначно с точностью до изоморфизма. Кроме того, поскольку АО-морфизм $\langle 1_C, e\rangle\colon C \rightarrow C \times W$ обратим слева (имеем $\pi_C \circ \langle 1_C, e\rangle = 1_C$, где $\pi_C\colon C \times W \rightarrow C$ — проекция), параллельное ему ребро кодекартова квадрата также обратимо слева (см. двойственное утверждение в [8, предложение 11.18]), т. е. база неразрушающим образом вкладывается в результат связывания. Ещё одним важным свойством связывания, легко проверяемым по вышеприведённой диаграмме, является независимость результата привязывания нескольких взаимно независимых советов к общей базе от порядка их связывания.

Технологии АОП предлагают разнообразные инструменты связывания, включая средства диспетчеризации вызова методов, преобразователи исполняемого байт-кода, генераторы исходных текстов. Однако без них можно обойтись, упрощая сборку сложных программ, если подключаемые аспекты можно модуляризовать, т. е. оформить единицами модульной архитектуры: классами (элементами объектно-ориентированной декомпозиции), таблицами в базе данных и т. д. Модуляризация всех аспектов, составляющих АО-модель, является наиболее явным способом их разметки, называется разделением ответственности и относится к числу классических проблем программной инженерии.

Модуляризируемые АО-модели выделяются среди прочих тем, что при интеграции с модулями они ведут себя так же, как модульные единицы. Возможности интеграции модулей в АО-модель определяются её модульной основой. В свою очередь, возможности интеграции АО-модели в модули определяются её аспектной структурой, поскольку при интеграции в модуль аспекты, составляющие АО-модель, выступают в качестве её элементарных единиц. Формально, модуляризируемые модели образуют полную подкатегорию в АО (мы будем обозначать её через m-АО), такую что существует аспектно-ориентированное расширение (АО-расширение) модульной технологии — вложение $\mathrm{am}\colon \mathrm{c\text{-}DESC} \hookrightarrow \mathrm{m\text{-}AO}$, полностью воспроизводящее интеграционные возможности модулей в следующем смысле. С одной стороны, все способы интеграции модуля $M \in \mathrm{Ob}\,\text{c-DESC}$ в АО-модель $A \in \mathrm{Ob}\,\text{m-AO}$ задаются множеством морфизмов $\mathrm{Mor}(\mathrm{am}(M), A)$, поэтому функтор выделения модульного интерфейса mod (точнее, его ограничение на m-AO, обозначаемое далее через am_*) должен устанавливать биекцию между множеством $\mathrm{Mor}(\mathrm{am}(M), A)$ и множеством $\mathrm{Mor}\big(M, \mathrm{mod}(A)\big)$. С другой стороны, все способы интеграции АО-модели A в модуль M задаются множеством морфизмов $\mathrm{Mor}\big(A, \mathrm{am}(M)\big)$, поэтому должен существовать функтор модуляризации аспектной структуры $\mathrm{am}^*\colon \text{m-AO} \to \text{c-DESC}$, тривиально действующий на модули ($\mathrm{am}^* \circ \mathrm{am} = 1_{\text{c-DESC}}$) и устанавливающий биекцию между множествами $\mathrm{Mor}\big(A, \mathrm{am}(M)\big)$ и $\mathrm{Mor}(\mathrm{am}^*(A), M)$. При этом возможность трактовать c-DESC-объект $\mathrm{am}^*(A)$ как «подъём» аспектной структуры АО-модели A на модульный уровень обеспечивается следующим дополнительным требованием естественности: функтор $\mathrm{sig} \circ \mathrm{am}^*$, выделяющий интеграционный интерфейс из модуляризированной аспектной структуры, должен совпадать с ограничением на m-AO функтора str, выявляющего исходную аспектную структуру на уровне интеграционных интерфейсов.

Примером АО-расширения, которым обладает любая формальная технология проектирования, служит изоморфизм, действующий как функтор mod^* между c-DESC и полной подкатегорией в AO с классом объектов $\{\langle A, 1_{\mathrm{sig}(A)}\rangle \mid A \in \mathrm{Ob}\,\text{c-DESC}\}$. В технологиях над **Set** он порождает дискретно размеченные модели, которые являются самыми «аспектно-неориентированными»: в них каждый класс задач помечает только один элемент основного множества, так что никакого рассеяния не происходит. Мы назовём такое АО-расширение тривиальным и покажем, что действие любого АО-расширения по существу (с точностью до естественного изоморфизма) совпадает с его действием: модули (т. е. c-DESC-объекты) всегда переходят в АО-модели, история получения интерфейсов которых из классов задач путём трансформации утрачена (тривиальна). Таким образом, АО-расширение по существу однозначно определяется своей кообластью — классом всех модуляризируемых АО-моделей.

На языке теории категорий требования к АО-расширению компактно формулируются при помощи конструкции сопряжения функторов.

Определение 10. Функтор $\mathrm{am}\colon \text{c-DESC} \to \text{m-AO}$, где m-AO — некоторая полная подкатегория в AO, называется *аспектно-ориентированным расширением* (АО-расширением) формальной технологии, если он обладает следующими сопряжёнными функторами:

- правым сопряжённым am_* с тождественной единицей, причём $\mathrm{am}_*(f) = \mathrm{mod}(f)$ для любого m-AO-морфизма f;
- левым сопряжённым am^* с тождественной коединицей, причём $\mathrm{sig}\big(\mathrm{am}^*(f)\big) = \mathrm{str}(f)$ для любого m-AO-морфизма f.

При выполнении указанных условий m-AO-объекты называются (am-)*модуляризируемыми* АО-моделями.

Зафиксируем произвольное АО-расширение $\mathrm{am}\colon \text{c-DESC} \to \text{m-AO}$. Пусть $\mathrm{mao}\colon \text{m-AO} \hookrightarrow \text{AO}$ — полное вложение.

Предложение 5. *АО-расширение* am *является полным вложением, обратимым слева*: $\mathrm{am}_* \circ \mathrm{am} = \mathrm{am}^* \circ \mathrm{am} = 1_{\text{c-DESC}}$.

Предложение 6. *Функтор* $\mathrm{mao} \circ \mathrm{am}$ *естественно изоморфен функтору* mod^*.

Доказательство. По определению для любых $X, Y \in \mathrm{am}(\mathrm{Ob}\,\text{c-DESC}) \subseteq \subseteq \mathrm{Ob}\,\text{m-AO}$ и любого АО-морфизма $f\colon X \to Y$ имеем

$$f = \mathrm{am}\bigl(\mathrm{am}_*(f)\bigr) = \mathrm{am}\bigl(\mathrm{mod}(f)\bigr),$$

поэтому ввиду предложения 5

$$\mathrm{str}(f) = \mathrm{sig}\bigl(\mathrm{am}^*(f)\bigr) = \Bigl(\mathrm{sig}\Bigl(\mathrm{am}^*\bigl(\mathrm{am}(\mathrm{mod}(f))\bigr)\Bigr)\Bigr) = \mathrm{int}(f).$$

Выберем произвольно SIG-объект I, положим $T = \mathrm{am}\bigl(\mathrm{sig}^*(I)\bigr)$, $l = \mathrm{asp}(T)$. Имеем

$$\mathrm{mod}(T) = \mathrm{am}_*\Bigl(\mathrm{am}\bigl(\mathrm{sig}^*(I)\bigr)\Bigr) = \mathrm{sig}^*(I)$$

и $\mathrm{str}(T) = \mathrm{int}(T) = I$, поэтому $\mathrm{dom}\, l = \mathrm{codom}\, l = I$. Для произвольного SIG-морфизма m, такого что $l \circ m = 1_I$, непосредственно проверяется существование АО-морфизма $\langle \mathrm{sig}^*(m \circ l), 1_I \rangle\colon T \to T$, так что $1_I = \mathrm{sig}\bigl(\mathrm{sig}^*(m \circ l)\bigr) = m \circ l$, поэтому $l \in \mathrm{Iso}\,\mathrm{SIG}$.

Для произвольного c-DESC-объекта P положим $k = \mathrm{asp}\bigl(\mathrm{am}(P)\bigr)$, $P^* = = \mathrm{sig}^*\bigl(\mathrm{sig}(P)\bigr)$, $i = \mathrm{asp}\bigl(\mathrm{am}(P^*)\bigr)$. Как мы только что доказали, $i \in \mathrm{Iso}\,\mathrm{SIG}$, так что непосредственно проверяется существование АО-морфизма $\langle \varepsilon_P, k \circ i^{-1} \rangle$: $\mathrm{am}(P^*) \to \mathrm{am}(P)$. Поэтому $k \circ i^{-1} = \mathrm{sig}(\varepsilon_P) = 1_{\mathrm{sig}(P)}$, откуда следует, что $k = i$. (Напомним, что через ε обозначается коединица сопряжения $\mathrm{sig}^* \dashv \mathrm{sig}$.) Следовательно, семейство АО-изоморфизмов $\langle 1_P, \mathrm{asp}\bigl(\mathrm{am}(P)\bigr) \rangle\colon \langle P, 1_{\mathrm{sig}(P)} \rangle \to \mathrm{am}(P)$, $P \in \mathrm{Ob}\,\text{c-DESC}$, образует естественный изоморфизм функтора mod^* в $\mathrm{mao} \circ \mathrm{am}$. □

Таким образом, АО-расширение совместимо и со сборкой систем (ввиду предложения 6 оно сохраняет копределы всех c-DESC-диаграмм), и с выделением интерфейсов (ввиду предложения 5 имеем $\mathrm{int}\bigl(\mathrm{am}(-)\bigr) = \mathrm{sig}\Bigl(\mathrm{am}_*\bigl(\mathrm{am}(-)\bigr)\Bigr) = = \mathrm{sig}(-)$), и с трассированием (ввиду предложения 5 оно переводит трассу любой трассируемой трансформации в tr-АО-морфизм).

Покажем, что категория модуляризируемых АО-моделей способна служить основной категорией моделей подтехнологии в $\mathrm{AO}_{\mathrm{mod}}(\mathbf{AR})$. Положим mtr-AO = = m-AO ∩ tr-AO. Обозначим через DAOExt класс всех c-DESC-диаграмм $\Theta \in \mathrm{AOInt}_{\mathrm{mod}}$ (см. определение 7), таких что любая АО-диаграмма из класса $\mathrm{mao} \circ \mathbf{D}\mathrm{am}_*^{-1}(\{\Theta\})$ имеет копредел, принадлежащий классу $\mathrm{mao} \circ \mathbf{D}\mathrm{am}_*^{-1}(\{\theta\})$, для любого копредела θ диаграммы Θ.

Определение 11. *Модуляризируемой АО-технологией* над произвольной формальной технологией проектирования $\mathbf{AR}$, порождённой АО-расширением am, называется четвёрка

$$\text{m-AO}_{\mathrm{am}}(\mathbf{AR}) = \langle \text{m-AO}, \mathbf{D}\mathrm{am}_*^{-1}(\mathrm{DAOExt}), \mathrm{am}_*, \text{mtr-AO}^{\mathrm{op}} \rangle.$$

Предложение 7. *Для любых формальной технологии* $\mathbf{AR}$ *и АО-расширения* am *четвёрка* $\text{m-AO}_{\mathrm{am}}(\mathbf{AR})$ *является коолнородной формальной технологией проектирования, наибольшей среди подтехнологий в* $\mathrm{AO}_{\mathrm{mod}}(\mathbf{AR})$*, обладающих категорией моделей* m-AO *и функтором выделения интерфейсов* am_**. Технология* $\text{m-AO}_{\mathrm{am}}(\mathbf{AR})$ *совпадает с* $\mathrm{AO}_{\mathrm{mod}}(\mathbf{AR})$ *тогда и только тогда, когда* $\text{m-AO} = \mathrm{AO}$.

Доказательство. Условие (ii) определения 1 вытекает из теоремы 1, условие (iii) — из определения 10. Условия (i), (iv)—(vii) выполняются по построению. Условие (viii) вместе с коолнородностью вытекает из аспектной замкнутости класса DAOExt (в смысле определения 7). Тройка $\langle \mathrm{mao}, 1_{\text{c-DESC}}, \mathrm{mao}(-^{\mathrm{op}})^{\mathrm{op}}\rangle$ задаёт морфизм формальной технологии проектирования $\text{m-AO}_{\mathrm{am}}(\mathbf{AR})$ в $\mathrm{AO}_{\mathrm{mod}}(\mathbf{AR})$. В любой подтехнологии, вложение которой в $\mathrm{AO}_{\mathrm{mod}}(\mathbf{AR})$ задаётся такой тройкой, все конфигурации содержатся в классе $\mathbf{D}\mathrm{am}_*^{-1}(\mathrm{DAOExt})$, а все трансформации — в $\mathrm{Mor}\,\text{mtr-AO}^{\mathrm{op}}$. $\square$

При формальном описании модуляризации аспектов, составляющих АО-модели, важную роль играет единица сопряжения $\mathrm{am}^* \dashv \mathrm{am}$, которая будет обозначаться через χ. По определению для любого m-AO-объекта S m-AO-морфизм $\chi_S\colon S \to \mathrm{am}\big(\mathrm{am}^*(S)\big)$ является прообразом c-DESC-морфизма $1_{\mathrm{am}^*(S)}$ при биекции

$$\mathrm{am}^*\colon \mathrm{Mor}\Big(S, \mathrm{am}\big(\mathrm{am}^*(S)\big)\Big) \cong \mathrm{Mor}\big(\mathrm{am}^*(S), \mathrm{am}^*(S)\big),$$

поэтому

$$\mathrm{str}(\chi_S) = \mathrm{sig}\big(\mathrm{am}^*(\chi_S)\big) = 1_{\mathrm{sig}(\mathrm{am}^*(S))} = 1_{\mathrm{str}(S)}$$

ввиду определения 10, т. е. действие морфизма χ_S нетривиально только на уровне модульной основы. А поскольку c-DESC-объект $\mathrm{am}^*(S)$ представляет на модульном уровне аспектную структуру АО-модели S, c-DESC-морфизм $\mathrm{am}_*(\chi_S)\colon \mathrm{am}_*(S) \to \mathrm{am}^*(S)$ можно рассматривать как канонический способ интеграции модульной основы модели в её модуляризированную аспектную структуру. Этот морфизм является регулярным c-DESC-эпиморфизмом (см. предложение 11 ниже), т. е. задаёт факторизацию модульной основы на аспекты. Поэтому отдельные аспекты можно извлекать из модели средствами модульного проектирования, трассируя классы задач вдоль этого морфизма (здесь применяется универсальная теоретико-категорная конструкция декартова квадрата, которая служит обобщением вычисления полного прообраза в теории множеств [30]). Семейство $\mathrm{am}_*(\chi_S)$, $S \in \mathrm{Ob}\,\text{m-AO}$, образует естественное преобразование функтора am_*, выделяющего модульную основу из модуляризируемых АО-моделей, в функтор выделения их аспектной структуры am^*.

Определение 12. (am-)*модуляризацией* (аспектной структуры) произвольного m-AO-объекта S называется c-DESC-морфизм $\mathrm{am}_*(\chi_S)$, где χ — единица сопряжения $\mathrm{am}^* \dashv \mathrm{am}$.

Естественным (хотя и не единственным) способом модуляризации аспектной структуры АО-модели является восстановление трассируемой трансформации, породившей её разметку. Действительно, если для АО-модели $\langle A, l\rangle$ существует подходящая трассируемая трансформация $s\colon X \to A$ некоторого c-DESC-объекта X в A, удовлетворяющая условию $\mathrm{sig}(s^{\mathrm{op}}) = l$, то X можно рассматривать как модульную единицу, состоящую из всех классов задач АО-модели, а s^{op} — как «каноническую» модуляризацию её аспектной структуры. Восстановление трансформации можно считать полноценным, если любое действие по интеграции модели в любую систему проецируется на модульный уровень в виде пары действий по интеграции модульных основ и аспектных структур, согласованных с восстановленными трансформациями. Часто удаётся аналогичным образом восстановить и трансформации АО-моделей: спроецировать их на модульный уровень в виде согласованной пары трансформаций модульных основ и аспектных структур.

В формальных технологиях над **Set** восстановление трансформации АО-модели $\langle A, l\rangle$ возможно в случае, когда разметка в определённой степени согласована со структурой объекта A. А именно, восстановление состоит в создании структуры на множестве $l(|A|)$, превращающей отображение l в трассу. Если такую структуру удаётся создать, то АО-модель обычно полностью разделяется на модуляризируемые аспекты: их совокупность имеет вид

$$\{\langle l^{-1}(x),\, l_x\colon y \mapsto x\rangle \mid x \in l(|A|)\}.$$

Например, трансформация помеченного сценария восстанавливается (очевидным образом) из такой и только такой разметки, которая разбивает частично упорядоченное множество A на хорошо упорядоченную совокупность прообразов точек. Очевидными примерами являются любой непустой дискретно размеченный сценарий, любой аспект, а также любой сценарий, в котором все элементы каждой компоненты связности порядка помечены одинаково. Поскольку технология SM поддерживает трассирование, восстановление всегда полноценно (см. предложение 9 ниже). Любая трансформация помеченных сценариев, допускающих восстановление трансформаций из своих разметок, сама восстанавливаема.

Мы будем называть экспликациями действия модульного уровня, порождающие разметки, интеграцию и трансформацию АО-моделей. Условие полноценности экспликации формализуется как подходящее требование универсальности.

Определение 13. *Экспликацией* (аспектной структуры) АО-модели $\langle A, l\rangle$ называется трассируемая трансформация s некоторого r-DESC-объекта в A, такая что $\mathrm{sig}(s^{\mathrm{op}}) = l$. *Экспликацией действия* АО-морфизма $f\colon S \to R$ (вдоль экспликаций s и r АО-моделей S и R соответственно) называется c-DESC-морфизм q, такой что $q \circ s^{\mathrm{op}} = r^{\mathrm{op}} \circ \mathrm{mod}(f)$. Экспликация s АО-модели S называется *универсальной*, если любой АО-морфизм с областью S эксплицируем вдоль s и любой экспликации своей кообласти. *Аспектным ядром* формальной технологии **AR** называется полная подкатегория c-AO в AO, состоящая из всех объектов, обладающих универсальной экспликацией. *Экспликацией трансформации* АО-моделей g называется трансформация некоторых r-DESC-объектов, обладающая трассой, эксплицирующей действие АО-морфизма g^{op}.

$$
\begin{tikzcd}
S \arrow[d, "f"'] \\
R
\end{tikzcd}
\quad\Longrightarrow\quad
\begin{tikzcd}
\mathrm{mod}(S) \arrow[r, "s^{\mathrm{op}}"] \arrow[d, "\mathrm{mod}(f)"'] & {} \arrow[d, dotted, "q"] \\
\mathrm{mod}(S) \arrow[r, "r^{\mathrm{op}}"] & {}
\end{tikzcd}
$$

Предложение 8. *Следующие утверждения эквивалентны для произвольной АО-модели $S = \langle A, l\colon \mathrm{sig}(A) \to L\rangle$ и её произвольной экспликации $s\colon P \to A$.*

(i) *Экспликация s универсальна.*

(ii) *Трасса s^{op} является регулярным* c-DESC-*эпиморфизмом.*

(iii) *Тождество коединицы $\varepsilon_P \circ \mathrm{sig}^*(l) = s^{\mathrm{op}} \circ \varepsilon_A$ задаёт кодекартов квадрат в* c-DESC.

Доказательство. Докажем импликацию (i) $\Longrightarrow$ (iii). Выберем произвольно пару c-DESC-морфизмов $y\colon \mathrm{sig}^*(L) \to X \leftarrow A :a$, такую что $y \circ \mathrm{sig}^*(l) = a \circ \varepsilon_A$. Применяя функтор sig к этому равенству, получаем, что $\mathrm{sig}(y) \circ l = \mathrm{sig}(a)$, так что имеется АО-морфизм $\langle a, \mathrm{sig}(y)\rangle\colon S \to \langle X, 1_{\mathrm{sig}(X)}\rangle$. Пусть $q\colon P \to X$ — его экспликация вдоль s и 1_X, так что $q \circ s^{\mathrm{op}} = a$. Тогда

$$(q \circ \varepsilon_P) \circ \mathrm{sig}^*(l) = q \circ s^{\mathrm{op}} \circ \varepsilon_A = a \circ \varepsilon_A = y \circ \mathrm{sig}^*(l),$$

так что и $q \circ \varepsilon_P = y$ ввиду наличия у морфизма l правого обратного. С учётом того, что ε_P — эпиморфизм (см. начало доказательства предложения 2), получаем, что q — стрелка копредела диаграммы $\mathrm{sig}^*(l)\colon \mathrm{sig}^*(L) \leftarrow \mathrm{sig}^*\big(\mathrm{sig}(A)\big) \to A :\varepsilon_A$, задаваемого тождеством коединицы.

$$
\begin{tikzcd}
\mathrm{sig}^*\big(\mathrm{sig}(A)\big) \arrow[r, "\varepsilon_A"] \arrow[d, "\mathrm{sig}^*(l)"'] & A \arrow[d, dashed, "s^{\mathrm{op}}"'] \arrow[ddr, "a"] & \\
\mathrm{sig}^*(L) \arrow[r, dashed, "\varepsilon_P"] \arrow[drr, "y"'] & P \arrow[dr, dotted, "q"] & \\
& & X
\end{tikzcd}
$$

Докажем импликацию (iii) $\Longrightarrow$ (ii). Ребро любого кодекартова квадрата, параллельное регулярному эпиморфизму, само является таковым (см. двойственное утверждение в [8, предложение 11.18 (2)]). А поскольку все ретракции являются регулярными эпиморфизмами [8, предложение 7.75 (1)], ребро копредела s^{op}, параллельное ретракции $\mathrm{sig}^*(l)$, представляет собой регулярный эпиморфизм.

Докажем импликацию (ii) $\Longrightarrow$ (i). Предположим, что s^{op} является коуравнителем пары c-DESC-морфизмов $u, v\colon Q \rightrightarrows A$. Выберем произвольно АО-модель R с экспликацией r и АО-морфизм $\langle h, b\rangle\colon S \to R$, положим $p = r^{\mathrm{op}} \circ h$, так что $\mathrm{sig}(p) = b \circ l$. Имеем

$$\mathrm{sig}(p \circ u) = b \circ l \circ \mathrm{sig}(u) = b \circ \mathrm{sig}(s^{\mathrm{op}} \circ u) = b \circ \mathrm{sig}(s^{\mathrm{op}} \circ v) = \mathrm{sig}(p \circ v),$$

откуда следует, что $p \circ u = p \circ v$ ввиду унивалентности функтора sig. По определению коуравнителя существует c-DESC-морфизм q, такой что $q \circ s^{\mathrm{op}} = p$. Согласно определению 13 этот морфизм является экспликацией действия морфизма $\langle h, b \rangle$. $\square$

Предложение 9. *Все* r-DESC*-объекты и универсальные экспликации всех объектов ядра образуют подкатегорию в* r-DESC*, содержащую все обратимые трансформации.*

Доказательство. Напомним, что все ретракции являются регулярными эпиморфизмами, т. е. любая обратимая трансформация s удовлетворяет условию (ii) предложения 8. В частности, тождественный морфизм 1_A служит универсальной экспликацией АО-модели $\langle A, 1_{\mathrm{sig}(A)} \rangle$. Кроме того, если $s\colon P \to A$ и $u\colon Q \to P$ — универсальные экспликации (АО-моделей $\langle A, \mathrm{sig}(s^{\mathrm{op}}) \rangle$ и $\langle P, \mathrm{sig}(u^{\mathrm{op}}) \rangle$ соответственно), то $v = s \circ u$ — универсальная экспликация (АО-модели $\langle A, \mathrm{sig}(v^{\mathrm{op}}) \rangle$) ввиду импликации (iii) $\Longrightarrow$ (i) предложения 8 и возможности строить композицию кодекартовых квадратов (см. двойственное утверждение в [8, предложение 11.10 (1)]): поскольку два квадрата, из которых составлена нижеприведённая диаграмма, кодекартовы, то таковым является и внешний прямоугольник.

$$\begin{tikzcd}
\mathrm{sig}^*\big(\mathrm{sig}(A)\big) \arrow[r, "\varepsilon_A"] \arrow[d, "\mathrm{sig}^*\left(\mathrm{sig}(s^{\mathrm{op}})\right)"'] & A \arrow[d, dashed, "s^{\mathrm{op}}"] \\
\mathrm{sig}^*\big(\mathrm{sig}(P)\big) \arrow[r, dashed, "\varepsilon_P"] \arrow[d, "\mathrm{sig}^*\left(\mathrm{sig}(u^{\mathrm{op}})\right)"'] & P \arrow[d, dashed, "u^{\mathrm{op}}"] \\
\mathrm{sig}^*\big(\mathrm{sig}(Q)\big) \arrow[r, dashed, "\varepsilon_Q"] & Q
\end{tikzcd}$$

$\square$

Теорема 2. *Существует АО-расширение*

$$\begin{aligned} \mathrm{ac}\colon &\text{c-DESC} \hookrightarrow \text{c-AO} \\ &: A \mapsto \langle A, 1_{\mathrm{sig}(A)} \rangle,\ g \mapsto \langle g, \mathrm{sig}(g) \rangle, \end{aligned}$$

причём $\mathrm{ac}^*(f)$ *эксплицирует любой* c-AO*-морфизм* f *и* ac*-модуляризация аспектной структуры любого* c-AO*-объекта представляет собой трассу его универсальной экспликации.*

Доказательство. Ввиду предложения 9 $\mathrm{ac}(A) \in \mathrm{Ob}\,\text{c-AO}$ для любого c-DESC-объекта A, так что пара отображений ac действительно является функтором. По теореме 1 правым сопряжённым к нему является функтор $\mathrm{mod} \circ \mathrm{cao}$, где $\mathrm{cao}\colon \text{c-AO} \hookrightarrow \text{AO}$ — полное вложение. Далее, пусть υ — отображение, ставящее в соответствие каждому c-AO-объекту какую-либо его универсальную экспликацию. Поскольку любая трасса является эпиморфизмом (предложение 3), экспликация q любого c-AO-морфизма $f\colon S \to R$ определяется равенством $q \circ \upsilon_S^{\mathrm{op}} = \upsilon_R^{\mathrm{op}} \circ \mathrm{mod}(f)$ однозначно. Отображение, переводящее f в q, является функцией морфизмов функтора ac^*, левого сопряжённого к ac, а семейство c-AO-морфизмов $\langle \upsilon_S^{\mathrm{op}}, 1_{\mathrm{str}(S)}\rangle \colon S \to \langle \operatorname{dom} \upsilon_S, 1_{\mathrm{str}(S)}\rangle$, $S \in \operatorname{Ob}$ c-AO, — единицей этого сопряжения. Проверим, что коединица этого сопряжения тождественна: для произвольных c-DESC-объекта P, c-AO-объекта S и c-DESC-морфизма $p\colon \mathrm{ac}^*(S) \to P$ найдём единственный c-AO-морфизм $r\colon S \to \langle P, 1_{\mathrm{sig}(P)}\rangle$, такой что $\mathrm{ac}^*(r) = p$. По определению $p \circ \upsilon_S^{\mathrm{op}} = 1_P \circ \mathrm{mod}(r)$, откуда следует, что $r = \langle p \circ \upsilon_S^{\mathrm{op}}, \mathrm{sig}(p)\rangle$, причём задать морфизм r иначе невозможно ввиду унивалентности функтора mod (теорема 1). $\square$

Следствие 2.1. *Универсальная экспликация АО-модели, если она существует, определяется однозначно с точностью до изоморфизма.*

Определение 14. *Ядерной АО-технологией* над произвольной формальной технологией проектирования $\mathbf{AR}$ называется модуляризируемая АО-технология над $\mathbf{AR}$, порождённая функтором ac.

Для формальных технологий, не все АО-модели над которыми обладают универсальной экспликацией, существуют более слабые подходы к модуляризации аспектной структуры. В частности, аспектную разметку произвольной АО-модели можно представить на модульном уровне частичным морфизмом модульной основы. Напомним, что частичным морфизмом объекта X в Y в категории C называется пара C-морфизмов с общим началом, один из которых является мономорфизмом и направлен в X (он выделяет «часть» объекта X, выступающую областью определения частичного морфизма), а другой произволен и направлен в Y (он задаёт действие частичного морфизма) [8, определение 28.1 (1)]. Если в категории C имеется достаточно декартовых квадратов, то определена композиция частичных морфизмов: композиция пары произвольных частичных морфизмов

$$m\colon X \leftarrow A \to Y : f, \quad m'\colon Y \leftarrow B \to Z : f'$$

представляет собой частичный морфизм

$$m \circ m''\colon X \leftarrow G \to Z : f' \circ g,$$

где $m''\colon A \leftarrow G \to B :g$ — рёбра декартова квадрата $f \circ m'' = g \circ m'$ (причём m'' является мономорфизмом ввиду стабильности мономорфизмов относительно декартовых квадратов [8, предложение 11.18]). Непосредственно проверяется, что совокупность всех C-объектов и всех их частичных морфизмов с таким законом композиции образует категорию (тождественный морфизм объекта T в ней имеет вид $1_T\colon T \leftarrow T \to T :1_T$, и вообще любой C-морфизм $p\colon T \to S$ порождает частичный морфизм $1_T\colon T \leftarrow T \to S :p$, так что C входит в эту категорию в качестве подкатегории).

Разметка произвольной АО-модели порождает частичный c-DESC-морфизм следующим образом. Воспользуемся тем, что коединица ε сопряжения $\mathrm{sig}^* \dashv \mathrm{sig}$ состоит из мономорфизмов (см. начало доказательства предложения 2). При помощи коединицы можно выделить в модульной основе АО-модели дискретную «часть» и далее подействовать на неё дискретной реализацией аспектной разметки.

Определение 15. *Частичной модуляризацией* произвольной АО-модели $\langle A, l\colon \mathrm{sig}(A) \to L\rangle$ называется следующий частичный c-DESC-морфизм, действующий из её модульной основы A в дискретную реализацию аспектной структуры L:

$$\varepsilon_A\colon A \leftarrow \mathrm{sig}^*\big(\mathrm{sig}(A)\big) \to \mathrm{sig}^*(L) :\!\mathrm{sig}^*(l).$$

Конечно, частичная модуляризация АО-модели не отражает трансформацию, порождающую модульную основу АО-модели из аспектной структуры, поскольку действие частичной модуляризации не обязано быть трассой трансформации. Тем не менее можно построить «аппроксимацию» модуляризации аспектной структуры, вычисляя копредел c-DESC-диаграммы, изображающей частичную модуляризацию (если он существует: ребро копредела частичного морфизма, параллельное его действию, можно рассматривать как универсальное расширение частичного морфизма на всю свою область). В литературе копределы диаграмм, представляющих частичные морфизмы некоторого специального вида, применяются при формализации технологий MDE для вычисления результатов процедур многошагового редактирования моделей [34]. По существу, речь идёт о соединении (в смысле раздела 1) модульной основы с дискретной реализацией аспектной структуры АО-модели. Согласно предложению 8 для любой АО-модели, содержащейся в аспектном ядре, именно так строится её универсальная экспликация. Здесь мы покажем, что таким способом можно вычислить модуляризацию аспектной структуры любой модуляризируемой АО-модели — имеет место слабая форма предложения 8.

Предложение 10. *Диаграмма частичной модуляризации любой модуляризируемой АО-модели имеет копредел, ребро которого, параллельное действию частичной модуляризации, задаёт модуляризацию аспектной структуры модели.*

Доказательство. Пусть $\mathrm{am}\colon \text{c-DESC} \to \text{m-AO}$ — произвольное АО-расширение, χ — единица сопряжения $\mathrm{am}^* \dashv \mathrm{am}$. Выберем произвольно модуляризируемую АО-модель $S = \langle A, l\colon \mathrm{sig}(A) \to L\rangle \in \mathrm{Ob}\,\text{m-AO}$. Напомним, что $\mathrm{str}(\chi_S) = 1_L$, так что $i \circ \mathrm{int}(\chi_S) = l$, где $i = \mathrm{asp}\Big(\mathrm{am}\big(\mathrm{am}^*(S)\big)\Big)$ — изоморфизм (по предложению 6). Положим $u = \varepsilon_{\mathrm{am}^*(S)} \circ \mathrm{sig}^*(i^{-1})$ и аналогично доказательству импликации (i) $\Longrightarrow$ (iii) предложения 8 проверим, что равенство $\mathrm{am}_*(\chi_S) \circ \varepsilon_A = u \circ \mathrm{sig}^*(l)$ (тождество коединицы для модуляризации аспектной структуры) задаёт кодекартов квадрат — копредел диаграммы частичной модуляризации АО-модели S. Действительно, произвольная пара c-DESC-морфизмов $a\colon A \to X \leftarrow \mathrm{sig}^*(L) :\!y$, удовлетворяющая условию $a \circ \varepsilon_A = y \circ \mathrm{sig}^*(l)$, порождает m-АО-морфизм $\langle a, j \circ \mathrm{sig}(y)\rangle\colon S \to \mathrm{am}(X)$, где $j = \mathrm{asp}\big(\mathrm{am}(X)\big)$. По определению сопряжения существует c-DESC-морфизм $q\colon \mathrm{am}^*(S) \to X$, такой что $\mathrm{am}(q) \circ \chi_S = \langle a, j \circ \mathrm{sig}(y)\rangle$. Применяя к этому равенству функтор am_*, получаем, что $q \circ \mathrm{am}_*(\chi_S) = a$. С учётом этого имеем

$$\mathrm{sig}(q \circ u) \circ l = \mathrm{sig}(q) \circ i^{-1} \circ l = \mathrm{sig}\big(q \circ \mathrm{am}_*(\chi_S)\big) = \mathrm{sig}(a \circ \varepsilon_A) = \mathrm{sig}(y) \circ l,$$

откуда следует, что и $q \circ u = y$ ввиду наличия у морфизма l правого обратного и унивалентности функтора sig. С учётом того, что $\varepsilon_{\mathrm{am}^*(S)}$ — эпиморфизм, получаем, что q — стрелка копредела диаграммы частичной модуляризации модуляризируемой АО-модели S.

$$\begin{tikzcd}
\mathrm{sig}^*\big(\mathrm{sig}(A)\big) \arrow[rr, "\varepsilon_A"] \arrow[d, "\mathrm{sig}^*(l)"'] & & A \arrow[d, dashed, "\mathrm{am}_*(\chi_S)"'] \arrow[dr, "a"] & \\
\mathrm{sig}^*(L) \arrow[r, dashed, "\mathrm{sig}^*(i^{-1})"] \arrow[drrr, "y"'] & \mathrm{sig}^*(L) \arrow[r, dashed, "\varepsilon_{\mathrm{am}^*(S)}"] & \mathrm{am}^*(S) \arrow[dr, dotted, "q"'] & \\
 & & & X
\end{tikzcd}$$

□

Предложение 11. *Модуляризация аспектной структуры любой модуляризируемой АО-модели является регулярным эпиморфизмом.*

Доказательство. Утверждение вытекает из предложения 10. Доказательство аналогично доказательству импликации (iii) $\Longrightarrow$ (ii) предложения 8. □

4. Проектирование технологий проектирования

Теоретико-категорный аппарат, развитый в предыдущих разделах, позволяет построить формальные технологии проектирования технологий, образующие теоретический фундамент подхода MDE. Будем обозначать через $\mathbf{CONF}$, $\mathbf{SPEC}$, $\mathbf{ARCH}$ категории, состоящие из всех формальных технологий конфигурирования, специфицирования и проектирования соответственно и всех их морфизмов. Имеются очевидные «забывающие» функторы $\mathbf{desc}\colon \mathbf{CONF} \to \mathbf{CAT}$, $\mathbf{conf}\colon \mathbf{SPEC} \to \mathbf{CONF}$, $\mathbf{spec}\colon \mathbf{ARCH} \to \mathbf{SPEC}$.

Функтор $\mathbf{desc}\colon \langle \text{c-DESC}, \mathrm{Conf}\rangle \mapsto \text{c-DESC}$ унивалентен и обладает левым сопряжённым $\mathbf{desc}^*\colon \mathbf{CAT} \to \mathbf{CONF}\colon \text{c-DESC} \mapsto \langle \text{c-DESC}, \varnothing\rangle$ с тождественной единицей сопряжения, поэтому он способен служить для выделения интерфейсов из формальных технологий конфигурирования. Возможности сборки сложных технологий конфигурирования определяются классом всех $\mathbf{CONF}$-диаграмм, переходящих под действием функтора $\mathbf{D}(\mathbf{desc})$ в предконфигурации, мы обозначим его через DSConf. Имеется следующая характеризация класса всех $\mathbf{desc}$-предконфигураций (ср. [4, предложение 15]).

Предложение 12. *Диаграмма $\Sigma\colon X \to \mathbf{CAT}$, имеющая копредел, является $\mathbf{desc}$-предконфигурацией тогда и только тогда, когда любое ребро её копредела σ_I, $I \in \mathrm{Ob}\, X$, сохраняет копределы всех $\Sigma(I)$-диаграмм, сохраняемые всеми функторами $\Sigma(f)$, где f — X-морфизм с областью I.*

Доказательство. Рассмотрим диаграмму $\Sigma\colon X \to \mathbf{CAT}$ с копределом, вершину которого мы обозначим через S, а рёбра через $\sigma_I\colon \Sigma(I) \to S$, $I \in \mathrm{Ob}\, X$.

Если диаграмма Σ удовлетворяет условию предложения, то любая $\mathbf{CONF}$-диаграмма $\Xi \in \mathbf{Ddesc}^{-1}(\{\Sigma\})$ имеет копредел с вершиной $\left\langle S, \bigcup_{I \in \mathrm{Ob}\, X} \sigma_I \circ \mathrm{Conf}_I \right\rangle$, где Conf_I — класс конфигураций технологии $\Xi(I)$, и рёбрами σ_I. Допустим теперь, что имеются X-объект I и $\Sigma(I)$-диаграмма Δ, такие что все функторы $\Sigma(f)$, где $f \in \mathrm{Mor}\, X$ и $\mathrm{dom}\, f = I$, сохраняют копределы диаграммы Δ, но σ_I не сохраняет их. Тогда $\mathbf{CONF}$-диаграмма

$$\begin{aligned}\Theta\colon & X \to \mathbf{CONF}\\ & : J \mapsto \langle \Sigma(J),\ \{\Sigma(f) \circ \Delta \mid f\colon I \to J \in \mathrm{Mor}\, X\}\rangle,\ h \mapsto \Sigma(h)\end{aligned}$$

не может иметь копредела с вершиной, которую функтор $\mathbf{desc}$ переводил бы в S, и рёбрами σ_I. Таким образом, функтор $\mathbf{desc}$ не сохраняет копределы диаграммы Θ, несмотря на то что $\mathbf{desc} \circ \Theta = \Sigma$. $\square$

Предконфигурациями являются, в частности, все дискретные $\mathbf{CAT}$-диаграммы. Приведём пример $\mathbf{CAT}$-диаграммы, обладающей копределом, но не удовлетворяющей критерию предложения 12. Пусть $\boldsymbol{U}$ — категория вида $1 \to 0 \leftarrow 1'$, Σ — $\mathbf{CAT}$-диаграмма вложений $\boldsymbol{U} \hookleftarrow \boldsymbol{U} \setminus \{0\} \hookrightarrow \boldsymbol{U}$. Двухточечная дискретная диаграмма $\ulcorner 1 \urcorner \amalg \ulcorner 1' \urcorner$ имеет копредел в $\boldsymbol{U}$, однако не имеет его в категории $\mathrm{colim}(\Sigma)$.

Что касается трансформаций формальных технологий, то они должны отражать процессы реализации программных систем, т. е. переходы от технологий конфигурирования интерфейсов к технологиям конфигурирования их реализаций. Произвольный функтор $\mathrm{sig}\colon \mathrm{c\text{-}DESC} \to \mathrm{SIG}$ можно интерпретировать как правило выделения интерфейсов моделей технологии $\langle \mathrm{c\text{-}DESC}, \mathrm{Conf}\rangle$, если он удовлетворяет условиям (ii)—(v) определения 1. Его можно рассматривать как $\mathbf{CONF}$-морфизм $\mathrm{sig}\colon \langle \mathrm{c\text{-}DESC}, \mathrm{Conf}\rangle \to \langle \mathrm{SIG}, \mathrm{sig}\circ\mathrm{Conf}\rangle$, удовлетворяющий этим условиям. Обозначим через r-CONF класс всех таких $\mathbf{CONF}$-морфизмов. *Формальной технологией проектирования технологий конфигурирования* называется четвёрка

$$\mathbf{SCONF} = \langle \mathbf{CONF},\ \mathrm{DSConf},\ \mathbf{desc},\ (\mathrm{Ob}\,\mathbf{CONF}, \mathrm{r\text{-}CONF})^{\mathrm{op}}\rangle.$$

На базе технологии $\mathbf{SCONF}$ строится технология проектирования технологий специфицирования. Примечательно, что для этого требуются формальные конструкции аспектно-ориентированного подхода: как показано ниже, функтор выделения интерфейсов, дополняющий произвольную формальную технологию конфигурирования до технологии специфицирования, представляет собой в точности её эксплицируемую разметку в технологии $\mathbf{SCONF}$. Следовательно, *формальной технологией проектирования технологий специфицирования* правомерно называть ядерную АО-технологию над $\mathbf{SCONF}$ (см. следствие 3.1 ниже), которую мы будем обозначать через $\mathbf{SSPEC}$. Получается, что категория интерфейсов отражает аспектную структуру своих реализаций (например ассортимент ролей, которые компоненты играют в составе систем [25]). Таким образом, можно снижать затраты на сборку сложных технологий специфицирования, пользуясь аспектно-ориентированными приёмами. Более того, мы докажем, что конструкция АО-технологии, в свою очередь, совместима с аспектной структурой технологий: если некоторая модульная технология проектирования аспектно полна (в том смысле, что сборка систем в ней не разрушает аспектную структуру), то переход к АО-технологии над ней представляет собой трансформацию. В связи с этим введём следующее понятие.

Определение 16. Формальная технология проектирования $\mathbf{AR}$ называется *аспектно полной*, если класс Conf аспектно замкнут относительно функтора mod.

Предложение 13. *Формальная технология* $\mathbf{AR}$ *аспектно полна тогда и только тогда, когда класс всех конфигураций любой АО-технологии над* $\mathbf{AR}$ *совпадает с* $\mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf})$.

Доказательство. Достаточность непосредственно вытекает из определения 8. Для доказательства необходимости рассмотрим произвольный функтор ai, порождающий АО-технологию над $\mathbf{AR}$. Согласно следствию 1.2 и определению 7 имеем

$$\mathbf{D}\mathrm{int}^{-1}(\mathrm{AOInt}_{\mathrm{int}}) \subseteq \mathbf{D}\mathrm{ai}^{-1}(\mathrm{AOInt}_{\mathrm{ai}}) \subseteq \mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf}).$$

Поскольку $\mathbf{D}\mathrm{int}^{-1}(\mathrm{sig} \circ \mathrm{Conf}) = \mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf})$, по предположению $\mathrm{sig} \circ \mathrm{Conf} \subseteq \subseteq \mathrm{AOInt}_{\mathrm{int}}$, так что

$$\mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf}) \subseteq \mathbf{D}\mathrm{ai}^{-1}(\mathrm{AOInt}_{\mathrm{ai}}) \subseteq \mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf}). \qquad \square$$

Например, технология событийного моделирования SM аспектно полна (см. следствие 4.6), а технология проектирования технологий конфигурирования $\mathbf{SCONF}$ — нет (см. ниже).

Теорема 3. *АО-модель над* $\mathbf{SCONF}$ *обладает экспликацией тогда и только тогда, когда она является технологией специфицирования; любая её экспликация универсальна. Трансформация АО-моделей над* $\mathbf{SCONF}$ *обладает экспликацией тогда и только тогда, когда её область и кообласть являются технологиями специфицирования.*

Доказательство. Зафиксируем произвольную формальную технологию специфицирования $\mathrm{SC} = \langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig}\colon \text{c-DESC} \to \mathrm{SIG}\rangle$, положим $\mathrm{SConf} = = \mathrm{sig} \circ \mathrm{Conf}$. Морфизм формальных технологий конфигурирования

$$\mathrm{sig}\colon \langle \text{c-DESC}, \mathrm{Conf}\rangle \to \langle \mathrm{SIG}, \mathrm{SConf}\rangle \in \text{r-CONF}$$

является трассой единственной экспликации технологии SC. Эта трансформация обратима, поскольку функтор sig^*, левый сопряжённый и правый обратный к sig, сохраняет все копределы [6, разд. 5.5] и потому индуцирует $\mathbf{CONF}$-морфизм $\mathrm{sig}^*\colon \langle \mathrm{SIG}, \mathrm{SConf}\rangle \to \langle \text{c-DESC}, \mathrm{Conf}\rangle$ (таким образом, технология $\mathbf{SCONF}$ поддерживает трассирование). Ввиду предложения 9 экспликация технологии SC универсальна.

Пусть $\langle \mathrm{cm}, \mathrm{sm}\rangle \colon \mathrm{SC} \to \langle \text{c-DESC}', \mathrm{Conf}', \mathrm{sig}' \colon \text{c-DESC}' \to \mathrm{SIG}'\rangle$ — $\mathbf{SPEC}$-морфизм, такой что $\mathrm{cm} \in \text{r-CONF}$, в частности, $\mathrm{sm} \circ \mathrm{sig} = \mathrm{sig}' \circ \mathrm{cm}$ и $\mathrm{Conf}' = \mathrm{cm} \circ \mathrm{Conf}$. Следовательно, имеется $\mathbf{CONF}$-морфизм

$$\mathrm{sm} \colon \langle \mathrm{SIG}, \mathrm{SConf}\rangle \to \langle \mathrm{SIG}', \mathrm{sig}' \circ \mathrm{Conf}'\rangle.$$

Требуется доказать, что $\mathrm{sm} \in \text{r-CONF}$. Рассмотрим функтор $\mathrm{scm} = \mathrm{sig}' \circ \mathrm{cm} \in \in \text{r-CONF}$. Функтор $\mathrm{scm}^* = \mathrm{cm}^* \circ \mathrm{sig}'^*$ является левым сопряжённым к нему (мы будем обозначать через ν коединицу этого сопряжения). Из равенства $\mathrm{sm} \circ \mathrm{sig} = \mathrm{scm}$ вытекает, что $\mathrm{sm} \circ \mathrm{SConf} = \mathrm{scm} \circ \mathrm{Conf} = \mathrm{sig}' \circ \mathrm{Conf}'$.

$$\begin{tikzcd}
\text{c-DESC} \arrow[r, "\mathrm{sig}"] \arrow[d, "\mathrm{cm}"'] \arrow[dr, "\mathrm{scm}"] & \mathrm{SIG} \arrow[d, "\mathrm{sm}"] \\
\text{c-DESC}' \arrow[r, "\mathrm{sig}'"] & \mathrm{SIG}'
\end{tikzcd}$$

Из этого же равенства следует, что $\mathrm{sm} = \mathrm{scm} \circ \mathrm{sig}^*$, так что функтор sm унивалентен (будучи композицией унивалентных функторов) и функтор $\mathrm{sig} \circ \mathrm{scm}^*$ является левым сопряжённым к нему с тождественной единицей. Действительно,

$$\mathrm{sm} \circ \mathrm{sig} \circ \mathrm{scm}^* = \mathrm{scm} \circ \mathrm{scm}^* = 1_{\mathrm{SIG}'}$$

и для произвольных $A \in \mathrm{Ob}\,\mathrm{SIG}'$, $X \in \mathrm{Ob}\,\mathrm{SIG}$, $f \colon A \to \mathrm{sm}(X)$ существует, причём единственный в силу унивалентности функтора sm, SIG-морфизм $g \colon \mathrm{sig}\big(\mathrm{scm}^*(A)\big) \to X$, удовлетворяющий условию $\mathrm{sm}(g) = f$. Положим $g = \mathrm{sig}\big(\nu_{\mathrm{sig}^*(X)} \circ \mathrm{scm}^*(f)\big)$, тогда

$$\mathrm{sm}(g) = \mathrm{scm}\big(\nu_{\mathrm{sig}^*(X)} \circ \mathrm{scm}^*(f)\big) = 1_{\mathrm{scm}(\mathrm{sig}^*(X))} \circ \mathrm{scm}\big(\mathrm{scm}^*(f)\big) = f.$$

Проверим, что функтор sm поднимает копределы диаграммы $\mathrm{sig} \circ \Delta$ для любой диаграммы $\Delta \in \mathrm{Conf}$. Если ξ — копредел диаграммы $\mathrm{sm} \circ (\mathrm{sig} \circ \Delta)$, то существует копредел δ диаграммы Δ, такой что $\mathrm{sm} \circ \mathrm{sig} \circ \delta = \xi$, поскольку функтор scm по построению поднимает копределы всех диаграмм из класса Conf. Тогда в силу предложения 2 коконус $\mathrm{sig} \circ \delta$ является копределом диаграммы $\mathrm{sig} \circ \Delta$, причём он переходит в ξ под действием функтора $\mathbf{D}\mathrm{sm}$.

Наконец, пусть произвольные диаграммы $\Delta \in \mathrm{Conf}$ и $\Xi \in \mathrm{Ob}\,\mathbf{D}\mathrm{SIG}$ таковы, что $\mathrm{sm} \circ (\mathrm{sig} \circ \Delta) = \mathrm{sm} \circ \Xi$. Тогда $\mathrm{scm} \circ \Delta = \mathrm{scm} \circ \mathrm{sig}^* \circ \Xi$, откуда получаем, что $\mathrm{sig}^* \circ \Xi \in \mathrm{Conf}$, следовательно, $\Xi = \mathrm{sig} \circ \mathrm{sig}^* \circ \Xi \in \mathrm{SConf}$. □

Следствие 3.1. *Ядерная АО-технология над* $\mathbf{SCONF}$ *имеет категорию моделей* $\mathbf{SPEC}$ *и функтор выделения интерфейсов* $\mathbf{conf}$.

Следствие 3.2. *Пусть* $\mathbf{AR} = \langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig}, \text{r-DESC}\rangle$ — *произвольная формальная технология проектирования. Для любого функтора* ai, *порождающего АО-технологию над* $\mathbf{AR}$, *функтор* $\mathrm{mod} \colon \mathrm{AO} \to \text{c-DESC}$, *построенный из её составляющих, индуцирует* $\mathbf{ARCH}$-*морфизм, действующий из* $\mathrm{AO}_{\mathrm{ai}}(\mathbf{AR})$ *в* $\mathbf{AR}$. *Технология* $\mathbf{AR}$ *аспектно полна тогда и только тогда, когда функтор* $\mathbf{spec}$ *переводит этот морфизм в трассу эксплицируемой трансформации технологий специфицирования.*

Доказательство. Рассмотрим тройку $\mathrm{ao_{ai}} = \langle \mathrm{mod}, \mathrm{int} \circ \mathrm{ai}^*, \mathrm{mod}(-^{\mathrm{op}})^{\mathrm{op}} \rangle$. Ввиду условия (ii) следствия 1.1 и определения 7 эта тройка представляет собой $\mathbf{ARCH}$-морфизм, действующий из $\mathrm{AO_{ai}}(\mathbf{AR})$ в $\mathbf{AR}$.

$$\begin{tikzcd}
\langle \mathrm{AO}, \arrow[d, "\mathrm{mod}"'] & \mathbf{D}\mathrm{ai}^{-1}(\mathrm{AOInt_{ai}}) \arrow[d, phantom, "\mathbf{D}\mathrm{mod}\,|\cap"] & \mathrm{AO} \arrow[r, "\mathrm{ai}"] \arrow[d, "\mathrm{mod}"'] & \mathrm{INT}, \arrow[d, "\mathrm{int}\circ\mathrm{ai}^*"] & \text{tr-AO}^{\mathrm{op}}\rangle \arrow[d, "\mathrm{mod}(-^{\mathrm{op}})^{\mathrm{op}}"] \\
\langle \text{c-DESC}, & \mathrm{Conf}, & \text{c-DESC} \arrow[r, "\mathrm{sig}"] & \mathrm{SIG}, & \text{r-DESC}\rangle
\end{tikzcd}$$

Если технология $\mathbf{AR}$ аспектно полна, то в силу теоремы 1 и предложения 13 $\mathbf{conf}\big(\mathbf{spec}(\mathrm{ao_{ai}})\big) \in \text{r-CONF}$, так что $\mathbf{SPEC}$-морфизм $\mathbf{spec}(\mathrm{ao_{ai}})$ является трассой трансформации в технологии $\mathbf{SSPEC}$, причём эта трансформация эксплицируема по теореме 3. В противном случае $\mathrm{mod} \circ \mathbf{D}\mathrm{mod}^{-1}(\mathrm{AOInt_{mod}}) \neq \mathrm{Conf}$, поэтому $\mathbf{SPEC}$-морфизм $\mathbf{spec}(\mathrm{ao_{mod}})$ не является трассой трансформации. □

Любопытно, что формальная технология $\mathbf{SCONF}$ сама аспектно неполна. Чтобы убедиться в этом, построим $\mathbf{SPEC}$-диаграмму Ξ, обладающую копределом в категории запятой $\mathbf{desc} \downarrow \mathbf{CAT}$, вершина которого не принадлежит категории АО-моделей над $\mathbf{SCONF}$, несмотря на то что $\mathbf{CONF}$-диаграмма $\mathbf{conf} \circ \Xi$ является конфигурацией в $\mathbf{SCONF}$. Рассмотрим следующие категории и функторы:

$$\begin{aligned}
&X = (\{A, B\},\ \{1_A,\ r\colon A \to A,\ 1_B\colon B \to B,\ p_1, p_2, q_1, q_2\colon A \to B\}),\\
&\quad \text{где } r \circ r = 1_A,\ \ p_i \circ r = q_i \quad (i = 1, 2),\\
&X_i = (\{A, B\},\ \{1_A, r, 1_B, p_i, q_i\}) \subseteq X \quad (i = 1, 2),\\
&X_0 = (\{A, B\},\ \{1_A, r, 1_B\}) \subseteq X,\\
&Y = (\{A\},\ \{1_A, r\}) \subseteq X_0,\\
&\mathrm{ex}_i\colon X_i \hookrightarrow X \quad (i = 0, 1, 2),\\
&\mathrm{jx}_i\colon X_0 \hookrightarrow X_i \quad (i = 1, 2),\\
&\mathrm{yx}\colon Y \hookrightarrow X_0,\\
&\mathrm{xy}\colon X \to Y\colon A \mapsto A,\ B \mapsto A,\ r \mapsto r,\ p_i \mapsto r,\ q_i \mapsto 1_A \quad (i = 1, 2),\\
&\mathrm{xy}_i = \mathrm{xy} \circ \mathrm{ex}_i \quad (i = 0, 1, 2).
\end{aligned}$$

Любой из функторов xy_i $(i = 1, 2)$ унивалентен, и функтор $\mathrm{jx}_i \circ \mathrm{yx}$ сопряжён слева к нему с тождественной единицей. Обозначим через Ξ следующую $\mathbf{SPEC}$-диаграмму, состоящую из вложений формальных технологий.

$$\langle \mathrm{jx}_1, \mathrm{xy}_0 \rangle\colon \langle X_1, \varnothing, \mathrm{xy}_1 \rangle \hookleftarrow \mathbf{conf}^*\big(\mathbf{desc}^*(X_0)\big) \hookrightarrow \langle X_2, \varnothing, \mathrm{xy}_2 \rangle :\!\langle \mathrm{jx}_2, \mathrm{xy}_0 \rangle,$$

$$\begin{tikzcd}
\overset{r}{\curvearrowright}A \arrow[r, shift left, "p_1"] \arrow[r, shift right, "q_1"'] \arrow[d] & B \arrow[dl] & \hookleftarrow & \overset{r}{\curvearrowright}A \arrow[d] & B \arrow[d] & \hookrightarrow & \overset{r}{\curvearrowright}A \arrow[r, shift left, "p_2"] \arrow[r, shift right, "q_2"'] \arrow[d] & B \arrow[dl] \\
\overset{r}{\curvearrowright}A & & & \overset{r}{\curvearrowright}A & B & & \overset{r}{\curvearrowright}A &
\end{tikzcd}$$

Пусть $\mathbf{dc}\colon \mathbf{SPEC} \hookrightarrow \mathbf{desc} \downarrow \mathbf{CAT}$ — вложение категорий. Диаграмма $\mathbf{dc} \circ \Xi$ обладает копределом в категории $\mathbf{desc} \downarrow \mathbf{CAT}$ с рёбрами

$$\langle \mathrm{ex}_1, 1_Y \rangle \colon \langle X_1, \varnothing, \mathrm{xy}_1 \rangle \hookrightarrow \langle X, \varnothing, \mathrm{xy} \rangle \hookleftarrow \langle X_2, \varnothing, \mathrm{xy}_2 \rangle : \langle \mathrm{ex}_2, 1_Y \rangle.$$

Функтор xy не унивалентен, поэтому $\mathrm{colim}(\mathbf{dc} \circ \Xi)$ не является АО-моделью над $\mathbf{SCONF}$, так что диаграмма Ξ не является аспектно детерминированной. В то же время $\mathbf{CONF}$-диаграмма $\mathbf{conf} \circ \Xi$ является конфигурацией в $\mathbf{SCONF}$. Действительно, рассмотрим произвольную диаграмму $\Delta\colon Z \to X_1$. Если её схема пуста, то она не имеет копредела, поскольку ни A, ни B не являются инициальными X_1-объектами. Если категория Z непуста и связна, то диаграмма Δ имеет копредел тогда и только тогда, когда для всякого $I \in \Delta^{-1}(A)$ каждое из следующих множеств состоит не более чем из одного элемента:

(i) множество $\Delta\big(\mathrm{Mor}(I, J)\big)$ для всякого $J \in \Delta^{-1}(A)$,

(ii) множество $\displaystyle\bigcup_{J \in \Delta^{-1}(B)} \Delta\big(\mathrm{Mor}(I, J)\big)$

(при этом если $\Delta(\mathrm{Ob}\, Z) = \{A\}$, то $\mathrm{colim}(\Delta) = A$, а в противном случае $\mathrm{colim}(\Delta) = B$). Если же категория Z дискретна и содержит более одного объекта, то диаграмма Δ имеет копредел тогда и только тогда, когда $\Delta(\mathrm{Ob}\, Z) = \{B\}$ (и тогда $\mathrm{colim}(\Delta) = B$). В любом из описанных случаев, если диаграмма Δ обладает копределом, то функтор ex_1 сохраняет его. Аналогично проверяется, что функтор ex_2 сохраняет копределы всех X_2-диаграмм. Из предложения 12 получаем, что диаграмма $\mathbf{desc} \circ \mathbf{conf} \circ \Xi$ является $\mathbf{desc}$-предконфигурацией, что и требовалось.

На основе полученных выше результатов строится формальная технология проектирования технологий проектирования. Для технологий проектирования технологии специфицирования не могут служить интерфейсами, поскольку функтор $\mathbf{spec}\colon \mathbf{ARCH} \to \mathbf{SPEC}$, «забывающий» категорию трансформаций, не унивалентен. Тем не менее он обладает левым сопряжённым

$$\begin{aligned} \mathbf{spec}^*\colon &\mathbf{SPEC} \to \mathbf{ARCH} \\ &: \langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig} \rangle \mapsto \langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig}, \mathrm{Iso}\, \text{c-DESC} \rangle, \end{aligned}$$

причём единица этого сопряжения тождественна, и $\mathbf{spec}^* \circ \mathbf{conf}^* \circ \mathbf{desc}^* = \mathbf{triv}$. Целесообразно извлекать интерфейсы из технологий проектирования таким же способом, каким функтор $\mathbf{conf}$ извлекает интерфейсы из технологий специфицирования — путём «забывания» функтора выделения интерфейсов (sic!). Поэтому интерфейсом технологии проектирования является любая тройка $\langle \text{c-DESC}, \mathrm{Conf}, \text{r-DESC} \rangle$, удовлетворяющая условиям (i), (vi)—(viii) определения 1. В [31] такие тройки названы *формализмами проектирования*. Морфизмом формализма $\langle \text{c-DESC}_1, \mathrm{Conf}_1, \text{r-DESC}_1 \rangle$ в формализм $\langle \text{c-DESC}_2, \mathrm{Conf}_2, \text{r-DESC}_2 \rangle$ служит любая пара функторов

$$\langle \mathrm{cm}\colon \text{c-DESC}_1 \to \text{c-DESC}_2,\ \mathrm{rm}\colon \text{r-DESC}_1 \to \text{r-DESC}_2 \rangle,$$

удовлетворяющая условиям (i), (ii), (iv) определения 3. Обозначим категорию всех формализмов и всех их морфизмов через $\mathbf{FORM}$. Имеются забывающие функторы

$$\begin{aligned}
\mathbf{form} &: \mathbf{ARCH} \to \mathbf{FORM} \\
&\quad : \langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig}, \text{r-DESC} \rangle \mapsto \langle \text{c-DESC}, \mathrm{Conf}, \text{r-DESC} \rangle, \\
\mathbf{fconf} &: \mathbf{FORM} \to \mathbf{CONF} \\
&\quad : \langle \text{c-DESC}, \mathrm{Conf}, \text{r-DESC} \rangle \mapsto \langle \text{c-DESC}, \mathrm{Conf} \rangle,
\end{aligned}$$

удовлетворяющие соотношению $\mathbf{fconf} \circ \mathbf{form} = \mathbf{conf} \circ \mathbf{spec}$. Функтор **form** унивалентен и обладает левым сопряжённым

$$\begin{aligned}
\mathbf{form}^* &: \mathbf{FORM} \to \mathbf{ARCH} \\
&: \langle \text{c-DESC}, \mathrm{Conf}, \text{r-DESC} \rangle \mapsto \langle \text{c-DESC}, \mathrm{Conf}, 1_{\text{c-DESC}}, \text{r-DESC} \rangle,
\end{aligned}$$

единица этого сопряжения тождественна (ср. конструирование функтора mod^* в доказательстве теоремы 1). Конфигурации технологий проектирования отбираются из прообразов **form**-предконфигураций по критерию совместимости с конфигурациями технологий специфицирования: введём класс **CONF**-диаграмм

$$\begin{aligned}
\mathrm{PForm} = \{\Delta \mid\ & \mathbf{D}(\mathbf{fconf})^{-1}(\{\Delta\}) \subseteq \mathrm{FPConf}, \\
& \mathbf{D}(\mathbf{conf})^{-1}(\{\Delta\}) \subseteq \mathrm{CSConf}, \\
& \mathbf{D}(\mathbf{conf} \circ \mathbf{spec})^{-1}(\{\Delta\}) \subseteq \mathrm{PCspec}\},
\end{aligned}$$

где FPConf — класс всех **form**-предконфигураций, CSConf — класс всех конфигураций в технологии **SSPEC**, PCspec — класс всех **ARCH**-диаграмм, копределы которых сохраняет функтор **spec**. Заметим, что класс PForm содержит все дискретные **CONF**-диаграммы. Трансформации технологий проектирования расширяют трансформации технологий специфицирования так, чтобы обеспечить трассируемость, для этого они должны быть согласованы с дискретной реализацией интерфейсов.

Определение 17. $\mathbf{ARCH}^{\mathrm{op}}$-морфизм $\langle \mathrm{cm}, \mathrm{sm}, \mathrm{rm} \rangle^{\mathrm{op}} \colon \mathbf{AR} \to \mathbf{AR}'$ называется *трансформацией технологий проектирования*, если он удовлетворяет следующим условиям.

(i) Пара $\langle \mathrm{cm}, \mathrm{sm} \rangle$ представляет собой трассу трансформации технологии специфицирования $\mathbf{spec}(\mathbf{AR})$ в $\mathbf{spec}(\mathbf{AR}')$.

(ii) Существует функтор drm, правый обратный к rm и удовлетворяющий условию $\mathrm{drm}(i) = \mathrm{cm}^*(i)$ для любого $i \in \mathrm{Iso}\,\mathrm{codom}\,\mathrm{cm}$.

Например, двойственным к трансформации является **ARCH**-морфизм $\mathrm{ao}_{\mathrm{ai}} \colon \mathrm{AO}_{\mathrm{ai}}(\mathbf{AR}) \to \mathbf{AR}$, построенный в доказательстве следствия 3.2, при условии, что технология **AR** аспектно полна и все трансформации в ней трассируемы (этому условию удовлетворяет, в частности, технология событийного моделирования SM). Более того, при этом условии трансформация $\mathrm{ao}_{\mathrm{int}}^{\mathrm{op}}$ обратима, причём **ARCH**-морфизм, правый обратный к её трассе, индуцируется АО-расширением: он имеет вид $\langle \mathrm{am}(-), 1_{\mathrm{SIG}}, \mathrm{am}(-^{\mathrm{op}})^{\mathrm{op}} \rangle \colon \mathbf{AR} \to \mathrm{AO}_{\mathrm{int}}(\mathbf{AR})$, где am — произвольное АО-расширение технологии $\mathbf{AR}$. Среди других трансформаций вида $\mathrm{ao}_{\mathrm{ai}}^{\mathrm{op}}$, вообще говоря, встречаются и необратимые.

Обозначим через r-ARCH класс всех $\mathbf{ARCH}$-морфизмов, двойственных к трансформациям технологий проектирования. *Формальной технологией проектирования технологий проектирования* называется четвёрка

$$\mathbf{SARCH} = \langle \mathbf{ARCH}, \mathbf{D}(\mathbf{conf} \circ \mathbf{spec})^{-1}(\mathrm{PForm}), \mathbf{form}, (\mathrm{Ob}\,\mathbf{ARCH}, \text{r-ARCH})^{\mathrm{op}} \rangle.$$

Все трансформации технологий проектирования, как и других видов технологий, трассируемы: правый обратный к $\mathbf{form}$-образу произвольного r-ARCH-морфизма

$$\langle \mathrm{cm}, \mathrm{sm}, \mathrm{rm} \rangle \colon \mathbf{AR} \to \mathbf{AR}'$$

имеет вид

$$\langle \mathrm{cm}^*, \mathrm{drm} \rangle \colon \mathbf{form}(\mathbf{AR}') \to \mathbf{form}(\mathbf{AR}).$$

Кроме того, технология $\mathbf{SARCH}$, так же как $\mathbf{SCONF}$ и $\mathbf{SSPEC}$, обладает свойством кооднородности.

Ввиду следствия 3.1 имеется $\mathbf{ARCH}$-морфизм

$$\langle \mathbf{spec}, \mathbf{fconf}, \mathbf{spec}(-^{\mathrm{op}})^{\mathrm{op}} \rangle \colon \mathbf{SARCH} \to \mathbf{SSPEC}.$$

В свою очередь, следствие 3.2 (при выборе технологии $\mathbf{SCONF}$ в качестве $\mathbf{AR}$) в совокупности с предложением 7 приводит к существованию $\mathbf{ARCH}$-морфизма

$$\langle \mathbf{conf}, \mathbf{desc}, \mathbf{conf}(-^{\mathrm{op}})^{\mathrm{op}} \rangle \colon \mathbf{SSPEC} \to \mathbf{SCONF}.$$

Эту цепочку $\mathbf{ARCH}$-морфизмов можно продолжить ещё на один шаг, введя «формальную технологию проектирования категорий моделей»: положим

$$\mathbf{SCAT} = \big\langle \mathbf{CAT}, \mathbf{desc} \circ \mathrm{DSConf}, 1_{\mathbf{CAT}}, \big(\mathrm{Ob}\,\mathbf{CAT}, \mathbf{desc}(\text{r-CONF})\big)^{\mathrm{op}} \big\rangle.$$

Четвёрка $\mathbf{SCAT}$ является кооднородной формальной технологией проектирования, поддерживающей трассирование: класс всех трансформаций в ней является двойственным по отношению к классу всех унивалентных функторов, обладающих левым сопряжённым с тождественной единицей (т. е. $\mathbf{desc}$-разметок). Существует $\mathbf{ARCH}$-морфизм

$$\langle \mathbf{desc}, 1_{\mathbf{CAT}}, \mathbf{desc}(-^{\mathrm{op}})^{\mathrm{op}} \rangle \colon \mathbf{SCONF} \to \mathbf{SCAT}.$$

Технология $\mathbf{SCAT}$ устроена так, что этот $\mathbf{ARCH}$-морфизм представляет собой трассу трансформации технологий проектирования, индуцированной универсальной экспликацией технологии специфицирования $\mathbf{spec}(\mathbf{SCONF})$ (см. первый абзац доказательства теоремы 3). Таким образом, технологии конфигурирования получаются из категорий моделей путём трансформации в формальной технологии проектирования технологий проектирования.

5. Технологии, порождённые разметкой

Рассмотрим порядок выполнения основной технологической процедуры MDE — конструирования некоторой предметно-ориентированной технологии проектирования. Первым шагом естественным образом служит выбор языка для формальной записи моделей программ. Приёмы сборки сложных моделей, предусмотренные конструкциями языка (макроподстановка, разрешение ссылок, перекомпоновка и т. д.), образуют морфизмы формальных моделей, порождающие категорию c-DESC. Как только она построена, на следующем шаге можно выбрать класс конфигураций Conf и получить технологию конфигурирования. Однако более практично сразу перейти к выбору функтора выделения интерфейса sig, поскольку условия (ii), (iii) определения 1 диктуют его вид достаточно жёстко. Он должен совпадать с точностью до изоморфизма с некоторым унивалентым корефлектором в c-DESC (соответствующий пример был приведён в разделе 1 при рассмотрении технологии событийного моделирования SM). Класс конфигураций можно затем задать на уровне интерфейсов, отобрав их среди sig-предконфигураций. В результате получается предметно-ориентированная технология специфицирования. Из технологий такого рода можно синтезировать комплексные, в том числе поддерживающие модели нескольких видов одновременно, при помощи формализованных модульных и аспектно-ориентированных приёмов в рамках технологии проектирования технологий специфицирования (см. следствие 3.1).

Для перехода к технологии проектирования остаётся задать класс трансформаций Mor r-DESC. Как указывалось во введении, главным критерием рационального выбора трансформаций является обеспечение как можно более широких возможностей для трассирования. Трансформации должны быть трассируемыми (в частности, двойственными по отношению к действиям по интеграции), нетривиальными (выходить за рамки изоморфизмов) и в то же время не требующими чрезмерных затрат для реализации трассирования. Приемлемым компромиссом выглядит требование, чтобы трансформации позволяли трассировать включения компонентов в системы — действия по интеграции, не изменяющие внутреннюю структуру компонентов и благодаря этому трассируемые «бесплатно» [3]. Воспользуемся семантикой трассирования, изложенной в разделе 2. Если компонент X включается в систему S (т. е. имеется включение — c-DESC-морфизм $m\colon X \to S$), а S трансформируется в систему T (т. е. имеется трассируемая трансформация — r-DESC-морфизм $r\colon S \to T$), то X должен включаться в T, причём так, чтобы композиция (в категории c-DESC) трассы трансформации $r^{\mathrm{op}}\colon T \to S$ с включением X в T давала m.

Это условие легко выполнить на уровне интерфейсов (причём для произвольного морфизма m) ввиду определения 4: искомое включение $\mathrm{sig}(X)$ в $\mathrm{sig}(T)$ имеет вид $s\circ\mathrm{sig}(m)$, где s — любой морфизм, удовлетворяющий условию $\mathrm{sig}(r^{\mathrm{op}})\circ s = 1_{\mathrm{sig}(S)}$. Поэтому трансформация r, обеспечивающая трассирование включений, должна позволять «поднять» любой sig-морфизм из $\mathrm{sig}(X)$ в $\mathrm{sig}(T)$, композиция $\mathrm{sig}(r^{\mathrm{op}})$ с которым равна sig-образу заданного включения m, на уровень моделей: до морфизма, действующего из X в T. Как мы увидим далее (определение 18), в теории категорий это условие является обобщением определения понятия инициального морфизма.

В связи с этим целесообразно конструировать хорошо трассируемые трансформации начиная с уровня интерфейсов: прежде всего зафиксировать класс L всех допустимых разметок формальных моделей, т. е. sig-образов трасс. Согласно утверждению (ii) предложения 3 он должен состоять из ретракций и содержать все изоморфизмы. Трассы всех трансформаций находятся в классе $\mathrm{sig}^{-1}(\mathrm{L})$, поэтому можно считать его состоящим из действий по интеграции, наиболее сильно изменяющих структуру компонентов. Следовательно, включения, не разрушающие её, можно определить как действия по интеграции, «ортогональные» им. В теории категорий имеется формальное понятие ортогональности, которое даёт следующую характеризацию класса всех включений: любое действие по интеграции должно однозначно (с точностью до изоморфизма) разлагаться в композицию включения с морфизмом, переходящим в разметку под действием функтора sig. Классы морфизмов, порождающие разложения такого рода, называются компонентами факторизационной системы [8, 13]. Мы воспроизведём точное определение ниже, а здесь приведём классический пример: факторизационной системой является пара $(\mathrm{Epi}, \mathrm{Mono})$ в категории $\mathbf{Set}$, где мономорфизмы задают включения подмножеств, а любой эпиморфизм размечает свою область элементами кообласти. Ортогональность включений трассам означает, что на уровне интерфейсов любое действие по интеграции компонента в систему сводится к включению в неё результата некоторой разметки компонента. Иначе говоря, воздействие системного окружения на интерфейс компонента заключается в «огрублении» — отмене некоторой трансформации, определяемой однозначно с точностью до изоморфизма.

Чтобы неразрушающий характер включения распространялся на дискретную структуру моделей, дополнительно накладывается условие сохранения дискретности, согласно которому в sig-дискретную систему должны включаться только дискретные компоненты. Напомним, что c-DESC-объект S называется дискретным, если ε_S является изоморфизмом, где, как обычно, ε — коединица сопряжения $\mathrm{sig}^* \dashv \mathrm{sig}$. Например, в технологиях над $\mathbf{Set}$ обычно все мономорфизмы сохраняют дискретность в указанном смысле. Кроме того, в целях обеспечения трассируемости на уровне конфигураций, накладывается условие кооднородности. Трансформации отбираются из класса $\mathrm{sig}^{-1}(\mathrm{L})^{\mathrm{op}}$ по критерию способности трассировать включения компонентов. Таким образом класс разметок L порождает формальную технологию проектирования, которую мы будем кратко называть L-технологией.

Следуя [3], мы покажем, что технологии, порождённые классами разметок, очень хорошо поддаются аспектной ориентации. Все трансформации в них обратимы в смысле определения 4, а мономорфные включения являются неразрушающими. Процедура порождения таких технологий трассируется вдоль трансформаций технологий специфицирования определённого вида (является естественной относительно подходящего класса $\mathbf{SPEC}$-морфизмов), в частности «опускается» на уровень интерфейсов и «поднимается» на уровень АО-технологий. Поэтому категория tr-AO может быть построена как категория запятой определённого вида и (при некоторых дополнительных ограничениях) обеспечивается аспектная полнота. В качестве показательного примера технологии проектирования, порождённой классом разметок, мы рассмотрим технологию событийного моделирования SM.

Определение 18. Пусть $\mathrm{ff}\colon C \to D$ — произвольный унивалентный функтор, M — произвольный класс C-морфизмов. C-морфизм $f\colon T \to S$ называется M-*инициальным*, если для любых M-морфизма $m\colon X \to S$ и D-морфизма $k\colon \mathrm{ff}(X) \to \mathrm{ff}(T)$, таких что $\mathrm{ff}(f)\circ k = \mathrm{ff}(m)$, существует C-морфизм $k^+\colon X \to T$, такой что $\mathrm{ff}(k^+) = k$.

Будем обозначать через M-Init класс всех M-инициальных морфизмов. Это понятие построено как обобщение известного в теории категорий понятия инициального морфизма [8, определение 8.6], который является в точности Mor-инициальным морфизмом, т. е. M-инициальным для любого класса M. В ряде случаев в категории c-DESC имеется факторизационная система вида $(\mathrm{sig}^{-1}(\mathrm{Iso}), \mathrm{Mor\text{-}Init})$, например, когда функтор sig является топологическим [8, упражнение 21*M* (3)]. Этот факт можно обобщить для M-инициальных морфизмов следующим образом. Пусть M — произвольный класс морфизмов некоторой категории. Будем говорить, что выполняется отношение M-Difip(e, f) между морфизмами e и f, если для всякого коммутативного квадрата $f \circ u = m \circ e$, такого что $m \in \mathrm{M}$, существует единственный «диагональный» морфизм d, такой что $d \circ e = u$ и $f \circ d = m$.

$$\begin{tikzcd} {} \arrow[r, "u"] \arrow[d, "e"'] & {} \arrow[d, "f"] \\ {} \arrow[r, "m"'] \arrow[ur, dotted, "d"] & {} \end{tikzcd}$$

Заметим, что если e — эпиморфизм, то достаточно найти морфизм d, такой что $d \circ e = u$, чтобы установить M-Difip(e, f). Напомним, что отношение Difip (diagonal fill-in property), совпадающее с нашим Mor-Difip, применяется в теории категорий при определении факторизационной системы. А именно, факторизационной системой называется пара классов морфизмов (E, F), такая что

(i) $\mathrm{Iso} \circ \mathrm{E} \subseteq \mathrm{E}$;
(ii) $\mathrm{F} \circ \mathrm{Iso} \subseteq \mathrm{F}$;
(iii) $\mathrm{F} \circ \mathrm{E} = \mathrm{Mor}$;
(iv) $\mathrm{Difip}(\mathrm{E}, \mathrm{F})$.

Перечислим несколько свойств произвольной факторизационной системы, установленных в [8, гл. 14]. (E, F)-разложение любого морфизма (т. е. его представление в виде композиции $f \circ e$, где $e \in \mathrm{E}$ и $f \in \mathrm{F}$) определяется однозначно с точностью до изоморфизма. Класс E замкнут относительно композиции и содержит все изоморфизмы. Он слабо сократим справа в том смысле, что условия $e \in \mathrm{E}$ и $e' \circ e \in \mathrm{E}$ влекут $e' \in \mathrm{E}$. Также он замкнут относительно формирования копроизведений и кодекартовых квадратов (если они существуют) в том смысле, что копроизведение E-морфизмов и ребро кодекартова квадрата, параллельное E-морфизму, содержатся в E. Двойственные утверждения справедливы для F.

Предложение 14. *Пусть* $\mathrm{M} \subseteq \mathrm{Mor}\,\text{c-DESC}$. c-DESC*-морфизм* f *является* M*-инициальным тогда и только тогда, когда условие* $\mathrm{sig}(e) \in \mathrm{Iso}\,\mathrm{SIG}$ *влечёт* $\text{M-Difip}(e, f)$ *для всякого* c-DESC*-морфизма* e.

Доказательство. Пусть $f \in \text{M-Init}$, $\mathrm{sig}(e) \in \mathrm{Iso}\,\mathrm{SIG}$ (так что $e \in \mathrm{Epi}\,\text{c-DESC}$), $f \circ u = m \circ e$ для некоторого $u \in \mathrm{Mor}\,\text{c-DESC}$ и $m \in \mathrm{M}$. Имеем $\mathrm{sig}(f) \circ (\mathrm{sig}(u) \circ \mathrm{sig}(e)^{-1}) = \mathrm{sig}(m)$, поэтому согласно определению 18 существует c-DESC-морфизм $d\colon \mathrm{codom}\,e \to \mathrm{dom}\,f$, такой что $\mathrm{sig}(d) = \mathrm{sig}(u) \circ \mathrm{sig}(e)^{-1}$. Поскольку функтор sig унивалентен, $d \circ e = u$, так что d — искомая диагональ.

Предположим теперь, что имеет место $\text{M-Difip}(\mathrm{sig}^{-1}(\mathrm{Iso}\,\mathrm{SIG}), f)$ для некоторого c-DESC-морфизма $f\colon T \to S$. Выберем морфизмы $m\colon X \to S \in \mathrm{M}$ и $k\colon \mathrm{sig}(X) \to \mathrm{sig}(T)$ так, что $\mathrm{sig}(f) \circ k = \mathrm{sig}(m)$. Введём обозначение $f^* = \mathrm{sig}^*\big(\mathrm{sig}(f)\big)$ для произвольного c-DESC-морфизма f. Имеем $f^* \circ \mathrm{sig}^*(k) = m^*$, поэтому

$$f \circ \big(\varepsilon_T \circ \mathrm{sig}^*(k)\big) = \varepsilon_S \circ f^* \circ \mathrm{sig}^*(k) = \varepsilon_S \circ m^* = m \circ \varepsilon_X.$$

Поскольку $\mathrm{sig}(\varepsilon_X) = 1_{\mathrm{sig}(X)} \in \mathrm{Iso}\,\mathrm{SIG}$, существует диагональ $d\colon X \to T$, такая что $d \circ \varepsilon_X = \varepsilon_T \circ \mathrm{sig}^*(k)$, откуда получаем, что $\mathrm{sig}(d) = k$. Таким образом, c-DESC-морфизм d служит в качестве k^+ из определения 18, так что $f \in \text{M-Init}$.

$$\begin{tikzcd}
X^* \arrow[r, "\mathrm{sig}^*(k)"] \arrow[dr, "m^*"'] \arrow[dd, "\varepsilon_X"'] & T^* \arrow[r, "\varepsilon_T"] \arrow[d, "f^*"] & T \arrow[dd, "f"] \\
 & S^* \arrow[dr, "\varepsilon_S"] & \\
X \arrow[rr, "m"'] \arrow[uurr, dotted, "d"] & & S
\end{tikzcd}$$

□

Предложение 15. *Пусть* $\mathrm{M} \subseteq \mathrm{Mor}\,\text{c-DESC}$. *Любой* c-DESC*-морфизм с дискретной кообластью является* M*-инициальным тогда и только тогда, когда любой* M*-морфизм с дискретной кообластью имеет дискретную область.*

Доказательство. Допустим, что любой c-DESC-морфизм с дискретной кообластью является M-инициальным. Если $m\colon X \to S \in \mathrm{M}$ и $\varepsilon_S \in \mathrm{Iso}\,\text{c-DESC}$, то $\varepsilon_S \circ m^*\colon X^* \to S \in \text{M-Init}$, так что в силу предложения 14 коммутативный квадрат $(\varepsilon_S \circ m^*) \circ 1_{X^*} = m \circ \varepsilon_X$ имеет диагональ d, такую что $d \circ \varepsilon_X = 1_{X^*}$. Поэтому ε_X, будучи эпиморфизмом, обратимым слева, является изоморфизмом [8, предложение 7.43].

Обратно, допустим, что любой M-морфизм с дискретной кообластью имеет дискретную область. Выберем произвольно морфизмы $f\colon T \to S$, $m\colon X \to S \in \mathrm{M}$, $k\colon \mathrm{sig}(X) \to \mathrm{sig}(T)$ так, что $\mathrm{sig}(f) \circ k = \mathrm{sig}(m)$ и $\varepsilon_S \in \mathrm{Iso}\,\text{c-DESC}$ (а значит, и $\varepsilon_X \in \mathrm{Iso}\,\text{c-DESC}$). Имеем $\mathrm{sig}(\varepsilon_T \circ \mathrm{sig}^*(k) \circ \varepsilon_X^{-1}) = k$, так что $f \in \text{M-Init}$. □

Определение 19. Формальная технология специфицирования $\mathrm{SC} = \langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig}\rangle$ *поддерживает* L*-разметки*, где L — некоторый класс SIG-морфизмов, если выполняются следующие условия:

(i) любой L-морфизм является ретракцией;

(ii) существует класс $\mathrm{M} \subseteq \mathrm{Mor}\,\text{c-DESC}$, такой что пара $(\mathrm{sig}^{-1}(\mathrm{L}), \mathrm{M})$ является факторизационной системой в c-DESC и любой M-морфизм с дискретной кообластью имеет дискретную область;

(iii) класс Conf замкнут относительно накачек $\mathrm{sig}^{-1}(\mathrm{L})$-морфизмами.

При этом четвёрка

$$\mathrm{SC}_{\mathrm{L}} = \langle \text{c-DESC}, \mathrm{Conf}, \mathrm{sig}, (\mathrm{Trl}\,\mathrm{L})^{\mathrm{op}}\rangle,$$

где

$$\mathrm{Trl}\,\mathrm{L} = (\mathrm{Ob}\,\text{c-DESC}, \mathrm{sig}^{-1}(\mathrm{L}) \cap \text{M-Init}),$$

называется технологией проектирования над SC, *порождённой классом разметок* L (L-технологией над SC), M-морфизмы называются SC_{L}*-включениями*.

Предложение 16. *Любая* L*-технология* SC_{L} *является коодноролной формальной технологией проектирования, поддерживающей трассирование, с классом всех разметок* L*, причём*

$$\begin{aligned}\mathrm{RegMono}\,\text{c-DESC} \subseteq \mathrm{Mono}\,\text{c-DESC} \cap \text{Mor-Init} = \mathrm{Mono}\,\text{c-DESC} \cap \mathrm{M} = \\ = \mathrm{Mono}\,\text{c-DESC} \cap \text{M-Init} \subseteq \mathrm{M} \subseteq \text{Mor-Init} \subseteq \text{M-Init}.\end{aligned}$$

Доказательство. Прежде всего проверим, что класс M-Init замкнут относительно композиции. Выберем произвольно M-инициальные морфизмы $f\colon T \to R, g\colon R \to S$, M-морфизм $m\colon X \to S$ и SIG-морфизм $k\colon \mathrm{sig}(X) \to \mathrm{sig}(T)$, такие что $\mathrm{sig}(g \circ f) \circ k = \mathrm{sig}(m)$. Найдём c-DESC-морфизм $k^+\colon X \to T$, такой что $\mathrm{sig}(k^+) = k$. Поскольку $g \in \text{M-Init}$ и $\mathrm{sig}(g) \circ (\mathrm{sig}(f) \circ k) = \mathrm{sig}(m)$, согласно определению 18 существует c-DESC-морфизм $n\colon X \to R$, такой что $\mathrm{sig}(n) = \mathrm{sig}(f) \circ k$. Отсюда следует, что $\mathrm{sig}(g \circ n) = \mathrm{sig}(m)$, поэтому $g \circ n = m$ ввиду унивалентности функтора sig. Установим, что $n \in \mathrm{M}$, и тогда искомый c-DESC-морфизм k^+ будет существовать согласно определению 18, поскольку $f \in \text{M-Init}$. Пусть $n = n' \circ e$ и $(g \circ n') = n'' \circ e'$ — $(\mathrm{sig}^{-1}(\mathrm{L}), \mathrm{M})$-разложения. Тогда $m = g \circ n = n'' \circ (e' \circ e)$ — $(\mathrm{sig}^{-1}(\mathrm{L}), \mathrm{M})$-разложение M-морфизма, поэтому $e' \circ e \in \mathrm{Iso}\,\text{c-DESC}$, так что e обратим слева. А любой обратимый слева эпиморфизм является изоморфизмом [8, предложение 7.43], следовательно, $n \in \mathrm{M} \circ \mathrm{Iso}\,\text{c-DESC} = \mathrm{M}$, что и требовалось.

$$\begin{tikzcd}
 & T \arrow[d, "f"] \\
 & R \arrow[d, "g"] \\
X \arrow[r, "m"] \arrow[ur, "n"] \arrow[uur, dashed, "k^+"] & S
\end{tikzcd}$$

Условие (ii) определения 19 гарантирует, что $\mathrm{Iso}\,\text{c-DESC} \subseteq \mathrm{sig}^{-1}(\mathrm{L}) \cap \mathrm{M}$. Отсюда, в частности, вытекает, что

$$\mathrm{Iso}\,\mathrm{SIG} = \mathrm{sig}\big(\mathrm{sig}^*(\mathrm{Iso}\,\mathrm{SIG})\big) \subseteq \mathrm{sig}(\mathrm{Iso}\,\text{c-DESC}) \subseteq \mathrm{L},$$

следовательно, $\mathrm{sig}^{-1}(\mathrm{Iso}\,\mathrm{SIG}) \subseteq \mathrm{sig}^{-1}(\mathrm{L})$, так что ввиду того же условия (ii) определения 19 имеет место $\mathrm{Difip}(\mathrm{sig}^{-1}(\mathrm{Iso}\,\mathrm{SIG}), \mathrm{M})$, поэтому $\mathrm{M} \subseteq \text{Mor-Init}$ по предложению 14. А поскольку снова ввиду условия (ii) определения 19 класс $\mathrm{sig}^{-1}(\mathrm{L})$ замкнут относительно композиции, получаем, что $\mathrm{Trl}\,\mathrm{L}$ действительно является подкатегорией в c-DESC, содержащей все изоморфизмы. (Фактически доказано более сильное утверждение: все c-DESC-объекты и все M-инициальные морфизмы образуют подкатегорию в c-DESC, обладающую факторизационной системой $(\mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}, \mathrm{M})$ [3, предложение 4].) Отсюда с учётом условия (iii) определения 19 выводим, что SC_{L} является кооднородной формальной технологией проектирования.

Согласно условию (i) определения 19 для любого $(\mathrm{Trl}\,\mathrm{L})$-морфизма $t\colon T \to S$ существует SIG-морфизм s, такой что $\mathrm{sig}(t) \circ s = 1_{\mathrm{sig}(S)}$. Поскольку $1_S \in \mathrm{M}$, согласно определению 18 существует c-DESC-морфизм t^+, такой что $t \circ t^+ = 1_S$, так что SC_{L} поддерживает трассирование.

Из заключительной части условия (ii) определения 19 и предложения 15 вытекает, что

$$\mathrm{sig}^*(\mathrm{L}) \subseteq \mathrm{sig}^{-1}(\mathrm{L}) \cap \text{M-Init} = \mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L},$$

поэтому

$$\mathrm{sig}(\mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}) \supseteq \mathrm{sig}\big(\mathrm{sig}^*(\mathrm{L})\big) = \mathrm{L}.$$

Осталось проследить цепочку включений классов морфизмов, указанную в условии доказываемого утверждения. Поскольку функтор sig ввиду наличия у него левого сопряжённого сохраняет регулярные мономорфизмы [6, разд. 5.5], любой регулярный c-DESC-мономорфизм является инициальным [8, предложение 8.7 (3)]. Пусть $m\colon P \to Q$ — произвольный M-инициальный c-DESC-мономорфизм, $m = m' \circ e$ — его $(\mathrm{sig}^{-1}(\mathrm{L}), \mathrm{M})$-разложение. Тогда e — мономорфизм, так что $\mathrm{sig}(e) \in \mathrm{Iso}\,\mathrm{SIG}$, поскольку функтор sig ввиду наличия у него левого сопряжённого сохраняет мономорфизмы [6, разд. 5.5]. По предложению 14 коммутативный квадрат $m \circ 1_P = m' \circ e$ имеет диагональ d, такую что $d \circ e = 1_P$, следовательно, $e \in \mathrm{Iso}\,\text{c-DESC}$, так что $m \in \mathrm{M}$. Таким образом, $\mathrm{Mono}\,\text{c-DESC} \cap \text{M-Init} \subseteq \mathrm{M}$. Наконец, соотношение $\mathrm{M} \subseteq \text{Mor-Init}$ установлено выше, во втором абзаце доказательства. □

Предложение 17. *В произвольной* L*-технологии* SC_{L} *любой* $\mathrm{sig}^{-1}(\mathrm{L})$*-морфизм является трассой трансформации тогда и только тогда, когда функтор* sig *является эквивалентностью категорий* c-DESC *и* SIG.

Доказательство. Имеем следующую цепочку равносильных утверждений:

$$\mathrm{sig} \text{ — эквивалентность} \iff$$

$$\overset{\text{по определению эквивалентности категорий}}{\iff} \quad \varepsilon \subseteq \mathrm{Iso}\,\text{c-DESC} \iff$$

$$\overset{f\circ\varepsilon_A=\varepsilon_B\circ f^*\ \text{для любого}\ f\colon A\to B}{\iff} \quad \mathrm{sig}^{-1}(\mathrm{Iso\,SIG}) = \mathrm{Iso\,c\text{-}DESC} \iff$$

$$\overset{\text{учитываем, что}\ \mathrm{L}\supseteq\mathrm{Iso\,SIG}\ \text{и}\ \mathrm{M}\supseteq\mathrm{Iso\,c\text{-}DESC}}{\iff} \quad \mathrm{M\text{-}Difip}\big(\mathrm{sig}^{-1}(\mathrm{Iso\,SIG}), \mathrm{sig}^{-1}(\mathrm{L})\big) \iff$$

$$\overset{\text{по предложению 14}}{\iff} \quad \mathrm{sig}^{-1}(\mathrm{L}) \subseteq \mathrm{M\text{-}Init}$$

$$\overset{\text{по определению 19}}{\iff} \quad \mathrm{Mor\,Trl\,L} = \mathrm{sig}^{-1}(\mathrm{L}). \qquad \square$$

Предложение 18. *В произвольной* L*-технологии* $\mathrm{SC_L}$ *любая трансформация является* c-DESC*-изоморфизмом тогда и только тогда, когда* $\mathrm{L} = \mathrm{Iso\,SIG}$.

Доказательство. Необходимость следует из предложений 16 и 3. Для доказательства достаточности заметим, что согласно условию (ii) определения 19 и предложению 14 условие $\mathrm{L} = \mathrm{Iso\,SIG}$ влечёт $\mathrm{M} = \mathrm{Mor\text{-}Init}$. Но любой инициальный c-DESC-морфизм, переходящий в изоморфизм под действием функтора sig, сам является изоморфизмом [8, предложение 8.14]. $\square$

Теорема 4. *Пусть* $\mathrm{SC}' = \langle \mathrm{c\text{-}DESC}', \mathrm{Conf}', \mathrm{sig}'\colon \mathrm{c\text{-}DESC}' \to \mathrm{SIG}'\rangle$ — *произвольная формальная технология специфицирования,* $\langle \mathrm{cm}, \mathrm{sm}\rangle\colon \mathrm{SC} \to \mathrm{SC}'$ — $\mathbf{conf}$*-трасса, причём функтор* sm *является изоморфизмом категорий. Если* SC *поддерживает* L*-разметки, то* SC' *поддерживает* $\mathrm{sm(L)}$*-разметки, причём класс всех* $\mathrm{SC}'_{\mathrm{sm(L)}}$*-включений имеет вид* $\mathrm{cm(M)} \circ \mathrm{Iso\,c\text{-}DESC}'$*, где* M — *класс всех* $\mathrm{SC_L}$*-включений.*

Доказательство. Введём обозначения:

$$\dot{\mathrm{L}} = \mathrm{sig}'^{-1}\big(\mathrm{sm(L)}\big), \quad \dot{\mathrm{M}} = \mathrm{cm(M)} \circ \mathrm{Iso\,c\text{-}DESC}',$$

ε' — коединица сопряжения $\mathrm{sig}'^* \dashv \mathrm{sig}'$, ζ — коединица сопряжения $\mathrm{cm}^* \dashv \mathrm{cm}$. Мы будем пользоваться следующими свойствами $\mathbf{SPEC}$-морфизма $\langle \mathrm{cm}, \mathrm{sm}\rangle$:

а) $\mathrm{sm} \circ \mathrm{sig} = \mathrm{sig}' \circ \mathrm{cm}$ (условие (iii) определения 3);

б) $\mathrm{cm}\big(\mathrm{sig}^{-1}(\mathrm{L})\big) \subseteq \dot{\mathrm{L}}$ (из а) следует, что $\mathrm{sig}'\big(\mathrm{cm}(q)\big) = \mathrm{sm}\big(\mathrm{sig}(q)\big) \in \mathrm{sm(L)}$ для любого $q \in \mathrm{sig}^{-1}(\mathrm{L})$);

в) $\mathrm{sig} = \mathrm{sm}^{-1} \circ \mathrm{sig}' \circ \mathrm{cm}$ (вытекает из а));

г) $\mathrm{cm}^*(\dot{\mathrm{L}}) \subseteq \mathrm{sig}^{-1}(\mathrm{L})$ (с учётом соотношения $\mathrm{cm} \circ \mathrm{cm}^* = 1_{\mathrm{c\text{-}DESC}'}$ из в) следует, что

$$\mathrm{sig}\big(\mathrm{cm}^*(h)\big) = \mathrm{sm}^{-1}\Big(\mathrm{sig}'\Big(\mathrm{cm}\big(\mathrm{cm}^*(h)\big)\Big)\Big) = \mathrm{sm}^{-1}\big(\mathrm{sig}'(h)\big) \in \mathrm{L}$$

для любого $h \in \dot{\mathrm{L}}$).

Проверим выполнение всех условий определения 19 для технологии SC' и класса SIG'-морфизмов $\mathrm{sm(L)}$.

Проверим выполнение условия (i). Любой функтор сохраняет ретракции, поэтому $\mathrm{sm(L)}$ состоит из ретракций.

Проверим выполнение условия (ii). Положим

$$\dot{\mathrm{M}}' = \{\mathrm{cm}(m) \mid m \in \mathrm{M},\ \zeta_{\mathrm{codom}\,m} = 1_{\mathrm{codom}\,m}\} \circ \mathrm{Iso\,c\text{-}DESC}'$$

и установим наличие факторизационной системы $(\dot{\mathrm{L}}, \dot{\mathrm{M}}')$ в c-DESC$'$. Сначала проверим, что $j \circ g \in \dot{\mathrm{L}}$ для любых $j \in \mathrm{Iso}\,\text{c-DESC}'$ и $g \in \dot{\mathrm{L}}$. Ввиду утверждения г) и условия (ii) определения 19

$$\mathrm{cm}^*(j \circ g) \in \mathrm{Iso}\,\text{c-DESC} \circ \mathrm{cm}^*(\dot{\mathrm{L}}) \subseteq \mathrm{Iso}\,\text{c-DESC} \circ \mathrm{sig}^{-1}(\mathrm{L}) \subseteq \mathrm{sig}^{-1}(\mathrm{L}).$$

Поэтому

$$j \circ g = \mathrm{cm}\big(\mathrm{cm}^*(j \circ g)\big) \in \mathrm{cm}\big(\mathrm{sig}^{-1}(\mathrm{L})\big) \subseteq \dot{\mathrm{L}}$$

с учётом утверждения б). Ввиду условия (ii) определения 19 для любого c-DESC$'$-морфизма f существует $(\mathrm{sig}^{-1}(\mathrm{L}), \mathrm{M})$-разложение $\mathrm{cm}^*(f) = n \circ e$, переходящее под действием функтора cm в равенство $f = \mathrm{cm}(n) \circ \mathrm{cm}(e)$, которое ввиду утверждения б) представляет собой $(\dot{\mathrm{L}}, \dot{\mathrm{M}}')$-разложение. Проверим $\mathrm{Difip}(\dot{\mathrm{L}}, \dot{\mathrm{M}}')$. Если $(\mathrm{cm}(m) \circ i) \circ u = v \circ l$ для некоторых $m\colon A \to B \in \mathrm{M}$, $i \in \mathrm{Iso}\,\text{c-DESC}'$, $l \in \dot{\mathrm{L}}$, $u, v \in \mathrm{Mor}\,\text{c-DESC}'$, то, полагая $w = \mathrm{cm}^*(i \circ u)$, имеем

$$m \circ (\zeta_A \circ w) = \zeta_B \circ \mathrm{cm}^*\big(\mathrm{cm}(m)\big) \circ w = \big(\zeta_B \circ \mathrm{cm}^*(v)\big) \circ \mathrm{cm}^*(l),$$

поэтому ввиду $\mathrm{Difip}(\mathrm{sig}^{-1}(\mathrm{L}), \mathrm{M})$ (условие (ii) определения 19) и утверждения г) существует c-DESC-морфизм d, такой что $d \circ \mathrm{cm}^*(l) = \zeta_A \circ w$. Действуя на это равенство функтором cm, получаем, что $\mathrm{cm}(d) \circ l = i \circ u$, так что $i^{-1} \circ \mathrm{cm}(d)$ — искомая диагональ.

$$\begin{tikzcd}
{} \arrow[r, "\mathrm{cm}^*(u)"] \arrow[d, "\mathrm{cm}^*(l)"'] & {} \arrow[r, "\mathrm{cm}^*(i)"] & \mathrm{cm}^*\big(\mathrm{cm}(A)\big) \arrow[r, "\zeta_A"] \arrow[d, "\mathrm{cm}^*\left(\mathrm{cm}(m)\right)"] & A \arrow[d, "m"] \\
{} \arrow[rr, "\mathrm{cm}^*(v)"'] \arrow[urrr, dotted, "d"] & & \mathrm{cm}^*\big(\mathrm{cm}(B)\big) \arrow[r, "\zeta_B"'] & B
\end{tikzcd}$$

Проверим, что для любого $\dot{\mathrm{M}}'$-морфизма $p\colon X \to Y$, такого что $\varepsilon'_Y \in \mathrm{Iso}\,\text{c-DESC}'$, имеет место $\varepsilon'_X \in \mathrm{Iso}\,\text{c-DESC}'$. Пусть $p = \mathrm{cm}(r) \circ t$ для некоторых $r\colon P \to Q \in \mathrm{M}$ и $t\colon X \to \mathrm{cm}(P) \in \mathrm{Iso}\,\text{c-DESC}'$, таких что $\zeta_Q = 1_Q$ и $\mathrm{cm}(Q) = Y$. Из утверждения в) и правил вычисления композиции сопряжений функторов [6, разд. 4.8] вытекает, что $\varepsilon_Z = \zeta_Z \circ \mathrm{cm}^*(\varepsilon'_{\mathrm{cm}(Z)})$ для любого c-DESC-объекта Z, в частности, по предположению $\varepsilon_Q = \mathrm{cm}^*(\varepsilon'_Y) \in \mathrm{Iso}\,\text{c-DESC}$. Поэтому ввиду условия (ii) определения 19 $\varepsilon_P \in \mathrm{Iso}\,\text{c-DESC}$. Подставляя P вместо Z, убеждаемся, что c-DESC-морфизм $\mathrm{cm}^*(\varepsilon'_{\mathrm{cm}(P)})$ обратим слева. Но тогда он является изоморфизмом, поскольку он эпиморфизм (коединица ε' состоит из эпиморфизмов, а поскольку функтор cm^* обладает правым сопряжённым, он сохраняет все копределы [6, разд. 5.5], в частности все эпиморфизмы). Отсюда вытекает, что

$$\varepsilon'_{\mathrm{cm}(P)} = \mathrm{cm}\big(\mathrm{cm}^*(\varepsilon'_{\mathrm{cm}(P)})\big) \in \mathrm{Iso}\,\text{c-DESC}',$$

следовательно,

$$\varepsilon'_X = t^{-1} \circ \varepsilon'_{\mathrm{cm}(P)} \circ \mathrm{cm}^*\big(\mathrm{cm}(t)\big) \in \mathrm{Iso}\,\text{c-DESC}'.$$

Осталось убедиться, что $\dot{\mathrm{M}} = \dot{\mathrm{M}}'$. С одной стороны, по построению $\dot{\mathrm{M}}' \subseteq \dot{\mathrm{M}}$. С другой стороны, имеет место $\mathrm{Difip}(\dot{\mathrm{L}}, \dot{\mathrm{M}})$ (это соотношение проверяется дословно так же, как $\mathrm{Difip}(\dot{\mathrm{L}}, \dot{\mathrm{M}}')$ выше), поэтому $\dot{\mathrm{M}} \subseteq \dot{\mathrm{M}}'$.

Проверим выполнение условия (iii). Пусть $\tau \rightrightarrows \Sigma$ — накачка произвольной диаграммы $\Sigma \in \mathrm{Conf}'$ произвольным семейством $\dot{\mathrm{L}}$-морфизмов τ. Рассмотрим диаграмму $\Delta = \mathrm{cm}^* \circ (\tau \rightrightarrows \Sigma) = \mathrm{cm}^*(\tau) \rightrightarrows \mathrm{cm}^* \circ \Sigma$. Ввиду утверждения г) семейство $\mathrm{cm}^*(\tau)$ состоит из $\mathrm{sig}^{-1}(\mathrm{L})$-морфизмов, а с учётом определения $\mathbf{conf}$-трассы соотношение $\mathrm{cm} \circ \mathrm{cm}^* \circ \Sigma = \Sigma$ влечёт $\mathrm{cm}^* \circ \Sigma \in \mathrm{Conf}$. Следовательно, $\Delta \in \mathrm{Conf}$ ввиду условия (iii) определения 19, так что $\tau \rightrightarrows \Sigma = \mathrm{cm} \circ \Delta \in \mathrm{Conf}'$ ввиду условия (i) определения 3. $\square$

Следствие 4.1. *Если* SC *поддерживает* L*-разметки, то тройка* $\mathrm{SSIG} = \langle \mathrm{SIG}, \mathrm{sig} \circ \mathrm{Conf}, 1_{\mathrm{SIG}} \rangle$ *является формальной технологией специфицирования, поддерживающей* L*-разметки, для которой класс* $\mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}$ *совпадает с* L*, а класс всех включений — с* $\mathrm{sig}(\mathrm{M})$*, где* M *— класс всех* $\mathrm{SC_L}$*-включений. Функтор* sig *индуцирует трансформацию* L*-технологии* $\mathrm{SSIG_L}$ *в* $\mathrm{SC_L}$.

Доказательство. Ввиду следствия 3.1 и предложения 7 имеем $\mathrm{SSIG} = \mathbf{conf}^*\big(\mathbf{conf}^{**}(\mathrm{SC})\big) \in \mathrm{Ob}\,\mathbf{SPEC}$. Из теорем 3 и 2 вытекает, что компонента единицы сопряжения $\mathbf{conf}^{**} \dashv \mathbf{conf}^*$ переводит формальную технологию SC в трассу $\langle \mathrm{sig}, 1_{\mathrm{SIG}} \rangle \colon \mathrm{SC} \to \mathrm{SSIG}$. Теперь применяем теорему 4. В частности, как установлено в её доказательстве, любое $\mathrm{SSIG_L}$-включение имеет вид $\mathrm{sig}(m) \circ i$ для некоторых $m \colon P \to Q \in \mathrm{M}$ и $i \in \mathrm{Iso}\,\mathrm{SIG}$, причём $\varepsilon_Q = 1_Q$. Но тогда ввиду условия (ii) определения 19 $\varepsilon_P \in \mathrm{Iso}\,\text{c-DESC}$, так что $\mathrm{sig}(m) \circ i = \mathrm{sig}\big(m \circ \varepsilon_P \circ \mathrm{sig}^*(i)\big) \in \mathrm{sig}(\mathrm{M})$, поскольку класс M замкнут относительно композиции и содержит все c-DESC-изоморфизмы.

Наконец, поскольку $\mathrm{SSIG_L} = \langle \mathrm{SIG}, \mathrm{sig} \circ \mathrm{Conf}, 1_{\mathrm{SIG}}, (\mathrm{Ob}\,\mathrm{SIG}, \mathrm{L})^{\mathrm{op}} \rangle$, имеется $\mathbf{ARCH}$-морфизм $\langle \mathrm{sig}, 1_{\mathrm{SIG}}, \mathrm{sig}(-^{\mathrm{op}})^{\mathrm{op}} \rangle \colon \mathrm{SC_L} \to \mathrm{SSIG_L}$, который является трассой трансформации технологий проектирования (условие (ii) определения 17 обеспечивается выбором функтора sig^* в качестве drm ввиду соотношения $\mathrm{sig}^*(\mathrm{L}) \subseteq \mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}$), вытекающего из предложения 15). $\square$

Следствие 4.2. *Если* SC *поддерживает* L*-разметки, то все* SIG*-объекты и все* L*-морфизмы образуют подкатегорию в* SIG*, содержащую все* SIG*-изоморфизмы.*

Доказательство. Утверждение непосредственно вытекает из следствия 4.1. $\square$

Следствие 4.3. *Пусть* $\mathrm{SC}' = \langle \text{c-DESC}', \mathrm{Conf}', \mathrm{sig}' \colon \text{c-DESC}' \to \mathrm{SIG}' \rangle$ *— произвольная формальная технология специфицирования,* $\langle \mathrm{cm}, \mathrm{sm} \rangle \colon \mathrm{SC} \to \mathrm{SC}'$ *—* $\mathbf{SPEC}$*-морфизм, такой что* cm *—* $\mathbf{desc}$*-разметка,* sm *— изоморфизм категорий. Если* SC *поддерживает* L*-разметки и* SC' *поддерживает* L'*-разметки, то соотношение* $\mathrm{sm}(\mathrm{L}) \subseteq \mathrm{L}'$ *выполняется тогда и только тогда, когда тройка* $\langle \mathrm{cm}, \mathrm{sm}, \mathrm{cm}(-^{\mathrm{op}})^{\mathrm{op}} \rangle$ *представляет собой* $\mathbf{ARCH}$*-морфизм из* $\mathrm{SC_L}$ *в* $\mathrm{SC}'_{\mathrm{L}'}$.

Доказательство. Обозначим через M' класс всех $\mathrm{SC}'_{\mathrm{L}'}$-включений. Мы будем пользоваться коединицей ζ, факторизационной системой $(\dot{\mathrm{L}}, \dot{\mathrm{M}})$ в $\text{c-DESC}'$

и утверждениями а)—г), которые вводятся точно так же, как в доказательстве теоремы 4.

Сначала предположим, что $\mathrm{sm}(\mathrm{L}) \subseteq \mathrm{L}'$, так что $\dot{\mathrm{L}} \subseteq \mathrm{sig}'^{-1}(\mathrm{L}')$, и проверим, что $\mathrm{cm}(f) \in \mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}'$ для произвольного $(\mathrm{Trl}\,\mathrm{L})$-морфизма $f\colon A \to B$. Ввиду утверждения б) по предположению имеем $\mathrm{cm}(f) \in \dot{\mathrm{L}} \subseteq \mathrm{sig}'^{-1}(\mathrm{L}')$. Далее воспользуемся предложением 14: установим, что $\mathrm{M}'\text{-}\mathrm{Difip}\big(e, \mathrm{cm}(f)\big)$ для произвольного $e \in \mathrm{sig}'^{-1}(\mathrm{Iso}\,\mathrm{SIG}')$. Пусть $\mathrm{cm}(f) \circ u = m' \circ e$ — произвольный коммутативный квадрат, такой что $m' \in \mathrm{M}'$. По предположению имеет место $\mathrm{Difip}(\dot{\mathrm{L}}, m')$, поэтому $m' \in \dot{\mathrm{M}}$, т. е. $m' = \mathrm{cm}(m) \circ i$ для некоторых $m \in \mathrm{M}$ и $i \in \mathrm{Iso}\,\mathrm{c\text{-}DESC}'$. Действуя функтором cm^* на наш квадрат, получаем соотношение

$$\mathrm{cm}^*\big(\mathrm{cm}(f)\big) \circ \mathrm{cm}^*(u) = \mathrm{cm}^*\big(\mathrm{cm}(m)\big) \circ \mathrm{cm}^*(i) \circ \mathrm{cm}^*(e).$$

Умножая это равенство слева на ζ_B и применяя тождество коединицы $f \circ \zeta_A = \zeta_B \circ \mathrm{cm}^*\big(\mathrm{cm}(f)\big)$, получаем соотношение $f \circ \zeta_A \circ \mathrm{cm}^*(u) = w \circ \mathrm{cm}^*(i) \circ \mathrm{cm}^*(e)$, где $w = \zeta_B \circ \mathrm{cm}^*\big(\mathrm{cm}(m)\big)$. Пусть $w = n \circ s$ — $(\mathrm{sig}^{-1}(\mathrm{L}), \mathrm{M})$-разложение. Ввиду следствия 4.1 соотношение $\mathrm{sig}(w) = \mathrm{sig}(n) \circ \mathrm{sig}(s)$ представляет собой $\big(\mathrm{L}, \mathrm{sig}(\mathrm{M})\big)$-разложение, а поскольку с учётом утверждения в) имеем

$$\mathrm{sig}(w) = \mathrm{sm}^{-1}\Big(\mathrm{sig}'\big(\mathrm{cm}(\zeta_B \circ m)\big)\Big) = \mathrm{sig}(m) \in \mathrm{sig}(\mathrm{M}),$$

получаем $\mathrm{sig}(s) \in \mathrm{Iso}\,\mathrm{SIG}$. Окончательно имеем в c-DESC коммутативный квадрат $f \circ h = n \circ g$, где $h = \zeta_A \circ \mathrm{cm}^*(u)$ и $g = s \circ \mathrm{cm}^*(i) \circ \mathrm{cm}^*(e)$, так что ввиду утверждения в)

$$\mathrm{sig}(g) = \mathrm{sig}(s) \circ \mathrm{sig}\big(\mathrm{cm}^*(i)\big) \circ \mathrm{sm}^{-1}\big(\mathrm{sig}'(e)\big) \in \mathrm{Iso}\,\mathrm{SIG}.$$

А поскольку $f \in \mathrm{M\text{-}Init}$ и $n \in \mathrm{M}$, ввиду предложения 14 существует диагональ d, такая что $d \circ g = h$. Действуя на это соотношение функтором cm, получаем, что $d' \circ e = u$, где $d' = \mathrm{cm}(d) \circ \mathrm{cm}(s) \circ i$, так что d' — искомая диагональ исходного квадрата. Следовательно,

$$\mathrm{cm}(f) \in \mathrm{sig}'^{-1}(\mathrm{L}') \cap \mathrm{M}'\text{-}\mathrm{Init} = \mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}'.$$

$$\begin{tikzcd}
{} \arrow[r, "\mathrm{cm}^*(u)"] \arrow[d, "\mathrm{cm}^*(e)"'] & \mathrm{cm}^*\big(\mathrm{cm}(A)\big) \arrow[r, "\zeta_A"] \arrow[d, "\mathrm{cm}^*(\mathrm{cm}(f))"] & A \arrow[ddd, "f"] \\
{} \arrow[r, "\mathrm{cm}^*(m')"] \arrow[d, "\mathrm{cm}^*(i)"'] & \mathrm{cm}^*\big(\mathrm{cm}(B)\big) \arrow[ddr, "\zeta_B"] & \\
{} \arrow[ur, "\mathrm{cm}^*(\mathrm{cm}(m))"'] \arrow[drr, "w"] \arrow[d, "s"'] & & \\
{} \arrow[rr, "n"] \arrow[uuurr, dotted, "d"] & & B
\end{tikzcd}$$

Теперь предположим, что $\mathrm{cm}(\mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}) \subseteq \mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}'$, и проверим, что $\mathrm{sm}(l) \in \mathrm{L}'$ для произвольного L-морфизма l. Из заключительной части условия (ii) определения 19 и предложения 15 вытекает, что $\mathrm{sig}^*(l) \in \mathrm{Mor}\,\mathrm{Trl}\,\mathrm{L}$, поэтому с учётом утверждения а)

$$\mathrm{sm}(l) = \mathrm{sm}\Big(\mathrm{sig}\big(\mathrm{sig}^*(l)\big)\Big) = \mathrm{sig}'\Big(\mathrm{cm}\big(\mathrm{sig}^*(l)\big)\Big) \in \mathrm{L}'. \qquad \square$$

Следствие 4.4. *Технология специфицирования* $\mathbf{spec}\big(\mathrm{AO}_{\mathrm{int}}(\mathrm{SC}_{\mathrm{L}})\big)$ *поддерживает* L*-разметки, причём* L*-технология* $\mathbf{spec}\big(\mathrm{AO}_{\mathrm{int}}(\mathrm{SC}_{\mathrm{L}})\big)_{\mathrm{L}}$ *является подтехнологией в* $\mathrm{AO}_{\mathrm{int}}(\mathrm{SC}_{\mathrm{L}})$. *Морфизм АО-моделей над* SC_{L} *является включением тогда и только тогда, когда функторы* mod *и* str *переводят его во включения* (*т. е. в* M*-морфизм и в* sig(M)*-морфизм соответственно*).

Доказательство. Построим категории AO, LAB, tr-AO и функторы il, mod, int, asp, str из составляющих технологии SC_{L}, как описано в разделе 2. Положим

$$\mathrm{M}_{\mathrm{AO}} = \{\langle a, x\rangle \mid a \in \mathrm{M},\ x \in \mathrm{sig}(\mathrm{M})\} \subseteq \mathrm{Mor}\,\mathrm{AO}$$

и проверим, что пара $(\mathrm{int}^{-1}(\mathrm{L}), \mathrm{M}_{\mathrm{AO}})$ является факторизационной системой в категории AO. Пусть $\langle f, b\rangle\colon \langle A, l\rangle \to \langle B, h\rangle$ — произвольный AO-морфизм, $f = m \circ e$ — $(\mathrm{sig}^{-1}(\mathrm{L}), \mathrm{M})$-разложение, $b = p \circ q$ и $\big(h \circ \mathrm{sig}(m)\big) = n \circ s$ — $\big(\mathrm{L}, \mathrm{sig}(\mathrm{M})\big)$-разложения. Поскольку

$$n \circ \big(s \circ \mathrm{sig}(e)\big) = h \circ \mathrm{sig}(m \circ e) = h \circ \mathrm{sig}(f) = b \circ l = p \circ (q \circ l) —$$

$\big(\mathrm{L}, \mathrm{sig}(\mathrm{M})\big)$-разложение, существует SIG-изоморфизм i, такой что $p = n \circ i$ и $s \circ \mathrm{sig}(e) = i \circ q \circ l$. Тогда $\langle f, b\rangle = \langle m, n\rangle \circ \langle e, i \circ q\rangle$ — $(\mathrm{int}^{-1}(\mathrm{L}), \mathrm{M}_{\mathrm{AO}})$-разложение с промежуточным объектом $\langle \mathrm{dom}\, m, s\rangle$. Диагональ произвольного коммутативного квадрата $r \circ u = v \circ w$ в AO, такого что $r \in \mathrm{M}_{\mathrm{AO}}$ и $\mathrm{int}(w) \in \mathrm{L}$, представляет собой пару $\langle d, d'\rangle$, где d — диагональ mod-образа этого квадрата, d' — диагональ его str-образа.

Предположим, что $\langle f, b\rangle \in \mathrm{Mor}\,\mathrm{M}_{\mathrm{AO}}$ и $\langle B, h\rangle$ — дискретный объект, т. е. $\varepsilon_B \in \mathrm{Iso}\,\text{c-DESC}$ и $h \in \mathrm{Iso}\,\mathrm{SIG}$ (согласно доказательству теоремы 1). Поскольку $f \in \mathrm{M}$, имеем $\varepsilon_A \in \mathrm{Iso}\,\text{c-DESC}$. Кроме того, равенство $\big(h \circ \mathrm{sig}(f)\big) = b \circ l$ задаёт $\big(\mathrm{L}, \mathrm{sig}(\mathrm{M})\big)$-разложение sig(M)-морфизма $h \circ \mathrm{sig}(f)$, поэтому $l \in \mathrm{Iso}\,\mathrm{SIG}$. Таким образом, $\langle A, l\rangle$ также дискретный объект.

Осталось показать, что если $\langle f, b\rangle \in \mathrm{int}^{-1}(\mathrm{L}) \cap \mathrm{M}_{\mathrm{AO}}\text{-Init}$, то $f \in \text{M-Init}$, так что категория $\mathrm{Trl}\,\mathrm{L}$, построенная из составляющих технологии $\mathbf{spec}\big(\mathrm{AO}_{\mathrm{int}}(\mathrm{SC}_{\mathrm{L}})\big)$, содержится в tr-AO в качестве подкатегории и функтор 1_{AO} индуцирует вложение технологии $\mathbf{spec}\big(\mathrm{AO}_{\mathrm{int}}(\mathrm{SC}_{\mathrm{L}})\big)_{\mathrm{L}}$ в $\mathrm{AO}_{\mathrm{int}}(\mathrm{SC}_{\mathrm{L}})$. Пусть $\mathrm{sig}(f) \circ k = \mathrm{sig}(m)$ для некоторых SIG-морфизма k и M-морфизма $m\colon X \to B$. Если $\big(h \circ \mathrm{sig}(m)\big) = n \circ s$ — $\big(\mathrm{L}, \mathrm{sig}(\mathrm{M})\big)$-разложение, то имеется M_{AO}-морфизм $\langle m, n\rangle\colon \langle X, s\rangle \to \langle B, h\rangle$, поэтому существует AO-морфизм $k'\colon \langle X, s\rangle \to \langle A, l\rangle$, такой что $\mathrm{int}(k') = k$, и можно выбрать $k^+ = \mathrm{mod}(k')$, чтобы удовлетворить определению 18 для f. $\square$

Следствие 4.5. *Категория* tr-AO*, порождённая технологией* $\mathrm{SC_L}$*, совпадает с категорией запятой* $\mathrm{sig_L} \downarrow (\mathrm{Ob\,SIG}, \mathrm{L})$*, где функтор* $\mathrm{sig_L}\colon \mathrm{Trl\,L} \to (\mathrm{Ob\,SIG}, \mathrm{L})$ *действует так же, как* sig*. Поэтому трансформация в АО-технологии над* $\mathrm{SC_L}$ *обладает экспликацией тогда и только тогда, когда её область и кообласть находятся в аспектном ядре.*

Доказательство. Для всякого tr-AO-морфизма $\langle t, b\rangle\colon \langle A, l\rangle \to \langle B, h\rangle$ имеем $b \in \mathrm{L}$, поскольку $b \circ l = h \circ \mathrm{sig}(t) \in \mathrm{L}$, $l \in \mathrm{L}$, а по следствию 4.1 класс L слабо сократим справа.

По предложениям 16 и 9 любая экспликация аспектной структуры АО-модели в АО-технологии над $\mathrm{SC_L}$ универсальна, поэтому эксплицируемая трансформация может иметь место только между элементами аспектного ядра. Установим, что любая их трансформация эксплицируема. Предположим, что АО-модели $\langle A, l\rangle$ и $\langle B, h\rangle$ обладают экспликациями $s\colon P \to A$ и $r\colon Q \to B$ соответственно. Пусть $q\colon P \to Q$ — экспликация действия tr-AO-морфизма $\langle t, b\rangle$ вдоль s и r, т. е. $q \circ s^{\mathrm{op}} = r^{\mathrm{op}} \circ t$. Ввиду трассируемости трансформации s имеем, что $\mathrm{sig}(q) = b \in \mathrm{L}$. Проверим, что $q \in \mathrm{M}$-Init, при помощи предложения 14: найдём диагональ у произвольного коммутативного квадрата $q \circ u = m \circ e$, где $\mathrm{sig}(e) \in \mathrm{Iso\,SIG}$ и $m \in \mathrm{M}$. Пусть s' — c-DESC-морфизм, правый обратный к s^{op}. По тому же предложению 14 коммутативный квадрат

$$(r^{\mathrm{op}} \circ t) \circ (s' \circ u) = (q \circ s^{\mathrm{op}}) \circ (s' \circ u) = m \circ e$$

имеет диагональ d, такую что $d \circ e = s' \circ u$. Тогда $(s^{\mathrm{op}} \circ d) \circ e = s^{\mathrm{op}} \circ s' \circ u = u$, так что $s^{\mathrm{op}} \circ d$ — искомая диагональ исходного квадрата. Окончательно имеем, что $q \in \mathrm{Mor\,Trl\,L}$.

$$\begin{tikzcd}
{} \arrow[r, "u"] \arrow[dd, "e"'] & P \arrow[r, "s'"] \arrow[dr, "1_P"'] & A \arrow[d, "s^{\mathrm{op}}"] \\
& & P \arrow[d, "q"] \\
{} \arrow[rr, "m"] \arrow[uurr, dotted, "d"] & & Q
\end{tikzcd}$$ □

Следствие 4.6. *Технология* $\mathrm{SC_L}$ *аспектно полна тогда и только тогда, когда любая* SIG*-диаграмма из класса* $\mathrm{str} \circ \mathbf{D}\mathrm{mod}^{-1}(\mathrm{Conf})$ *обладает копределом.*

Доказательство. Необходимость вытекает из определений 16 и 7. Для доказательства достаточности ввиду условия (iii) определения 19 нужно показать, что произвольная диаграмма $\Delta\colon X \to \mathrm{AO}$, такая что $\mathrm{mod} \circ \Delta \in \mathrm{Conf}$, аспектно детерминирована. Сначала опишем процедуру построения копредела LAB-диаграммы $\mathrm{asp} \circ \Delta$. Функтор $\mathrm{int} \circ \Delta$ связан с $\mathrm{str} \circ \Delta$ естественным преобразованием β (т. е. SIG^X-морфизмом) с компонентами

$\beta_I = \mathrm{asp}\big(\Delta(I)\big)\colon \mathrm{int}\big(\Delta(I)\big) \to \mathrm{str}\big(\Delta(I)\big)$, $I \in \mathrm{Ob}\,X$, которое индуцирует $\mathbf{D}$SIG-морфизм $\langle\beta, 1_X\rangle\colon \mathrm{int}\circ\Delta \to \mathrm{str}\circ\Delta$. Поэтому для любых копределов $\rho\colon \mathrm{int}\circ\Delta \to \ulcorner P\urcorner$ и $\upsilon\colon \mathrm{str}\circ\Delta \to \ulcorner Q\urcorner$ имеется стрелка копредела — единственный SIG-морфизм $u\colon P \to Q$, такой что $\upsilon\circ\langle\beta, 1_X\rangle = \ulcorner u\urcorner\circ\rho$. Непосредственно проверяется, что поскольку согласно следствию 4.1 выполняется $\mathrm{Difip}\big(\{\beta_I \mid I \in \mathrm{Ob}\,X\}, \mathrm{sig(M)}\big)$, имеет место $\mathrm{Difip}\big(u, \mathrm{sig(M)}\big)$ (общее двойственное утверждение приводится в [13]). Следовательно, $u \in \mathrm{L}$, так что коконус $\langle\rho, \upsilon\rangle\colon \mathrm{asp}\circ\Delta \to \ulcorner u\urcorner$ является копределом LAB-диаграммы $\mathrm{asp}\circ\Delta$.

В качестве ρ выберем коконус $\mathrm{sig}\circ\delta$, где $\delta\colon \mathrm{mod}\circ\Delta \to \ulcorner A\urcorner$ — копредел c-DESC-диаграммы $\mathrm{mod}\circ\Delta$ (его существование вытекает из условия (i) определения 1). Копредел υ существует по предположению, так что с учётом вышеизложенного непосредственно проверяется, что коконус $\langle\delta, \upsilon\rangle\colon \Delta \to \ulcorner\langle A, u\rangle\urcorner$ является копределом диаграммы Δ, т. е. функтор $\langle\mathrm{mod}, \mathrm{str}\rangle\colon \mathrm{AO} \to \text{c-DESC} \times \mathrm{SIG}$ поднимает её копределы. По предложению 1 диаграмма Δ аспектно детерминирована. $\square$

Чтобы проиллюстрировать значение конструкции L-технологии, покажем, что технология событийного моделирования является единственной технологией над технологией специфицирования $\mathrm{SS} = \mathbf{spec}(\mathrm{SM})$, порождённой некоторым классом разметок. Действительно, класс разметок L над технологией SS ввиду условия (i) определения 19 должен состоять из сюръективных отображений множеств, а ввиду следствия 4.1 он должен служить первой компонентой факторизационной системы в категории $\mathbf{Set}$. Всего в $\mathbf{Set}$ есть четыре факторизационные системы [8, пример 14.2 (4)], причём первая компонента состоит из сюръекций только у двух: $(\mathrm{Iso}, \mathrm{Mor})$ и $(\mathrm{Epi}, \mathrm{Mono})$. Выбору класса $\mathrm{Iso}\,\mathbf{Set}$ всех биекций множеств в качестве L препятствует условие (ii) определения 19: класс всех биективных $\mathbf{Pos}$-морфизмов не образует первой компоненты никакой факторизационной системы в $\mathbf{Pos}$, поскольку он не замкнут относительно кодекартовых квадратов (вершина копредела $\mathbf{Pos}$-диаграммы $\{a < b\} \leftarrow \{a, b\} \to \{a > b\}$, состоящей из биекций $a \mapsto a$, $b \mapsto b$, является одноточечным множеством). Класс $\mathrm{Epi}\,\mathbf{Set}$ всех сюръективных отображений, напротив, удовлетворяет всем условиям определения 19: в качестве M можно выбрать класс всех регулярных $\mathbf{Pos}$-мономорфизмов (категория $\mathbf{Pos}$ обладает факторизационной системой $(\mathrm{Epi}, \mathrm{RegMono})$, причём область любого регулярного $\mathbf{Pos}$-мономорфизма с дискретно упорядоченной кообластью дискретно упорядочена) и, кроме того, класс конфигураций сценариев CPos замкнут относительно любых накачек.

Убедимся, что категория $\mathrm{Trl}\,\mathrm{Epi}\,\mathbf{Set}$ над технологией SS является двойственной по отношению к категории трансформаций сценариев r-Pos. Сначала выберем произвольное сюръективное отображение частично упорядоченных множеств $f\colon T \to S$, такое что условие $f(y) < f(y')$ эквивалентно $y < y'$ для любых $y, y' \in T$: все такие отображения образуют класс $\mathrm{Mor}\,\text{r-Pos}^{\mathrm{op}}$ (см. раздел 1). Ясно, что отображение f монотонно. Выберем произвольно отображения $m\colon X \to S \in \mathrm{RegMono}\,\mathbf{Pos}$ и $k\colon X \to T$, такие что $f\big(k(x)\big) = m(x)$ для любого $x \in X$, и проверим, что отображение k монотонно, откуда будет следовать, что

$f \in (\mathrm{RegMono}\,\mathbf{Pos})$-Init. Пусть произвольные $x, x' \in X$ удовлетворяют условию $x < x'$. Поскольку отображение m инъективно, $m(x) < m(x')$, следовательно $k(x) < k(x')$ ввиду выбора отображения f.

Обратно, пусть $t\colon P \to Q$ — произвольное сюръективное RegMono-инициальное отображение частично упорядоченных множеств. Выберем произвольно $p, p' \in P$, такие что $t(p) < t(p')$, и проверим, что $p < p'$, откуда будет следовать, что $t^{\mathrm{op}} \in \mathrm{Mor}\,\mathrm{r\text{-}Pos}$. Рассмотрим отображения $s\colon \mathit{2} \to P\colon 0 \mapsto p,\ 1 \mapsto p'$ и $n\colon \mathit{2} \to Q\colon x \mapsto t\big(s(x)\big)$, где через $\mathit{2}$ обозначено двухэлементное линейно упорядоченное множество $\{0 < 1\}$. Ясно, что $n \in \mathrm{RegMono}\,\mathbf{Pos}$, поэтому отображение s монотонно ввиду выбора отображения t, следовательно, $p < p'$.

Используя пример технологии событийного моделирования, легко показать, что ограничения, накладываемые на $\mathbf{SPEC}$-морфизм $\langle \mathrm{cm}, \mathrm{sm}\rangle\colon \mathrm{SC} \to \mathrm{SC}'$ в условии теоремы 4 в целях переноса способности своей области поддерживать некоторый класс разметок на кообласть, не могут быть существенно ослаблены. Будем исходить из того факта, что технология специфицирования сценариев SS не поддерживает $(\mathrm{Iso}\,\mathbf{Set})$-разметки. $\mathbf{SPEC}$-морфизм $\langle 1_{\mathbf{Pos}}, |{-}|\rangle\colon \mathrm{TS} \to \mathrm{SS}$, где $\mathrm{TS} = \mathbf{conf}^{*}\big(\mathbf{conf}(\mathrm{SS})\big)$, является $\mathbf{conf}$-трассой, но его вторая компонента не является изоморфизмом, и TS поддерживает $(\mathrm{Iso}\,\mathbf{Pos})$-разметки. В свою очередь, $\mathbf{SPEC}$-морфизм $\langle \mathrm{dord}, 1_{\mathbf{Set}}\rangle\colon \mathrm{US} \to \mathrm{SS}$, где $\mathrm{US} = \mathbf{conf}^{*}\big(\mathbf{conf}^{**}(\mathrm{SS})\big)$, не является $\mathbf{conf}$-трассой, и US поддерживает $(\mathrm{Iso}\,\mathbf{Set})$-разметки.

Следствие 4.3 позволяет выявить функториальную природу процедуры порождения технологий разметкой. Обозначим через $\mathbf{SPL}$ категорию, класс объектов которой состоит из всех пар $\langle \mathrm{SC}, \mathrm{L}\rangle$, где SC — технология специфицирования, поддерживающая L-разметки, а морфизмом пары $\langle \mathrm{SC}, \mathrm{L}\rangle$ в $\langle \mathrm{SC}', \mathrm{L}'\rangle$ служит любой $\mathbf{SPEC}$-морфизм $\langle \mathrm{cm}, \mathrm{sm}\rangle\colon \mathrm{SC} \to \mathrm{SC}'$, такой что cm — $\mathbf{desc}$-разметка, sm — изоморфизм категорий и $\mathrm{sm}(\mathrm{L}) \subseteq \mathrm{L}'$. Согласно следствию 4.3 существует вложение $\mathbf{ispl}\colon \mathbf{SPL} \hookrightarrow \mathbf{ARCH}\colon \langle \mathrm{SC}, \mathrm{L}\rangle \mapsto \mathrm{SC}_{\mathrm{L}}$.

Привлекая следствие 4.4, покажем, что процедура AO-расширения естественна относительно этого вложения. Будем для краткости обозначать технологию специфицирования $\mathbf{spec}\big(\mathrm{AO}_{\mathrm{int}}(\mathbf{AR})\big)$ через s-$\mathrm{AO}(\mathbf{AR})$. Пусть $\mathbf{SPLL}$ — подкатегория в $\mathbf{SPL}$, класс объектов которой совпадает с $\mathrm{Ob}\,\mathbf{SPL}$, а морфизмом объекта $\langle \mathrm{SC}, \mathrm{L}\rangle$ в $\langle \mathrm{SC}', \mathrm{L}'\rangle$ служит любой $\mathbf{SPL}$-морфизм $\langle \mathrm{cm}, \mathrm{sm}\rangle\colon \langle \mathrm{SC}, \mathrm{L}\rangle \to \langle \mathrm{SC}', \mathrm{L}'\rangle$, удовлетворяющий условию $\mathrm{sm}(\mathrm{L}) = \mathrm{L}'$ (о нетривиальности класса таких морфизмов свидетельствует теорема 4). Построим для произвольного $\mathbf{SPLL}$-морфизма $\langle \mathrm{cm}, \mathrm{sm}\rangle$ функтор $\mathrm{aom}\colon \mathrm{AO} \to \mathrm{AO}'$, индуцирующий $\mathbf{SPLL}$-морфизм

$$\langle \mathrm{aom}, \mathrm{sm}\rangle\colon \langle s\text{-}\mathrm{AO}(\mathrm{SC}_{\mathrm{L}}), \mathrm{L}\rangle \to \langle s\text{-}\mathrm{AO}(\mathrm{SC}'_{\mathrm{L}'}), \mathrm{L}'\rangle.$$

Начнём с установленного в разделе 2 факта, что категория AO-моделей AO, построенная из составляющих L-технологии SC_{L}, является вершиной предела $\mathbf{CAT}$-диаграммы со схемой $\boldsymbol{U}\,\mathrm{sig}\colon \mathrm{c\text{-}DESC} \to \mathrm{SIG} \leftarrow \mathrm{LAB} :\mathrm{dom} \circ \mathrm{il}$, где LAB — полная подкатегория в категории стрелок $\mathrm{SIG}^{\mathit{2}}$, класс объектов которой состоит из всех L-морфизмов. Легко проверить, что $\mathbf{SPLL}$-морфизм $\langle \mathrm{cm}, \mathrm{sm}\rangle$

индуцирует естественное преобразование таких диаграмм, которое мы обозначим через ξ, с компонентами $\mathrm{cm}\colon \text{c-DESC} \to \text{c-DESC}'$, $\mathrm{sm}\colon \mathrm{SIG} \to \mathrm{SIG}'$, $\mathrm{sml}\colon \mathrm{LAB} \to \mathrm{LAB}'$, где sml — функтор, действующий так же, как sm[2]. Отметим, что композиция **SPLL**-морфизмов индуцирует композицию таких естественных преобразований. Положим $\mathrm{aom} = \lim(\langle \xi, 1_{\boldsymbol{U}} \rangle)\colon \mathrm{AO} \to \mathrm{AO}'$ (функтор lim определяется по двойственности аналогично функтору colim, введённому в разделе 1). По определению предела имеем $\mathrm{mod}' \circ \mathrm{aom} = \mathrm{cm} \circ \mathrm{mod}$ (откуда вытекает, что $\mathrm{int}' \circ \mathrm{aom} = \mathrm{sig}' \circ \mathrm{mod}' \circ \mathrm{aom} = \mathrm{sig}' \circ \mathrm{cm} \circ \mathrm{mod} = \mathrm{sm} \circ \mathrm{sig} \circ \mathrm{mod} = \mathrm{sm} \circ \mathrm{int}$) и $\mathrm{asp}' \circ \mathrm{aom} = \mathrm{sml} \circ \mathrm{asp}$ (откуда вытекает, что $\mathrm{str}' \circ \mathrm{aom} = \mathrm{codom}' \circ \mathrm{il}' \circ \mathrm{asp}' \circ \mathrm{aom} = \mathrm{codom}' \circ \mathrm{il}' \circ \mathrm{sml} \circ \mathrm{asp} = \mathrm{sm} \circ \mathrm{codom} \circ \mathrm{il} \circ \mathrm{asp} = \mathrm{sm} \circ \mathrm{str}$). Следовательно, имеем

$$\begin{aligned}\mathrm{aom}\colon & \mathrm{AO} \to \mathrm{AO}' \\ & \colon \langle A, l \rangle \mapsto \langle \mathrm{cm}(A), \mathrm{sm}(l) \rangle,\ \langle f, b \rangle \mapsto \langle \mathrm{cm}(f), \mathrm{sm}(b) \rangle.\end{aligned}$$

Функтор aom унивалентен, поскольку правая часть соотношения $\mathrm{mod}' \circ \mathrm{aom} = \mathrm{cm} \circ \mathrm{mod}$ состоит из унивалентных функторов. Кроме того, функтор aom обладает левым сопряжённым, который имеет вид

$$\begin{aligned}\mathrm{aom}^*\colon & \mathrm{AO}' \to \mathrm{AO} \\ & \colon \langle A', l' \rangle \mapsto \langle \mathrm{cm}^*(A'), \mathrm{sm}^{-1}(l') \rangle,\ \langle f', b' \rangle \mapsto \langle \mathrm{cm}^*(f'), \mathrm{sm}^{-1}(b') \rangle,\end{aligned}$$

единица этого сопряжения тождественна. Функтор aom сохраняет копределы всех конфигураций технологии $s\text{-AO}(\mathrm{SC_L})$, поскольку функторы $\mathrm{cm} \circ \mathrm{mod}$, int и str сохраняют копределы любой конфигурации, а функтор sm, будучи изоморфизмом категорий, сохраняет копределы любой диаграммы. Проверим, что функтор $\mathbf{D}\mathrm{aom}$ переводит конфигурации технологии $s\text{-AO}(\mathrm{SC_L})$ в конфигурации технологии $s\text{-AO}(\mathrm{SC}'_{\mathrm{L}'})$, т. е. что $\mathrm{sm} \circ \mathrm{AOInt_{int}} \subseteq \mathrm{AOInt_{int'}}$. Выберем произвольно диаграммы $\Theta \in \mathrm{AOInt_{int}}$ и $\Delta \in \mathbf{D}\mathrm{int}'^{-1}(\{\mathrm{sm} \circ \Theta\})$, положим $\Sigma = \mathrm{aom}^* \circ \Delta$. Имеем $\mathrm{aom} \circ \Sigma = \Delta$ и $\mathrm{int} \circ \Sigma = \mathrm{sm}^{-1} \circ \mathrm{sm} \circ \mathrm{int} \circ \Sigma = \mathrm{sm}^{-1} \circ \mathrm{int}' \circ \Delta = \Theta$, так что $\mathrm{mod}' \circ \Delta = \mathrm{mod}' \circ \mathrm{aom} \circ \Sigma = \mathrm{cm} \circ \mathrm{mod} \circ \Sigma \in \mathrm{Conf}'$ и, кроме того, диаграмма $\mathrm{str}' \circ \Delta = \mathrm{str}' \circ \mathrm{aom} \circ \Sigma = \mathrm{sm} \circ \mathrm{str} \circ \Sigma$ обладает копределом. Отсюда так же, как в доказательстве следствия 4.6, выводится, что диаграмма Δ аспектно детерминирована. Поэтому аспектно детерминированной является и любая её накачка $\text{tr-AO}'$-морфизмами, так что $\mathrm{sm} \circ \Theta \in \mathrm{AOInt_{int'}}$.

Таким образом, пара $\langle \mathrm{aom}, \mathrm{sm} \rangle$ действительно представляет собой **SPLL**-морфизм АО-технологий, так что ввиду следствия 4.3 существует функтор

$$\begin{aligned}\mathbf{aom}\colon & \mathbf{SPLL} \to \mathbf{ARCH} \\ & \colon \langle \mathrm{SC}, \mathrm{L} \rangle \mapsto \mathrm{AO_{int}}(\mathrm{SC_L}),\ \langle \mathrm{cm}, \mathrm{sm} \rangle \mapsto \langle \mathrm{aom}, \mathrm{sm}, \mathrm{aom}(-^{\mathrm{op}})^{\mathrm{op}} \rangle.\end{aligned}$$

Существует естественное преобразование этого функтора в стандартное вложение $\mathbf{ispl} \circ \mathbf{ispll}\colon \mathbf{SPLL} \hookrightarrow \mathbf{ARCH}$, где $\mathbf{ispll}\colon \mathbf{SPLL} \hookrightarrow \mathbf{SPL}$ — вложение подкатегории. Это естественное преобразование ставит в соответствие каждому **SPLL**-объекту $\langle \mathrm{SC}, \mathrm{L} \rangle$ **ARCH**-морфизм $\mathrm{ao}_{\mathrm{int}} \colon \mathrm{AO}_{\mathrm{int}}(\mathrm{SC}_{\mathrm{L}}) \to \mathrm{SC}_{\mathrm{L}}$, построенный в доказательстве следствия 3.2. Это иллюстрирует естественность процедуры АО-расширения технологий, порождённых разметкой.

Заключение

Технологии программной инженерии типа MDE и АОП обладают большим потенциалом в качестве средств снижения затрат на проектирование программных систем. На практике проверено, что они позволяют заменять десятки программистов-кодировщиков небольшими группами, создающими инструменты для автоматического порождения массивного программного кода [5]. В качестве математического аппарата для строгого анализа и совершенствования процессов проектирования, особенно в условиях совместного применения разнородных моделей, целесообразно привлечь теорию категорий. Категорные конструкции отражают ключевые особенности технологических процедур проектирования программных систем:

— выделение интеграционного интерфейса из формальной модели программы описывается функтором специального вида — унивалентным корефлектором;
— конфигурации систем («мегамодели») находятся среди диаграмм, копределы которых детерминируются подходящим унивалентным корефлектором;
— трансформации, предоставляющие оптимальные возможности для трассирования, отбираются некоторым регулярным образом из морфизмов, двойственных к таким, которые переходят в ретракции под действием унивалентного корефлектора.

Теоретические результаты такого рода находят применение в преодолении проблем, трудоёмкость решения которых «подручными средствами» возрастает с ростом масштаба и сложности программных систем. Такие проблемы возникают при согласовании моделей разных частей и аспектов системы друг с другом и с текстами программ (в том числе путём трассирования), при достижении требуемых значений интегральной производительности системы и других общесистемных показателей качества и т. д. Строятся теоретико-категорные конструкции, формально описывающие возможные способы решения такой проблемы на абстрактном концептуальном уровне: компонентам, задействованным в решении, ставятся в соответствие объекты подходящих категорий, технологическим операциям — морфизмы, переходам — функторы и т. д. Путём вычислений в категориях оцениваются свойства способов решения проблемы, из альтернативных способов выбирается доставляющий экстремальное значение целевому функционалу (стоимости, надёжности и т. д.). Конструкция, соответствующая выбранному способу, интерпретируется в понятиях подходящей технологии, и выбираются или создаются нужные инструменты для его автоматизации. Примеры применения такого подхода приведены в [5].

Развивая такой теоретико-категорный подход, в перспективе можно будет значительно повысить уровень интеллектуальности инструментов программной инженерии — переложить на них построение, анализ и оптимизацию процедур проектирования. Современные CASE-средства практически не умеют этого делать: они способны только механически выполнять команды инженеров. Если использовать язык теории категорий для формального внутримашинного представления технологий и приёмов проектирования, то можно свести интеллектуальную деятельность к распознаванию и расчёту категорных конструкций. Отметим, что такой подход мог бы привести к существенному повышению эффективности и в автоматизированном проектировании изделий материального производства.

Одним из первых шагов в указанном направлении можно считать реализацию автоматического решателя категорных задач разработки алгебраических спецификаций в рамках технологии SPECWARE [36]. Создание полноценных мультимодельных средств интеллектуального автоматизированного проектирования систем открывает множество перспективных направлений дальнейших исследований.

Литература

[1] Васильев С. Н. Формализация знаний и управление на основе позитивно-образованных языков // Информ. технол. и вычисл. сист. — 2008. — № 1. — С. 3—17.

[2] ГОСТ Р ИСО 9126-93. Информационная технология. Оценка программной продукции. Характеристики качества и руководства по их применению. — М.: Госстандарт России, 1994.

[3] Ковалёв С. П. Формальный подход к аспектно-ориентированному моделированию сценариев // Сиб. журн. индустр. мат. — 2010. — Т. 13, № 3. — С. 30—42.

[4] Ковалёв С. П. Диаграммное описание комплексирования программных систем // Вестн. Новосибирск. гос. ун-та. Сер.: Математика, механика, информатика. — 2012. — Т. 12, № 3. — С. 103—126.

[5] Ковалёв С. П. Повышение эффективности процессов проектирования больших информационно-управляющих систем // Тр. XII Всероссийск. совещ. по проблемам управления ВСПУ-2014. — М.: ИПУ РАН, 2014. — С. 9291—9300.

[6] Маклейн С. Категории для работающего математика. — М.: Физматлит, 2004.

[7] Соммервилл И. Инженерия программного обеспечения. — М.: Вильямс, 2002.

[8] Adámek J., Herrlich H., Strecker G. Abstract and Concrete Categories. — New York: Wiley and Sons, 1990.

[9] Aizenbud-Reshef N., Nolan B., Rubin J., Shaham-Gafni Y. Model traceability // IBM Systems J. — 2006. — Vol. 45, no. 3. — P. 515—526.

[10] Allen R. J., Garlan D. A formal basis for architectural connection // ACM Trans. Software Engineering Methodology. — 1997. — Vol. 6, no. 3. — P. 213—249.

[11] Brichau J., Chitchyan R., Rashid A., D'Hondt T. Aspect-oriented software development: an introduction // Wiley Encyclopedia of Computer Science and Engineering. Vol. 1. — New York: Wiley and Sons, 2008. — P. 188—198.

[12] Brunette C., Talpin J.-P., Gamatié A., Gautier T. A metamodel for the design of polychronous systems // J. Logic Algebraic Programming. — 2009. — Vol. 78, no. 4. — P. 233—259.

[13] Carboni A., Janelidze G., Kelly G. M., Paré R. On localization and stabilization for factorization systems // Appl. Categorical Structures. — 1997. — Vol. 5. — P. 1—58.

[14] Diskin Z., Maibaum T. S. E. Category theory and model-driven engineering: from formal semantics to design patterns and beyond // Proc. 7th Workshop ACCAT'2012. Electronic Proc. Theor. Comput. Sci. — 2012. — Vol. 93. — P. 1—21.

[15] Douence R., Fradet P., Südholt M. Trace-based aspects // Aspect-Oriented Software Development. — Reading: Addison-Wesley, 2004. — P. 201—218.

[16] Eclipse Modeling Project. — `http://www.eclipse.org/modeling/`. — The Eclipse Foundation, 2014.

[17] Egyed A., Grünbacher P., Heindl M., Biffl S. Value-based requirements traceability: lessons learned // Design Requirements Engineering: A Ten-Year Perspective. Design Requirements Workshop, Cleveland, OH, USA, June 3—6, 2007, Revised and Invited Papers. — Berlin: Springer, 2009. — (Lect. Notes Business Inform. Processing; Vol. 14). — P. 240—257.

[18] Fiadeiro J. L., Lopes A., Wermelinger M. A mathematical semantics for architectural connectors // Generic Programming. — Berlin: Springer, 2003. — (Lect. Notes Comput. Sci.; Vol. 2793). — P. 190—234.

[19] Goguen J. Categorical foundations for general systems theory // Advances in Cybernetics and Systems Research. — London: Transcripta Books, 1973. — P. 121—130.

[20] Goguen J. A categorical manifesto // Math. Struct. Comput. Sci. — 1991. — Vol. 1, no. 1. — P. 49—67.

[21] Gotel O., Finkelstein A. An analysis of the requirements traceability problem // Proc. 1st Int. Conf. on Requirements Engineering. — Colorado Springs, 1994. — P. 94—101.

[22] Graaf B., van Deursen A. Visualisation of domain-specific modelling languages using UML // Proc. 14th IEEE Int. Conf. on the Engineering of Computer-Based Systems ECBS'2007. — Tucson, 2007. — P. 586—595.

[23] Groher I., Völter M. Aspect-oriented model-driven software product line engineering // Transactions on Aspect-Oriented Software Development VI. — Berlin: Springer, 2009. — (Lect. Notes Comput. Sci.; Vol. 5560). — P. 111—152.

[24] Guitart R., van den Bril L. Décompositions et lax-complétions // Cahiers de Topologie et Géométrie Différentielle Catégoriques. — 1977. — Vol. 18, no. 4. — P. 333—407.

[25] Hanenberg S., Unland R. Roles and aspects: similarities, differences, and synergetic potential // Object-Oriented Information Systems. 8th Int. Conf., OOIS 2002 Montpellier, France, September 2—5, 2002. Proceedings. — Berlin: Springer, 2002. — (Lect. Notes Comput. Sci.; Vol. 2425). — P. 507—520.

[26] Jouault F., Vanhooff B., Bruneliere H., Doux G., Berbers Y., Bezivin J. Inter-DSL coordination support by combining megamodeling and model weaving // Proc. 2010 ACM Symp. on Appl. Computing. — Sierre, 2010. — P. 2011—2018.

[27] Kannenberg A., Saiedian H. Why software requirements traceability remains a challenge // J. Defense Software Engineering. — July/August 2009. — P. 14—19.

[28] Kiczales G., Lamping J., Mendhekar A., Maeda C., Lopes C. V., Loingtier J.-M., Irwin J. Aspect-oriented programming // ECOOP '97 — Object-Oriented Programming:

11th Europ. Conf., Jyväskylä, Finland, June 9—13, 1997, Proceedings, Vol. 11. — Berlin: Springer, 1997. — (Lect. Notes Comput. Sci.; Vol. 1241). — P. 220—242.

[29] Kolovos D. S., Paige R. F., Polack F. A. C. The grand challenge of scalability for model driven engineering // Models in Software Engineering: Workshops and Symposia at MODELS 2008, Toulouse, France, September 28 — October 3, 2008. Reports and Revised Selected Papers. — Berlin: Springer, 2009. — (Lect. Notes Comput. Sci.; Vol. 5421). — P. 48—53.

[30] Kovalyov S. P. Modeling aspects by category theory // Proc. 9th Workshop on Foundations of Aspect-Oriented Languages. — Rennes, France, 2010. — P. 63—68.

[31] Lopes A., Fiadeiro J. L. Revisiting the categorical approach to systems // Algebraic Methodology and Software Technology: 9th Int. Conf., AMAST 2002, Saint-Gilles-les-Bains, Reunion Island, France, September 9—13, 2002. Proceedings. — Berlin: Springer, 2002. — (Lect. Notes Comput. Sci.; Vol. 2422). — P. 426—440.

[32] Pinto M., Fuentes L., Troya J. M. DAOP-ADL: an architecture description language for dynamic component and aspect-based development // Generative Programming and Component Engineering: Second Int. Conf., GPCE 2003, Erfurt, Germany, September 22—25, 2003, Proceedings, Vol. 2. — Berlin: Springer, 2003. — (Lect. Notes Comput. Sci.; Vol. 2830). — P. 118—137.

[33] Pratt V. R. Modeling concurrency with partial orders // Int. J. Parallel Programming. — 1986. — Vol. 15, no. 1. — P. 33—71.

[34] Rutle A., Rossini A., Lamo Y., Wolter U. A formalisation of the copy-modify-merge approach to version control in MDE // J. Logic Algebraic Programming. — 2010. — Vol. 79, no. 7. — P. 636—658.

[35] Schmidt D. C. Model-driven engineering // IEEE Computer. — 2006. — Vol. 39, no. 2. — P. 25—32.

[36] Srinivas Y. V., Jüllig R. SPECWARE: formal support for composing software // Mathematics of Program Construction '95. — Berlin: Springer, 1995. — (Lect. Notes Comput. Sci.; Vol. 947). — P. 399—422.

[37] Steimann F. The paradoxical success of aspect-oriented programming // Proc. Int. Conf. OOPSLA'06. — Portland, 2006. — P. 481—497.