

Научная статья

УДК 004.5; DOI: 10.61260/2218-130X-2025-4-156-165

МЕТОД СТРУКТУРНО-ПАРАМЕТРИЧЕСКОГО СИНТЕЗА ИНТЕРФЕЙСА ИНФОРМАЦИОННОГО СЕРВИСА ЗАПРОСНОГО ТИПА

✉ Курта Павел Андреевич.

Санкт-Петербургский университет ГПС МЧС России, Санкт-Петербург, Россия

✉ expert@kurt.a.ru

Аннотация. Изучена проблема шаблонности существующих интерфейсов информационных систем при необходимости их персонализации под конкретного пользователя и решаемые им задачи. Рассматривается ограниченное множество информационных систем, относящихся к запросному типу и предназначенных для сбора данных от пользователя, на основании которых производятся необходимые вычисления. Ставится цель, повышения эффективности интерфейсов, под которой понимается совокупность следующих показателей: безошибочность работы с графическими элементами (результативность), скорость работы (оперативность) и мера отсутствия психоэмоциональной нагрузки (ресурсоэкономность). Задача сводится к оптимизационной, целевой функцией в которой является указанная эффективность, а варьируемыми параметрами – структура интерфейса и параметры его элементов. Также в модели присутствуют статические параметры, такие, как атомарные эффективности каждого из элементов и др. Предлагается метод структурно-параметрического синтеза интерфейса, состоящий из следующих последовательных шагов, выполняемых с разветвлениями в логике: вычисление атомарной эффективности, выбор решаемой в информационной системе задачи, получение логики работы с информационной системой, определение типов данных, выбор режима настройки, проверка режима настройки, сбор данных о пользователе, выбор профиля пользователя, ввод критериев оптимизации, ввод атомарных эффективностей, поиск синтезированного ранее интерфейса, проверка наличия синтезированного интерфейса, построение эффективностной модели интерфейса, выбор алгоритма решения оптимизационной задачи, решение оптимизационной задачи, динамическая генерация интерфейса, подключение интерфейса к информационной системе, инициализация сессии пользователя, мониторинг действий пользователя, адаптивная корректировка интерфейса, вывод результата решения задачи, оценка реальной эффективности, проверка соответствия эффективности заданной, корректировка атомарных эффективностей, корректировка параметров модели, сохранение синтезированного интерфейса. Приводится блок-схема метода, а также указываются значимость и пути продолжения исследования.

Ключевые слова: информационная система, интерфейс, структурно-параметрический синтез, искусственный интеллект, модель, оптимизация, метод, схема, эффективность

Для цитирования: Курта П.А. Метод структурно-параметрического синтеза интерфейса информационного сервиса запросного типа // Научно-аналитический журнал «Вестник Санкт-Петербургского университета Государственной противопожарной службы МЧС России». 2025. № 4. С. 156–165. DOI: 10.61260/2218-130X-2025-4-156-165.

Scientific article

METHOD OF STRUCTURAL-PARAMETRIC SYNTHESIS OF THE QUERY-TYPE INFORMATION SERVICE INTERFACE

✉ Kurta Pavel A.

Saint-Petersburg university of State fire service of EMERCOM of Russia, Saint-Petersburg, Russia

✉ expert@kurta.ru

Abstract. This paper addresses the problem of the generic nature of existing information system interfaces, which requires personalization for specific users and their tasks. A limited set of query-based information systems designed to collect user data, which is then used to perform calculations, is considered. The goal is to improve interface efficiency, defined as a combination of the following metrics: error-free operation of graphical elements (effectiveness), speed of operation (efficiency), and the degree of absence of psycho-emotional stress (resource efficiency). The problem is reduced to an optimization one, in which the target function is the stated efficiency, and the variable parameters are the interface structure and the parameters of its elements. The model also includes static parameters such as atomic efficiencies of each element, etc. A method for structural-parametric interface synthesis is proposed, consisting of the following sequential steps performed with logical branches: calculating atomic efficiency, selecting the task to be solved in the information system, obtaining the logic for working with the information system, determining data types, selecting the setup mode, checking the setup mode, collecting user data, selecting a user profile, entering optimization criteria, entering atomic efficiencies, searching for a previously synthesized interface, checking for the presence of a synthesized interface, building an efficiency model of the interface, selecting an algorithm for solving the optimization problem, solving the optimization problem, dynamically generating the interface, connecting the interface to the information system, initializing the user session, monitoring user actions, adaptive adjustment of the interface, displaying the result of solving the problem, assessing the actual efficiency, checking whether the efficiency corresponds to the specified one, adjusting atomic efficiencies, adjusting the model parameters, and saving the synthesized interface. A block diagram of the method is provided, and the significance and ways of continuing the research are indicated.

Keywords: information system, interface, structural-parametric synthesis, artificial intelligence, model, optimization, method, scheme, efficiency

For citation: Kurta P.A. Method of structural-parametric synthesis of the query-type information service interface // Scientific and analytical journal «Vestnik Saint-Petersburg university of State fire service of EMERCOM of Russia». 2025. № 4. Р. 156–165. DOI: 10.61260/2218-130X-2025-4-156-165.

Введение

Информационные технологии стали неотъемлемой частью современного общества и повседневным помощником для его представителей. Связывающим же звеном между технологиями и социумом являются интерфейсы, трансформирующие соответствующим образом передаваемую информацию между машиноориентированной и человекоориентированной формами. При этом успешность такого взаимодействия зависит в том числе и от эффективности используемых для этого интерфейсов [1]. Однако в предметной области существует следующее противоречие, задаваемое противопоставлением потребностей и возможностей. С одной стороны, исходя из особенностей любого человека, ему будет удобнее работать с адаптированным под него интерфейсом [2], притом учитывающим решаемую задачу. Например, пожилым пользователям нежелательно представление форм, перегруженных графическими элементами. Молодое же поколение способно успешно оперировать огромным количеством полей ввода. С другой стороны, современные интерфейсы хотя и строятся с учетом определенной эргономики, принятых шаблонов и богатого практического опыта, однако они недостаточно учитывают предпочтения

отдельных пользователей [3]. Например, отсутствуют полноценные решения по динамическому изменению интерфейса в зависимости от возрастной группы, социального положения или других характеристик субъекта. Также необходимо отметить и общее разнообразие интерфейсов киберфизических систем [4], их собственную небезопасность [5].

Ранее автором было предложено применение для отображения интерфейса структурно-параметрического синтеза, суть которого заключается в создании графических форм и расположении элементов на них (структура системы), а также выборе типа элементов (параметры системы); при этом учитываются особенности конкретного пользователя в момент его работы с конкретными данными. Целью такой генерации являлось повышение эффективности самого интерфейса, сводя данную задачу к многокритериальной оптимизационной (поскольку эффективность представляет собой совокупность показателей, имеющих взаимно обратную связь). Текущим же этапом исследования будет создание самого метода синтеза таких интерфейсов, описание которого приводится далее. Также под системой, обладающей таким интерфейсом, рассматривается один из ее классов, а именно – информационный сервис запросного типа (ИСЗТ), предназначенный для получения данных от пользователя, осуществления на их основании вычислений и выдачи информационного результата.

Оптимизационная задача

Задача структурно-параметрического синтеза была сведена к оптимизационной, целевой функцией которой является эффективность интерфейса, под которым, с точки зрения автора, понимается совокупность следующих показателей: результативность – отсутствие ошибок в процессе взаимодействия; оперативность – итоговая скорость взаимодействия; ресурсоэкономность – мера отсутствия психоэмоциональной нагрузки (ПЭН) пользователя. Эффективность в формальном виде представима как:

$$\text{Efficiency} \equiv \langle \text{Potency}, \text{Operativeness}, \text{ResourceSaving} \rangle,$$

где *Potency* – результативность; *Operativeness* – оперативность; *ResourceSaving* – ресурсоэкономность.

Каждый из показателей эффективности вычисляется на основании некоторой модели интерфейса, описывающей (согласно логике взаимодействия с ИСЗТ) существующие формы, а также присутствующие на них в определенном порядке элементы [6]. Сложность решения оптимизационной задачи заключается в том, что как «разбивка» элементов по формам, так и порядок элементов и их типы оказывают различное влияние на каждый из показателей эффективности. Например, с одной стороны, ввод большого числового значения с помощью текстового поля будет более быстрым, чем использованием ползунка – оперативность элемента 1-го типа выше. Однако при посимвольном вводе числа шанс допустить ошибку будет выше (особенно если числа идут непоследовательно), чем выбор гарантированно верных значений из списка – результативность элемента 1-го типа ниже. Ресурсоэкономность же (как ПЭН) хотя и будет близкой для элементов обоих типов, однако также не тождественной. Аналогичным образом на все показатели эффективности будет оказывать влияние и количество элементов на форме – оперативность будет снижаться (из-за необходимости нажимать кнопку перехода к следующей форме), а ресурсоэкономность – увеличиваться (так как пользователю нет необходимости в анализе перегруженной формы); зависимость результативности также будет нетривиальной.

Следовательно, оптимизационная задача может быть поставлена следующим образом (упрощенно):

$$\left\{ \begin{array}{l} \exists \{Form_i\} \in Interface \\ \exists \{Element_{i,j}\}, Element_{i,j} \in Form_i : \text{Efficiency} \rightarrow max, \end{array} \right.$$

где $Interface$ – интерфейс для работы с ИСЗТ; $Form_i$ – i -ая форма интерфейса; $Element_{i,j}$ – j -ый элемент, принадлежащий i -й форме (то есть имеющий индекс « i,j »); \max – указание максимизации (в данном случае общей эффективности). Постановку задачи можно проинтерпретировать как поиск такой совокупности форм, их элементов и их типов, при которой итоговая эффективность будет максимальна.

Также при расчете целевой функции необходимо учитывать и так называемую атомарную эффективность каждого элемента интерфейса (так, если бы он был единственным на форме), зависящую не только от его типа, но и вводимых данных. Конкретные характеристики пользователя (например, возрастная группа) также будут оказывать влияние на значение атомарных показателей эффективности элемента, что можно записать как:

$$\left\{ \begin{array}{l} Efficiency_{i,j} = Atomic(Element_{i,j}^{Type}, Data_{i,j}.User), \\ Element_{i,j}^{Type} \in \{Type_k\} \end{array} \right.$$

где $Efficiency_{i,j}$ – атомарная эффективность элемента; $Atomic(\dots)$ – функция вычисления атомарной эффективности элемента; $Element_{i,j}^{Type}$ – тип соответствующего элемента; $Type_k$ – k -ый тип элемента из множества всех типов; $Data_{i,j}$ – данные, с которыми оперирует элемент с индексами « i,j »; $User$ – пользователь интерфейса (его характеристики или модель [7]). Запись можно проинтерпретировать как зависимость атомарной эффективности каждого элемента как от самих элементов и данных, так и от работающего с ними пользователя.

Есть некоторые жестко заданные правила проектирования интерфейсов, которые, хотя и не будут считаться с точки зрения модели эффективными, однако отступление от них не допускается. Такие ограничения также требуется учесть в целевой функции при решении.

Метод синтеза интерфейса

Решение указанной оптимизационной задачи лишь в ограниченных случаях достижимо полным перебором из-за эффекта комбинаторного взрыва [8] – экспоненциального роста количества возможных решений (то есть конкретных вариаций интерфейса) при линейном увеличении параметров (то есть количества элементов). Если предполагается построение интерфейсов в режиме реального времени, то время самого процесса решения становится решающим. В этом случае классическим способом может считаться применение эвристических алгоритмов [9], которые хотя и не гарантируют оптимального решения, но получаемые результаты считаются удовлетворительными; например, популяционные, локального поиска, на базе машинного обучения с подкреплением и др. Однако простого применения выбранного алгоритма будет недостаточно, поскольку он является более абстрактным механизмом, работающим с данными и функциями, составление которых является отдельными подзадачами. Далее будет предложен «метод структурно-параметрического синтеза интерфейса информационного сервиса запросного типа» (Метод), представляющий собой полноценное средство решения оптимизационной задачи предметной области.

Область применения Метода может быть задана в качестве элемента адаптации или персонализации интерфейса ИСЗТ к пользователю. Соответственно, сам сценарий его использования является следующим: пользователь начал работу с ИСЗТ; затем он произвел выбор решаемой задачи, ввел свои персонализированные данные; произошло выполнение динамической части Метода; отобразился интерфейс, наиболее подходящий для вводимых данных и адаптированный к конкретному пользователю. Деление Метода на динамическую и статическую части целесообразно, поскольку часть его действий, занимающих особо продолжительное время, могут быть выполнены единожды вначале. В Метод целесообразно ввести ручную настройку режима его работы, автоматическую корректировку интерфейса в процессе работы, а также базу уже сформированных интерфейсов (то есть решений оптимизационной задачи).

Исходя из указанных предпосылок, была создана блок-схема Метода, приведенная на рисунке; серым фоном обозначаются служебные элементы, синим – внешние процедуры, зеленым – ввод и вывод данных, желтым – условное выполнение.

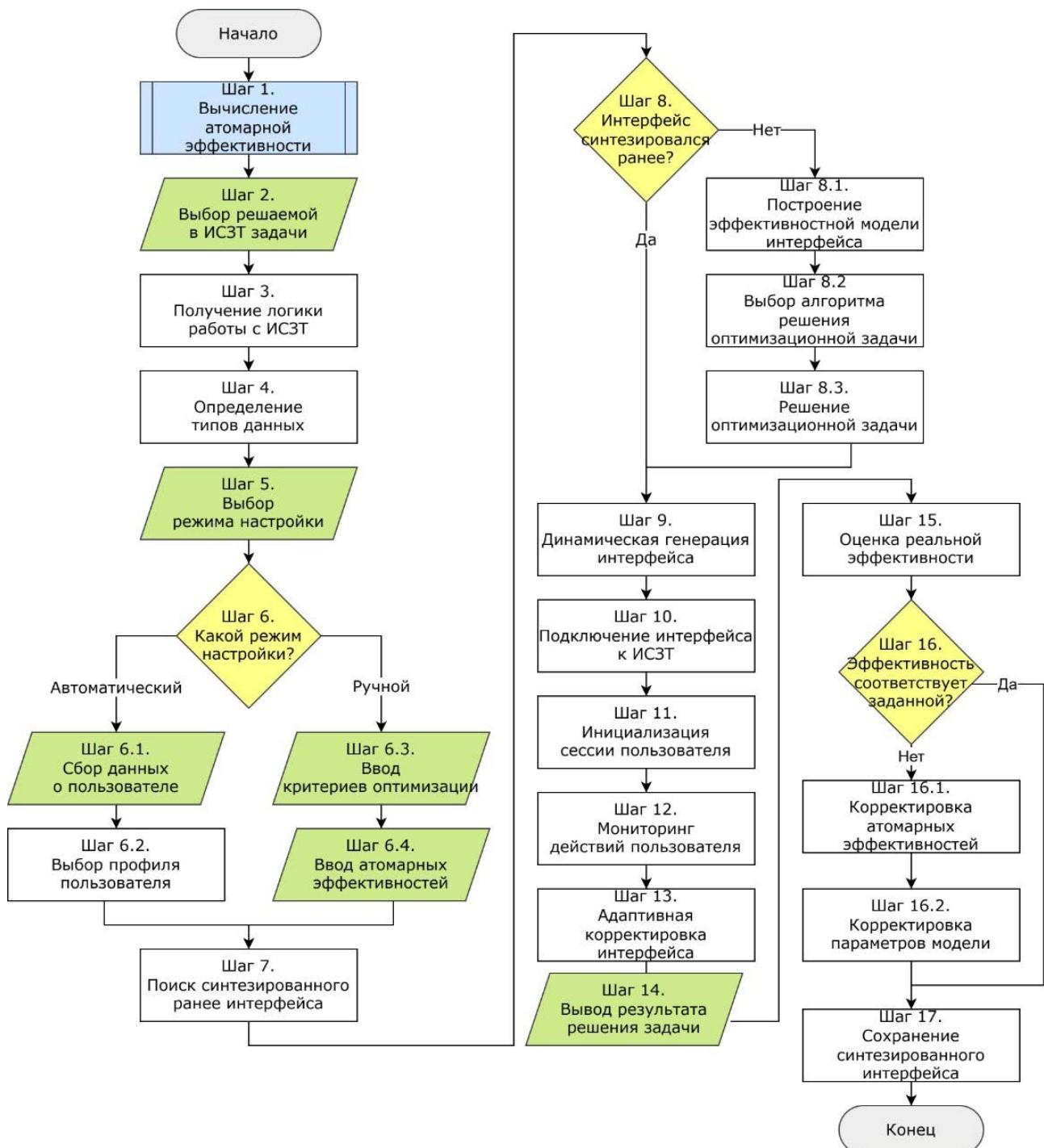


Рис. Схема метода структурно-параметрического синтеза интерфейса ИСЗТ

Далее дано краткое описание шагов Метода, представленных на Рисунке; «Начало» и «Конец» определяют точки входа и выхода в процесс его выполнения.

Шаг 1. Вычисление атомарной эффективности.

Для каждого типа графического элемента интерфейса (текстовое поле, выпадающий список, радиокнопка и т.д.) рассчитываются показатели эффективности. Вычисления основываются на статистических данных, эргономике и других моделях, учитывающих общие закономерности человека-машинного взаимодействия. Соответствующие измерения,

их результаты и необходимая для этого система была описана в предыдущей авторской статье [10]. Шаг относится к статической части Метода, поскольку имеет высокую длительность (то есть производится сбор статистических данных от пользователей различных групп) и выполняется один раз.

Шаг 2. Выбор решаемой в ИСЗТ задачи.

Пользователь производит выбор конкретной задачи, необходимой для решения с помощью данной ИСЗТ. Данный выбор напрямую влияет на критерии оптимизации, например, для финансовых операций важна точность, для получения ознакомительной информации – скорость, для изучения большого массива данных – низкая ПЭН.

Шаг 3. Получение логики работы с ИСЗТ.

Из ИСЗТ извлекается формализованное описание логики решения выбранной задачи. Таким образом определяется некоторый шаблон (или прообраз) будущего интерфейса и набор его элементов.

Шаг 4. Определение типов данных.

Аналогичным с Шагом 3 образом извлекаются данные, с которыми необходимо будет работать пользователю (включая диапазоны значений), а также их связь с соответствующими элементами. Данная информация необходима для выбора варьируемых параметров оптимизационной задачи.

Шаг 5. Выбор режима настройки.

Пользователю предоставляется выбор режима настройки интерфейса – автоматический, когда алгоритмически или статистически определяются необходимые параметры решения оптимизационной задачи при генерации интерфейса, или ручной – когда пользователь их задает сам (например, из списка имеющихся шаблонов).

Шаг 6. Какой режим настройки?

На основании выбранного режима настройки выполняются следующие шаги: 6.1 и 6.2 для автоматического, 6.3 и 6.4 для ручного.

Шаг 6.1. Сбор данных о пользователе.

В случае автоматического режима настройки о целевом пользователе собираются возрастные, профессиональные, поведенческие и другие характеристики, например, через опрос или анализ истории работы [11].

Шаг 6.2. Выбор профиля пользователя.

На основании характеристик пользователя выбирается профиль с заранее заданными параметрами модели – критерии оптимизации, атомарные эффективности элементов и т.п.

Шаг 6.3. Ввод критериев оптимизации.

В случае ручного режима настройки пользователь сам указывает критерии оптимизации для построения интерфейса, например, пожилой человек может задать минимальную ПЭН при средней вероятности ошибок и скорости работы.

Шаг 6.4. Ввод атомарных эффективностей.

Аналогично Шагу 6.3, пользователь вручную указывает атомарные эффективности, например, если он традиционно испытывает сложности при работе с выпадающими списками. Данный шаг, как правило, выполняется только подготовленными пользователями.

Шаг 7. Поиск синтезированного ранее интерфейса.

Исходя из того, что многократное решение одной и той же оптимизационной задачи (то есть для одинаковых варьируемых параметров, критериев эффективности и т.п.) было бы излишне, то целесообразно хранить базу уже полученных решений. Поиск такого существующего решения – то есть интерфейса – и производится на данном шаге.

Шаг 8. Интерфейс синтезировался ранее?

Осуществляется проверка результатов поиска уже синтезированного ранее интерфейса (на Шаге 7). В случае наличия такового, информация о нем используется для динамической генерации на Шаге 9. В ином случае требуется первоначальное создание интерфейса – на Шагах 8.1–8.3.

Шаг 8.1. Построение эффективностной модели интерфейса.

В случае отсутствия созданного ранее интерфейса требуется его генерация, для чего создается соответствующая эффективностная модель, которая позволяет вычислять целевую функцию (как совокупность показателей эффективности).

Шаг 8.2. Выбор алгоритма решения оптимизационной задачи.

На основании сложности модели (количество элементов, типы данных, программно-технические и иные ограничения) выбирается алгоритм решения оптимизационной задачи – от точного и с низкой оперативностью полного перебора до приблизительных и более скоростных эвристических.

Шаг 8.3. Решение оптимизационной задачи.

Выбранный на Шаге 8.2 алгоритм с использованием построенной на Шаге 8.1 модели и на основании заданных на Шагах 6.1–6.4 пользовательских данных и ограничений решает оптимизационную задачу, в результате которой определяется структура интерфейса (формы, элементы, порядок) и его параметры (типы данных).

Шаг 9. Динамическая генерация интерфейса.

По решению оптимизационной задачи в режиме реального времени производится построение интерфейса [12], персонализированное под текущего пользователя.

Шаг 10. Подключение интерфейса к ИСЗТ.

Сгенерированный интерфейс интегрируется с бэкендом ИСЗТ [13] – настраиваются обработчики событий для отправки и получения данных, производится подключение к API, реализуется обработка состояний загрузки и ошибок, обеспечивается маршрутизация между формами и др.

Шаг 11. Инициализация сессии пользователя.

Осуществляется подготовка интерфейса к работе – загружаются начальные данные (например, значения по умолчанию), устанавливаются сессионные параметры, создается инструкция для пользователя [14]. Затем происходит отображение первой формы и приглашение пользователя к работе с ИСЗИ через созданный интерфейс.

Шаг 12. Мониторинг действий пользователя.

В процессе работы системы в фоновом режиме собирает телеметрию о пользовательской активности – время работы с элементами и формами, количество допущенных ошибок и их классы [15], перемещение внимания пользователя, получение справок и др. Данная информация будет использоваться для общей корректировки параметров эффективностной модели при будущей работе с ИСЗИ других пользователей близких групп.

Шаг 13. Адаптивная корректировка интерфейса.

Если в режиме реального времени по результатам выполнения Шага 12 при работе пользователя с интерфейсом обнаруживаются критические проблемы (например, постоянные ошибки при вводе данных в элемент конкретного типа), целесообразно оперативно применить микро-коррекцию [16] – улучшить подсказку, поменять тип элемента на более подходящий (например, текстовое поле на выпадающий список) или перестроить порядок элементов.

Шаг 14. Вывод результата решения задачи.

После завершения пользователем всех шагов интерфейс отображает итоговый результат работы (сообщение об успехе, созданный документ, сформированный отчет), что сигнализирует об окончании текущего сеанса. Таким образом процедура решения задачи в ИСЗТ для пользователя считается завершенной.

Шаг 15. Оценка реальной эффективности.

По результатам мониторинга на Шаге 12 производится оценка реальной эффективности интерфейса, то есть ее показателей, вычисленных по работе определенного пользователя с конкретным интерфейсом.

Шаг 16. Эффективность соответствует заданной?

Реальная эффективность интерфейса, полученная на Шаге 15, сравнивается теоретической эффективностью интерфейса, вычисленной при решении оптимизационной задачи на Шаге 8.3. Их различия в том числе могут происходить из-за того, что реальная

максимально учитывает специфику пользователя и решаемой задачи, а теоретическая основана на среднестатистических значениях атомарных эффективностей, аналитически и с некоторой погрешностью преобразованных в интегральные показатели интегральной.

Шаг 16.1. Корректировка атомарных эффективностей.

Фактические данные об эффективности элементов для данного типа пользователя используются для обновления и уточнения базы знаний об атомарных эффективностях, делая будущие расчеты более точными.

Шаг 16.2. Корректировка параметров модели.

Аналогичным с Шагом 16.1 образом, на основе анализа расхождений между прогнозом и реальной работой пользователя корректируются параметры эффективностной модели (например, коэффициенты влияния компоновки на нагрузку) или веса в целевой функции.

Шаг 17. Сохранение синтезированного интерфейса.

Итоговая конфигурация интерфейса, ассоциированная с параметрами модели (варируемыми и стационарными), сохраняется в базе данных для возможного повторного использования в будущем.

Все приведенные шаги обоснованы, реализуемы и логически связаны, что позволяет говорить и об общей работоспособности Метода на практике.

Заключение

В работе описывается метод структурно-параметрического синтеза интерфейса для работы с информационными сервисами, предназначенными для реагирования на информационные запросы пользователя; в основе метода лежит решение оптимизационной задачи, целью которой является повышение эффективности интерфейса, а варируемыми переменными – структура интерфейса и его элементов, а также их типы-параметры. Приведенная схема метода, состоящая из 26 шагов (включая вложенные: 6.1–6.4, 8.1–8.3, 16.1–16.2), является основным научным результатом текущего этапа исследования. Теоретическая значимость работы состоит в схематизации логики работы метода динамического построения интерфейса, а практическая – в возможности непосредственного построения соответствующего генерирующего программного модуля как части ИСЗИ. Продолжением исследования должна стать реализация прототипа средства, выполняющего шаги модуля, проведение с ним экспериментов и получение оценок его практического применения.

Список источников

1. Вострых А.В. Алгоритм оценки эффективности визуальной эстетики интерфейсов специализированных программных продуктов, используемых экстренными службами // Национальная безопасность и стратегическое планирование. 2024. № 3 (47). С. 77–89. DOI: 10.37468/2307-1400-2024-3-77-89.
2. Ахунова Д.Г., Вострых А.В., Курта П.А. Оценка пользовательского интерфейса информационных систем посредством моделей качества программного обеспечения // Информатизация и связь. 2020. № 2. С. 127–135. DOI: 10.34219/2078-8320-2020-11-2-127-135.
3. Курта П.А., Израилов К.Е. Обзор способов построения динамических адаптивных интерфейсов и их интеллектуализация // Научно-аналитический журнал «Вестник Санкт-Петербургского университета Государственной противопожарной службы МЧС России». 2023. № 4. С. 119–132. DOI: 10.61260/2218-130X-2024-2023-4-119-132.
4. Израилов К.Е. Обобщенная классификация уязвимостей интерфейсов транспортной инфраструктуры Умного города // Информационные технологии. 2021. Т. 27. № 6. С. 330–336. DOI: 10.17587/it.27.330-336.
5. Израилов К.Е., Левшун Д.С., Чечулин А.А. Модель классификации уязвимостей интерфейсов транспортной инфраструктуры «умного города» // Системы управления, связи и безопасности. 2021. № 5. С. 199–223. DOI: 10.24412/2410-9916-2021-5-199-223.

6. Курта П.А. Эффективностная модель интерфейса взаимодействия пользователя с информационным сервисом запросного типа // Труды учебных заведений связи. 2023. Т. 9. № 6. С. 102–115. DOI: 10.31854/1813-324X-2023-9-6-102-115.
7. Рыбанов А.А. Количественная оценка несоответствия ментальных моделей пользователя и дизайнера интерфейсов при разработке структуры навигационного меню мобильного приложения // Южно-Сибирский научный вестник. 2021. № 1 (35). С. 19–24.
8. Куделя В.Н. Решение задач перебора путей в сложных графах // Информатика и автоматизация. 2024. Т. 23. № 6. С. 1643–1664. DOI: 10.15622/ia.23.6.3.
9. Atilgan C., Nuriyev U. Hybrid heuristic algorithm for the multidimensional knapsack problem // The proceeding of IV International Conference «Problems of Cybernetics and Informatics». Baku, 2012. Р. 1–4. DOI: 10.1109/ICPCI.2012.6486459.
10. Курта П.А. Система статистического измерения атомарной эффективности графических элементов интерфейсов // Труды учебных заведений связи. 2024. Т. 10. № 6. С. 99–110. DOI: 10.31854/1813-324X-2024-10-6-99-110.
11. Новикова М.В. Как данные о пользователях помогают проектировать сайты // Интернет-маркетинг. 2018. № 4. С. 296–302.
12. Усачев А.Ю. Динамическая генерация визуальных интерфейсов Flutter-приложений // Электронные библиотеки. 2020. Т. 23. № 5. С. 1093–1103. DOI: 10.26907/1562-5419-2020-23-5-1093-1103.
13. Ивченко Д.Д., Горячkin Б.С. Вопросы проектирования и создания эффективного пользовательского интерфейса // Технологии дополненной и виртуальной реальности: сб. статей I Науч.-практ. конф. Москва, 2025. С. 168–174.
14. Мухетдинов А.Р. Методики обучения пользователей работе в информационных системах // Оригинальные исследования. 2022. Т. 12. № 12. С. 259–262.
15. Курта П.А. Взаимодействие пользователя с информационной системой. Часть 2. Алгоритмы обнаружения недостатков // Известия СПБГЭТУ ЛЭТИ. 2020. № 10. С. 34–44.
16. Кравцова Е.Ю., Болбаков Р.Г. Адаптивная генерация пользовательского интерфейса с помощью обучения с подкреплением // Международный журнал информационных технологий и энергоэффективности. 2025. Т. 10. № 7-1 (57). С. 79–84.

References

1. Vostryh A.V. Algoritm ocenki effektivnosti vizual'noj estetiki interfejsov specializirovannyh programmnyh produktov, ispol'zuemyh ekstrennymi sluzhbami // Nacional'naya bezopasnost' i strategicheskoe planirovanie. 2024. № 3 (47). S. 77–89. DOI: 10.37468/2307-1400-2024-3-77-89.
2. Ahunova D.G., Vostryh A.V., Kurta P.A. Ocenka pol'zovatel'skogo interfejsa informacionnyh sistem posredstvam modelej kachestva programmnogo obespecheniya // Informatizaciya i svyaz'. 2020. № 2. S. 127–135. DOI: 10.34219/2078-8320-2020-11-2-127-135.
3. Kurta P.A., Izrailov K.E. Obzor sposobov postroeniya dinamicheskikh adaptivnyh interfejsov i ih intellektualizaciya // Nauchno-analiticheskij zhurnal «Vestnik Sankt-Peterburgskogo universiteta Gosudarstvennoj protivopozharnoj sluzhby MCHS Rossii». 2023. № 4. S. 119–132. DOI: 10.61260/2218-130X-2024-2023-4-119-132.
4. Izrailov K.E. Obobshchennaya klassifikaciya uyazvimostej interfejsov transportnoj infrastruktury Umnogo goroda // Informacionnye tekhnologii. 2021. Т. 27. № 6. S. 330–336. DOI: 10.17587/it.27.330-336.
5. Izrailov K.E., Levshun D.S., Chechulin A.A. Model' klassifikacii uyazvimostej interfejsov transportnoj infrastruktury «umnogo goroda» // Sistemy upravleniya, svyazi i bezopasnosti. 2021. № 5. S. 199–223. DOI: 10.24412/2410-9916-2021-5-199-223.
6. Kurta P.A. Effektivnostnaya model' interfejsa vzaimodejstviya pol'zovatelya s informacionnym servisom zaprosnogo tipa // Trudy uchebnyh zavedenij svyazi. 2023. Т. 9. № 6. S. 102–115. DOI: 10.31854/1813-324X-2023-9-6-102-115.
7. Rybanov A.A. Kolichestvennaya ocenka nesootvetstviya mental'nyh modelej pol'zovatelya i dizajnera interfejsov pri razrabotke struktury navigacionnogo menu mobil'nogo prilozheniya // Yuzhno-Sibirskij nauchnyj vestnik. 2021. № 1 (35). S. 19–24.

8. Kudelya V.N. Reshenie zadach perebora putej v slozhnyh grafov // Informatika i avtomatizaciya. 2024. T. 23. № 6. S. 1643–1664. DOI: 10.15622/ia.23.6.3.
9. Atılgan C., Nuriyev U. Hybrid heuristic algorithm for the multidimensional knapsack problem // The proceeding of IV International Conference «Problems of Cybernetics and Informatics». Baku, 2012. P. 1–4. DOI: 10.1109/ICPCI.2012.6486459.
10. Kurta P.A. Sistema statisticheskogo izmereniya atomarnoj effektivnosti graficheskikh elementov interfejsov // Trudy uchebnyh zavedenij svyazi. 2024. T. 10. № 6. S. 99–110. DOI: 10.31854/1813-324X-2024-10-6-99-110.
11. Novikova M.V. Kak dannye o pol'zovatelyah pomogayut proektirovat' sajty // Internet-marketing. 2018. № 4. S. 296–302.
12. Usachev A.Yu. Dinamicheskaya generaciya vizual'nyh interfejsov Flutter-prilozhenij // Elektronnye biblioteki. 2020. T. 23. № 5. S. 1093–1103. DOI: 10.26907/1562-5419-2020-23-5-1093-1103.
13. Ivchenko D.D., Goryachkin B.S. Voprosy proektirovaniya i sozdaniya effektivnogo pol'zovatel'skogo interfejsa // Tekhnologii dopolnennoj i virtual'noj real'nosti: cb. statej I Nauch.-prakt. konf. Moskva, 2025. S. 168–174.
14. Muhedinov A.R. Metodiki obucheniya pol'zovatelej rabote v informacionnyh sistemah // Original'nye issledovaniya. 2022. T. 12. № 12. S. 259–262.
15. Kurta P.A. Vzaimodejstvie pol'zovatelya s informacionnoj sistemoj. Chast' 2. Algoritmy obnaruzheniya nedostatkov // Izvestiya SPbGETU LETI. 2020. № 10. S. 34–44.
16. Kravcova E.Yu., Bolbakov R.G. Adaptivnaya generaciya pol'zovatel'skogo interfejsa s pomoshch'yu obucheniya s podkrepleniem // Mezhdunarodnyj zhurnal informacionnyh tekhnologij i energoeffektivnosti. 2025. T. 10. № 7-1 (57). S. 79–84.

Информация о статье:

Статья поступила в редакцию: 12.11.2025; одобрена после рецензирования: 10.12.2025; принята к публикации: 15.12.2025

Information about the article:

The article was submitted to the editorial office: 12.11.2025; approved after review: 10.12.2025; accepted for publication: 15.12.2025

Информация об авторах:

Курта Павел Андреевич, соискатель Санкт-Петербургского университета ГПС МЧС России (196105, Санкт-Петербург, Московский пр., д. 149), e-mail: expert@kurta.ru, <https://orcid.org/0009-0005-6073-8626>, SPIN-код: 9667-2220

Information about authors:

Kurta Pavel A., applicant of Saint-Petersburg university of State fire service of EMERCOM of Russia (196105, Saint-Petersburg, Moskovsky ave., 149), e-mail: expert@kurta.ru, <https://orcid.org/0009-0005-6073-8626>, SPIN: 9667-2220