

Проектирование для многообразия – конструктор и DSL вместо жесткой реализации требований



Максим Цепков

IT-архитектор и аналитик,
навигатор по миру Agile
и бирюзовых организаций

<http://mtsepkov.org>

WORLD IA DAY 2019
LOCAL CONNECTIONS. GLOBAL IMPACT.

DESIGN FOR DIFFERENCE

Санкт-Петербург 23.02.2019



В чем проблемы?

- Заказная разработка:
 - Аналитик описал процессы, сделал проект системы и интерфейсов,
 - Разработчик реализовал по проекту для **текущего** состояния бизнеса
 - Но бизнес-процессы меняются – получается вечный поток больших изменений
 - А если появляется новый клиент, оказывается, что вариативность не заложена
- Продуктовая разработка:
 - Есть обобщенная платформа, на ней сделаны различные решения, казалось бы, их можно произвольно настраивать
 - Оказывается, настроенные документы согласованы между собой, но это не документировано, поэтому при доработках работа целого нарушается ☹
 - А еще многообразие мира часто больше заложенного в платформу, предусмотренных настроек не хватает

Общий подход к решению

- Чувствуем и фиксируем точки предполагаемых изменений
 - Набор типов и атрибутов у объектов
 - Представление данных на интерфейсах
 - Набор состояний и переходов, бизнес-логика на них
- В точках расширения делаем гибкую реализацию
 - Для начала – легкие декларативные описания, изменяемые разработчиком
 - Если нужно – выносим настройку технологам через DSL или делаем специализированные формы
 - Помним, что пользователи справляются со сложными формулами в Excel – так что с простым DSL вполне справляется, а DSL дает гибкость
 - Не забываем, что могут потребоваться расширения в коде «по месту»
 - Избегаем чрезмерных обобщений, слишком сложный DSL становится кодом...

Интерфейсные формы

Гибкая настройка форм пользователем

Таблицы документов и отчетов часто содержат много колонок.
Для разных задач пользователю интересны **разные** колонки и
фильтры – надо переключаться между вариантами настроек

Справка о возможности фильтров и колонок форм классификации ЛС

Классификация проводок

Шаблоны Порядок фильтров

Шаблон: Установленные настройки

Наименование	Фильтр	Колонка
Дата открытия	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Дата закрытия	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Состояние	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Счет 2-го порядка	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Номер счета	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Название	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Валюта	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Владелец счета	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Экономический агент	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Филиал	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Дата изменения.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ответисполнитель в КВ	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Статья	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Тип счета	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Сохранить
Установить
Создать
Удалить
Отмена

Проводки Обновить Просмотр Настройки Дата действия классификаторов: 29.01.2015

Основной классификатор: 29.01.2015 Основной классификатор: Не установлен:

Связанный классификатор ЛС: Связанный классификатор БС2: Связанный классификатор контрагента: Связанный классификатор страны:

Дата проводки с: по: Сумма (руб.) от: до: Раздел плана счетов: Балансовый счет: БС дебета: БС кредита: Лицевой счет: Владелец счета:

ID операции в АС ГБК: ID проводки в АС ГБК: Валюта: Филиал: Исполнитель: Архивная пачка: GUID операции:

ID проводки в А/Дата проводки Номер документа Счет дебета Счет кредита Сумма проводки Сумма Дт Сумма Кт Назначение пла Филиал ID операции Архивная пачка GUID операции Исправления Тест Банк ГПБ (АО) 133962 111

18705891	03.03.2014	2168	4230197880001 4230197880001	1 048 839,30	23 307,54	(VO21200)	Банк ГПБ (АО) 47969	123xxx	F942C9F9-12
20285267	14.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150770	CCCCCCCQ-4F	
25445087	21.05.2014	123		900,00	900,00				
62219167	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 48471	123xxx	F942C919-12
62219168	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150618	123xxx	F942C919-12
62219169	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150619	123xxx	F942C919-12
62219170	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150620	123xxx	F942C919-12
62219171	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150621	123xxx	F942C919-12
62219172	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150622	123xxx	F942C919-12
62219173	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150623	123xxx	F942C919-12
62219174	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150624	123xxx	F942C919-12
62219175	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150625	123xxx	F942C919-12
62219176	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150626	123xxx	F942C919-12
62219177	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150627	123xxx	F942C919-12
62219178	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150628	123xxx	F942C919-12
62219179	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150629	123xxx	F942C919-12
62219180	13.05.2014	2168	4230197880001 4230197880001	1 129 853,98	23 307,54	(VO21200)	Банк ГПБ (АО) 150630	123xxx	F942C919-12

Основной классификатор Обновить Изменить Просмотр Удалить

Значение Дата нача/ Дата оконч/ Дата блоку/ Путь

Связанный классификатор Обновить

Вид	Объект	Значение	Дата нача/	Дата оконч-	Дата блоку/	Путь
Аналитиче	202821096222222222	TEST	30.09.2014	30.09.2015		KLS_UPD_TYPE/KLS_UPD_UZEL1
Балансов	20202. Касса кредитны	ПРКПС_ДОК	31.07.1990	31.12.2049		Б2/РБО/Ф652/ПРКПС/
Балансов	20202. Касса кредитны	TEST_LIST2	09.02.2013	09.02.2015		TEST/LALALA/TEST2/
Страны	РОССИЯ	СНГ	08.12.1991	31.12.2049		СТРАНА/МГО/МГО/

История изменения основного классификатора Ручное обновление Обновить

Номер лога Сущность/Атрибут Дата Пользователь Старое значение Новое значение

Декларативное описание форм

```
///!<object name="f_GrcOpenPayDocTable" type="function">
///!<abstract>Открытие формы для работы
///! с платежными поручениями.
///!</abstract>
function f_GrcOpenPayDocTable()
{
    w_top.g_lib.m_OpenForm(
        'pay_doc_table_old',
        'grc_std_table.html',
        'Платежные поручения',
        {
            std_table: {
                title: 'Платежные поручения',
                request: 'pay.pay_doc_table_old',
                result: 'pay_doc_table_old',
                filter: {
                    request: 'pay.pay_doc_old_flt',
                    result: 'pay_doc_old_flt'
                },
                actions: {
                    new_pay: {
                        button: 'Новые платежи',
                        url: "grc_std_table.html",
                        name: '"pay_doc_table_new"',
                        title: '"Новые платежные поручения"',
                        features: 'status=1,toolbar=0,location=0,menubar=0,resizable=1',
                        args: {
                            std_table: {
                                title: 'Новые платежные поручения',
                                request: 'pay.pay_doc_table',
                                result: 'pay_doc_table',
                                filter: {
                                    request: 'pay.filter',
                                    result: 'pay_filter'
                                }
                            },
                            actions: {
                                create: {
                                    button: 'Создать платежи',
                                    request: "pay.create",
                                    args: '{'
                                        + 'dt_pay: document.all.flt_dt_pay.value,' +
                                        + 'id_account: f_GrcTblGetRowsAttr(w_table,"id_account",true),' +
                                        + 'summa: f_GrcTblGetRowsAttr(w_table,"summa",true)' +
                                    '}'
                                },
                                report_total: {
                                    button: 'Расщепление общее',
                                    url: 'f_GrcOpenPrintForm( "wrapper.pay_total_report", {dt_received:document.all.flt_dt_pay.value})'
                                },
                                report: {
                                    button: 'Расщепление по ЛС',
                                    url: 'f_GrcOpenPrintForm( "wrapper.pay_report", {dt_received:document.all.flt_dt_pay.value})'
                                }
                            }
                        }
                    }
                }
            }
        }
    )
}
```

Json описывает структуру форм, позволяя гибко компоновать интерфейс разработчику, а в нужных местах – **позднее добавить интерактивную настройку аналогичную предыдущей**

```
actions: {
    create: {
        button: 'Создать платежи',
        request: '"pay.create"',
        args: '{'
            + 'dt_pay: document.all.flt_dt_pay.value,' +
            + 'id_account: f_GrcTblGetRowsAttr(w_table,"id_account",true),' +
            + 'summa: f_GrcTblGetRowsAttr(w_table,"summa",true)' +
        '}'
    },
    report_total: {
        button: 'Расщепление общее',
        url: 'f_GrcOpenPrintForm( "wrapper.pay_total_report", {dt_received:document.all.flt_dt_pay.value})'
    },
    report: {
        button: 'Расщепление по ЛС',
        url: 'f_GrcOpenPrintForm( "wrapper.pay_report", {dt_received:document.all.flt_dt_pay.value})'
    }
}
```

А если у вас жесткий фреймворк с wysiwyg?

Подход тоже работает, хотя и ограничено

- Обычно можно сделать скрытие и перестановку колонок в таблицах
- Можно заложить несколько сменных настроек для фильтров и таблиц
- Бывает, внутри фреймворка есть текстовое представление (xml или аналог), и в арі есть доступ на его изменение

Навигация по интерфейсу



Этого не хватает в очень
многих приложениях

■ Кейсы использования

- По любому полю-ссылке – открыть форму с описанием объекта, например, открыть карточку клиента из любых форм с документами клиента
- По любому полю-ссылке открыть форму с установленными фильтрами, например, из заказа клиента перейти ко всем платежам этого клиента
- Для выбора, помимо удобного списка, уметь открыть справочник, или форму для создания: клиент для договора заполняется по номеру телефона, но можно прямо сразу создать нового клиента или открыть поиск клиентов по атрибутам

■ Простой способ сделать

- Не прописывать это для каждой формы, а добавить как аспект ко всем
- Опираться на имена полей (сделать соглашение), но оставить исключения
- Сделать декларативное описание: поле → действие

Структуры данных и бизнес-логика

Структура объектов

- В любом объекте нужен произвольный комментарий пользователя
 - К любому объекту может потребоваться добавить атрибут, и не один
 - Списки являются расширяемыми и превращаются в справочники
- Лучше, когда новые атрибуты сразу подхватывает интерфейс

Документооборот

Примеры решений

- Настраиваемый граф переходов документов, сделок, задач
- Система виз или утверждений на документе
- Система планирования задач с описаниями зависимостей

Плюсы и минусы

- 😊 Гибкая настройка вариантов переходов
- 😢 С переходами связана бизнес-логика – как настраивать?
- 😢 Изменение логики часто нарушает целостность работы системы

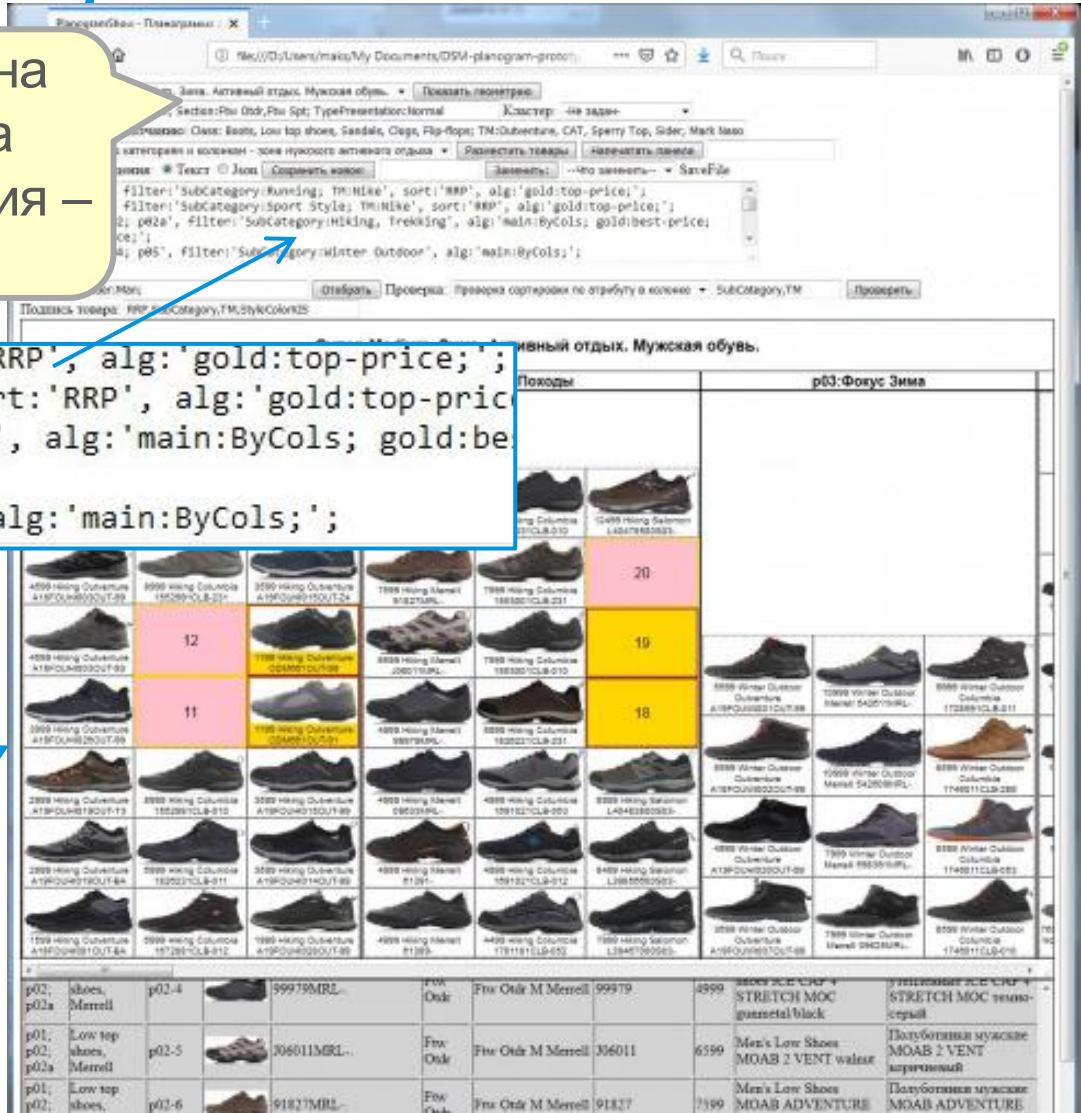
Важно, чтобы интерфейс подхватывал изменения «на лету».

Калькуляторы с объяснением результата

- Вычисление скидок покупателя при множестве различных акций
 - Исходная постановка: множество конкретных правил, накопленных за историю
 - Сделано: обобщенные правила попарного взаимодействия и автомат, который применяет все возможные варианты и, ищет наилучший и объясняет
 - Профит: парное взаимодействие пользователям легко указать и объяснить, а форма объяснения показывает, что выбранный вариант – самый выгодный
- Коммунальный биллинг
 - Исходная постановка: множество нормативных актов по правилам начисления в зависимости от характеристик квартиры, проживающим и их льготам
 - Сделано: промежуточная сущность «база тарифа», содержит все характеристики, на которых опирается вычисление суммы и льгот
 - Профит: сложный алгоритм разбит на два, с показом промежуточных данных, что позволяет проверить работу на калькуляторе и объяснить потребителю

Легкие прототипы на html+js

Свободные структуры данных на json, логика на jscript, простые формы и вывод результатов на html, работа в браузере как desktop-приложения – прототип разрабатывается быстро и легко



The screenshot shows a desktop application window titled "РаспродажиБаза - Планшеты". The interface includes a toolbar with buttons for "Показать ленту", "Категории и коллекции", "Фильтр", "Сохранить как...", "Завершить", and "Сохранить файл". A dropdown menu "Кластер" is open, showing options like "не задан", "Текст", "JSON", "Сохранить как...", "Завершить", and "Save File". Below the toolbar, there's a search bar with placeholder text "Что ищем..." and a "Поиск" button. The main area displays a grid of shoe products. At the top of the grid, there are several filters: "area: 'nike1', filter: 'SubCategory:Running; TM:Nike', sort: 'RRP', alg: 'gold:top-price';", "area: 'nike2', filter: 'SubCategory:Sport Style; TM:Nike', sort: 'RRP', alg: 'gold:top-price';", "area: 'p01; p02; p02a', filter: 'SubCategory:Hiking, Trekking', alg: 'main:ByCols; gold:best-price; sort: focus=top-price';", and "area: 'p03; p04; p05', filter: 'SubCategory:Winter Outdoor', alg: 'main:ByCols; ;'". The grid itself has columns for "Покупки" and "p03:Фокус Зима". The grid cells contain shoe images, product names, and prices. A yellow arrow points from the text "Свободные структуры данных на json, логика на jscript, простые формы и вывод результатов на html, работа в браузере как desktop-приложения – прототип разрабатывается быстро и легко" to the application window.

```
area: 'nike1', filter: 'SubCategory:Running; TM:Nike', sort: 'RRP', alg: 'gold:top-price';
area: 'nike2', filter: 'SubCategory:Sport Style; TM:Nike', sort: 'RRP', alg: 'gold:top-price';
area: 'p01; p02; p02a', filter: 'SubCategory:Hiking, Trekking', alg: 'main:ByCols; gold:best-price; sort: focus=top-price';
area: 'p03; p04; p05', filter: 'SubCategory:Winter Outdoor', alg: 'main:ByCols; ;'
```

```
    },
    p16: { title: "Фокус Тренировки", type: "focus",
    nike1: { title: "Nike Бег", type: "panel", gold: [-11, -12, 18, 19],
    nike2: { title: "Nike Спортивный стиль", type: "panel", gold: [18, 19, -20],
    },
    floor: {
      rLeft: { type: "rowLeft",
        a00g: { type: "gSingle", size: 5, title: "Columbia" },
        b00b: { type: "cell",
          b00g: { type: "gDouble", size: 5, title: "Columbia" },
          b00t: { type: "tail", size: 2 }
        }
      },
      row01: { type: "row",
        a01g: { type: "gDouble", size: 5, title: "Merrell" },
        b01b: { type: "cell",
          b01g: { type: "gSingle", size: 5, title: ["Columbia", "Merrell"] },
          b01t: { type: "tail", size: 2 }
        }
      },
    },
  },
```

Отчеты

- Шаблоны в Excel (или Word), наполнение из системы
 - Система хранит шаблоны и формирует по ним готовые отчеты
 - Или шаблоны лежат отдельно, и в них информация о запрашиваемых данных
 - Для запроса данных полезен легкий DSL формул в терминах бизнес-объектов
 - Excel/Word позволяют редактировать отчеты после создания, решая проблемы с печатью и оперативным исправлением, если надо – в отличие от pdf
- От разовых запросов к повторяющему отчету
 - Разработчики умеют быстро писать SQL запросы и выгружать результат
 - Но пользователю часто надо повторить запрос, иногда с другими параметрами
 - Решение – форма, которая показывает результаты запросов в обобщенном виде с минимальными настройками колонок и параметров отчета

Кто настраивает DSL

- DSL уровня пользователей – самые высокие требования к простоте
 - Многие пользователи успешно работают с формулами Excel – а там непросто
 - Часто достаточно DSL, доступного опытным пользователям, а не всем
- DSL настраивают специально обученные технологии бизнеса
 - Помните – технологии часто заняты и будут просить о настройке поддержку
 - Однако, они часто могут быстро исправить «по аналогии», и это – удобно
 - Настройка не требуют пересборки софта и это – ускоряет изменения
- DSL для разработчиков
 - Выносит часть алгоритма в декларативные описания
 - Позволяет быстро вносить изменения, в том числе не знакомому с кодом
 - Разносит логику и конечные решения, получается аналог MVC шаблона

Что я хотел донести своим рассказом?

- Главное – чувствовать: это место будет меняться, надо быть готовым
- DSL или конструктор в точках изменений повышает эффективность
- Даже скрытые микро-DSL «для разработчика» – полезны
- DSL можно добавить не сразу, это как с выделением функций
- Делать интерфейс настройки надо после реализации кейсов, убедившись в покрытии многообразия, и оставляя места расширения

Насколько это получилось – судить вам



Максим Цепков
<http://mtsepkov.org>
maks.tsepkov@ya.ru

На сайте много материалов по [анализу и архитектуре](#),
[Agile](#), и [ведению проектов](#), [управлению знаниями](#),
мои [доклады, статьи](#) и [конспекты книг](#).