

1. Въведение

ЛЕКЦИОНЕН КУРС: ШАБЛОНИ ЗА ПРОЕКТИРАНЕ

ДОЦ. Д-Р ЕМИЛ ДОЙЧЕВ

Мотивация

- ✓ Проектирането на ОО софтуер е трудно
- ✓ Проектирането на софтуер за *многократна употреба* е още по-трудно
- ✓ Проектът трябва да е:
 - специфичен за текущия проблем
 - достатъчно общ за да отговори на бъдещите изисквания

Мотивация

- ✓ Опитът е ключов фактор за създаване на добър обектно-ориентиран проект
 - проблемите не се решават чрез „откриване на колелото“ всеки път
- ✓ Проектантите с опит – експерти
 - използват добри решения, които са послужили в миналото
- ✓ В много ОО системи има повтарящи се примери за класове и комуникиращи по между си обекти
 - ✓ това са т.нар. шаблони, които разрешават специфични проектантски проблеми
 - ✓ правят ОО проекта:
 - ✓ по-гъвкав
 - ✓ по-елегантен
 - ✓ с повече възможности за повторна (многократна) употреба

Какво са шаблоните за дизайн?

- ✓ Шаблонът за дизайн (проектиране) систематично именува, разяснява и оценява важен и повтарящ се дизайн в ОО системи.
- ✓ Улесняват многократната употреба на успешни дизайни и архитектури.
- ✓ Документирането им ги прави по-достъпни за разработчиците на нови системи.
- ✓ Спомагат за правилния избор на дизайнерски алтернативи, които правят една система годна за многократно използване и отхвърлят онези, които пречат на тази цел.
- ✓ Подобряват документацията и поддръжката на системата като въвеждат изрична спецификация на взаимодействията между класове и обекти.

Шаблон за дизайн

- ✓ Шаблоните документират често срещан проблем и неговото решение в определен контекст.
- Шаблоните правят връзка между проблеми и решения
- Дефиниция на термините:
 - **Контекст** – среда, обкръжение, ситуация или взаимосвързани условия, при които нещо съществува.
 - **Проблем** – отворен въпрос; нещо, което трябва да бъде изследвано и решено; обикновено проблемът е ограничен в рамките на контекста.
 - **Решение** – отговор на проблема, в рамките на контекста, който помага за разрешаването му.

Дефиниция за шаблон (1)

Всеки шаблон представлява правило, състоящо се от три части, които дават връзката между някакъв контекст, проблем и решение.

--- Кристофър Александър, “A Pattern Language”

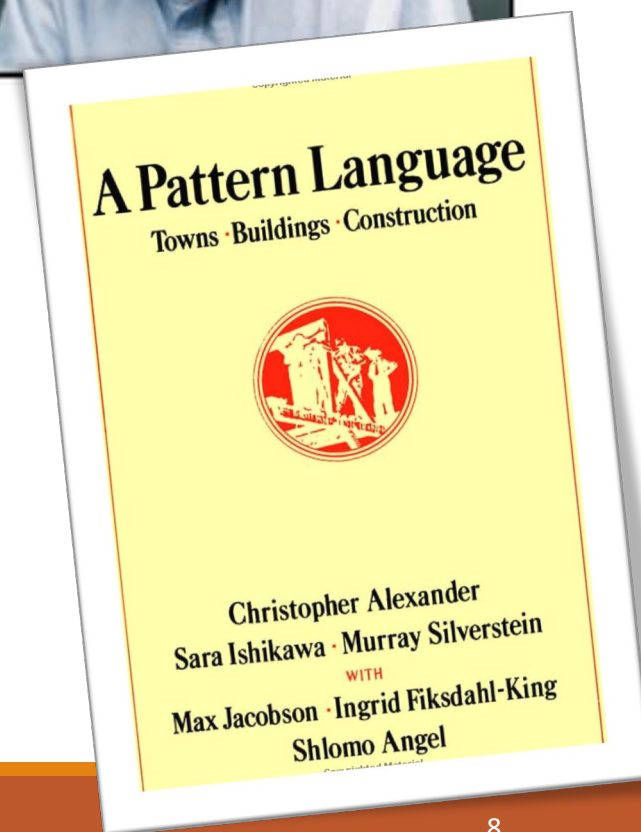
Christopher Alexander

- ✓ Кристофър Александър, 70те години на миналия век – няколко книги свързани със шаблоните в строителството и архитектурата.
 - Роден 1936 във Виена
 - Над 200 сгради в Калифорния, Япония, Мексико и др.

“A Pattern Language”, Oxford University Press, 1977
ISBN-10: 0195019199, ISBN-13: 978-0195019193

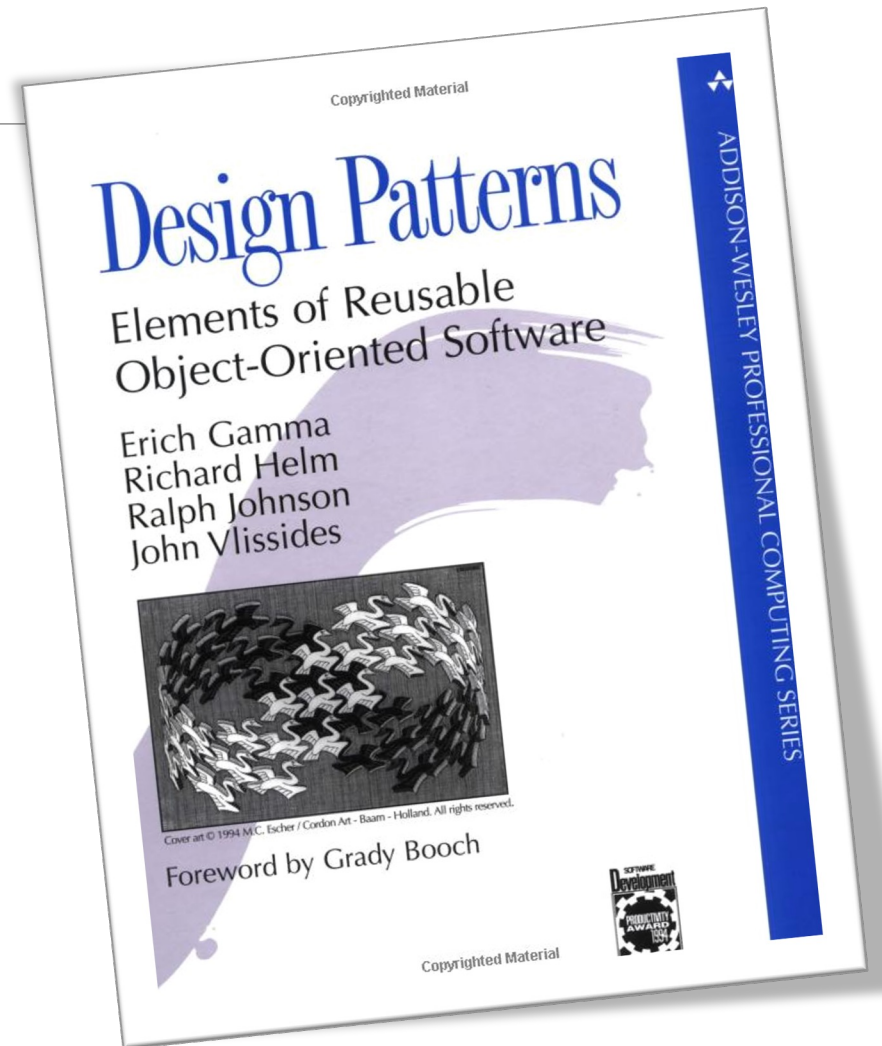
<http://www.patternlanguage.com/>

Тази идея постепенно е възприета и в софтуерните технологии.



Шаблоните в софтуера

- ✓ Популяризирането на шаблоните в софтуера става с книгата “**Design Patterns: Elements of Reusable Object-Oriented Software**”
 - Авторы: Ерик Гама, Ричард Хелм, Ралф Джонсън, Джон Влсидес – известни като “Gang of Four”.
 - Ноември, 1994, ISBN-10: 0201633612, ISBN-13: 978-0201633610
- ✓ Описаните шаблони, не са разработка на авторите, а са резултат от изследване на много проекти и документиране на общите елементи в дизайна.



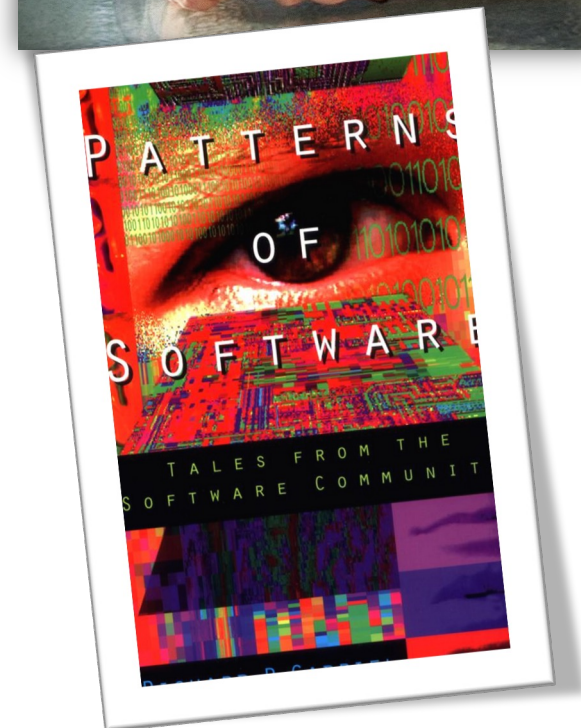
Дефиниция за шаблон (2)

Всеки шаблон представлява правило, състоящо се от три части, които дават връзката между някакъв контекст, някаква система от условия, които се появяват често в този контекст, и някаква софтуерна конфигурация, която позволява тези условия да бъдат изпълнени.

--- Ричард Гейбриъл

Richard P. Gabriel

- ✓ Ричард Гейбриъл, роден 1949 г. е американски учен известен с работата си свързана с езика Lisp. В момента работи в IBM.
- ✓ Най-известната му работа е „**Lisp: Good News, Bad News, How to Win Big**“, която въвежда фразата „Worse is Better“ – в смисъл, че по-малко функционалност (*worse*) е за предпочитане (*better*) що се отнася до практичност и използваемост.
 - Т.е. софтуер, който е ограничен, но лесен за използване, може да е по-привлекателен за потребителите, отколкото обратното.
- ✓ **Patterns of Software: Tales from the Software Community**, 1998, ISBN-10: 0195121236, ISBN-13: 978-0195121230



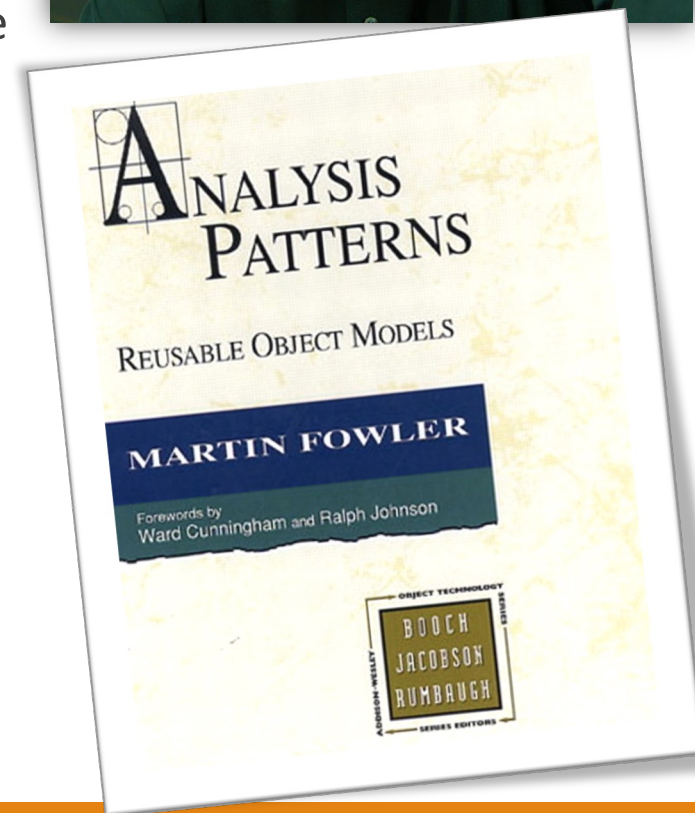
Дефиниция за шаблон (3)

Шаблонът е идея, която е била полезна в един конкретен контекст и вероятно може да бъде използвана и в други.

--- Мартин Фаулър, Analysis Patterns

Martin Fowler

- ✓ Мартин Фаулър, роден 1963 г. е английски автор и публицист в сферата на компютърно програмиране, по-специално в областта на прилагането на модели, UML, и гъвкавите методологии за създаване на софтуер (SCRUM, Rational Unified Process, Agile).
- ✓ Неговите статии и книги са използвани като учебни помагала от редица университети по цял свят.
- ✓ Автор на 8 книги
- ✓ **Analysis Patterns**, 1996, ISBN-10: 9780201895421, ISBN-13: 978-0201895421
- ✓ <http://martinfowler.com/>



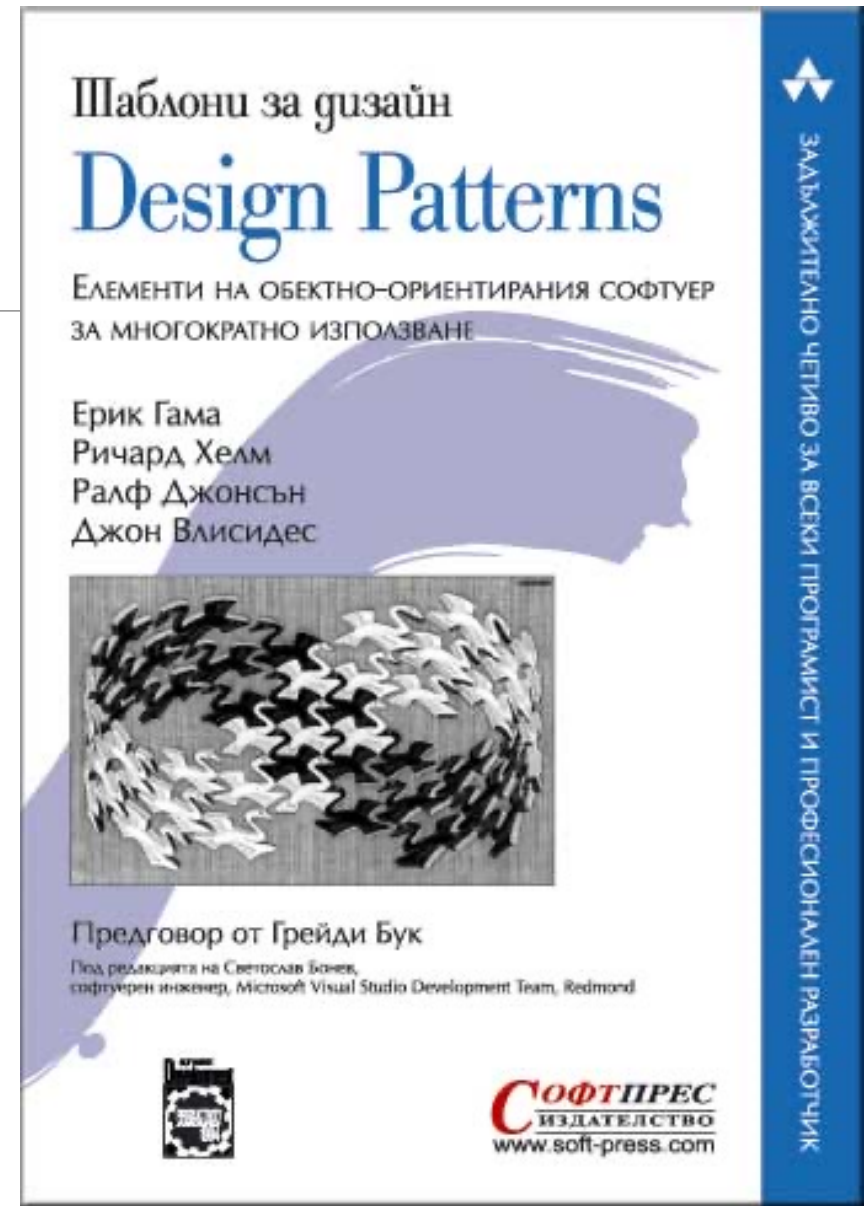
Категоризация на шаблоните

✓ Най-често използваните категории шаблони са:

- шаблони за проектиране (Design patterns)
- шаблони за архитектура (Architectural patterns)
- шаблони за анализ (Analysis patterns)
- създаващи шаблони (Creational patterns)
- структурни шаблони (Structural patterns)
- поведенчески шаблони (Behavioral patterns)

Литература

- ✓ Erich Gamma et al, **Design Patterns**, ISBN: 0-201-63361-2, Addison-Wesley Publ. Co., January 15, 1995.
 - Ерик Гама, Хелм Р., Джонсън Р., **Шаблони за дизайн**, ISBN: 954-685-352-6, СофтПрес, 2005



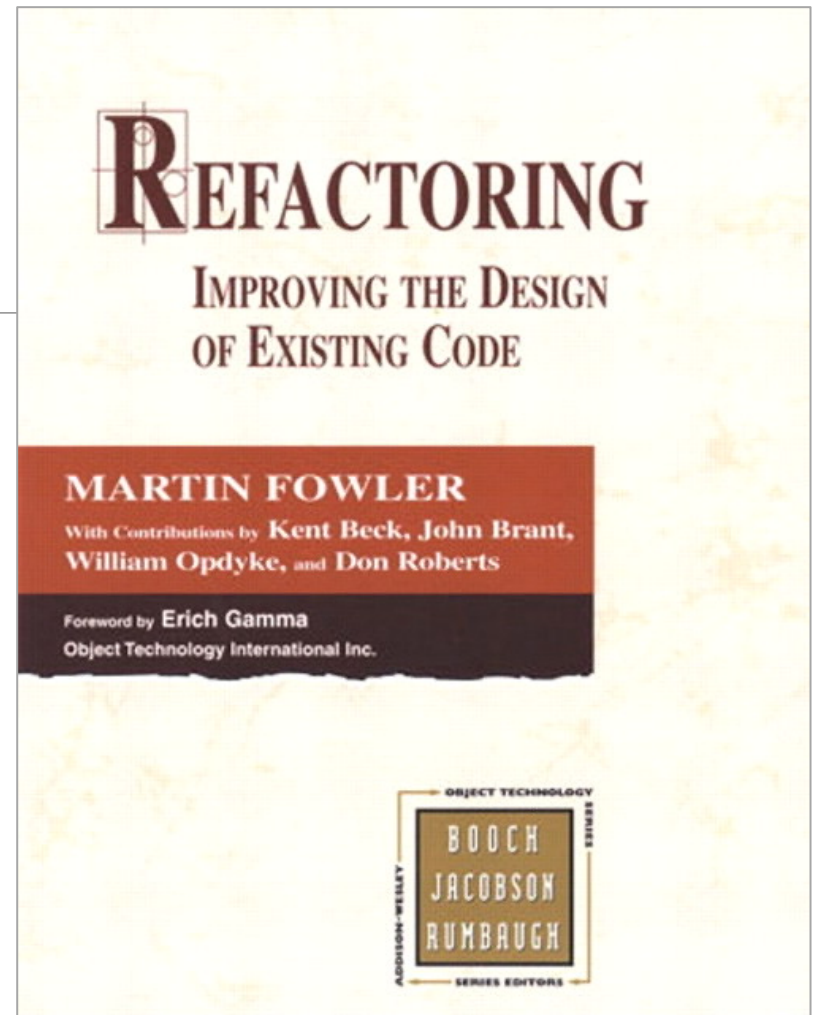
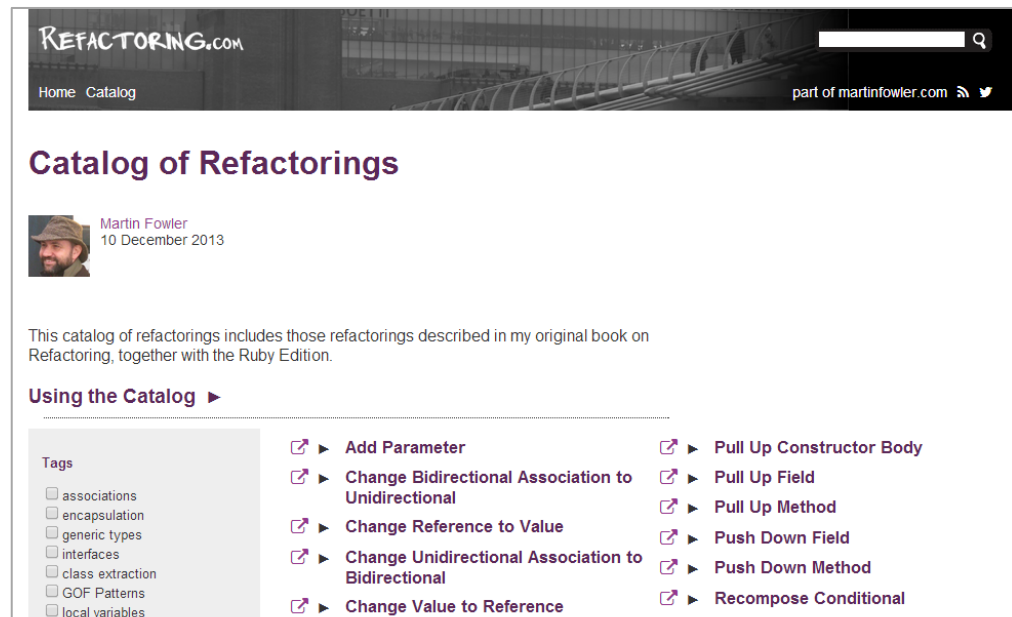
Литература

- ✓ Eric Freeman et al, **Head First Design Patterns**, ISBN: 0-596-00712-4, O'Reilly Media Inc., 2004

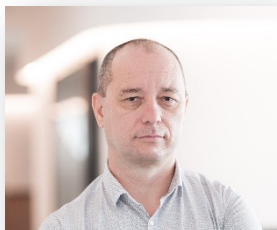


Литература

- ✓ Martin Fowler et al, **Refactoring: Improving the Design of Existing Code**, ISBN: 0-201-48567-2, Addison-Wesley Publ. Co., June 28, 1999.
- ✓ <http://refactoring.com/catalog/>



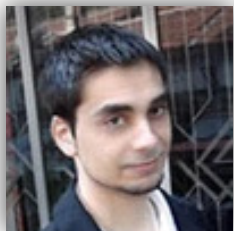
Преподавателите



✓ доц. д-р Емил Дойчев

каб. 240

е-mail: e.doychev@uni-plovdiv.bg



✓ ас. Александър Петров

е-mail: apetrov@uni-plovdiv.bg

Катедра:
Компютърни системи
ФМИ, ПУ

Материалите

- ✓ Учебни материали:
 - ✓ Google Classroom
- ✓ Тестова система:
 - ✓ DeLC: <http://delc.fmi.uni-plovdiv.net>
 - ✓ Задължително поне едно влизане в DeLC преди провеждането на самия тест.
 - ✓ Достъп с акаунтът от e-portal.

Оценяване

✓ **A**: Оценка от упражненията (оценка от 2 до 6) – последната седмица

✓ **B**: Теоретичен изпит (оценка от 2 до 6) – електронен тест

- На редовната дата – само с получена оценка **A** > 2
- На поправка – може и без оценка **A** (ще се решава и практическа задача)

✓ Крайната оценка

$$(\mathbf{A} == 2 \mid \mid \mathbf{B} == 2) \text{ ? } 2 : (\text{int})\text{Math.floor}((\mathbf{A} + \mathbf{B}) / 2 + 0.5)$$

Въпроси?

Край: Въведение

ЛЕКЦИОНЕН КУРС: ШАБЛОНИ ЗА ПРОЕКТИРАНЕ