

Umění systemového návrhu

Objektově orientovaná
tvorba informačních
systémů pomocí
původní metody BORM

Jiří Poblák
Květa Merunková
Antonín Grada



Česká společnost
pro systémovou integraci

Upozornění pro čtenáře a uživatele této knihy

Všechna práva vyhrazena. Žádná část této tištěné či elektronické knihy nesmí být reprodukována a šířena v papírové, elektronické či jiné podobě bez předchozího písemného souhlasu nakladatele. Neoprávněné užití této knihy bude **trestně stíháno**.

Používání elektronické verze knihy je umožněno jen osobě, která ji legálně nabyla a jen pro její osobní a vnitřní potřeby v rozsahu stanoveném autorským zákonem. Elektronická kniha je datový soubor, který lze užívat pouze v takové formě, v jaké jej lze stáhnout s portálu. Jakékoliv neoprávněné užití elektronické knihy nebo její části, spočívající např. v kopírování, úpravách, prodeji, pronajímání, půjčování, sdělování veřejnosti nebo jakémkoliv druhu obchodování nebo neobchodního šíření je zakázáno! Zejména je zakázána jakákoliv konverze datového souboru nebo extrakce části nebo celého textu, umístování textu na servery, ze kterých je možno tento soubor dále stahovat, přitom není rozhodující, kdo takovéto sdílení umožnil. Je zakázáno sdělování údajů o uživatelském účtu jiným osobám, zasahování do technických prostředků, které chrání elektronickou knihu, případně omezují rozsah jejího užití. Uživatel také není oprávněn jakkoliv testovat, zkoušet či obcházet technické zabezpečení elektronické knihy.



Umění systémového návrhu

Objektově orientovaná tvorba informačních systémů
pomocí původní metody BORM

Jiří Polák, Vojtěch Merunka, Antonín Carda

Ediční rada:

doc. Ing. Josef Basl, CSc. – ZČU Plzeň, VŠE Praha – předseda
prof. Ing. Zdeněk Molnár, CSc. – UTB Zlín, FS ČVUT Praha – místopředseda

PhDr. Jiří Adamík – Grada Publishing a.s.
prof. Ing. Jan Ehleman, CSc. – Technická univerzita v Liberci
doc. Ing. Karol Matiaško, CSc. – Žilinská univerzita v Žiline
doc. RNDr. Jaroslava Mikulecká, CSc. – Univerzita Hradec Králové
prof. RNDr. Jaroslav Pokorný, CSc. – MFF UK Praha
doc. Ing. Jan Pour, CSc. – VŠE Praha
doc. Ing. Karel Richta, CSc. – FEL ČVUT Praha
doc. Ing. Milena Tvrdíková, CSc. – VŠB-TU Ostrava
prof. Ing. Ivan Vrana, DrSc. – Česká zemědělská univerzita v Praze

Ing. Jiří Polák, CSc.
Ing. Vojtěch Merunka, Ph.D.
Ing. Antonín Carda, CSc.

Umění systémového návrhu

Objektově orientovaná tvorba informačních systémů
pomocí původní metody BORM

© Grada Publishing a.s., 2003

Cover Design © Grada Publishing a.s., 2003

Vydala Grada Publishing a.s., U Průhonu 22, Praha 7
tel.: +420 220 386 401, fax: +420 220 386 400, www.grada.cz

jako svou 1613. publikaci

Odpovědný redaktor Petr Somogyi

Sazba Antonín Plicka

Počet stran 196

První vydání, Praha 2003

Vytiskly Tiskárny Havlíčkův Brod, a. s.

Husova ulice 1881, Havlíčkův Brod

ISBN 80-247-0424-2 (tištěná verze)

ISBN 978-80-247-6332-3 (elektronická verze ve formátu PDF)

© Grada Publishing, a.s. 2011

Obsah

O autorech	8
Předmluva	9
Poděkování	10
Úvod	11
1. Paralely s architekturou	13
1.1 Architekti a stavitelé informačních systémů	13
1.1.1 Umění stavět	13
1.1.2 Lákadla moderního šamana a skutečnost	15
1.1.3 Když šaman selže	16
1.1.4 „Systémový integrátor“, jeho metody a nástroje	16
1.1.5 Stavební stroje nestačí	17
1.2 Model procesů a informačního systému	18
1.2.1 Abstraktní myšlení a modely	18
1.2.2 Funkční model – prototyp ještě neexistujícího originálu	20
1.3 Stavba bez modelu	21
1.3.1 I bez modelu lze stavět, ale jaké jsou následky	21
1.3.2 Příklady stavby bez modelu	22
1.3.3 Hlavní rizika stavby bez modelu	22
1.3.4 Jak je možné ošetřit rizika	22
1.3.5 Kdy stavět bez modelu	23
1.3.6 Modelky a modely	23
1.3.7 Shoda modelu a řešení	23
1.4 Evoluce modelů	25
1.4.1 Zpětný pohled na modely a modelování	25
1.4.2 Vývoj programovacích technik	27
1.4.3 Vývoj metodik analýzy a návrhu informačních systémů	31
1.4.4 Objektově orientované metodologie	32
1.4.5 Životní cyklus vývoje I. S.	35
1.4.6 Současné objektové metodologie	36
1.5 Stavíme s modelem: objekt, proces, abstrakce reality	37
1.5.1 Okolnosti stavby	37
1.5.2 Procesy	39
1.5.3 Stavební prvky – objekty	39
1.5.4 Objekty v procesech	41
1.6 Objekty kolem nás	43
1.6.1 Komunikace mezi objekty	44
1.6.2 Zapouzdření	45
1.6.3 Objektové kostky systémové stavebnice	45
1.6.4 Objektový model procesů	47
1.6.5 Objektová skládanka	48

1.6.6	Objekt jako chameleon	48
1.6.7	Třídy – továrny na objekty	49
1.6.8	Jak nezahodit, co se už udělalo	50
1.6.9	Softwarové čipy	50
1.7	Draze vykoupené zkušenosti	50
1.7.1	Nespolehlivý a drahý software?	50
1.7.2	Historické souvislosti	51
1.7.3	Řešení	52
1.8	Změna procesů – Business Process Reengineering	53
1.8.1	Historie business inženýrství	54
1.8.2	Myšlenka konvergenčního inženýrství	55
1.8.3	Komponentový pohled na model architektury organizace	56
1.9	Náš přístup je BORM	57
1.9.1	Náš přístup jako metoda	58
1.9.2	Struktura metody	59
1.9.3	Vlastnosti metody BORM	60
1.10	Evoluce přístupů k životnímu cyklu	61
1.10.1	Konceptuální modelování	61
1.10.2	Varianty životního cyklu	62
1.11	Objektově orientovaný proces tvorby softwaru	64
1.11.1	Potřeba provozní, testovací a vývojové platformy	64
1.11.2	Proces technického zabezpečení projektu	64
1.11.3	Jednotlivé role v životním cyklu systému	67
1.12	Petriho sítě, teorie automatů a lambda kalkulus	68
1.12.1	Petriho sítě	68
1.12.2	Teorie automatů	74
1.12.3	Lambda kalkulus	78

2.	Od myšlenky k hotovému	81
2.1	Životní cyklus systému	81
2.1.1	Fáze životního cyklu	81
2.1.2	Vývoj pojmu objekt	83
2.2	Co stačí k modelování	85
2.2.1	Základní pojmy objektového modelování	85
2.2.2	Odvozené pojmy objektového modelování	89
2.3	Uživatelé a analytici – autoprovaz	93
2.3.1	Strategická analýza	93
2.3.2	Vstupní analýza	97
2.4	Analytici a vývojáři	113
2.4.1	Analýza systému	113
2.4.2	Konceptuální počítačový model	120
2.5	Vývojáři a uživatelé	121
2.5.1	Závěrečné fáze modelování systému	121
2.5.2	Objektové komponenty	124
2.6	Manažer projektu	131
2.6.1	Hlavní činnosti a odpovědnosti	131

2.6.2	Softwarové profese	131
2.6.3	Organizace pracovních týmů	132
2.6.4	Řízení popisovaného autoprovozu	133
2.7	Pohled uživatele	134
3.	E-Shop – praktická aplikace metody BORM	137
3.1	Specifikace problému	137
3.2	Analýza aplikace	137
3.2.1	Interview	137
3.2.2	Scénáře a funkce	138
3.2.3	Sestavení diagramů podnikových procesů	141
3.3	Návrh aplikace	143
3.3.1	Návrhové vzory	143
3.3.2	Doménový model	145
3.3.3	Uživatelské rozhraní	148
3.3.4	Rozdělení aplikace	148
3.4	Implementace	151
Příloha 1.	Příručka nástrojů a technik BORMu	155
A.	OBA – Object Behaviour Analysis	156
B.	Procesy systému	158
B.1	Procesy a jejich účastníci	158
B.2	Procesní diagramy	160
C.	Od BO k CO	164
C.1	Pravidla přechodu k CO	164
C.2	Diagramy CO	165
D.	Od CO k SO	166
D.1	SO a jejich diagramy	167
D.2	Příklady postupné přeměny hierarchií v BORMu	167
D.3	Problémy se softwarovými objekty	170
E.	Další nástroje a techniky BORMu	173
E.1	Metoda funkčních bodů	173
E.2	Metamodelování	174
E.3	Návrhové vzory	177
Příloha 2.	Slovníček pojmů BORMu	179
Literatura	191
Rejstřík	193

O autorech

Autoři jsou členy organizačních a programových výborů konferencí Systémová integrace, Tvorbá softwaru a také konference Objekty – celostátní odborné konference věnované objektovým technologiím, kterou v roce 1996 založili.

Ing. Jiří Polák, CSc.

Po ukončení studií přednášel na FEL ČVUT v Praze, kde publikoval řadu článků, recenzí, skript a knih. Je členem několika odborných společností a ediční rady mezinárodního odborného časopisu Knowledge Based Systems. Na počátku 90. let se začal věnovat konzultační činnosti, dnes působí především v této oblasti, je partnerem firmy Deloitte & Touche CE a vede ve střední Evropě skupinu pro služby zákazníkům v sektoru pro informační technologie, telekomunikace a média. Příležitostně publikuje a přednáší v ČR i zahraničí.

Ing. Vojtěch Merunka, Ph.D.

Po ukončení studií začal pracovat na České zemědělské univerzitě, kde přednáší dodnes. Kromě toho vyučuje také na FEL ČVUT v Praze. Publikoval řadu článků a učebních textů, týkajících se objektového přístupu při tvorbě informačních systémů. Je členem několika odborných společností a redakčních rad různých konferencí a seminářů. Pobýval na přednáškových a výzkumných pobytech v zahraničí, především v USA a Velké Británii. Působí také jako konzultant v oboru zlepšení procesů a návrhu systémů.

Ing. Antonín Carda, CSc.

Pracoval nejprve jako konstruktér, ale brzy se začal věnovat informačním technologiím, kde působil jako vedoucí manažer projektů a vývojových týmů. Řídil rozsáhlé implementace informačních systémů a v řadě případů se také podílel na jejich návrhu. Publikoval více než dvě desítky odborných knih a skript a bezpočet odborných článků. Nyní několik let působí ve firmě Deloitte & Touche CE a přednáší na VŠE v Praze.

Předmluva

Moje generace, tedy lidé středního věku, se během své odborné kariéry stala svědkem i aktérem dramatických změn v oblasti tvorby informačních systémů – a to počínaje jejich vymezením a názvem. Bývaly časy, kdy jsme sami sebe nazývali programátory a přestože již tehdy se mnozí z nás zabývali něčím, co se dnes popisuje jako systémová analýza a návrh, nebyly tyto oblasti nijak ostře odděleny od vlastního psaní kódu. Zcela vážně jsme tehdy vedli debaty, zda je programování spíše vědou, nebo uměním.

Technický rozvoj a ekonomická realita poskytly na tuto otázku úplně jinou odpověď: ani věda, ani umění, je to inženýrská disciplína, tedy obor, který by měl být přiměřenou analogií strojního, stavebního či chemického inženýrství. K tomu patří vysoký stupeň formalizace, účinné a jednoznačné metodiky, významná role norem a normování všeho druhu. Inženýři nevymýšlejí nikdy nic od nuly: sestavují řešení podle přesného zadání z předem připravených a prověřených stavebních bloků.

Informatika je podle mého názoru dnes někde na své cestě k tomu, aby se stala vskutku plnohodnotnou inženýrskou disciplínou, za polovinou cesty. Jeden z nesčetného množství kroků tímto směrem představuje kniha, kterou máte v ruce. Je věnována technikám, postupům a metodikám obecného návrhu informačního systému, zhruba ve smyslu neklasičtější z klasických definic: „Systém je, abychom na něco nezapomněli.“ Autoři prošli svou životní cestu k teorii přes praxi, což je zdlouhavější, ale nadějnější než opačný postup; čtenáře své knihy vedou podobným způsobem.

Vedle souhrnu obecných názorů, poznatků, zkušeností a doporučení předkládají autoři i výsledek původní práce, svou vlastní metodu návrhu systémů BORM – Business Objects Relation Modelling. Knize to dodává jednak další rozměr, jednak je to více než dostatečný důkaz, že ti pánové vědí, o čem mluví. V kontextu české původní literatury jde o neobvyklý přístup, který jen zvyšuje hodnotu díla. Máte-li při své práci co do činění s informačními systémy, ať už jako tvůrce či kvalifikovaný uživatel, neměli byste tuto knihu odložit po přečtení předmluvy, ale mnohem, mnohem později.

Petr Koubský
šéfredaktor *Inside*
Vydavatelství Softwarové noviny, s. r. o.

Poděkování

Žádné dílo nevzniká ve vzduchoprázdnu. Pro tuto práci byl kromě jiného zapotřebí patřičný *impuls*, aby práce na díle skutečně započaly. Dále bylo nutné dostatečně silné přesvědčení autorů, že práci má smysl napsat a hlavně u rozdělané práce vytrvat. A konečně bylo třeba hotové dílo závěrečnými zásahy *upravit* do konečné podoby. Nápad knihu napsat přinesl Jiří Kofránek a byl to on, kdo nastartoval celou akci a jehož věty lze v textu nalézt. Proto mu patří naše zvláštní poděkování, a pokud za vámi přijde, a bude říkat, že jeho kniha už vyšla, tak mu prosím věřte.

Na rozvoji metody BORM pak spolupracoval Roger Knott. Kniha vycházela z nezpochybnitelného přesvědčení, že BORM představuje správný přístup. Na možnosti praktického ověření jeho účelnosti, účinnosti a správnosti má zásluhu společnost Deloitte & Touche a její projektanti. Jmenovitě to byli především Jaroslav Beneš, Štěpán Jaroš, Petr Kubát, Jiří Marzoll, Jakub Růžička, Ivan Šindelář a Michal Šindelář. Patří jim zvláštní uznání, neboť mají významnou zásluhu na vydání.

Řadu průběžně vznikajících pochybností pomáhali rozptýlit a některé dílčí problémy pomáhali vyřešit v průběhu inspirujících diskusí Martin Buldra, Pavel Drbal, Jiří Duben, Petr Koubský, Martin Lukáš, Martin Molhanec, Petr Štěpánek, Jaroslav Švasta i jiní. A zajímavá diplomová práce Petra Dvořáka se stala jádrem kapitoly o elektro-nickém obchodu e-Shop.

Onu únavnou a nezáživnou práci při závěrečných úpravách pomohl dokončit Miroslav Virius, a to celé bylo úspěšně završeno vstřícností nakladatele.

Je snadné něco tvořit ve fázi entuziasmu a euforie. Vytrvat ve fázích pochybností a překonat chvilkové tvůrčí deprese nám pomohli naši nejbližší.

Všem patří náš dík.

Jiří Polák, Vojtěch Merunka, Antonín Carda

Úvod

Tato publikace je věnována *životnímu cyklu systému*, počínaje vymezením systému v realitě a implementací informačního systému do reálného prostředí konče. Ponecháváme zde stranou skutečnost, že pro *informační systém* překvapivě neexistuje vlastně žádná všeobecně přijímaná rozumná definice¹, a spokojíme se s tím, že jej budeme považovat za něco, co obsahuje procesy na zpracování informací od jejich vstupu do systému po jejich výstup a že to všechno má nějaký účel. Pod spojením ‚systémový návrh‘ si představujeme celý životní cyklus systému, tedy ne ‚návrh systému‘ jako jednu etapu životního cyklu.

Životní cyklus *systému* se zasazuje do prostředí reálného světa, přičemž systému přisuzujeme cíle. A abychom se ve spleťtém systému vyznali, dělíme jej na části. Ty se vymezovaly podle různorodých pravidel, až jsme dospěli k dnešním *objektům a procesům* jako stavebním kamenům (popisu, modelování) systému.

Analýza, dedukce a dekompozice systému účelově pokračuje až do potřebné úrovně detailu, třeba elementární fyzikální částice pro astrofyzika nebo programové instrukce pro programátora. Odtud, velmi obecně řečeno, může začít syntéza, indukce, konstrukce systému nových vlastností jako postup, jež nakonec skončí opět v reálném okolí, z něhož původně vyšel. Což nebrání představě vývojové spirály, která vnáší do tohoto mentálního modelu stále kvalifikovanější, ale věčně opakovaní.

V průběhu minulých let vznikla a rozšířila se řada metod a nástrojů pro podporu analýzy, návrhu a realizace systémů – avšak podporujících jen dílčí etapy celého cyklu – a některé z nich jsou používány s velmi dobrými výsledky. O co se však autoři snaží zejména, je snaha po zdůvodnění potřeby komplexního přístupu k celému životnímu cyklu. Snaží se ukázat, že dílčí řešení nemusí být špatné, ale problém **umění systémového návrhu** spočívá ve schopnosti postihnout všechny fáze systémové práce od analýzy a vymezení systému po implementaci nového systému v jednom celku a při jednoduchém konzistentním popisu resp. modelu.

K popisu dynamiky chování systému jsou použity *procesy*. Pro ověření správnosti představ a požadavků jsou potom *procesy modelovány*. Klíčem správného postupu pro návrh systému je tedy koncept procesního pohledu na systém: systém je proces², jehož složky jsou dále podle potřeby členěny na menší a menší součásti – podprocesy, v případě potřeby až do úrovně elementů (resp. *objektů*) tak, jak jsou obvykle chápány třeba v objektovém programování. A to vše podle metody návrhu systémů BORM³, jak ji dva z autorů⁴ několik let vytvářeli, postupně ověřovali a zdokonalovali při řešení reálných problémů firem z nejrůznějších průmyslových sektorů.

Reálné procesy podnikového prostředí jsou základem k pochopení nejlepších průmyslových praktik při procesním řízení a jsou důležité pro definici potřebných změn procesů. Mezi takovými procesy je užitečné rozlišovat procesy hodnototvorné a procesy podpůrné. Hodnototvorné procesy jsou obsáhle v knize probírány, a to od jejich prvotního zachycení nebo návrhu, přes modelování jejich činnosti, až po jejich pozici v rámci řešeného systému

¹ Pro zajímavost zde uvedme definici použitou našimi zákonodárci: *Informačním systémem (se rozumí) funkční celek nebo jeho část, zabezpečující cílevědomou a systematickou informační činnost. Každý informační systém zahrnuje data, která jsou uspořádána tak, aby bylo možné jejich zpracování a zpřístupnění, a dále nástroje umožňující výkon informačních činností (zák. 365 /2000 Sb).*

² Systém je pak součástí procesů, nejčastěji procesů reálného světa.

³ BORM je zkratka pro Business Object Relation Modelling.

⁴ Jiří Polák, Vojtěch Merunka

a začlenění jako složky do procesů vyšší úrovně nebo rozčlenění procesů na složky v nižších úrovních. Toto všechno včetně metod a prostředků jejich formálního zobrazení a zajištění jejich informační podpory. Řadu čtenářů pravděpodobně nejvíce osloví právě srozumitelnost a jednoduchost popisu komplexních procesů a jistě si sami dobře představí využití tohoto aparátu i k jiným účelům, než je tvorba informačních systémů.

Kniha je rozdělena do tří částí. První představuje onu argumentační část, nezbytnou k pochopení jak nutnosti komplexního pohledu, tak i atributů takového komplexního pohledu. V určitých místech je výklad doplňován odbočkami buď k podrobnějšímu výkladu, teorie nebo k podrobnějšímu vymezení některých argumentů. Tyto části mají zřetelně odlišnou grafickou úpravu (svislá přerušovaná čára na okraji stránky). Snahou autorů bylo udržet souvislý výklad ve stejné úrovni a detaily uvést pro zájemce, nebo pro druhé čtení, do zřetelně označených míst se specifictějším zpracováním.

Důležitou složkou druhé části je úplný příklad, který je postaven na zmíněném principu – pohledu na objekty a procesy a který používá vhodných metod řešení, vycházejících z předchozí motivační části. Spolu s využitím grafických prostředků nabízí při průchodu řešením také paralelní výklad metody BORM, jež pak tvoří metodický nástroj řešení.

Třetí část obsahuje záměrně zvlášť umístěný příklad e-podnikání. Důvodem jeho zařazení bylo nejen zopakovat přístupy, prezentované ve druhé části, ale hlavně uvést ucelený, logicky uzavřený problémový okruh, podaný bez dalších doprovodných komentářů a přiblížit tak formu i obsah příkladu přiměřeného rozsahu skutečné praxi. Součástí knihy jsou také přílohy. Je to samostatná příručka, slovníček pojmů a stručný přehled, který lze odstříhnout od obálky a použít jako záložku. Další informace pak najdete také na webových stránkách nakladatelství Grada (<http://www.grada.cz/content/prikłady.php>).

Umění návrhu systému tedy spočívá podle autorů ve schopnosti správného *systémového pohledu* na realitu, správné volby *systémových nástrojů* pro různé etapy návrhu, správného postupu při analýze, návrhu a modelování komplikovaných systémových souvislostí, a to vše v rámci *odpovídající metody* jako kritéria i vodítka pro efektivní a účinnou vlastní práci. Autoři věří, že tuto roli může BORM jako poměrně stručná, velmi jasná a značně efektivní metoda sehrát.

K četbě této publikace tedy zveme ty, kteří chtějí znát principy správného návrhu systému a příslušných procesů. Zveme k četbě této publikace dále ty, kteří se chtějí naučit komplexní přístup a být schopni ho aplikovat ve své práci. Všem zájemcům z okruhu navrhovatelů systémů i jejich tvůrců pak publikace nabízí konfrontaci s jejich dosavadními zkušenostmi i názory, spolu se snahou nabídnout v řadě případů užitečný, efektivní a nový jednoduchý přístup.

1. Paralely s architekturou

1.1 Architekti a stavitelé informačních systémů

Naši knihu začínáme paralelou s jedním z nejstarších umění – umění budovat, tedy stavět – a přicházíme s konceptem architekta informačních systémů.

Velcí duchové minulosti často představovali v jedné osobě autora i interpreta, vynálezce i konstruktéra, architekta i stavitele. Zosobňují to namátkou Johann Sebastian Bach, Leonardo da Vinci či Petr Parléř. Unikátní díla těchto velikánů však nezůstala osamocená hlavně díky tomu, že se postupně lidská dovednost prohlubovala a vyžadovala si specializaci, samostatnou výchovu, vzdělávání a praxi. Objevují se skupiny věcně vázaných profesí, jako je *skladatel – hudebník a virtuos – dirigent* nebo *architekt – stavitel*.

Žasneme nad ohromujícími produkty lidského ducha, abychom vzápětí zjišťovali, jak omezené a nedokonalé je naše myšlení a poznání. Víme, že lidské myšlení začíná v těsné a nerozlučné závislosti na emocích a tuto asociaci si zachovává. Tak získává naše nazírání na svět dimenzi etiky a umění a tak se stává složkou kultury. „V běžném životě rozhodují hlavně emoce a fikce, pramenící z banálních struktur našeho archaického mozku“⁵ Potřebovali bychom znát formulaci obecného principu vzájemného strukturálního působení, který by vystihl vztahy mezi základními částmi a celkem. Tím by pomohl vysvětlit starou filozofickou poučku, že celek je víc než pouhý součet částí.

Velká složitost je špatně pochopitelná a obtížně mentálně zvládnutelná. Podpořit nás mohou mnohé pomůcky a triky. Ty však často selhávají při nejjednodušší úloze – jak navázat na velká díla alespoň metodicky. Jak postupovat při řešení velmi složitých věcí; jak postupovat při interpretaci komplikovaných idejí; jak postupovat při nezbytné diverzifikaci různých profesí při spojování jejich dílčích výsledků v jeden celek; jak postupovat při přenosu *know-how*. Mezi složitá díla patří též vybudování komplexního informačního systému nebo komplexních procesů. Je v těchto historických paralelách nástin řešení současných těžkostí? Od nepaměti se používaly plány, modely a účinné nástroje. Jsme tudíž při tvorbě informačních systémů na správné cestě?

Chceme zde, v samém úvodu knihy, vysvětlit, jak chápeme naši paralelu se stavitelem, jak nebezpečná zjednodušení a možné „všeléký“ ohrožují kýžené výsledky, a zdůvodnit, proč *model* a *abstrakce* jsou to, na čem stavíme dále.

1.1.1 Umění stavět

Stavitelé v minulosti představovali dvojedinou profesi: dokázali navrhnout, jak stavba bude vypadat, a zároveň ji uměli postavit. Petr Parléř, středověký stavitel, chrám sv. Víta v Praze vyprojektoval i zároveň řídil jeho stavbu.

Umění navrhovat i stavět budovy a chrámy tak, aby se nezřítily a aby i co nejlépe sloužily svému účelu, se často složitě předávalo z jednoho pokolení stavitelů na druhé. Předávání esotericky složitého *know how* o záladnostech středověkých staveb zřejmě přispělo i k prazákladům současného inženýrství a úcty k poznání obecně.⁶

⁵ Charvát, J.: Život, adaptace a stres, Avicenum, 1973.

⁶ První obor na Českém vysokém učení technickém v roce 1704 bylo právě stavitelství.

S rozvojem mechaniky, statiky, nauky o materiálech a dalších technických disciplín se profese stavitelů rozdělila. Jedni se věnovali vlastnímu navrhování vzhledu a vnitřnímu uspořádání staveb podle toho, k čemu měly sloužit, co se v nich bude dít a kdo to bude dělat, druzí zajišťovali jejich stavební realizaci podle již vytvořených plánů. Z jednoho obecného oboru stavitelů se vyvinulo povolání architekta a profese stavaře. Každý z nich má jiné cíle.

Pro architekta je velmi důležitý pohled uživatele – snaží se navrhnout takovou stavbu, která by co nejlépe splňovala jeho představy o funkci budovy a zároveň nepřesahovala jeho finanční možnosti. Funkce budovy je spoluurčena procesy, které v ní budou probíhat (např. vaření v kuchyni, praní v prádelně, mytí v kuchyni, jídlo v restauraci, spaní v pokojích). V architektonických soutěžích se utkává jeden návrh s návrhy jiných architektů a potenciálním uživatelům představuje stavbu nejen v nákresech a náčrtech, ale i v modelu (dnes někdy realizovaném i jako funkční třídídimenzionální model na počítači). Pokud architekt v soutěži uspěje, zpravidla též vykonává stavební dozor nad realizací stavby.

Stavař se snaží co možná nejlépe realizovat záměry architekta. Sestavuje rozpočet stavby a v průběhu stavby dbá o jeho dodržování. Řídí a realizuje stavbu a snaží se sladit práce tak, aby na sebe vhodně navazovaly. Využívá přitom i různé subdodavatele a snaží se skloubit jejich činnost. Dbá, aby jednotlivé stavební komponenty, nakupované od nejrůznějších dodavatelů, do sebe bez problémů zapadaly. Řeší různé komplikace, jako jsou skluzky dodávek, neočekávané zvýšení obtížnosti některých prací, pohyb cenové hladiny apod. Je v neustálém styku s architektem, s nímž řeší zejména případné změny projektu, vzniklé v důsledku komplikací nebo díky změně požadavků zadavatele. Vychází hodně z norem a standardů.

Obdobné změny v posledních letech prodělává i profese stavitelů informačních systémů. Stále více se zde vyhraňují dvě vzájemně spolupracující profese: návrháři architektury informačních systémů a vlastní realizátoři těchto návrhů.

Jsou zde však i podstatné rozdíly. Oddělení dvou profesí nepostoupilo k jejich škodě tak daleko, jako tomu bylo u stavitelů budov. Velmi často se etapy návrhu a realizace stále ještě vzájemně prolínají. A mnohem častěji než u staveb budov dochází během výstavby informačního systému k zásadním změnám projektů. Proč tomu tak je, je ostatně velmi zajímavá otázka. Pravidlem potom bývá překročení časových i finančních plánů a není výjimkou, že se započatá stavba informačního systému vůbec nedokončí. To se stane až v 50 % případů!

Pokud se systém dokončí, záhy přicházejí náročné požadavky na jeho údržbu. Díky rychlému pokroku v informačních technologiích se velmi brzo, a to mnohem dříve než u nové budovy, vynoří i nutnost jeho modernizace. A zde je opět velmi důležité mít podrobnou projektovou dokumentaci. Za absurdní bychom dnes považovali postavení velké, rozsáhlé budovy bez architektonického návrhu i její postavení a provoz bez náležité dokumentace. Přesto je tento stav u informačních systémů možný i dnes, a to i za situace, kdy všichni vědí, že rekonstrukce budovy je jev řídký, ale rekonstrukce informačního systému bývá na denním pořádku.

Zajímavý je i rozdíl ve způsobu zadávání zakázek. U stavby hotelu, obchodního centra či řešení obytné zástavby je zpravidla nejdříve vypisována architektonická soutěž a teprve po vypracování a schválení příslušného architektonického návrhu je vypisováno výběrové řízení na dodavatele stavby. Na dodávku informačních systémů je obvykle vyhlášováno jediné výběrové řízení, a to jak na návrh informačního systému, tak i na jeho realizaci. Což je právě ona velká příležitost pro ty, kteří sami sebe nazývají hrdým titulem „systémový integrátor“. Když si nevíme rady, často použijeme klišé, že „to ví bůh“. Za jakousi odrůdu *vševědoucích* se pak vydávají ti, kteří v informačních systémech všechno vědí, všechno umí, všemu rozumí a vše jim bude fungovat. Tito šamani lákají důvěřivé uživatele do pasti nefungujících systémů. Šamani vytvářejí vysoká očekávání, přitom realita je bohužel jiná.

1.1.2 Lákadla moderního šamana a skutečnost

Ty tam jsou doby, kdy počítače, ukryté za zdmi výpočetních středisek, byla tajemná poblíkávající monstra, chroupající sady dřevných štítků, s rachotem plivající stránky skládaného papíru z řádkových tiskáren. Tvorba programů byla tvůrčí činností, při které se cenilo umění maximálně využít omezené možnosti tehdejší technologie: schopnost vtěsnat vše potřebné do drahé a malé paměti a využít z dnešního pohledu liknavě loudavý procesor. Kromě toho se při tehdejšímu způsobu komunikace s počítačem na každý překlad a sestavení programu spotřebovalo neskonale více času a úsilí než dnes, a tak každá chyba při ladění programu byla dost drahá. To vyžadovalo velkou pečlivost. Programování muselo brát velké ohledy na hardware, na němž daná úloha běžela. A vzhledem k tomu, že možnosti bezprostřední interakce uživatele s počítačem byly omezené, architektura tehdejších systémů se více blížila architektuře stroje než potřebám uživatele⁷.

Dnes je všechno jinak... Místo psacích strojů trůní na kancelářských stolech osobní počítače. Bouřlivý vývoj přinesl zářivě barevné obrazovky monitorů, distribuované zpracování, intranety a Internet. Atributem pokroku jsou barnumské reklamy v počítačových časopisech i hrady triumfujících stánků počítačových firem na Comdexu, Cebitu či Inxexu a jiných výstavách. Tento obrovský rozvoj počítačových technologií ovšem přivedl na scénu týmy jak počítačových specialistů, tak informačních šamanů. Ať jsou to individua „na volné noze“, malé programátorské týmy či velké firmy „systémových integrátorů“, všichni soupeří o nemalé zakázky informačních technologií v podnicích, službách i státní správě⁸ a slibují a slibují...

Přes obrovský pokrok informačních technologií i reklamu a sliby tvůrců informačních systémů je však denní realita více než neuspokojivá. Umožňují snad informační systémy našich podniků *bezpapírový* chod? Mohou se manažeři spolehnout na informace, poskytované systémem na podporu rozhodování a nehledají podklady pro svou práci stále ještě ve výstupech účetnictví, mezd, saldokonta či skladové inventury? Nemusí každý občan vyplňovat v daňovém přiznání desítky údajů, které opakovaně jiným úřadům znova a znova sděluje? Nezredukovaly se naše zdravotnické systémy na výkazy pro zdravotní pojišťovny a kartotéku pacientů?

Kdekdou používá kancelářské balíky Microsoft Office, ale jen málokde se setkáme s propojením aplikací, třeba s využitím knihoven Visual Basicu. Místo toho se používá nepropojený Word, Excel a další součásti v „surovém stavu“.

V čem tedy spočívá, že informační technologie uživatelům nakonec nepřinášejí tak rychlý pokrok a efektivnost, jak původně slibovali jejich tvůrci a prodejci? Čím to je, že pracně vybudovaný informační systém, postavený na nejnovějších technologiích, nakonec nedělá vždy to, co si od něj uživatel sliboval, a počítač se tak stává spíše esoterickým lákadlem moderního šamana či nezbytným *módním doplňkem*, než užitečným nástrojem pro řešení konkrétních problémů?

⁷ To ovšem neznamená, že tehdejší v potu tváře naprogramované systémy byly nekvalitní. Řada z nich (obalená modernějším okolím a uživatelským rozhraním) přežila do roku 2000 a celý problém Y2K byl vlastně způsoben hodnotou dvoubajtové položky pro datum v diluviálním jádře fungujících systémů. O tom, že tyto s rozmyslem navržené systémy (programy) v podstatě uspokojovaly požadavky uživatelů, svědčí i to, že se pro ně za řádově dvě desetiletí provozu nenašla náhrada, což by v době vzniku jejich autoři jistě netušili. Když už systém funguje a uživatelé si na něj zvyknou, bývá málo důvodů to měnit.

⁸ Objem trhu informačních technologií je celosvětově asi 500 miliard USD, v ČR má tento trh velikost asi 60 miliard Kč ročně.

1.1.3 Když šaman selže

Vraťme se k naší paralele se stavebnictvím, neboť technologii budování informačních systémů lze v řadě rysů porovnat se stavbou budov. Při výstavbě domů se postupuje podle předem vypracovaných plánů. Ty zajistí, že dům nespadne a že bude sloužit tomu účelu, který požaduje jeho budoucí obyvatel či uživatel.

Stavba informačních systémů však nezřídka probíhá podle hrubého plánu, který se dotváří až během vlastní stavební činnosti. A tak se pak stane, že do některých místností je horší přístup, někam se musí zbytečně nahoru a dolů po schodišti a jinde zas chybějí okna či přebývají dveře.

Při stavbě domů se dnes běžně využívají standardizované komponenty: tvárnice, panely, standardizované nosníky, příčky, stavební hmoty. Stejně tak i stavitelé informačních systémů mají dnes k dispozici prefabrikované komponenty, připomeňme např. SAP, SSA aj. Problém s prefabrikáty je však v tom, že při stavbě informačních systémů se někdy nakoupí komponenty dřív, než je přesně známo, co a jak se použije a jaké procesy se tím mají podporovat. Navíc jádro nabídek stavitelů informačních systémů nezřídka spočívá právě v prodeji příslušného nástroje. Je to jako kdyby firma, nabízející rodinné domky, nabízela místo typových tvarů vilek ve svém katalogu seznam stavebních strojů, opentlený příznivými recenzemi renomovaných stavebních podniků.

Nejsou pak řídké případy, kdy hořící termíny uvedení informačního systému do provozu vede velké a bohaté infostavitele – kteří svou obratností dokázali sehnat velkou zakázku – k tomu, že se obracejí na skutečné experty a žádají je o pomoc. Díky snaze řešit kvalifikovaně tyto problémy vznikla profese systémových integrátorů.

1.1.4 „Systémový integrátor“, jeho metody a nástroje

Každá doba má některá nadužívaná slova, jejichž skutečný význam se pozvolna ztrácí za mlžným oparem znova a znova opakovaných frází. Mezi takovými postiženými termíny je zřejmě i dnes často používaný termín „systémový integrátor“. V původním slova smyslu by se za tímto termínem měl vlastně skrývat ten, kdo umí účelně vyhledat, provázat a propojit jednotlivé komponenty informačního systému do jednoho celku tak, aby uspokojil potřeby uživatele a podpořil jeho procesy. Obdobně jako stavař umí najít příslušné dodavatele oken, dveří a dalších stavebních komponent a postavit nebo rekonstruovat z nich budovu podle příslušného plánu architekta.

To, co jsme komentovali jako vhodnou dělbu práce a zvýšení profesní úrovně, se v organizacích *systémových integrátorů* znova sceluje. Systémový integrátor zhusťa nabízí jak profesi dovedných stavařů informačních systémů, tak disponuje ateliérem renomovaných architektů. Dosahuje se tak výhody bezprostřední spolupráce a týmové organizace na společném díle.

Jisté klady uvedené profesní struktury jsou však doprovázeny i zápory. K nevýhodám např. patří to, že při návrhu architektury řešení se často preferují ty nástroje, které má příslušný systémový integrátor k dispozici. Na ty si pořídil licence, zaškolil specialisty a dosáhl určité praktické zručnosti při jejich využití. Platí zde totiž jako všude jinde, že ne všechny nástroje jsou pro daný případ nejlepší, byť by byly *právě ony* těmi nejmodernějšími. Dostupná technologie nemusí být pro zadavatele, jeho potřeby a jeho informační systém optimální.

Je to podobné situaci, kdy jistá stavební firma vyhraje konkurs na stavbu kostela a shodou okolností tatáž firma vlastní závod na výrobu betonových panelů. Efektivní pracovní metody a výrobní postupy spočívají u této firmy na aplikaci panelové technologie výstavby. Ta je pro tuto firmu nejlevnější, nejrychlejší, nejmechanizovanější, nejdostupnější... Ne, že by nešlo konat mše v panelovém příbytku. Nicméně naše tradiční (a dodejme oprávněné) představy o konzervativnějším pojetí realizace svatostánku by měly mít přednost a rozhodující váhu!

K preferenci nových nástrojů však přispívá i to, že značná část zisků systémových integrátorů má původ v rabatech z prodeje hardwaru i softwaru, zprostředkovaného pro různé prodejce či subdodavatele. Je nabíledni, že propojování často i nesourodých prvků do funkčně spolehlivého celku vyžaduje kvalifikovaný přístup a průměrnou míru zkušeností. Je to ovšem i cesta, která může provokovat otázku: není to přístup *l'art pour l'art*, kdy si systémový integrátor vytváří image nepostradatelného specialisty, nad jehož technickým umem a řešitelskou brilancí odborníci žasnou? Faktem je, že se u systémových integrátorů investice do vlastního projektového týmu leckdy podceňovaly. Dnes se však situace obrací: nové zkušenosti uživatelů i nové technologie vedou k tomu, že jsou zvládnutelné nové, dosud nevidané požadavky na funkčnost informačních systémů.

Kvalita odvedené práce, metody a zkušenosti projekčního týmu při používání nástrojů se tak stávají jedním z klíčových prvků, který může do budoucna určit úspěšnost systémového integrátora na trhu.

1.1.5 Stavební stroje nestačí

Pokrok v informačních technologiích přináší řadu nástrojů a komponent, z nichž je možné informační systémy stavět. Existuje i řada účinných nástrojů a pomůcek, které návrh a údržbu informačních systémů podstatným způsobem usnadňují. Jsou to nástroje typu *Computer Aided Software Engineering (CASE)* apod. Přestože jsou tyto prostředky poměrně drahé, samy o sobě nestačí. Nejsou totiž ničím více ani ničím méně, než pouze nástroje. A tak jako pro stavbu dálnice nestačí mít sadu sebelepších stavebních strojů, nemáme-li zároveň lidi, kteří je umí dokonale ovládat, a podrobný stavební plán, tak ani sebedražším nástrojem CASE bez příslušných znalostí žádný systém nepostavíme.

To však není důvod jediný. Je zde ještě další problém, který představuje jednu z hlavních, klíčových překážek rychlé, efektivní, účinné a účelné tvorby pružných informačních systémů. Je to problém pochopení, co je vlastně pro budoucího uživatele tohoto systému důležité, co je vlastně zapotřebí udělat. Je zapotřebí *pochopit problém uživatele, pochopit jeho procesy a umět přeložit specifikaci jeho potřeb z jazyka uživatele do jazyka tvůrce a realizátora informačního systému*. A tady bohužel nemohou pomoci významnou měrou žádné nástroje CASE ani jiné podobné prostředky. Naznačili jsme již, že povaha potíží spočívá v *komunikaci* uživatele a tvůrce. Tyto dvě zúčastněné strany mají totiž každá jinou kvalifikaci, jiné zázemí, jiné vzdělání a zkušenosti i *jiné podpůrné nástroje pro svou práci*.

Nástin situace je tento: existuje budoucí uživatel nového nebo inovovaného informačního systému. Tento uživatel má problém, jehož podstatu nemusí ani sám chápat (!), může řídit procesy, jimž ne zcela rozumí (!!), a to se snaží pro sebe přijatelným způsobem a pomocí jemu známých a dostupných komunikačních technik sdělit tvůrci informačního systému (!!!) spolu s (ne)vyjádřenou nadějí, že řešení bude *omnipotentně* odstraňovat i řadu nevyřešených potíží, které současný stav provázejí⁹.

Jde tedy o schopnost *přemostit* propast mezi světem uživatelů a světem tvůrců informačních systémů, mezi něž samozřejmě patří též uživatelé nástrojů CASE a programátoři. A právě na překlenutí této mezery mezi reálným světem uživatelů a světem počítačů, čili na uchopení světa uživatele a jeho převedení do *vidění a chápání* počítače, jsou zaměřeny metody návrhu systémů, které jsou pracovními postupy nové profese – **architektů informačních systémů**. Jim je určena tato kniha.

⁹ Ona *barokně koncipovaná* věta chce přiblížit dosavadní stav, kdy uživatel často neví, co chce, z toho odvodí, co potřebuje, sdělí to špatným způsobem a věří, že řešení toho všeho ho zbaví starostí!

Souhrnně

Umění stavět je dnes dílo mnoha profesí, z nichž vybíráme architekta a stavitele. Paralela k výstavbě informačních systémů vede ke konceptu architekta informačních systémů a ukazuje, že generální dodavatel/systémový integrátor se nedá hodnotit podle nástrojů, které používá, ani podle toho, kolik je schopen naslibovat, ale jak využívá modelů, jak je zorganizován a co jeho lidé umí, co vše dokázali.

1.2 Model procesů a informačního systému

Architekti a stavitelé si porozumí proto, že se domlouvají pomocí modelů. Vzhledem k neschopnosti řešit myšlenkové procesy o mnoha prvcích saháme od nepaměti k intuitivní pomůcce: členění složitého celku na zvládnutelný počet menších částí a vyhledávání vztahů a vazeb mezi nimi. Toto zpracování vzájemného působení nám jde zřetelně lépe, než spekulace a přemýšlení o větším počtu objektů najednou¹⁰. To je první motivace, kde od reality přecházíme k jejímu modelu dělením na části.

O Archimédově projektu na zapálení nepřátelského loďstva víme z dostupných písemností. Nad da Vinciho náčrty létajících strojů jsme začali žasnout po vzlétnutí prvních helikoptér a naše schopnosti jsou nedostatečné při pokusu interpretovat dílo Nicollo Paganiniho. O těchto výbojích lidského ducha se dozvídáme ze zvláštní formy záznamů procesů tvořivého myšlení – modelů a simulací. Jsou to pomůcky, jak zkoumat budoucí možné jevy nebo situace, jak odhalovat skryté zákonitosti skutečných procesů, jejichž přímé pozorování však může být nepřiměřeně nákladné, riskantní nebo dokonce nemožné. Jsou to namátkou procesy v nitru Země či Slunce, tektonická aktivita na Venuši, průběh destrukce prostředí při atomovém výbuchu nebo deformace automobilu při čelním nárazu do bariéry.

Vyčerpávající představu o složitém informačním systému, který neposkytuje očekávané výsledky nebo který vyžaduje přestavbu nebo který není dokonce ještě realizován, nezískáme většinou jeho slovním, byť dokonalým vylíčením. Již v historii nalézáme poučení, že pro tyto práce, postavené na abstraktním myšlení, jsou výhodné modely a simulace. Abychom mohli něco simulovat, musíme mít model. Model však obvykle představuje velmi silnou abstrakci reality. Model sídliště je obvykle dřevěnou maketou ve velmi malém měřítku a se zanedbáním obrovského množství pro danou situaci nepodstatných detailů. Bližší realitě jsou funkční *polotovary*, které sice nahrazují skutečně realizovaný produkt, avšak jsou též výrazně jednodušší – jsou to funkční prototypy.

1.2.1 Abstraktní myšlení a modely

Jedním z často užívaných pojmů v této knize je „model“. Tento pojem známe např. z architektury, kde modely projektovaných staveb slouží k názorné prezentaci vytvořeného díla. Jiný význam má „model“ pro malíře či sochaře, kde je jakousi „předlohou“, „reálným vzorem“ pro umělcovu kreaci. Jako „model“ však můžeme označit též notový part, sloužící jako „návod“, jak interpretovat skladatelovu invenci a představu. Za „model“ přípravy určitého jídla bychom pak mohli vydávat též třeba kuchařský předpis nebo za „model“ krajiny můžeme považovat příslušný výsek mapy.

¹⁰ Tento počet je překvapivě nízký a představuje *pět plus minus dva objekty*, s nimiž jsme schopni provádět myšlenkové operace, řešit mentálně najednou obsáhlé systémové úlohy. Každý větší počet částí vede k provedení nezbytné agregace a postupné, obvykle hierarchické dekompozici podobné úlohy.

Modelování chápeme většinou jako vytvoření nebo odhalení určitého jasně vymezeného druhu podobnosti mezi systémy pomocí dvojice systémů: modelovaného a modelujícího systému. Říká se též, že modelující systém je „modelem“ druhého systému. Z naznačeného je zřejmé, že existuje celá řada různých významů pojmu „model“. Je to model logický, sémantický, strukturně syntaktický, model jako isomorfismus, jako hypotéza, abstrakce, idealizace, matematická konstrukce, formalizovaný systém, atp. Většinou se přitom jedná buď o myšlenkově nebo prakticky vytvořené struktury, které reprodukují určitou stránku nebo část skutečnosti v idealizované a názorné formě. Nebo jde o reprodukci určité oblasti jevů za pomoci jiné, lépe prostudované nebo zvládnuté oblasti: nepoznané se vysvětluje poznaným. Jde tedy také o jistou *analogii* jako vztah podobnosti systémů.

Z uvedeného plyne, že modelování je jakousi *myšlenkovou imitací, abstrakcí, reprodukcí* reálně existujícího systému pomocí speciálně konstruovaných modelů – analogů. Modelování je tedy jednou z forem poznání, zvláštním prostředkem reprodukce reality. Vztah modelu a objektu je vztahem shody, ne totožnosti. Model je v určitém vztahu *podobný* objektu, v jiných vztazích se však od něj nutně *liší*. A právě proto, že se model od originálu liší, vzniká možnost vydělit určité podstatné vztahy, snáze měnit podmínky, hledat určitá řešení, tedy pracovat s modelem tak, jak je s originálem nemožné. Je-li model „příliš přesný“, ztrácí svůj smysl, je-li příliš nedokonalý, může být zdrojem chyb a zkruslovat docílené výsledky.

V širším slova smyslu je tedy modelem každá teorie a každá hypotéza. Model by měl umožnit odvodit chování modelovaného objektu, tj. jinak řečeno odpovědět na otázku „co se stane, když“. Modelování v tomto širším pojetí pak vlastně užívá každý, kdo se zabývá experimentálním výzkumem. Na základě znalostí o zkoumaných objektech badatel dedukuje, jak se bude chovat příslušný objekt, je-li hypotéza, teorie (model) platná.

Platnost teorie se verifikuje experimentem, resp. pozorováním, měřením, kdy se zjišťuje chování originálu. Pokud jsou výsledky v rozporu s tím, co bylo možné očekávat na základě dosud platné teorie (tj. chování modelu je v rozporu s chováním originálu) pak je nutno teorii (tj. dosud platný model) zamítnout a přepracovat.

Zkoumání reality je možné myšlenkovými postupy, které lze rozdělit do dvou skupin: postupy deduktivní a postupy induktivní. U *deduktivních* postupů závěr vyplývá z výchozích premis nebo je vyvozen jako nepochybný s jistotou. V empirických vědách převládají *induktivní* postupy. Při nich se shromažďuje empirický materiál, na jehož základě se vytvářejí obecné teorie, hypotézy. Při induktivním postupu výsledek nutně neplyne z výchozích vět či zjištěných tvrzení, takže výsledek, vzniklý induktivním postupem na pravdivých faktech, může být pravdivý i nepravdivý.

Dedukci je možno provádět přesněji, je-li teorie (tj. v našem pojetí model) vyjádřena pomocí formalizovaného jazyka (např. jazyka matematiky)¹¹. Pak je tyto dedukce možné provádět na základě pravidel daného jazyka (např. řešením příslušných rovnic).

¹¹ Využívání formalizovaného popisu se u jednotlivých vědních disciplín liší. Jestliže např. ve fyzice je vyjadřování teorie pomocí rovnic běžné, ve společenských vědách (s výjimkou ekonomie) je to však vzácností. Proces formalizace jednotlivých vědních disciplín je spojen s vytvořením adekvátního formálního jazyka po sémantické i syntaktické stránce. Tento proces proběhl u fyzikálních věd v podstatě převzetím jazyka matematiky a vývojem obou věd (matematiky a fyziky) ve vzájemné vazbě již v 18. a v první polovině 19. století, u chemie a ekonomie formalizace začíná v devatenáctém a vrcholí kolem poloviny dvacátého století. U společenských a biologických věd (např. management nebo lékařství) je proces formalizace (z důvodů složitosti a komplexnosti systémů) relativně opožděn a přichází až s kybernetikou a výpočetní technikou.