DOI: 10.35191/medsoft_2026_1_38_merunka

OBJECT-PROCESS METHODOLOGY

Vojtěch Merunka[1], Jiří Kofránek, Pavel Beránek

Abstrakt

Object-Process Methodology (OPM) je metodika konceptuálního modelování, která byla v roce 2024 přijata jako mezinárodní norma ISO 19450. Její hlavní předností je možnost zachytit strukturu, chování i stavy modelovaných objektů v jediném typu diagramu. Každému diagramu navíc odpovídá rovnocenný textový zápis ve strukturovaném přirozeném jazyce, který je srozumitelný odborníkům a současně umožňuje automatizované zpracování. Vytvořený model lze také spustit a jeho chování ověřovat pomocí simulace. Model se tak stává nejen prostředkem komunikace, ale také nástrojem pro kontrolu správnosti a úplnosti zadání i pro ověřování výstupů velkých jazykových modelů. Praktické využití OPM ukazujeme na příkladu elektronické žádanky, jejíž konceptuální model odhaluje podstatné nedostatky právní úpravy.

Klíčová slova

Object-Process Methodology, OPM, ISO 19450, konceptuální modelování, elektronická žádanka, eHealth, Modelica

1 Úvod

Každý, kdo se v České republice podílel na zavádění centrálních zdravotnických informačních systémů, tento postup dobře zná. Zadání zpravidla vzniká jako právní předpis, metodický pokyn nebo technická specifikace a následně prochází připomínkovým řízením, do kterého se zapojují desítky odborníků. Každý se přitom soustředí především na oblast, které nejlépe rozumí. Teprve během provozu systému se pak může ukázat, že v některé části procesu chybí důležitý krok, který žádné z předchozích kol připomínkování neodhalilo.

Příčinou nemusí být nedostatek pečlivosti, ale samotný způsob, jakým je zadání vyjádřeno. Běžný slovní popis totiž neumožňuje snadno posoudit, zda je proces úplný a všechny jeho části na sebe správně navazují. Například tvrzení, že se„žádanka elektronicky předává poskytovateli", může být správné, aniž by z něj bylo zřejmé, co předání předchází, co po něm následuje a jak má celý proces skončit. Text tak může působit jednoznačně a úplně, přestože popisuje pouze část požadovaného chování systému.

Právě tento problém pomáhá řešit metodika Object-Process Methodology (OPM), která byla v roce 2024 přijata jako mezinárodní norma ISO 19450 [1]. Umožňuje zachytit strukturu, chování i stavy objektů v jediném typu diagramu označovaném jako Object-Process Diagram (OPD). Každému diagramu odpovídá rovnocenný textový zápis ve strukturovaném přirozeném jazyce Object-Process Language (OPL), který lze z diagramu automaticky generovat. Obě podoby modelu tak umožňují jeho společné čtení a kontrolu odborníky z různých oblastí [2,3].

Významnou předností OPM je také možnost vytvořený model spustit a simulací ověřovat jeho chování. Například prostředí OPCloud umožňuje sledovat, zda jednotlivé procesy vedou k očekávaným výsledkům, zda jsou požadované stavy dosažitelné a zda v modelu nechybějí důležité návaznosti [4]. Konceptuální model se tak stává nejen prostředkem komunikace mezi odborníky, ale také nástrojem pro ověřování správnosti a úplnosti zadání.

[1] Vojtěch Merunka je autorem původního konceptu tohoto článku, ze kterého text vychází, i prací, o nichž pojednávají kapitoly 4 a 10. Ze zdravotních důvodů se nemohl podílet na dokončení textu; za jeho konečnou podobu odpovídají ostatní autoři.

2 Metodika OPM

Cílem této kapitoly není naučit čtenáře vytvářet diagramy OPM, ale především jim porozumět. Základní principy metodiky proto představíme na příkladu elektronické žádanky, který využijeme i v dalších částech článku.

2.1 Tři základní pojmy OPM

Jednou z hlavních předností OPM je jednoduchost základních stavebních prvků. Pro pochopení metodiky stačí zpočátku rozlišovat tři pojmy, kterými jsou objekt, proces a stav [1–3,5].

Objekt představuje něco, co existuje a má svou identitu. Ve zdravotnictví může jít například o žádanku, pacienta, laboratoř nebo výsledek vyšetření. V diagramu se znázorňuje obdélníkem.

Proces představuje činnost nebo děj, při kterém objekty vznikají, mění se nebo zanikají. Příkladem je vystavení, odeslání nebo vyhodnocení žádanky. Proces se znázorňuje elipsou.

Stav vyjadřuje, v jaké situaci se objekt právě nachází. Elektronická žádanka může být například vystavená, odeslaná, přijatá nebo vyhodnocená. Stavy se zobrazují uvnitř obdélníku příslušného objektu.

Jednoduchost těchto základních prvků umožňuje pracovat s modelem také odborníkům bez znalosti formálních modelovacích jazyků. Lékař tak může diagram nejen číst, ale také posuzovat jeho věcnou správnost a navrhovat úpravy. To je důležitý rozdíl oproti notacím, které vyžadují znalost většího množství symbolů a pravidel jejich použití.

2.2 Bimodálnost jako propojení diagramu a textu

Jednou z hlavních vlastností OPM je bimodálnost, tedy možnost vyjádřit stejný model graficky i textově. Grafickou podobu představuje Object-Process Diagram (OPD), zatímco textovou podobu tvoří Object-Process Language (OPL), strukturovaný přirozený jazyk s přesně definovanými pravidly (obr. 1).

Textový zápis není pouhým komentářem k diagramu, ale jeho rovnocennou reprezentací, kterou lze automaticky generovat. Vztahy zachycené v diagramu tak mají odpovídající vyjádření ve větách OPL.

Bimodálnost má dva důležité praktické přínosy. Lékař nebo jiný odborník může model kontrolovat prostřednictvím srozumitelných vět, aniž by musel podrobně ovládat grafickou notaci. Současně lze formálně definovanou strukturu modelu automatizovaně zpracovávat, což otevírá možnosti jejího využití například při ověřování výstupů velkých jazykových modelů, kterému se věnujeme v kapitole 6.

2.3 Řízení složitosti pomocí postupného zpřesňování modelu

Složité zdravotnické procesy zpravidla nelze přehledně zachytit v jediném diagramu. OPM proto umožňuje postupné zpřesňování modelu prostřednictvím mechanismu označovaného jako in-zooming. Vybraný proces lze rozvinout a zobrazit jeho vnitřní strukturu v podobě navazujících podprocesů (obr. 2).

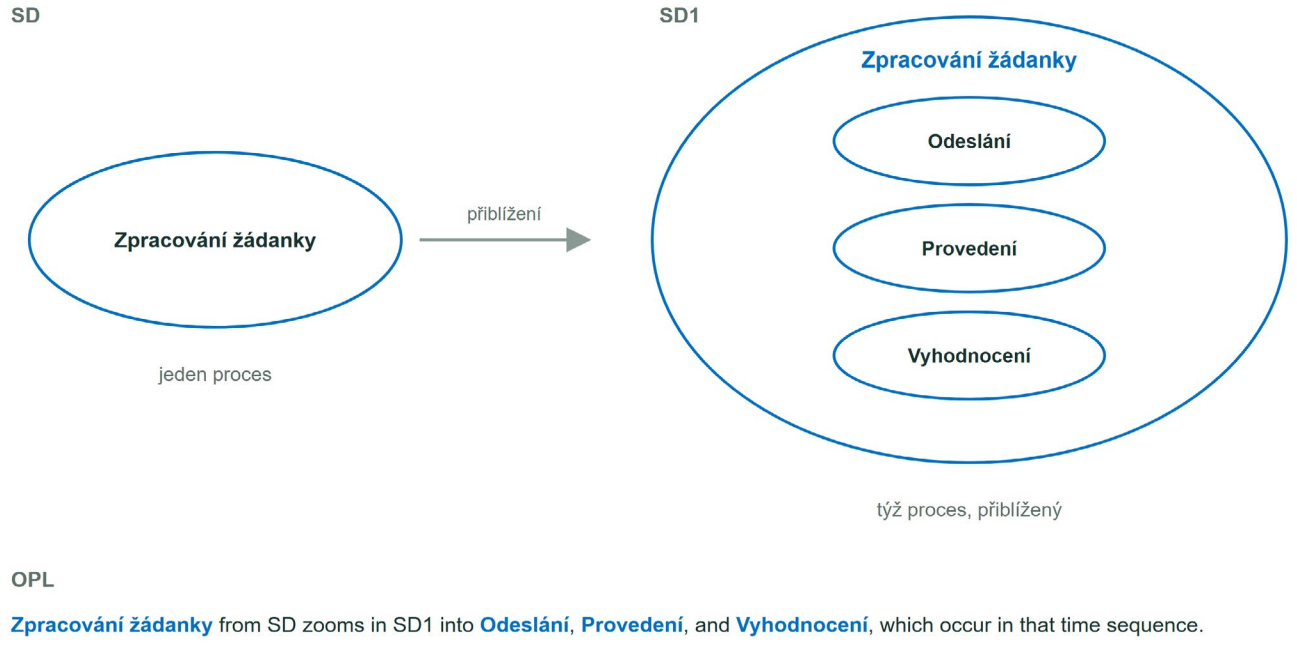
Při tomto zpřesňování není nutné přecházet na jiný typ diagramu. Jednotlivé úrovně podrobnosti zůstávají součástí jednoho provázaného modelu, který lze prohlížet od celkového pohledu na systém až po konkrétní činnosti.

Tento princip usnadňuje práci s rozsáhlými systémy, u nichž bývá obtížné udržovat soulad mezi množstvím samostatných diagramů. OPM umožňuje soustředit se na vybranou část systému, aniž by se ztratila její návaznost na celek.

DOI: 10.35191/medsoft_2026_1_38_merunka



Obrázek 1 – bimodálnost OPM: týž model elektronické žádanky jako diagram (OPD) a jako věty v omezené přirozené řeči (OPL). Text není komentář k obrázku, ale jeho rovnocenná podoba – věty vygeneroval nástroj z diagramu, proto jsou vyhrazená slova anglicky a jména pocházejí z modelu. Objekty jsou zeleně, procesy modře, stavy objektu leží uvnitř jeho obdélníku a jsou olivové. Stín u objektu Lékař znamená, že je fyzický; plné kolečko na konci spojnice říká, kdo proces provádí.



Obrázek 2 – řízení složitosti přibližováním: proces otevřený do podprocesů, ze kterých se skládá. Hierarchie nevzniká přechodem na jiný druh diagramu. Svislé pořadí podprocesů je jejich posloupnost.

2.4 Propojení stavů objektů s procesy, které je mění

Pro náš článek je zvláště důležitá schopnost OPM zachycovat stavy objektů společně s procesy, které je mění. Struktura systému, jeho chování a stavy objektů tak tvoří jeden vzájemně propojený model.

V případě elektronické žádanky můžeme sledovat nejen její vystavení, odeslání nebo přijetí, ale také stav, ve kterém se nachází před každým procesem a po jeho dokončení. Díky tomu lze posuzovat návaznost jednotlivých činností i úplnost celého životního cyklu žádanky.

Z takového modelu vyplývají dvě zásadní otázky. Lze dosáhnout každého definovaného stavu objektu? Pokud některý stav není dosažitelný, může v modelu chybět proces, který do něj vede, nebo je samotný stav nadbytečný. Lze z každého stavu pokračovat do další předpokládané fáze? Pokud ne, je třeba posoudit, zda jde o zamýšlené ukončení procesu, nebo o nedostatek modelu.

Právě možnost ověřovat tyto návaznosti představuje významný přínos OPM. Umožňuje odhalovat neúplnosti, které mohou v běžném textovém zadání zůstat skryté. Praktický význam této vlastnosti ukážeme při analýze životního cyklu elektronické žádanky.

3 Původ metodiky OPM

Metodika OPM vznikla v oblasti systémového inženýrství, které se zabývá návrhem složitých systémů propojujících fyzické, technické a informační procesy. Jejím autorem je profesor Dov Dori z izraelského technologického institutu Technion, který ji rozvíjí od devadesátých let minulého století [2]. Cílem OPM bylo od počátku vytvořit jednotný způsob zachycení struktury a chování systémů, jejichž součásti nelze posuzovat odděleně.

DOI: 10.35191/medsoft_2026_1_38_merunka

V roce 2015 byla metodika vydána jako veřejně dostupná specifikace ISO/PAS 19450:2015. Po dalších devíti letech vývoje ji v lednu 2024 nahradila plnohodnotná mezinárodní norma ISO 19450:2024. V březnu 2026 byla převzata také do české soustavy technických norem jako ČSN ISO 19450 [5]. Její standardizace má praktický význam zejména pro veřejnou správu, protože umožňuje odkazovat na jednotně definovanou metodiku také v zadávací dokumentaci veřejných zakázek při dodržení příslušných pravidel veřejného zadávání.

OPM má vazby také na české akademické prostředí. Profesor Dov Dori přednášel o metodice v Praze přibližně sedmdesáti studentům anglického studijního programu na univerzitě jednoho z autorů tohoto článku. Navázaná spolupráce pokračuje prostřednictvím odborných konzultací s jeho výzkumnou skupinou, mimo jiné nad otázkami, kterým se věnujeme v kapitole 8.

Ve světě se OPM využívá především v systémovém inženýrství, leteckém průmyslu a obranných technologiích. Casebolt, Jbara a Dori popsali v roce 2020 v časopise Systems Engineering metodu zlepšování procesů založenou na ISO 19450 [6]. Její použití ukázali na postupu vyjmutí a opětovné montáže součásti letadla a na řízení kvality u amerického výrobce letecké techniky. Další studie se zabývaly modelováním operační architektury podle rámce DoDAF a systémů protiraketové obrany [7]. Starší materiály autora metodiky uvádějí také zkušební využití ve společnostech National Semiconductor, Ford Motor Company a u dodavatele společnosti Lockheed. Tyto případy jsou však doloženy převážně samotným autorem, a nelze je proto považovat za nezávislé potvrzení širokého průmyslového rozšíření OPM.

Pro zdravotnictví je zajímavé zejména využití OPM v molekulární biologii. Dvě studie publikované v časopise PLOS ONE se zabývaly životním cyklem mRNA [8]a jejím transkripčním cyklem [9]. Vytvořené modely pomohly nejen uspořádat známé poznatky, ale také odhalit neúplně popsané vztahy a mezery v porozumění biologickým procesům. Právě schopnost odhalovat chybějící souvislosti představuje jeden z hlavních přínosů, které chceme využít také při modelování elektronické žádanky.

Ve zdravotnické informatice je použití OPM zatím méně rozšířené. Významným příkladem je studie z roku 2009 v časopise Journal of Biomedical Informatics [10], která rozšířila OPM o modelování výjimek a předvedla tento přístup na klinickém doporučeném postupu pro antibiotickou léčbu. Současně upozornila na tehdejší omezení metodiky při zachycování asynchronních událostí. V modelování zdravotnických procesů nadále převažuje BPMN a nezávislých aplikací OPM je v literatuře poměrně málo. Uvedené studie navíc spojuje spoluautorství Dova Doriho, což je při hodnocení jejich výsledků nutné zohlednit.

Zájemcům o OPM jsou prostřednictvím platformy edX dostupné dva kurzy Technionu zaměřené na základní a pokročilé modelování. Pro praktickou práci existuje několik nástrojů, zejména původní desktopový OPCAT, webový OPCloud, ve kterém vznikly diagramy tohoto článku, a otevřený OpenOPM Editor, na jehož vývoji se podílí jeden z autorů [4,11]. Ve srovnání s UML nebo BPMN je však nabídka nástrojů výrazně menší, což představuje jednu z překážek širšího využití metodiky.

OPM tak spojuje dlouhodobý odborný vývoj, mezinárodní standardizaci a doložené příklady praktického využití. Její rozšíření ve zdravotnictví však zůstává omezené. Proto považujeme za důležité hodnotit ji především podle konkrétních problémů, které může pomoci řešit při návrhu a ověřování zdravotnických informačních systémů.

4 Porovnání OPM s ostatními přístupy k modelování

OPM není jedinou metodikou pro modelování zdravotnických informačních systémů. V praxi se používá několik zavedených notací, které se liší způsobem zachycení struktury, chování a stavů systému. Pro naše účely je důležité především to, zda umožňují sledovat životní cyklus konkrétního objektu, zda jsou srozumitelné zdravotnickým odborníkům a zda podporují ověřování správnosti a úplnosti modelu. Základní srovnání uvádí tabulka 1.

Unified Modeling Language (UML) je zavedeným standardem softwarového inženýrství, který umožňuje podrobně popisovat různé aspekty systému [12]. Strukturu zachycuje například diagram tříd a komponent, chování zachycují sekvenční a stavové diagramy a požadavky diagramy případů užití. Nevýhodou je nutnost udržovat soulad mezi těmito pohledy, který samotná notace nezaručuje. UML navíc vyžaduje znalost specifických symbolů a pravidel, takže pro lékaře bez zkušeností s modelováním nemusí být snadno srozumitelný.

Systems Modeling Language (SysML) rozšiřuje možnosti modelování na technické a fyzické systémy a jejich požadavky [13]. Podporuje parametrické vztahy i propojení se simulačními nástroji, například s jazykem Modelica [14]. Zatímco SysML v1 využívá devět typů diagramů, novější SysML v2 zavádí také standardizovanou textovou syntaxi [15]. Ta je však určena především systémovým inženýrům, zatímco OPM nabízí textový zápis OPL založený na srozumitelných větných konstrukcích.

Business Process Model and Notation (BPMN) patří mezi nejrozšířenější notace pro modelování organizačních a zdravotnických procesů. Má rozsáhlou softwarovou podporu a umožňuje také simulaci nebo vykonávání procesů. Její hlavní předností je přehledné zachycení posloupnosti činností, rozhodování a předávání informací mezi účastníky [16]. Přestože BPMN umožňuje pracovat s datovými objekty a jejich stavy, soustředí se především na průběh činností. Například vystavení, odeslání, přijetí a vyhodnocení elektronické žádanky lze znázornit jako posloupnost úloh, aniž by bylo na první pohled zřejmé, že jde o změny stavů jediného objektu. OPM naproti tomu staví objekt a jeho životní cyklus přímo do středu modelu.

metodika	typů diagramů	struktura a chování v jednom	stavy objektu	text = diagram	řízení složitosti	Simulace	norma
OPM	1	Ano	ano	ano (OPL)	přibližování	konceptuální; u nás i spojitá	ISO 19450
UML	14	ne, dělí se	ano (stavový diagram)	ne	balíčky	jen v nástrojích	OMG
SysML	9	částečně	ano	ne	bloky	některé nástroje	OMG
BPMN	1 (+2)	ne – jen proces	ne	ne	podprocesy	ano, hojně	OMG, ISO 19510
BORM	1	Ano	ano	ne	částečně	ano (Craft.CASE)	ne
ArchiMate	1 (vrstvený)	částečně	ne	ne	vrstvy	ne	Open Group
Petriho sítě	1	ne – jen chování	značky	ano (formálně)	hierarchické sítě	ano, důkazově	ISO/IEC 15909

Tabulka 1 – porovnání OPM s běžně používanými notacemi.

DOI: 10.35191/medsoft_2026_1_38_merunka

Zvláštní pozornost si zaslouží **Business Object Relation Modeling** (BORM), se kterým máme vlastní zkušenosti při modelování zdravotnických procesů [17]. V předchozím článku [18] jsme popsali jeho využití při specifikaci elektronického receptu. Společných pracovních setkání se účastnili odborníci z různých oblastí včetně zástupců České lékařské komory. Ukázalo se, že vizuální modely objektů, jejich stavů a vzájemných interakcí jsou srozumitelné i účastníkům bez znalosti softwarového inženýrství. Simulace navíc odhalila objekty, které čekali na událost, jež v modelu nemohla nastat.

Přechod od BORM k OPM proto není motivován nedostatečnými schopnostmi původní metodiky. Oba přístupy umožňují modelovat objekty, jejich stavy a chování. Hlavní výhody OPM spočívají v mezinárodní standardizaci podle ISO 19450, rovnocenné textové reprezentaci OPL a možnosti vytvářet nezávislé nástroje podporující stejnou normu [1,3]. Zatímco naše zkušenost s BORM byla spojena zejména s proprietárním prostředím Craft.CASE, OPM poskytuje standardizovaný základ pro další vývoj a automatizované zpracování modelů.

ArchiMate a **ARIS** se zaměřují především na architekturu organizací, jejich procesů a informačních systémů [19,20]. Jsou vhodné pro plánování rozsáhlých informačních prostředí a zachycení vztahů mezi organizačními, aplikačními a technologickými prvky. Jejich hlavním cílem však není ověřování životního cyklu konkrétního objektu pomocí simulace, takže mohou OPM vhodně doplňovat.

Petriho sítě představují matematicky přesně definovaný přístup k modelování souběžných procesů [21]. Umožňují formálně analyzovat dosažitelnost stavů a za stanovených podmínek také dokazovat nepřítomnost uváznutí. Jejich nevýhodou je vyšší náročnost na znalost formalismu, která omezuje jejich srozumitelnost pro zdravotnické odborníky.

OPM tedy nepředstavuje univerzální náhradu ostatních metodik. UML a SysML jsou vhodné pro podrobný návrh technických a softwarových systémů, BPMN pro modelování pracovních postupů, ArchiMate pro podnikovou architekturu a Petriho sítě pro formální analýzu. Hlavní předností OPM je propojení srozumitelného konceptuálního modelu s možností ověřovat jeho chování, zejména v počátečních fázích návrhu, kdy se teprve upřesňují požadavky a hledá společná představa o fungování systému.

Pro zdravotnictví je důležitá také schopnost OPM zachytit v jediném modelu softwarové, organizační i fyzické součásti systému [1]. Metodika přitom rozlišuje fyzické a informační prvky i součásti samotného systému a jeho okolí. Pacient, laboratorní vzorek nebo zdravotnický přístroj tak mohou být modelovány společně s elektronickou žádankou, centrálním registrem a procesy, které je propojují. Právě možnost zachytit vztahy mezi skutečným poskytováním zdravotní péče a jeho informační podporou považujeme za jeden z hlavních důvodů pro využití OPM. Praktický význam tohoto přístupu ukážeme v následující kapitole při analýze elektronické žádanky.

5 Elektronická žádanka v OPM

Možnosti OPM nyní ukážeme na příkladu elektronické žádanky (eŽádanky), která představuje jednu z důležitých součástí digitalizace českého zdravotnictví. Umožňuje elektronické vystavování a předávání požadavků na odborná vyšetření a další zdravotní služby. Její zavádění vyžaduje propojení právních požadavků, klinických postupů a informačních technologií, a proto je vhodným příkladem pro konceptuální modelování. Naším cílem není navrhnout definitivní podobu systému elektronických žádanek, ale ukázat, jak může OPM pomoci odhalit nedostatky, které v textovém zadání nejsou na první pohled patrné.

5.1 Požadavky ČLS JEP a právní úprava elektronických žádanek

Právní rámec elektronických žádanek vymezuje zákon č. 325/2021 Sb., o elektronizaci zdravotnictví, zejména jeho § 31 a § 31a. Zákon zavádí služby výměnné sítě a dočasné úložiště, jehož prostřednictvím se elektronické dokumenty předávají mezi poskytovateli zdravotních služeb. Odesílatel žádanku do úložiště vloží a adresát si ji na základě identifikátoru vyzvedne. Obecná doba uchovávání zásilek je do jejich vyzvednutí, nejdéle však třicet dnů. Pro elektronické žádanky platí zvláštní pravidlo, podle kterého se nevratně odstraňují až po třech letech od poskytnutí vyžádané služby, případně od jejich uložení, pokud služba poskytnuta nebyla [22].

Vystavující poskytovatel má povinnost žádanku bezodkladně uložit do dočasného úložiště a písemně sdělit její identifikátor pacientovi. Na jeho žádost musí zajistit také listinnou podobu žádanky nebo sdělit identifikátor konkrétnímu poskytovateli. Adresát si žádanku vyzvedává prostřednictvím identifikátoru a má ji přijmout, pokud tomu nebrání technické vybavení nebo zákonné důvody pro odmítnutí služby. Po jejím poskytnutí musí bezodkladně zaznamenat tuto skutečnost do dočasného úložiště [22].

Právní úprava se tak soustředí především na předávání žádanky, povinnosti účastníků a pravidla jejího uchovávání. Samotný klinický proces, který vystavení žádanky vyvolal, však nestojí v centru její pozornosti.

Odlišný pohled nabízí pracovní skupina předsednictva České lékařské společnosti Jana Evangelisty Purkyně (ČLS JEP) pro elektronické zdravotnictví. Dne 7. října 2026 formulovala sedm



Obrázek 3 – sedm požadavků na elektronické žádanky, jak je 7. října 2026 formulovala pracovní skupina předsednictva ČLS JEP pro elektronické zdravotnictví: základní standard, skutečný přínos, provázání s výsledkem, distribuce žádanky, srozumitelnost pro pacienta, informace v reálném čase a dostupnost výsledků; uprostřed je cesta od žádanky k výsledku. Je to diskutovaný návrh, ne přijaté stanovisko. Infografiku vytvořil ChatGPT z textu sedmi požadavků; otištěno se souhlasem Milana Cabrnocha.

DOI: 10.35191/medsoft_2026_1_38_merunka

požadavků na elektronické žádanky, které shrnuje obrázek 3 [23]. V době přípravy článku šlo o návrh určený k připomínkování, nikoli o definitivní stanovisko ČLS JEP.

Pro naši analýzu je zásadní třetí požadavek:

„Vydání elektronické žádanky musí být plně provázáno s elektronickým získáním výsledku. Vydání žádanky i přijetí výsledku musí být plně integrováno (a před povinným používáním otestováno) do zdravotnických informačních systémů [23]."

Sedmý požadavek navíc zdůrazňuje dostupnost výsledků v reálném čase pacientovi a zdravotnickým pracovníkům podílejícím se na jeho péči.

Zatímco právní úprava popisuje především předání žádanky a zaznamenání poskytnutí služby, návrh pracovní skupiny požaduje uzavřený klinický proces zahrnující také získání a zpřístupnění výsledku. Elektronická žádanka přitom podle zákonného vymezení sama o sobě není nositelem výsledku vyšetření. Právě tato odlišnost představuje výchozí bod naší analýzy pomocí OPM [22,23].

5.2 Pacient jako hlavní příjemce přínosu systému

Při konceptuálním modelování je důležité nejprve určit, jakému účelu má systém sloužit a komu má přinášet prospěch. Odpověď vymezuje hlavní funkci systému a určuje, které objekty a procesy musí model obsahovat.

V případě elektronických žádanek může systém přinášet výhody lékařům, poskytovatelům zdravotních služeb i zdravotním pojišťovnám. Za hlavního příjemce přínosu však v našem modelu považujeme pacienta, v jehož zájmu je vyžádaná služba poskytována.

Toto rozhodnutí přímo ovlivňuje rozsah modelu. Pokud bychom sledovali pouze potřebu předat žádanku jinému poskytovateli, mohl by proces končit jejím úspěšným doručením. Z pohledu pacienta však jde pouze o jednu z jeho fází. Úplný model musí zachytit cestu od vzniku potřeby vyšetření přes jeho provedení až po získání výsledku a jeho využití při další zdravotní péči.

Volba hlavního příjemce přínosu tedy není jen otázkou hodnotového postoje. Určuje, kde začíná a končí modelovaný proces a podle čeho budeme posuzovat jeho úplnost.

5.3 Elektronická žádanka jako objekt s definovanými stavy

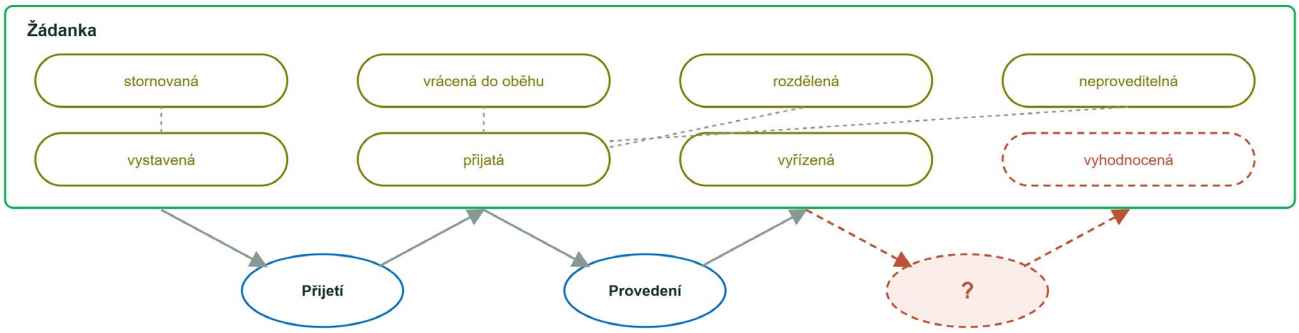
V OPM představuje elektronická žádanka jeden objekt, který během svého životního cyklu prochází různými stavy. Z provozního hlediska lze podle popisu Ministerstva zdravotnictví ČR rozlišit následující stavy a situace:

1. **Vystavená (aktivní).** Lékař vytvoří žádanku, doplní potřebné údaje a uloží ji do centrálního úložiště. Žádanka získá jednoznačný identifikátor a čeká na převzetí oprávněným poskytovatelem.
2. **Přijatá (v péči).** Poskytovatel převezme žádanku na základě identifikátoru nebo QR kódu, který mu může pacient předložit například prostřednictvím aplikace EZKarta. Žádanka se přiřadí příslušnému pracovišti, aby nedocházelo k nežádoucímu duplicitnímu čerpání služby.
3. **Vrácená do oběhu.** Pokud poskytovatel nemůže službu provést, například kvůli technické poruše nebo změně rozhodnutí pacienta, žádanka se vrací do aktivního stavu a může ji převzít jiné pracoviště.
4. **Rozdělená.** U komplexních vyšetření lze žádanku rozdělit na dílčí požadavky, například pro různé laboratoře. Vzniklé žádanky mají vlastní životní cyklus, ale zůstávají propojené s původním požadavkem.
5. **Vyřízená (provedená).** Po poskytnutí služby se žádanka uzavírá a nelze ji znovu použít pro stejné čerpání. Systém může současně informovat vystavujícího lékaře a zpřístupnit související výsledky.
6. **Neproveditelná.** Poskytnutí služby znemožní například kontraindikace nebo odmítnutí výkonu pacientem. Informace o této skutečnosti má být dostupná vystavujícímu lékaři, aby mohl rozhodnout o dalším postupu.
7. **Stornovaná.** Žádanka je zrušena například kvůli chybným údajům nebo změně klinického rozhodnutí. Její životní cyklus tím končí bez poskytnutí původně požadované služby.

Z pohledu OPM je důležité rozlišovat skutečné stavy objektu od procesů, které tyto stavy mění. Například vrácení žádanky do oběhu je vhodnější chápat jako proces, který žádanku převádí z přijatého do aktivního stavu. Podobně rozdělení žádanky představuje proces, při kterém vznikají nové dílčí objekty.

Obrázek 4 zachycuje elektronickou žádanku jako objekt, jehož stav se mění prostřednictvím jednotlivých procesů. Vedle diagramu je uveden odpovídající textový zápis v jazyce OPL. Čtenář tak může porovnat grafickou a textovou podobu stejného modelu.

Právní úprava přitom popisuje životní cyklus poněkud odlišně. Lze z ní odvodit vystavení žádanky, její uložení, přidělení identifikátoru, vyzvednutí adresátem, zaznamenání poskytnutí nebo neposkytnutí služby a následné odstranění po uplynutí zákonné lhůty [22]. Provozní a právní pohled se nemusí vzájemně vylučovat, ale zachycují stejný proces s odlišnou mírou



Sedm stavů, které vede Ministerstvo zdravotnictví: dole hlavní cesta, nad ní odbočky z „vystavené" a „přijaté".

Osmý stav „vyhodnocená" je nakreslen, protože ho žádá návrh pracovní skupiny ČLS JEP — ale nevede do něj žádný proces, který by úprava ukládala.

Červeně přerušovaně: co v právní úpravě chybí. Je to naše značení, ne prvek notace OPM.

Obrázek 4 – životní cyklus žádanky jako stavy jednoho objektu. Dole hlavní cesta, nad ní odbočky, do kterých se žádanka dostane z „vystavené" a „přijaté" – dohromady sedm stavů, které vede Ministerstvo zdravotnictví. Osmý stav „vyhodnocená" je nakreslen červeně přerušovaně, protože ho žádá návrh pracovní skupiny ČLS JEP, ale nevede do něj žádný proces, který by právní úprava ukládala. Červené přerušování je značení autorů, ne prvek notace OPM.

DOI: 10.35191/medsoft_2026_1_38_merunka

podrobnosti. Některé provozní stavy a funkce systému proto nemusí být v zákoně výslovně pojmenovány.

Při převodu do OPM identifikujeme základní objekty, mezi které patří elektronická žádanka, její identifikátor, dočasné úložiště a údaj o poskytnutí služby. Model zahrnuje také pacienta, vystavujícího lékaře, poskytovatele vyžádané služby a procesy, kterými tito účastníci se žádankou pracují. Jde zejména o její vystavení, uložení, převzetí, poskytnutí služby a následné odstranění. Každý vztah zachycený v diagramu má zároveň odpovídající textové vyjádření v OPL.

Základní model elektronické žádanky ukazuje obrázek 5. Žádanka zde vystupuje jako informační objekt uložený v centrálním úložišti, se kterým prostřednictvím jednotlivých procesů pracují pacient, lékař, poskytovatel vyžádané služby a informační systém.

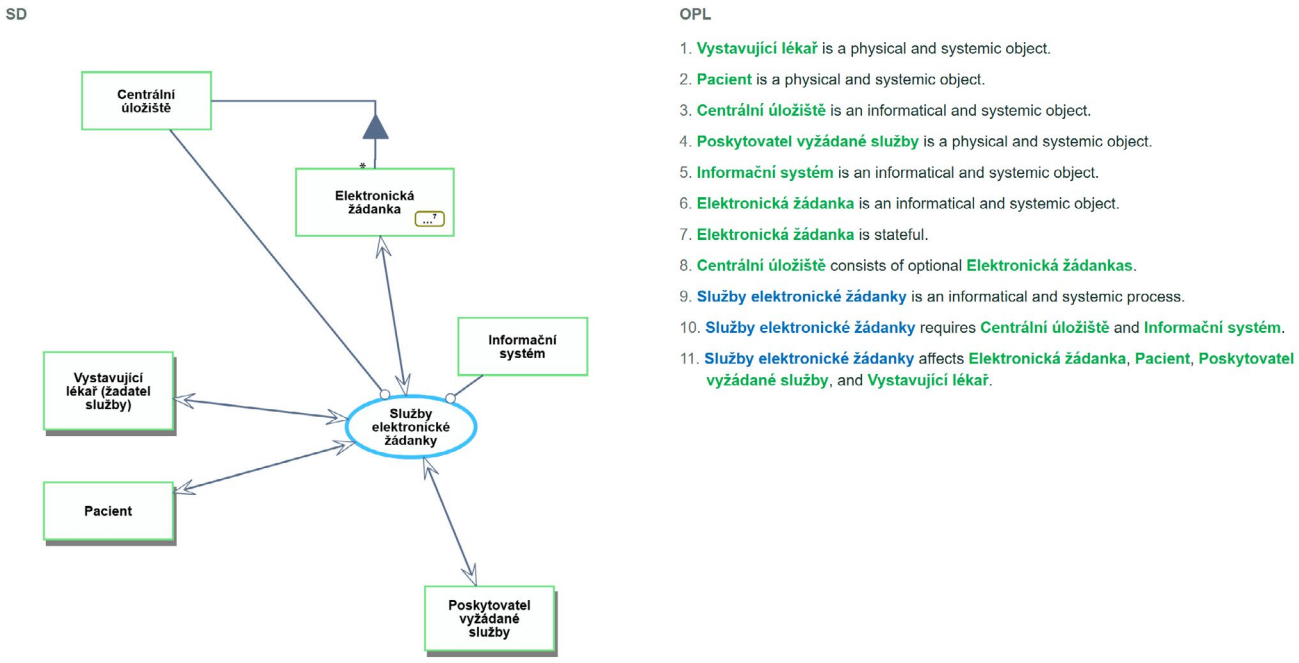


Diagram i věty pocházejí z téhož modelu v OPCloudu; OPL vypsal nástroj sám.

Obrázek 5 – kořenový diagram služeb elektronické žádanky (SD). Elektronická žádanka se ukládá do Centrálního úložiště; proces Služby elektronické žádanky vyžaduje úložiště a informační systém a dotýká se pacienta, vystavujícího lékaře a poskytovatele vyžádané služby. Vlevo diagram z OPCloudu, vpravo jeho OPL – totéž ve větách, které vypsal nástroj sám. Objekty jsou zeleně, procesy modře, stavy olivově; jsou to barvy nástroje. Stín u objektu znamená, že je fyzický. Vyhrazená slova OPL jsou anglicky, jména pocházejí z modelu.

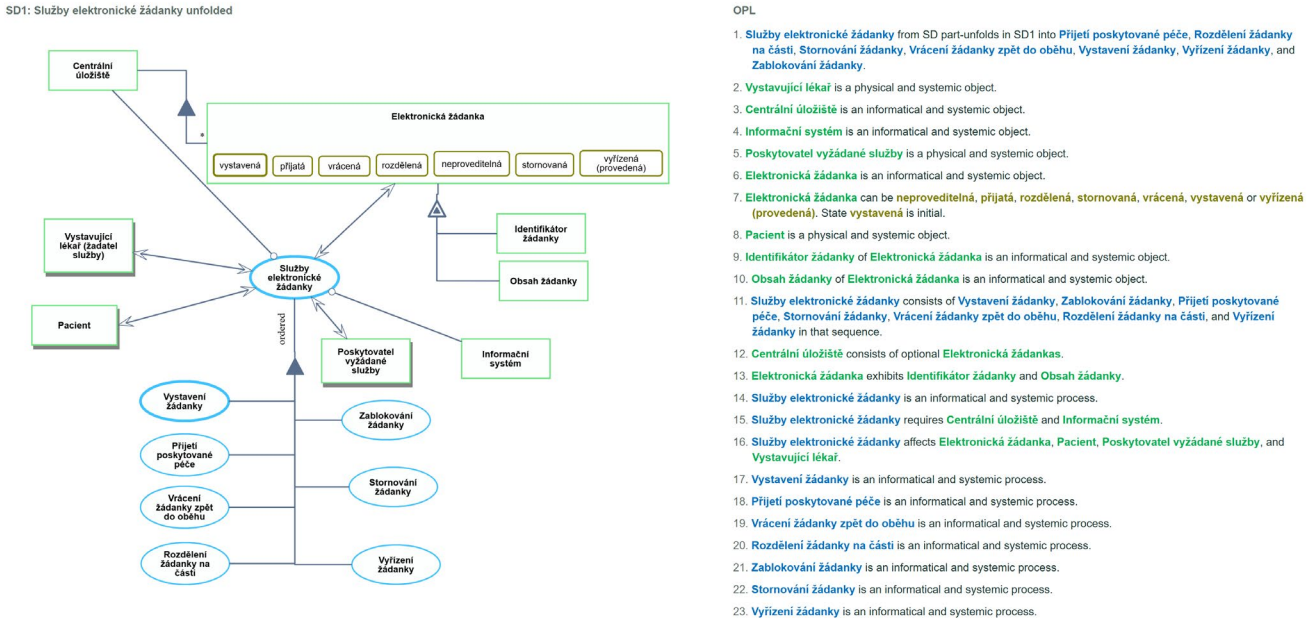
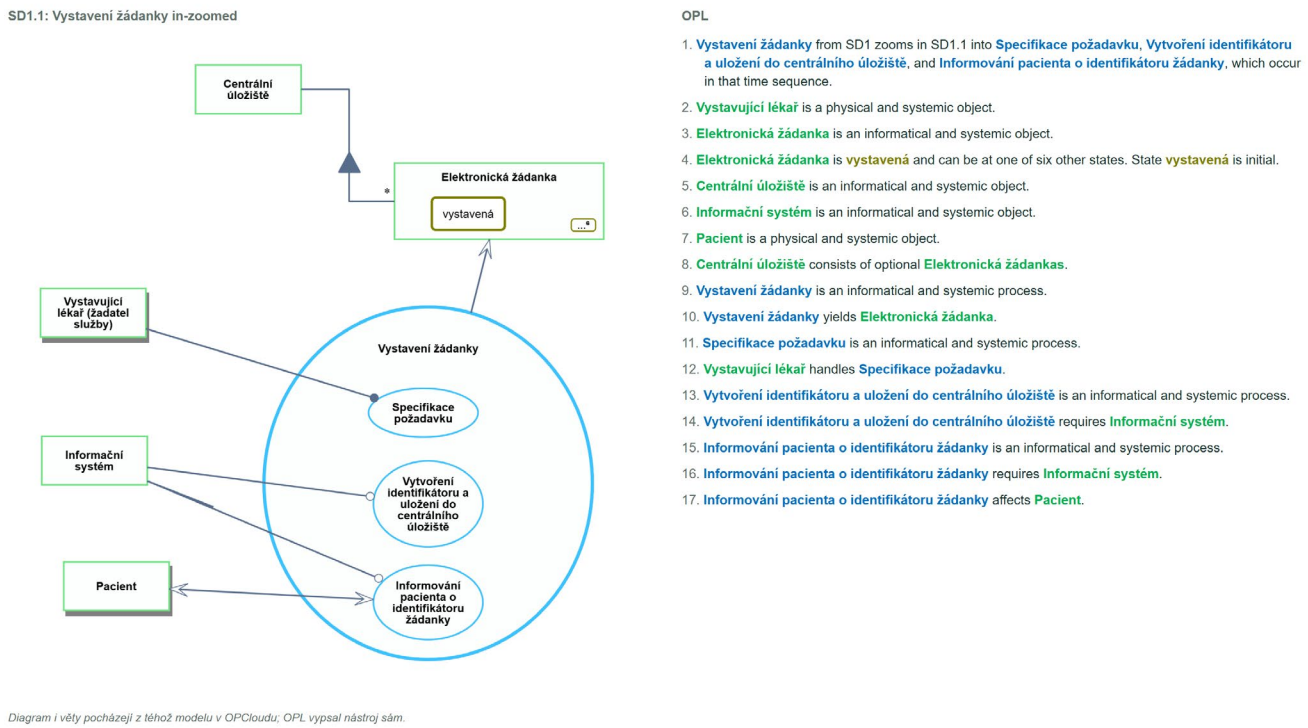


Diagram i věty pocházejí z téhož modelu v OPCloudu; OPL vypsal nástroj sám.

Obrázek 6 – rozbalený kořenový diagram (SD1): sedm služeb žádanky – vystavení, zablokování, přijetí poskytované péče, stornování, vrácení zpět do oběhu, rozdělení na části a vyřízení – a sedm stavů žádanky uvnitř jejího obdélníku; výchozí je „vystavená". Je to týž výčet jako v kapitole 5.2. Žádanka vykazuje identifikátor a obsah. Vlevo diagram z OPCloudu, vpravo jeho OPL – totéž ve větách, které vypsal nástroj sám. Objekty jsou zeleně, procesy modře, stavy olivově; jsou to barvy nástroje. Stín u objektu znamená, že je fyzický. Vyhrazená slova OPL jsou anglicky, jména pocházejí z modelu



Obrázek 7 – přiblížení procesu Vystavení žádanky (SD1.1): uvnitř elipsy jsou tři podprocesy ve svislém pořadí, které je jejich posloupností – specifikace požadavku, kterou provádí vystavující lékař, vytvoření identifikátoru a uložení do centrálního úložiště a informování pacienta o identifikátoru. Z procesu vychází žádanka ve stavu „vystavená". Vlevo diagram z OPCloudu, vpravo jeho OPL – totéž ve větách, které vypsal nástroj sám. Objekty jsou zeleně, procesy modře, stavy olivově; jsou to barvy nástroje. Stín u objektu znamená, že je fyzický. Vyhrazená slova OPL jsou anglicky, jména pocházejí z modelu.

OPM umožňuje model postupně zpřesňovat, aniž by bylo nutné měnit typ diagramu. Na obrázku 6 jsou rozpracovány služby elektronické žádanky, její stavy, identifikátor a obsah včetně vazeb na výsledky vyžádané služby.

Obrázek 7 podrobněji zachycuje proces vystavení žádanky. Lékař zadává údaje o pacientovi a požadované službě, informační systém zajistí přidělení identifikátoru a uložení žádanky do centrálního úložiště. Identifikátor je následně zpřístupněn pacientovi, který jej může využít při komunikaci s poskytovatelem.

Takto vytvořený model zachycuje nejen jednotlivé činnosti, ale také jejich důsledky pro stavy žádanky a souvisejících objektů. Díky tomu lze systematicky zkoumat úplnost jejich životního cyklu.

5.4 Co konceptuální model odhalil

Převod právní úpravy elektronických žádanek do OPM umožnil identifikovat tři podstatné nedostatky ve struktuře popsaného procesu, které nemusí být při běžném čtení právního textu zřejmé.

1. **Právní úprava popisuje především předávání žádanky, nikoli celý její životní cyklus.** Vymezuje vystavení, uložení, předání a uchovávání žádanky, ale jednoznačně neurčuje hlavní účel celého procesu z pohledu pacienta. Samotné doručení žádanky přitom ještě neznamená naplnění klinické potřeby, kvůli které byla vystavena.
2. **Zaznamenání poskytnutí služby neuzavírá celý klinický proces.** Poskytovatel má povinnost vložit do dočasného úložiště údaj o poskytnutí služby [22]. Tento údaj však není totožný s výsledkem vyšetření ani s jeho odborným vyhodnocením. Z právního popisu proto nelze přímo odvodit celý proces předání výsledku vystavujícímu lékaři a jeho využití při další péči. Provozní systém může obsahovat notifikace a zpřístupnění výsledků, ale tyto návaznosti je nutné v modelu výslovně zachytit.
3. **Stav „vyhodnocena" není v modelu odvozeném z právní úpravy dosažitelný.** Z klinického hlediska je přirozené očekávat, že výsledek vyšetření bude odborně posouzen a využit při rozhodování o další péči. Právní úprava však výslovně nepopisuje proces, který by žádanku nebo související klinický požadavek do tohoto stavu převedl. Bez jeho doplnění zůstává stav v konceptuálním modelu nedosažitelný.

Podstatné není pouze to, že model tyto rozdíly odhaluje, ale především to, že je umožňuje jednoznačně pojmenovat a ověřovat. Právní předpis může být v rámci svého účelu správný, přestože nepopisuje celý klinický proces potřebný k naplnění pacientovy potřeby.

To zároveň neznamená, že elektronická žádanka musí sama obsahovat všechny výsledky vyšetření. Výsledek může představovat samostatný informační objekt. Model však musí zachytit jeho vazbu na žádanku i procesy, které zajišťují jeho předání a následné využití.

5.5 Proč tyto nedostatky nemusí být patrné v běžném procesním schématu

Rozdíl mezi OPM a běžným procesním modelováním se nejlépe ukazuje právě na životním cyklu elektronické žádanky. V BPMN lze celý postup znázornit jako posloupnost úloh zahrnujících vystavení, odeslání, přijetí a provedení požadované služby. Takový diagram může být správně sestavený, aniž by z něj bylo zřejmé, zda celý proces skutečně naplňuje původní klinickou potřebu pacienta.

BPMN sice umožňuje pracovat také s datovými objekty a jejich stavy, ale při modelování zaměřeném především na tok čin-

DOI: 10.35191/medsoft_2026_1_38_merunka

ností nemusí být životní cyklus konkrétní žádanky dostatečně patrný [16]. Může tak zůstat nepovšimnuto, že některý požadovaný stav není dosažitelný nebo že proces končí předčasně.

OPM naproti tomu propojuje procesy přímo se stavy objektů, které mění [1,3]. Pokud má například žádanka dosáhnout stavu „vyhodnocena", musí model obsahovat proces, který tuto změnu zajistí. Jestliže chybí, je nedostatek patrný přímo v diagramu. Stejně lze rozpoznat stav, ze kterého neexistuje cesta k očekávanému výsledku.

Konceptuální model tak není pouze grafickým vyjádřením zadání, ale umožňuje také ověřovat jeho úplnost a vnitřní návaznosti. V následujících kapitolách ukážeme, jak lze tuto možnost dále rozšířit prostřednictvím simulace a automatizované kontroly chování modelu.

6 Spustitelný konceptuální model

V předchozích kapitolách jsme ukázali, jak OPM umožňuje zachytit strukturu a chování systému a odhalovat nedostatky v návaznosti jeho procesů. Další možností je konceptuální model skutečně spustit a simulací ověřovat jeho chování. V této kapitole představíme, co tím získáváme a jak lze OPM modely převádět do spustitelné podoby.

6.1 Co může simulace odhalit nad rámec vizuální kontroly

Při čtení diagramu si člověk často nevědomě doplňuje souvislosti, které v něm nejsou výslovně zachyceny. Lékař například předpokládá, že výsledek laboratorního vyšetření bude předán vystavujícímu lékaři, přestože odpovídající proces může v modelu chybět. Simulace naopak pracuje pouze s definovanými procesy, vztahy a podmínkami. Chybějící návaznosti si nedomýšlí, a proto může odhalit problémy, které při běžném čtení uniknou pozornosti.

Jde zejména o čtyři typy nedostatků:

- **Uvíznutí procesu.** Dva procesy například vzájemně čekají na své dokončení, takže žádný nemůže pokračovat. Při simulaci se tato situace projeví zastavením vykonávání procesů.
- **Nedosažitelný stav.** Objekt má definovaný stav, do kterého však nevede žádná přípustná posloupnost procesů. Příkladem je stav elektronické žádanky „vyhodnocena", pokud chybí proces zajišťující její vyhodnocení.
- **Chybějící podmínka spuštění procesu.** Proces se může vykonat i v situaci, kdy by to skutečná pravidla systému neměla dovolovat. Příčinou bývá chybějící nebo nesprávně definovaná podmínka.
- **Nepokrytá větev procesu.** Model neřeší určitou situaci, která může v praxi nastat. Pokud například laboratoř nedodá výsledek, ale model neobsahuje odpovídající postup, může žádanka zůstat ve stavu, ze kterého nelze pokračovat.

S podobnými problémy jsme se setkali již při modelování elektronického receptu [18]. Simulace tehdy odhalila objekty čekající na událost, která v daném průběhu nemohla nastat. Tento nedostatek přitom neodhalilo předchozí posuzování textového zadání za účasti odborníků z různých oblastí.

Simulaci OPM modelů podporují prostředí OPCloud i námi vyvíjený otevřený OPM Editor [4,11]. Uživatel může přímo v diagramu sledovat spouštění procesů a změny stavů objektů podle definovaných pravidel. Nedostatky tak lze identifikovat již při přípravě konceptuálního modelu, ještě před implementací informačního systému.

6.2 Simulace jako prostředek ověřování správnosti zadání

Význam spustitelného modelu však přesahuje samotné hledání chyb. Běžné textové zadání obsahuje požadavky, které mohou být jednotlivě správné, ale dohromady nemusí tvořit úplný a konzistentní celek. Jejich kontrola proto závisí především na zkušenostech a úsudku odborníků.

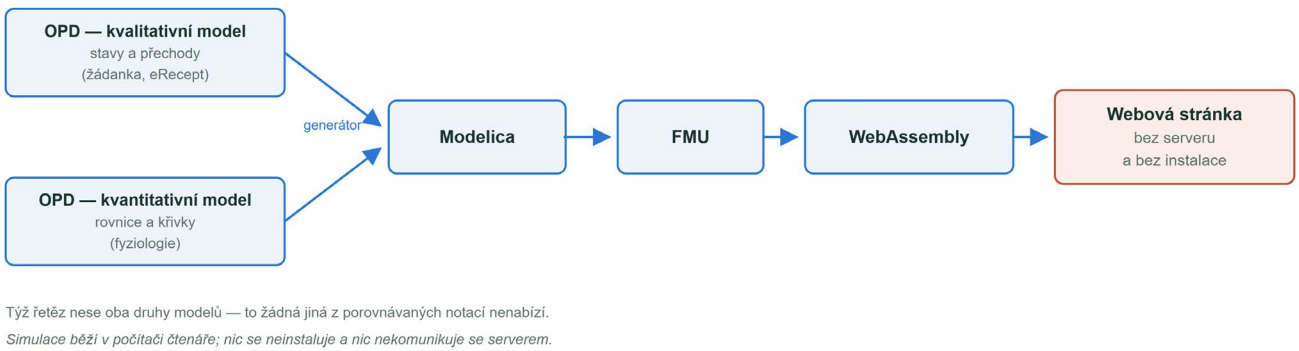
Spustitelný model umožňuje formulovat konkrétní předpoklady o chování systému a následně je ověřovat. Pokud například požadujeme, aby elektronická žádanka mohla dospět až k odbornému vyhodnocení výsledku, musí model obsahovat všechny procesy a vazby, které to za stanovených podmínek umožňují. Simulace může ukázat, zda je takový průběh možný a kde případně chybí potřebná návaznost.

Konceptuální model se tím stává ověřitelným, a za určitých podmínek také vyvratitelným tvrzením o fungování systému. Při společné práci lékařů, analytiků, právníků a vývojářů tak nemusí diskuse zůstat pouze u různých výkladů textu. Odlišné představy lze vyjádřit samostatnými modely a porovnat jejich chování.

Simulace samozřejmě sama nerozhodne, který cíl systému je správný. Umožňuje však ověřit, zda navržené procesy vedou k předem stanovenému výsledku. Odborná diskuse se tak může opírat o reprodukovatelné chování modelu.

6.3 Jednotná metodika pro kvalitativní a kvantitativní simulace

Náš přístup umožňuje využít OPM jako společný základ pro dva odlišné typy simulací, kvalitativní a kvantitativní.



Obrázek 8 – cesta z konceptuálního diagramu do simulátoru v prohlížeči: OPD → Modelica → FMU → WebAssembly→ webová stránka bez serveru. Týž řetěz nese kvalitativní i kvantitativní modely.

DOI: 10.35191/medsoft_2026_1_38_merunka

Kvalitativní simulace sledují především průběh procesů a změny stavů objektů. V případě elektronické žádanky nás zajímá, zda může přejít z jednoho stavu do druhého, jaké podmínky umožňují spuštění jednotlivých procesů a zda některý průběh nevede k uvíznutí. Výsledkem je sled událostí, který lze pozorovat přímo v diagramu [4].

Kvantitativní simulace se zaměřují na vývoj číselných veličin v čase. Ve fyziologii může jít například o krevní tlak, průtok, koncentraci látek nebo saturaci krve kyslíkem. Chování těchto systémů bývá popsáno diferenciálními rovnicemi a výsledkem simulace jsou časové průběhy sledovaných veličin.

V našem řešení mohou oba typy modelů vznikat prostřednictvím stejné metodiky OPM a následně být převedeny do programů spustitelných ve webovém prohlížeči. Společná je konceptuální reprezentace, nikoli samotné výpočetní postupy. Pro kvantitativní simulace je nadále nutné použít odpovídající matematické metody a numerické algoritmy.

OPM tedy nenahrazuje specializované jazyky, jako je Modelica, ani notace určené pro podrobný popis pracovních postupů, například BPMN. Její přínos spočívá v možnosti zachytit organizační, informační a fyziologické souvislosti pomocí společných konceptuálních principů a následně pro ně využít různé simulační postupy.

6.4 Simulace jako nástroj kontroly výstupů velkých jazykových modelů

Možnost automatizovaného zpracování OPM otevírá také nové možnosti kontroly výstupů velkých jazykových modelů (LLM) [24, 25]. V předchozím článku jsme navrhli využívat OPM jako formální vrstvu pro odhalování nepřesností, rozporů a nepodložených tvrzení v generovaném obsahu [26]. Původně jsme se zaměřovali především na kontrolu struktury modelu. Jeho převod do spustitelné podoby nyní umožňuje ověřovat také chování.

Navrhovaný postup má pět kroků:

1. **Vytvoření návrhu.** Jazykový model zpracuje výchozí text, například právní předpis nebo technickou specifikaci, a navrhne odpovídající model v jazyce OPL.
2. **Odborná kontrola.** Člověk posoudí, zda navržené objekty, procesy, stavy a jejich vztahy odpovídají původnímu zadání, a případně model upraví.
3. **Formální kontrola.** Modelovací nástroj ověří soulad modelu s pravidly OPM a přípustnost jeho struktury.
4. **Převod do spustitelné podoby.** Z modelu se automaticky vytvoří program nebo simulační reprezentace.
5. **Simulace a porovnání.** Chování modelu se porovná s požadavky, očekávanými scénáři nebo referenčními výsledky.

Tento postup může odhalit například nedosažitelné stavy, chybějící návaznosti nebo nesprávně definované podmínky. Nezaručuje však, že model odpovídá skutečnému významu výchozího textu. I formálně správný a úspěšně simulovatelný model může vycházet z chybné interpretace zadání. Odborná kontrola proto zůstává nezbytná.

Rozdíl mezi běžným generovaným obsahem a formálním modelem ukazuje také obrázek 3 tohoto článku. Infografika se sedmi požadavky pracovní skupiny ČLS JEP vznikla pomocí velkého jazykového modelu na základě textového zadání. Její správnost musí čtenář posoudit porovnáním s původním textem, protože obrázek neobsahuje formálně definované vztahy, které by bylo možné automaticky analyzovat.

Pokud bychom stejné požadavky vyjádřili modelem OPM, získali bychom reprezentaci, jejíž strukturu lze formálně kontrolovat a jejíž chování lze při splnění potřebných předpokladů simulovat. Jazykový model tak může pomáhat s převodem textového zadání do strukturované podoby, odborník ověřuje její věcnou správnost a specializovaný nástroj kontroluje formální vlastnosti a chování.

6.5 Hranice možností simulace

Přestože simulace umožňuje systematicky ověřovat chování konceptuálního modelu, úspěšné spuštění samo o sobě nedokazuje, že model správně popisuje skutečnost. Simulace ověřuje důsledky předpokladů, které jsme do modelu vložili. Pokud jsou tyto předpoklady nesprávné nebo neúplné, může simulace probíhat bez problémů, přestože její výsledky neodpovídají reálnému systému.

U kvantitativních fyziologických modelů je proto nutné porovnávat vypočtené hodnoty s experimentálními nebo klinickými měřeními, případně model kalibrovat a validovat na nezávislých datech. Ani dobrá shoda výsledků však není absolutním důkazem jeho správnosti.

Podobně může simulace elektronické žádanky ukázat, že všechny definované procesy vedou k požadovanému výsledku, aniž by tím bylo zaručeno stejné chování skutečného informačního systému. V praxi mohou nastat situace, které model nezachycuje, například technické výpadky, organizační omezení nebo nesprávné vstupní údaje.

Simulace proto nenahrazuje odborné posouzení ani provozní testování. Její hlavní hodnotou je možnost odhalit nedostatky a zpřesnit zadání ještě před realizací systému. Spustitelný konceptuální model tak představuje prostředek pro systematické ověřování předpokladů, nikoli automatický důkaz správnosti navrhovaného řešení.

7 Od diagramu ke spustitelné simulaci v prohlížeči

V této kapitole představíme postup, který umožňuje automaticky převést konceptuální model OPM do simulace spustitelné přímo ve webovém prohlížeči. Technické podrobnosti využití jazyka Modelica, standardu Functional Mock-up Interface (FMI) a technologie WebAssembly popisují další dva příspěvky tohoto sborníku [27]. Zde se soustředíme na hlavní přínos z hlediska konceptuálního modelování. Výchozím bodem je diagram OPD, nikoli ručně vytvořený simulační model.

Z diagramu se automaticky vygeneruje model v jazyce Modelica, který se převede do standardizované podoby Functional Mock-up Unit (FMU) a následně pomocí WebAssembly do webové aplikace. Výpočty probíhají přímo v prohlížeči uživatele bez nutnosti komunikace se serverem během simulace. Není proto třeba instalovat specializovaný software.

Lékař, student nebo jiný zájemce tak může model otevřít prostřednictvím internetového odkazu, spustit jej, měnit parametry a sledovat výsledné chování. Konceptuální diagram se tím stává interaktivním nástrojem pro experimentování a ověřování.

7.1 Ověření shody s referenčním simulátorem

Pro posouzení správnosti převodu nestačí, že výsledná aplikace funguje. Je také nutné ověřit, zda její numerické výsledky odpovídají původnímu matematickému modelu.

Proto jsme simulace vytvořené naším postupem porovnali s referenčními výpočty v zavedených prostředích, zejména v průmyslovém simulátoru Dymola. Výsledky shrnuje tabulka 2.

DOI: 10.35191/medsoft_2026_1_38_merunka

co se počítá	odkud model	odchylka
jednoduchá respirace	z OPM diagramu	0,0026 %
normální oběh (van Meurs)	z OPM diagramu	0,0157 %
respirace z knihovny	Modelica, OpenModelica	0,0025 %
metabolismus železa, ustálený stav	Modelica, Dymola	0,0000 %

Tabulka 2 – odchylka simulátoru běžícího v prohlížeči proti průmyslovému simulátoru nad týmž modelem. První dva řádky jsou modely odvozené z OPM diagramu.

Nejdůležitější jsou první dva výsledky, protože jde o modely automaticky odvozené z OPM diagramů. Odchylka dosáhla pouze 0,0026 % u jednoduché respirace a 0,0157 % u modelu krevního oběhu podle van Meurse. Oba modely přitom pocházejí z odlišných oblastí fyziologie a využívají rozdílné knihovny. Výsledky tedy ukazují, že navržený převod není omezen na jediný konkrétní fyziologický model.

Další dva řádky tabulky představují modely vytvořené přímo v jazyce Modelica, bez použití OPM. Slouží jako doplňující ověření technického postupu převodu do webového prostředí a ukazují jeho použitelnost také pro již existující matematické modely.

7.2 Co výsledky porovnání skutečně dokazují

Velmi nízká numerická odchylka nedokazuje fyziologickou správnost modelu, ale věrnost jeho převodu do webového simulátoru. Pokud například původní model respirace vychází z nepřesného fyziologického předpokladu, může tuto nepřesnost obsahovat jak referenční simulace, tak výsledná webová aplikace, přestože se jejich výpočty téměř dokonale shodují.

V této fázi proto ověřujeme především to, zda automatický převod zachovává výpočetní chování původního modelu. Posouzení jeho fyziologické platnosti vyžaduje samostatné porovnání s experimentálními nebo klinickými daty.

Dosažené výsledky potvrzují velmi dobrou numerickou shodu automaticky vytvořených webových simulací s referenčními výpočty v obou testovaných modelech odvozených z OPM. To podporuje využití OPM jako výchozí konceptuální reprezentace také pro kvantitativní fyziologické simulace.

7.3 Interaktivní modely dostupné čtenářům

Praktickou výhodou našeho řešení je možnost zpřístupnit simulace prostřednictvím běžných internetových odkazů. Čtenář nemusí instalovat specializovaný software a může si jednotlivé modely sám spustit a vyzkoušet.

Pro tento sborník jsme připravili následující interaktivní modely. Všechny jsou dostupné také v angličtině změnou parametru lang=cs na lang=en.

Jednoduchá respirace vytvořená z OPM diagramu. Model umožňuje sledovat základní fyziologické děje při dýchání. Stránka se otevírá přímo s diagramem OPD, ze kterého simulace vznikla.

Otevřít simulaci jednoduché respirace (*https://bodylight.physiome.cz/SvelteBodylight.js/medsoft2026/fmu-lab.html?experiment=simple-respiration-opm/opm-derived&lang=cs&view=opd*)

Normální krevní oběh podle van Meurse vytvořený z OPM diagramu. Kvantitativní model hemodynamiky umožňuje porovnávat konceptuální diagram s časovými průběhy fyziologických veličin. Zachycuje jej také obrázek 9.

Otevřít simulaci krevního oběhu (*https://bodylight.physiome.cz/SvelteBodylight.js/medsoft2026/fmu-lab.html?experiment=normal-circulation-opm/opm-derived&lang=cs&view=opd*)

Kvalitativní model aktivace uživatele. Na rozdíl od fyziologických simulací zachycuje procesy a změny stavů objektů. Diagram zde současně slouží jako interaktivní rozhraní, ve kterém lze spouštět jednotlivé procesy.

Otevřít model aktivace uživatele (*https://bodylight.physiome.cz/SvelteBodylight.js/medsoft2026/fmu-lab.html?experiment=user-activation-process/tutorial&lang=cs*)

Pro přímé porovnání zpřístupňujeme také referenční modely vytvořené bez použití OPM:

Referenční model jednoduché respirace (*https://bodylight.physiome.cz/SvelteBodylight.js/medsoft2026/fmu-lab.html?experiment=simple-respiration/student-basic&lang=cs*)

Referenční model normálního krevního oběhu (*https://bodylight.physiome.cz/SvelteBodylight.js/medsoft2026/fmu-lab.html?experiment=normal-circulation/student-basic&lang=cs*)

Možnost přímo spustit model popisovaný v odborném článku považujeme za důležitý přínos pro výzkum i vzdělávání. Čtenář není odkázán pouze na statické diagramy a tabulky, ale může sám měnit parametry, sledovat průběh procesů a porovnávat výsledky.

Ve výuce medicíny a zdravotnické informatiky tak lze propojit studium struktury systému s praktickým experimentováním. Student nejprve porozumí konceptuálnímu diagramu a následně může pozorovat, jak se modelovaný systém skutečně chová.

8 Kauzální a akauzální modelování v OPM

V této kapitole představujeme vlastní výzkumný příspěvek k OPM, kterým je návrh akauzálního modelování kvantitativních a fyziologických systémů. Zatímco standardní OPM popisuje strukturu systému, procesy a změny stavů objektů, náš přístup umožňuje vyjadřovat také vztahy mezi veličinami pomocí matematických rovnic.

Navrhované řešení jsme prakticky ověřili na dvou fyziologických modelech. Zatím však jde o výzkumný návrh, nikoli o standardizované rozšíření OPM. Jeho cílem je využít stávající konceptuální principy metodiky také pro popis spojitých dynamických systémů.

8.1 Rozdíl mezi kauzálním a akauzálním modelováním

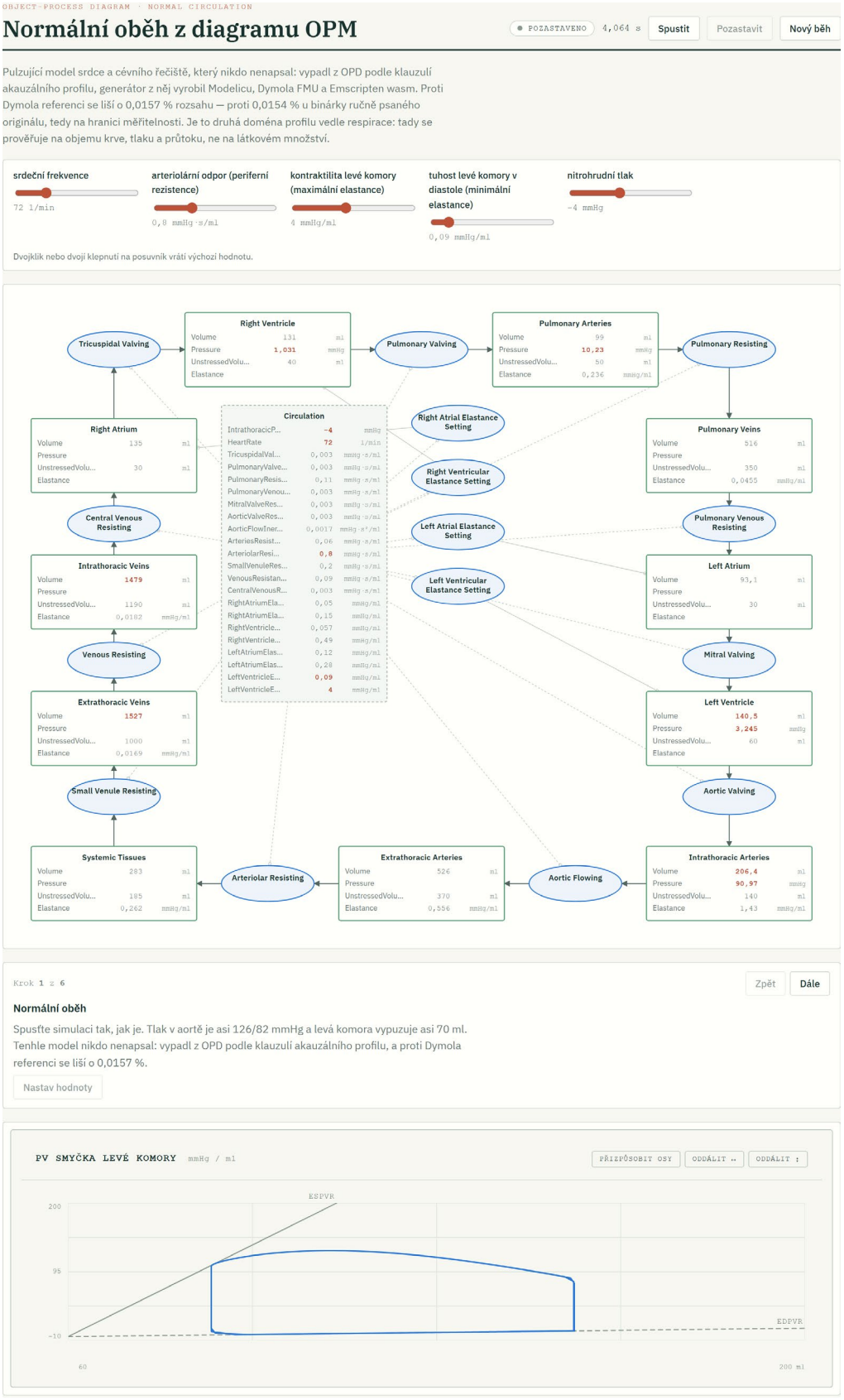
Kauzální modelování zpravidla předem určuje, které veličiny jsou vstupní a které výstupní. Proces se chová podobně jako funkce, která ze známých vstupů vypočítává výstupy. Směr výpočtu tak musí být určen již při tvorbě modelu.

Akauzální modelování naproti tomu vyjadřuje vztahy mezi veličinami pomocí rovnic, aniž by předem stanovoval směr výpočtu. Rovnice představuje omezení, které musí všechny zúčastněné veličiny splňovat. O tom, které hodnoty budou známé a které se mají vypočítat, rozhoduje až konkrétní experiment a řešení matematického modelu [28,29].

Tento rozdíl je důležitý zejména ve fyziologii. Při modelování krevního oběhu nás například zajímá vztah mezi tlakovým rozdílem, průtokem a cévním odporem. Pokud známe tlaky a odpor, můžeme vypočítat průtok. Za jiných podmínek však může být známý průtok a hledanou veličinou tlakový rozdíl. Samotný fyziologický vztah se přitom nemění.

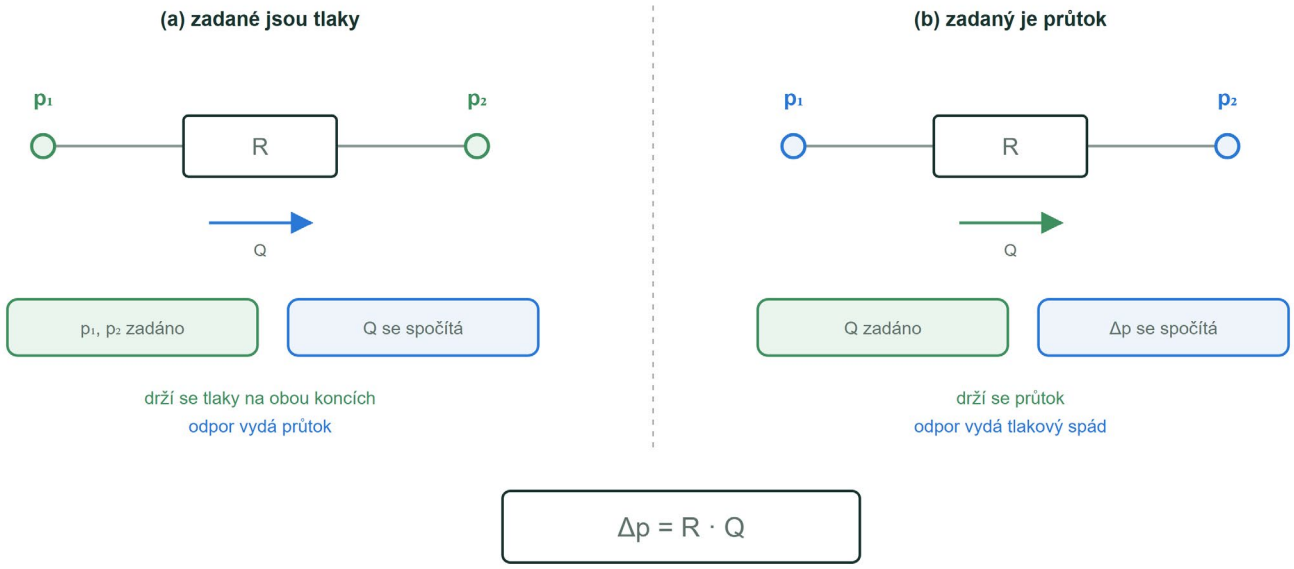
Akauzální přístup tak odděluje odborný popis fyziologických vztahů od technického způsobu jejich výpočtu. Lékař nebo fyziolog se může soustředit na modelované mechanismy, zatímco sestavení a numerické řešení rovnic zajišťuje výpočetní nástroj.

DOI: 10.35191/medsoft_2026_1_38_merunka



Obrázek 9 – simulátor vygenerovaný z OPM diagramu, jak ho vidí čtenář v prohlížeči: model krevního oběhu nakreslený jako OPD, pozastavený ve čtvrté sekundě běhu. Zelené obdélníky jsou objekty, modré elipsy procesy; červeně jsou hodnoty, které se za běhu mění, černě zadané parametry. Posuvníky nad diagramem působí za běhu. Dole je první z šesti grafů, PV smyčka levé komory se šikmou ESPVR a plochou EDPVR; zbylých pět je kvůli velikosti mimo obrázek. Jména objektů a procesů zůstávají anglická i na české stránce – jsou to jména z OPD a OPL je anglická z podstaty notace. Jako jediný obrázek je to snímek stránky, tedy jen PNG, ne vektor. Simulátor běží na adrese https://bodylight.physiome.cz/SvelteBodylight.js/medsoft2026/fmu-lab.html?experiment=normal-circulation-opm/opm-derived&lang=cs&view=opd otevře stránku rovnou na diagramu, totiž na volbě „Diagram OPD" v přepínači Zobrazení nahoře; bez něj se otevře „Standardní".

DOI: 10.35191/medsoft_2026_1_38_merunka



Obrázek 10 – kauzalita je vlastností experimentu, ne modelu: táž rovnice odporu počítá průtok z tlaků, nebo tlakový spád z průtoku, podle toho, co je v pokusu zadáno. Zeleně je zadané, modře to, co se počítá.

8.2 Kauzalita a její závislost na podmínkách experimentu

Rozdíl mezi oběma přístupy ukazuje jednoduchý vztah mezi tlakovým rozdílem, průtokem a odporem.

Známe-li tlakový rozdíl a odpor, vypočítáme průtok. Známe-li průtok a odpor, určíme tlakový rozdíl. Rovnice zůstává stejná, mění se pouze podmínky experimentu. Směr výpočtu tedy není pevnou vlastností fyzikálního vztahu.

Na tomto principu je založen také náš simulační přístup. Základní model akauzálně popisuje vztahy mezi veličinami, zatímco experiment určuje parametry, počáteční a okrajové podmínky, vnější zásahy a sledované veličiny. Výpočetní nástroj poté sestaví a vyřeší odpovídající soustavu rovnic.

Možnost opakovaného využití jednoho modelu ukazujeme na krevním oběhu, pro který jsme připravili osm experimentálních scénářů. Zahrnují normální oběh, selhání srdečních komor, chlopenní vady, systolické a diastolické srdeční selhání, krvácení, atrioventrikulární blokády a změny nitrobřišního tlaku.

Všechny scénáře využívají stejný základní fyziologický model. Liší se pouze nastavením parametrů, počátečními podmínkami nebo charakterem zásahu. Není proto nutné pro každou klinickou situaci vytvářet samostatný matematický popis celého oběhového systému.

8.3 Možnosti a hranice normy ISO 19450

Pro náš výzkum je důležité rozlišovat mezi tím, co již definuje norma ISO 19450:2024, a tím, co představuje náš návrh.

Standardní OPM popisuje chování především prostřednictvím principu Event-Condition-Action (ECA), tedy událostí, podmínek a akcí [1]. Tento přístup je vhodný pro diskrétní procesy a změny stavů objektů. Norma však neposkytuje podrobně definovanou syntaxi ani výpočetní sémantiku pro akauzální soustavy diferenciálních a algebraických rovnic.

Zajímavým východiskem je ustanovení 11.3, které připouští vztahové omezení (relationship constraint) u veličin s reálnou hodnotou [1]. Norma však tento pojem dále nerozpracovává do podoby samostatného aparátu pro kvantitativní simulace.

Právě zde vidíme prostor pro navrhovanou interpretaci. Vztahové omezení lze chápat jako matematický vztah mezi veličinami, který předem neurčuje směr výpočtu. Takový výklad by umožnil doplnit standardní kvalitativní popis procesů o kvantitativní vztahy potřebné pro simulaci fyziologických systémů.

Netvrdíme přitom, že norma ISO 19450 již akauzální modelování v této podobě definuje. Náš návrh zkoumá možnost přiřadit existujícím vztahům další matematický význam, aniž by bylo nutné zavádět nové základní grafické prvky nebo větné konstrukce OPL. Jeho slučitelnost s původní sémantikou normy je předmětem dalšího výzkumu a konzultací se skupinou profesora Doriho.

Důležité je, že navrhovaná akauzální interpretace zůstává oddělitelná. Modely, které ji nevyužívají, mohou nadále pracovat výhradně se standardními principy OPM.

8.4 Současné využití kauzálního a akauzálního modelování

Naším cílem není nahradit dosavadní kauzální popis procesů, ale umožnit jeho souběžné využití s akauzálními vztahy. Mnohé reálné systémy totiž kombinují diskrétní události se spojitými změnami fyzikálních veličin.

Ve fyziologii může například přechod z klidu do fyzické zátěže představovat diskrétní změnu podmínek, zatímco krevní tlak, srdeční výdej nebo koncentrace kyslíku se následně mění spojitě. Navrhujeme proto zachovat standardní sémantiku OPM pro diskrétní procesy a stavové přechody a doplnit ji akauzální interpretací kvantitativních vztahů.

Podobný princip využívá jazyk Modelica, který umožňuje vytvářet hybridní modely kombinující matematické rovnice s diskrétními událostmi. Obdobně prostředí Simscape rozšiřuje možnosti Simulinku o fyzikální sítě, ve kterých jsou komponenty propojeny akauzálními vztahy [28,29].

Dalším příkladem je standard SysML Modelica Transformation organizace OMG, který umožňuje propojovat systémové modely SysML s matematickými modely v jazyce Modelica a rozlišuje kauzální signálové a akauzální fyzikální vazby [14].

Náš přístup se však snaží jít o krok dál. Nejde pouze o převod mezi dvěma modelovacími jazyky, ale o možnost vyjádřit akau-

DOI: 10.35191/medsoft_2026_1_38_merunka

zální vztahy přímo v konceptuálním modelu OPM. Modelica pak slouží jako jeden z možných nástrojů pro jejich výpočetní realizaci, nikoli jako nezbytný základ samotného modelu.

Takové řešení by umožnilo zachovat jednotný způsob modelování i u systémů, které kombinují organizační procesy, informační toky a spojité fyziologické děje.

8.5 OPD jako prostředek pro porozumění složitým matematickým modelům

Původní motivací našeho výzkumu byla snaha umožnit lékařům a fyziologům vytvářet simulační modely bez nutnosti programování. Během práce se však ukázal další, možná ještě významnější přínos. Akauzální OPD může usnadnit porozumění složitým matematickým modelům, jejichž struktura není ze samotného programového zápisu snadno čitelná.

Rozsáhlé modely v jazyce Modelica bývají obtížně srozumitelné právě proto, že vztahy mezi komponentami neurčují předem směr výpočtu. Ze spojení dvou komponent nelze přímo poznat, které veličiny budou vypočítávány z ostatních. To určí až překladač při sestavení a řešení soustavy rovnic.

Pokud bychom takový model převedli do diagramu s výhradně kauzálními vazbami, museli bychom některým vztahům uměle přiřadit směr, který jejich fyzikální podstata nevyžaduje. Výsledný diagram by pak nemusel věrně zachycovat význam původního modelu. Akauzální interpretace umožňuje zobrazit fyzikální vztahy v OPD bez tohoto zkreslení.

Přináší to tři důležité možnosti.

První je přirozené zachycení zákonů zachování. Například v uzlu fyzikální sítě musí být součet příslušných tokových veličin nulový. Taková bilanční podmínka může vyplývat přímo z propojení komponent, aniž by bylo nutné každý vztah samostatně zapisovat [28]. Tento princip je běžný v akauzálním fyzikálním modelování a představuje možné východisko pro jeho interpretaci v OPM.

Druhou možností je konceptuální zpřesňování modelu. Rozdělení rozsáhlého modelu do programových balíčků nemusí odpovídat způsobu, jakým o systému uvažuje fyziolog. OPM umožňuje vytvářet různé úrovně podrobnosti podle významu jednotlivých procesů a objektů, nikoli pouze podle organizace zdrojového kódu [3].

Třetí možností je kontrola matematické struktury modelu přímo v diagramu. Překladače obvykle oznamují problémy, například singulární soustavu, prostřednictvím rozsáhlých výpisů zpracovaných rovnic. Pro uživatele bez hlubších znalostí numerického modelování mohou být obtížně interpretovatelné. Nástroj založený na akauzálním OPD by mohl upozornit přímo u příslušného objektu, že například dvě neznámé veličiny jsou omezeny pouze jednou nezávislou rovnicí.

Taková kontrola by umožnila odhalovat některé nedostatky matematického zadání ještě před překladem modelu a současně je vysvětlovat v pojmech, kterým rozumí odborník z modelované oblasti.

Akauzální modelování v OPM proto nechápeme pouze jako cestu k automatickému generování simulací. Jeho širším cílem je vytvořit srozumitelnou konceptuální reprezentaci složitých matematických systémů, která umožní jejich společné čtení, analýzu a postupné zpřesňování.

9 Co OPM neumí

Přes uvedené přednosti má OPM také omezení, která je nutné zohlednit při jejím praktickém využití.

Prvním problémem je omezená nabídka nástrojů. Zatímco UML a BPMN podporují desítky aplikací, pro OPM jsou dostupné především tři nástroje popsané v kapitole 3: desktopový OPCAT, webový OPCloud a otevřený OpenOPM Editor, na jehož vývoji se podílí jeden z autorů [4,11]. Malý ekosystém a omezená dostupnost otevřených řešení představují překážku zejména pro vysoké školy a veřejnou správu.

Druhým omezením je nedostatek odborníků. V České republice má s OPM praktické zkušenosti jen několik lidí. Požadavek na použití této metodiky ve veřejné zakázce by proto mohl výrazně omezit počet potenciálních dodavatelů.

Třetí omezení vyplývá ze samotné normy. ISO 19450 definuje konceptuální modelování, nikoli spojitou simulaci založenou na diferenciálních rovnicích [1]. Automatický převod OPM diagramů do kvantitativních simulátorů představuje naše výzkumné rozšíření, nikoli standardní vlastnost metodiky. OPM také není nástrojem pro automatické generování produkčních informačních systémů. Jejím hlavním přínosem je možnost vytvořit zadání, jehož chování lze před implementací ověřovat.

Další dvě omezení jsme identifikovali při vlastní práci s fyziologickými modely. OPD poskytuje konceptuální, nikoli výpočetní mapu systému. Samotný diagram neukazuje pořadí řešení rovnic, algebraické smyčky ani přesné příčiny singularity soustavy. Tyto informace vznikají až při zpracování matematického modelu a vyžadují odpovídající diagnostické nástroje.

Druhým zjištěním je, že formální správnost modelu ještě nezaručuje jeho věcnou správnost. V první verzi našeho diagramu krevního oběhu byly dva prvky zapojeny paralelně, přestože ve zdrojovém modelu byly zapojeny sériově. Chyba prošla kontrolou počtu neznámých a rovnic, protože obě zapojení byla z tohoto hlediska vyvážená. Přesto představovala dva odlišné matematické modely. Pouhá kontrola struktury proto nestačí. Je nutné ověřit také chování modelu, například numerickým porovnáním s referenční simulací, jak jsme ukázali v kapitole 7.

Zůstává také oprávněná otázka, proč by se OPM měla prosadit tam, kde se dřívější pokusy o širší využití formálních metod potýkaly s obtížemi. Překážkou totiž nebyla pouze samotná notace, ale především dostupnost nástrojů a nedostatek lidí schopných s nimi pracovat.

Dnes se však podmínky v několika ohledech změnily. OPM má mezinárodní normu, na kterou lze odkazovat při specifikaci požadavků. Její textová reprezentace OPL umožňuje automatizované zpracování modelů a jejich propojení s velkými jazykovými modely [1,3]. A konečně možnost spouštět simulace přímo ve webovém prohlížeči odstraňuje nutnost instalovat specializovaný software.

Tyto výhody samy o sobě nezaručují širší přijetí OPM. Vytvářejí však příznivější podmínky pro její praktické využití a umožňují, aby konceptuální modely kontrolovali a zkoušeli také odborníci, kteří nejsou specialisty na modelování.

10 Návaznost a další směřování

V loňském sborníku jsme se zabývali možnostmi formalizace legislativních požadavků ve zdravotnictví a navrhli OPM jako metodiku pro jejich konceptuální modelování. Diagramy jsme tehdy označili za „hierarchicky uspořádané (a potenciálně simulovatelné)“ [18]. Dnes již můžeme slovo „potenciálně“ vypustit.

Naše předchozí zkušenosti s elektronickým receptem ukázaly, že vizuální modelování propojené se simulací usnadňuje spolupráci lékařů, analytiků a dalších odborníků [18]. Současný výzkum na tyto zkušenosti navazuje a přináší možnost automatického převodu OPM diagramů do simulací spustitelných přímo ve webovém prohlížeči. OPM navíc dnes poskytuje standardizovaný základ v podobě ISO 19450 a textovou reprezentaci OPL vhodnou pro automatizované zpracování.

Další výzkum a vývoj zaměřujeme na několik oblastí, které zatím představují plánované kroky, nikoli dokončené výsledky.

DOI: 10.35191/medsoft_2026_1_38_merunka

Zpětný převod existujících modelů do OPM. V současnosti dokážeme z OPM diagramu vytvořit spustitelný model. Opačný postup, tedy automatické načtení rozsáhlých existujících knihoven a jejich zobrazení v podobě OPD, zatím nemáme k dispozici. Právě tato možnost by mohla zásadně usnadnit porozumění složitým fyziologickým modelům, jak jsme naznačili v kapitole 8.5. Dosavadní numerické výsledky se týkají modelů vytvořených naším postupem.

Propojení modelu s grafickou podobou simulátoru. Pracujeme na jednotném způsobu propojení modelu s výtvarně připravenými schématy prostřednictvím názvů instancí a grafických vrstev. Výsledkem by mohly být interaktivní fyziologické simulátory s názorným lékařským zobrazením, nikoli pouze s diagramy a grafy časových průběhů.

Knihovna Modelica pro OPM. Připravujeme specializovanou knihovnu komponent podporující náš přístup k převodu OPM modelů do kvantitativních simulací. První verzi plánujeme dokončit do března 2027.

Další odborné ověření akauzálního přístupu. Výsledky výzkumu z kapitoly 8 chceme podrobněji představit na 17. mezinárodní konferenci Modelica v Praze v září 2027, kde bude možné diskutovat navrhovanou interpretaci s odborníky na akauzální modelování.

Naším dlouhodobým cílem je, aby OPM nesloužila pouze k vytváření nových modelů, ale také k porozumění již existujícím složitým systémům a jejich ověřování. Další vývoj proto směřuje především k lepšímu propojení konceptuálních diagramů, matematických modelů a interaktivních simulací.

Dosavadní zkušenosti s digitalizací české veřejné správy, včetně problémů při zavádění digitálního stavebního řízení, ukazují rizika vývoje informačních systémů bez jednotné architektury a důsledného ověřování požadavků. Výsledkem může být rostoucí technický dluh, roztříštěnost systémů, závislost na jednotlivých dodavatelích a ztráta důvěry veřejnosti.

Od března 2026 je přitom k dispozici česká verze mezinárodní normy ČSN ISO 19450, která nabízí standardizovaný způsob konceptuálního modelování [5]. Navrhujeme proto zvážit její začlenění do metodických a legislativních pravidel pro přípravu kritických informačních systémů státu. Smyslem není předepsat jediný technologický postup, ale zajistit, aby požadavky na tyto systémy byly jednoznačně vyjádřené, kontrolovatelné a jejich chování bylo možné ověřovat ještě před realizací.

Nový zákon o kybernetické bezpečnosti č. 264/2025 Sb. a zákon o kritické infrastruktuře č. 266/2025 Sb. posilují bezpečnost a odolnost důležitých služeb. Tyto zákony však nenahrazují systematické inženýrské postupy pro návrh, ověřování a řízení životního cyklu informačních systémů. U kritických systémů by proto mělo být standardem vytvoření formálně definovaného konceptuálního modelu, jehož vlastnosti a chování lze nezávisle kontrolovat.

Navrhujeme pět navazujících opatření:

1. Modelovat požadavky před zadáním veřejné zakázky, například podle ČSN ISO 19450, aby bylo možné ověřit jejich úplnost a vzájemnou konzistenci.
2. Ověřovat procesy simulací za účasti odborníků z praxe a kontrolovat jejich soulad s právními i provozními požadavky.
3. Zavést nezávislé posouzení modelu a dodržení stanovených standardů před uvedením kritického systému do provozu.
4. Jasně upravit práva k dokumentaci, modelům a výslednému softwaru a stanovit požadavky na interoperabilitu, přenositelnost a omezení závislosti na jediném dodavateli.
5. Stanovit odpovědnost za nedodržení předepsaných inženýrských postupů, včetně přiměřených kontrolních a sankčních mechanismů.

Cílem těchto opatření není přidávat další administrativu, ale přesunout podstatnou část kontroly do fáze, kdy lze chyby napravovat s výrazně nižšími náklady. Spolehlivá digitalizace veřejné správy vyžaduje nejen bezpečný provoz informačních systémů, ale především kvalitní, srozumitelné a ověřitelné zadání. Právě v této oblasti může OPM nabídnout důležitý příspěvek.

11 Závěr

Object-Process Methodology (OPM) je mezinárodně standardizovaná metodika konceptuálního modelování, která umožňuje zachytit strukturu, chování i stavy objektů v jediném typu diagramu a jeho rovnocenné textové podobě.

Na příkladu elektronické žádanky jsme ukázali, že OPM dokáže odhalit nedostatky, které při běžném čtení právního předpisu nemusí být zřejmé. Model odvozený z právní úpravy se soustředí na předávání žádanky, ale nezachycuje celý klinický proces od jejího vystavení až po získání a vyhodnocení výsledku. Právě propojení procesů se stavy objektů umožňuje takové chybějící návaznosti rozpoznat.

Naším hlavním příspěvkem je možnost automaticky převést OPM diagram do simulace spustitelné přímo ve webovém prohlížeči. Konceptuální model se tím stává nejen prostředkem komunikace, ale také ověřitelným návrhem, jehož předpokládané chování lze simulací potvrzovat nebo vyvracet. Navrhovaná akauzální interpretace OPM rozšiřuje tento přístup také na kvantitativní fyziologické modely. Její použitelnost jsme doložili na dvou modelech, zatím však představuje výzkumný návrh, nikoli dokončené rozšíření normy.

Hlavní přínos OPM tak spatřujeme v možnosti přejít od zadání, které lze pouze číst a připomínkovat, k modelu, jehož chování lze ještě před realizací systému skutečně ověřovat.

OBJECT-PROCESS METHODOLOGY

Abstract

Object-Process Methodology (OPM) is a conceptual modeling methodology that was adopted as the international standard ISO 19450 in 2024. Its main advantage is the ability to represent the structure, behavior, and states of modeled objects within a single type of diagram. Each diagram also has an equivalent textual representation in structured natural language that is understandable to domain experts while supporting automated processing. The resulting model can also be executed, and its behavior can be verified through simulation. This makes the model not only a means of communication but also a tool for assessing the correctness and completeness of system requirements and validating the outputs of large language models. We demonstrate the practical application of OPM using the example of an electronic medical referral, whose conceptual model reveals significant shortcomings in the relevant legal framework.

Keywords

Object-Process Methodology, OPM, ISO 19450, conceptual modelling, electronic requisition, eHealth, Modelica

Literatura

[1.] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO 19450:2024. Automation systems and integration – Object-Process Methodology. 1. vyd. Ženeva: ISO, 2024. Dostupné z: https://www.iso.org/standard/84612.html

[2.] DORI, Dov. Object-Process Methodology: A Holistic Systems Paradigm. Berlin, Heidelberg: Springer, 2002. ISBN 978-3-540-65471-1. DOI: https://doi.org/10.1007/978-3-642-56209-9

[3.] DORI, Dov. Model-Based Systems Engineering with OPM and SysML. New York: Springer, 2016. ISBN 978-1-4939-3294-8. DOI: https://doi.org/10.1007/978-1-4939-3295-5

[4.] OPCloud User Guide. Version 9.2 [online]. Duben 2026. Dostupné z: https://objekty.cz/OPM/OPCloud/2026-04%20OPCloud%20User%20Guide%20v9.2.pdf

[5.] ČSN ISO 19450 (18 4321). Automatizační systémy a integrace – Objektově procesní metodika. Praha: Česká agentura pro standardizaci, 2026. Katalogové číslo 523021. Vydáno v březnu 2026. Dostupné z: https://csnonlinefirmy.agentura-cas.cz/html_nahledy/18/523021/523021_nahled.htm

[6.] CASEBOLT, Jason M., JBARA, Ahmad a DORI, Dov. Business process improvement using Object-Process Methodology. Systems Engineering. 2020, 23(1), 36–48. DOI: https://incose.onlinelibrary.wiley.com/doi/10.1002/sys.21499

[7.] MORDECAI, Yaniv, JAMES, Nicholas K. a CRAWLEY, Edward F. Object-Process Model-Based Operational Viewpoint Specification for Aerospace Architectures. In: 2020 IEEE Aerospace Conference. IEEE, 2020, s. 1–15. DOI: https://doi.org/10.1109/AERO47225.2020.9172685

[8.] DORI, Dov a CHODER, Mordechai. Conceptual Modeling in Systems Biology Fosters Empirical Findings: The mRNA Lifecycle. PLoS ONE. 2007, 2(9), e872. DOI: https://doi.org/10.1371/journal.pone.0000872

[9.] SOMEKH, Judith, CHODER, Mordechai a DORI, Dov. Conceptual Model-Based Systems Biology: Mapping Knowledge and Discovering Gaps in the mRNA Transcription Cycle. PLoS ONE. 2012, 7(12), e51430. DOI: https://doi.org/10.1371/journal.pone.0051430

[10.] PELEG, Mor, SOMEKH, Judith a DORI, Dov. A methodology for eliciting and modeling exceptions. Journal of Biomedical Informatics. 2009, 42(4), 736–747. DOI: https://doi.org/10.1016/j.jbi.2009.05.003

[11.] OpenOPM Editor [software, online]. Verze 0.0.1. GitHub: pavelberanek91. Dostupné z: https://github.com/pavelberanek91/Object-Process-Generator

[12.] OBJECT MANAGEMENT GROUP. OMG Unified Modeling Language (OMG UML). Version 2.5.1 [online]. 2017. Dostupné z: https://www.omg.org/spec/UML/2.5.1/

[13.] OBJECT MANAGEMENT GROUP. OMG Systems Modeling Language (OMG SysML). Version 1.7 [online]. 2024. Dostupné z: https://www.omg.org/spec/SysML/1.7/

[14.] OBJECT MANAGEMENT GROUP. SysML-Modelica Transformation Specification. Version 1.0 [online]. Listopad 2012. Dokument OMG formal/12-11-09. Dostupné z: https://www.omg.org/spec/SyM/1.0/

[15.] OBJECT MANAGEMENT GROUP. OMG Systems Modeling Language (OMG SysML). Version 2.0 [online]. 2025. Dostupné z: https://www.omg.org/spec/SysML/2.0/

[16.] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION a INTERNATIONAL ELECTROTECHNICAL COMMISSION. ISO/IEC 19510:2013. Information technology – Object Management Group Business Process Model and Notation. 2013. Dostupné z: https://www.iso.org/standard/62652.html

[17.] KNOTT, Roger, MERUNKA, Vojtěch a POLAK, Jiri. The BORM methodology: a third-generation fully object-oriented methodology. Knowledge-Based Systems. 2003, 16(2), 77–89. DOI: https://doi.org/10.1016/S0950%E2%80%937051(02)00075%E2%80%938

[18.] VOJTĚCH, Adam, MERUNKA, Vojtěch a KOFRÁNEK, Jiří. „Od umění k algoritmu" – formalizace a modelování legislativních procesů se zaměřením na zdravotnictví. In: MEDSOFT 2025 [online]. 2025, s. 1–15. DOI: https://doi.org/10.35191/medsoft_2025_1_37_vojtech

[19.] THE OPEN GROUP. ArchiMate® 3.2 Specification [online]. The Open Group Standard, dokument C226. 2022. Dostupné z: https://pubs.opengroup.org/architecture/archimate32-doc

[20.] SCHEER, August-Wilhelm. Architecture of Integrated Information Systems: Foundations of Enterprise Modelling. Berlin, Heidelberg: Springer, 1992. DOI: https://doi.org/10.1007/978-3-642-97389-5

[21.] INTERNATIONAL ORGANIZATION FOR STANDARDIZATION a INTERNATIONAL ELECTROTECHNICAL COMMISSION. ISO/IEC 15909–1:2019. Systems and software engineering – High-level Petri nets – Part 1: Concepts, definitions and graphical notation. 2. vyd. 2019. Dostupné z: https://www.iso.org/standard/67235.html

[22.] ČESKO. Zákon č. 325/2021 Sb., o elektronizaci zdravotnictví, ve znění pozdějších předpisů. In: Sbírka zákonů České republiky. 2021, částka 143. Zejména § 31 a § 31a. Dostupné z: https://www.zakonyprolidi.cz/cs/2021%E2%80%93325

[23.] ČESKÁ LÉKAŘSKÁ SPOLEČNOST JANA EVANGELISTY PURKYNĚ. Pracovní skupina předsednictva ČLS JEP pro elektronické zdravotnictví. Požadavek ČLS JEP na elektronické žádanky [nepublikovaný pracovní návrh]. 7. října 2026.

[24.] VOGELSANG, Andreas. From Specifications to Prompts: On the Future of Generative Large Language Models in Requirements Engineering. IEEE Software. 2024, 41(5), 9–13. DOI: https://doi.org/10.1109/MS.2024.3410712.

[25.] HOU, Xinyi a kol. Large Language Models for Software Engineering: A Systematic Literature Review. ACM Transactions on Software Engineering and Methodology. 2024, 33(8), článek 220, s. 1–79. DOI: https://doi.org/10.1145/3695988

[26.] BERÁNEK, Pavel a MERUNKA, Vojtěch. Velké jazykové modely a agentní systémy v medicíně: principy, současné aplikace a perspektivy. In: MEDSOFT 2025 [online]. 2025, s. 1–11. DOI: https://doi.org/10.35191/medsoft_2025_1_37_beranek

[27.] KOFRÁNEK, Jiří, STRAUSS, Eduard a PUGACHOV, Michail. Svelte bodylight: nová technologie tvorby webových simulátorů. In: MEDSOFT 2026 [online]. 2026.

[28.] MATHWORKS. Basic Principles of Modeling Physical Networks. In: Simscape Documentation [online]. R2026b. Dostupné z: https://www.mathworks.com/help/simscape/ug/basic-principles-of-modeling-physical-networks.html

[29.] MODELICA ASSOCIATION. Modelica Language Specification. Version 3.7 [online]. 19. května 2026. Dostupné z: https://specification.modelica.org/maint/3.7/MLS.html

DOI: 10.35191/medsoft_2026_1_38_merunka

Kontakty

Ing. Mgr. Pavel Beránek, MBA
Přírodovědecká fakulta
Univerzita Jana Evangelisty Purkyně
v Ústí nad Labem
Pasteurova 3544/
400 96 Ústí nad Labem

Provozně ekonomická fakulta
Česká zemědělská univerzita v Praze
Kamýcká 129, 165 00 Praha-Suchdol

+420 475 286 724
pavel.beranek@ujep.cz
https://ki.ujep.cz/cs/personalni-slozeni/pavel-beranek/

doc. Ing. Vojtěch Merunka, Ph.D.
Provozně ekonomická fakulta, Česká zemědělská univerzita v Praze, Kamýcká 129, 165 00 Praha-Suchdol
+420 224 382 272
merunka@pef.czu.cz
https://vojtech.merunka.eu/

doc. MUDr. Jiří Kofránek, CSc.
Laboratoř biokybernetiky
Ústav patologické fyziologie
1. lékařská fakulta, Univerzita Karlova
U Nemocnice 5, 128 53 Praha 2
+420 444 686 868
jiri.kofranek@lf1.cuni.cz