

DIGITÁLNÍ A INFORMAČNÍ AGENTURA_

Export z Národní architektury eGovernmentu ČR

Obsah

Best practices a principy modelování	3
<i>Tvorba modelu obecně</i>	3
<i>Elementy</i>	5
<i>Vazby</i>	5
<i>Atributy a doplňkové informace</i>	5
<i>Hlediska a pohledy</i>	6
<i>Členění modelu</i>	14

Best practices a principy modelování

Jak je z kapitoly výše patrné, tvorba modelu nespočívá v kreslení diagramů, nýbrž v budování systému elementů a vazeb, které mají definovaný konkrétní význam podle předepsaných pravidel. Tato pravidla umožňují čtenáři pochopení obsahu vizualizovaných částí modelu, aniž by modelář musel všechny informace vypisovat v textové podobě. Pravidla NAR a této metodiky stanovují nejen význam elementů a vazeb daného typu, ale říkají i, jak lze elementy mezi sebou propojovat, a definují jejich možné využití. Dále ukazují vazby v různých perspektivách. Tím, že model umožňuje použít jeden element či vazbu pro více pohledů, je eliminováno duplikování informací, což bývá zásadní problém pro jeho dlouhodobou udržitelnost a rozvoj.

Modelování dle předem dohodnutých pravidel zajišťuje, že uložené informace v modelu budou pro čtenáře pochopitelné a model bude možné dlouhodobě udržovat a rozvíjet.

Tvorba modelu obecně

Obecně platné principy pro dobrý model jsou popsány níže.

Princip	Popis
Budte poctiví	Nazývejte každou věc z reality při jejím zachycování do modelu pravým jménem, vyhýbejte se opisům "úřednického newspeaku".
Budte věrní	Usilujte o to, aby kterýkoli diagram, v jakékoli míře podrobnosti byl vždy co nejvěrnějším obrazem Vašeho porozumění realitě. To znamená v praxi mimo jiné používání prvků modelovacího jazyka pro to, pro co jsou určeny. Tedy aplikační komponenty pro aplikace a ne pro procesy, nenahrazovat business služby aplikacemi, dát si pozor na rozdíl mezi rolemi a aktéry atd.
Budte výstižní	Modelujte a popisujte realitu tak, aby bylo dodrženo pravidlo „Dokonalosti je dosaženo, když není co odebrat, ne když není co přidat“. Model by měl být tak jednoduchý, jak je možné, v kontextu očekávaných čtenářů a metodiky. Snažte se, aby model na dané úrovni detailu neobsahoval v jiné podobě stejné informace vícekrát. Informace se mohou opakovat při modelování stejné oblasti na nižší/vyšší úrovni detailu – jedná se o popis stejné části reality –, ale musí být poplatné dané úrovni detailu – viz další princip.
Dávejte popis ke každému prvku	Ke každému prvku (strukturní prvky, element, vazba, pohled) vedle jeho názvu navíc doplňte vyčerpávající slovní popis (do příslušného pole – např. Note, Documentation apod.). Pomůže to čtenáři pochopit vaše myšlenky a dále zamezí možnosti nesprávného pochopení. Zejména u diagramů je vhodné čtenáři poskytnout popis jako průvodce zobrazenými elementy a vazbami, aby byly zřejmé prezentované myšlenky. Popisy také zvyšují hustotu informací ve výstupu z modelu a čtenáři pomáhají lépe pochopit kontext.
Čleňte model dle úrovní detailu	Model by měl být členěn stejným způsobem přes celý rozsah modelované reality. Je vhodné definovat jasně úrovně detailu a ty dodržet. Např. L0, L1, L2, jak je popsáno níže v této metodice (kap. 3.6.4). Nejvyšší úroveň by měla ukazovat celý rozsah modelované reality, detailnější úrovně se mohou zaměřit jen na zajímavé části. Lze využít hierarchického strukturování modelu v katalozích elementů a digramů zaměřených na danou úroveň detailu. To implicitně zlepšuje čtivost dokumentace z modelu, neboť jde přirozenou cestou od celku k detailům („top-down“).
Budte konzistentní a dodržujte standardy napříč modelem	Dodržujte stejný způsob popisu přes všechny části dané oblasti a úrovně detailu modelu. Významně to usnadní sdílení informací v modelu s ostatními, zastupitelnost modelářů a také to umožní formální kontrolu kvality modelu. Mezi důležitá rozhodnutí patří i jazyk, v němž bude model tvořen. Jakmile je potenciálně možné, že bude model tvořit a číst i cizojazyčný uživatel, zvažte tvorbu modelu v angličtině. V modelech pro státní správu dle NAR je čeština vždy primárním jazykem.

Princip	Popis
Používejte správné prvky	Dobře rozmyslete, jaký typu elementu použijete pro uložení dané informace. Např. aplikační komponenty ať reprezentují aplikace a ne procesy, nenahrazujte business služby aplikacemi, rozlišujte mezi rolemi a aktéry atd. Buďte co nejlépe seznámeni s modelovacím standardem (vč. metamodelu). Cílem je předejít duplikaci informací a mít informace uložené na očekávaných místech.
Zaveďte a dodržujte jmenné konvence a identifikátory prvků modelu	Standardizaci, jednoznačnosti a pochopitelnosti pomáhá jednotný způsob označování prvků modelu a používání ustálených označení pro jednotlivé prvky. Jmenná konvence umožňuje zavést do názvu prvku doplňkové informace, které pak není třeba hledat v attributech prvku. Systém identifikátorů pak zajistí, že se lze na konkrétní prvek odkázat, aniž by bylo třeba vypisovat jeho celý název, či dokonce kontext.
Modelování není programování	I když je samozřejmě vhodné maximálně využít možností nabízené metodikou či standardem, je nutné co nejvíce přiblížit výstup čtenáři. Sebelepší metodika nepokrývá všechny potřebné situace a informace, které v je v modelu nutné uvést. Proto v případě, že by forma informace v modelu byla na úkor pochopitelnosti, je možné volit náhradní (byť nekonfliktní) způsoby zanesení informace do modelu. Vždy je např. možné popsat potřebné detaily textově do popisu na daném místě. Na diagramy je možné doplnit doplňující poznámky přímo, pod elementy je možné přidat další vysvětlující diagramy, matice apod. Buďte v tomto kreativní a mějte na mysli pohodlí pro čtenáře, neboť bez čtenáře je model zbytečný.
Počítejte s budoucím rozšířením modelu	Model strukturujte více robustně, než je aktuálně potřeba, abyste umožnili jeho snadné průběžné rozšiřování bez nutnosti refaktorování (změny struktury či formy). Je vhodné rozdělit modelovaný systém na moduly, které jsou stejným způsobem použity ve všech vrstvách/doménách modelu (nebo jsou alespoň konzistentní v rámci jedné vrstvy/domény). Tyto moduly pak slouží k logickému členění obsahu modelu a umožňují tvořit systematicky a jednoznačné identifikátory prvků.
Přizpůsobujte výstupy pro konkrétní čtenáře	Při modelování je nutné myslet na čtenáře. Jednotlivé výstupy z modelu by měly být přizpůsobené čtenáři daného hlediska tak, aby dostal informaci, kterou potřebuje, ve formě, kterou snadno pochopí, ideálně v podobě uceleného příběhu. Je vhodné strukturovat model už s ohledem na způsob vytváření výstupů pro čtenáře (export modelu, generovaná dokumentace, ručně udržovaná dokumentace se vstupy z modelu apod.). Většinou je pro dosažení optimálního výsledku potřeba přizpůsobit strukturu modelu podle očekávané formy výstupu pro čtenáře.
Ambici pro verzování modelu nastavte na začátku	Definujte si potřeby pro udržování časových řezů modelu. Přístupů a potřeb verzování je mnoho, od jedné verze po schopnost okamžitého pohledu do minulosti i budoucnosti. Pro dlouhodobou udržitelnost přehledu o verzích a platnosti obsahu modelu je zásadní si tuto ambici na začátku nastavit a podle toho pak nastavit proces tvorby a údržby modelu a výstupů.
Automatizujte tvorbu výstupů	Je vhodné mít schopnost časté tvorby dohodnutých výstupů s co nejmenší námahou. Ideální je zcela automatizovaný způsob tvorby generování výstupů. To umožňuje iterativně ladit výstup v dané verzi do nejlepší kvality, eliminuje lidské chyby a také umožňuje snadno identifikovat rozdíly mezi verzemi. Při tvorbě šablon dbejte na to, aby výstup obsahoval co nejméně prázdného místa. Výstup tak bude působit mnohem kompaktněji a nebude mít tolik stránek, což je zásadní pro jeho čtivost.
Zajistěte si podporu od organizace	Podpora se získá zejména tím, že výstupy architektů jsou vytvářeny tak, aby byly užitečné konkrétním lidem pro řešení jejich konkrétních problémů. Modelování na úrovni podnikové architektury vyžaduje získávání co nejpřesnějších informací prakticky od všech týmů v organizaci. Aby bylo možné tyto informace získat a ideálně znát i jejich očekávaný vývoj, je potřeba, aby stakeholderi (majitelé těchto informací) věděli, že mají poskytnout součinnost a že to je pro ně přidaná hodnota. V budoucnu budou moci čerpat z modelu přesné a provázané informace v celofiremním kontextu. Podpora od organizace spočívá zejména v adopci týmu Enterprise architektury (EA) a jeho procesů do organizace, zejména na nejvyšších úrovních jejího řízení. Tým EA by měl být aktivní a snažit se být viditelný v příslušných diskusích. Dále je pro adopci výstupů EA dobré, když jsou přizpůsobené čtenářům (viz výše).

Výše popsané principy lze uplatnit na jednotlivé prvky modelu. Základními prvky modelu jsou elementy, vazby, atributy a pohledy. Dále je detailněji popsáno, které principy jsou na nich uplatněné a jakou jsou uplatněny

formou.

Elementy

Element modelu reprezentuje jednu konkrétní část reality z pohledu její struktury či chování a popisuje její vlastnosti. Je to základní stavební prvek modelu.

Při modelování elementů doporučujeme mít na paměti následující:

Zvolit správný typ elementu	Respektovat kontext z hlediska metodiky (arch. doména, úroveň detailu)
Název	Výstižný a co nejkratší
Jednoznačný identifikátor	Zapadající do pravidel pro daný kontext (stereotyp, arch. doména, modul)
Dostatečný popis	Popis významně pomůže pochopení významu elementu, a to nejen pro čtenáře, ale i pro modeláře. Popis např. umožní rozhodnout, zda je třeba zakládat nový element, či rozšířit působnost existujícího.
Relevantní pohledy	Rozmyslete do jakých pohledů element patří přidat. Obvykle to definuje metodika a nástroj jej může do nich zahrnout automaticky na základě jeho umístění či stereotypu. Je ale vhodné se zamyslet, zda např. pro element nezaložit diagram ukazující kontext jeho použití pro lepší pochopení čtenářem.
Zanořování elementů	V případě této metodiky je doporučeno vytvářet lineární seznamy položek pro snadnou katalogizaci. Pro vizuální zapouzdřování je vhodné používat diagramy (včetně elementů typu Boundary), případně korektní vazby a elementy držet uložené lineárně. Cílem je zajistit předvídatelný způsob evidence elementů ve struktuře modelu a podpořit jednotný způsob stavění výstupní dokumentace.

Vazby

Propojuje elementy a říká jakým způsobem spolu dané elementy souvisejí. Pro vazbu jsou zásadní následující aspekty:

Typ	Musí být v souladu s metamodellem příslušné arch. domény a úrovní detailu)
Název	Zvážit, zda je třeba a neváhat jej doplnit, pokud to pomůže při čtení modelu na příslušných pohledech. U názvu je možné čtenáři indikovat směr čtení významu vazby mezi dvěma elementy.
Popis vazby	Není na první pohled zásadní, ale může významně pomoci při určování opodstatněnosti nové vazby mezi stejnými elementy. U vhodně strukturovaných dokumentů popis vazby pomůže významně k pochopení myšlenek a informací v modelu.
Směrnost vazby	Obvykle definovaná metodikou. Pozor na správný směr.
Vlastnosti koncových bodů vazby	V případě datového modelování je velmi přínosné doplnit k danému koncovému bodu vazby.

Atributy a doplňkové informace

Elementy, vazby i strukturní prvky v modelu mohou nést doplňkové informace, které je často vhodné držet ve strukturované podobě. Tyto údaje jsou označeny jako atributy. Základními atributy jsou název a popis, k nim se ale přidávají další. Atributy lze rozřítit do skupin dle využití (profilů).

Základní atributy a jmenná konvence

Nejdůležitějšími a nejčastěji použitými atributy elementů i vazeb jsou název, popis a jednoznačný identifikátor.

Název

Název prvku by měl být lidsky čitelný, a proto je důležité zvolit vhodný jazyk. Dále by měl být co nejkratší a výstižný. Tato metodika v souladu s ArchiMate definuje pro jednotlivé typy elementů formu jakou by měl název mít:

- Behaviorální elementy musí být nazývány průběhovým časem (musí být patrná činnost/chování) (na příklad "Poskytování katalogu služeb", "Editování údajů". "Prodej zboží").
- Pasivní elementy jsou nazývány podstatným jménem ("Dokument", "Faktura").
- Aktivní elementy jsou pojmenovány podstatným jménem ("Jan Novák", Oddělení fakturace, Zákon 111/2009 Sb.).

Technický identifikátor

Technický identifikátor slouží pro jednoznačné označení prvku modelu. Obvykle není využíván mimo technický kontext modelovacího nástroje.

Globální identifikátor

Pro účely jednoznačné identifikace prvku v rámci modelování celé státní správy je očekávané zavedení *globálního identifikátoru*. Globální identifikátor nenahrazuje technický identifikátor, jedná se o další údaj.

Tento identifikátor by byl prvku přiřazován nejspíše po importu modelu úřadu do plánovaného centrálního úložiště.

Atribut bude nutné u daného prvku udržovat i lokálně po opětovném přenosu z centrálního úložiště do místního modelu. Takto budou označené zejména všechny sdílené prvky modelů.

Klasifikační atributy

Důležitým obecným profilem je Klasifikace. Klasifikační atributy slouží k jednoznačnému zařazení prvku modelu v kontextu požadované struktury, určení a umístění v centrálním úložišti modelů DIA. Jsou obvykle realizované pomocí Tagů (záleží ale na nástroji).

Klasifikační atributy jsou součástí metainformací o modelu (viz kap. 3.6.3).

Klasifikační atributy přes všechny prvky modelů jsou uvedené v externím dokumentu [DIA Klasifikace modelů - klasifikační atributy - final - 241022.xlsx](#)

Hlediska a pohledy

Hlediska

Pro potřeby sladění očekávání čtenářů a tvůrců modelu a pro tvorbu výstupů z něj je potřeba správně

rozhodnout, jak vizualizovat informace v modelu uložené (viz princip Přizpůsobujte výstupy pro konkrétní čtenáře). K tomu pomáhá definice Hledisek popisujících, co daný čtenář očekává, že najde v pohledech na model. Jedno hledisko může být realizované na více pohledech, ale čtenář by měl vždy být schopný najít pro sebe zajímavé informace v plném potřebném kontextu.

V NAR a této metodice pro zjednodušení vždy jednomu hledisku zainteresovaných odpovídá jedna definice pohledu typu Diagram, která pak může mít v jednom modelu více výskytů pohledů, například v různé míře podrobnosti (L0 až L2) nebo se zaměřením na některý konkrétní koncept. Artefakty typu diagram vycházejí ze společných artefaktů typu Katalog.

Katalogy jsou pro každý modelovaný koncept vždy jednotné a unikátní, objektové. Na rozdíl od Diagramů se u Katalogů neuplatní Hlediska.

Hlediska pro očekávané čtenáře definuje NAR:

https://archi.gov.cz/nar_dokument:ramec_obsahu_a_vystupu_architektur#hlediska_a_definice_pohledu_narodni_architektury_vs_cr

Druhy pohledů

Pohledy v notaci ArchiMate jsou klíčové pro pochopení různých aspektů podnikové architektury. Každý pohled se zaměřuje na specifickou část architektury a poskytuje různé úhly pohledu na systém. Byznys pohledy se zaměřují na byznys procesy, role, aktéry a produkty. Pomáhají pochopit, jak byznys funguje a jaké jsou jeho hlavní komponenty a vztahy mezi nimi. Aplikační pohledy se zaměřují na aplikační komponenty a jejich interakce. Ukazují, jak aplikace podporují byznys procesy a jak spolu aplikace komunikují. Technologické pohledy se zaměřují na technologickou infrastrukturu, včetně serverů, sítí a dalších technických komponent. Pomáhají pochopit, jak je technologická infrastruktura uspořádána a jak podporuje aplikační a byznys vrstvy. Motivační pohledy se zaměřují na cíle, principy a požadavky, které řídí architekturu. Pomáhají pochopit, proč je architektura navržena určitým způsobem a jaké jsou její hlavní cíle.

Pohledy zobrazují (vizualizují) vybrané části modelu tak, aby naplnily hlediska jeho čtenářů.

Základními typy pohledů jsou:

- Diagram – zobrazuje graficky vybrané elementy, jejich atributy a vazby mezi nimi. Jsou nejobvyklejším druhem pohledu.
- Matice – ukazuje propojenost dvou skupin elementů. Na osách X a Y jsou výčty elementů z vybrané části modelu a na průsečících mezi nimi je indikace existence vazby daného typu a směru. Obvykle jsou použité pro vizualizaci mapování.
- Katalog – je výčet elementů v podobě tabulky. U každého elementu je vypsaná vybraná množina atributů. Katalogový výpis může být uživatelsky filtrovatelný, dle zvolené formy vizualizace.

Využitelné typy pohledů definuje metodika, zároveň určuje jejich roli při tvorbě modelu.

Diagramy

U diagramů je nejdůležitější, aby:

1. Co nejpřesněji vyjadřovaly sdělení (příběh) svého autora, nositele myšlenky;
2. Co nejlépe posloužily potřebě svého čtenáře, úměrně jeho specifickým možnostem vnímání a porozumění.

Dále je třeba věnovat největší pozornost počtu a rozmístění prvků na nich. Správné rozmístění, zarovnání a celková uklizenost diagramu naprosto zásadně zlepšuje schopnost pochopení myšlenek a informací.

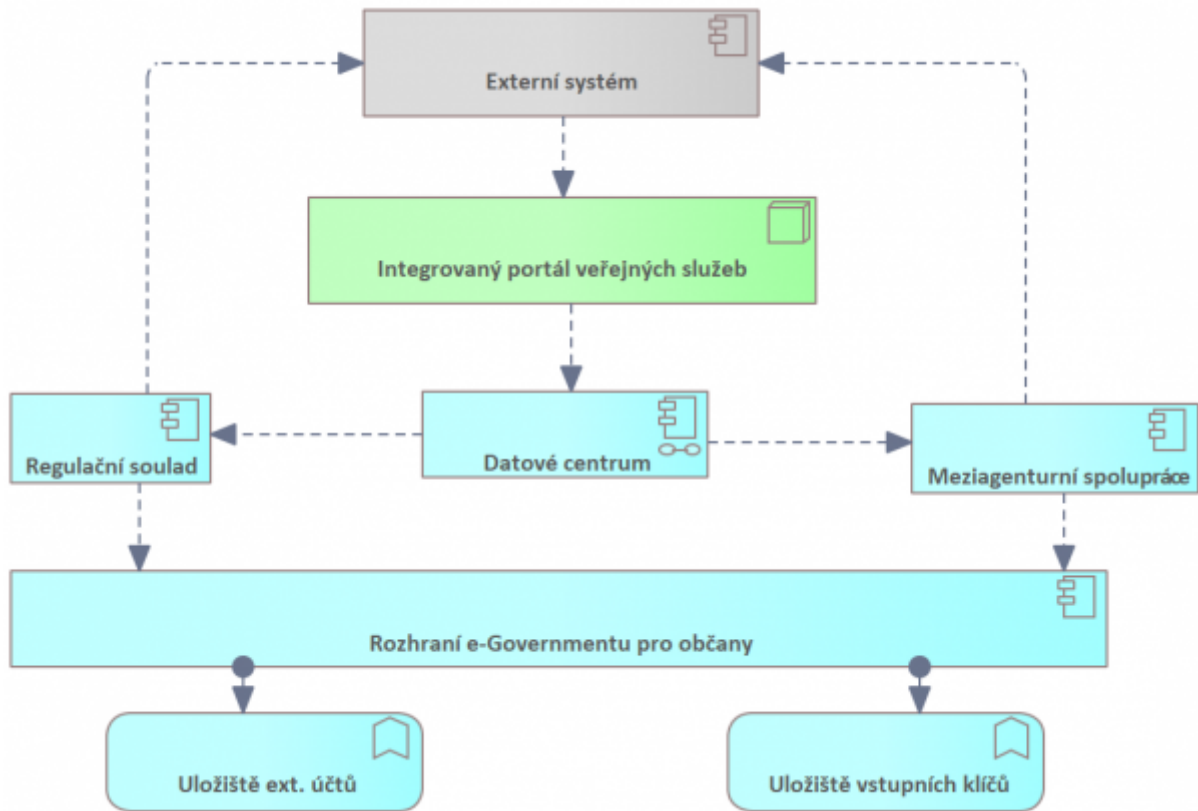
Pro diagram jsou zásadní následující aspekty:

Formální

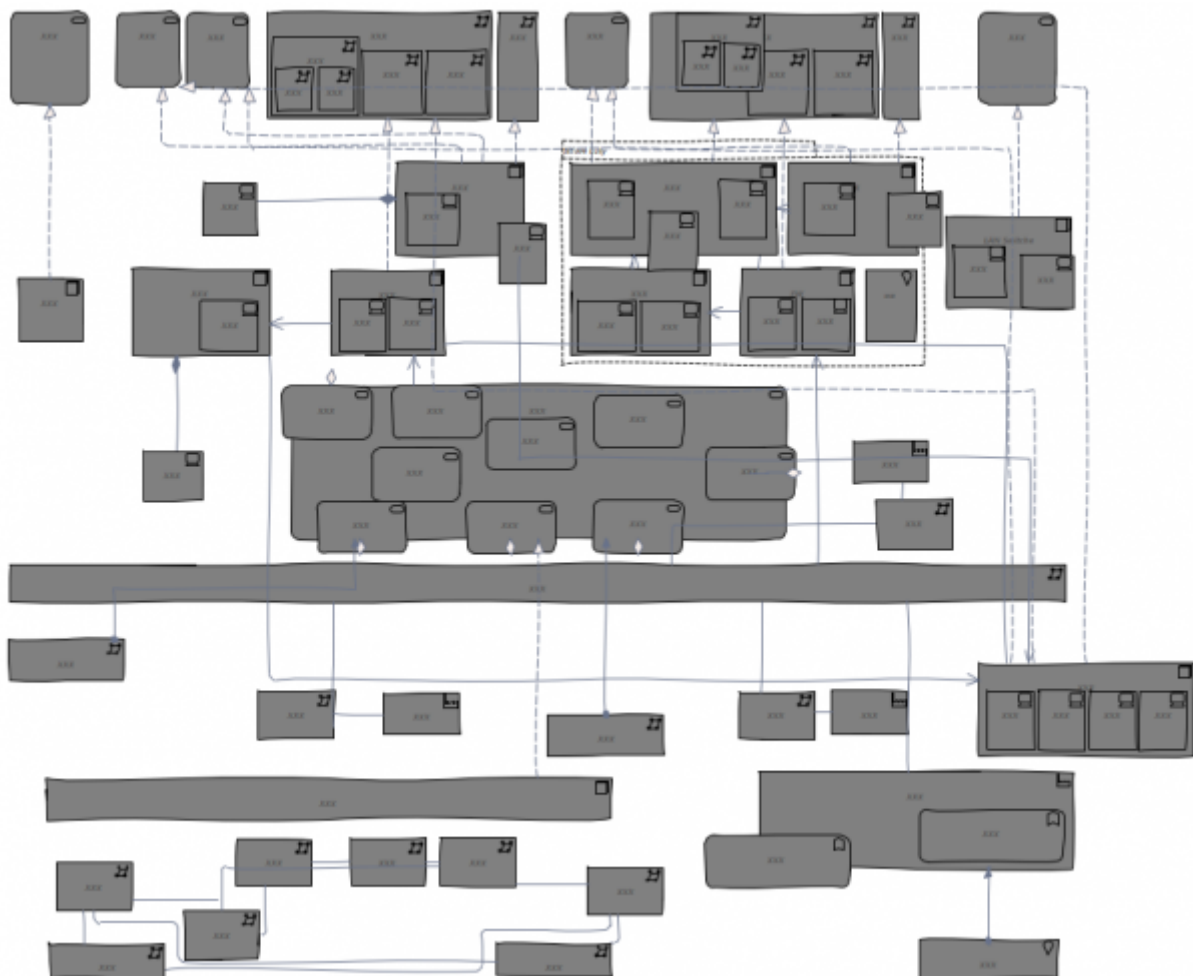
Soulad s metamodelem	Na diagramu by měly být pouze elementy, vazby a atributy předepsané metamodelem. Zásadní je nesnažit se na diagram zanést všechny informace najednou - nejspíše porušíte metodická pravidla a diagram půjde velmi obtížně naformátovat, číst i aktualizovat
Barvy a legenda	metodika předepíše barevnost elementů a vazeb. Přesto je silně doporučeno doplnit vždy diagram o legendu, která zdůrazní barevný kód. Povinně vždy, pokud děláme složitější barevné schéma, než je zvyklost ArchiMate (žlutý business, modré aplikace, zelená infrastruktura)

Vizualizace

Čitelnost	Diagram by měl být snadno čitelný v podobě, v jaké je zobrazen ve výstupu (obvykle dokument formátu A4, či zobrazení na displeji, zařazení do jednoho slide v PowerPointu). Platí pravidlo, že čtenář by měl být schopen význam a hlavní myšlenky diagramu pochopit do 30 sekund. K tomu pomáhají body níže.
Rozměr	Diagram by měl být čitelný celý, pokud bude zobrazen v zamýšleném výstupním formátu a pozorovaný z obvyklé vzdálenosti bez potřeby „zoomování“.
Počet elementů	Velmi záleží na druhu diagramu a zamýšleném způsobu prezentace. Doporučeno je pro diagramy jiného než přehledového účelu použít do cca 15 elementů. Přehledové diagramy, u nichž je předpoklad tisku na větší formát, mohou mít počet elementů i násobně větší a stále zachovat pochopitelnost.
Uspořádání elementů	Elementy na diagramu rozmísťujeme tak, aby byla patrná myšlenka, příběh a vztahy elementů. Vhodnou topologii (umístění) lze čerpat z referenčních modelů, pokud jsou OHA publikovány.
Zvýraznění rozměrem	Centrální prvek diagramu může být rozměrově zvětšen, aby upoutal oko, odtud začne čtenář diagram číst.
Zobrazení detailů elementu	Využijte možnosti nástroje zobrazit či naopak nezobrazit vybrané atributy elementu, podle zprávy, kterou chceme předat čtenáři a aby se detaily zbytečně neopakovaly na více diagramech. Např. u elementů navázaných na ústřední element diagramu není třeba uvádět všechny detaily, zajímá nás pouze výčet těchto elementů a jejich detailní zobrazení najdeme na diagramu jim dedikovaným. Eliminuje se tím také riziko, že se pracně odladěný diagram rozbije při přidání nových atributů k elementu.
Zobrazení vazeb	Zvažte, zda na diagramu potřebujeme všechny vazby mezi zobrazenými elementy. Často nás zajímají vazby jen na centrální element, ale ne vazby mezi sekundárními elementy. Případně nás zajímají vazby jen vybraných typů. V takovém případě je vhodné nezajímavé vazby na diagramu schovat. Nástroje obvykle podporují možnost jak samotného vypínání viditelnosti vazeb ad-hoc a tak i plošně vypnout zobrazení dané vazby na jiných diagramech (to se hodí v případě, kdy přidáváme novou vazbu a daná kombinace elementů se objevuje i na jiných diagramech).
Zarovnání elementů	Doporučujeme věnovat velkou pozornost optimalizaci rozmístění prvků na diagramu tak, aby byly elementy zarovnané (lze např. použít mřížku v modelovacím nástroji)
Zarovnání vazeb	Vazby by měly být ideálně zalamované tak, aby se co neméně křížily a aby neprocházely přes elementy. U přehledových diagramů lze využít zarovnání vazeb do „sběrnice“ kdy vazby vycházejí z jednoho bodu elementu, vedou přes sebe a odbočují k cílovým elementům.

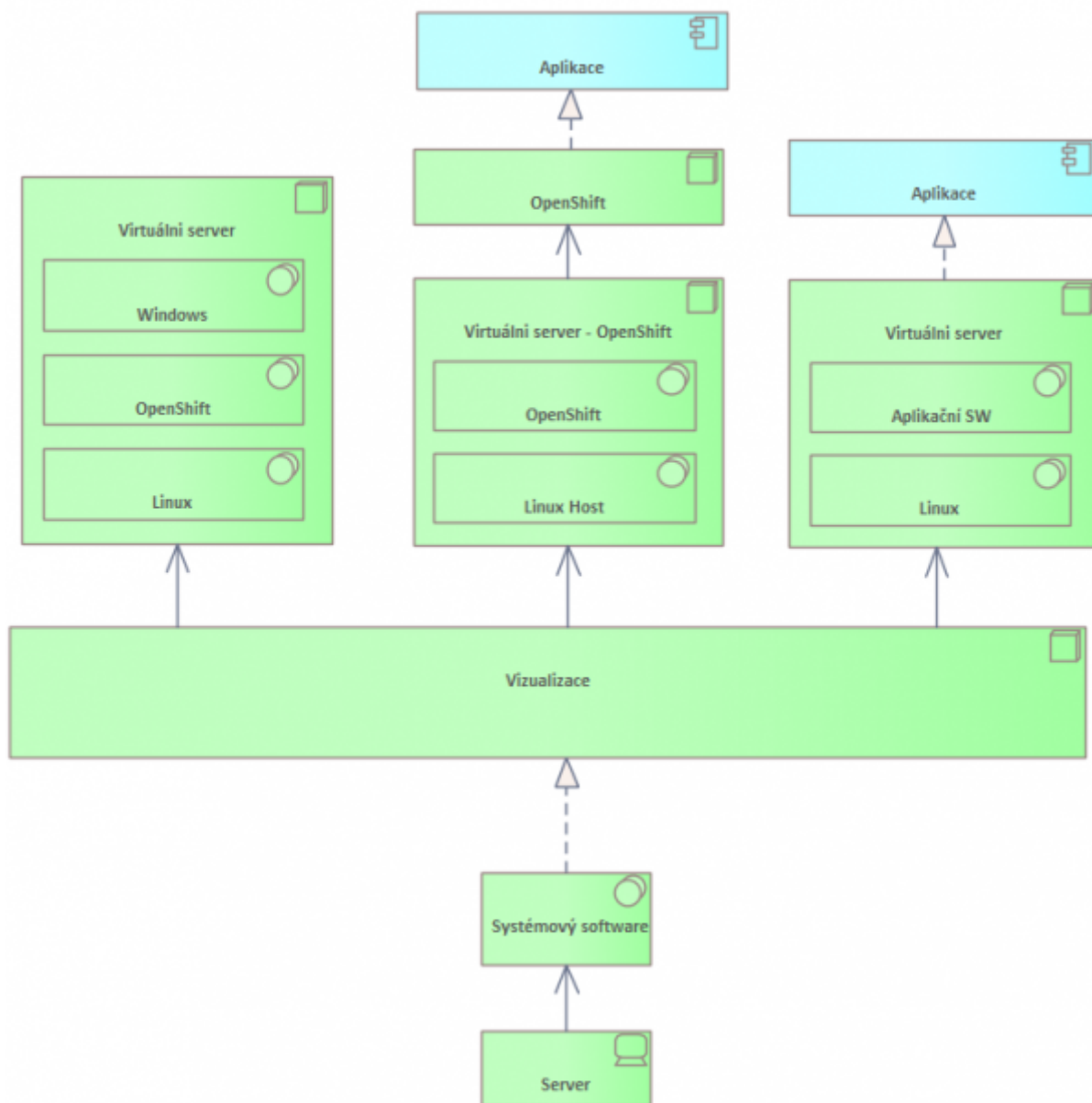


Příklad čitelného diagramu

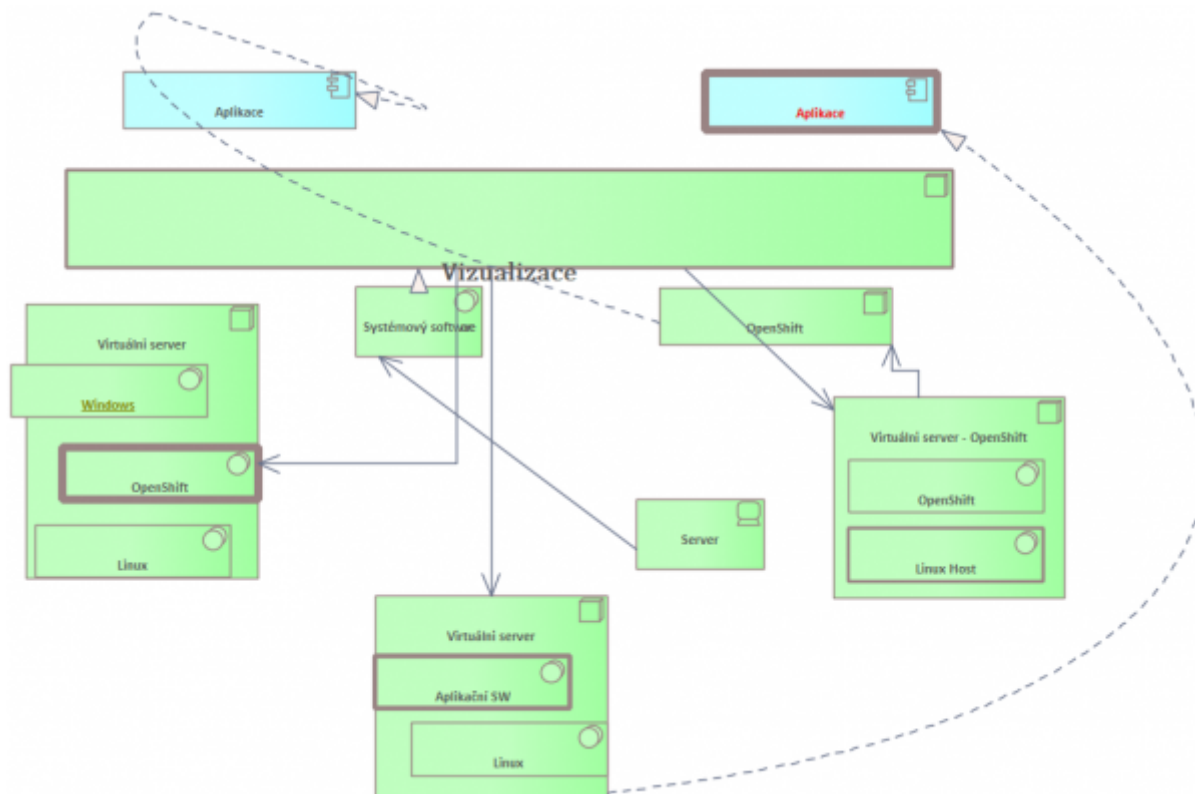


Příklad nečitelného diagramu (diagram obsahuje chaoticky rozmístěné elementy, kontrast písma a pozadí je

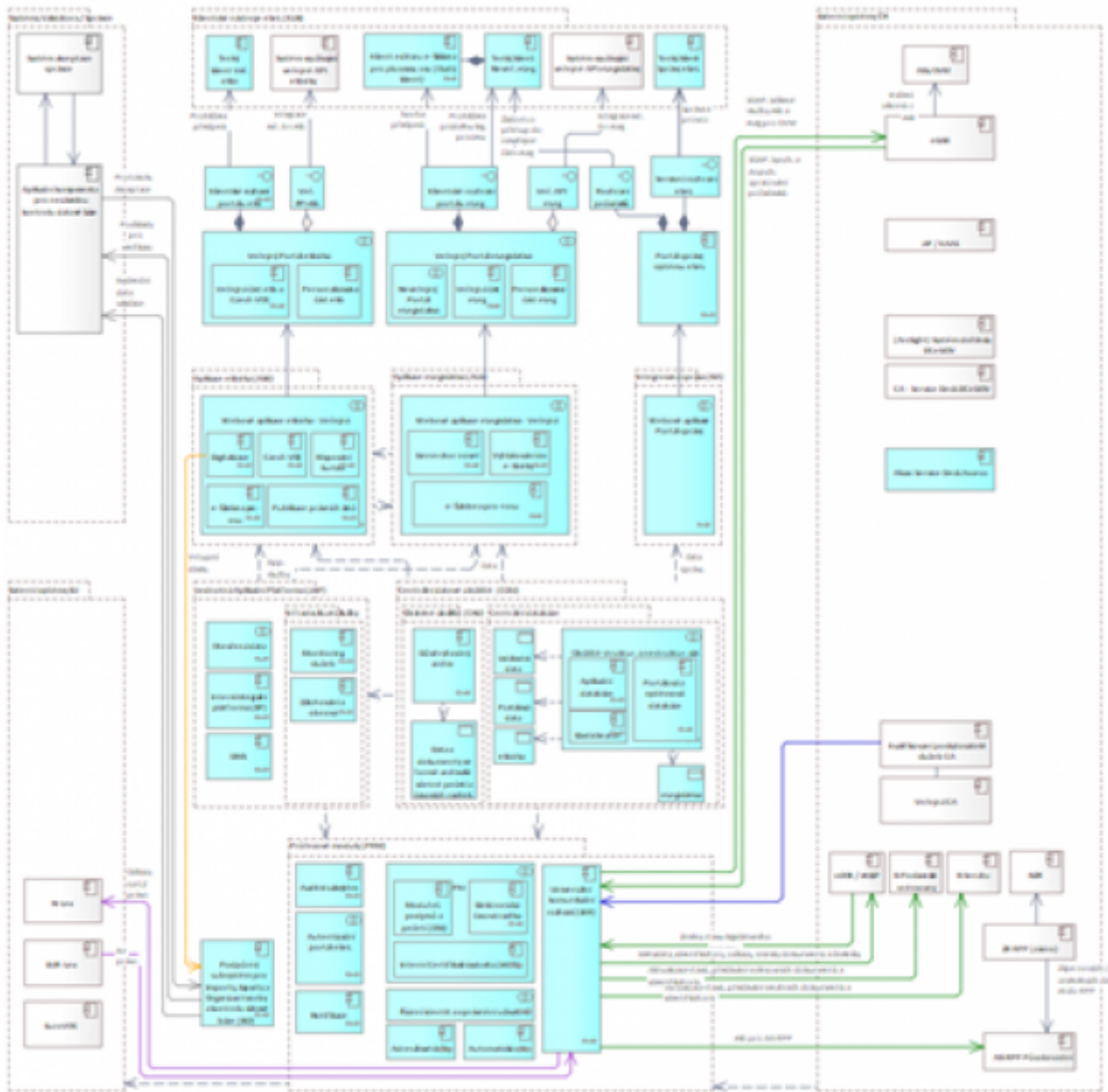
nevhodný a nedodržuje standard, vazby jsou nepřehledné a velikost písma je pro čtenáře příliš malá)



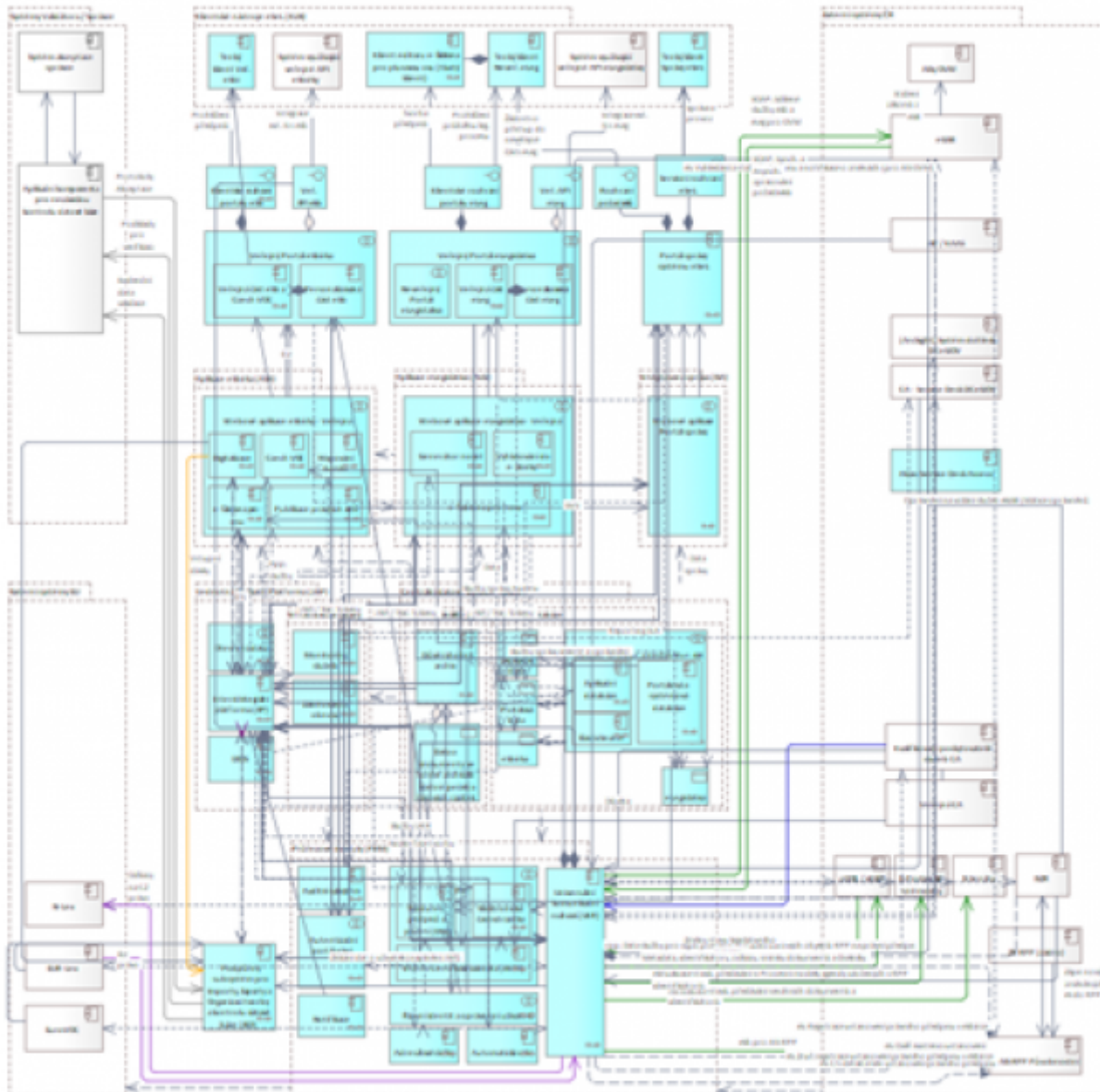
Příklad uspořádaného diagramu



Příklad neuspořádaného diagramu (elementy nejsou zarovnané, vazby se překrývají a nejsou přehledné, velikost rámečků není jednotná a font, velikost a barva písma je zvolená bezmyšlenkovitě – není obsažena legenda)



Příklad diagramu s uklizenými a vyfiltrovanými vazbami



Příklad diagramu se všemi vazbami, navíc bez zarovnání



Příklad centrického elementu v diagramu



Příklad přehledového diagramu

Matice

Matice jsou specifickým vizualizačním prvkem umožňují přehledně ukázat mapování mezi dvěma sadami elementů. Jsou užitečné pro kontrolu a ladění propojenosti, kdy jsou sady elementů ve vztahu m:n. V takovém případě se obtížně tvoří, a hlavně udržuje, podobný diagram.

Matice slouží pro i mapování vazeb mezi objekty napříč vrstvami architektury, což umožňuje čtenáři pochopit provázanost objektů a jejich využívání.

NAR tvorbu a použití matic dosud blíže neupravuje, tato metodika uvádí možné využití matic tam, kde to je efektivní.

Na co myslet při tvorbě matic:

Počet elementů na osách	Při velkém počtu elementů může být při tisku na obvyklé formáty papíru mapa příliš jemná a nečitelná. Doporučujeme zvážit rozpad na více matic
Aktualizace matice	Při průběžné tvorbě obsahu se výčet elementů na osách matice mění. Proto je vhodné matici aktualizovat, aby ukázala vždy platný seznam elementů. Totéž se týká sledovaných vazeb.
Vytvoření vazby v matici vs. v diagramech	Pokud vytváříte vazby přímo přes matici, zkontrolujte, zda se nechtěně neobjeví na diagramech s danou kombinací elementů.

Pozn.: Některé nástroje pro správu modelů matice nepodporují (například Archi).



Ukázka matice přímo v nástroji Sparx Enterprise Architect

Katalogy

Katalogy jsou výčty elementů daného typu, zobrazené ve formě tabulky s možností řazení či filtrace a nastavení zobrazených atributů.

Katalogy jsou spolu s maticemi více interaktivní než diagramy a hodí se zejména pro práci přímo v nástroji. Obvykle se katalogem řeší potřeba výčtu elementů s konkrétním řazením a atributy, či pro gap analýzy pomocí filtrace určitých hodnot atributů.

V tištěné dokumentaci je obvykle katalog realizován pomocí šablony pro vygenerování dokumentu pro příslušnou část modelu.

NAR doporučuje používat pro inventarizaci a správu portfolií klíčových konceptů (procesů, rolí, aplikačních a technologických komponent a služeb apod.) katalogy v tabulkovém kalkulátoru nebo ve specializovaných nástrojích, s obsahem podle OHA publikovaných akceleratorů. Takto připravené tabulkové zdroje je možné hromadně naimportovat do nástroje, čímž se ušetří čas. Pro hromadné úpravy je možné katalog dočasně vyexportovat, upravit vně nástroje a opět naimportovat. **Vždy je ale nutné zajistit, že model je primární zdroj katalogů pro potřeby podnikové architektury.**

U katalogů jsou zásadní následující aspekty:

Definice typů elementů	Definice typů elementů v jednom katalogu - je vhodné členit katalogy dle typu elementu podle metodiky
Řazení elementů	Řazení elementů v katalogu – obvyklé je výchozí řazení abecední dle názvu, přičemž je možné při vizualizaci katalogu volit řazení dle libovolného zobrazeného atributu.
Vnitřní strukturování katalogu	Vnitřní strukturování katalogu – s ohledem na očekávané množství položek a použitý způsob vizualizace lze někdy doporučit katalog vnitřně členit, např. dle modulů. Obvyklejší je ale ponechat seznam lineární a veškerou filtraci a řazení ponechat na vizualizačním nástroji pro katalog.

Pozn.: Některé nástroje pro správu modelů katalogy nepodporují (například Archi).



Ukázka katalogu přímo v nástroji Sparx Enterprise Architect

Členění modelu

Tato metodika modelování platí pro enterprise architekturu celku, domén a segmentů OVS, v mírně omezené

míře, respektive aplikovaně, pro enterprise architekturu řešení příležitostí, identifikovaných v doménových nebo segmentových architekturách.

V architektonickém popisu řešení lze pomocí této metodiky odpovídat na existencionální otázky CO? a PROČ? bude obsahovat hledané řešení, nikoli na otázky JAK MÁ FUNGOVAT? Takové modely tzv. solution architektury nejsou upraveny NAR ani touto metodikou, zejména proto, že:

1. Mají jiný účel
2. Jazyk ArchiMate se pro ně málo hodí (spíše UML).

Segmentace

Z hlediska rozsahu modelovaných OVS a jejich korporací, či z hlediska jejich vnitřní struktury dělené na segmenty a schopnosti je tato metodika modelování univerzálně použitelná.

Sem ale patří otázka, kolik má mít jedno OVS modelů. Před každým modelování nějakého katalogu nebo diagramu je nutno rozhodnout do jakého z nich daná informace patří. OVS by mělo být schopno rozlišit a samostatně spravovat modely:

- **VLST** – Vlastní model architektury úřadu (OVS). Obsahuje prvky a pohledy specifické pro tento kontext. Úřad může obsahovat několik modelů tohoto typu pro své jednotlivé organizace.
- **SPOL** – Společný model úřadu a všech jím kontrolovaných (a metodicky vedených) organizací (resort, krajská korporace atp.). Zde jsou umístěny pohledy kombinující prvky z několika modelů typu VLST. Mohou zde být i další prvky, spadající jen do kontextu SPOL.
- **ROZS** – Rozšířený model s prvky od spolupracujících organizací, úřadů, kdekoli v systému veřejné správy (případně soukromé sféry). Zde by měly být pohledy ukazující kontext mimo rámec SPOL, typicky se sdílenými prvky jiných úřadů. DO doby, než bude k dispozici centrální správa sdílených prvků, budou zde uloženy jejich kopie, které budou použité pro vazby a pohledy v modelu daného úřadu.

Na obrázku níže je ilustrován princip strukturování modelů těchto typů.

Ilustrace segmentace modelu z pohledu jednoho úřadu

Struktura modelu

Jednotná struktura modelů významně pomáhá jejich pochopitelnosti a umožňuje zastupitelnost jak na straně čtenářů, tak na straně tvůrců. Dlouhodobě si DIA klade za cíl udržovat modely podnikových architektur úřadů centrálně, což vede také na potřebu jednotného strukturování modelů. Proto v této metodice popisujeme hlavní myšlenky v této oblasti.

Model by měl být strukturován zhruba taktak, jak je vidět na obrázku níže. Struktura počítá s možností hierarchického členění modelu úřadu (prvky „Část úřadu N“). Následuje dělení dle architektonických domén, kde je očekávána úplná nezávislost obsažených prvků (elementů, vazeb) mezi doménami. Na další úrovni jsou pak elementy sdružené do katalogů a diagramy a matice, na nichž jsou elementy a vazby zobrazené.

Samotné elementy jsou drženy buď ve vlastní struktuře, nebo jako součást katalogů – záleží na konkrétním nástroji. To samé platí o uložení vazeb.

Zatímco elementy jsou jasně přiřazené k architektonickým doménám, na pohledech mohou být elementy z více domén současně. Umístění pohledu se pak řídí převažujícím typem elementů, či tím, kam spíše logicky patří. V kapitole 4.64.6 je pak uvedeno zařazení pohledů do jednotlivých architektonických domén.

Nelze zapomenout na pohledy, které jsou striktně průřezové. Pro ně je vhodné vyčlenit uzel struktury „Přehledová hlediska“, který je sdružuje.

Princip strukturování modelu úřadu

Prvek	Význam
Kořenový uzel	Strukturní prvek, pod nímž je celý model. Nese doplňkové informace určující o jaký úřad se jedná.
Část úřadu	Pokud je úřad vnitřně strukturně členěn ne víceméně nezávislé části, doporučujeme to reprezentovat i v modelu. Zejména v situaci, kdy mají části různé architektury.
Architektonická doména	Modelovaná vrstva/doména architektury (viz kap. 4.1)
Katalogy elementů	Obsahují seznamy elementů daného typu
Pohledy	Obsahuje jednotlivé pohledy na model v příslušné arch. doméně. Typicky se jedná o diagramy a matice.
Přehledová hlediska	Uzel struktury, pod nímž jsou umístěné pohledy obsahující přehledy a průřezové pohledy, které obsahují elementy z více domén najednou.

Úrovně detailu

Aby byla podpořena myšlenka jednotné granularity informací přes celý model, jsou definované tři úrovně detailnosti modelovaného pohledu na realitu. Tyto úrovně jsou pojmenované L0, L1 a L2 a jsou popsány v tabulce níže.

	Název	Význam
L0	Přehledová	Ukazuje principy a podstatu. Dává celkový přehled, celkový pohled na systém v dané doméně/oblasti; hlavní prvky a konzumenti. Často jsou zde agregované (seskupující) prvky.
L1	Základní	Ukazuje všechny klíčové prvky pro ucelený pohled. Doplnuje se s přehledovou úrovní.
L2	Detailní	Přidává detaily vybrané části z vyšší úrovně. Na této úrovni doporučujeme modelovat jen výseky, které si to zaslouží, např. díky komplexitě.

Detailní informace o konceptu úrovně detailu v NAR najdete na

[https://archi.gov.cz/nar_dokument:proces_tvorby_architektur?s\[\]=l1%2A#rozhodnuti_o_hloubce_obsahu](https://archi.gov.cz/nar_dokument:proces_tvorby_architektur?s[]=l1%2A#rozhodnuti_o_hloubce_obsahu)



Přehledová úroveň L0



Základní úroveň L1 (červený rámeček ukazuje proces rozpracovaný na úrovni L2)




Detailní úroveň L2

Modelování časové dimenze

Vždy musí být možné identifikovat časové období, k němuž se obsah modelu vztahuje. Zejména je důležité být schopný rozlišit, zda modelujeme současný (AsIs) či budoucí (ToBe) stav. Způsob realizace tohoto požadavku je vysoce závislý na způsobu řízení změn v podniku. Tato metodika doporučuje následující přístup:

Model aktuální situace je základ	Mějte v potřebných okamžicích aktuální model stávající situace (AsIs) podnikové architektury. Lze řešit např. aktualizací modelu po releasu.
Vývojové varianty modelujte odděleně od hlavního modelu	Pokud teprve vzniká návrh cílového stavu, průběžně existuje více možných pracovních variant. Aby byla zajištěna integrita modelu, doporučujeme pracovat na modelu vzniklém kopíí hlavního modelu, či ve zcela pracovním modelu. Po schválení cílového stavu jej přeneste do primárního modelu a navažte korektně na daný Stav architektury (vizte dále).

Model aktuální situace je základ	Mějte v potřebných okamžicích aktuální model stávající situace (AsIs) podnikové architektury. Lze řešit např. aktualizací modelu po releasu.
Časový vývoj modelujte s pomocí vrstvy Implementace a migrace	Pro modelování plánovaných změn použijeme část Implementace a migrace (viz kap. 4.6.5). Časové ukotvení nese příslušný Stav architektury. Na něj jsou navázané elementy (např. pomocí vazeb. D). Diagramy ukazují detailně situaci v tomto Stavu architektury, příslušnost ke Stavu architektury je řešena pomocí názvu či místěním diagramu. Je také využito barevné konvence pro indikaci změn v portfoliu.
V případě potřeby doplňte interval platnosti k prvkům a udržujte jej aktuální	Primárně by mělo k určení časové platnosti elementu stačit přiřazení k vybraným Stavům architektury. Pokud je to ale potřeba (např. pro zdůraznění využití až v nějakém období), doplníme element, vazbu či diagram obdobím platnosti. Je důležité nezapomenout tento údaj udržovat aktuální.
Označujte model datem platnosti	Označujeme okamžik, k němuž je model platný (Platnost k). Může to být datum, ale praktičtější může být označení obdobím, či releasem. Označení datem platnosti pomůže udržovat časové řezy modelu.
Archivujte časové řezy	Model verzujeme a starší řezy archivujeme pro účely udržování časových řezů z minulosti.

 Diagram s vyznačením barevné konvence pomocí ohraničení: zelená = nový, modrá = změna, červená = zrušený

Metainformace

Pro jednoznačné zařazení dané části modelu pro čtenáře, ale hlavně pro potřeby automatizace zpracování obsahu modelu, vč. např. validace formální správnosti je potřeba jednotlivé prvky modelu doplnit o informace, které toto umožní.

Některé údaje popsané v kapitolách výše jsou elegantně zaznamenány do modelu právě pomocí metainformací.

Metainformace jsou v modelovacích nástrojích obvykle udržované jako specifické atributy, či tagy. Doporučujeme prozkoumat možnosti použitého modelovacího nástroje pro automatizaci a systematizaci jejich vyplňování.

Pro jakoukoli integrovatelnost modelů mezi sebou je nutné metainformace k prvkům modelu udržovat.

Základní metainformace jsou:

Metainformace	Význam	Obvykle přiřazená k
Datum platnosti	K jakému datu je platný obsah dané části modelu	Strukturní prvky na vyšší úrovni
Autor/odpovědná osoba	Jméno autora či osoby odpovědné za obsah dané části modelu	Strukturní prvky, elementy, vazby, pohledy
Datum poslední aktualizace	Kdy byl daný prvek naposledy upraven	Strukturní prvky, elementy, vazby, pohledy
Vrstva architektury	Architektonická doména (viz kap. 4.14.1)	Strukturní prvky
Úroveň detailu	Míra detailu modelovaného pohledu na realitu (viz. kap. 3.6.43.6.4)	Pohledy
Reference na příslušnou část metamodelu	Určení, jakou část předpisu pro modelování daný prvek realizuje	Elementy, vazby, pohledy

From:

<https://archi.gov.cz/> - **Architektura eGovernmentu ČR**

Permanent link:

https://archi.gov.cz/playgroud:metodika_ey:metodika-2?rev=1743081493

Last update: **2025/03/27 14:18**

