
Modelování informačních systémů

Standard architektury

MPSV



Verze dokumentu: 1.0
Datum: 19. 10. 2015



Informace o dokumentu

Název standardu:	Standard Architektury - Metodika pro naplnění modelu Architektury		
Architekt řešení:	Aleš Škvor	Verze dokumentu:	1.0
Oblast působnosti standardu:	Infrastruktura	Datum dokumentu:	19. 10. 2015
Schválil:	Jiří Károly	Platnost od:	<datum>
Podpis:		Účinnost od:	<datum>

Historie verzí

Číslo verze	Datum verze	Vypracoval	Popis	Jméno souboru
0.1	31. 07. 2015	Aleš Škvor	První verze dokumentu	STD_ARCH_Modelování_informačních_systémů_150731
1.0	29. 10. 2015	Aleš Škvor	Zpracovány připomínky ICT MPSV.	STD_ARCH_Modelování_informačních_systémů_151029

Obsah

1. Úvod	4
2. Modelovací jazyky a standardy	4
3. Softwarové vybavení pro tvorbu modelů	5
4. Kategorie modelů.....	5
5. Typy modelů.....	5
5.1 Model požadavků	6
5.2 Model předmětné činnosti	7
5.3 Model architektury systémů a aplikací	7
5.4 Model architektury infrastruktury.....	7
5.5 Model procesů.....	8
5.6 Model případů užití.....	8
5.7 Model tříd	8
5.8 Model komponent.....	8
5.9 Model uživatelského rozhraní	9
5.10 Model perzistence	9
5.11 Model interakcí.....	9
5.12 Model rozhraní	9
5.13 Model nasazení.....	10
6. Všeobecné zásady.....	10
6.1 Zásady pro psaný text.....	10
6.2 Používání zkratk.....	10
6.3 Zásady pro popis elementů a vazeb	10
6.4 Zásady pro tvorbu diagramů	11
6.5 Tvorba a aktualizace modelu	11
6.6 Přístupy a odpovědnosti.....	11
7. Přílohy.....	12
7.1 Specifikace požadovaných modelů pro softwarové dílo	12
7.2 Specifikace požadovaných modelů pro dodávku systémové infrastruktury	13
7.3 Specifikace požadovaných modelů pro dodávku hardwarové infrastruktury.....	14
7.4 Specifikace požadovaných modelů pro dodávku síťové infrastruktury.....	15

1. Úvod

S ohledem na rostoucí počet informačních systémů, jejich narůstající složitost a komplexnost jejich technologické infrastruktury shledává resort jako nezbytné udržovat související informace nikoliv pouze ve formě strukturované dokumentace ale primárně ve formě modelů jak informačního systému jako celku tak i modelů jeho dílčích částí (subsystémů).

Pro účely standardu se modelem rozumí informační struktura, která je obrazem modelované (vymezené) skutečnosti a zachycuje klíčové strukturální, dynamické a relační aspekty skutečnosti. Na model je možno nahlížet z mnoha různých hledisek reprezentujících potřeby a zájmy různých zájmových skupin přicházejících do kontaktu s modelovanou realitou. Takovéto náhledy jsou typicky reprezentovány graficky pomocí diagramů s relevantním významovým popisem prvků a vazeb zachycených diagramy.

Reprezentace modelů je v maximální možné míře založená na využití standardizovaných, převážně grafických, modelovacích jazyků s pevnou syntaxí a sémantikou. Takovýto přístup zajišťuje čitelnost vytvářených modelů různými subjekty podílejícími se na tvorbě a provozu informačních systémů resortu MPSV.

Tvorba (a údržba) modelů je odborná, kvalifikovaná činnost spojující jak znalost modelovacích jazyků, tak i schopnosti správné abstrakce a přístupu k modelované skutečnosti. V neposlední řadě je tvorba modelů neoddelitelná od věcné znalosti modelované skutečnosti. Modelování je tak činností náročnou a musí být zejména v úrovni implementace řešení prováděno samotným řešitelem.

Cílem standardu tak je:

- Vymezit kategorie a typy modelů, které je řešitel informačního systému (či jeho částí) povinen zpracovávat v souvislosti s charakterem jeho dodávaného řešení.
- Vymezit technologické prostředky a nástroje, které jsou mandatorně využívány resortem pro tvorbu a údržbu modelů.
- Vymezit nezbytné role a odpovědnosti, které vyvstávají s přípravou modelů v prostředí mnoha působících řešitelů za účelem dosažení vzájemné konzistence a úplnosti modelů.

Tento standard je závazný pro všechny řešitelské projekty v oblasti informačních a komunikačních technologií resortu a projekty nelze finálně akceptovat bez dodání požadovaných modelů ze strany řešitele a to v kvalitě požadované standardem.

Standard nedefinuje, jaké konkrétní typy modelů jsou předmětem dodávky projektu. Toto vymezení je vždy specifické pro daný projekt řešení a je předmětem jeho zadání.

2. Modelovací jazyky a standardy

Pro potřeby tvorby dále uvedených typů modelů je využíváno několik typů modelovacích jazyků (notací) s ohledem na povahu typu modelu a jeho určení.

Jedná se o následující modelovací jazyky:

- **ArchiMate** – modelovací jazyk určený pro modelování podnikové architektury¹ v předmětné rovině, rovině informačních systémů a technologické rovině. Jazyk vznikl jako doplněk k architektonickému rámci TOGAF9². Jazyk je specifikován sdružením The Open Group a jeho specifikace je volně dostupná na adrese <http://www.opengroup.org/subjectareas/enterprise/archimate>.
- **Unified Modeling Language** verze 2 (UML2) – modelovací jazyk primárně určený pro modelování softwarových systémů a způsobu jejich nasazení do technologické infrastruktury. Jazyk je specifikován sdružením Object Management Group (OMG) a jeho specifikace je volně dostupná na adrese <http://www.uml.org/>.
- **Business Model and Notation** verze 2 (BPMN2) – modelovací jazyk určený pro pokročilé modelování organizačních procesů. Jazyk je specifikován sdružením Object Management Group (OMG) a jeho specifikace je dostupná na adrese <http://www.bpmn.org/>.

¹ Enterprise Architecture.

² The Open Group Architecture Framework verze 9 (více viz <http://www.togaf.org>).

Pro určité typy modelů (například modely požadavků, modely databázových struktur) není ustálená běžně rozšířená konvence jazyka určená pro modely. Pro tyto modely budou využívány prostředky poskytované softwarovým vybavením požadovaným pro zpracování modelů (viz dále v dokumentu).

Pro potřeby realizace dále uvedených typů modelů předpokládá vydavatel standardu, MPSV, základní znalost výše uvedených standardů³ od řešitelů informačních systémů. Formální soulad způsobu modelování skutečností bude předmětem oponentních seminářů ze strany útvaru architektury MPSV v průběhu tvorby modelů řešení.

3. Softwarové vybavení pro tvorbu modelů

Dále uvedené typy modelů jsou vytvářeny v rámci modelovacího nástroje SparxSystems Enterprise Architect⁴ verze 12 a novější (dále též jen EA) se společným sdíleným úložištěm (repositář). Každé úložiště reprezentuje projekt, ve kterém je vytvářeno více modelů. Projekt má přiřazeného vlastníka, kterým může být „Sekce náměstka ministryně pro informační technologie“ MPSV či pověřená řešitelská třetí strana. Vlastník spravuje model, řídí přístup uživatelů k modelu a určuje jejich práva – to vše v souladu s tímto standardem.

Vlastní úložiště projektu je realizováno (jako samostatná databáze) v technologické infrastruktuře MPSV. Ta poskytuje základní služby zabezpečeného přístupu k modelu prostřednictvím sítě Internet, provoz a zřízení databáze projektu a zálohování na úrovni databáze projektu.

Pro přístup k modelu potřebuje řešitel (uživatel) aplikaci SparxSystems Enterprise Architect minimálně ve verzi Corporate Edition. Licence pro přístup pracovníků řešitele MPSV neposkytuje a jsou tak nákladem řešitele v rámci jeho dodávky. Pro potřeby přístupu k projektu modelů pouze pro nahlížení je možno použít volně šiřitelnou čtečku.

4. Kategorie modelů

V rámci modelování skutečností resortu MPSV za účelem tvorby a provozu informačních systémů jsou vytvářeny následující kategorie modelů

- Modely globální architektury – jedná se primárně o horizontální modely zachycující organizaci, její funkce a informační podporu v míře detailu určenému řídicím pracovníkům a podnikovým architektům. Modely globální architektury budou vytvářeny primárně s pomocí modelovacího jazyka ArchiMate.
- Modely řešení – jedná se o modely dílčích řešení, jejichž účelem je zachycovat detailní informace o informačních systémech a technologické infrastruktuře. Modely řešení budou primárně vytvářeny s pomocí modelovacích jazyků UML2 a BPMN2.
- Referenční modely – jedná se o modely poskytující koncepční, generickou informaci, na jejímž základě mohou či musí být realizovány určité aspekty či části modelů globální architektury či modelů řešení. Referenční modely mohou podle své povahy používat libovolné modelovací jazyky uvedené ve standardu.
- Metamodely – jedná se o modely specifikující dodatečnou normu či specializaci pro modelování skutečností. Metamodely mohou využívat prostředků pro rozšíření jazyka výše uvedených standardů či mohou být realizovány základními prostředky UML2 (typicky modely tříd).

5. Typy modelů

Následující kapitola podává výčet a popis typů modelů, které mohou být vytvářeny řešitelem v závislosti na typu jím dodávaného řešení.

³ Vždy v aktuálních verzích (podporovaných zvoleným softwarovým vybavením pro tvorbu modelů). K 30. 7. 2015 se jedná o verze ArchiMate 2.1, UML 2.5 a BPMN 2.0.

⁴ Viz <http://www.sparxsystems.com>.

5.1 Model požadavků

Model požadavků je model, který ve strukturované podobě zachycuje požadavky na budoucí dílo (typicky informační systém). To musí být realizováno v souladu s požadavky. V rámci změnových řízení či řízení projektu může po vzájemné dohodě řešitele a zadavatele docházet k revizi požadavků, vždy však způsobem, který je plně v souladu se zákonem.

Každý z požadavků je tvrzení ideálně v rozsahu jednoho odstavce, které je:

1. srozumitelné,
2. správné,
3. úplné,
4. splnitelné,
5. nutné,
6. ověřitelné,
7. identifikovatelné
8. a lze stanovit jeho prioritu.

Katalog požadavků jako celek musí dále zaručovat, že v něm evidované požadavky jsou:

1. unikátní,
2. vzájemně konzistentní,
3. trasovatelné v kontextu zpřesňujících požadavků
4. a evidence požadavků je úplná.

Katalog požadavků bude v rámci aplikace SparxSystems Enterprise Architect realizován s využitím meta-objektů tvořených nástroji (tools) ve skupině „Requirements“.

Požadavky musí být podle své povahy seskupovány do skupin (formou balíků požadavků) uvedených v následující tabulce (přičemž ne všechny skupiny se musí vyskytovat v každém katalogu).

Skupina	Popis
Funkční požadavky	Požadavky na funkcionality poskytované budoucím dílem svému okolí (uživatelům, systémům, apod.)
Požadavky na architekturu	Požadavky na dekompozici řešení, vazeb a charakteristik jednotlivých komponent.
Požadavky na uživatelské rozhraní	Požadavky na strukturu, chování a způsob implementace uživatelského rozhraní díla.
Požadavky na bezpečnost	Požadavky na bezpečnostní aspekty díla, včetně ochrany zpracovávaných údajů.
Integrační požadavky	Požadavky na integraci díla s okolním prostředím.
Výkonnostní požadavky	Požadavky v souvislosti s výkonem a dostupností funkcionalit díla.
Provozní požadavky	Požadavky související s provozem a podporou díla.
Požadavky na testování	Požadavky související se způsobem testování a předáváním / přebíráním díla.
Požadavky na dokumentaci	Požadavky na dokumentaci předávanou a udržovanou společně s dílem.
Požadavky na migraci	Požadavky na migraci, přechod či nahrazení stávajícího díla dílem novým.
Požadavky na školení	Požadavky na školení poskytované v souvislosti s dodáním díla.

Katalog požadavků může v obdobném členění jako výše uvedené skupiny požadavků zahrnovat též součinnostní požadavky tj. požadavky, které musí splnit (a garantuje) zadavatel, aby řešitel mohl naplnit na něho kladené požadavky.

5.2 Model předmětné činnosti

Model předmětné činnosti je model vytvářený za účelem popisu předmětné architektury („business architecture“) organizace jako kontextu pro modelování architektury systémů a aplikací.

Model je vytvářen v souladu se standardem ArchiMate 2.1 částí „3 Business Layer“. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupině „ArchiMate2, Business“.

Na nejvyšší úrovni bude model strukturován pomocí balíků odpovídajícím funkčním či organizačním oblastem, ke které se model vztahuje. Struktura může být víceúrovňová.

Na model předmětné činnosti zpravidla navazuje model architektury systémů a aplikací. Jsou-li v rámci projektu vytvářeny oba modely, jsou vazby mezi prvky modelů provázány v souladu se standardem ArchiMate 2.1 částí „6 Cross-Layer Dependencies“.

Model předmětné činnosti v rozsahu předmětné činnosti celé organizace bude vytvářen a udržován útvarem architektury na základě vstupů připravovaných řešiteli dílčích problematik.

5.3 Model architektury systémů a aplikací

Model architektury systémů a aplikací je model vytvářený za účelem popisu existujících a zamýšlených informačních systémů organizace a dílčích aplikací, jejich vazeb, chování a typů zpracovávaných dat („information systems architecture“).

Model je vytvářen v souladu se standardem ArchiMate 2.1 částí „4 Application Layer“. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupině „ArchiMate2, Application“.

Na nejvyšší úrovni bude model strukturován pomocí balíků odpovídajícím logickému členění systémů (například agendové systémy, ekonomické systémy, apod.). Struktura může být víceúrovňová.

Na model architektury systémů a aplikací zpravidla navazuje model architektury infrastruktury. Jsou-li v rámci projektu vytvářeny oba modely, jsou vazby mezi prvky modelů provázány v souladu se standardem ArchiMate 2.1 částí „6 Cross-Layer Dependencies“. O návaznosti s modelem předmětné činnosti pojednává kapitola „Model předmětné činnosti“.

Model architektury systémů a aplikací v rámci celé organizace bude vytvářen a udržován útvarem architektury na základě vstupů připravovaných řešiteli dílčích problematik.

5.4 Model architektury infrastruktury

Model architektury infrastruktury je model vytvářený za účelem popisu existující a zamýšlené technologické infrastruktury pro provoz informačních systémů organizace („technology layer“). Technologická infrastruktura primárně zahrnuje výpočetní prostředky (hardware, disková úložiště, specializovaná zařízení jako HSM (Hardware Security Module) apod.), komunikační prostředky (zařízení a sítě hlasové komunikace, datové sítě a zařízení, specializovaná zařízení jako je „firewall“, „load balancer“ či „SSL terminátor“, apod.) a systémové programové vybavení (operační systémy, běhová prostředí jako J2EE či Microsoft.NET, databázová prostředí, integrační platformy, dohledové a monitorovací systémy, infrastrukturní služby jako je DNS či DHCP apod.).

Model architektury infrastruktury popisuje jak rovinu fyzických prvků infrastruktury, tak i rovinu prvků logických a virtuálních.

Model je vytvářen v souladu se standardem ArchiMate 2.1 částí „5 Technology Layer“. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupině „ArchiMate2, Technology“.

Na nejvyšší úrovni bude model strukturován pomocí balíků odpovídající vhodné logické kategorizace infrastruktury například podle dislokace prvků, podle poskytovaných služeb či podle pojmenovaných celků. Struktura může být víceúrovňová.

O návaznosti s modelem architektury systémů a aplikací pojednává kapitola „Model architektury systémů a aplikací“.

Model architektury infrastruktury v rámci celé organizace bude vytvářen a udržován útvarem architektury na základě vstupů připravovaných řešiteli dílčích problematik.

5.5 Model procesů

Model procesů je model vytvářený pro potřeby zpracování přesného, detailního, strukturovaného a více či méně formálního popisu jednoho nebo více procesů (řízené činnosti).

Model je vytvářen v souladu se standardem BPMN v2.0. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupině „BPMN 2.0“.

Model procesů může navazovat na model předmětné činnosti ve smyslu přesnějšího, detailnějšího rozpracování předmětných procesů a funkcí evidovaných a modelovaných prostředky ArchiMate 2.1 v rámci modelu předmětné činnosti.

Model procesů je typicky součástí modelů řešení.

5.6 Model případů užití

Model případů užití („use case model“) je model vytvářený pro potřeby zachycení interakcí modelovaného systému, komponenty či třídy se svým okolím za účelem poskytování svých funkcionalit (služeb), typicky pro potřeby formálního vymezení požadavků na funkcionalitu.

Model je vytvářen v souladu se standardem OMG Unified Modeling Language 2.5, primárně částí „18 Use Cases“. V rámci nástroje SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupině „Use Case“.

V případě, že model je vytvářen za účelem popisu informačního systému (aplikace, aplikační komponenty) jsou scénáře případů užití popisovány v aplikaci EA pomocí strukturovaného popisu – „Structured Specification“. V ostatních případech je dostatečné popsat případ užití v rámci popisu („notes“) modelovaného prvku strukturovaným textem.

Model případů užití může navazovat na modely architektury systémů a aplikací a architektury infrastruktury ve smyslu detailnějšího rozpracování popisu aplikačních a infrastrukturních funkcí („application and infrastructure functions“).

Model případů užití je typicky součástí modelů řešení softwarových informačních systémů.

5.7 Model tříd

Model tříd („class model“) je model vytvářený za účelem popisu konceptuálních pojmů (reálného světa v případě analytických modelů, či informačních struktur v případě modelů návrhů informačních systémů), jejich vlastností, chování a vzájemných vazeb.

Model je vytvářen v souladu se standardem OMG Unified Modeling Language 2.5, primárně částmi „7 Common Structure“, „9 Classification Classifiers“ a „11 Structured Classifiers“. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupinách „Class“ a „Object“.

Pro potřeby modelování životního cyklu instancí (objektů) tříd může být využíváno modelů stavových automatů v souladu s částí „14 State Machines“ specifikace UML.

Model tříd může navazovat na modely předmětné činnosti a modely architektury systémů ve smyslu detailnějšího rozpracování popisu předmětných entit („business objects“) a datových struktur („data objects“).

Model případů užití je typicky součástí modelů řešení softwarových informačních systémů.

5.8 Model komponent

Model komponent („component model“) je model vytvářený za účelem popisu dílčích (softwarových, systémových) komponent, jejich vzájemných vazeb a rozhraní.

Model je vytvářen v souladu se standardem OMG Unified Modeling Language 2.5, primárně částmi „7 Common Structure“, „9 Classification Classifiers“ a „11 Structured Classifiers“. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupinách „Class“ a „Object“.

Model tříd může navazovat na modely předmětné činnosti a modely architektury systémů ve smyslu detailnějšího rozpracování popisu předmětných entit („business objects“) a datových struktur („data objects“).

Model případů užití je typicky součástí modelů řešení softwarových informačních systémů.

5.9 Model uživatelského rozhraní

Model uživatelského rozhraní slouží k vytvoření popisu struktury a chování grafického uživatelského rozhraní („graphical user interface (GUI)“) vytvářených aplikací a systému.

Model uživatelského rozhraní je vytvářen s pomocí „proprietárních“ nástrojů aplikace SparxSystems Enterprise Architect:

1. Pro modelování grafického uživatelského rozhraní nativních aplikací operačních systémů s rozhraním typ „oken“ je využíváno preferenčně nástrojů „Wireframing, Dialog“, či alternativně nástrojů „User Interface“ či „Win32 User Interface“.
2. Pro modelování grafického uživatelského rozhraní aplikací s rozhraním webových (WWW) stránek nástrojů „Wireframing, Webpage“.
3. Pro modelování grafického uživatelského rozhraní aplikací určených pro mobilní zařízení (chytře telefony, tablety) je využíváno nástrojů „Wireframing, Apple“ a / nebo „Wireframing, Android“.

Model případů užití je typicky součástí modelů řešení zákaznických softwarových informačních systémů.

5.10 Model perzistence

Model perzistence slouží k vytvoření detailního popisu struktur databází informačních systémů.

Model perzistence může reprezentovat trvalé (perzistentní) uložení instancí (objektů) tříd popsaných modelem tříd.

Model perzistence je vytvářen s pomocí „proprietárních“ nástrojů aplikace SparxSystems Enterprise Architect ve skupině „Data Modeling“. Pokud v rámci aplikace EA neexistuje předefinovaný typ databáze (popis jejich datových typů) jsou tyto vytvořeny v nastavení projektu (Project ► Settings ► Database Datatypes).

Model perzistence je typicky součástí modelů řešení softwarových informačních systémů.

5.11 Model interakcí

Model interakcí detailně popisuje způsoby interakce, struktury komunikace, výměny informací mezi třídami, systémovými a aplikačními komponentami. Jedná se o formální popis vhodný k modelování složitých interakcí, komunikačních prostředí. Model často doplňuje modely rozhraní.

Model je vytvářen v souladu se standardem OMG Unified Modeling Language 2.5, primárně částí „17 Interactions“. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupinách „Communication“, „Interaction“ a „Timing“.

Model perzistence je typicky součástí modelu softwarových děl implementující složité protokoly výměny informací mezi interními komponentami či s ostatními (například externími) systémy.

5.12 Model rozhraní

Model rozhraní slouží k detailnímu popisu struktury a popisu rozhraní poskytovaných aplikačními komponentami.

Na abstraktní úrovni je model vytvářen v souladu se standardem OMG Unified Modeling Language 2.5, primárně částmi „10.4 Interfaces“ a „11.6 Components“. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupině „Component“.

Pro potřeby detailního popisu rozhraní na bázi webových služeb (Web Services) může být využíváno v rámci aplikace EA proprietárních nástrojů „WSDL“ a „XML Schema“.

Model rozhraní je typicky součástí softwarových děl vyžadujících interakci (komunikaci) s externími systémy (z pohledu řešeného projektu). Popis interní komunikace v rámci dodávaného může být řešen pouze v kontextu modelu komponent.

5.13 Model nasazení

Model nasazení („deployment model“) je model vytvářený za účelem popisu nasazení aplikačního (softwarového) vybavení na technologické výpočetní prostředky a popisu vazeb a propojení těchto prostředků.

Model je vytvářen v souladu se standardem OMG Unified Modeling Language 2.5, primárně částí „19 Deployments“. V rámci aplikace SparxSystems Enterprise Architect bude využíváno nástrojů (tools) ve skupině „Deployment“.

Model nasazení může navazovat na model architektury infrastruktury ve smyslu detailnějšího rozpracování popisu způsobu nasazení a komunikace konkrétního programového vybavení (informačního systému, aplikací).

Model nasazení je typicky součástí modelů řešení softwarových informačních systémů za účelem specifikace požadavků či popisu na hardwarové infrastruktury pro zprovoznění komponent softwarového vybavení.

6. Všeobecné zásady

Všeobecné zásady popisují obecná pravidla aplikovatelná na všechny typy modelů uvedených v předcházejících kapitolách. Pravidla určují klíčovou formální kvalitu modelu a budou předmětem kontroly kvality zpracovávaných modelů ze strany projektového řízení kvality či útvaru architektury MPSV.

6.1 Zásady pro psaný text

Veškeré popisy v modelu budou používat český jazyk a dodržovat ustálené zvyklosti používání správných slovních druhů a konvencí požadovaných standardů – například název třídy je podstatné jméno začínající velkým písmenem, název případu užití je podstatné jméno slovesné začínající velkým písmenem.

Ve speciálních případech (například při modelování rozhraní vůči systémům orgánů působících v rámci Evropské unie) může být požadován popis v jazyce anglickém. Takovýto požadavek však bude explicitně stanoven útvarem architektury MPSV.

V obecných a analytických modelech využívajících standardů ArchiMate, UML2 či BPMN2 budou víceslovné názvy zapisovány nezkráceně s využíváním mezer mezi slovy. V případě modelů návrhu (například model softwarových tříd, model perzistence, model rozhraní webové služby) mohou být mezery u víceslovných pojmenování nahrazeny či vypuštěny v souladu s běžnými konvencemi – například využití podtržítka místo mezery či používání názvu v „camel“ zápisu.

6.2 Používání zkratk

V rámci popisů uvedených v modelech projektu je možno využívat zkratk a ustálených výrazů v jiných než českém jazyce.

Takovéto zkratky a výrazy však musí být uvedeny ve slovníku modelu, vytvářeného pomocí nástroje „glossary“ aplikace SparxSystems Enterprise Architect s uvedením celého názvu v případě zkratky a vysvětlujícím popisem jak v případě zkratky, tak i v případě ustáleného výrazu.

6.3 Zásady pro popis elementů a vazeb

Pokud je v modelu zaznamenán element, jeho atribut či operace, podmínky, vazby mezi elementy atd. musí tyto ve svých vlastnostech v poli „notes“ obsahovat významový popis.

Výjimku tvoří případy užití, které budou popisovány formou strukturovaných scénářů.

Názvy elementů a dalších objektů stejného typu musí být nazývány jednoznačně v daném kontextu, zpravidla daném balíčkem či dalším elementem, do kterého náleží. V případě požadavků je v začátku názvu uveden prefix s identifikátorem požadavku oddělený pomlčkou (včetně okolních mezer). Prefix je tvořen kódem typu požadavku (zvoleným uživatele) a pořadovým číslem fixní délky, tj. včetně předcházejících nul. Například „FP001 – Evidence osob“.

6.4 Zásady pro tvorbu diagramů

Obdobně jako elementy i diagramy musí mít ve svých vlastnostech v části „notes“ uveden významový popis diagramu – co zobrazuje, pro jaký účel, komu je určen.

Diagramy musí reflektovat svoji čitelnost při jejich tisku na formát maximálně velikosti A4 (ideálně svisle orientovaný). Dále musí reflektovat lidské kognitivní schopnosti a zobrazovat rozumný (ideálně ne více než deset) počet zobrazených prvků (a jejich vazeb). Je-li povaha modelované skutečnosti natolik komplexní, že zahrnuje větší množství prvků a vazeb musí být model a diagramy strukturován tak, aby byla dodržena zásada reflektování kognitivních schopností čtenáře diagramu. Ve výše uvedeném duchu je vhodné v diagramu potlačit zobrazení z jeho povahy nepodstatných prvků – například diagram vytvořený za účelem zobrazení jedné zvolené třídy a jejích vazeb, by měl zobrazovat atributy a operace zobrazované třídy, zatímco atributy a operace zobrazených tříd, ke kterým existuje vazba, by zobrazovány být neměly.

Vazby zachycené v diagramu se v rámci možností nesmí křížit.

Diagramy mohou využívat možnosti obarvení prvků – obarvení však nemůže nést primární význam. Například třídu, na kterou se chce soustředit čtenářova pozornost je možno podbarvit, zatímco ostatní třídy ne. Označit například třídy obchodních entit podbarvením pro potřeby primárního rozlišení od servisních tříd je nepřijatelné, neboť takováto informace musí být zaznamenána v meta-datech třídy⁵, ne pouze v jejím zobrazení.

Barevnost je nezbytné využívat umírněně, aby nebyla na úkor čitelnosti a přehlednosti diagramu.

6.5 Tvorba a aktualizace modelu

Tento standard se vztahuje k modelům vytvářených řešiteli⁶ dílčích informačních systémů a technologické infrastruktury resortu MPSV.

Technologické prostředky pro provoz centrálního úložiště projektů modelů (repositáře) pro potřeby projektu realizovaného řešitelem jsou zajišťovány MPSV. Řešitel dostane k modelu přístupy pro správu (viz další kapitola). Od tohoto okamžiku přebírá řešitel odpovědnost za obsah modelu, jeho tvorbu a aktualizaci.

Řešitel je povinen provádět věcnou a formální aktualizaci modelu za účelem odstranění připomínek k jím vytvářenému modelu, které jsou výstupem oponentur modelu ze strany útvaru architektury či projektového řízení kvality.

6.6 Přístupy a odpovědnosti

Po založení projektu modelů útvaru architektury MPSV jsou řešiteli předány přístupové údaje pro správu (jedná se o jediný účet správce v modelu). Tím řešitel přebírá plnou zodpovědnost za obsah modelu a řízení přístupu k modelům.

Řešitel si musí změnit předané heslo pro správu – to uloží v zapečetěné obálce a předá odpovědnému pracovníkovi útvaru bezpečnosti ICT MPSV.

Řešitel založí (i průběžně) uživatelský účet pro přístup k modelu s právy pro čtení pro pracovníky MPSV, popřípadě třetích stran na základě písemného pověření útvaru architektury MPSV. Na základě obdobného pověření tyto účty pozastaví či odstraní. Případnou opětovnou obnovu hesla či opětovnou inicializaci k účtu může řešitel provést též pouze na základě pověření útvaru architektury MPSV.

Založení a správa účtů pracovníků řešitele a případných jeho subdodavatelů a nastavení oprávnění těchto uživatelů je plně v kompetenci řešitele. Ten však je povinen předat na vědomí informaci útvaru architektury o vzniklých účtech a případných změnách. Informace bude obsahovat jméno a příjmení uživatele, název organizace či společnosti, pro kterou pracuje, adresu elektronické pošty, telefonní kontakt a nastavení přístupových práv. Pokud je to možné název účtu uživatele by měl vyhovovat konvenci, že je psán malými písmeny bez diakritiky, první znak je první písmeno jména uživatele a následuje (bez mezery) příjmení uživatele.

⁵ Například pomocí „tag values“ či stereotypů jazyka UML.

⁶ I když i modely vytvářené v kompetenci útvaru architektury MPSV budou vytvářeny v souladu s tímto standardem.

Před zpřístupněním účtu musí budoucí uživatel podepsat dohodu o mlčenlivosti mezi jím a MPSV v souvislosti s ochranou a nakládáním s informacemi obsaženými v modelech k nimž bude mít uživatel přístup. Dohoda musí být před zpřístupněním doručena na MPSV. Znění dohody připraví MPSV a její šablony předá řešiteli či ji zveřejní na svých stránkách.

Řešitel zajistí, že informace o přístupových účtech a údajích budou uživatelům předávány odděleně:

1. Název projektu modelů (repositáře) a uživatelské jméno bude předáno společně s návodem pro přístup k modelu⁷ pomocí elektronické pošty.
2. Heslo bude samostatně zasláno na mobilní telefon uživatele pomocí SMS.

Heslo pro správu a přístupová hesla uživatelů přistupujících k projektu modelů musí vyhovovat bezpečnostním standardům ICT MPSV. Porušení těchto standardů (včetně trvalého používání inicializačních hesel, sdílení účtu více uživateli) je považováno za bezpečnostní incident a může být sankcionováno.

7. Přílohy

Následující podkapitoly ukazují příklady specifikace požadavků na modelování informačních systémů v souladu s tímto standardem a to pro různé typy řešitelských projektů.

Požadavky jsou reprezentovány tabulkovým výčtem požadovaných modelů, jejich typů uvedených v kapitole 5 tohoto standardu a vymezení rozsahu a účelu modelu.

7.1 Specifikace požadovaných modelů pro softwarové dílo

Model	Typ(y) modelu	Rozsah a účel
Katalog požadavků	Model požadavků	Katalog požadavků obsahuje detailní požadavky na řešení odsouhlasené v rámci analýzy řešení. Požadavky musí být v souladu se zadávací dokumentací či být modifikovány pouze v souladu k tomu určenými procesy (změnové řízení). Finální akceptace plnění bude prováděna vůči tomuto katalogu.
Model předmětné činnosti	Model předmětné činnosti	Jedná se o model předmětné činnosti ve vztahu k dodávanému řešení tj. jinými slovy model předmětné činnosti podporované dodávaným řešením. Model bude převzat a zahrnut do celkového modelu globální architektury MPSV udržovaného útvarem architektury. Primární verze modelu bude zpracována v analytické fázi projektu.
Model architektury řešení	Model architektury systémů a aplikací	Jedná se o model architektury řešeného systému, jeho funkcionalit a služeb. Model je provázán s modelem předmětné činnosti. Model bude převzat a zahrnut do celkového modelu globální architektury MPSV udržovaného útvarem architektury. Primární verze modelu bude zpracována v analytické fázi projektu.
Model procesů	Model procesů	Jedná se o detailní model předmětných procesů podporovaných řešených systémem. Model bude vytvářen pouze v nezbytných případech, kdy je detailní popis procesu nezbytný pro porozumění a realizaci řešení. Model je výstupem analytické fáze projektu.
Model případů užití	Model případů užití	Jedná se o model popisující strukturu a interakce zamýšlené funkcionality řešeného systému. Model je výstupem analytické fáze projektu a bude případně modifikován tak, aby byl v souladu s finálně dodaným řešením.

⁷ Generický návod předá řešiteli útvarem architektury MPSV.

Model	Typ(y) modelu	Rozsah a účel
Model entit	Model tříd	Model pojmenovává analytické entity (obrazy skutečností reálného světa, které je předmětem zpracování informací řešeným systémem), jejich struktura a vzájemné vazby. Model je výstupem analytické fáze projektu a bude případně modifikován tak, aby byl v souladu s finálně dodaným řešením.
Model komponent	Model komponent Model tříd	Model pojmenovává softwarové komponenty řešení, nezbytné systémové komponenty a externí komponenty. Pojmenovává rozhraní a vazby komponent. Model je doplněn o model softwarových tříd, pokud je řešení dodáváno formou zakázového vývoje s využitím jazyka podporujícího objektově orientované programování (OOP). Model je výstupem fáze návrhu projektu. Model musí být v souladu (zpřesňuje) s modelem architektury řešení.
Model uživatelského rozhraní	Model uživatelského rozhraní	Model popisuje grafické uživatelské rozhraní aplikace zpřístupněné koncovým uživatelům. Model je výstupem fáze návrhu projektu. Model musí být v souladu (zpřesňuje) s modelem architektury řešení a modelem komponent.
Model databází	Model perzistence	Model popisuje strukturu databází použitých v řešení či částí databází v řešeních vycházejících z produktově orientovaných aplikací. Model je výstupem fáze návrhu projektu. Model musí být v souladu (zpřesňuje) s modelem architektury řešení a modelem entit, popřípadě i s modelem komponent.
Model externích rozhraní	Model rozhraní Model interakcí	Model popisuje strukturu rozhraní poskytovaných externím (z pohledu řešeného systému) systémů, případně protokoly komunikace v souvislosti s používáním rozhraní. Model je výstupem fáze návrhu projektu. Model musí být v souladu (zpřesňuje) s modelem architektury řešení a modelem komponent.
Model nasazení	Model nasazení	Model popisuje řešitelem předpokládaný způsob nasazení jeho řešení do technologické infrastruktury MPSV. Model je výstupem fáze návrhu projektu a je finálně modifikován, aby reflektoval finální nasazení systému. Model musí být v souladu (zpřesňuje) s modelem architektury řešení a modelem komponent.
Model migrace	Model perzistence Model procesů	Model popisuje požadované struktury pro primární migraci a způsob provedení migrace. Model struktury je popisován ve struktuře tabulek modelu perzistence (ty můžou reprezentovat též struktury textových souborů). Způsob provedení migrace je reprezentován jako model procesů provedení migrace. Model může být zahrnut do celkového modelu migrace větších projektových celků. Model je výstupem fáze návrhu projektu.

7.2 Specifikace požadovaných modelů pro dodávku systémové infrastruktury

Model	Typ(y) modelu	Rozsah a účel
-------	---------------	---------------

Model	Typ(y) modelu	Rozsah a účel
Katalog požadavků	Model požadavků	Katalog požadavků obsahuje detailní požadavky na řešení odsouhlasené v rámci analýzy řešení. Požadavky musí být v souladu se zadávací dokumentací či být modifikovány pouze v souladu k tomu určenými procesy (změnové řízení). Finální akceptace plnění bude prováděna vůči tomuto katalogu.
Model systémové infrastruktury	Model architektury systémů a aplikací Model architektury infrastruktury	Model popisuje využívaný systémový software, jeho vzájemné vazby a nasazení do hardwarové infrastruktury. Model bude převzat a zahrnut do celkového modelu globální architektury MPSV udržovaného útvarem architektury. Model bude zpracován ve fázi návrhu realizace projektu.
Model komponent	Model komponent	Model pojmenovává systémové komponenty a externí komponenty. Pojmenovává rozhraní a vazby komponent. Model je výstupem fáze návrhu projektu. Model musí být v souladu (zpřesňuje) s modelem systémové architektury.
Model systémových rozhraní	Model rozhraní Model interakcí	Model popisuje strukturu rozhraní pro poskytování systémových služeb, případně protokoly komunikace v souvislosti s používáním rozhraní. Model je výstupem fáze návrhu projektu. Model musí být v souladu (zpřesňuje) s modelem systémové architektury.
Model nasazení	Model nasazení	Model popisuje předpokládaný způsob nasazení komponent softwarové infrastruktury do technologické infrastruktury MPSV. Model je výstupem fáze návrhu projektu a je finálně modifikován, aby reflektoval finální nasazení systému. Model musí být v souladu (zpřesňuje) s modelem systémové architektury.
Model migrace	Model perzistence Model procesů	Model popisuje požadované struktury pro primární migraci a způsob provedení migrace. Model struktury je popisován ve struktuře modelu perzistence. Způsob provedení migrace je reprezentován jako model procesů provedení migrace. Model může být zahrnut do celkového modelu migrace větších projektových celků. Model je výstupem fáze návrhu projektu.

7.3 Specifikace požadovaných modelů pro dodávku hardwarové infrastruktury

Model	Typ(y) modelu	Rozsah a účel
Katalog požadavků	Model požadavků	Katalog požadavků obsahuje detailní požadavky na řešení odsouhlasené v rámci analýzy řešení. Požadavky musí být v souladu se zadávací dokumentací či být modifikovány pouze v souladu k tomu určenými procesy (změnové řízení). Finální akceptace plnění bude prováděna vůči tomuto katalogu.
Model hardwarové infrastruktury	Model architektury infrastruktury	Model popisuje hardwarové vybavení, jeho vazby a je dislokaci do datových center. Model je udržován průběžně s ohledem na změny v infrastruktuře a může být zahrnut do celkového modelu globální architektury.
Model virtuální infrastruktury	Model architektury infrastruktury	Model popisuje vrstvu virtualizace výpočetních prostředků, způsob jejího nasazení na hardwarovou infrastrukturu, virtuální výpočetní prostředky a jejich vazby. Model je udržován průběžně s ohledem na změny v infrastruktuře a může být zahrnut do celkového modelu globální architektury.



Model	Typ(y) modelu	Rozsah a účel
Model technologických řešení	Model nasazení Model komponent	Model popisuje detailní strukturu klíčových technologických řešení jako je vysoce dostupná infrastruktura, struktura a způsob nasazení datových úložišť, systémy zálohování a archivace. Model je udržován průběžně s ohledem na změny v řešeních.

7.4 Specifikace požadovaných modelů pro dodávku síťové infrastruktury

Model	Typ(y) modelu	Rozsah a účel
Katalog požadavků	Model požadavků	Katalog požadavků obsahuje detailní požadavky na řešení odsouhlasené v rámci analýzy řešení. Požadavky musí být v souladu se zadávací dokumentací či být modifikovány pouze v souladu k tomu určenými procesy (změnové řízení). Finální akceptace plnění bude prováděna vůči tomuto katalogu.
Model fyzické síťové vrstvy	Model architektury infrastruktury	Model popisuje fyzické síťové prvky a topologii jejich fyzického propojení. Model je udržován průběžně s ohledem na změny v infrastruktuře a může být, či jeho části mohou být zahrnuty do globální architektury.
Model virtuálních sítí	Model architektury infrastruktury	Model popisuje strukturu virtuálních sítí (VLAN, VPN) a způsob jejich implementace v prostředí fyzické síťové infrastruktury. Model dále popisuje začlenění výpočetních prostředků do virtuálních sítí. Model je udržován průběžně s ohledem na změny v infrastruktuře a může být, či jeho části mohou být zahrnuty do globální architektury.
Model IP sítí	Model architektury infrastruktury	Model popisuje strukturu virtuálních sítí (VLAN, VPN) a způsob jejich implementace v prostředí fyzické síťové infrastruktury. Model je udržován průběžně s ohledem na změny v infrastruktuře a může být, či jeho části mohou být zahrnuty do globální architektury.
Model síťových služeb	Model architektury infrastruktury	Model popisuje strukturu a adresaci IP (Internet Protocol) sítí a způsob jejich implementace v prostředí virtuálních sítí a / nebo fyzické síťové infrastruktury. Model dále popisuje začlenění výpočetních prostředků do IP sítí. Model je udržován průběžně s ohledem na změny v infrastruktuře a může být, či jeho části mohou být zahrnuty do globální architektury.
Model technologických řešení	Model nasazení Model komponent	Model popisuje detailní strukturu klíčových technologických řešení jako je adresářová služba, jmenný systémy, řešení vysoké dostupnosti, firewally, apod. Model je udržován průběžně s ohledem na změny v řešeních.
Model síťového zabezpečení	Model nasazení Model komponent	Model popisuje strukturu bezpečnostních prvků sítě (firewall, IDS, HSM, apod.) a povolenou komunikaci. Model je udržován průběžně s ohledem na změny v infrastruktuře.