

Zadavatel disponuje uvedenými rozhraními provozovaného IS VERA Radnice, včetně jejich aktuálního kompletního popisu.

Tento dokument obsahuje principy komunikace IS Radnice VERA provozovaného v prostředí města Ústí nad Orlicí.

Použité zkratky:

SSL – Spisová služba

ISSD – Informační systém spravující dokumenty

NS – Národní standard

EÚD – elektronická úřední deska

JOS – Jednotná organizační struktura

Spisová služba IS VERA Radnice®

Rozhraní SSL VERA (NS 2) je určeno k napojení externích ISSD na Spisovou službu VERA Radnice.

Rozhraní je implementováno dle příloh a kapitoly 9 ROZHRANÍ K PROPOJENÍ INFORMAČNÍCH SYSTÉMŮ SPRAVUJÍCÍCH DOKUMENTY nového znění národního standardu pro elektronické systémy spisové služby, který nabyl účinnost dne 4. července 2017.

Zadavatel vítěznému uchazeči poskytne další dokumentaci k upřesnění, jakým způsobem je obecné rozhraní implementováno Spisovou službou VERA Radnice.

Webové služby:

Komunikace webových služeb je založena na protokolu SOAP.

Přílohou národního standardu pro spisové služby jsou xml schémata, která popisují datovou strukturu požadavků a přenášených dat. SSL při přijetí požadavku validuje přijaté xml vůči těmto schématům.

Zabezpečení:

Komunikace mezi systémy probíhá pomocí zabezpečeného protokolu HTTPS s autentizací serveru certifikátem. Autentičnost dat je zajištěna elektronickými podpisy přenášeného xml obsahu.

Jiné způsoby autentizace (např. basic autentizace) nejsou v tuto chvíli podporovány. V testovacím provozu je možné ověřování elektronických podpisů vypnout.

Rozhraní pro externí aplikace na podsystém Příjmy systému VERA Radnice®

Komunikace probíhá pomocí protokolu založeném na XML standardu.

Třída: VERA-PD

SEZNAM METOD ROZHRANÍ

- predpis-uloz
- predpis-uloz-rozpis
- predpis-oprav
- predpis-oprav-rozpis
- predpis-uhrazeno
- predpis-storno
- platce-info
- platce-sezn
- platce-uloz
- platce-oprav

- platce-rus
- plat-centrum-uloz
- plat-centrum-oprav
- plat-centrum-rus

PRINCIP KOMUNIKACE

Rozhraní Radnice VERA vystupuje v roli serveru, rozhraní externího systému vystupuje v roli klienta. Tedy rozhraní VERA poskytuje údaje pouze a jen na základě požadavku externího systému.

Metody požadavku

- klient externího systému osloví server Radnice VERA a předá mu požadavek (variabilní symbol, id předpisu atd.), VERA odpoví OK/Chyba+popis chyby. Požadavek startuje nový proces.

Metody odpovědi

- server Radnice VERA zpracuje požadavek a předá odpověď (status, informace o subjektu). Každý proces (rozuměno vyřízení nějakého požadavku až do úplné konce) je jednoznačně identifikován svým ID. Tedy všechny dotazy/odpovědi vztahující se k danému procesu budou obsahovat request_id/response_id se stejnou hodnotou. V rámci různých procesů bude toto ID unikátní.

AUTORIZACE PŘÍSTUPU

Autorizace přístupu ke službám VERA bude zajištěna v rámci protokolu https (s omezením na konkrétní IP adresu a port). K serveru VERA se budou klienti autorizovat přiděleným login jménem případně certifikátem.

Zadavatel poskytne kompletní dokumentaci jednotlivých metod vítěznému uchazeči po podpisu Smlouvy.

Příklad detailního popisu metody:

Metoda: predpis-uloz

Metoda umožňuje uložení předpisu případně plátce.

Vstupní data:

ID případu <secondary_key> (CH25) - sekundární klíč (číslo přestupku)

kód poplatku <code_fee> (N4) – povinné, pokud není variabilní symbol

ID období <type_period> (CH1) - povinné, pokud není variabilní symbol, R - roční, P - pololetní, Q - čtvrtletní, M - měsíční, D – jednorázové, možnost uživatelských ID období

identifikace registru subjektu <type_subject> (CH1) – R - obyvatel, I - organizace povinné, pokud není variabilní symbol

identifikace subjektu <id_subject> (N10) – vazba do registru povinné, pokud není variabilní symbol (vzhledem k tomu, že máme ID rezidentů i nerezidentů bude zde vždy ID subjektu)

částka <amount> (N15) – povinné

variabilní symbol <variable_symbol> (N10) – povinné, pokud plátce není definován viz výše

specifický symbol <specific_symbol> (N10)

měna <currency> (CH4) - povinné

okamžik případu <case_date> (CH10) - pokud datum není vyplněno, pak se doplní dnešní datum vystavení <entry_date> (CH10) - pokud datum není vyplněno, pak se doplní dnešní datum splatnosti <maturity_date> (CH10) – pokud datum není vyplněno, pak se doplní podle konfigurace termínů splatnosti u období poplatku
datum nabytí právní moci <law_date> (CH10)
pořadí cyklu období <order_period> (N2)
druh výstupu <type_output> (N1) – 1 – soupis předpisů, 2 – poukázka, 3 – faktura, 4 – inkaso z účtu plátce, 5 – příkaz k úhradě pro plátce, 6 – stvrzenka, 7 – SIPO, 8 – inkaso ze sporožirového účtu plátce
SIPO <sipo> (N10) - číslo SIPO plátce
kód rozpisu <code_breakdown> (N4)- pokud není vyplněno použije se automaticky první kód rozpisu

Výstupní data:

status <status> (N5) – 0 (o.k.), záporná hodnota - chyba

popis výjimky <error_msg> (CH80)

- 1 systémová chyba
- 2 chyba při zápisu do agendy Příjmy
- 10 částka nevyplněna
- 11 chybně vyplněné údaje k identifikaci plátce
- 12 kód rozúčtování neidentifikován

variabilní symbol <variable_symbol> (N10)

ID předpisu <id_rule> (N10)

Formát dotazu:

```
<?xml version="1.0" encoding="windows-1250" ?>
<request id="223">
  <authorization>
    <id_user>EXSYS:user</id_user>
  </authorization>
  <class name="vera-pd" version="1.0" revision="1">
    <method name="predpis-uloz">
      <secondary_key>3214</secondary_key>
      <code_fee>9211</code_fee>
      <type_period>D</type_period>
      <type_subject>R</type_subject>
      <id_subject>245789</id_subject>
      <variable_symbol/>
    </method>
  </class>
  <order_period>1</order_period>
  <type_output>1</type_output>
  <specific_symbol/>
  <amount>400</amount>
  <currency>CZK</currency>
  <entry_date>2006-01-10</entry_date>
  <maturity_date>2006-01-31</maturity_date>
  <sipo>3178478714</sipo>
  <code_breakdown>1</code_breakdown>
</request>
```

```
</class>
</request>
```

Formát odpovědi

```
<?xml version="1.0" encoding="windows-1250" ?>
<response id="223">
  <class name="vera-pd" version="1.0" revision="1">
    <method name="predpis-uloz">
      <status>0</status>
      <error_msg/>
      <variable_symbol>540796016341</variable_symbol>
      <id_rule>1010</id_rule>
    </method>
  </class>
</response>
```

Rozhraní pro externí aplikace na podsystém Komunální odpad systému VERA Radnice[®]

Komunikace probíhá pomocí protokolu založeném na XML standardu.

Třída: VERA-PJ

SEZNAM METOD ROZHRAŇÍ

- poplatnik-uloz-oprav
- poplatnik-uloz-oprav-stav
- platce-ko-info
- poplatnik-ko-info

PRINCIP KOMUNIKACE

Rozhraní Radnice VERA vystupuje v roli serveru, rozhraní externího systému vystupuje v roli klienta. Tedy rozhraní VERA poskytuje údaje pouze a jen na základě požadavku externího systému.

Metody požadavku

- klient externího systému osloví server Radnice VERA a předá mu požadavek (variabilní symbol, id předpisu atd.), VERA odpoví OK/Chyba+popis chyby. Požadavek startuje nový proces.

Metody odpovědi

- server Radnice VERA zpracuje požadavek a předá odpověď (status, informace o subjektu). Každý proces (rozuměno vyřízení nějakého požadavku až do úplné konce) je jednoznačně identifikován svým ID. Tedy všechny dotazy/odpovědi vztahující se k danému procesu budou obsahovat request_id/response_id se stejnou hodnotou. V rámci různých procesů bude toto ID unikátní.

AUTORIZACE PŘÍSTUPU

Autorizace přístupu ke službám VERA bude zajištěna v rámci protokolu https (s omezením na konkrétní IP adresu a port). K serveru VERA se budou klienti autorizovat přiděleným login jménem případně certifikátem.

Zadavatel poskytne kompletní dokumentaci jednotlivých metod vítěznému uchazeči po podpisu Smlouvy.

Příklad detailního popisu metody:

Metoda umožňuje uložení nebo opravu poplatníka komunálního odpadu. Vytváří žádost, která je zařazena do fronty a připravena k zpracování referentem.

Vstupní data:

opakovaně

Poplatník:

typ akce <action> (CH1) – Z - uložení (obnovení), O - změna (sazby, plátce), U - ukončení - povinné

typ požadavku <type_request> (CH30) – popis požadavku

identifikace registru subjektu <type_subject> (CH1) – R - obyvatel, I - organizace - povinné

identifikace subjektu <subject_id> (N1) – vazba do registru - povinné

datum změny <change_date> (CH10) - povinné

typ sazby <rate_type> (CH4) – dle číselníku sazeb KO

Rekreační objekt - chata:

typ akce <action> (CH1) – Z - uložení (obnovení), O - změna (sazby, plátce), U - ukončení - povinné

typ požadavku <type_request> (CH30) – popis požadavku

číslo chaty <number_cottage> (CH10) označení chaty

poznámka <comment> (CH80)

datum změny <change_date> (CH10) - povinné

typ sazby <rate_type> (CH4) – dle číselníku sazeb KO

Neopakovaně

Plátce:

požadavek zapsal <officer> (CH30) – referent, který zapsal požadavek

ID případu <secondary_key> (CH25) - sekundární klíč (ID požadavku CIC)

kód poplatku <code_fee> (N4) – povinné, pokud není variabilní symbol

ID období <type_period> (CH1) - povinné, pokud není variabilní symbol, , R - roční, P - pololetní, Q - čtvrtletní, M - měsíční, D – jednorázové, možnost uživatelských ID období

identifikace registru subjektu <type_subject> (CH1) – R - obyvatel, I - organizace povinné, pokud není variabilní symbol

identifikace subjektu <id_subject> (N10) – vazba do registru povinné, pokud není variabilní symbol (vzhledem k tomu, že máme ID rezidentů i nerezidentů bude zde vždy ID subjektu

variabilní symbol <variable_symbol> (N10) – povinné, pokud plátce není definován viz výše

SIPO <sipo> - číslo SIPO plátce

druh výstupu <type_output> (N1) – 1 – soupis předpisů, 2 – složenky, 3 – faktury, 4 – inkaso z účtu plátce, 5 – příkaz k úhradě pro plátce, 6 – stvrzenka, 7 – SIPO – výstup na disketu, 8 – inkaso ze sporožirového účtu plátce

kontaktní adresa plátce <address>

<city> (CH30)

<district> (CH30)

<state> (CH3)

<city_part> (CH40)

<house_number> (N5)
<number_type> (N1)
<street_name> (CH30)
<number_add> (N5)
<postcode> (CH10)

Výstupní data:

status <status> (N5) – 0 (o.k.), záporná hodnota - chyba

popis výjimky <error_msg> (CH80)

- 1 systémová chyba
- 11 chybně vyplněné údaje k identifikaci plátce
- 15 plátce nenalezen
- 16 žádost s tímto id byla již zadána
- 18 chybně vyplněn typ akce
- 19 chybně vyplněno ID poplatníka
- 20 chybně vyplněno datum změny
- 21 poplatníci nezadáni

Formát dotazu:

```
<?xml version="1.0" encoding="windows-1250" ?>
<request id="378">
  <authorization>
    <id_user>eSMO:pilar</id_user>
  </authorization>
  <class name="vera-pj" version="1.0" revision="1">
    <method name="poplatnik-uloz-oprav">

      <payer>
        <officer>Novák František</officer>
        <variable_symbol>9011000245</variable_symbol>
        <id_subject>245789</id_subject>
        <type_subject>R</type_subject>
        <type_period>D</type_period>
        <code_fee>9211</code_fee>
        <secondary_key>333333</secondary_key>
        <sipo>3178478714</sipo>
        <type_output>1</type_output>
        <address>
          <city>Nemcice</city>
          <district>Pardubice</district>
          <state>CR</state>
          <city_part>Nemci</city_part>
          <house_number>12</house_number>
          <number_type>?</number_type>
          <street_name>V lipach</street_name>
          <number_add>12</number_add>
```

```

        <postcode>54444</postcode>
    </address>
</payer>

<ratepayers>
    <ratepayer>
        <subject_id>1111</subject_id>
        <type_subject>R</type_subject>
        <action>Z</action>
        <type_request>Změna adresy</type_request>
        <rate_type>?</rate_type>
        <change_date>2006-01-31</change_date>
    </ratepayer>
    <ratepayer>
        <subject_id>1112</subject_id>
        <type_subject>R</type_subject>
        <action>M</action>
        <type_request>Změna adresy</type_request>
        <rate_type>?</rate_type>
        <change_date>2006-01-31</change_date>
    </ratepayer>
    <ratepayer>
        <subject_id>1121</subject_id>
        <action>D</action>
        <type_request>Změna adresy</type_request>
        <rate_type>1</rate_type>
        <change_date>2006-01-31</change_date>
    </ratepayer>
</ratepayers>

<cottages>
    <cottage>
        <number_cottage>1247811</number_cottage>
        <comment>Palučiny 18</comment>
        <action>Z</action>
        <type_request>Změna adresy</type_request>
        <rate_type>1</rate_type>
        <change_date>2006-01-31</change_date>
    </cottage>
</cottages>

</method>
</class>
</request>

```

Formát odpovědi

```
<?xml version="1.0" encoding="windows-1250" ?>
```

```

<response id="378">
  <class name="vera-pj" version="1.0" revision="1">
    <method name="poplatnik-uloz-oprav">
      <status>0</status>
      <error_msg/>
    </method>
  </class>
</response>

```

Rozhraní pro externí aplikace na podsystém Evidence psů systému VERA Radnice®

Komunikace probíhá pomocí protokolu založeném na XML standardu.

Třída: VERA-PS

SEZNAM METOD ROZHRANÍ

- pes-uloz-oprav
- pes-uloz-oprav-stav
- platce-ps-info
- ciselnik-ps
- pes-info

PRINCIP KOMUNIKACE

Rozhraní Radnice VERA vystupuje v roli serveru, rozhraní externího systému vystupuje v roli klienta. Tedy rozhraní VERA poskytuje údaje pouze a jen na základě požadavku externího systému.

Metody požadavku

- klient externího systému osloví server Radnice VERA a předá mu požadavek (variabilní symbol,id předpisu atd.), VERA odpoví OK/Chyba+popis chyby. Požadavek startuje nový proces.

Metody odpovědi

- server Radnice VERA zpracuje požadavek a předá odpověď (status, informace o subjektu). Každý proces (rozuměno vyřízení nějakého požadavku až do úplné konce) je jednoznačně identifikován svým ID. Tedy všechny dotazy/odpovědi vztahující se k danému procesu budou obsahovat request_id/responce_id se stejnou hodnotou. V rámci různých procesů bude toto ID unikátní.

AUTORIZACE PŘÍSTUPU

Autorizace přístupu ke službám VERA bude zajištěna v rámci protokolu https (s omezením na konkrétní IP adresu a port). K serveru VERA se budou klienti autorizovat přiděleným login jménem případně certifikátem.

Zadavatel poskytne kompletní dokumentaci jednotlivých metod vítěznému uchazeči po podpisu Smlouvy.

Příklad detailního popisu metody:

5.1 Metoda: pes-uloz-oprav

Metoda umožňuje zaevidování žádosti o přihlášení nebo změnu psa.

Opakovaně

Pes:

typ akce <action> (CH1) – Z - uložení , O - změna , R - odhlášení - povinné

Číslo známky <dog_tag> (N10) – povinné pro O,R pokud není tattoo a number_chip

Tetování <tattoo> (CH20) – povinné pro O,R pokud není dog_tag a number_chip

Číslo čipu <number_chip> (CH20) – povinné pro O,R pokud není tattoo a dog_tag

Jméno psa <dog_name> (CH40)

Kód barvy <color_code> (CH4)

Barva <color> (CH64)

Kód plemene <code_race> (CH4)

Plemeno <race> (CH64)

Pohlaví psa <dog_sex> (CH1) – povinné pro Z

Stáří psa <dog_age> (N5)

Popis psa <dog_description> (CH25)

Poznámka ke psu <dog_notice> (CH40)

Poznámka k identifikaci <identification_notice> (CH20)

Od kdy držen <date_keep> (CH10)

Kód účelu <code_function> (N5) – povinné pro Z

Kód sazby <code_rate> (N5) – povinné pro Z, pro O jen v případě vyplnění change_date

Vznik poplatkové povinnosti <duty_from> (CH10) – povinné pro Z

Zánik poplatkové povinnosti <duty_to> (CH10) – povinné pro R

Datum změny <change_date> (CH10) – povinné pro O v případě vyplnění code_rate

Očkování <vaccination_date> (CH10)

Sleva od <discount_from> (CH10)

Sleva do <discount_to> (CH10)

Datum přidělení čipu <chip_date> (CH10)

Příhlášení psa <registration_from> (CH10)

Odhlášení psa <registration_to> (CH10)

Datum uložení psa do útulku <asylum_from> (CH10)

Datum vyzvednutí psa z útulku <asylum_to> (CH10)

Evidenční číslo psa v útulku <number_dog_asylum> (CH20)

Číslo přiznání cislo_priznani <number_declaration> (N10)

Datum narození psa <date_birth> (CH10)

Datum tetování <date_tattoo> (CH10)

Neopakovaně

Plátce:

Požadavek zapsal <officer> (CH30) – referent, který zapsal požadavek

ID případu <secondary_key> (CH25) - sekundární klíč (ID požadavku CIC)

Kód poplatku <code_fee> (N4) – povinné, pokud není variabilní symbol

ID období <type_period> (CH1) - povinné, pokud není variabilní symbol, , R - roční, P - pololetní, Q - čtvrtletní, M - měsíční, D – jednorázové, možnost uživatelských ID období

Identifikace registru subjektu <type_subject> (CH1) – R - obyvatel, I - organizace povinné, pokud není variabilní symbol

Identifikace subjektu <id_subject> (N10) – vazba do registru povinné, pokud není variabilní symbol

Variabilní symbol <variable_symbol> (N10) – povinné, pokud plátce není definován viz výše
Číslo průkazu ZTP-P <card_ztpp> (CH10)
Email <email> (CH50)
Telefon <phone> (CH50)

Výstupní data:

status <status> (N5) – 0 (o.k.), záporná hodnota - chyba
popis výjimky <error_msg> (CH80)

- 1 systémová chyba
- 11 chybně vyplněné údaje k identifikaci plátce
- 15 plátce nenalezen
- 16 žádost s tímto id byla již zadána
- 18 chybně vyplněn typ akce
- 19 chybně vyplněné údaje psa
- 21 psi nezadáni

Formát dotazu:

```
<?xml version="1.0" encoding="windows-1250" ?>
<request id="378">
  <authorization>
    <id_user>EXSYS:user</id_user>
  </authorization>
  <class name="vera-ps" version="1.0" revision="1">
    <method name="pes-uloz-oprav">

      <payer>
        <officer>Novák František</officer>
        <variable_symbol>9011000245</variable_symbol>
        <id_subject>245789</id_subject>
        <type_subject>R</type_subject>
        <type_period>D</type_period>
        <code_fee>9211</code_fee>
        <secondary_key>333333</secondary_key>
        <card_ztpp>4571</card_ztpp>
        <email>hotline@vera.cz</email>
        <phone>603111234</phone>
      </payer>

      <dogs>
        <dog>
          <action>Z</action>
          <dog_tag>24781</dog_tag>
          <tattoo>57175</tattoo>
          <number_chip>24487</number_chip>
          <dog_name>Alík</dog_name>
        </dog>
      </dogs>
    </method>
  </class>
</request>
```

```

        <color_code>1</color_code>
        <color>černá</color>
        <code_race>2</code_race>
        <race>jezevčík</race>
        <dog_sex>P</dog_sex>
        <dog_age>5</dog_age>
        <dog_description>pěkný pes</dog_description>
        <dog_notice>kouše</dog_notice>
        <identification_notice>nosí známku</identification_notice>
        <date_keep>2006-01-31</date_keep>
        <code_function>1</code_function>
        <code_rate>3</code_rate>
        <duty_from>2006-01-01</duty_from>
        <duty_to>2008-12-31</duty_to>
        <change_date>2006-01-31</change_date>
        <vaccination_date>2006-08-01</vaccination_date>
        <discount_from>2007-08-01</discount_from>
        <discount_to>2007-08-31</discount_to>
        <chip_date>2006-07-20</chip_date>
        <registration_from>2006-01-31</registration_from>
        <registration_to>2008-01-31</registration_to>
        <asylum_from>2007-08-01</asylum_from>
        <asylum_to>2006-09-10</asylum_to>
        <number_dog_asylum>475728</number_dog_asylum>
        <number_declaration>2477</number_declaration>
        <date_birth>2006-08-01</date_birth>
        <date_tattoo>2006-08-01</date_tattoo>
    </dog>
    <dog>
        .....
    </dog>
</dogs>

</method>
</class>
</request>

```

Formát odpovědi

```

<?xml version="1.0" encoding="windows-1250" ?>
<response id="378">
    <class name="vera-ps" version="1.0" revision="1">
        <method name="pes-uloz-oprav">
            <status>0</status>
            <error_msg/>
        </method>
    </class>
</response>

```

Rozhraní pro externí aplikace na Registry systému VERA Radnice ®

Komunikace probíhá pomocí protokolu založeném na XML standardu.

Třída: VERA-VR

SEZNAM METOD ROZHRAŇÍ

- obyvatel-hledej
- obyvatel-uloz-oprav
- obyvatel-rus
- organizace-hledej
- organizace-uloz-oprav
- organizace-rus
- skupina-list
- skupina-data

PRINCIP KOMUNIKACE

Rozhraní Radnice VERA vystupuje v roli serveru, dotazující se aplikace v roli klienta, kterému je zodpovídan dotaz.

Klientský systém zašle požadavek na webové rozhraní, které požadavek po vyřešení přístupových práv zpracuje a odpoví požadovanými údaji.

Metody dotazu

- klient osloví server Radnice VERA a předá mu požadovaná data prostřednictvím XML.

Metody odpovědi

- server Radnice VERA zpracuje požadavek a předá odpověď (identifikaci, status a požadované informace) na základě požadavku klienta a vyhodnocení přístupových práv uvnitř systému Radnice VERA. Odpověď obsahuje záznamy vyhovující dotazovacím kritériím. Pokud dotaz vede na vrácení dat, které obsahují citlivé údaje o obyvateli, je takovýto dotaz zalogován.

AUTORIZACE PŘÍSTUPU

Autorizace přístupu ke službám VERA bude zajištěna v rámci protokolu https (s omezením na konkrétní IP adresu a port). K serveru VERA se budou klienti autorizovat přiděleným id uživatele, případně certifikátem.

Zadavatel poskytne kompletní dokumentaci jednotlivých metod vítěznému uchazeči po podpisu Smlouvy.

Příklad detailního popisu metody:

Metoda: obyvatel-hledej

Metoda vrací informace o nalezených obyvatelích v registrech VERA.

Identifikátor subjektu	<id_subject>
Typ subjektu	<subject_type> R – místní, N - externí
Jméno	<firstname>
Příjmení	<lastname>
Rodné jméno	<born_lastname />
Datum narození	<birthday> rrrr-mm-dd
Místo narození	<birth_place>
Okres narození	<birth_district>
Datum úmrtí	<death_date> rrrr-mm-dd
Místo úmrtí	<death_place>
Okres úmrtí	<death_district>
Rodné číslo	<birth_number>
Pohlaví	<sex> M – muž, Z - žena
Stav	<marital_status> S – svobodný, R – rozvedený, M – ženatý / vdaná, B – vdovec / vdova, P – partner, Z – zánik partnerství (rozluka), U – zánik partnerství (úmrtí)
Datum změny stavu	<marital_date> rrrr-mm-dd
Titul před jménem	<title1>
Titul za jménem	<title2>
Svéprávnost	<competence> 1 - svéprávný, 2 - nesvéprávný, 3 – částečně
Státní příslušnost	<nationality1>
Další státní příslušnost	<nationality2>
Identifikátor adresy	<id_residence>
Obec	<city>
Okres	<district>
Stát	<state>
Část obce	<city_part>
Domovní číslo	<house_number>
Typ domovního čísla	<number_type> 1 – popisné číslo, 2 – evidenční číslo, 3 – náhradní číslo
Jméno ulice	<street_name>
Řadové číslo	<number_add>
PSČ	<postcode>

Formát dotazu:

```
<?xml version="1.0" encoding="windows-1250" ?>
<request id="23">
  <authorization>
    <id_user>VERA:test</id_user>
  </authorization>
  <class name="vera-vr" version="1.0">
    <method name="obyvatel-hledej">
      <id_subject>12748</id_subject>
    </method>
  </class>
</request>
<subject_type>R</subject_type>
<lastname>Zdepa</lastname>
<firstname>Josef</firstname>
<born_lastname />
```

```

        <birthday>1912-12-12</birthday>
        <birth_place>Pardubice</birth_place>
        <birth_district>Pardubice</birth_district>
<death_date>2006-02-01</death_date>
<death_place>Nemcice</death_place>
<death_district>Pardubice</death_district>
        <birth_number>8812121234</birth_number>
        <sex>M</sex>
        <marital_status>M</marital_status>
        <marital_date>2006-02-22</marital_date>
<title1>Ing</title1>
        <title2>CSC</title2>
        <competence>1</competence>
        <nationality1>CZ</nationality1>
        <nationality2>PL</nationality2>
<id_residence>12748</id_residence>
<city>Nemcice</city>
<district>Pardubice</district>
<state>CR</state>
<city_part>Nemci</city_part>
<house_number>12</house_number>
<number_type></number_type>
<street_name>V lipach</street_name>
<number_add>12</number_add>
<postcode>54444</postcode>
</method>
</class>
</request>

```

Formát odpovědi

```

<?xml version="1.0" encoding="windows-1250" ?>
<response id="23">
  <class name="vera-vr" version="1.0">
    <method name="obyvatel-hledej">
      <status>0</status>
      <error_msg/>
      <subjects>
        <subject>
          <basic>
            <id_subject>12748</id_subject>
          </basic>
          <subject_type>R</subject_type>
          <lastname>Zdepa</lastname>
          <firstname>Josef</firstname>
          <middlename/>
          <born_lastname/>
          <birthday>1912-12-12</birthday>

```

```

        <birth_place>Pardubice</birth_place>
        <birth_district>Pardubice</birth_district>
    </basic>
    <death>
        <death_date>2006-02-01</death_date>
        <death_place>Nemcice</death_place>
        <death_district>Pardubice</death_district>
    </death>
        <others>
            <birth_number>8812121234</birth_number>
            <sex>M</sex>
            <marital_status>M</marital_status>
            <marital_date>2006-02-22</marital_date>
        <title1>Ing</title1>
            <title2>CSC</title2>
            <competence>Y</competence>
            <nationality1>CZ</nationality1>
            <nationality2>PL</nationality2>
        </others>
    <residence>
        <id_residence>12748</id_residence>
        <subject_type>R</subject_type>
        <city>Nemcice</city>
        <district>Pardubice</district>
        <state>CR</state>
        <city_part>Nemci</city_part>
        <house_number>12</house_number>
        <number_type></number_type>
        <street_name>V lipach</street_name>
        <number_add>12</number_add>
        <postcode>54444</postcode>
    </residence>
</subject>
..... (další ne/rezidenti)
</subjects>
</method>
</class>
</response>

```

Rozhraní Elektronická úřední deska

Rozhraní je oboustranné, události Vytvoření a Sejmnutí primárně vznikají v SSL ve správě ÚD a jsou ihned „propagovány“ do aplikace eÚD pomocí REST rozhraní.

PRINCIP KOMUNIKACE – SMĚR EÚD -> SSL, Třída VERA-VPUD

Rozhraní je realizováno jako webová služba, komunikace probíhá pomocí SOAP protokolu a zprávy jsou předávány ve formátu XML, tvar XML je definován pro každou metodu rozhraní.

Rozhraní SSL IS VERA Radnice vystupuje v roli serveru, rozhraní externích aplikací vystupuje v roli klienta. Tedy rozhraní poskytuje údaje pouze a jen na základě požadavku externí aplikace.

Metody požadavku

klient externí aplikace osloví server IS VERA Radnice a předá mu požadavek

Metody odpovědi

server IS VERA Radnice zpracuje požadavek a vrátí odpověď

- status = 0 - OK + požadovaná data
- status != 0 - Chyba + popis chyby

Při komunikaci je možné použít jedno z následujících kódování XML zpráv:

UTF-8

Windows-1250

ISO-8859-2

Kódování odpovědi je automaticky nastaveno dle kódování přijatého požadavku. Pokud je požadavek zaslán např. v kódování UTF-8, ve stejném kódování je vrácena i odpověď.

Uživatelé externích aplikací musí být zavedeni ve Správě uživatelů IS VERA RADNICE.

Upozornění: Písemnosti bez připojené přílohy nejsou předávány na elektronickou úřední desku.

PRINCIP KOMUNIKACE – SMĚR SSL -> EÚD

Rozhraní je realizováno jako REST služba a zprávy jsou předávány ve formátu JSON.

SSL IS VERA Radnice vystupuje v roli klienta, rozhraní eÚD vystupuje v roli serveru.

Události Vyvěšení a Sejmutí, které v SSL uživatel při obsluze správy úřední desky vykoná, jsou ihned „propagovány“ do aplikace eÚD pomocí REST rozhraní.

Metody rozhraní:

- Vyvěšení písemnosti

SSL zašle v REST požadavku identifikátor vyvěšení právě vyvěšené písemnosti, aplikace eÚD požadavek přijme a pomocí metod rozhraní SSL si vyžádá informace o vyvěšení, tedy detail vyvěšení pomocí metody Metoda: dej-eud-pisemnost a obsah el. souborů pomocí metody Metoda: dej-eud-prilohu, nakonec vrátí v odpovědi http status kód výsledku operace. Výsledek operaci si SSL zaznamená. Pokud je status OK, SSL si nastaví příznak synchronizováno, který znamená, že písemnost je úspěšně vyvěšena v aplikaci eÚD.

- Svěšení písemnosti

SSL zašle v REST požadavku identifikátor vyvěšení právě svěšené písemnosti, aplikace eÚD požadavek přijme a pomocí metod rozhraní SSL si vyžádá informace o sejmutí, tedy detail sejmutí pomocí metody Metoda: dej-eud-pisemnost, nakonec vrátí v odpovědi http status kód výsledku operace. Výsledek operaci si SSL zaznamená.

Pokud v procesu komunikace nastane chyba, je v SSL zatím pouze formou ikony zobrazena informace, že událost vyvěšení nebo sejmutí nebyla v aplikaci eÚD úspěšně provedena. Někjaký mechanismus opakovaného volání události není zatím implementován.

Případný soulad dat v aplikaci eÚD s daty v SSL je nyní možné zajistit pouze mechanismem opakovaného prvotního načtení dat pomocí metody Metoda: dej-eud-seznam-prvotni-nacteni.

Obecný popis struktury XML zpráv třídy Vera-vpud

Formát požadavku

```
<?xml version="1.0" encoding="iso-8859-2" ?>
<request id="hodnota"><!-- id požadavku -->
  <authorization>
    <id_user>EXSYS:user</id_user>
  </authorization>
  <class name="vera-vpud" version="2.0"><!--název třídy, verze rozhraní-->
    <method name="název metody">    <!-- název metody, např. "dej-eud-seznam" -->
      <data param1="hodnota1" param2="hodnota2" param3="hodnota3"/>
    </method>
  </class>
</request>
```

Autorizační část slouží k identifikaci externího systému a identifikaci uživatele.

Název	Popis	Datový typ	Povinnost
EXSYS	identifikace externího systému a aplikace doporučená identifikace: firma-zkratka agendy (systému) Příklad: firma Abraka s.r.o. s aplikací ABCD má doporučenou identifikaci Abraka-ABCD	C16	ano
User	login jméno uživatele v informačním systému Radnice VERA	C64	ano

Formát odpovědi

```
<?xml version="1.0" encoding="iso-8859-2"?>
<response id="hodnota"><!-- id požadavku, ke kterému odpověď patří -->
  <class name="vera-vpud" version="2.0" revision="0">
    <method name="název metody">
      <status>výsledek požadavku</status>    <!-- 0=OK / <>0=chyba -->
      <data klic="hodnota" polozka2="hodnota2" polozka3="hodnota3"/>
    </method>
  </class>
</response>
```

Formát odpovědi při chybě

```
<?xml version="1.0" encoding="iso-8859-2"?>
<response id="hodnota"><!-- id požadavku, ke kterému odpověď patří -->
  <class name="vera-vpud" version="2.0" revision="0">
    <method name="název metody">
      <status>!=0</status>
      <error_msg>Důvod, č.chyby, popis chyby</error_msg>
```

```
</method>  
</class>  
</response>
```

popis ROZHRANÍ PRO Elektronickou úřední desku
Metody jsou zpracovány v rámci rozhraní vera-vpud.

1. Seznam metod rozhraní

- Metoda: dej-eud-seznam
- Metoda: dej-eud-pisemnost
- Metoda: dej-eud-prilohu
- Metoda: storno-eud-pisemnost
- Metoda: potvrd-eud-pisemnost
- Metoda: dej-eud-seznam-prvotni-nacteni

Metoda: dej-eud-seznam

Metoda vrátí seznam identifikátorů vyvěšení těch písemností, které jsou v SSL aktuálně vyvěšené a SSL dosud neobdržela informaci z aplikace eÚD o jejich úspěšném vyvěšení (potvrzení pomocí metody Metoda: potvrd-eud-pisemnost).

Volitelně lze zadat element ke_dni. Pokud je naplněn, metoda vrátí pouze ty písemnosti, které byly (případně ještě jsou) vyvěšené k zadanému dni.

Podrobné údaje o vyvěšované písemnosti lze získat metodou Metoda: dej-eud-pisemnost.

Požadavek

```
<?xml version="1.0" encoding="iso-8859-2" ?>
<request id="226">
  <authorization>
    <id_user>EXSYS:user</id_user>
  </authorization>
  <class name="vera-vpud" version="2.0">
    <method name="dej-eud-seznam">
<data>
  <ke_dni>2013-05-31</ke_dni>
</data>
    </method>
  </class>
</request>
```

Formát odpovědi

```
<?xml version="1.0" encoding="iso-8859-2"?>
<response id="226">
  <class name="vera-vpud" version="2.0" revision="0">
    <method name="dej-eud-seznam">
      <status>0</status>
      <seznam>
        <id_vyves>158</id_vyves>
        <id_vyves>159</id_vyves>
        ...
        <id_vyves>654</id_vyves>
      </seznam>
```

```
</method>
</class>
</response>
```

Metoda: dej-eud-seznam-prvotni-nacteni

Metoda vrátí seznam identifikátorů všech vyvěšení písemností z evidence SSL.

XML struktura odpovědi je shodná, jako pro metodu Metoda: dej-eud-seznam.

Slouží pro prvotní naplnění dat do aplikace eÚD.

Metoda: dej-eud-pisemnost

Metoda vrátí detailní informace o vyvěšované písemnosti a připojených el. souborech.

Verze rozhraní 2.0 obsahuje nově přidané položky a umožňuje vyvěšení i příchozích písemností.

Požadavek

```
<?xml version="1.0" encoding="iso-8859-2" ?>
<request id="226">
  <authorization>
    <id_user>EXSYS:user</id_user>
  </authorization>
  <class name="vera-vpud" version="2.0">
    <method name="dej-eud-pisemnost">
      <data id_vyves="269"/>
    </method>
  </class>
</request>
```

Formát odpovědi

Vrací informace o písemnosti, o vyvěšení a seznam připojených příloh - el. souborů typu „hlavní dokument“ nebo „příloha“, (pro každý soubor obsahuje informace: ID, název a velikost souboru, obsah souboru lze získat metodou Metoda: dej-eud-prilohu).

Příznak storno slouží pro okamžité svěšení (hodnota A - stornováno, N - nestornováno).

```
<?xml version="1.0" encoding="iso-8859-2"?>
<response id="226">
  <class name="vera-vpud" version="2.0" revision="0">
    <method name="dej-eud-pisemnost">
      <status>0</status>
      <dokument>
        <id_vyves>269</id_vyves>
        <storno>N</storno> <!-- hodnota A / N -->
        <dat_vyveseni>28.04.2015</dat_vyveseni>
        <cas_vyveseni>09:45:56</cas_vyveseni>
        <dat_sejmout>14.05.2015</dat_sejmout>
        <dat_sejmuti></dat_sejmuti>
        <cas_sejmuti/>
        <popis>Rozhodnutí o vydání stavebního povolení</popis>
        <adresat>Jan Novák, Želivského 12, Brno</adresat>
        <zpracovatel>Správce databáze</zpracovatel>
        <odbor>Stavební odbor</odbor>
```

```

<retizekcj>OVŽP-339/2015/PSE</retizekcj>
<oblast_kod>0011</oblast_kod>
<oblast_popis>Stavební řízení</oblast_popis>
<puvodce/>
<dorg_vyves_od/>
<dorg_vyves_do/>
<prilohy>
  <priloha>
<id>183</id> <!-- Identifikátor souboru -->
<nazev>rozhodnuti.pdf</nazev> <!-- Název souboru -->
<velikost>756</velikost> <!-- Velikost souboru v kB -->
    </priloha>
    <priloha>
<id>128</id>
<nazev>oznameni.doc</nazev>
<velikost>150</velikost>
    </priloha>
  </prilohy>
</dokument>
</method>
</class>
</response>

```

Atributy vyvěšení písemnosti		
Název	Popis	Datový typ
id_vyves	Identifikátor vyvěšení písemnosti	INTETEG
Storno	Příznak stornování vyvěšení: A-stornováno, N-platný záznam Slouží pro okamžité svěšení před stanoveným termínem.	STRING-1
dat_vyveseni	Datum vyvěšení	DATE
cas_vyveseni	Čas vyvěšení	STRING-10
dat_sejmout	Datum předpokládaného termínu sejmutí	DATE
dat_sejmuti	Datum skutečného sejmutí	DATE
cas_sejmuti	Čas skutečného sejmutí	STRING-10
popis	Stručný popis písemnosti	STRING-100
adresat	Označení komu je doručováno: název/jméno a adresa u vlastní písemnosti = adresát, u doručené písemnosti = odesílatel	STRING-255
zpracovatel	Jméno zpracovatele, který písemnost vyřizuje	STRING-255
odbor	Název vyřizujícího odboru	STRING-64
retizekcj	Číslo jednací písemnosti: u vlastní písemnosti = naše Čj., u doručené písemnosti = Čj. odesílatele	STRING-50
oblast_kod	Oblast – kód, uživatelský číselník oblastí vyvěšení	STRING-4
oblast_popis	Oblast – popis	STRING-120
puvodce	Označení původce, který doručuje: název/jméno a adresa u vlastní písemnosti = původce u doručené písemnosti = odesílatel	STRING-255

dorg_vyves_od	Doručující orgán: vyvěšení na úřední desce původce, který doručuje datum vyvěšení od	DATE
dorg_vyves_do	datum vyvěšení do	DATE

Atributy el. souboru (obsah elementu příloha)		
Název	Popis	Datový typ
Id	Identifikátor souboru	INTETEG
nazev	Název souboru	STRING-64
velikost	Velikost souboru v kB	INTEGER

Pozn. vlastní obsah el. souboru lze získat metodou Metoda: dej-eud-prilohu

Metoda: dej-eud-prilohu

Metoda vrátí obsah přílohy (el. souboru) v kódování Base64.

Požadavek

```
<?xml version="1.0" encoding="iso-8859-2" ?>
```

```
<request id="226">
```

```
  <authorization>
```

```
    <id_user>EXSYS:user</id_user>
```

```
  </authorization>
```

```
  <class name="vera-vpud" version="2.0">
```

```
    <method name="dej-eud-prilohu">
```

```
      <id_priloha>183</id_priloha>
```

```
    </method>
```

```
  </class>
```

```
</request>
```

Formát odpovědi

```
<?xml version="1.0" encoding="iso-8859-2"?>
```

```
<response id="226">
```

```
  <class name="vera-vpud" version="2.0" revision="0">
```

```
    <method name="dej-eud-prilohu">
```

```
      <status>0</status>
```

```
    <priloha>
```

```
      <obsah>Binární data zakódovaná v base64</obsah>
```

```
    </priloha>
```

```
  </method>
```

```
</class>
```

```
</response>
```

Metoda: storno-eud-pisemnost

Metoda provede stornování vyvěšení písemnosti na popud elektronické úřední desky.

Požadavek

```
<?xml version="1.0" encoding="iso-8859-2" ?>
```

```
<request id="226">
```

```
  <authorization>
```

```
    <id_user>EXSYS:user</id_user>
```

```
  </authorization>
```

```
  <class name="vera-vpud" version="2.0">
```

```
    <method name="storno-eud-pisemnost">
```

```
      <data id_vyves="269"/>
```

```
    </method>
```

```
  </class>
```

```
</request>
```

Formát odpovědi

Vrací návratovou hodnotu o výsledku zpracování události stornování vyvěšení (status=0 je OK).

```
<?xml version="1.0" encoding="iso-8859-2"?>
<response id="226">
  <class name="vera-vpud" version="2.0" revision="0">
    <method name="storno-eud-pisemnost">
      <status>0</status>
    </method>
  </class>
</response>
```

Metoda: potvrd-eud-pisemnost

Metoda potvrdí, že došlo k úspěšnému vyvěšení. U potvrzených písemností bude nastaven příznak ve sloupci synchronizováno na „A“ a písemnost již nebude nabízena do seznamu k vyvěšení.

Požadavek

```
<?xml version="1.0" encoding="iso-8859-2" ?>
<request id="226">
  <authorization>
    <id_user>EXSYS:user</id_user>
  </authorization>
  <class name="vera-vpud" version="2.0">
    <method name="potvrd-eud-pisemnost">
      <data id_vyves="269">
    </method>
  </class>
</request>
```

Odpověď

Vrací návratovou hodnotu o výsledku zpracování události potvrzení vyvěšení (status=0 je OK).

```
<?xml version="1.0" encoding="iso-8859-2"?>
<response id="226">
  <class name="vera-vpud" version="2.0" revision="0">
    <method name="potvrd-eud-pisemnost">
      <status>0</status>
    </method>
  </class>
</response>
```

Rozhraní Jednotná organizační struktura

Principy komunikace aplikace JOS systému Radnice VERA®.

Využití tohoto rozhraní je komunikace s externími aplikacemi.

Následující popis definuje obsah 4. parametru (String_4) webové služby VeraWS.

Seznam metod rozhraní

- misto-uloz
- místo-oprav
- misto-dej-seznam
- misto-dej-detail
- utvar-uloz
- utvar-oprav
- utvar-dej-seznam
- utvar-dej-detail
- funkce-uloz
- funkce-oprav
- funkce-dej-seznam
- funkce-dej detail
- uzivatel-uloz
- uzivatel-oprav
- uzivatel-dej-seznam
- uzivatel-dej-detail
- role-uloz
- role-oprav
- role-dej-seznam
- role-dej-detail

Princip komunikace

Rozhraní Radnice VERA vystupuje v roli serveru, externí aplikace vystupuje v roli klienta.

Metody požadavku

Externí aplikace osloví server JOS a předá mu požadavek., JOS odpoví OK/Chyba+popis chyby.

Metody odpovědi

Server JOS zpracuje požadavek a předá odpověď (status, informace o subjektu). Každý proces (rozuměno vyřízení nějakého požadavku až do úplné konce) je identifikován svým ID. Tedy všechny dotazy/odpovědi vztahující se k danému procesu budou obsahovat request_id/response_id se stejnou hodnotou. V rámci různých procesů bude toto ID unikátní.

Změny v datech musí vždy proběhnout všechny. Pokud dojde k chybě je celá operace odmítnuta a vrácena chybová zpráva.

Autorizace přístupu

Autorizace přístupu ke službám VERA bude zajištěna v rámci protokolu https (s omezením na konkrétní IP adresu). K serveru VERA se budou klienti autorizovat přiděleným login jménem.

Třída vera-jos

Popis XML

Formát dotazu:

```
<?xml version="1.0" encoding="windows-1250" ?>
<request id="184">
  <authorization>
```

```

<id_user>EXSYS:user</id_user>
</authorization>
<class name="vera-jos" version="1.0" revision="1">
    <method name="misto-uloz">
        všechny smysluplné atributy
    </method>
</class>
</request>

```

Všechny uvedené elementy a atributy budou uloženy nebo aktualizovány. Pokud XML obsahuje neznámý element, tak je ignorován.

Pokud jsou v XML datech zmíněny identifikátory, jsou myšleny identifikátory externí aplikace.

V případě, že atribut obsahuje datum, tak bude uveden ve formátu *rrrr-mm-dd*.

Formát odpovědi

```

<?xml version="1.0" encoding="windows-1250" ?>
<response id="184">
    <class name="vera-jos" version="1.0" revision="1">
        <method name="misto-uloz">
            <status>0</status>
            <error_msg/>
        </method>
    </class>
</response>

```

Formát data

Pokud je v datech přenášen datum, pak musí být ve formátu „yyyy-mm-dd“.

Zrušení záznamu

Žádný záznam v JS nejde smazat. Pokud se založí nějaký záznam omylem, tak jeho korektní zrušení se provede tak, že se ukončí jeho platnost a jako datum platnosti do se zadá datum menší než je datum platnosti od. Například platnost od = 2015-07-01 a platnost do = 2015-06-30. Takto ukončený záznam nebyl da facto nikdy platný, a proto s ním nebude nikdy pracováno. Z vnějšího pohledu neexistuje, je tedy zrušen.

Vyhledávání entit podle identifikátorů

Externí identifikátory jsou unikátní v rámci VOJ. Interní identifikátory jsou unikátní v rámci všech záznamů dané entity. Pokud tedy vyhledáváte entitu pomocí externího identifikátoru, musíte uvést v jaké VOJ se má hledat (extIdVOJ nebo intIdVOJ). Pokud vyhledáváte entitu pomocí interního identifikátoru (intId), pak identifikátor VOJ můžete vynechat. Jen je potřeba mít na paměti, že všechny entity, se kterými se pracuje, musí patřit do stejné VOJ. Pokud toto pravidlo porušíte, tak vám rozhraní vrátí chybovou zprávu, že např. entita pro vytvoření vazby nebyla nalezena.

Práce se seznamy vazeb

Metody, které provádějí uložení nebo opravu entity, pracují se seznamy vazeb na jiné entity. Pokud bude v předaném XML element *<něco-seznam>*, kde *něco* odpovídá označení entity, na kterou se provádí vazba, tedy uživatel, funkce, místo nebo role bude rozhraní zpracovávat XML tímto scénářem:

- Zkontroluje, zdali se vazby mezi entitami liší od existujících vazeb. Shodná vazba je taková, která se shoduje v identifikaci entit na obou stranách vazby a v „platnosti od“ a „platnosti do“. Pokud zjistí, že ano, tak vytvoří kopii entit do historie, aby byl zaznamenán stav před změnou.

- Následně systém prochází seznam vazeb, a pokud najde stejnou vazbu, tak nic nedělá. Pokud existující vazbu nenajde v XML, tak ji zruší. A pokud vazbu nenajde v DB, tak ji založí. Shoda vazeb v XML a DB se provádí stejným způsobem jako v předchozím bodě.
- Pokud se platnosti vazeb mezi stejnými entitami (např. 2 vazby mezi stejným uživatelem a funkcí) platnostmi překrývají nebo přesahují platnost entit na obou koncích vazby je celý požadavek odmítnut.
- Element <platnostDo> je povinným elementem.
- Element <něco-seznam> je nepovinný a pokud není uveden, tak se vazby nemění.

Pokud je ukončována platnost entity voláním metody <entita-oprav> s vyplněným elementem <platnostDo>, která má vytvořeny vazby, pak pokud neuvedete element <něco-seznam>, tak jsou všechny vazby ukončeny k datu ukončení platnosti entity. Pokud <něco-seznam> uvedete, tak musí mít jednotlivé vazby korektně vyplněn element <platnostDo>, jinak bude celý požadavek odmítnut. Pokud nebudou v seznamu uvedeny všechny existující vazby, tak neuvedené vazby budou ukončeny k datu ukončení platnosti entity.

Zadavatel poskytne kompletní dokumentaci jednotlivých metod vítěznému uchazeči po podpisu Smlouvy.

Příklad detailního popisu metody:

Metoda: misto-uloz

Metoda umožňuje vkládat záznamy o místech (budovy, místnosti)

Vstupní data:

Označení	Název	Popis
misto, atribut extIdVOJ	Vrcholová organizační jednotka, externí identifikátor	Slouží k určení do které organizace bude záznam uložen nebo ve které bude hledán. Povinný je buď <i>extIdVOJ</i> nebo <i>intIdVOJ</i> .
misto, atribut intIdVOJ	Vrcholová organizační jednotka, interní identifikátor	Povinný je buď <i>extIdVOJ</i> nebo <i>intIdVOJ</i> .
misto, atribut extId	Externí identifikátor místa	Jednoznačný identifikátor záznamu. Slouží při opravě k vyhledání záznamu. Musí být v rámci VOJ unikátní.
misto, atribut intId	Identifikátor místa	Je využíván hlavně ve funkcích <i>dej</i> v případě, že ext. syst. pracuje se záznamy, které nezaložil.
nadrazeneMisto atribut extId	Externí identifikátor nadřazeného místa	Určuje nadřazené místo. Slouží k vytvoření hierarchické struktury míst. Pokud zakládáte místo na první úrovni stromu (nemá nadřazené místo), tak se element nesmí uvést.
nadrazeneMisto atribut intID	Identifikátor nadřazeného místa	dtto
druhMista	Druh místa	Pole <i>Druhu místa</i> z číselníku druhů míst. ¹
zkratka	Zkratka	Unikátní označení místa. Musí být unikátní v rámci platn. záznamů VOJ. GUI konvertuje pole ZKRATKA na velká písmena. Rozhraní tuto konverzi nedělá.
nazev	Název	Název místa

¹ Je potřeba zajistit shodu mezi vyplněním v JOS a externím systémem.

Označení	Název	Popis
lokalita	Lokalita	Označení lokality pro generování čárového kódu. Využívá se v agendě Majetek.
platnostOd	Datum platnosti od	
platnostDo	Datum platnosti do	
utvar-seznam	Kontejner pro atributy týkající se vazby místo-útvary	Seznam vazebních identifikátorů na útvary
utvar, atribut extId	Externí identifikátor útvaru	Identifikátor vazební entity útvar. Povinný je buď <i>extId</i> nebo <i>intId</i> .
utvar, atribut intId	Interní identifikátor útvaru	Povinný je buď <i>extId</i> nebo <i>intId</i> .
platnostOd	Datum platnosti od	Platnost vazby (nikoli vazební entity).
platnostDo	Datum platnosti do	Platnost vazby (nikoli vazební entity).
funkce-seznam	Kontejner pro atributy týkající se vazby místo-funkce	Seznam vazebních identifikátorů na funkční místa
funkce, atribut extId	Externí identifikátor funkce	Identifikátor vazební entity funkce. Povinný je buď <i>extId</i> nebo <i>intId</i> .
funkce, atribut intId	Interní identifikátor funkce	Povinný je buď <i>extId</i> nebo <i>intId</i> .
platnostOd	Datum platnosti od	Platnost vazby (nikoli vazební entity).
platnostDo	Datum platnosti do	Platnost vazby (nikoli vazební entity).

Povinnost položky vyjadřuje povinnost v okamžiku ukládání nového záznamu. Otazník ve sloupci „Povinné“ → upřesnění je ve sloupci „Popis“.

Příklad XML (pouze obsah elementu *method*).

```
<method name="misto-uloz">
  <misto extId="ABC" idVOJ="87654321">
    <druhMista>Budova</druhMista>
    <zkratka>BSR</zkratka>
    <lokalita>5588226644</lokalita>
    <platnostOd>2010-01-31</platnostOd>
    <utvar-seznam>
      <utvar extId="6587">
        <platnostOd>2010-01-31</platnostOd>
      </utvar>
    </utvar-seznam>
  </misto>
</method>
```

!—Dále mohou pokračovat další útvary navázané na dané místo.

V seznamu může být jeden útvary vícekrát s různou platností. V tom případě

je potřeba dávat pozor na kontroly platnosti vazeb. Platnosti vazeb se nesmí překrývat! →

```
</utvar-seznam>
<funkce-seznam>
  <funkce extId="995511">
    <platnostOd>2010-01-31</platnostOd>
  </funkce>
</funkce-seznam>
```

</funkce>

<!--Dále mohou pokračovat další funkce navázané na dané místo.

V seznamu může být jedna funkce vícekrát s různou platností. V tom případě

je potřeba dávat pozor na kontroly platnosti vazeb. Platnosti vazeb se nesmí překrývat! →

</funkce-seznam>

</misto>

</method>

Výstupní data:

| Označení | Název | Popis |
|-----------|--------|--|
| status | Status | Oznámení o výsledku operace |
| error_msg | | V případě chybového statusu, bližší popis chyby. |

Otazník ve sloupci „Povinné“ → upřesnění je ve sloupci „Popis“

Tabulka chybových stavů

| Status | Význam statusu | Obvyklá zpráva *) | |
|--------|------------------------------------|---|--|
| 200 | bez chyby - jiné | Úspěšně provedeno | |
| 201 | ulozeno/vloženo | Data změněna | |
| 202 | bez chyby - beze zmen | Beze změny | |
| 400 | chybný požadavek – jiné | | |
| 401 | neautorizovano | | |
| 406 | údaje neprošly validací | Údaje obsahují chyby | |
| 409 | konflikt s jiným požadavkem | Údaje změnil jiný uživatel | |
| 410 | entita nenalezena | Data nenalezena | |
| 417 | chybí povinný element nebo atribut | | |
| 500 | interní chyba serveru | Není posílána zpráva.
Více informací lze získat v logu aplikace. | |

*) Zpráva popisující chybu se může lišit. Podle konkrétního důvodu chyby bývá doplněna vysvětlujícím textem, který lze zobrazit uživateli.

Webová služba VeraWS

Informační systém Radnice VERA disponuje rozhraním pro komunikaci s externími aplikacemi. Rozhraní je realizováno jako webservises (http, soap). Z povahy použité technologie se jedná o synchronní komunikaci párových zpráv dotaz-odpověď (request-response). Ve zprávě „dotaz“ je uvedena metoda, kterou externí systém volá a k ní odpovídající data. Ve zprávě „odpověď“ je informace o provedení příslušné metody, případně návratová data. Formát „dotazu“ i „odpovědi“ je vždy XML.

Webová služba je jednotná pro všechny druhy komunikace (všechna přenášená data). Zpráva, kterou služba očekává, se skládá ze 4 částí:

String_1 Identifikace externího systému. Určuje se dohodou při implementaci.

String_2 Identifikace uživatele (login). Ověřuje se, zdali uvedený uživatel má potřebná práva.

String_3 Označení třídy, která bude zpracovávat data. Např. „vera-jos“.

String_4 Data ve formátu XML. Obsah datové části zprávy je popsán v samostatném dokumentu.

Služba standardně poslouchá na URL „https://<host>:8443/veraws/ws“.

WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="Vera" targetNamespace="urn:Vera/wsdl" xmlns:tns="urn:Vera/wsdl"
xmlns=http://schemas.xmlsoap.org/wsdl/ xmlns:xsd=http://www.w3.org/2001/XMLSchema
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/">
  <types/>
  <message name="Vera_get">
    <part name="String_1" type="xsd:string"/>
    <part name="String_2" type="xsd:string"/>
    <part name="String_3" type="xsd:string"/>
    <part name="String_4" type="xsd:string"/>
  </message>
  <message name="Vera_getResponse">
    <part name="result" type="xsd:string"/>
  </message>
  <portType name="Vera">
    <operation name="get" parameterOrder="String_1 String_2 String_3 String_4">
      <input message="tns:Vera_get"/>
      <output message="tns:Vera_getResponse"/>
    </operation>
  </portType>
  <binding name="VeraBinding" type="tns:Vera">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="rpc"/>
    <operation name="get">
      <soap:operation soapAction=""/>
      <input>
        <soap:body use="literal" namespace="urn:Vera/wsdl"/>
      </input>
      <output>
        <soap:body use="literal" namespace="urn:Vera/wsdl"/>
      </output>
    </operation>
  </binding>
  <service name="Vera">
    <port name="VeraPort" binding="tns:VeraBinding">
      <soap:address location="REPLACE_WITH_ACTUAL_URL"/>
    </port>
  </service>
</definitions>
```