



T-MAPY spol. s r. o.  
Špitálská 150  
500 03 Hradec Králové  
tel.: 498 511 111  
fax: 495 513 371  
e-mail: [info@tmapy.cz](mailto:info@tmapy.cz)  
<http://www.tmapy.cz>  
<http://www.tmapserver.cz>

---

## Usnesení a úkoly volených orgánů eUsnIDM konektor

---

### Obsah dokumentu:

1	Změnové informace dokumentu .....	2
2	eUsnIDM konektor .....	3
2.1	WSDL soubor .....	3
2.2	URL rozhraní .....	3
3	Datové typy .....	4
3.1	typ unit .....	4
3.2	typ user .....	4
3.3	typ role .....	5
3.4	typ right .....	5
3.5	typ result .....	5
3.6	typ array pole .....	6
4	Číselníky používané v konektoru .....	7
4.1	Číselník unit .....	7
5	Metody poskytované eUsnIDM konektorem .....	8
5.1	Metoda GetUsersRolesAndRights .....	8
5.2	metoda SetUser .....	8
5.3	Metoda SetActive .....	9
5.4	Metoda setRole .....	9
5.5	Metoda SetRights .....	10
5.6	Metoda SetUsersRolesAndRights .....	10
5.7	Metoda SetUnit .....	11
6	Workflow práce s eUsnIDM .....	12
6.1	Načtení oprávnění všech uživatelů do IDM .....	12
6.2	Vytvoření nového uživatele a jeho role bez přidělení práv .....	12
6.3	Přidělení práv k roli uživatele .....	12
6.4	Smazání uživatele .....	12
6.5	Smazání role u daného uživatele .....	12
6.6	Smazání práv u dané role .....	12

# 1 Změnové informace dokumentu

Provedené změny v dokumentu:

Verze dokumentu	Popis změn	Provedl
15. 7. - 21. 7. 2014	Vznik dokumentu, soupis metod	Malý
13. 8. 2014	Přidána metoda SetUsersRolesAndRights	Malý
18. 8. 2014	Přidána metoda SetUnit	Malý

## 2 eUsnIDM konektor

Pro přístup IDM systémů při řízení uživatelských oprávnění v aplikaci Usnesení a úkoly volených orgánů je k dispozici eUsnIDM konektor. Služby tohoto konektoru jsou poskytovány jako serverová část a jsou obsluhovány pomocí standardizovaného SOAP protokolu.

### 2.1 WSDL soubor

Základem eUsnIDM konektoru je WSDL soubor definující základní typy a metody:

Aktuální WSDL dokument je dostupný na adrese

<http://demo-zis.tmapserver.cz/rt/plugins/idm/server.php?wsdl>

### 2.2 URL rozhraní

Konektor eUsnIDM je dostupný (pokud není řečeno jinak), u zákazníka na adrese  
[http://ADRESA\\_SERVERU\\_U\\_ZAKAZNIKA/rt/plugins/idm/server.php](http://ADRESA_SERVERU_U_ZAKAZNIKA/rt/plugins/idm/server.php)

WSDL soubor je dostupný po zadání url  
[http://ADRESA\\_SERVERU\\_U\\_ZAKAZNIKA/rt/plugins/idm/server.php?wsdl](http://ADRESA_SERVERU_U_ZAKAZNIKA/rt/plugins/idm/server.php?wsdl)

## 3 Datové typy

Datové typy užívané v eUsnIDM konektoru jsou definovány v samotném WSDL souboru.

### 3.1 typ unit

Definice jednotky organizační struktury úřadu. Každá unita obsahuje identifikátor, název, zkratku a ancestorID, kdy ancestorID je definice předka, pod kterou unitu patří daná unita.

```
<xsd:complexType name="unit">
  <xsd:element name="unitID" type="xsd:int" nillable="false"/>
  <xsd:element name="unitName" type="xsd:string" nillable="false" maxLength="100"/>
  <xsd:element name="unitAbbreviation" type="xsd:string" nillable="false" maxLength="10"/>
  <xsd:element name="ancestorID" type="xsd:int" nillable="false" maxLength="10"/>
  <xsd:element name="active" type="xsd:int" nillable="true" maxLength="10"/>
</xsd>
```

Pozn: V unitách jsou definovány kromě samotné organizační struktury úřadu definovány i jednotlivé výbory a komise daného úřadu. Identifikace unity mezi systémem eUsnIDM a vlastním IDM je provedena pomocí parametru unitID. Je potřeba, aby IDM bylo schopno tuto hodnotu uchovávat a pracovat s ní (z důvodu aktualizace unit)

### 3.2 typ user

Definice uživatele, který může pracovat s aplikací.

```
<xsd:complexType name="user">
  <xsd:element name="surname" type="xsd:string" nillable="true" maxLength="40"/>
  <xsd:element name="firstname" type="xsd:string" nillable="true" maxLength="40"/>
  <xsd:element name="degree_infrontof" type="xsd:string" nillable="true" maxLength="40"/>
  <xsd:element name="degree_behind" type="xsd:string" nillable="true" maxLength="40"/>
  <xsd:element name="login" type="xsd:string" nillable="true" maxLength="30"/>
  <xsd:element name="email" type="xsd:string" nillable="true" maxLength="60"/>
  <xsd:element name="tel_work" type="xsd:string" nillable="true" maxLength="50"/>
  <xsd:element name="mob_work" type="xsd:string" nillable="true" maxLength="50"/>
  <xsd:element name="tel_home" type="xsd:string" nillable="true" maxLength="50"/>
  <xsd:element name="mob_home" type="xsd:string" nillable="true" maxLength="50"/>
  <xsd:element name="active" type="xsd:int" nillable="true" maxLength="50"/>
  <xsd:element name="party" type="xsd:string" nillable="true" maxLength="200"/>
  <xsd:element name="role" type="tns:roleArray" nillable="true" maxLength="200"
    minOccurs="0"/>
</xsd>
```

Identifikace uživatele mezi systémem eUsnIDM a vlastním IDM je provedena pomocí parametru login.

### 3.3 typ role

Definice role uživatele v aplikaci.

```
<xsd:complexType name="role">
  <xsd:element name="roleID" type="xsd:int" nillable="false"/>
  <xsd:element name="roleName" type="xsd:string" nillable="false" maxLength="100"/>
  <xsd:element name="unitAbbreviation" type="xsd:int" nillable="true" maxLength="100"/>
  <xsd:element name="active" type="xsd:int" nillable="true" maxLength="100"/>
  <xsd:element name="rights" type="tns:rightArray" nillable="true" maxLength="100"/>
</xsd>
```

Identifikace role mezi systémem eUsnIDM a vlastním IDM je provedena pomocí parametru roleID. Je potřeba, aby IDM bylo schopno tuto hodnotu uchovávat a pracovat s ní (z důvodu aktualizace rolí)

### 3.4 typ right

Definice samotného práva v aplikaci.

```
<xsd:complexType name="right">
  <xsd:element name="rightName" type="xsd:string"/>
  <xsd:element name="value" type="xsd:int"/>
</xsd>
```

Systém vrátí seznam práv pro danou roli. Název práva (rightName) je tvořen hodnotou APP\_NAME, kdy APP je zkratka aplikace (např. rm pro radu města, zm pro zastupitelstvo města) a NAME je název samotného práva (např. editor, auditor). Hodnota value určuje, zda je dané právo přiřazeno (value=1) nebo není (value=0)

### 3.5 typ result

Návratová hodnota provedení metody.

```
<xsd:complexType name="result">
  <xsd:element name="status" type="xsd:string"/>
  <xsd:element name="description" type="xsd:string"/>
</xsd>
```

V případě, že volaná metoda proběhla v pořádku, je vrácen navíc tento komplexní typ. Hodnota status=OK interpretuje správné provedení metody.

V případě, že došlo při volání metody k chybě, je vráceno volání pomocí standardizované hodnoty SoapFault.

### 3.6 typ array pole

V případě vrácení více hodnot daného typu jsou hodnoty vráceny v odpovídajícím poli, obalené elementem Array

```
<xsd:complexType name="unitArray">
  <xsd:element name="unit" type="tns:unit" maxOccurs="unbounded" minOccurs="0" nillable="true"/>
</xsd:complexType>

<xsd:complexType name="userArray">
  <xsd:element name="user" type="tns:user" maxOccurs="unbounded" minOccurs="0" nillable="true"/>
</xsd:complexType>

<xsd:complexType name="roleArray">
  <xsd:element name="role" type="tns:role" maxOccurs="unbounded" minOccurs="0" nillable="true"/>
</xsd:complexType>

<xsd:complexType name="rightArray">
  <xsd:element name="right" type="tns:right" maxOccurs="unbounded" minOccurs="0" nillable="true"/>
</xsd:complexType>
```

## 4 Číselníky používané v konektoru

Pro správnou obsluhu konektoru je potřeba udržovat organizační strukturu. Identifikace organizační struktury mezi systémem eUsnIDM a vlastním IDM je prováděna pomocí zkratky organizační struktury.

### 4.1 Číselník unit

Číselník unit je dostupný v konektoru zavoláním metody GetUnits

#### **GetUnits**

- IN: null (xsd:string)
- OUT: units

## 5 Metody poskytované eUsnIDM konektorem

Všechny metody jsou identifikovány pomocí počítačícího názvu na metody GET (pro získání potřebných údajů) a SET (pro nastavení odpovídajících údajů).

V této verzi eUsnIDM konektoru jsou definovány následující funkce:

### 5.1 Metoda GetUsersRolesAndRights

#### GetUsersRolesAndRights

- IN: userLogin (xsd:string)
- OUT: result (tns:result), users (tns:userArray)

#### Vstupní parametry:

userLogin – login konkrétního uživatele

Metoda vrátí seznam atributů uživatele, seznam jeho rolí v aplikaci a seznam práv přiřazených k dané roli. Pokud není výběr ovlivněn vstupní hodnotou userLogin, pak je vrácen kompletní seznam uživatelů v aplikaci.

#### Výstupní parametry:

result – potvrzení výsledku volání metody  
users - pole hodnot výsledných uživatelů.

### 5.2 metoda SetUser

#### SetUser

- IN: user (tns:user)
- OUT: result (tns:result)

#### Vstupní parametry:

user - vstupní pole hodnot uživatele

Metoda umožní nastavit vlastnosti uživatele. Identifikace uživatele se provádí pomocí hodnoty login. Pokud login uživatele nebude nalezen v aplikaci eUsnesení, bude záznam založen jako nový. Pokud login bude nalezen, dojde k aktualizaci konkrétního záznamu.

Pokud hodnota active nebude uvedena, bude automaticky nastavena na active=1.

#### Výstupní parametry:

result – potvrzení výsledku volání metody



## 5.3 Metoda SetActive

### SetActive

- IN: userLogin (xsd:string), active (xsd:int)
- OUT: result (tns:result)

#### Vstupní parametry:

userLogin – login konkrétního uživatele  
active - příznak uživatele

Metoda umožní nastavit, jestli je uživatel aktivní v aplikaci USN (může se přihlásit do aplikace) nebo jestli je uživatel "skrytý".

Hodnota active=1 - uživatel může pracovat s aplikací, jeho jméno se nabízí v výběrových seznamech

Hodnota active=0 - uživatel se nemůže přihlásit do aplikace, jeho jméno se nenabízí v výběrových seznamech

Pokud hodnota active nebude uvedena, bude automaticky nastavena na active=1.

#### Výstupní parametry:

result – potvrzení výsledku volání metody

## 5.4 Metoda setRole

### SetRole

- IN: userLogin (xsd:string), role (tns:role)
- OUT: result (tns:result)

#### Vstupní parametry:

userLogin – login konkrétního uživatele  
role – pole hodnot dané role

Metoda umožňuje změnu nebo vytvoření role. Rolí je chápáno zařazení pracovníka do organizační struktury. Pokud bude uvedena hodnota roleID, pak dojde k aktualizaci dané role. Pokud hodnota roleID nebude uvedena, dojde k založení nové role.

Při aktualizaci nebo vytvoření role musí být správně identifikováno organizační zařazení pomocí zkratky dané organizační struktury. Pokud nebude daná zkratka nalezena, bude vráceno SoapFault.

Pokud hodnota active nebude uvedena, bude automaticky nastavena na active=1.

#### Výstupní parametry:

result – potvrzení výsledku volání metody

## 5.5 Metoda SetRights

### SetRights

- IN: roleID (xsd: int), rights (tns:rightArray)
- OUT: result (tns:result)

#### Vstupní parametry:

roleID – identifikátor role  
rights - pole práv

Metoda umožní nastavit práva pro danou roli. Pro zjednodušení stačí zaslat seznam práv, které se mají nastavit. Pokud hodnota value nebude u daného práva uvedena, bude automaticky nastavena u zaslaných práv na value=1.

#### Výstupní parametry:

result – potvrzení výsledku volání metody

## 5.6 Metoda SetUsersRolesAndRights

### SetUsersRolesAndRights

- IN: user (tns:user)
- OUT: result (tns:result)

#### Vstupní parametry:

user - vstupní pole hodnot uživatele

Metoda umožní nastavit kompletní údaje o daném uživateli. Očekává se uvedení všech údajů o uživateli - včetně jeho rolí a práv k daným rolím. Pokud některé hodnoty nebudou uvedeny, budou v aplikaci eUsnesení nastaveny jako neviditelné.

V sekci role je uvedeno zařazení pracovníka do organizační struktury. Pokud bude uvedena hodnota roleID, pak dojde k aktualizaci dané role. Pokud hodnota roleID nebude uvedena, dojde k založení nové role.

Při aktualizaci nebo vytvoření role musí být správně identifikováno organizační zařazení pomocí zkratky dané organizační struktury. Pokud nebude daná zkratka nalezena, bude vráceno SoapFault.

Pokud hodnota role (parametr active) nebude uvedena, bude automaticky nastavena na active=1.

U každé role je také očekáván seznam práv. Pro zjednodušení stačí zaslat seznam práv, které se mají nastavit. Pokud hodnota value nebude u daného práva uvedena, bude automaticky nastavena u zaslaných práv na value=1. Pokud právo nebude uvedeno, bude nastaveno v aplikaci na hodnotu 0

#### Výstupní parametry:

result – potvrzení výsledku volání metody

## 5.7 Metoda SetUnit

### SetUnit

- IN: unit (tns:unit)
- OUT: result (tns:result)

#### Vstupní parametry:

unit – pole hodnot dané unity

Metoda umožňuje změnu nebo vytvoření unity. Unitou je chápána jednotka organizační struktury. Pokud bude uvedena hodnota unitID, pak dojde k aktualizaci dané unity. Pokud hodnota unitID nebude uvedena, dojde k založení nové unity.

Při aktualizaci nebo vytvoření unity je potřeba také specifikovat parametr ancestorID, který označuje ID nadřazené organizační struktury. Pokud jde o první úroveň (nejvýše postavenou), použijte hodnotu 1.

Pokud hodnota active nebude uvedena, bude automaticky nastavena na active=1.

#### Výstupní parametry:

result – potvrzení výsledku volání metody

## 6 Workflow práce s eUsnIDM

Pro obecnější představu by měly sloužit následující workflow

### 6.1 Načtení oprávnění všech uživatelů do IDM

- `GetUsersRolesAndRights` (bez uvedení `userLogin`)

### 6.2 Vytvoření nového uživatele a jeho role bez přidělení práv

- `SetUser` (s uvedením `loginu`)
- `SetRole` (bez uvedení `roleID`)

výsledkem je založení relevantního uživatele, který v aplikaci eUsnIDM bude moci vyhledávat pod svým jménem, ale není mu umožněna žádná editace (nemá nastavena žádná práva)

### 6.3 Přidělení práv k roli uživatele

- `SetRole` (s uvedením `roleID` a pole `rights`)

### 6.4 Smazání uživatele

- `SetActive` (s uvedením `userLogin` a parametrem `active=0`)

Aplikace primárně nemaže zadané hodnoty uživatele. Pro mazání je tedy potřeba použít příznak `active=0`, kdy dojde k zneaktivnění uživatele.

### 6.5 Smazání role u daného uživatele

- `SetRole` (s uvedením `roleID` a parametrem `active=0`)

Aplikace primárně nemaže zadané hodnoty uživatele. Pro mazání je tedy potřeba použít příznak `active=0`, kdy dojde k zneaktivnění role.

### 6.6 Smazání práv u dané role

- `SetRights` (s uvedením `roleID` a bez pole `rights`)

V případě, kdy nejsou zaslána práva k dané roli, dojde v aplikaci k nastavení všech práv dané role na hodnotu `value=0`.