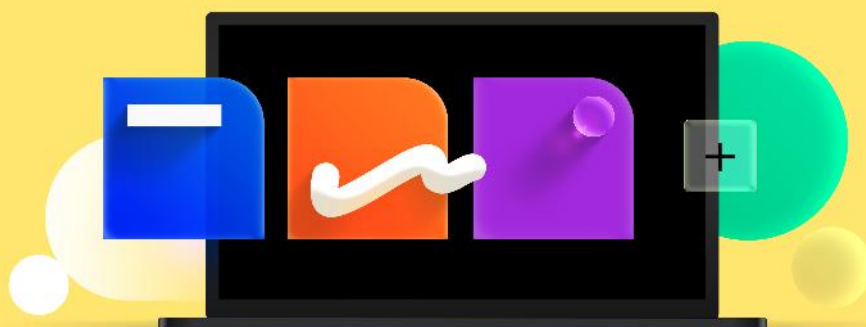




XPath výrazy, funkce a aplikační události

Aktualizováno: 17.03.2017

Obsah

1	Úvod	3
2	XPath – úvod	3
3	XPath – terminologie	4
3.1	Uzly	4
3.2	Vztahy uzlů	4
4	XPath – syntax	5
4.1	Výběr uzlů	5
4.2	Identifikátory osy	6
4.3	Testy uzlu a predikáty	6
4.4	Zástupné znaky	7
4.5	Operátory	8
5	XPath – funkce	9
5.1	Popis standardních funkcí XPath	9
5.2	Rozšiřující funkce jazyka XPath implementované na úrovni MSXML4 parseru	11
6	Funkce a aplikační události v rozšíření 602XML	13
6.1	Popis funkcí rozšíření 602XML modulu XMLcore pro výpočty	13
6.2	Popis funkcí rozšíření 602XML modulu XMLcore pro formulářové objekty	18
6.3	Popis funkcí rozšíření 602XML modulu XMLcore pro práci s certifikáty	29
6.4	Popis aplikačních událostí Form Filler	31
7	Použití XPath v aplikacích 602XML	42

1 Úvod

Tento dokument stručně popisuje principy jazyka XPath verze 1.0, jeho syntax, možné operace a funkce, a to včetně rozšíření 602XML. Dokument by měl přispět ke zjednodušení návrhu složitějších konstrukcí v XML formulářích pomocí aplikace Form Designer.

2 XPath – úvod

XPath je jazykem, který stojí na pozadí téměř veškeré práce s XML. Slouží k adresování částí XML dokumentu. Pomocí výrazů jazyka XPath je možno vyhledávat informace v XML dokumentu a dále pohybovat se mezi elementy a atributy v tomto dokumentu. XPath je součástí standardu XSLT podle specifikace W3C.

Pomocí XPath výrazů je tedy možno vybírat množiny uzlů (node-sets) v XML dokumentu. XPath výrazy jsou konstruovány obdobným způsobem, jakým jsou adresovány soubory a adresáře v tradičním počítačovém systému souborů. XPath specifikace obsahuje přes 100 zabudovaných funkcí – funkce pro práci s řetězcí, číslý, datovými a časovými údaji, funkce pro manipulaci s uzly a prvky QName (kvalifikované jméno), práci se sekvencemi, logickými hodnotami a další.

Podrobněji se s jednotlivými součástmi jazyka XPath seznámíme v následujících kapitolách.

3 XPath – terminologie

3.1 Uzly

Specifikace XPath definuje 7 druhů uzlů – elementy, atributy, texty, jmenné prostory (namespace), předpisy (processing-instruction), komentáře (comment) a uzly dokumentů (root). Všechny XML dokumenty jsou chápány jako stromy uzlů. Kořen takového stromu je nazýván uzlem dokumentu (tzv. kořenový uzel).

3.2 Vztahy uzlů

Mezi uzly ve stromu příslušného XML dokumentu definujeme jednotlivé vztahy, které usnadňují orientaci a práci s XML dokumentem pomocí XPath. Role v těchto vztazích jsou – rodič (parent) neboli přímý předek, dítě (child) neboli přímý potomek, sourozenec (sibling), předek (ancestor) a potomek (descendant).

Příklad:

```
        </dsig:SignatureProperty>
    </dsig:SignatureProperties>
</dsig:Object>
<dsig:Object>
  <root>
    <adresa>
      <ulice>Havrdova 7/1258</ulice>
      <psc>12589</psc>
      <mesto>Praha</mesto>
      <stat>Česká republika</stat>
    </adresa>
  </root>
</dsig:Object>
</dsig:Signature>
```

V tomto příkladu je element *adresa* rodičem (a zároveň předchůdcem) elementů *ulice*, *psc*, *mesto* a *stat*. Elementy *ulice*, *psc*, *mesto* a *stat* jsou dětmi (a zároveň potomky) elementu *adresa*, zároveň jsou mezi sebou sourozenci. Element *root* je předchůdcem elementů *ulice*, *psc*, *mesto* a *stat*, a tyto jsou naopak jeho potomky.

Význam těchto vztahů bude objasněn v následující kapitole, která se zabývá vlastní syntaxí jazyka XPath pro výběr uzlů a vyhodnocování výrazů.

4 XPath – syntax

4.1 Výběr uzlů

Nejdůležitější konstrukcí jazyka XPath je zřejmě konstrukce cesty ke XML uzlu (Location Path). Cesta obsahuje vždy několik **kroků**, přičemž každý z nich sestává z **identifikátoru osy**, **testu uzlu** a tzv. **podmínky (predikátu)**. Povinný je pouze test uzlu. Jednotlivé kroky XPath výrazu se pak spojují lomítky, vyhodnocují se zleva doprava a vždy se vychází z předchozích výsledků. Na začátku zpracování celého výrazu se vychází z aktuálního uzlu. Výsledkem může obecně být libovolný typ uzlu, tedy nejen element nebo atribut, ale třeba i textový uzel.

Při vyhodnocování cesty se nejdříve podle tzv. identifikátoru osy určí uzly, které se budou v tomto kroku výrazu zpracovávat. Dle výchozí osy (*child: :*) jsou to uzly podřízené aktuálnímu uzlu. Množina vybraných uzlů se omezí podle testu uzlu, kterým může být třeba název elementu, a nakonec se ještě zohlední dodatečné podmínky. Uzly, které v tomto procesu obstály, postupují do dalšího kroku. Ten se pak bude vyhodnocovat postupně pro všechny uzly, které do něj postoupily. Takto se projde celý XPath výraz. Výsledkem výběru je pak množina uzlů, která prošla vyhodnocením.

Stejně jako u adresářové struktury jednotlivé kroky XPath výrazu oddělujeme pomocí lomítka (/). Pokud je lomítko na začátku výrazu, znamená to, že výraz není vztažen k aktuálnímu elementu, ale počítá se od kořene dokumentu. Dvě lomítka bezprostředně za sebou (//) slouží k překonání víceúrovňové struktury. Pokud jsou dvě lomítka na začátku výrazu, opět se jedná o absolutní cestu od kořene dokumentu, takže lze snadno vybrat libovolný element odkudkoli.

Příklady jednoduchých XPath výrazů pro výběr uzlů z příkladového souboru „XPath sample.xml“ :

XPath výraz	Aktuální uzel	Vyhodnocení
/root/clovek/jmeno	jakýkoliv	Všechny elementy <i>jmeno</i> , které jsou přímými potomky elementu <i>clovek</i> , jež je přímým potomkem elementu <i>root</i>
/root	jakýkoliv	Kořenový element dokumentu
telefon/domu	clovek	Všechny elementy <i>domu</i> , které jsou přímými potomky elementu <i>telefon</i> , obsaženého v elementu <i>clovek</i>
//domu	jakýkoliv	Všechny elementy <i>domu</i> v jakémkoliv kontextu
root//domu	kořen dokumentu	Všechny elementy <i>domu</i> , i když nejsou přímými potomky elementu <i>root</i>

4.2 Identifikátory osy

Testu uzlu může předcházet takzvaný *identifikátor osy*. Ten určuje směr procházení XML dokumentu, tedy říká, kterým směrem se uzly k vyhodnocení budou vybírat. Pokud identifikátor osy neuvedeme, použije se výchozí osa `child::`, a dále tedy postupují pouze potomci aktuálního uzlu.

Identifikátor	Uzly k vyhodnocení
<code>child::</code>	Přímí potomci aktuálního uzlu.
<code>descendant::</code>	Všichni potomci aktuálního uzlu.
<code>descendant-or-self::</code>	Aktuální uzel a všichni potomci.
<code>self::</code>	Aktuální uzel.
<code>ancestor-or-self::</code>	Aktuální uzel a všichni jeho předci.
<code>ancestor::</code>	Všichni předci aktuálního uzlu.
<code>parent::</code>	Rodič aktuálního uzlu.
<code>following::</code>	Všechny uzly, které jsou v toku XML dokumentu za aktuálním uzlem.
<code>preceding::</code>	Všechny uzly, které jsou v toku XML dokumentu před aktuálním uzlem.
<code>following-sibling::</code>	Všichni následující sourozenci aktuálního uzlu.
<code>preceding-sibling::</code>	Všichni předcházející sourozenci aktuálního uzlu.
<code>attribute::</code>	Atributy aktuálního uzlu.
<code>namespace::</code>	Deklarované jmenné prostory

4.3 Testy uzlu a predikáty

Test uzlu dále vymezuje množinu uzlů, která byla určena identifikátorem osy. Rozlišujeme dva základní typy testů uzlu, a to *uzel určený názvem* a *uzel určený typem*. Test uzlu nazývaný *cílený test předpisu* (Targeted Processing Instruction Test) v 602XML nevyužíváme.

Uzel určený názvem je základním testem uzlu. Vybere všechny XML elementy s daným názvem. Tento test zapíšeme jednoduchým zápisem testovaného názvu elementu.

Uzel určený typem bere ohled na typ uzlu, a vybírá pouze určený typ. Tento test se zapisuje pomocí určení typu uzlu následovaným prázdnými kulatými závorkami. K dispozici jsou tyto volby: `comment()`, `text()`, `processing-instruction()` a `node()`.

Do hranatých závorek se dále zapisují další podmínky (predikáty), které zužují výsledky předchozího vyhodnocení. Tyto podmínky se používají k nalezení specifického uzlu nebo uzlu, který obsahuje danou hodnotu. Pokud místo podmínky napíšeme pouze celé číslo, vybere podmínka pouze uzel umístěný na pozici s daným pořadovým číslem.

Příklady testů uzlů v XPath výrazech pro výběr uzlů z příkladového souboru „*XPath sample.xml*“

XPath výraz	Aktuální uzel	Vyhodnocení
<code>jmeno</code>	clovek	Všechny podřízené elementy s daným názvem
<code>root</code>	kořen dokumentu	Kořenový element dokumentu
<code>//text()</code>	jakýkoliv	Vybere všechny textové uzly v dokumentu.
<code>//clovek[@id="3"]</code>	jakýkoliv	Všechny elementy <i>clovek</i> , jejichž atribut <i>id</i> má hodnotu 3.
<code>*[2]</code>	jakýkoliv	V pořadí druhý element podřízený aktuálnímu.

4.4 Zástupné znaky

Jelikož některé konstrukce nacházejí použití v jazyce XPath velice často, existují pro ně zkrácené tvary a zástupné znaky. Velmi doporučujeme tyto zkrácené tvary používat, protože zvyšují přehlednost celého XPath výrazu a usnadňují tak práci s těmito výrazy. Mezi tyto znaky patří:

- **Hvězdička (*)** – hvězdička použitá jako test uzlu vybírá všechny poskytnuté elementy.
- **Tečka (.)** – tečka nahrazuje odkaz na sebe sama. Je náhražkou celého kroku, takže k němu již nelze připojit podmínky.
- **Dvě tečky (..)** – dvě tečky (stejně jako v adresářové cestě) umožní pohyb o jednu úroveň výše. Jsou opět náhražkou celého kroku, takže k němu již nelze připojit podmínky.
- **Zavináč (@)** – zavináč nahrazuje identifikátor osy `attribute:..`. Jako test uzlu pak obvykle poslouží název atributu.
- **Kombinace (@*)** – tato kombinace použitá jako test uzlu vybírá všechny poskytnuté atributy
- **Příkaz node()** – jako test uzlu vybírá všechny uzly (bez ohledu na jejich typ)

Příklady použití zástupných znaků v XPath výrazech pro výběr uzlů z příkladového souboru „*XPath sample.xml*“ :

XPath výraz	Aktuální uzel	Vyhodnocení
.	jakýkoliv	Vybere aktuální uzel
./*	jakýkoliv	Vybere všechny přímé potomky aktuálního uzlu
..	jakýkoliv	Vrací rodiče aktuálního uzlu
../*	jakýkoliv	Vybere všechny sourozence aktuálního uzlu včetně jeho samotného
../*[@*=3]	jakýkoliv	Vybere všechny sourozence aktuálního uzlu včetně jeho samotného, pokud mají libovolný svůj atribut roven 3

4.5 Operátory

Výrazy XPath vždy vrací buď množinu uzlů, řetězec, logickou hodnotu nebo číslo. Pro operace s návratovými hodnotami se v XPath používají operátory obvyklé téměř ve všech programovacích jazycích. Využití nacházejí především v podmínkách, a při použití v 602XML také při nastavování vlastností datových vazeb, kde se provádí řada výpočtů a je třeba vyhodnotit a zpracovat velké množství XPath výrazů. Pořadí vyhodnocení operací je možné klasickým způsobem upravovat závorkami.

Operand	Popis	Příklad	Navracená hodnota
	Spojí dvě množiny uzlů	//domu //prace	Vrátí množinu uzlů se jmény <i>domu</i> a <i>prace</i>
+	Sčítání	8 + 4	12
-	Odčítání	8 - 4	4
*	Násobení	8 * 4	32
div	Dělení	8 div 4	2
=	Rovnost	id = 2	TRUE, pokud id je 2, FALSE jinak
!=	Nerovnost	id != 2	TRUE, pokud id není 2, FALSE jinak
<	Menší než	id < 2	TRUE, pokud id je menší než 2, FALSE jinak
<=	Menší než nebo rovno	id <= 2	TRUE, pokud id je menší nebo rovno 2, FALSE jinak
>	Větší než	id > 2	TRUE, pokud id je větší než 2, FALSE jinak
=>	Větší než nebo rovno	id => 2	TRUE, pokud id je větší nebo rovno 2, FALSE jinak
or	Logické OR	id = 2 or id = 3	TRUE, pokud id je 2 nebo 3, FALSE jinak
and	Logické AND	id < 4 and id > 2	TRUE pokud je id menší než 4 a větší než 2, FALSE jinak
not()	Logický zápor	not(id = 2)	TRUE, pokud id je různé od 2, FALSE jinak
mod	Zbytek po dělení (modulo)	5 mod 2	1

5 XPath – funkce

Specifikace XPath obsahuje celou řadu užitečných funkcí, které lze využít především ke tvorbě podmínek a výrazů pro výpočty. V této kapitole popíšeme jak funkce náležející přímo do standardu specifikace XPath, tak funkce použitelné v aplikacích 602XML při konstrukci XPath výrazů jako rozšíření 602XML. Funkce popsané v této kapitole je možno použít ke konstrukci výrazů ve vlastnostech datových vazeb.

5.1 Popis standardních funkcí XPath

V této kapitole budou popsány XPath funkce, které jsou vždy přítomny v knihovně funkcí implementace XPath, která je využívána pro vyhodnocení výrazů.

Každá z těchto funkcí je popsána tzv. funkčním prototypem, který specifikuje typ návratové hodnoty, jméno funkce a typy argumentů funkce. Pokud je typ argumentu následován otazníkem, znamená to, že použití argumentu je volitelné; v opačném případě je argument pro funkci povinný.

Funkce pro práci s uzly:

Funkce	Popis
<i>number</i> last ()	Vrací počet uzlů v právě zpracovávaném seznamu uzlů
<i>number</i> position ()	Vrací aktuální pozici uzlu, který je právě zpracováván
<i>number</i> count (<i>node-set</i>)	Vrací počet uzlů v argumentu <i>node-set</i>
<i>node-set</i> id (<i>object</i>)	Vrací elementy vybrané na základě jejich unikátního ID
<i>string</i> local-name (<i>node-set</i> ?)	Vrací lokální jméno aktuálního uzlu nebo prvního uzlu v zvolené množině uzlů – bez prefixu jmenného prostoru
<i>string</i> namespace-uri (<i>node-set</i> ?)	Vrací URI jmenného prostoru aktuálního uzlu nebo prvního uzlu ve zvolené množině uzlů
<i>string</i> name (<i>node-set</i> ?)	Vrací kvalifikované jméno aktuálního uzlu nebo prvního uzlu v zvolené množině uzlů

Funkce pro práci s řetězci:

Funkce	Popis
<i>string</i> string (<i>object</i> ?)	Konvertuje zvolený objekt na objekt typu řetězec. Popis konverze různých typů objektů je popsán na těchto stránkách
<i>string</i> concat (<i>string</i> , <i>string</i> , ...)	Vrací řetězec vzniklý zřetězením všech argumentů 057 064 082
<i>boolean</i> starts-with (<i>string</i> , <i>string</i>)	Vrací TRUE, pokud první řetězec začíná druhým řetězcem, FALSE jinak
<i>boolean</i> contains (<i>string</i> , <i>string</i>)	Vrací TRUE, pokud první řetězec obsahuje druhý řetězec, FALSE jinak
<i>string</i> substring-before (<i>string</i> , <i>string</i>)	Vrací podřetězec prvního argumentu, který předchází výskytu druhého argumentu v prvním řetězci, nebo prázdný řetězec
<i>string</i> substring-after (<i>string</i> , <i>string</i>)	Vrací podřetězec prvního argumentu, který následuje po výskytu druhého argumentu v prvním řetězci, nebo prázdný řetězec
<i>string</i> substring (<i>string</i> , <i>number</i> , <i>number</i> ?)	Vrací podřetězec prvního argumentu, začínající na pozici specifikované druhým argumentem, o délce rovné třetímu argumentu. Pokud je třetí argument vynechán, je vrácen podřetězec začínající na pozici dle druhého argumentu až po konec původního řetězce 058
<i>number</i> string-length (<i>string</i> ?)	Vrací počet znaků v řetězci. Pokud je argument vynechán, vrací počet znaků v názvu aktuálního uzlu převedenému na řetězec 089

<i>string</i> normalize-space (<i>string</i> ?)	Vrací řetězec, z něž jsou odstraněny tzv. whitespaces (bílé znaky). Pokud je argument vynechán, funkce se opět vztahuje k aktuálnímu uzlu převedenému na řetězec
<i>string</i> translate (<i>string</i> , <i>string</i> , <i>string</i>)	Vrací první řetězec, který má znaky z druhého argumentu nahrazeny znaky z odpovídajících pozic řetězce v třetím argumentu

Booleovské funkce:

Funkce	Popis
<i>boolean</i> boolean (<i>object</i>)	Konvertuje zvolený objekt na objekt typu boolean. Popis konverze různých typů objektů je popsán na těchto stránkách
<i>boolean</i> not (<i>boolean</i>)	Vrací TRUE pokud je argument funkce roven FALSE, a naopak
<i>boolean</i> true ()	Vrací TRUE
<i>boolean</i> false ()	Vrací FALSE
<i>boolean</i> lang (<i>string</i>)	Vrací TRUE, pokud je jazyk kontextového uzlu (specifikován atributem xml:lang) je shodným s jazykem specifikovaným v argumentu funkce, nebo je jeho podjazykem.

Číselné funkce:

Funkce	Popis
<i>number</i> number (<i>object</i> ?)	Konvertuje zvolený objekt na objekt typu number. Popis konverze různých typů objektů je popsán na těchto stránkách
<i>number</i> sum (<i>node-set</i>)	Vrací součet hodnot množiny uzlů v argumentu. V případě textových hodnot jsou řetězce převedeny na čísla
<i>number</i> floor (<i>number</i>) 058 082	Vrací nejvyšší celé číslo, které je nižší než argument
<i>number</i> ceiling (<i>number</i>) 076	Vrací nejnižší celé číslo, které je větší než argument
<i>number</i> round (<i>number</i>) 076	Vrací nejbližší celé číslo

V konkrétním prostředí (například XSLT) mohou k funkcím popsaným výše být doplněny ještě další funkce. Navíc je možné (jak je tomu také v aplikacích 602XML) přidat k výchozím funkcím i vlastní doprogramované rutiny.

5.2 Rozšiřující funkce jazyka XPath implementované na úrovni MSXML4 parseru

Kromě standardních funkcí jazyka XPath jsou v aplikacích 602XML přístupné další funkce, jež jsou implementovány v XML parseru MSXML4, s nímž 602XML aplikaci pracují. Při volání funkcí XPath rozšíření MSXML4 parser je nutno v rámci názvu funkce vždy uvést předponu „ms:“.

Přehled rozšiřujících funkcí jazyka XPath implementovaných v MSXML4 parseru:

Funkce	Popis
<i>boolean</i> ms:type-is (<i>string</i> URI, <i>string</i> local-name)	Funkce testuje, zda datový typ daného uzlu spadá do určitého jmenného prostoru. Prvním parametrem je identifikátor jmenného prostoru datového typu, druhým parametrem je pak název datového typu.
<i>string</i> ms:type-local-name ([<i>node-set</i>])	Funkce vrací nekvalifikovaný název XSD typu aktuálního uzlu, nebo prvního uzlu z množiny uzlů specifikovaných parametrem, pokud je uveden.
<i>string</i> ms:type-namespace-uri ([<i>node-set</i>])	Funkce vrací identifikátor jmenného prostoru asociovaného s XSD typem aktuálního uzlu, nebo prvního uzlu z množiny uzlů specifikovaných parametrem, pokud je uveden.
<i>boolean</i> ms:xsd-info-available ()	Funkce testuje, zda jsou XSD informace k aktuálnímu uzlu k dispozici.
<i>number</i> ms:string-compare (<i>string</i> x, <i>string</i> y, [<i>string</i> language _], [<i>string</i> options])	Funkce provádí lexikografické porovnání dvou řetězců. Prvními dvěma parametry jsou tak řetězce, jež se mají porovnat. Hodnoty, jež nejsou řetězci, jsou před porovnáním nejprve na řetězce převedeny. Třetí parametr nepovinně specifikuje jazyk použitý pro porovnání řetězců (např. „en-US“, či „fr-CA“ nebo „cs-CZ“). Čtvrtý parametr specifikuje zpracovávání velkých a malých písmen pro porovnání. Výchozím způsobem je zachování citlivosti velikosti písmen s tím, že malá písmena jsou „menší“. Povolené hodnoty: „u“ – přepne zachování citlivosti velikosti písmen tak, že velká písmena jsou „menší“ „i“ – vypne citlivost na velikost písmen Funkce vrací: -1 pokud $x < y$ 0 pokud $x = y$ 1 pokud $x > y$
<i>string</i> ms:utc (<i>string</i>)	Převede datové/časové hodnoty do UTC tvaru (normalizovaná reprezentace, která poté může být např. lexikograficky porovnána). Parametr této funkce je interpretován jako XSD formát pro čas a převeden do následujícího formátu: YYYY-MM-DDTHH:NN:SS.III
<i>string</i> ms:namespace-uri (<i>string</i>)	Funkce vyhodnotí hodnotu URI jmenného prostoru parametru (jež musí být předán jako kvalifikovaný název)
<i>string</i> ms:local-name (<i>string</i>)	Funkce vrací lokální jméno uzlu specifikovaného parametrem (jež musí být předán jako kvalifikovaný název)
<i>number</i> ms:number (<i>string</i>)	Funkce převede řetězec specifikovaný parametrem do formátu XPath čísla.

<i>string</i> ms:format-date(<i>string datetime, string format [,locale]</i>)	Funkce převede standardní XSD tvar datového údaje do řetězce podle předpisu. Prvním parametrem je datumová hodnota ve standardním XSD formátu. Druhým parametrem je řetězec obsahující pravidlo přeformátování vstupní hodnoty. Třetím nepovinným parametrem definuje lokální nastavení pro formátování (např. „EN_us“).
<i>string</i> ms:format-time(<i>string datetime, string format [,locale]</i>)	Funkce převede standardní XSD tvar časového údaje do řetězce podle předpisu. Prvním parametrem je časová hodnota ve standardním XSD formátu. Druhým parametrem je řetězec obsahující pravidlo přeformátování vstupní hodnoty. Třetím nepovinným parametrem definuje lokální nastavení pro formátování (např. „EN_us“).

6 Funkce a aplikační události v rozšíření 602XML

Funkce rozšíření 602XML přinášejí další možnosti, jež obohacují standardní možnosti jazyka XPath, resp. jeho funkcí o další funkčnost využitelnou v elektronických formulářích 602XML. Dělíme je na:

- Funkce rozšíření jazyka XPath modulu XMLcore
 - Funkce rozšíření 602XML pro výpočty
 - Funkce rozšíření 602XML pro formulářové objekty
 - Funkce rozšíření 602XML pro práci s certifikáty
- Aplikační události Form Filleru

Jednotlivé funkce budou postupně popsány v následujících kapitolách.

6.1 Popis funkcí rozšíření 602XML modulu XMLcore pro výpočty

Funkcionalitu, která je nad rámec možností výpočetního modelu XPath jazyka (implementovaného v komponentě MSXML4), je možné integrovat do formuláře pomocí funkcí tzv. 602XML rozšíření. Při používání funkcí 602XML rozšíření XPath je nutno brát v úvahu několik faktů:

- Vyhodnocování funkcí 602XML rozšíření neprobíhá přímo na úrovni XML parseru MSXML4, který 602XML využívá, ale přes základní vrstvu XMLCore, což může vést ke zpomalení funkčnosti.
- Funkce 602XML rozšíření volané v rámci některé z vlastností datové vazby je nutno uzavírat do uvozovek, jinak nebudou fungovat!
- Závorkovací konvence – pokud je potřeba, aby nebyla textová hodnota předaná funkcí rozšíření zpracovávána interním XML parserem, je potřeba tuto hodnotu uzavřít do složených závorek. Je nutno dodržovat zejména pokud předáváme funkci jako parametr řetězec složený z více jednotlivých řetězců (např. výraz `{../d:delitel div ../d:delenec}`).
- Doporučujeme nekombinovat funkce vlastní jazyku XPath, jež jsou přítomny v MSXML4 parseru společně s funkcemi XMLcore. Tam, kde je tomu možno, je doporučeno použít ekvivalentní funkci XMLcore.

Přehled funkcí 602XML rozšíření specifikace XPath použitelných pro výpočty vlastností datových vazeb je v následující tabulce:

Funkce	Popis	Příklady volání
DataTransform (string, string, [string , string])	Funkce volá transformaci, kdy na jejím vstupu je dokument nebo jeho část určená referenčním uzlem a transformovaný dokument je vrácen jako návratová hodnota. První parametr specifikuje XSL stylesheet, který se má použít, druhý parametr pak referenční XPath výraz pro určení uzlu. Poslední dva (nepovinné) parametry specifikují tzv. mode, resp. modeuri parametr XSLT transformace (pokud je v transformaci použit) – více o parametrech mode a modeuri viz http://www.w3.org/TR/xslt#data-model	DataTransform(rc ...) DataTrasform(pocet_dni, ../d:doba_trvani)

Funkce	Popis	Příklady volání
StrCat (<i>string, string, ...</i>) 010 039 081	Funkce spojuje jednotlivé parametry a vytváří jeden řetězec jako návratovou hodnotu.	StrCat(DataXpath(..d:jmeno),{ }, DataXpath(..d:prijmeni))
StrCmp (<i>string, string</i>) 077	Funkce porovná dva řetězce předané jako parametry a vrátí hodnotu TRUE v případě shody, FALSE jinak.	StrCmp(DataXpath(..d:jmeno), DataXpath(..d:skolitel/d:jmeno))
Stop (<i>bool state</i>)	Funkce provede break (ukončí vyhodnocování výrazů, např. Spojených pomocí funkce LastParam) v případě, že parametrem break je logická hodnota TRUE.	Stop(DataXpath(..d:stop = 1))
Round (<i>string XPath, string digits, string direction</i>)	Funkce vrátí zaokrouhlenou vstupní hodnotu. První parametr specifikuje původní hodnotu. Druhý parametr specifikuje počet desetinných míst použitých pro zaokrouhlení. Třetí parametr specifikuje směr zaokrouhlení (up/down).	Round(DataXpath(..d:cislo), 2, {up})
And (<i>string, string</i>)	Logické AND. Vrací TRUE, pokud jsou oba argumenty rovny TRUE, FALSE jinak.	And(DataXpath(..d:bool1), DataXpath(..d:bool2))
Or (<i>string, string</i>)	Logické OR. Vrací TRUE, pokud je alespoň jeden argument roven TRUE, FALSE jinak.	Or(DataXpath(..d:bool1), DataXpath(..d:bool2))
Not (<i>string</i>)	Logický zápor. Vrací TRUE, pokud je argument roven FALSE, a naopak.	Not(DataXpath(..d:bool1))
CallDll (<i>string, string, string, string, ...</i>)	Umožňuje volat metodu z externí DLL knihovny. První argument je cesta k DLL knihovně, druhý je jméno volané metody. Třetí parametr definuje typ volání metody; může nabývat hodnot typu <i>hwnd_ppStr_XpStr</i> , kde X je počet parametrů (tedy např. <i>hwnd_ppStr_2pStr</i> pro 2 parametrou metodu). Dalšími argumenty funkce jsou pak jednotlivé parametry. Návratová hodnota funkce je typu <i>string</i> .	CallDll(c:_sample\test.dll, _RunFunction@12, hwnd_ppStr_pStr, GetFormDataXML())
If (<i>string condition, string trueresult, string falseresult</i>) 010 058 091	Podle vyhodnocení logické hodnoty v prvním argumentu provede příslušnou akci. Druhý parametr specifikuje akci odpovídající hodnotě TRUE, třetí hodnotě FALSE.	If(DataXpath({../d:polozka1}), spravne, chyba)
IsEmpty (<i>string param</i>)	Funkce vrátí TRUE v případě, že vstupní parametr je prázdný, FALSE jinak.	IsEmpty(DataXpath({../d:sest}))
CmdLineParam ()	Funkce vrátí jako <i>string</i> text příkazové řádky použité pro spuštění Form Filleru.	
CurrentUserName ()	Funkce vrátí uživatelské jméno přihlášeného uživatele do systému.	

Funkce	Popis	Příklady volání
CurrentFileName (<i>type</i>)	Funkce vrací název, resp. cestu k otevřenému formuláři. Pomocí nepovinného parametru <i>type</i> lze specifikovat typ informace: 'short' – funkce vrátí pouze jméno souboru 'path' – funkce vrátí pouze cestu k souboru " – funkce vrátí cestu obsahující název souboru	
CurrentDate (<i>format</i>) 008	Funkce vrací aktuální systémové datum počítače. Pokud je zadán parametr pro formát data, je systémové datum přeformátováno do požadovaného tvaru.	CurrentDate(ddd)
CurrentTime (<i>format</i>) 008	Funkce vrací aktuální systémový čas počítače. Pokud je zadán parametr pro formát času, je systémový čas přeformátován do požadovaného tvaru.	CurrentTime(HH:mm)
CurrentTimeUTC (<i>format</i>)	Funkce vrací aktuální UTC čas (Coordinated universal time). Pokud je zadán parametr pro formát času, je hodnota přeformátována do požadovaného tvaru.	CurrentTimeUTC(HH:mm)
FormID ()	Funkce vrací ID otevřeného formuláře.	
FormCreationDate ()	Funkce vrací datum posledního uložení aktuálního formuláře. Toto datum je uloženo uvnitř každého formuláře a je identifikováno jako „Datum vytvoření formuláře“.	
GetListItemData (<i>listbox, dataXPath, atribut, element</i>) 030	Funkce vrací datovou hodnotu vybrané položky z vícesloupcového položkového seznamu. První argument je XPath adresa formulářového prvku, který představuje položkový seznam. Druhý argument je relativní XPath adresa datové hodnoty vybrané položky. Třetí, resp. čtvrtý argument specifikuje atribut, resp. element položkového seznamu, ze kterého má pocházet navrácená hodnota.	GetListItemData(mesic, //d:mesic, @zkratka) GetListItemData(mesic, //d:mesic, fm:zkratka)
GetItemValue (<i>string listDataXPath, string keyValuedataXPath, string value, string resultDataXPath</i>) 095	Funkce zobrazí hodnotu vybrané položky datového číselníku. Výraz má čtyři části. První parametr musí být absolutní XPath na node u "opakovačky", druhý je atribut, podle kterého se selektuje, třetí je DataXPath na pole s listboxem a čtvrtý je atribut, který chci zobrazit.	"GetItemValue(/d:root/d:skupina/d:opak, @value , DataXPath(..d:lisbox) @value)"

Funkce	Popis	Příklady volání
LastParam (<i>param</i> , <i>param</i> , ...) 078	Funkce, která umožňuje sekvenčně vyhodnotit skupinu funkcí zadaných jako parametr. Návrátová hodnota funkce LastParam odpovídá návratové hodnotě poslední volané funkce. Argumenty této funkce jsou jednotlivé funkce v pořadí tak, jak mají být vyhodnoceny.	LastParam(DataTransform(rc ...), DataTrasform(pocet_dni, ../d:dooba_trvani))
GetNumSigns () 036	Funkce vrací počet platných elektronických podpisů v dokumentu	
GetEnvVariable (<i>string</i>)	Funkce vrací textovou hodnotu systémové proměnné specifikované parametrem.	GetEnvVariable(PROGRAMFILES)
GetSignProperty (<i>signIndex</i> , <i>signParam</i> , <i>signOfOID</i>) 025 036 038 039	Funkce vrací konkrétní informace o elektronických podpisech, které jsou k dokumentu připojeny. Prvním parametrem je index podpisu, ze kterého se informace budou extrahovat. Druhým parametrem je pole certifikátu, ze kterého se budou informace extrahovat. Povolené hodnoty tohoto parametru jsou {SignSubject, SignIssuer, SignNotBefore, SignNotAfter, SignTime, SignComment}. Třetím parametrem je OID označení dané položky (Object identifier).	GetSignProperty(3, SignSubject, 2.5.4.3) GetSignProperty(GetFieldSignerIndex(/*[@form602bind="supervizor"], SignSubject, 2.5.4.3)
DataXPath (<i>string dataXPath</i>) 014 029 075 078 088 043	Vyhodnotí datový XPath výraz v argumentu.	DataXPath(..d:ks * ../d:cena)
GetDataStr (<i>string dataXPath</i> , <i>param</i>) 054 055 084	Vrací z formuláře kompletní XML data vyjma datové položky specifikované v atributu. Druhý parametr je nepovinný a pokud je true, tak vrací kanonizované xml. POZOR! Funkce nevrací kanonizované XML, a její výsledek tak může být různý i při vložení identických dat!	GetDataStr(@hash, true)
GetHash (<i>string</i> , <i>method</i>) 054 055 083 084	Vrací kontrolní součet (hash) vzniklý aplikací specifického algoritmu na vstupní řetězec. Prvním parametrem jsou XML data či řetězec, z něhož má být kontrolní součet počítán. Druhým parametrem je metoda použitá k výpočtu kontrolního součtu (MD5 nebo SHA1). POZOR! Nedoporučuje se jednoduché využití funkce GetDataStr pro získání celé datové věty formuláře, neboť výsledkem této funkce je nekanonizový XML string, a při použití stejných dat tak může být její výsledek, a tím potažmo i kontrolní součet jiný!	GetHash(StrCat(DataXPath(../dmeno),{}),DataXPath(../d:prijmeni)), SHA1) GetHash(GetDataStr(@hash), MD5)

Funkce	Popis	Příklady volání
SignLevel (<i>string</i>) 036	Vrací TRUE, pokud aktuální úroveň parciálního podpisu odpovídá parametru. Jinak vrací FALSE.	SignLevel(1) SignLevel(DataXpath(..d:signs))
PartSignLevel (<i>string</i>) 036	Vrací TRUE, pokud aktuální úroveň parciálního podpisu odpovídá parametru, přičemž ignoruje elektronický podpis návrháře formuláře (PUBLISHER), pokud je ve formuláři obsažen. Jinak vrací FALSE.	PartSignLevel(3)
GetFormParam (<i>XPath</i>)	Vrací hodnotu formulářového parametru podle XPath výrazu specifikovaného parametrem.	GetFormParam(//muj_parametr)
GetNumPages ()	Vrátí hodnotu čísla stránky, na niž vede datová vazba, ve které se volá.	
GetFieldValue () 091	Vrátí hodnotu pole.	
SendAppEvent (<i>event, param, ...</i>) 024 025 034 038 073	Funkce provádí tzv. aplikační události. Každá aplikační událost je specifikována prvním parametrem. Následující argumenty jsou pak vlastními parametry každé události. Podrobný popis aplikačních událostí viz následující kapitolu.	SendAppEvent(printpdf)
Position (<i>string relatXPath</i>) 073	Funkce vrátí pozici (index) formulářového prvku specifikovaného relativním XPath výrazem v druhém parametru (tento se vyhodnotí oproti aktuální XPath pozici)	Position(..../d:opak_sekce)

6.2 Popis funkcí rozšíření 602XML modulu XMLcore pro formulářové objekty

Sada rozšíření 602XML modulu XMLcore neobsahuje jen funkce použitelné pro tvorbu výrazů v dialogu s vlastnostmi datových vazeb, ale také funkce použitelné ve formulářové části formuláře. Jedná se např. o funkce volatelné po stisku tlačítka, funkce napojitelné na některé atributy formulářových polí atd. Tyto funkce jsou popsány v následující tabulce.

Tyto funkce tvoří obsah určitých atributů FO části formuláře. V případě tlačítek je lze jednoduchým způsobem nastavovat pomocí dialogu s vlastnostmi tlačítka.

Přehled funkcí modulu XMLcore pro práci s formulářovými objekty je v následující tabulce:

Funkce	Popis	Příklady volání od verze 3.0
GetAttribute (string AttrName, string FormXPath?) 025 038 039	Funkce vrací hodnotu atributu podřízenému elementu, který je specifikován formulářovým XPath výrazem. POZOR! Jedná se pouze o formulářové prvky ze jmenného prostoru FO!	GetAttribute(form602state,*)
SetAttribute (string AttrName, string Value) 091 010	Funkce nastavuje hodnotu atributu, nad nímž je volána na hodnotu určenou druhým parametrem.	SetAttribute(border-style, buttonpushed)
StrCat (string, string, ...) 038 039 054 055 083 084 025 083	Funkce spojuje jednotlivé parametry a vytváří jeden řetězec jako návratovou hodnotu.	StrCat(jmeno: , DataXPath(//d:prijmeni))
CallDll (string dllName , string procName, string procType , string param, ...)	Umožňuje volat metodu z externí DLL knihovny. První argument je cesta k DLL knihovně, druhý je jméno volané metody. Třetí parametr definuje typ volání metody; může nabývat hodnot typu <i>hwnd_ppStr_XpStr</i> , kde X je počet parametrů (tedy např. <i>hwnd_ppStr_2pStr</i> pro 2 parametrou metodu). Dalšími argumenty funkce jsou pak jednotlivé parametry. Návratová hodnota funkce je typu string.	CallDll(c:_sample\test.dll, _RunFunction@12, hwnd_ppStr_pStr, GetFormDataXML())
SaveData (string fileName, string user?, string password?, string proxyuser?, string proxypwd?)	Funkce provede export dat a podle parametru fileName exportovaná data zpracuje. Parametr fileName obsahuje cíl, kam mají být data uložena, popř. odeslána. Může to být buď cesta k souboru, do kterého mají být data uložena, nebo URL adresa v případě, kdy mají být exportovaná data předána serveru pomocí http požadavku. Parametry user a password specifikují autentifikaci uživatele pro přístup ke složce na HTTP serveru. Parametry proxyuser a proxypwd specifikují autentifikaci uživatele pro přístup přes PROXY server.	SaveData(c:\temp.xml)
Save (string fileName?) 085	Funkce uloží formulář do aktuálního souboru. Pomocí nepovinného parametru je možné definovat alternativní soubor pro uložení formuláře. Parametr fileName obsahuje cestu a název souboru pro uložení formuláře.	Save(c:\temp.fo)

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
SaveSigned (<i>string fileName?</i> , <i>bool continueWithoutSign</i>)	Funkce uloží formulář s elektronickým podpisem do aktuálního souboru. Pomocí parametru je možné definovat alternativní soubor pro uložení formuláře. Parametr <i>fileName</i> obsahuje cestu a název souboru pro uložení formuláře. Pokud je druhý parametr roven 'ContinueIfCancelSign', pak při stornování dialogu pro výběr podpisu nedojde k přerušení akce, ale formulář bude uložen bez podpisu.	SaveSigned(c:\temp.fo)
CleanForm ()	Funkce provede vyčištění všech formulářových polí a nastavení jejich výchozích hodnot.	
Recalc ()	Funkce provede přepoččet všech automaticky doplňovaných hodnot ve formuláři.	
CheckData () 079	Funkce provede kontrolu korektnosti vyplnění všech formulářových polí.	
If (<i>string condition</i> , <i>string trueresult</i> , <i>string falseresult</i>)) 010 058 091	Podle vyhodnocení logické hodnoty v prvním argumentu provede příslušnou akci. Druhý parametr specifikuje akci odpovídající hodnotě TRUE, třetí hodnotě FALSE.	If(DataXpath({../d:polozka1}), spravne, chyba)
IsEmpty (<i>string param</i>) 038	Funkce vrací TRUE v případě, že vstupní parametr není prázdný řetězec, v opačném případě vrací FALSE.	IsEmpty(DataXpath({ ../d:sest}))
RefreshElements (<i>string refXPath</i>)	Funkce vyvolá akci grafického přeformátování oblasti dokumentu určené XPath výrazem v parametru funkce.	RefreshElements(//fo:block[6])
SaveAs (<i>filename?</i>) 078	Funkce uloží formulář do souboru a vrací cestu k souboru jako návratovou hodnotu. Parametr <i>filename</i> obsahuje cestu a název souboru.	SaveAs(c:\temp.fo)
SaveAsSigned (<i>filename?</i>)	Funkce uloží formulář s elektronickým podpisem do souboru a vrací cestu k souboru jako návratovou hodnotu. Parametr <i>filename</i> obsahuje cestu a název souboru.	SaveAsSigned(c:\temp.fo)
GetTmpFileName (<i>filename?</i>)	Funkce vrátí cestu k adresáři, kam jsou umísťovány dočasné soubory. V případě, že byl jako parametr funkce předán název souboru, bude vrácena cesta včetně názvu dočasného souboru.	
Is_Changed ()	Funkce vrací logickou hodnotu true v případě, že byl dokument změněn, v opačném případě vrací false.	
StrCmp (<i>string</i> , <i>string</i>) 077	Funkce porovná dva řetězce předané jako parametry a vrací hodnotu TRUE v případě shody, FALSE jinak.	StrCmp(DataXpath(../d:jmenor), DataXpath(../d:skolitel/d:jmeno))
Stop (<i>state</i>)	Funkce provede break (ukončí vyhodnocování výrazů, např. Spojených pomocí funkce LastParam) v případě, že parametrem break je logická hodnota TRUE.	Stop(DataXpath(//d:bool1))
Not (<i>string</i>)	Logický zápor. Vrací TRUE, pokud je argument roven FALSE, a naopak.	Not(DataXpath(../d:bool1)

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
And (string, string)	Logické AND. Vrací TRUE, pokud jsou oba argumenty rovny TRUE, FALSE jinak.	And(DataXpath(..d:bool1), DataXpath(..d:bool2))
Or (string, string)	Logické OR. Vrací TRUE, pokud je alespoň jeden argument roven TRUE, FALSE jinak.	Or(DataXpath(..d:bool1), DataXpath(..d:bool2))
Transform (string stylesheetname, string refxpath, string mode, string modeuri) 020	Funkce volá transformaci, kdy na jejím vstupu je dokument nebo jeho část určená referenčním uzlem a transformovaný dokument je vrácen jako návratová hodnota. První parametr specifikuje XSL stylesheet, který se má použít, druhý parametr pak referenční XPath výraz pro určení uzlu. Poslední dva parametry specifikují tzv. mode, resp. modeuri parametr XSLT transformace (pokud je v transformaci použit) – více o parametrech mode a modeuri viz např. http://www.w3.org/TR/xslt#data-model	Trasform(pocet_dni, .)
GetEnvVariable (string)	Funkci vrací hodnotu systémové proměnné specifikované parametrem.	GetEnvVariable(PROGRAMFILES)
GetSaveFileNameDialog (type?, name?)	Funkce zobrazí systémový dialog pro výběr souboru pro uložení na disk. Návratová hodnota je cesta a název vybraného souboru, nebo prázdný řetězec v případě, že byl dialog stornován. První parametr nepovinně specifikuje typ souboru pro uložení. Druhý parametr specifikuje jméno.	
LoadFormData (file?)	Funkce importuje do formuláře data zvolená pomocí parametru. Parametr specifikuje cestu k souboru s XML daty.	
InsertSection (string XPath, string data) 094	Funkce vloží do formuláře další instanci opakovací sekce. Prvním parametrem je formulářový XPath výraz, který specifikuje pozici opakovací sekce, která má být přidána. Druhý parametr umožňuje automaticky naplnit do sekce data.	InsertSection(//fo:table[5]/fo:table-row[2])
RemoveSection (string Xpath) 094	Funkce odebere z formuláře jednu instanci opakovací sekce. Parametrem je XPath výraz, který určuje pozici sekce.	RemoveSection(//fo:table[5]/fo:table-row[2])
GetOpenFileNameDialog (type?, name?)	Funkce zobrazí systémový dialog pro výběr souboru pro otevření. Návratová hodnota je cesta a název vybraného souboru a nebo prázdný řetězec v případě, že byl dialog stornován. První parametr nepovinně specifikuje typ souboru pro otevření. Druhý parametr nepovinně specifikuje jméno.	GetOpenFileNameDialog(fo)

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
Message-Box (<i>string message, string title, string</i>)	Funkce zobrazí potvrzovací dialog se zprávou. Při potvrzení dialogu je návratová hodnota funkce true, při stornování dialogu false. Prvním parametrem je text zprávy, druhým parametrem je titulek dialogu. Pomocí třetího nepovinného parametru lze specifikovat typ dialogu: 'ok' – zobrazí se dialog jen s tlačítkem OK 'ok-cancel' – zobrazí dialog s tlačítky OK a Cancel 'yes-no' – zobrazí dialog s tlačítky Yes a No	MessageBox(Ahoj, Zpráva, ok)
QueryTextDialog (<i>string title, string message</i>)	Funkce zobrazí dotazovací dialog. Jako návratová hodnota je vrácen string vložený do editačního pole dialogu, popř. prázdná hodnota v případě stornování dialogu. Prvním parametrem je titulek dialogu, druhým parametrem je text zobrazený v editačním poli.	QueryTextDialog(Dotaz, {Chcete opravdu...})
CurrentUserName ()	Funkce vrací uživatelské jméno přihlášeného uživatele do systému.	
CurrentFileName (<i>type</i>)	Funkce vrací název, resp. cestu k otevřenému formuláři. Pomocí nepovinného parametru <i>type</i> lze specifikovat typ informace: 'short' – funkce vrátí pouze jméno souboru 'path' – funkce vrátí pouze cestu k souboru ' ' – funkce vrátí cestu obsahující název souboru	CurrentFileName (<i>type</i>)
CurrentTime (<i>format</i>) 008	Funkce vrací aktuální systémový čas počítače. Pokud je zadán parametr pro formát času, je systémový čas přeformátován do požadovaného tvaru.	CurrentTime(HH:mm)
CurrentTimeUTC (<i>format</i>)	Funkce vrací aktuální UTC čas (Coordinated universal time). Pokud je zadán parametr pro formát času, je systémový čas přeformátován do požadovaného tvaru.	CurrentTimeUTC(HH:mm)
CurrentDate (<i>format</i>) 008	Funkce vrací aktuální systémové datum počítače. Pokud je zadán parametr pro formát data, je systémové datum přeformátováno do požadovaného tvaru.	CurrentDate(ddd)
FormID ()	Funkce vrací ID otevřeného formuláře.	
FormCreationDate ()	Funkce vrací datum posledního uložení aktuálního formuláře. Toto datum je uloženo uvnitř každého formuláře a je identifikováno jako „Datum vytvoření formuláře“.	

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
SectionNo() 007 038 081	Funkce vrací číslo opakovací sekce, na kterou je její volání navázáno.	
RowNo() 007	Funkce vrací číslo řádku tabulky, na kterou je její volání navázáno.	
FormatNumb (<i>string value, string format</i>)	Funkce vrací přeformátovanou hodnotu. Prvním parametrem je původní hodnota. Druhým parametrem je maska, na kterou se má původní hodnota naformátovat.	FormatNumb(DataXpath(//d:vysledek) ,#.00)
IntAdd (<i>int, int (,...)</i>) 007 038	Funkce vrací výsledek celočíselného součtu parametrů.	IntAdd(SectionNo(),2)
IntMult (<i>int, int (,...)</i>)	Funkce vrací výsledek celočíselného násobení parametrů.	IntMult(FormID(),3)
CellNo()	Funkce vrací číslo buňky tabulky, na kterou je její volání vázáno.	
GetFormParam (<i>string XPath</i>)	Vrací hodnotu formulářového parametru podle XPath výrazu specifikovaného parametrem. Formulářové parametry jsou obsahem elementů/atributů, které jsou uzavřeny v elementu dsig:object[3] v XML struktuře formuláře.	GetFormParam(//muj_parametr)
XpathExpr (<i>string XPath, string relatXPath</i>)	Funkce vyhodnotí XPath výraz a výsledek vrátí jako návratovou hodnotu. Výsledkem vyhodnocení může být textová hodnota (hodnota elementu apod.), uzel (node) popř. sada uzlů (nodeset). V případě, že vyhodnocením výrazu je uzel, potažmo sada uzlů, je návratová hodnota funkce textový obsah elementů. Při vyhodnocování výrazu je možné zadat referenční uzel, ke kterému bude zadán XPath vztahován. Prvním parametrem je XPath výraz, který má být vyhodnocen. Druhým parametrem je nepovinná specifikace referenčního uzlu.	XpathExpr({//*['@*='path']})
GetListItemData (<i>string listBox, string dataXPath, string atribut, string element</i>) 030	Funkce vrací datovou hodnotu vybrané položky z vícesloupcového položkového seznamu. První argument je XPath adresa formulářového prvku, který představuje položkový seznam. Druhý argument je relativní XPath adresa datové hodnoty vybrané položky. Třetí, resp. čtvrtý argument specifikuje atribut, resp. element položkového seznamu, ze kterého má pocházet navrácená hodnota.	GetListItemData(mesic, //d:mesic, @zkratka) GetListItemData(mesic, //d:mesic, fm:zkratka)
LastParam (<i>param, param (,...)</i>) 038 039	Funkce umožňuje zřetězení několika akcí za sebe, např. po stisknutí tlačítka. Návratová hodnota je identická s návratovou hodnotou poslední akce.	LastParam(InsertSection(), SaveSigned())
GetNumSigns() 036	Funkce vrací počet platných elektronických podpisů v dokumentu.	

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
GetSignProperty (<i>string signIndex, string signParam, string signOfOID</i>) 036 038 039 025	Funkce vrací konkrétní informace o elektronických podpisech, které jsou k dokumentu připojeny. Prvním parametrem je index podpisu, ze kterého se informace budou extrahovat. Druhým parametrem je pole certifikátu, ze kterého se budou informace extrahovat. Povolené hodnoty tohoto parametru jsou {SignSubject, SignIssuer, SignNotBefore, SignNotAfter, SignTime, SignComment}. Třetím parametrem je OID označení dané položky (Object identifier), resp. formát zobrazení získané hodnoty.	GetSignProperty(3, SignSubject, 2.5.4.3) GetSignProperty(GetFieldSignerIndex(/*[@form602bind="supervizor"], SignSubject, 2.5.4.3)
GetFormDataXML (<i>string sign?</i>)	Funkce vrací XML data z vyplněného formuláře. Data mohou být volitelně podepsána elektronickým podpisem. Nepovinný parametr – v případě, že odpovídá hodnotě „signed“, je po zavolání funkce vyvolán dialog pro elektronický podpis a funkce tak vrátí data XML, která jsou elektronicky podepsána.	
GetFormXML (<i>string sign?</i>)	Funkce vrací XML textový zdroj formuláře jako návratovou hodnotu. Pokud je parametr roven 'signed', pak bude zobrazen dialog pro výběr elektronického podpisu, a XML data budou při uložení podepsána.	
Print ()	Funkce vyvolá dialog pro tisk formuláře.	
GetHash (<i>string, string method</i>) 054 055 083 084	Vrací kontrolní součet XML dat ve formuláři. Prvním parametrem jsou XML data, která mají být použita (většinou výsledek funkce GetDataStr). Druhým parametrem je metoda použitá k výpočtu kontrolního součtu (MD5 nebo SHA1). POZOR! Funkce nevrací kanonizované XML, a její výsledek tak může být různý i při vložení identických dat!	GetHash(GetDataStr(DataXPath(@hash)), MD5)
ActivateField (<i>string formXPath, string formXPathReference?</i>) 003	Funkce způsobí přidělení focusu formulářovému poli specifikovanému parametrem XPath.	ActivateField(/fo:root/fo:page-sequence[1]/fo:flow[1]/fo:block[4]/fo:inline[1])
SetValueToData (<i>string value</i>)	Funkce nastaví datovou hodnotu do datové položky určené parametrem form602bindref, který přísluší formulářovému prvku, na něž je volání této funkce napojeno. Parametr value určuje hodnotu, která bude do dat vložena.	SetValueToData(ano)
GetCurrentDataXPath (<i>string bindXPath?</i>)	Funkce vrací datový XPath aktuálního prvku. Pokud je specifikován parametr, pak vrací relativní datový XPath prvku specifikovaného parametrem vzhledem k aktuálnímu prvku.	

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
SetMultipleDataTo (<i>string dataXPath, string data, string dataXPath, string data,...</i>) 071 075	Funkce nastaví datové hodnoty do příslušných prvků datové věty určeným pomocí datových XPath výrazů. Je možno specifikovat libovolné množství dvojic parametrů. První z každé dvojice parametrů udává název specifikující položku datové vazby, druhý z každé dvojice parametrů udává hodnotu, která má být do položky vložena.	SetMultipleDataTo(a,1,b,2,c,3)
SetListDataTo (<i>string node, string bindXPath</i>) 060	Funkce vloží data z číselníku, který je tvořen položkami datové věty a nad kterým je volána, do datové položky specifikované datovou vazbou. Prvním parametrem je datový XPath výraz specifikující položku číselníku, která má být přenesena do datové vazby. Tento XPath se zadává relativně podle referenčního XPath výrazu pro daný datový číselník (z atributu form602reference). Druhým parametrem je název datové vazby, do níž má být hodnota vložena.	SetListDataTo(d:ciselnik, jmeno_uzivatele)
Switch (<i>string DataXPath, string defvalue, string value, string switchvalue, string value, string switchvalue, ...</i>) 096	Funkce slouží k vytvoření zástupných hodnot vzhledem k možným hodnotám nějaké datové položky. Prvním parametrem specifikuje hodnotu datové položky, na základě které se má vytvořit zástupná hodnota. Druhým parametrem je výchozí zástupná hodnota. Dále následují dvojice parametrů specifikující interpretaci jednotlivých hodnot – první parametr z každé dvojice specifikuje původní hodnotu datové položky a druhá hodnota specifikuje zástupnou hodnotu, jež bude výsledkem funkce Switch, pokud je datová položka rovna předchozí hodnotě.	Switch(DataXPath(.),vychozi moznost,3,c,2,b,1,a)
DataXPath (<i>string XPath</i>) 083	Funkce vrací vyhodnocení výrazu XPath specifikovaného parametrem.	DataXPath(@hash)
SubmitData (<i>string profileName</i>)	Funkce slouží k odeslání dat pomocí odesílacího profilu specifikovaného parametrem.	SubmitData(http_send)
GetFieldSignerIndex (<i>string fieldXPath</i>)	Funkce vrací index parciálního podpisu formulářové části specifikované XPath výrazem specifikujícím daný formulářový prvek resp. prvky.	GetFieldSignerIndex(//*[@form602bind="admin"])
Round (<i>string XPath, string digits, string direction</i>)	Funkce vrací zaokrouhlenou vstupní hodnotu. První parametr specifikuje původní hodnotu. Druhý parametr specifikuje počet desetinných míst použitých pro zaokrouhlení.	Round(DataXPath(..d:cislo), 2)
SendAppEvent (<i>event, param, ...</i>) 022 056 078 003	Funkce provádí tzv. aplikační události. Každá aplikační událost je specifikována prvním parametrem. Následující argumenty jsou pak vlastními parametry každé události. Podrobný popis aplikačních událostí viz následující kapitolu.	SendAppEvent(printpdf)
PageNo ()	Funkce vrací číslo stránky, na které leží element, nad nímž je volána.	

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
XSLCheck (<i>string xsltname, string referenceXPath?</i>)	Funkce provede XSLT transformaci, jejíž jméno je specifikováno prvním parametrem. V případě chyby vrátí funkce XPath výraz, jinak vrátí prázdný string. Zároveň označí červeně pole s chybou. Pro nastavení fokusu na toto pole je nutno volat ji ve spojení s funkcí ActivateField.	XSLCheck(muj_datacheck)
SystemDataChanged ()	Funkce vyvolává přepočítání těch datových položek, které závisí na pseudovazbě „system“.	
TransformDataBranche (<i>string stylesheetname, string bindname1, string bindname2, string decryptCertPath?, string decryptCertPswrd?</i>) 051	Funkce spustí XSLT transformaci specifikovanou prvním parametrem. Tato XSLT transformace je spuštěna nad tou částí datové větvy, jež je obsažena v datové položce odpovídající vazbě specifikované v parametru <i>bindname1</i> . Výsledek transformace je poté vložen do části datové větvy specifikované datovou vazbou zadanou v parametru <i>bindname2</i> . Poslední dva nepovinné parametry specifikují cestu k certifikátu a jeho heslo pro případné rozšifrování vstupních dat.	TransformDataBranche(moje_transformace, opak_sekce_zadatel, rodne_cislo_seznam)
LoadXMLFragment (<i>string FileName, string XPath</i>)	Funkce vrátí část XML dat z datového souboru specifikovaného prvním parametrem z části specifikované XPath výrazem v druhém parametru.	LoadXMLFragment(c:\form.fo, /root/opak_sekce)
CleanFields (<i>string dataXPath?</i>)	Funkce obnoví výchozí hodnoty v podřízených položkách kontextové vazby, nebo v podřízených položkách výsledku datového XPath výrazu specifikovaného parametrem, pokud je zapsán.	CleanFields(d:zadavatel)
CloseApp ()	Funkce uzavře aplikaci Form Filler.	
CmdLineParam ()	Funkce vrátí jako string text příkazové řádky použité pro spuštění Form Filleru.	
SignLevel (<i>string</i>) 036	Vrací TRUE pokud aktuální úroveň parciálního podpisu odpovídá parametru. Jinak vrátí FALSE.	SignLevel(1) SignLevel(DataXPath(..:/d:signs))
ReadBinDataFile AndB64Encode (<i>string file</i>)	Funkce načte soubor specifikovaný parametrem a vrátí jej ve formátu BASE64DATA	ReadBinDataFilterAndB64Encode(c:\form.fo)
SaveBase64DataToFile (<i>string file, string bindata</i>)	Funkce uloží soubor specifikovaný první parametrem, a naplní ho rozkódovanými BASE64DATA daty.	SaveBase64DataToFile(c:\form.fo, ../d:binary)
Position (<i>string relatXPath</i>) 073	Funkce vrátí pozici (index) formulářového prvku specifikovaného relativním XPath výrazem v druhém parametru (tento se vyhodnotí oproti aktuální XPath pozici)	Position(..../d:opak_sekce)
ListItemNo ()	Funkce vrátí pozici aktuální položky ve formulářovém objektu fo:list-block.	
RecheckField (<i>string formXPath?</i>)	Funkce vyvolá přepočítání formulářového prvku, nad kterým je volána, nebo prvku, jenž je specifikován relativním formulářovým XPath výrazem.	RecheckField(..../..)

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
ChoosePrinter()	Funkce vyvolá dialog pro výběr tiskárny a vrací název vybrané tiskárny.	
GetCertDir()	Funkce vrátí název adresáře, jenž je nastaven jako úložiště certifikátů.	
SetCertDir (string dirname)	Funkce nastaví adresář pro úložiště certifikátů podle parametru	SetCertDir(c:\certs)
FileDelete (string filename)	Funkce smaže soubor specifikovaný parametrem.	FileDelete(c:\form.fo)
SignIcon (string signLevel, string activeLevel, string iconSize)	Funkce nastavuje chování pečete pro podepisování v módu opakovacích sekcí. První parametr slouží k přepnutí hodnoty atributu form602state pro změnu chování podepisovací pečete (tlačítko/podepsaná pečeť). Druhým parametrem je úroveň podpisu, ve které se má pečeť poprvé objevit. Třetím parametrem je velikost ikony pečete (16/24/32).	SignIcon(SetAttribute(form602state,GetFieldSignerIndex(..../fo:block[1]/fo:inline[1])),2,24)
RepeSignIcon (string ActiveLevel, string iconSize)	Funkce nastavuje chování pečete pro podepisování v módu opakovacích sekcí. Prvním parametrem je úroveň podpisu, ve které se má pečeť poprvé objevit. Druhým parametrem je velikost ikony pečete (16/24/32).	RepeSignIcon(2,16)
GetPlainText() 098	Funkce převede formátovaný text do prostého textu bez formátování.	GetPlainText(..d:element,1)
ListXpath (string jmeno ciselniku, Xpath) 106	Funkce slouží k procházení rozbalovacího seznamu. První parametr udává jméno číselníku, druhá XPathový výraz. Procházení začíná elementem fm:list.	"ListXpath (jednotlivá_cisla, StrCat({count(/fm:list/fm:listitem[@itemtext='],DataXpath(..d:element3),{' and @value='],DataXpath(..d:substring),{'}}))"
SaveOrgData (param1)	Param1 – cesta na cílový soubor, do kterého se mají data uložit.	Ukládá do souboru původní dokument v originálním tvaru, tedy i s podpisy, pokud je obsahoval, a to i ve formátu PKCS#7. (Původním dokumentem mohou být buď data, ke kterým se pak našel vhodný formulář, nebo formulář.)
Foreach (konstanta, dataXPath, bind name, konstanta)	konstanta – sekvence prováděná v cyklu dataXPath – data, pro která se dělá cyklus bind name – datová vazba, do které se v cyklu zapisuje current index konstanta – sekvence prováděná po úspěšném ukončení cyklu	(!!!nezarazovat do sekvence 'LastParam') Od verze 3.0
ReplaceString (DataXpath, string, string)	DataXPath – určuje string, kde se výměna textu provede string1 – jaký text se má vyměnit string2 – za co se má vyměnit	"ReplaceString(DataXpath(..pozdrav), {ahoj}, {nazdar})" <i>od verze 3.0</i>

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
GetNamedParam (AppAcc Par)	Zjišťuje cookie ohledně přihlášení do Datových schránek, rozpoznává, zda je formulář spuštěn z portálu nebo z lokálu	<i>od verze 3.0</i>
ScanStart ()	Vrací identifikátor naskenovaného dokumentu. Nemá parametry.	<i>od verze 3.0</i>
GetTotPagesOfScannedDocument (param1)	Vrací počet stránek naskenovaného dokumentu param1 = identifikátor naskenovaného dokumentu	<i>od verze 3.0</i>
CreatePDFfromScan (param1, param2, param3, param4)	param1 = identifikátor naskenovaného dokumentu param2 = hash certifikátu, kterým se má podepsat PDF param3 = doložka param4 = GUID	Vrací hash naskenovaného dokumentu – tedy identifikátor konvertovaného dokumentu. <i>od verze 3.0</i>
GetSignedPDFfromScan (param1, param2)	param1 = identifikátor naskenovaného dokumentu param2 = časové razítko, které se má přidat	vrací data PDF dokumentu zakódovaná v B64 <i>od verze 3.0</i>
ScanEnd (param1)	Param1 = identifikátor naskenovaného dokumentu	Ukončuje skenování a uvolňuje resources zadaného skenu <i>od verze 3.0</i>
GetOS ()	Pro zjištění operačního systému. Linuxový binár vrací hodnotu „linux“, windowsový vrací „windows“.	<i>od verze 3.0</i>
Switch (string XPath, string defvalue, string value, string switchvalue, string value, string switchvalue, ...) 096	Funkce slouží k vytvoření zástupných hodnot vzhledem k možným hodnotám nějaké datové položky. První parametrem specifikuje hodnotu datové položky, na základě které se má vytvořit zástupná hodnota. Druhým parametrem je výchozí zástupná hodnota. Dále následují dvojice parametrů specifikující interpretaci jednotlivých hodnot – první parametr z každé dvojice specifikuje původní hodnotu datové položky a druhá hodnota specifikuje zástupnou hodnotu, jež bude výsledkem funkce Switch, pokud je datová položka rovna předchozí hodnotě.	Switch(DataXPath(.),výchozí možnost,3,c,2,b,1,a)
LastParam (param, param (...)) 051 043	Funkce umožňuje zřetězení několika akcí za sebe, např. po stisknutí tlačítka. Návrátová hodnota je identická s návratovou hodnotou poslední akce.	LastParam(InsertSection(), SaveSigned())

Funkce	Popis	Příklady volání <i>od verze 3.0</i>
GetHash(string, method) 054 055 083 084	Vrací kontrolní součet XML dat ve formuláři. Prvním parametrem jsou XML data, která mají být použita (většinou výsledek funkce GetDataStr). Druhým parametrem je metoda použitá k výpočtu kontrolního součtu (MD5 nebo SHA1). POZOR! Funkce nevrací kanonizované XML, a její výsledek tak může být různý i při vložení identických dat!	GetHash(GetDataStr(DataXPath(@hash)), MD5)
SwitchBFToEmbedded(XPath)	Nastaví u přílohového pole příznak, že se příloha nemá detašovat a že se má vložit přímo do formuláře. Má jeden parametr – formulářový XPath na přílohové pole (relativní nebo absolutní).	<i>Od verze 3.17</i>
SwitchBFToDetached(XPath)	Nastaví u přílohového pole příznak, že se má příloha detašovat – defaultní nastavení u pole, které má nastaven plugin pro detašování. Má jeden parametr – formulářový XPath na přílohové pole (relativní nebo absolutní).	<i>Od verze 3.17</i>

6.3 Popis funkcí rozšíření 602XML modulu XMLcore pro práci s certifikáty

Poslední skupinou funkcí rozšíření 602XML modulu XMLcore jsou funkce pro práci s certifikáty. Tyto funkce jsou použitelné pro nastavení vlastností odesílacích profilů, primárně filtraci zobrazení nabídky certifikátů při podepisování dat/formuláře.

Přehled funkcí XMLcore pro práci s certifikáty je v následující tabulce:

Funkce	Popis	Příklady volání
IssuerStr (<i>string OID?</i>) 077	Funkce vrací vydavatele certifikátu, resp. jednu jeho část podle specifikovaného OID identifikátoru	IssuerStr(2.5.4.3)
SubjectStr (<i>string OID?</i>)	Funkce vrací předmět certifikátu, resp. jednu jeho část podle specifikovaného OID identifikátoru	SubjectStr(2.5.4.3)
StrCat (<i>string, string, ...</i>) 038 039 054 055 083 084	Funkce spojuje jednotlivé parametry a vytváří jeden řetězec jako návratovou hodnotu.	StrCat(jmeno: , DataXpath(//d:prijmeni))
StrCmp (<i>string, string</i>) 077	Funkce porovná dva řetězce předané jako parametry a vrací hodnotu TRUE v případě shody, FALSE jinak.	StrCmp(DataXpath(..d:jmeno), DataXpath(..d:skolitel/d:jmeno))
ContainsSubstr (<i>string full, string substr</i>)	Funkce vrací TRUE, pokud první parametr obsahuje jako podřetězec druhý parametr, FALSE jinak.	ContainsSubstr(..d:jmeno_full, ..d:prijmeni)
Stop (<i>bool state</i>)	Funkce provede break (ukončí vyhodnocování výrazů, např. Spojených pomocí funkce LastParam) v případě, že parametrem break je logická hodnota TRUE.	Stop(DataXpath(..d:stop = 1))
Round (<i>string XPath, string digits, string direction</i>)	Funkce vrací zaokrouhlenou vstupní hodnotu. První parametr specifikuje původní hodnotu. Druhý parametr specifikuje počet desetinných míst použitých pro zaokrouhlení. Třetí parametr specifikuje směr zaokrouhlení (up/down)	Round(DataXpath(..d:cislo), 2, {up})
And (<i>string, string</i>)	Logické AND. Vrací TRUE, pokud jsou oba argumenty rovny TRUE, FALSE jinak.	And(DataXpath(..d:bool1), DataXpath(..d:bool2))
Or (<i>string, string</i>)	Logické OR. Vrací TRUE, pokud je alespoň jeden argument roven TRUE, FALSE jinak.	Or(DataXpath(..d:bool1), DataXpath(..d:bool2))
Not (<i>string</i>)	Logický zápor. Vrací TRUE, pokud je argument roven FALSE, a naopak.	Not(DataXpath(..d:bool1))
If (<i>string, string, string</i>) 038	Podle vyhodnocení logické hodnoty v prvním argumentu provede příslušnou akci. Druhý parametr specifikuje akci odpovídající hodnotě TRUE, třetí hodnotě FALSE.	If(DataXpath({..d:polozka1}), spravne, chyba)
IsEmpty (<i>string</i>) 038	Funkce vrací TRUE v případě, že vstupní parametr je prázdný, FALSE jinak.	IsEmpty(DataXpath({ ..d:sest}))

Funkce	Popis	Příklady volání
GetTmpFileName (<i>filename?</i>)	Funkce vrátí cestu k adresáři, kam jsou umístovány dočasné soubory. V případě, že byl jako parametr funkce předán název souboru, bude vrácena cesta včetně názvu dočasného souboru.	
GetEnvVariable (<i>string</i>)	Funkci vrací hodnotu systémové proměnné specifikované parametrem.	GetEnvVariable(PROGRAMFILES)
CurrentUserName ()	Funkce vrací uživatelské jméno přihlášeného uživatele do systému.	
CurrentTime (<i>format</i>) 008	Funkce vrací aktuální systémový čas počítače. Pokud je zadán parametr pro formát času, je systémové čas přeformátován do požadovaného tvaru.	CurrentTime(HH:mm)
CurrentTimeUTC (<i>format</i>)	Funkce vrací aktuální UTC čas (Coordinated universal time). Pokud je zadán parametr pro formát času, je systémové čas přeformátován do požadovaného tvaru.	CurrentTimeUTC(HH:mm)
CurrentDate (<i>format</i>) 008	Funkce vrací aktuální systémové datum počítače. Pokud je zadán parametr pro formát data, je systémové datum přeformátováno do požadovaného tvaru.	CurrentDate(ddd)
IntAdd (<i>int, int (,...)</i>) 007 038	Funkce vrací výsledek celočíselného součtu parametrů.	IntAdd(SectionNo(),2)
IntMult (<i>int, int (,...)</i>)	Funkce vrací výsledek celočíselného násobení parametrů.	IntMult(FormID(),3)

6.4

6.5 Popis aplikačních událostí Form Filler

Aplikační události představují speciální skupiny funkcí rozšíření aplikací 602XML. Představují sadu událostí (event), které lze předat aplikaci Form Filler, která je poté zpracovává. Výhodou je možnost stornování akcí volaných pomocí aplikačních událostí, zejména v případě jejich zřetězení za sebou pomocí funkce XMLcore *LastParam*.

Přehled aplikačních událostí rozšíření 602XML je v následující tabulce:

Výraz	Parametry	Akce
SendAppEvent (<i>printpdf</i> , [<i>typedataXPath</i> , <i>ruledataXPath</i> , <i>filedataXPath</i>]) 093	typedataXPath – datový <i>typedataXPath</i> – datový typ tisku: '0' – <i>etOptPrintAll</i> (výchozí) '1' – <i>etOptPrintNoFields</i> (tisk prázdného formuláře bez obsahu formulářových polí) '2' – <i>etOptPrintOnlyFields</i> (tisk pouze obsahu formulářových polí) ruledataXPath – datový XPath – pokud je roven hodnotě '1' dojde k použití pravidla pro tisk (výchozí nastavení je NEPOUŽÍVAT) filedataXPath – datový XPath – pro přímou specifikaci výstupního PDF souboru	Tisknout formulář do PDF
SendAppEvent (<i>print</i> , [<i>typedataXPath</i> , <i>ruledataXPath</i>]) 093	typedataXPath – datový XPath – určuje typ tisku '0' – <i>etOptPrintAll</i> (výchozí) '1' – <i>etOptPrintNoFields</i> (tisk prázdného formuláře bez obsahu formulářových polí) '2' – <i>etOptPrintOnlyFields</i> (tisk pouze obsahu formulářových polí) ruledataXPath – datový XPath – pokud je roven hodnotě '1' dojde k použití pravidla pro tisk (výchozí nastavení je NEPOUŽÍVAT)	Tisknout formulář
SendAppEvent (<i>checkdata</i> , [<i>infoincomplete</i>],[<i>stopincomplete</i>]) 079 043	infoincomplete – nepovinný parametr, pokud je specifikován, dojde při špatném vyplnění formuláře k dotazu, zda se má pokračovat v provádění dalších akcí stopincomplete – při použití parametru se hlásí pouze chyba (nehlásí se, že je formulář OK)	Zkontrolovat správnost vyplnění formuláře
SendAppEvent (<i>recalc</i>) 045		Přepočítat hodnoty ve formuláři

Výraz	Parametry	Akce
SendAppEvent (<i>submit</i> , <i>submit_name</i>) 013 037 051 052 089	submit_name – název odesílacího profilu specifikovaný ve formuláři	Odeslat data
SendAppEvent (<i>submitpart</i> , <i>submit_name</i> , <i>sendDataXPath</i> , <i>reponseDataXPath</i>)	submitName – název odesílacího profilu specifikovaného ve formuláři sendDataXPath – datový XPath, datová položka pro specifikaci části datové věty k odeslání dat; pokud je odesílací profil typu SaveData, pak se jedná o jméno souboru pro uložení. responseDataXPath – datový XPath specifikující část datové věty pro uložení odpovědi; pokud je odesílací profil typu SaveData, pak se jedná o nepovinný parametr umožňující specifikaci dané opakovací sekce, z níž se mají vyexportovat data	Odeslat pouze část datové věty
SendAppEvent (<i>addsect</i> , <i>formXPath</i> , <i>number</i>) 024 025 034 038 073	formXPath – formulářový XPath do opakovací sekce number – počet sekcí pro přidání	Přidat opakovací sekci
SendAppEvent (<i>delsect</i> , <i>formXPath</i>)	formXPath – formulářový XPath vedoucí do opakovací sekce	Odstranit opakovací sekci
SendAppEvent (<i>fillsect</i> , <i>formXPath</i> , [<i>fileDataXPath</i> , <i>dlgFormats</i> , <i>xsltName</i>]) 073	formXPath – formulářový XPath pro určení opakovací sekce k vyplnění fileDataXPath – datový XPath pro získání cesty na soubor s daty pro předvyplnění sekce dlgFormats – konstanta pro nastavení dialogu pro výběr datového souboru (volá se, je-li předchozí parametr prázdný) – povolené hodnoty jsou 'form', 'xml', 'fo', 'zfo', 'p7m' (ve výchozím nastavení jsou to všechny typy) xsltName – jméno transformace, která se má provést před načtením dat do sekce	Naplnit opakovací sekci daty

Výraz	Parametry	Akce
SendAppEvent (loadformdata, [fileDataXPath, destDataXPath, dlgFormats, xsltName])	fileDataXPath – datový XPath pro získání cesty na soubor s daty pro načtení destDataXPath – datový XPath – určuje datovou položku pro vložení dat, pokud se jedná o načtení pouze části datové větý dlgFormats – konstanta pro nastavení dialogu pro výběr datového souboru (volá se, je-li parametr <i>fileDataXPath</i> prázdný) – povolené hodnoty jsou 'form', 'xml', 'fo', 'zfo', 'p7m' (ve výchozím nastavení jsou to všechny typy) xsltName – jméno transformace, která se má provést před načtením dat do sekce	Načíst data do formuláře
SendAppEvent (openform, [fileDataXPath]) 024 038 039	fileDataXPath – datový XPath pro získání cesty souboru k otevření; pokud není tento parametr specifikován, načte se znovu aktuální formulář	Otevřít soubor s formulářem
SendAppEvent (saveform, [format, fileDataXPath, dlg]) 085	format – konstanta pro formát uloženého souboru. Výchozím formátem je 'fo'. fileDataXPath – datový XPath – výsledkem je název souboru k uložení dlg – určuje pravidla pro zobrazení dialogu pro uložení souboru ▪ 'dlg' – zobrazí SaveAs dialog ▪ 'nodlg' – nezobrazí SaveAs dialog ▪ prázdný řetězec – zobrazí SaveAs dialog, je-li parametr <i>fileDataXPath</i> prázdný ▪ jiný řetězec – Form Filler se pokusí vyhodnotit nastavenou hodnotu pomocí vyhodnocení XPath (SaveAs dialog tedy lze nastavovat i dynamicky)	Uložit formulář

Výraz	Parametry	Akce
SendAppEvent (<i>saveformsign</i> , [<i>format</i> , <i>fileDataXPath</i> , <i>dlg</i>]) 024 036 038 039	format – konstanta pro formát uloženého souboru. Výchozím formátem je 'fo'. fileDataXPath – datový XPath – výsledkem je název souboru k uložení dlg – určuje pravidla pro zobrazení dialogu pro uložení souboru ▪ 'dlg' – zobrazí SaveAs dialog ▪ 'nodlg' – nezobrazí SaveAs dialog ▪ prázdný řetězec – zobrazí SaveAs dialog, je-li parametr <i>fileDataXPath</i> prázdný ▪ jiný řetězec – Form Filler se pokusí vyhodnotit nastavenou hodnotu pomocí vyhodnocení XPath (lze tedy SaveAs dialog nastavovat i dynamicky)	Uložit formulář s elektronickým podpisem
SendAppEvent (<i>closeapp</i> , [<i>closeType</i>]) 036 051	closeType – konstanta; je-li rovna hodnotě '0', pak nedojde k dotazu na uložení aktuálního souboru	Zavřít aplikaci
SendAppEvent (<i>openbindata</i> , <i>bindName</i> [, <i>formXPath</i>]) 009 047	bindName – název datové vazby s binární přílohou formXPath – nepovinný parametr specifikující formulářový XPath tlačítka s přílohou POZOR! Pro otevření souboru z přílohy je nutné, aby v datové větě formuláře byly přítomny i prvky obsahující název a příponu souboru přílohy!	Otevřít soubor z přílohy
SendAppEvent (<i>setdatato</i> , <i>destFormXPath</i> , <i>string</i>) 043 090	destFormXPath – formulářový XPath pro určení cílového prvku string – data, která se do pole nastaví POZOR! Tato aplikační událost není vhodná pro nastavování hodnoty datové vazby, je přítomna zejména pro zachování kompatibility s formuláři vytvořenými v datovém modelu verze 1.0-1.3. Pro nastavení hodnoty prvku datové věty použijte raději funkci <i>setdatavalue</i> , resp. <i>setevaldata</i> !	Nastavit hodnotu do formulářového prvku
SendAppEvent (<i>setevaldata</i> , <i>bindName</i> , <i>destDataXPath</i> , <i>XPathexpr</i>)	bindName – jméno datové vazby pro vložení dat destDataXPath – ABSOLUTNÍ datový Xpath položky pro vložení dat (cíl) XPathexpr – data, která se do pole nastaví, přičemž je zde možno provést vyhodnocení výrazu	Nastavit hodnotu do prvku datové věty na základě vyhodnocení XPath výrazu

Výraz	Parametry	Akce
SendAppEvent (<i>setdatavalue</i> , <i>bindName</i> , <i>destDataXPath</i> , <i>string</i>) 013 024 043 045 055 090	bindName – jméno datové vazby pro vložení dat destDataXPath – ABSOLUTNÍ datový XPath položky pro vložení dat (cíl) string – data, která se do pole nastaví (konstanta)	Nastavit hodnotu (konstantu) do prvku datové věty
SendAppEvent (<i>messagebox</i> , <i>string</i> , [<i>dlgType</i>]) 059 079	string – zobrazený text dlgType – počet tlačítek v dialogu '1' – jedno tlačítko 'OK' (výchozí hodnota, jedná se o klasický dialog MessageBox) '2' – dvě tlačítka 'ANO/NE' (dotazový AskYesNo dialog); stisknutí tlačítka 'NE' vyvolá Storno	Zobrazit dialog se zprávou
SendAppEvent (<i>gotofield</i> , <i>formXPath</i>) 043	formXPath – formulářový XPath určující formulářový prvek, kam se posune fokus	Přejít na formulářový prvek
SendAppEvent (<i>startfiller</i> , <i>fileDataXPath</i> , <i>xmldataDataXPath</i>) 081	fileDataXPath – datový XPath výraz specifikující formulář, který má být v nové instanci aplikaci načten xmldataDataXPath – datový XPath výraz specifikující soubor s XML daty, která se mají naimportovat do formuláře	Otevřít novou instanci Form Filler
SendAppEvent (<i>formtohtml</i>)		Provést tisk formuláře jako HTML stránky
SendAppEvent (<i>printparam</i> , [<i>printerName</i> , <i>FromPage</i> , <i>ToPage</i> , <i>Copies</i> , <i>dataXPath</i>])	printerName – nepovinný parametr, specifikuje výběr tiskárny; v případě, že není zadán, použije se výchozí tiskárna FromPage – nepovinný parametr, specifikuje počáteční stranu pro tisk ToPage – nepovinný parametr, specifikuje poslední stranu pro tisk Copies – nepovinný parametr, specifikuje počet vytisknutých kopií dataXPath – pokud je roven hodnotě 'enabledlg', bude před tiskem zobrazen dialog pro výběr tiskárny	Provést přímý tisk formuláře (bez nutnosti výběru tiskárny)

Výraz	Parametry	Akce
SendAppEvent (<i>openfiller</i> , <i>fileDataXPath</i> , <i>xmldataDataXPath</i> , <i>[readonly]</i> , <i>[windowType]</i>) 078	fileDataXPath – datový XPath výraz specifikující formulář, který má být v nové instanci aplikaci načten xmldataDataXPath – datový XPath výraz specifikující soubor s XML daty, která se mají nainportovat do formuláře readonly – nepovinný parametr; pokud je roven konstantě 'readonly', pak není zobrazený formulář editovatelný, konstanta windowType – určení velikosti okna 'max' : maximalizované modální okno 'parent' : okno přesně překrývá Form Filler 'child' : okno o něco menší než Form Filler 'min' : minimalizované modální okno	Otevřít novou instanci Form Filler v modálním okně.
SendAppEvent (<i>loaddata</i> , <i>submit_name</i>) 015 016 041	submit_name – název odesílacího profilu pro načtení dat specifikovaný ve formuláři	Načíst data do formuláře pomocí daného profilu
SendAppEvent (<i>signclick</i> , <i>constant</i> , <i>string</i> , <i>size</i>) 038 039	constant – konstanta, jedná se o řetězec GetAttribute(form602state) jež je vyhodnocen pomocí XMLcore string – specifikuje sekvenci akcí prováděných, pokud je předchozí parametr roven hodnotě 'none' size – udává velikost použité ikony (16/24/32)	Slouží k podpoře elementu pečeť pro parciální podepisování formuláře
SendAppEvent (<i>csvtoxml</i> , <i>fileDataXPath1</i> , <i>fileDataXPath2</i>)	fileDataXPath1 – datový XPath výraz – cesta ke vstupnímu CSV souboru fileDataXPath2 – datový XPath výraz – cesta k výstupnímu XML souboru	Převést vstupní CSV soubor do formátu XML

Výraz	Parametry	Akce
SendAppEvent (<i>openfillerfrombuffer, foxslbuffer, xmlbuffer, [readonly], [windowType]</i>)	foxslbuffer – buffer s FO souborem, jež má být v nové instanci aplikace Form Filler otevřen xmlbuffer – buffer s XML daty pro naplnění FO formuláře readonly – nepovinný parametr; pokud je roven konstantě 'readonly', pak není zobrazený formulář editovatelný windowType – určení velikosti okna 'max' : maximalizované modální okno 'parent' : okno přesně překrývá Form Filler 'child' : okno o něco menší než Form Filler 'min' : minimalizované modální okno	Otevřít novou instanci Form Filler pouze s částí formuláře danou vstupními soubory Tato aplikační událost se hodí zejména v případě, kdy je zobrazený FO soubor výsledkem nějaké transformace.
SendAppEvent (<i>printpreview</i>) 093		Zobrazit náhled tisku s formulářem
SendAppEvent (<i>deletefile, fileDataXPath</i>) 078	fileDataXPath – datový XPath výraz specifikující soubor, který má být smazán	Smazat zadaný formulář
SendAppEvent (<i>savebindata, bindname, formXPath, [fileDataXPath, expDataXPath], dlg?</i>)	bindname – název vazby obsahující BASE64 data formXPath – formulářový XPath prvku obsahujícího přílohu fileDataXPath – datový XPath výraz, nepovinně specifikuje soubor, do kterého mají být data zapsána expDataXPath – datový XPath výraz, nepovinně specifikuje příponu souboru, do kterého mají být data vloženy. dlg – zapnutí nebo vypnutí dialogu pro uložení formuláře	Naplnit automaticky datový soubor binárními daty 3.0.40
SendAppEvent (<i>openlink, linkdataXPath</i>) 022	linkdataXPath – datový XPath výraz specifikující URL nebo lokální adresu cíle	Přejít na adresu odkazu
SendAppEvent (<i>viewdata, relevance</i>) 093	relevance – nepovinný parametr, pokud je roven nule, budou zobrazena i nerelevantní data	Zobrazit XML data

Výraz	Parametry	Akce
SendAppEvent (<i>savebase64datatofile, binDataXPath, fileDataXPath, extDataXPath</i>) 078	binDataXPath – datový XPath specifikující položku obsahující binární data fileDataXPath – nepovinný parametr specifikující jméno cílového souboru. Pokud není uveden, otevře se SaveAs dialog. extDataXPath – nepovinný parametr umožňující nastavení povolených typů souborů pro uložení pomocí SaveAs dialogu.	Uložit binární data uložené v položce datové větvy do souboru. Tato aplikační událost se hodí zejména pro uložení binárního souboru v případě, že není ve formuláři napojen jako obsah pole pro vložení přílohy.
SendAppEvent (<i>updatelist, listNameDataXPath</i>)	listNameDataXPath – datový XPath výraz nebo statický řetězec specifikující název číselníku, který se má aktualizovat	Aktualizovat obsah číselníku SendAppEvent(updatelist,{'ciselnik'})
SendAppEvent (<i>mailmergeedit, roFieldsListDataXPath</i>)	roFieldsListDataXPath – datový XPath výraz nebo statický řetězec specifikující, které položky datového zdroje pro mailmerge mají být uzamčeny pro editaci (tedy pouze pro čtení). Jednotlivé datové položky datového zdroje vypisujeme pomocí jejich pořadí v položce a oddělujeme čárkou nebo svislým oddělovačem. Můžeme zadávat i rozsahy položek. Příklad: chceme, aby byly zamčeny první, druhá a pátá hodnota datové položky. Do parametru tedy musíme zapsat {'1-2,5'} nebo {'1 2 5'}	Editovat datový zdroj mailmerge
SendAppEvent (<i>printallbindata</i>) 092		Vytisknout binární přílohy.
SendAppEvent (<i>stop, dataXPath</i>) 107	dataXPath – je-li 'true', přeruší se vykonávání sekvence událostí.	Přerušování vykonávání sekvence událostí.
SendAppEvent (<i>savevisualization, dataXPath</i>)	dataXPath – umístění výstupní vizualizace. Není-li parametr zadán, objeví se 'save as' dialog. Příklad: SendAppEvent(savevisualization,{'C:\TEMP\vis.fo'})	Uložení vizualizace. (Pozn.: od verze 2.60)
SendAppEvent (<i>processaction, string</i>)	String – sekvence akcí prováděných přes SDK XmlCore.	Spouštění sekvencí přes SDK XmlCore.

Výraz	Parametry	Akce
SendAppEvent (fileexists, dataXPath) 111	dataXPath – jméno testovaného souboru	Testování existence souboru (funguje pro lokální soubory). Použít lze např. Při loadování dat ze souboru, pokud soubor existuje. LastParam(SendAppEvent(fileexists,/d:root/d:datafile[1]), SendAppEvent(loaddata,data_z_tempu))
SendAppEvent (csvtoxmllist, dataXPath, dataXPath)	dataXPath – výsledek = file s csv dataXPath – výsledek = output xmlfile Příklad: SendAppEvent (csvtoxmllist, /d:root/d:aa[1],/d:root/d:bb[1]) constant – Separators (oddělovače záznamu) – default {;} constant – Kvalifikátor (ohraničení řetězce) – default {"	Načte csv do xml vhodného pro načtení do číselníku. (Aktualizace číselníku z csv souboru.) <i>Od verze 3.0.37</i>
SendAppEvent (opendocpanel , dataXPath, dataXPath ,dataXPath, dataXPath)	dataXPath 1 = file s dokumentem dataXPath 2 = EditForm -> je-li 'true', form se otevře pro editaci dataXPath 3 = AutoDelete -> je-li 'true', soubor se automaticky smaže dataXPath 4 = minimální výška panelu v procentech nebo bodech	Aplikační událost pro otevření – obsluhu DocPanelu (v prohlížeči) <i>Od verze 2.63.10</i>
SendAppEvent (cleandocpanel)		Vyprázdní docPanel <i>od verze 2.63.10</i>
SendAppEvent (hidedocpanel)		Schová DocPanel <i>od verze 2.63.10</i>
SendAppEvent (ReadBinDataFileAnd B64Encode)		vložení formuláře do dat

Výraz	Parametry	Akce
SendAppEvent (read,dataXPath)	DataXPath = text, který se má přečíst	V „mluvících“ formulářích. Text, který se má říkat, není ve formuláři, ale v aplikační události – zapnutí čtení, výsledek se přečte přes JAWS/SAPI <i>od verze 2.61.8</i> LastParam(SendAppEvent(readrun,{true})), SendAppEvent(read,{This is the worked hours form. Pick month from the list. start by double press down arrow key. to confirm the value press Tab key})))
SendAppEvent (readrun,dataXPath)	DataXPath = true nebo false	Zapne nebo vypne čtení u „mluvících“ formulářů. <i>od verze 2.61.8</i>
SendAppEvent (bindatatodocpanel, bind name, formXPath, dataXPath, dataXPath)	Bind name = pole s binární přílohou formXPath = pole s binární přílohou dataXPath = výsledek – edit form => Je-li true, pak se form otevře pro editaci dataXPath = minimální výška panelu v procentech nebo bodech	Jednoduché otevření binární přílohy do DocPanelu. <i>Od verze 2.64.2</i>
SendAppEvent (printbindata,bindname,formXPath)	bindname fieldu s binární přílohou formXPath fieldu s binární přílohou	Vytiskne přílohu.
SendAppEvent (addbindata,param1,param2,param3)	Param1 = dataXPathLoginID - ID loginu (identifikace loginu) Param2 = dataXPathLoginShow - podmínka pro zobrazení dialogu login (např. pro ISDS se dialog nezobrazuje, pokud se odesílá z portálu); dialog se nezobrazí, je-li výsledek vyhodnocení 'false' nebo prázdná hodnota Param3 = dataXPathLoginType – typ login dialogu (prozatím implementován pouze jeden -> nyní není třeba zadávat)	Hodnoty získané pomocí dialogu se nastaví do parametru aplikace. Formulář tyto hodnoty může získat pomocí funkce xmlcore 'GetNamedParam(param_name)', kde param_name se sestaví z ID loginu tak, že se přidá klíč pro požadovanou hodnotu, např.: mojeID-user mojeID-password mojeID-usecert <i>verze 3.0.37</i>
SendAppEvent (cleanapplogin,param1)	param1 = dataXPath loginID	Vymazání autentizačních údajů <i>verze 3.0.37</i>

Výraz	Parametry	Akce
SendAppEvent (applogin,param1,param2,param3)	Param1 = dataXPathLoginID – ID loginu param2 = dataXPathLoginShow – podmínka pro zobrazení loginu (pro ISDS nezobrazovat z portálu) param3 = dataXPathLoginType – typ login dialogu. ▪ prázdná hodnota – výchozí volba. Pak dialog nabízí pouze volbu přihlášení certifikátem. ▪ <i>certname</i> – dialog nabízí volbu přihlášení certifikátem a zároveň jméno uživatele a heslo. ▪ <i>certorname</i> – dialog nabízí volbu přihlášení certifikátem a jméno uživatele a heslo, ale jméno a heslo nelze zadat zároveň s certifikátem	Přihlášení do aplikace. verze 3.0.37

7 Použití XPath v aplikacích 602XML

XPath výrazy (jak obecně definované normou, tak přístupné díky rozšíření 602XML) je možno používat v aplikacích 602XML k mnoha úkonům a na mnoha místech. Patří mezi ně:

- Vlastnosti datových vazeb nastavované ve Form Designeru – ve většině případů (kromě pole **Výpočet**) je předpokládán **logický výraz**, tj. takový, který vrací hodnotu TRUE nebo FALSE, podle ní je daná vlastnost aktivní či nikoliv. V poli **Výpočet** se uvádí **obecný XPath výraz**, který bude vyhodnocen a hodnota následně přenesena do pole. Výpočetní model provádí operace nad datovou větou formuláře, tzn. operandem logického i obecného XPath výrazu může být libovolná část datové věty formuláře. Pokud v poli **Výpočet** bude vložen logický XPath výraz, tj. vracející hodnotu TRUE nebo FALSE, bude příslušné pole nabývat právě jedné z uvedených hodnot.
- Napojování akcí a metod, které využívají XPath výrazy, na formulářová tlačítka.
- Nastavení hodnot některých atributů formulářových polí
- Výběr formátovacích elementů grafické podoby formuláře pomocí zapsání XPath cesty na spodní lištu aplikace Form Designer
- Výběr položek datové věty pomocí zapsání absolutní XPath cesty na lištu pod panelem s datovou větou formuláře
- Filtrace zobrazení certifikátů v nastavování pravidel pro odesílání vyplněných dat