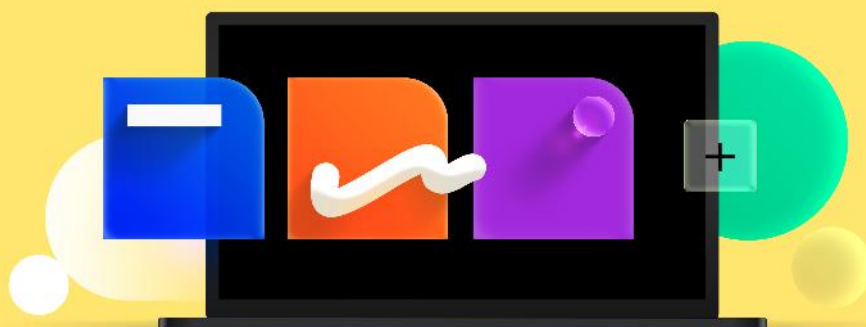




FormFlow

Příručka návrháře formulářů

Aktualizováno: 02.12.2020

Obsah

1	Úvod	4
1.1	Software602 Form Designer	4
1.2	Software602 Form Filler	4
1.3	FormApps Server	4
1.4	FormFlow	4
2	Princip práce s formulářem	5
2.1	Návrh formuláře	5
2.2	Instalace formuláře do FormFlow	5
2.3	Stažení formuláře uživatelem	5
2.4	Odeslání vyplněného formuláře do FormFlow	5
2.5	Postup zpracování formuláře ve FormFlow	5
3	Obsah datové věty FormFlow	6
3.1	Data přihlášeného uživatele	6
3.2	Informace o podepisujících	7
3.3	Informace o formuláři	7
3.4	Nastavení lhůt pro vyřízení formuláře	8
3.5	ASiC formuláře	8
4	URL pro odeslání formuláře do FormFlow	9
4.1	Odesílací URL	9
4.2	Odeslání pomocí tlačítka ve formuláři	10
5	Koloběh XML formulářů	15
5.1	Obecně	15
5.2	Datová věta	16
5.3	Co čte FormFlow z XML dat	19
6	Úpravy formuláře pro koloběh dokumentů	20
6.1	Informace o podepisujícím uživateli	20
6.2	Datový číselník	20
6.3	Zobrazení podpisu	23
6.4	Vložení a podepsání dat na základě relevance	23
6.5	Přepínací sekce pro podpisy	24
6.6	Přepínací sekce pro uzavření schváleného nebo zamítnutého formuláře	24
6.7	Přepínací sekce postoupení předvyplněného formuláře prvnímu podepisujícímu	25
6.8	Oznamování členství ve skupinách	25
6.9	Editace (oprava) vyplněného uzavřeného formuláře	25
6.10	Konečný příjemce pro schválený formulář	26
6.11	Konečný příjemce pro zamítnutý formulář	26
7	Možnosti ladění funkce formuláře ve FormFlow	27
7.1	Formulář	27
7.2	Software602 Form Filler	27
7.3	FormFlow	28

8	Archiv	29
8.1	Úvod	29
8.2	Zapnutí archivu	29
8.3	Cílová složka pro archivaci formuláře v archivu	29
8.4	Archivace dokumentů v archivu	29
8.5	Archivace dokumentů v archivu – LTV položky	30
8.6	Archivace dokumentů v archivu – LTA položky	30
8.7	Podpisová kniha	31
8.8	Spisové znaky v archivu	31
8.9	Datum a čas archivace	31
8.10	Mazání archivovaných formulářů, kterým prošla skartační lhůta	32
8.11	Verze	32
8.12	Odkaz na dokument uložený ve formuláři	32
9	Převod původního typu formulářů na ASiC formulář	33
9.1	Úprava datové věty	33
9.2	Nastavení formuláře	33
9.3	Úprava odesílacích skriptů	34
9.4	Zobrazení historie	34
9.5	Nahrazení původního formuláře	34
10	Apendix A – Odkazy	35
10.1	Odkazy na šablonu formuláře	35
10.2	Univerzální odkaz na formulář (rozpracovaný/uzavřený)	35
10.3	URL pro zjištění informací o souboru v SFLF	35

Tato příručka je určena pro návrháře 602XML formulářů spolupracujících s programem FormFlow a zabývá se zejména návrhem formulářů s podepisovací cestou. Pro návrh formuláře se používá aplikace Software602 Form Designer. Pro další text předpokládáme alespoň základní znalost aplikací Software602 Form Designer, Software602 Form Filler, FormApps Server a FormFlow.

1.1 Software602 Form Designer

Aplikace Software602 Form Designer je určena pro vytváření a úpravy formulářů ve WYSIWYG prostředí. Umožňuje připojení datového zdroje se strukturou XML dat či návrh od ruky, komfortní formátování formuláře, vkládání formulářových prvků a opakovacích sekcí, připojování číselníků a volbu jejich vlastností i vkládání XSL transformací. Návrhář rovněž nastavuje zabezpečení formuláře a určuje pravidla pro podání vyplněného formuláře pomocí aplikace Software602 Form Filler nebo s pomocí FormApps Serveru v internetovém prohlížeči.

1.2 Software602 Form Filler

Aplikace Software602 Form Filler se nainstaluje na stanici uživatele a slouží pro vyplnění formuláře a jeho odeslání do FormFlow.

1.3 FormApps Server

FormApps Server je nástroj pro zpřístupnění formulářových aplikací uživatelům jakéhokoliv zařízení vybaveného internetovým prohlížečem (počítače, tablety, chytré telefony). Pomocí pluginů nebo mobilních aplikací lze umožnit využití elektronických podpisových certifikátů pro autorizaci dokumentů.

1.4 FormFlow

FormFlow je formulářový server využívající technologie 602XML pro zajištění rychlého a snadného sběru dat a koloběhu formulářů. Je ideálním nástrojem pro centrální správu formulářů, jejich následné uzavření, archivaci či uložení rozpracovaných formulářů s případnou informací o průběhu schvalovacích procesů.

2 Princip práce s formulářem

Zde uvedme základní postup práce s formulářem v prostředí 602XML a FormFlow:

2.1 Návrh formuláře

Návrhář navrhne v aplikaci Software602 Form Designer vzhled formuláře, který přiváže vazbami k datové větě. Datová věta bude obsahovat elementy a atributy pro data vyplňovaná do formuláře a také elementy a atributy nutné pro komunikaci s FormFlow. Na závěr Software602 Form Designer formulář publikuje, tedy především data uloží do souboru s koncovkou FO nebo ZFO ve formátu XML-FO a formulář elektronicky podepíše.

2.2 Instalace formuláře do FormFlow

Administrátor nainstaluje formulář do FormFlow (formulář se zapíše do tabulek serveru, u formuláře pro koloběh se definují pravidla podepisovací cesty). Jakmile je šablona formuláře instalována, může ji administrátor uvolnit k použití – oprávnění uživatelé jej vidí ve složce, mohou si ho stáhnout a pracovat s ním.

2.3 Stažení formuláře uživatelem

Formuláře, které spravuje FormFlow, jsou jeho uživatelům přístupné podle systému přístupových práv nastavených administrátorem.

Uživatelé si formulář vyberou a stáhnou pomocí FormFlow, který před vlastním stažením do formuláře přidá svou datovou větu. Ta obsahuje informace o uživateli, který formulář stahuje, URL adresu, kam se bude formulář po vyplnění odesílat, a řadu dalších hodnot. Pokud se jedná o formulář pro koloběh, pak přidá i data o podepisujících.

Obsah formuláře se pak uživateli otevře přímo v prohlížeči, případně pomocí aplikace Software602 Form Filler.

2.4 Odeslání vyplněného formuláře do FormFlow

Uživatel formulář vyplní a poté jej odešle zpět do FormFlow na jednu z předvyplněných URL (`url_release`, `url_forward` atd., viz níže). Tam je formulář uložen pro další zpracování uživateli, případně se data uloží do tabulek a formulář uzavře a archivuje.

2.5 Postup zpracování formuláře ve FormFlow

Zpracování vyplněného formuláře vráceného do FormFlow může proběhnout dvěma různými způsoby:

1. Formulář je určen jen **pro sběr dat** (nemá žádné podepisovací kroky). Pak se po vyplnění uživatelem a odeslání data z formuláře uloží ve FormFlow do databáze (u anonymního uživatele může být vyžadováno, aby uložení nejprve odsouhlasit arbitři).
2. Pokud se jedná o formulář určený **ke koloběhu** (s podepisovacími kroky), pak se data obvykle ukládají do databáze při uzavření – až když formulář projde úspěšně celým schvalovacím procesem. Není to však podmínkou; data lze uložit podle nastavení i jinak, třeba hned po prvním vyplnění.

3 Obsah datové věty FormFlow

V této kapitole si ukážeme, jak navrhovat formuláře pro použití s programem FormFlow. Nejprve si vyjmenujeme, které údaje FormFlow předvyplňuje do formuláře před jeho stažením k uživateli, tj. pokud tyto elementy nalezne v datové větě, předvyplní je hodnotami.

3.1 Data přihlášeného uživatele

FormFlow předvyplní data uživatele, který formulář vyplnil nebo nyní vyplní. Tyto elementy jsou přímým potomkem elementu root:

<code>gw_salutation</code>	Oslovení
<code>gw_pre_title</code>	Titul před jménem
<code>gw_name</code>	Jméno
<code>gw_middle_name</code>	Druhé jméno
<code>gw_surname</code>	Příjmení
<code>gw_birth_name</code>	Rodné jméno
<code>gw_post_title</code>	Titul za jménem
<code>gw_fullname</code>	= „surname name pre_title (job)”
<code>gw_id_user</code>	ID uživatele (definuje systém)
<code>gw_602_form_identity</code>	ID z tabulky USERS ve FormFlow (definuje systém)
<code>gw_uid</code>	ID uživatele z FormFlow (nastavuje administrátor)
<code>gw_birth_date</code>	Datum narození
<code>gw_id_card_number</code>	Číslo OP
<code>gw_pin</code>	Rodné číslo
<code>gw_email</code>	E-mail
<code>gw_phone_number</code>	Telefon
<code>gw_mobile_number</code>	Mobilní telefon
<code>gw_fax_number</code>	Fax
<code>gw_web</code>	Webová adresa
<code>gw_job</code>	Funkce, zařazení (např. „Vedoucí IT“)
<code>gw_company</code>	Jméno firmy
<code>gw_department</code>	Odbor, oddělení (např. „Odbor vnitřních záležitostí“)
<code>gw_section</code>	Sekce (další úroveň pod oddělením)
<code>gw_role</code>	Role uživatele (pouze libovolný text, nesouvisí s rolí u skupin!)
<code>gw_note</code>	Poznámka s libovolnou informací o uživateli
<code>gw_street</code>	Ulice
<code>gw_street_no</code>	Číslo popisné nebo orientační
<code>gw_orientation_no</code>	Číslo orientační
<code>gw_city</code>	Obec
<code>gw_city_district</code>	Část obce
<code>gw_zip_code</code>	PSČ
<code>gw_state</code>	Stát
<code>gw_country</code>	Země
<code>gw_message</code>	Text zprávy od FormFlow nebo předchozího podepisujícího

3.2 Informace o podepisujících

Bližší vysvětlení a seznam odpovídajících atributů je v následující kapitole.

<code>gw_signX</code>	Do atributů toho elementu vloží FormFlow informace o předpokládaných podepisujících
<code>gw_fwdX</code>	Do atributů toho elementu vloží FormFlow informace o příštím příjemci formuláře
<code>gw_grp_fwdX</code>	Informace umožňující zřízení číselníku pro výběr příštího příjemce formuláře
<code>gw_current_user</code>	Informace o přihlášeném uživateli

3.3 Informace o formuláři

Bližší vysvětlení a seznam odpovídajících atributů je v následující kapitole.

<code>gw_sn</code>	Sériové číslo vyplňovaného formuláře generované FormFlow dle zadanych pravidel při instalaci formuláře (dle nastavení ve FormFlow se přiděluje buďto při prvním stažení formuláře nebo po jeho definitivním uložení).
<code>gw_form_name</code>	Změna názvu formuláře s vyplněnými daty – platí pro formuláře uložené v sekci „ <i>Rozpracované</i> “ a „ <i>Uzavřené</i> “. Nový název se též fulltextově zaindexuje a lze tak podle něj vyplněný formulář vyhledat.
<code>gw_form_code</code>	Kód formuláře
<code>gw_id_ufrm</code>	Vnitřní ID formuláře generuje FormFlow pro jednoznačnou identifikaci formuláře v rámci celého FormFlow
<code>gw_document_id</code>	Obsahuje identifikátor anonymem vyplněného formuláře. Používá se pro jeho další vyhledání úředníkem.
<code>gw_status</code>	Průběh schvalovacího procesu formuláře (0 – v procesu, 1 – schválen, 2 – zamítnut, 3 – v prvním kroku při vyplňování anonymem nebo autorem)
<code>gw_form_status_code</code>	Obsahuje kód aktuálního stavu formuláře (stavy definované administrátorem a nastavené uživatelem nebo FormFlow)
<code>gw_602_form_identity</code>	Identifikace daného vyplňovaného formuláře, kterou vytváří FormFlow a používá ji k detekci změn formuláře, které mohly nastat mezi odesláním formuláře do Form Filleru a příjmem XML dat zpět.
<code>gw_archive</code>	Příznak o uzavření formuláře
<code>gw_numsigns</code>	Aktuální počet uživatelských podpisů ve formuláři
<code>gw_notify</code>	Příznak zajišťující posílání informačních emailů autorovi formuláře o každém kroku
<code>gw_accept</code>	Pokud je nastaven na 1, FormFlow vyhodnotí jako schválení schvalovacího formuláře ještě dříve, než jej podepíše poslední schvalující.
<code>gw_refuse</code>	Pokud je nastaven na 1, FormFlow vyhodnotí jako zamítnutí schvalovacího formuláře.

3.4 Nastavení lhůt pro vyřízení formuláře

Ve formuláři je možné změnit nastavení avizování lhůty k vyřízení tohoto formuláře. Prázdná hodnota klíče znamená, že se hodnota nemá měnit, text NULL znamená, že se v databázi nastaví hodnota NULL. Časová jednotka údaje jsou implicitně „dny“, druhá možnost je „hodiny“. V názvu klíčů je stále „days“ kvůli kompatibilitě se staršími verzemi FormFlow, které možnost „hodiny“ nenabízely. Tyto klíče FormFlow pouze čte, nepředvyplňuje je.

<code>gw_processing_max_days</code>	Maximální počet dnů k vyřízení formuláře
<code>gw_reminder_days</code>	Termín upozornění
<code>gw_reminder_frequency_days</code>	Interval odesílání upozornění
<code>gw_processing_time_unit</code>	„1“=dny, „2“=hodiny (od FormFlow verze 2.5.6)
<code>gw_reminder_info_days</code>	Termín avíza
<code>gw_reminder_info_frequency_days</code>	Interval odesílání avíza

3.5 ASiC formuláře

ASiC formuláře neukládají do databáze celý formulář, ale pouze datovou větu s informací, v jaké šabloně se má tato zobrazit. Tyto formuláře následně umožňují detailnější nastavení workflow formuláře a obsahují několik dalších elementů navíc. Jedná se o elementy:

```
<b:gw_form_signatures step="" substep="" code_form="" name_form="" id_form=""
id_fmve="" id_layout="" relative_url_layout=""
xmlns:b="http://www.software602.cz/forms/history">
<b:history>
<b:step relative_url="" step="" substep="" id_user="" login_user="" name=""
surname="" pre_title="" job="" department="" fullname="" signtime="">
<b:signininfo>
<b:signtime/>
<dsig:KeyInfo xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
<dsig:X509Data>
<dsig:X509Certificate/>
</dsig:X509Data>
</dsig:KeyInfo>
</b:signininfo>
</b:step>
</b:history>
</b:gw_form_signatures>
```

kteří obsahují informace o stávajícím kroku i historii formuláře.

4 URL pro odeslání formuláře do FormFlow

Formulář je třeba z aplikace Software602 Form Filler do FormFlow odeslat na některou z URL specifikovaných FormFlow v attributech `URL_XXX`.

4.1 Odesílací URL

Odesílací URL jsou součástí atributů kořenového elementu:

```
<root url_send="..." url_release="...">
```

Tyto URL se obvykle používají v odesílacím profilu formuláře. Použitá URL částečně definuje další postup zpracování formuláře FormFlow.

<code>url_send</code>	Uloží částečně vyplněný XML formulář do FormFlow, později lze formulář stáhnout a doplnit. Takto uložený formulář se objeví ve složce „ <i>Rozpracované formuláře</i> “. XML data zatím nejsou uložena do databázových tabulek. Odesílá se formulář bez podpisu. V případě ASiC formulářů se posílají XML data či One Step ASiCE.
<code>url_release</code>	XML data z formuláře jsou definitivně uložena do databázových tabulek a ve webovém rozhraní se vyplněný formulář objeví ve složce „ <i>Uzavřené formuláře</i> “. Odesíláme jen XML data a zaškrťovací pole „ <i>Odeslat relevantní data</i> “ nesmí být zaškrtnuté. Neboli odesíláme i nerelevantní data, aby se vše z formuláře uložilo do databázových tabulek serveru. V případě ASiC formulářů se posílají XML data či One Step ASiCE.
<code>url_forward</code>	Postoupení formuláře dalšímu podepisujícímu k podpisu (viz následující kapitolu). V závislosti na situaci se posílá podepsaný nebo nepodepsaný formulář. V případě ASiC formulářů se posílají XML data či One Step ASiCE. Dále je u ASiC formulářů třeba nastavit parametr <code>forward</code> na hodnotu „ <code>next_step</code> “ nebo „ <code>next_substep</code> “.
<code>url_back</code>	Vrácení formuláře předchozímu podepisujícímu k přepracování (viz následující kapitolu). Odesílá se formulář bez podpisu, FormFlow naopak předchozí podpis odstraní. ASiC formuláře jsou vráceny do přechodícího kroku, ale nedojde k odstranění záznamu z historie formuláře.
<code>url_base</code>	Pouze základní část URL, používá se při volání uživatelských pluginů FormFlow.

Anonymní uživatel nemá nastaven atribut `url_send`, protože nemůže ukládat rozpracované formuláře. Nemá také nastavený atribut `url_back`, protože může být pouze prvním podepisujícím. Pokud odešle formulář na `url_release`, může formulář v závislosti na konfiguraci před uzavřením čekat na schválení arbitrem.

4.2 Odeslání pomocí tlačítka ve formuláři

Odeslání formuláře nebo vyplněných dat do FormFlow se obvykle provede zmáčknutím tlačítka ve formuláři, které vyvolá aplikační událost programu Form Filler (`SendAppEvent`). Pomocí parametrů ve zvoleném odesílacím profilu formuláře dojde k odeslání formuláře nebo dat do FormFlow pomocí protokolu http.

4.2.1 Profil

V odesílacím profilu formuláře se URL obvykle definuje pomocí výrazu `$(Eval())`, kde je argumentem buď přímo atribut předvyplněný FormFlow:

Příklad: `$(Eval(//@url_forward))`

nebo předvyplněný atribut rozšířený pomocí funkce `concat` (viz dokument o XPATH výrazech, funkcích a aplikačních událostech) o další řetězec:

Příklad: `$(Eval(concat(//@url_send, '&to_tables=true')))`

Výjimku z tohoto postupu tvoří formulář připravený pro anonymního uživatele šířený jinými prostředky než stahováním z FormFlow, kdy se do URL запиše konkrétní URL zkopírováním z odpovídající řádky ve vlastnostech již zavedeného formuláře.

Obsahem připojovaných řetězců jsou obvykle parametry uvedené v následujících kapitolách.

4.2.2 JavaScript

Odeslání formuláře lze zajistit také přes JavaScript, který dává návrháři větší svobodu při návrhu. JS pro odeslání XML dat na adresu `url_send` může vypadat následovně:

```
function Submit_http_send()
{
    var doc = GetDocument();
    var res = false;
    var datapar = new DataParam();
    datapar.flags = GFD_INCLUDINGIRRELEVANT;
    datapar.requiredFormat = VarFXML;
    var signpar = new SignParam();
    signpar.flags = GFD_NOSIGNDATA;
    var data = doc.GetFormdata(datapar, signpar);
    if (data && data.type != VarEmpty) {
        var sendpar = new SendParam();
        sendpar.url = doc.GetElement().DataXPath("/d:root/@url_send");
        sendpar.flags = SHTTP_FIXURL | SHTTP_WEB_SERVERSENDS | SHTTP_NODLG;
        sendpar.logname = "http_send";
        var response = doc.SendData(data, sendpar);
        if (response && response.code == SR_OK) {
            var responsedata = response.data;
            if (responsedata.format != VarFundef) {
                //Sending OK response processing
                res = response.ShowResponse();
                doc.CloseApp(CLOSE_IGNORECHANGES);
            }
        }
        else {
            if (response && response.code != SR_CANCELED)
                response.ShowResponse();
        }
    }
    return res;
}
```

V případě, že je třeba data před odesláním podepsat, stačí nastavit hodnotu `signpar.flags` na „GFD_SIGNDATA“. Formát odesílaných dat lze nastavit v `datapar.requiredFormat` následující hodnoty:

VarFZfo	Komprimovaný formulář ve formátu ZFO
VarFXML	Datová věta ve formátu XML
VarF602ASiCE	Data ve formátu One Step ASiCE. V tomto případě je dále potřeba nastavit hodnotu <code>datapar.workflowStep</code> . Toho lze docílit následovně:

```
datapar.requiredFormat = VarF602ASiCE;  
var step = ele.XPath("//d:root/b:gw_form_signatures[1]/@step");  
var substep = ele.XPath("//d:root/b:gw_form_signatures[1]/@substep");  
datapar.workflowStep = step + "_" + substep;
```

4.2.3 Předčasné zpracování dat do tabulek

V odesílacím profilu se za URL přidá parametr `to_tables`.

Např.: `$(Eval(concat("//@url_send, '&to_tables=true')))`

(použít lze pouze v souvislosti s `url_send`)

Parametr `to_tables` způsobí, že se po přijetí dat do FormFlow provede uložení hodnot do uživatelských SQL tabulek.

4.2.4 Předčasné zpracování dat do uzavřených

V odesílacím profilu se za URL přidá parametr `to_archive`.

Např.: `$(Eval(concat("//@url_forward, '&to_archive=true')))`

(je třeba napřed zajistit, aby data již byla v uživatelských SQL tabulkách, viz výše. Použít lze pouze v souvislosti s `url_forward`)

4.2.5 Volba typu odpovědi FormFlow po přijetí dat

Funkce: Po odeslání vyplněného formuláře na URL doplněnou o volání `answer=<param>` vrátí FormFlow odpověď v požadovaném tvaru.

Obecně může mít proměnná `answer` hodnoty:

=info	implicitní, FormFlow vrátí textovou zprávu do okna Form Filleru.
=xml	FormFlow vrátí celou datovou větu, Form Filler ji načte a data vloží do zobrazeného formuláře – načtení vrácených XML dat do formuláře musí být nastaveno v odesílacím profilu.
=form	FormFlow vrátí do Form Filleru kompletní formulář, což má význam především při odesílání podepsaného formuláře, protože je tím umožněno uživateli zobrazit jím odeslaný již podepsaný formulář k dalšímu zpracování, např. odeslání emailem, tisku atd.

Proměnná `answer` se týká jen odesílacích profilů `url_release` a `url_forward` u postoupení předvyplněného formuláře prvnímu podepisujícímu.

Příklad použití v URL: `$(Eval(concat("//@url_release, '&answer=xml')))`

4.2.6 Posílání a postupování formuláře v koloběhu

Obecně platí: v prvním kroku je další předvyplnitel nebo první podepisující uveden v `gw_fwd1`, podepisující pro druhý krok je v `gw_fwd2`. Formulář může ponechat tyto elementy beze změny, tedy tak, jak je vyplnil FormFlow, nebo může jejich hodnotu měnit podle nabídnutých hodnot v `gw_grp_fwdX` nebo do nich může zapsat ID jiného existujícího uživatele.

- 1) **Základní koloběh** s předvyplňujícími a podepisujícími uvedenými v pravidlech jednotlivých kroků.

Odesílá se na URL: `$(Eval(//@url_forward))`

Když se v prvním kroku pošle na `url_forward` celý podepsaný formulář, pošle se uživateli uvedenému v `gw_fwd2`. Obsah `gw_fwd2` se může ve formuláři měnit podle uživatelů uvedených v použitém pravidlu pro krok 2 (tedy ve struktuře/číselníku `gw_grp_fwd2`).

Když se v prvním kroku pošle na `url_forward` celý nepodepsaný formulář, pošle se podle pravidel na `gw_fwd1`. To znamená, že (bez zásahu formuláře do hodnoty v `gw_fwd1`) předvyplňující posílá formulář prvnímu podepisujícímu, resp. první podepisující vrací nepodepsaný formulář předvyplňujícímu.

Pokud se odešle formulář jinému uživateli, pro kterého tedy FormFlow podle pravidel nenajde novou hodnotu pro `gw_fwd1`, nedovolí FormFlow takovému uživateli formulář stáhnout.

Z toho plyne, že i pro uživatele, u kterého při návrhu koloběhu předpokládáme, že formulář podepíše, musíme založit pravidlo, aby např. mohl formulář postoupit bez podpisu např. sám sobě, jinak mu FormFlow nedovolí formulář stáhnout k vyplnění.

Postoupit tedy lze uživatelům uvedeným v pravidlech tohoto kroku. Výjimku tvoří situace, kde se formulář do daného kroku dostal přeskočením podepisovacích kroků, pak musí být příjemce postupovaného formuláře uveden mezi příjemci v přeskakovacích pravidlech.

- 2) **Koloběh s forward=true** (od verze 2.5.3)

Při tomto typu postoupení celého nepodepsaného formuláře dojde v prvním kroku k běžnému postoupení jinému uživateli podle proměnné `gw_fwd1`. Příjemce musí být uveden v pravidlech prvního kroku. Při druhém postoupení v prvním kroku tímto způsobem tedy nedojde k automatickému návratu autorovi.

Odesílá se typicky na URL sestavenou takto:

`$(Eval(concat(//@url_forward, '&forward=true')))`

V dalších krocích nemá parametr `forward=true` žádný význam, provede se postoupení stejným způsobem jako bez tohoto parametru.

Pozn.: První uživatel, který stáhne formulář, nemusí mít vyplněné `gw_fwd2`, tedy nenajde se žádný příjemce podle pravidel pro druhý krok. Kdyby pak odeslal podepsaný formulář do 2. kroku, ohlásí se chyba. Podstatné je, že se mu podaří formulář stáhnout k vyplnění.

- 3) **Opakované postupování podepsaného formuláře stále v prvním kroku**

Od verze 2.5.5 přibyla možnost postupování formuláře v prvním kroku s podepisováním. Cílem tohoto postupu je umožnit podepisování formuláře předem neznámým počtem podepisujících nebo velkým počtem podepisujících, kdy není možné/reálné definovat podpisové kroky standardním způsobem.

Typy odeslání formuláře na URL předvyplněnou v elementu `url_forward`:

bez parametru	s podpisem	přechod do dalšího kroku (podle <code>gw_fwd2</code>)
bez parametru	bez podpisu	postoupení mezi předvyplnitelem a podepisujícím v 1. kroku, druhé použití je zpět k předvyplňujícímu
forward=true	bez podpisu	běžné postoupení jinému uživateli (podle <code>gw_fwd1</code>)
forward=sign	s podpisem	postoupení (podle <code>gw_fwd1</code>), stále v 1. kroku

<code>forward=removesign</code>	bez podpisu	odejme podpis a postoupí (podle <code>gw_fwd1</code>), stále v 1. kroku
<code>forward=next_step</code>		ASiC formulář bude předán do dalšího kroku bez ohledu na to, zda je podepsán či nikoli. Formulář bude předán uživateli dle hodnoty elementu <code>gw_fwd</code> (aktuální step+1)
<code>forward=next_substep</code>		ASiC formulář bude předán do dalšího horizontálního kroku bez ohledu na to, zda je podepsán či nikoli. Formulář bude předán uživateli dle hodnoty elementu <code>gw_fwd</code> (aktuální step)
<code>forward=back_substep</code>		ACiC formulář se vrátí do předcházejícího horizontálního kroku bez ohledu na to, zda je podepsán či nikoli. Formulář bude předán uživateli, který odesílal formulář z předchozího horizontálního kroku.

U formulářů původního typu se vždy odesílá „kompletní formulář včetně vyplněných dat“. U ASiC formulářů se posílají XML data, nebo One Step ASiCE. ASiC formuláře se dají odeslat s podpisem i bez podpisu.

V rámci prvního kroku se formulář postoupí uživateli, jehož ID je v elementu `gw_fwd1`, do druhého kroku uživateli, jehož ID je v elementu `gw_fwd2` atd.

Při `forward=removesign` lze použít element `gw_message` (nebo jiný nakonfigurovaný) pro předání zprávy následujícímu podepisujícímu (podobně jako při vracení předchozímu podepisujícímu prostřednictvím `url_back`).

Upozorníme, že uživatelské podpisy vložené do formuláře v rámci postupování formuláře v prvním kroku se NEZAPOČÍTÁVAJÍ do počtů podpisů v rámci standardních podepisovacích kroků (`gw_numsigs`, `sign_level` atd.)

4.2.7 Použití přeskakovacích pravidel v koloběhu formuláře

Přeskočit podepisovací krok lze až s podepsaným formulářem, tj. ve všech podepisovacích krocích kromě prvního. Po podpisu odešle uživatel (podepisující) z aplikace Software602 Form Filler podepsaný XSL-FO formulář na URL `url_forward`, do něhož doplní URL proměnné `rule_set` s názvem přeskakovacího pravidla a `jump_to` s nominálním číslem podepisovacího kroku, do kterého se má přeskočit. Např.

```
$ (Eval (concat (//@url_forward, '&jump_to=4&rule_set=jump_2_4')))
```

přeskočí do čtvrtého kroku (`jump_to=4`) a použije přeskakovací pravidlo se jménem „jump_2_4“, podle něhož určí uživatele, jemuž má formulář doručit.

Přeskakování podepisovacích kroků je implementováno až ve FormFlow od verze 2.5.3.

Od FormFlow verze 2.5.6 je možné nahradit proměnnou `rule_set` proměnnou `id_user`, jejíž hodnota je ID uživatele, jemuž se má formulář doručit. Stále však platí, že i pro takto určeného uživatele musí existovat přeskakovací pravidlo. Pokud by v URL byly současně proměnné `rule_set` a `id_user`, má přednost `id_user`.

U ASiC formulářů je třeba do url přidat ještě proměnnou `forward=next_step`, jinak bude formulář pouze předán v rámci kroku.

Bližší popis přeskakování podepisovacích kroků najdete v nápovědě k FormFlow, viz www.602.cz.

4.2.8 Tlačítko a aplikační událost

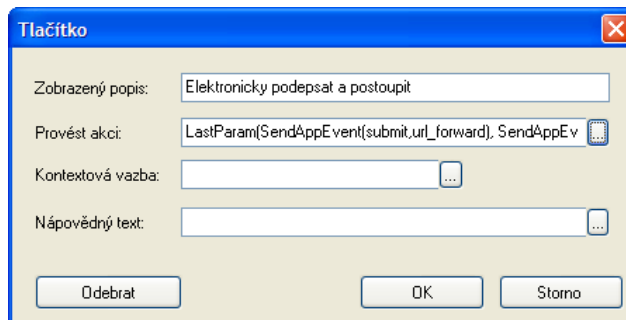
Tlačítko pro odeslání může obsahovat jednoduché volání akce programu Form Filler:

Příklad: `SendAppEvent(submit,profil_forward)`

nebo skupiny akcí:

Příklad: po odeslání do FormFlow zavřít aplikaci Form Filler:

`LastParam(SendAppEvent(submit,profil_forward),SendAppEvent(closeapp))`



Tlačítko

Zobrazený popis: Elektronicky podepsat a postoupit

Provést akci: LastParam(SendAppEvent(submit,url_forward), SendAppEv

Kontextová vazba: ...

Nápovědný text: ...

Odebrat OK Storno

5 Koloběh XML formulářů

5.1 Obecně

Koloběh formulářů lze rozdělit dle praxe na dvě úlohy:

Předem stanovená trasa schvalování (podepisování) formuláře – u formuláře je předem pevně stanoveno, kdo jej podepisuje v druhém, třetím atd. kroku.

Množina uživatelů podepisujících formulář je nastavena v pravidlech na serveru a uživatelé během podepisování formuláře tuto cestu nemohou měnit.

Trasa s volbou dalšího podepisujícího – v každém kroku podepisování může být na serveru nastavena množina uživatelů, takže podepisující uživatel si může vybrat z datového číselníku, komu formulář postoupí k následujícímu podepisovacímu kroku.

Formulář nabývá během schvalovacího procesu následujících stavů:

1. Formulář je ve schvalování, ještě nemá všechny podpisy.
2. Formulář byl zamítnut (libovolný účastník jej může během schvalovacího procesu zamítnout).
3. Formulář je schválený (podepsal jej poslední podepisující).

Formuláře se během schvalování mohou vyskytovat v následujících složkách:

- **Moje k vyřízení** – v této složce se zobrazují formuláře, které má uživatel právě podepsat nebo jinak zpracovat. Současně zde vidí i své rozpracované a dočasně uložené formuláře. Nadřízení (vedoucí skupin) zde vidí pouze formuláře, které mají podepsat, nevidí formuláře k podpisu svých podřízených.
- **V procesu** – v této složce se uživatelům zobrazují všechny rozpracované formuláře, ke kterým má přístupové právo. Zeleně podbarvené jsou ty, které čekají na jeho zpracování, žlutě podbarvené jsou ty, které může převzít, ostatní jsou podbarveny standardně bíle.
- **Uzavřené formuláře** – uzavřený je formulář poté, kdy byla jeho XML data uložena do databázových tabulek FormFlow. V případě formulářů určených pro sběr dat dochází k uzavření po správném vyplnění formuláře a odeslání dat. Formuláře, které musí projít schvalovacím procesem, se uzavírají po finálním schválení nebo zamítnutí formuláře. Vyplněný uzavřený formulář vidí ti, kteří jej podepsali, a ti, kteří jej uzavřeli. Nadřízení vidí formuláře svých podřízených.

Podle výše uvedených pravidel vidí formulář účastník schvalovacího procesu. Dále může existovat tzv. Správce skupiny, který vidí formuláře všech uživatelů ze své skupiny. Administrátor FormFlow vidí vše.

Ve FormFlow je u každého formuláře k dispozici:

- Popis, čeho se formulář týká, resp. co se v něm schvaluje.
- Seznam uživatelů, kteří se účastní schvalovacího procesu, s poznámkou u těch, kteří jej již schvalovali.
- Kdo jej odeslal, kdo jej právě vyřizuje a kdo jej vrátil.
- Zpráva přečtená z libovolného XML elementu datové věty použita pro informaci při postupování nebo vracení formuláře.

Stáhnout formulář a zobrazit si jej v aplikaci Software602 Form Filler může každý podle svých přístupových práv, ale odeslat jej s podpisem do systému může jen ten, kdo je právě na řadě podle pravidel schvalovacího procesu.

5.1.1 Postupování a převzetí formulářů

FormFlow umožňuje uživateli **postoupit** nepodepsaný formulář jinému uživateli, který má právo jej „vhodit“ do schvalovacího procesu (podepisovat jako první). Tato funkcionality slouží k tomu, aby jeden uživatel mohl založit formulář, předvyplnit jej daty a poté postoupit jinému uživateli k podpisu. Od FormFlow verze 2.5.3 je možné postoupit formulář ve všech podepisovacích krocích. **Převzetí** formuláře uživatelem je proces, kdy uživatel může převzít formulář čekající na zpracování jiným uživatelem a sám ho podepsat a tím pokračovat ve schvalovacím koloběhu.

5.1.2 Zastupování při podpisu formuláře

FormFlow řeší problematiku zastupování v případě nepřítomnosti uživatele. U každého uživatele se ukládá informace o době jeho nepřítomnosti.

Pro celý formulář nebo pro každý podepisovací krok zvlášť může být nastaven k uživateli jeho zástupce v případě nepřítomnosti. Pokud FormFlow zjistí, že uživatel, kterému postupuje formulář k podpisu, není přítomen, pokusí se najít zástupce. Pokud se mu to podaří, uvidí zástupce ve složce k podpisu formulář, kterým se má zabývat. V případě neúspěchu (zástupce není definován) formulář pošle do složky k podpisu původnímu uživateli. Při ukončení nepřítomnosti uživatele přesune FormFlow formuláře dosud nezpracované zástupcem zpět původnímu uživateli.

5.1.3 Koloběh ASiC formulářů

ASiC formuláře mohou být vyřízeny v rámci jednoho kroku více uživateli. Pokud je formulář předán jinému uživateli bez předání do dalšího podepisovacího kroku, jedná se o tzv. podkrok. Podkroků v rámci jednoho podepisovacího kroku může být více. Je-li formulář vrácen do předchozího kroku, je vždy vrácen poslednímu uživateli, který jej měl v rámci daného kroku na stole.

5.2 Datová věta

Koloběh formulářů pracuje s libovolným formulářem, kterému je nutné upravit datovou větu. Ta bude obsahovat níže uvedené tagy a atributy:

```
<root url_send="" url_forward="" url_back="" url_release="">
  <gw_numsigns></gw_numsigns>
  <gw_status></gw_status>
  <gw_notify></gw_notify>
  <gw_accept></gw_accept>
  <gw_refuse></gw_refuse>
  <gw_message></xg_message>
  <gw_signX sign_level="a" signed="b"
    pre_title="" name="" surname="" post_title="" email=""
    phone_number="" mobil_number="" fax_number="" street=""
    street_no="" zip="" city="" job="" department="" section=""
    note="" />
  <gw_fwdX>25</gw_fwdX>
  <gw_grp_fwdX>
    <gw_fwd_itemX id="" fullname=""
      pre_title="" name="" surname="" post_title="" email=""
      phone_number="" mobil_number="" fax_number="" street=""
      street_no="" zip="" city="" job="" department="" section=""
      note="" />
  </gw_grp_fwdX>
</root>
```

kde X je číslo nominálního kroku, tedy hodnoty 1 až maximální počet podpisů ve formuláři. Tagy vyznačené zelenou barvou nemusí být použity.

5.2.1 Při prvním volání

```
<root url_send="http://webserver/save.php"
      url_forward="http://webserver/save_forward.php"
      url_back=""
      url_release="">
```

K URL přibude parametr ...?id=ID_DOKUMENTU, kde bude informace o skutečném ID dokumentu v databázi a informace o tom, zda dokument otevřel správný uživatel s právem odeslat data na server.

```
<gw_numsigns>0</gw_numsigns>
```

Do tohoto tagu FormFlow nastaví číslo odpovídající počtu uživatelských podpisů ve formuláři. Toto číslo nabývá hodnot 0 až maximální počet podpisů ve formuláři. Při přeskakování podepisovacích kroků bude počet podpisů menší než nominální počet kroků.

```
<gw_status>0</gw_status>
```

Nastaví se dle stavu formuláře:

- 0 – formulář je ve schvalovacím procesu
- 1 – formulář je schválen
- 2 – formulář je zamítnut
- 3 – v prvním kroku při vyplňování anonymem nebo autorem

```
<gw_archive>0</gw_archive>
```

FormFlow nastaví na 1, pokud byla XML data uzavřena, jinak nastaví 0. Při editaci uzavřeného formuláře má hodnotu 2.

```
<gw_message></gw_message>
```

Tag se zprávou od serveru nebo pro předchozího podepisujícího uživatele. Při přechodu do dalšího podepisovacího kroku FormFlow tento tag vymaže.

XML data obsahující informaci o uživateli podepisujících formulář ve všech krocích:

```
<gw_signX sign_level="a" signed="b"
  pre_title="" name="" surname="" post_title="" email=""
  phone_number="" mobil_number="" fax_number="" street=""
  street_no="" zip="" city="" job="" department="" section=""
  note="" />
```

Kde x odpovídají nominálním číslům kroků. Začíná od 1 (číslo 1 obsahuje data autora). Tagů gw_signX je tolik, kolik je nadefinovaných nominálních podepisovacích kroků ve FormFlow.

Zde budou v prvním volání atributy u gw_sign1 nastaveny takto:

```
sign_level="1" signed="0"
```

FormFlow vloží dva datové číselníky se seznamem uživatelů.

```
<gw_fwdX>25</gw_fwdX>
```

V tomto tagu je ID uživatele, který dostává formulář v daném kroku jako první.

```
<gw_grp_fwdX>
  <gw_fwd_itemX id="11" fullname="Vonasek Karel Ing. (vedoucí IT)"
    pre_title="" name="" surname="" post_title="" email=""
    phone_number="" mobil_number="" fax_number="" street=""
    street_no="" zip="" city="" job="" department="" section=""
    note="" />
  <gw_fwd_itemX id="25" fullname="Navratil Jan Bc. (zástupce vedoucího
    IT)" pre_title="" name="" surname="" post_title="" email=""
    phone_number="" mobil_number="" fax_number="" street=""
    street_no="" zip="" city="" job="" department="" section=""
    note="" />
</gw_grp_fwdX>
```

Kde X odpovídají nominálním číslům kroků:

- **x=1** – datový číselník obsahuje seznam uživatelů, kteří mohou formulář vhodit do schvalovacího procesu, neboli kteří jej mohou podepisovat jako první.
- **x=2** – datový číselník obsahuje seznam uživatelů, kteří mohou formulář podepisovat jako druzí.

5.2.2 V každém kroku (včetně postupování nebo vracení)

```
<root url_send="http://webserver/save.php"
      url_forward="http://webserver/save_forward.php"
      url_back="http://webserver/save_back.php">
  <gw_numsigs>N</gw_numsigs>
```

Do tohoto tagu FormFlow nastaví číslo odpovídající počtu uživatelských podpisů ve formuláři. Toto číslo nabývá hodnot 0 až maximální počet podpisů ve formuláři. Při přeskakování podepisovacích kroků bude počet podpisů menší než nominální počet kroků.

```
<gw_status>0</gw_status>
```

Nastaví se dle stavu formuláře:

- 0 – formulář je ve schvalovacím procesu
- 1 – formulář je schválen
- 2 – formulář je zamítnut
- 3 – v prvním kroku při vyplňování anonymem nebo autorem

```
<gw_archive>0</gw_archive>
```

FormFlow nastaví na 1, pokud byla XML data uzavřena, jinak 0. Při editaci uzavřeného formuláře má hodnotu 2.

```
<gw_message></gw_message>
```

Tag se zprávou od FormFlow nebo pro předchozího podepisujícího uživatele. Při přechodu do dalšího podepisovacího kroku FormFlow tento tag vymaže.

```
<gw_signX sign_level="a" signed="b"
  pre_title="" name="" surname="" post_title="" email=""
  phone_number="" mobil_number="" fax_number="" street="" street_no=""
  zip="" city="" job="" department="" section="" note="" />
```

FormFlow nastaví **gw_signX** pro všechny nominální kroky podle následujících pravidel:

- v již provedených krocích nastaví atributy **sign_level** na počet podpisů ve formuláři před přidáním podpisu v daném nominálním kroku a **signed** na 1.
- u uživatele, který právě stahuje formulář a bude jej podepisovat, nastaví všechny atributy s informací o uživateli a **sign_level** na počet podpisů ve formuláři před podepsáním tímto uživatelem a **signed** na 0.
- U zbývajících podpisů nastaví všechny atributy s informací o uživateli a **sign_level** na "" a **signed** na "".

Atribut **sign_level** obsahuje skutečné číslo podpisu ve formuláři, které nemusí souhlasit s nominálním číslem kroku.

Atribut **signed** má tři hodnoty:

- "" – tento podpis není aktivní
- 0 – tento podpis právě bude podepisovat
- 1 – tento podpis již podepsal

FormFlow vloží datový číselník se seznamem uživatelů, kteří mohou podepisovat v následujícím kroku. Z tohoto číselníku si uživatel, který právě stahuje formulář, může vybrat z následujícího:

```
<gw_fwdY>id_fwd</gw_fwdY>
```

ID uživatele, který dostává formulář v následujícím kroku jako první (implicitní hodnota datového číselníku).

```
<gw_grp_fwdY>
```

```
  <gw_fwd_itemY id="11" fullname="Vonasek Karel Ing. (vedoucí IT)"
    pre_title="" name="" surname="" post_title="" email=""
    phone_number="" mobil_number="" fax_number="" street=""
    street_no="" zip="" city="" job="" department="" section=""
    note="" />
```

```
  <gw_fwd_itemY id="25" fullname="Navratil Jan Bc. (zástupce vedoucího
    IT)" pre_title="" name="" surname="" post_title="" email=""
    phone_number="" mobil_number="" fax_number="" street=""
    street_no="" zip="" city="" job="" department="" section=""
    note="" />
```

```
</gw_grp_fwdY>
```

5.2.3 V posledním kroku před uzavřením formuláře a uložením dat

```
<root url_release="http://webserver/release.php"
  url_send=""
  url_forward=""
  url_back="">
  <gw_numsigs>N</gw_numsigs>
  <gw_status>1</gw_status>
  <gw_message></xg_message>
  <gw_signX ...
```

5.3 Co čte FormFlow z XML dat

Po prvním podpisu (podpis autora):

```
<gw_notify>0</gw_notify>
```

Pokud si autor zaškrtně příslušnou volbu, dostane e-mailem zprávu o každém pohybu formuláře – tedy pokaždé, když jej někdo podepsal a odeslal na server.

V každém kroku:

```
<gw_refuse>0</gw_refuse>
```

Pokud formulář nastaví na 1, FormFlow vyhodnotí jako zamítnutí schvalovacího formuláře.

```
<gw_accept>0</gw_accept>
```

Pokud formulář nastaví na 1, FormFlow vyhodnotí jako schválení schvalovacího formuláře ještě dříve, než jej podepíše poslední schvalující.

```
<gw_message></xg_message>
```

Tag se zprávou pro předchozího podepisujícího uživatele.

```
<gw_fwdY>25</gw_fwdY>
```

ID uživatele, který bude podepisovat formulář v následujícím kroku. Tento element může být nastaven formulářem nebo FormFlow (záleží na autorovi formuláře). Y je nominální číslo kroku.

6 Úpravy formuláře pro koloběh dokumentů

V této kapitole si postupně ukážeme, jak upravit formulář v aplikaci Software602 Form Designer pro jeho nasazení do koloběhu dokumentů, který FormFlow poskytuje.

6.1 Informace o podepisujícím uživateli

Datová vazba elementu `signX` vypadá takto:

```
gw_signX
  sX.sign_level
  sX.signed
  sX.id
  sX.pre_title
  sX.name
  sX.surname
  sX.post_title
  sX.email
  sX.phone_number
  sX.mobil_number
  sX.fax_number
  sX.street
  sX.street_no
  sX.zip
  sX.city
  sX.job
  sX.department
  sX.section
  sX.note
```

Navíc jsou zde hodnoty všech tagů, které FormFlow používá pro přednastavené hodnoty (viz kapitolu *Data přihlášeného uživatele* na straně 6), ale bez předpony „gw“.

6.2 Datový číselník

Datová vazba datového číselníku:

```
gw_fwdX
gw_fwd_itemX
  fwdX.id
  fwdX.fullname           = „surname name pre_title (job)“
  fwdX.pre_title
  fwdX.name
  fwdX.surname
  fwdX.post_title
  fwdX.email
  fwdX.phone_number
  fwdX.mobil_number
  fwdX.fax_number
  fwdX.street
  fwdX.street_no
  fwdX.zip
  fwdX.city
  fwdX.job
  fwdX.department
  fwdX.section
  fwdX.note
```

Navíc jsou zde hodnoty všech elementů, které FormFlow používá pro přednastavené hodnoty (viz kapitulu *Data přihlášeného uživatele* na straně 6), ale bez předpony „gw“.

Příklad:

Uživatel si má vybrat dalšího podepisujícího z komba (=rozbalovacího seznamu), do kterého FormFlow naplní jména a ID ostatních příjemců. FormFlow tedy naplní strukturu `gw_grp_fwdX`, Form Filler naplní tato data do číselníku a uživatel z komba vybere podle příjmení uživatele, jehož ID se vloží do `gw_fwdX`.

V Software602 Form Designeru vytvoříme číselník, nazveme jej např. `forward` a navážeme na element `gw_fwdX`. Následně doplníme atributy číselníku:

- `form602indexlist` `gw_fwdX`
- `form602listname` `forward`
- `form602bind` `gw_fwdX`
- `form602ct` `ilb`
- `valuexp` `@id`
- `textxp` `@fullname`
- `form602reference` `//d:gw_fwd_itemX`

Název	Hodnota
form602indexlist	gw_fwd2
form602listname	forward
form602bind	gw_fwd2
form602ct	ilb
valuexp	@id
textxp	@fullname
	...

Příklad:

Z číselníku je také možné vybrat „skrytě“, tedy pomocí XPath výrazu zapsaného do pole Výpočet ve vazbě.

Atribut naplňovaný hodnotou z datového číselníku bude mít například následující výpočet:

Název: `sX.surname`

Vazební výraz: `@surname`

Výpočet:

`(//d:gw_grp_fwdX/d:gw_fwd_itemX [@id = //d:gw_fwdX])/@surname`

kde `X` odpovídají nominálním číslům kroků a nabývá hodnot 1 až maximální počet podpisů. Závislost musí být nastavena na `gw_fwdX`.

Příklad:

Element naplňovaný hodnotou z datového číselníku bude mít například následující výpočet:

Název: `mesto`

Vazební výraz: `/d:root/d:mesto`

Výpočet:

`(//d:gw_grp_fwdX/d:gw_fwd_itemX [@id = //d:gw_fwdX])/@city`

6.2.1 Podpisy ve formuláři

Při práci s formuláři je řada akcí závislá na elektronických podpisech ve formuláři. Níže popsaná problematika se týká především parciálně podepsaných formulářů, kde v každém podpisovém kroku jsou na základě relevancí (výraz zapsaný v datové vazbě) do datové věty zařazeny pouze relevantní elementy a formulář je pak opatřen elektronickým podpisem. Při další práci s formulářem bývá třeba správně zobrazovat dříve podepsaná data a jméno vlastníka odpovídajícího podpisu.

6.2.2 Typy podpisů ve formuláři

Formulář publikovaný programem Software602 Designer vždy obsahuje elektronický podpis vlastníka licence a může obsahovat i elektronický podpis vydavatele formuláře. Dále pak může obsahovat (parciální) podpisy jednotlivých uživatelů FormFlow. Každý podpis má své pořadové číslo (=úroveň v terminologii Designeru). Budeme tedy hovořit o **celkovém počtu podpisů** ve formuláři (tedy včetně licenčního a vydavatelského) nebo o **počtu uživatelských podpisů** (pouze podpisy uživatelů FormFlow ve vztahu k podepisovacím krokům).

6.2.3 Počty podepisovacích kroků ve FormFlow a odpovídající podpisy ve formuláři

Ve FormFlow se pro daný formulář definují podepisovací kroky. Provedení podepisovacího kroku uskutečňuje uživatel podepsáním formuláře svým elektronickým podpisem. Počty kroků jsou nominální a reálné:

Nominální počet kroků odpovídá počtu kroků nadefinovaných v konfiguraci FormFlow.

Nominální číslo kroku je pořadí kroku podle kroků nadefinovaných v konfiguraci FormFlow.

Reálný počet kroků je počet reálně provedených kroků a je menší nebo roven nominálnímu počtu kroků, menší je v případě přeskočení kroku(ů).

Pozn.: Podpisy tedy ve formuláři přibývají „jeden za druhým“, ale kroky se mohou přeskakovat. Navíc formulář může a nemusí obsahovat podpis vydavatele. Postup pro stanovení vztahu podepsaných dat, podpisu uživatele a případně podepisovacího kroku uvádíme níže.

Pro práci s podpisy ve formuláři jsou k dispozici dva typy funkcí a informací:

1. Funkce programu Software602 Form Filler pro práci s elektronickými podpisy ve formuláři

Tyto funkce jsou uvedeny v dokumentu „XPath výrazy, funkce a aplikační události.pdf“ (dostupný na www.602.cz), především v části „1.11 Popis funkcí rozšíření 602XML modulu XMLcore pro formulářové objekty“. Zde uvedeme pouze jejich výčet a základní vlastnosti:

- **GetNumSigns** vrací aktuální počet podpisů ve formuláři
- **SignLevel()** vrací TRUE, pokud aktuální počet podpisů ve formuláři odpovídá parametru.
- **PartSignLevel** vrací TRUE, pokud aktuální počet podpisů ve formuláři odpovídá parametru, přičemž ignoruje elektronický podpis návrháře formuláře (PUBLISHER), pokud je ve formuláři obsažen.
- **GetSignProperty** vrací konkrétní informace o elektronických podpisech, které jsou k dokumentu připojeny.

Příklad:

```
GetSignProperty(GetFieldSignerIndex(//*[@form602bind="supervisor"], SignSubject, 2.5.4.3)
```

Příklad syntaxe z form602extern:

```
GetSignProperty(GetFieldSignerIndex({//*[@form602bind="v_zprava1"}]), SignSubject, 2.5.4.3)
```

2. Informace o elektronických podpisech ve formuláři dodané FormFlow

Element **gw_signX** obsahuje informace o uživateli podepisujícím formulář. Mimo jiné se v něm nalézají atributy **sign_level** a **signed**:

+ **sign_levelX** – počet podpisů ve formuláři v kroku X v okamžiku před podpisem v daném kroku (na začátku 1 nebo 2), kde X je nominální číslo kroku

+ atribut **signed** má tři hodnoty:

- "" – tento podpis není aktivní
- 0 – tento podpis bude právě podepisovat
- 1 – tento podpis již podepsal

Při otevření čisté šablony formuláře nepodepsané vydavatelem je v **gw_sign1** atribut **sign_level=1** a **signed=0**. Přeskočený krok má **sign_level=0** a **signed=""**. A např. po skoku z prvního do čtvrtého kroku je pro **gw_sign4** atribut **sign_level=2** a **signed=0**.

– **gw_numsigns** – aktuální počet uživatelských podpisů ve formuláři.

6.3 Zobrazení podpisu

Elektronický podpis vytvoří uživatel FormFlow svým certifikátem. Při zobrazení jména uživatele tedy může autor formuláře zobrazit jméno uživatele FormFlow, který formulář podepsal, nebo jméno majitele certifikátu použitého v podpisu. Z bezpečnostního hlediska je správné pracovat s certifikátem.

K zobrazení celého jména podepsaného uživatele použijeme následující výraz:

```
GetSignProperty (Y, SignSubject, 2.5.4.3)
```

kde *Y* je číslo podpisu ve formuláři. Toto číslo může být pořadové číslo uživatelského podpisu + 1 nebo 2 (formulář podepsán vydavatelem). Tento postup bude pracovat správně pouze za předpokladu očekávaného počtu podpisů ve formuláři – přeskočení kroku nebo přidání podpisu vydavatele funkci naruší. Poslední parametr funkce je *Object Identifier* podle normy X.509, 2.5.4.3 je *cn* (jméno), 2.5.4.5 je sériové číslo certifikátu, 2.5.4.6 je *c* (země) atd.

Vhodnější varianta výrazu:

```
GetSignProperty (GetFieldSignerIndex ({/*[@form602bind="element"]}), SignSubject, 2.5.4.3)
```

kde *element* je libovolný element podepsaný daným podpisem (na základě *Relevance*). Tento výraz zobrazí jméno vlastníka certifikátu, kterým je podepsaná část formuláře obsahující uvedený element.

6.4 Vložení a podepsání dat na základě relevance

Vložení výrazu do položky *Relevance* zajistí autor formuláře, že element odpovídající aktuální vazbě bude vložen do podepisované datové věty pouze v případě platnosti vloženého výrazu. Naopak pomocí výrazu v položce *Pouze pro čtení* zajistí, že tento element bude zobrazen pouze pro čtení. V praxi se tedy tyto dva výrazy obvykle negují.

V položce *Relevance* můžeme použít následujících výrazů:

```
"SignLevel (X) "
```

tedy s pevným číslem podpisu ve formuláři,

nebo lépe:

```
"SignLevel (DataXpath (/d:gw_signX/@sign_level)) "
```

kde *X* je číslo podpisu a *sign_level* je počet vložených podpisů v daném podepisovacím kroku. Při použití tohoto výrazu se stane formulář nezávislým na případném podpisu vydavatele i na případném přeskočení kroku.

Do položky *Pouze ke čtení* vložíme adekvátně následující výrazy:

```
"Not (SignLevel (X) ) "
```

nebo

```
"Not (SignLevel (DataXpath (/d:gw_signX/@sign_level)) ) "
```

6.5 Přepínací sekce pro podpisy

Pokud chceme na základě počtu vložených uživatelských podpisů zobrazovat jen některé části formuláře, můžeme to provést pomocí přepínacích sekcí a použijeme pomocné elementy nazvané např. `signX`, které níže uvedeným výpočtem nabývají tří hodnot:

- 0 – `signLevel x` ještě není na řadě (není aktuální)
- 1 – `signLevel x` je právě na řadě – bude nyní podepisovat
- 2 – `signLevel x` již podepsal

Výpočet pro `signX` vypadá takto:

```
(//d:gw_signX/@sign_level > 0)*(  
  (//d:gw_signX/@signed=0)+2*(//d:gw_signX/@signed > 0))
```

Atribut `sign_level` udává počet podpisů ve formuláři před přidáním podpisu v daném nominálním kroku, `signed` udává stav tohoto podpisu:

Zopakujme, že atribut `signed` má tři hodnoty:

- "" – tento podpis není aktivní
- 0 – tento podpis bude právě podepisovat
- 1 – tento podpis již podepsal

6.6 Přepínací sekce pro uzavření schváleného nebo zamítnutého formuláře

Vytvoříme si pomocný element, např. `to_archive`, který se bude počítat dle následujícího vzorce:

```
((//d:gw_status >0) and (//d:gw_status <3) and  
  (//d:gw_archive = 0)) + (//d:gw_archive = 1)*2
```

Hodnota elementu `gw_status` je přednastavena FormFlow a nabývá tří hodnot:

- 0 – formulář je ve schvalovacím procesu
- 1 – formulář je schválen
- 2 – formulář je zamítnut

Hodnota elementu `gw_archive` je přednastavena FormFlow a nabývá těchto hodnot:

- 0 – XML data z formuláře ještě nejsou uzavřena
- 1 – XML data jsou již ve FormFlow uzavřena
- 2 – při editaci již uzavřeného formuláře

Pomocný element `to_archive` bude nabývat tří hodnot:

- 0 – ještě se nebude uzavírat
- 1 – formulář je schválen nebo zamítnut a je připraven k uzavření XML dat
- 2 – XML data z formuláře jsou již v FormFlow uzavřena

6.7 Přepínací sekce postoupení předvyplněného formuláře prvním podepisujícím

Pokud formulář ještě nebyl postoupen, má element `gw_status` hodnotu **3**, viz výše.

Vytvoříme si tedy pomocný element, např. `sign0`, jehož hodnota se bude počítat například podle následujícího vzorce

```
(//d:gw_status!=3)*1
```

Tento výpočet nabývá hodnot:

- 0 – formulář ještě nebyl postoupen
- 1 – formulář již byl postoupen prvním podepisujícím

Jestliže používáme ve FormFlow postupování formuláře, musí být pro první podpis (pro `sign1`) použit následující vzorec, který počítá s elementem `gw_status`:

```
(//d:gw_sign1/@sign_level>0)*((//d:gw_sign1/@signed=0)+2*(//d:gw_sign1/@signed > 0))*(//d:gw_status!=3)
```

Tento výpočet pak nabývá hodnot:

- 0 – formulář ještě nebyl postoupen
- 1 – formulář má podepsat první podepisující (ten, kterému byl postoupen)
- 2 – formulář je podepsán prvním podepisujícím

6.8 Oznamování členství ve skupinách

Při nastavování koloběhu formuláře lze také nastavit „Oznamování členství ve skupinách“.

Funkce: Pro daný formulář je ve FormFlow možné zvolit skupiny a jim odpovídající aliasy. Pokud je aktuální uživatel členem této skupiny, přidá FormFlow k elementům `<gw_signX>` a `<gw_grp_fwdX>` atribut `member_of`, do něhož zapíše zvolený alias. Obdobně přidá k elementu `<gw_current_user>` atribut `gw_member_of`. Stejný alias vloží v prvním kroku i do elementu `gw_member_of` = členství autora vyplněného formuláře ve skupině (v dalších krocích FormFlow tento element nenaplní). Pokud je uživatel členem více sledovaných skupin, FormFlow zapíše do uvedeného atributu alias odpovídající první nalezené skupině, proto lze v uživatelském prostředí volit pořadí sledovaných skupin pomocí šipek vpravo.

Význam této funkce spočívá v tom, že umožňuje sledovat ve formuláři členství uživatele v určité skupině a v závislosti na tom měnit chování formuláře.

6.9 Editace (oprava) vyplněného uzavřeného formuláře

Oprávněný uživatel může opravit data ve formuláři, který je již uzavřen. FormFlow ve formuláři nastaví pouze element `gw_archive` na hodnotu **2** a naplní `url_release` platnou URL pro odeslání opravené verze formuláře. Od verze 3.0 naplní ještě element `gw_current_user` daty uživatele, který si formulář k opravě otevřel. Opravovat data v uzavřeném formuláři může uživatel, který na to má právo, nebo uživatel, který formulář uzavřel nebo mohl uzavřít a je editace pro daný formulář povolena. U sběrového formuláře jej může opravovat i arbitr.

6.10 Konečný příjemce pro schválený formulář

Možnost z formuláře vybrat Konečného příjemce pro schválený formulář umožní uložení ID uživatele do elementu `<gw_final_app>`. Ve formuláři se seznam uživatelů vyplní do struktury:

```
<d:gw_grp_final_app>
  <gw_final_app_item id="25" fullname="Navratil Jan Bc. (zástupce vedoucího IT)"
    pre_title="" name="" surname="" post_title="" email="" phone_number="" mobil_number=""
    fax_number="" street="" street_no="" zip="" city="" job="" department="" section=""
    note="" />
</d:gw_grp_final_app>
```

6.11 Konečný příjemce pro zamítnutý formulář

Možnost z formuláře vybrat Konečného příjemce pro zamítnutý formulář umožní uložení ID uživatele do elementu `<gw_final_ref>`. Ve formuláři se seznam uživatelů vyplní do struktury:

```
<d:gw_grp_final_ref>
  <gw_final_ref_item id="25" fullname="Navratil Jan Bc. (zástupce vedoucího IT)"
    pre_title="" name="" surname="" post_title="" email="" phone_number="" mobil_number=""
    fax_number="" street="" street_no="" zip="" city="" job="" department="" section=""
    note="" />
</d:gw_grp_final_ref>
```

7 Možnosti ladění funkce formuláře ve FormFlow

Méně zkušenému autorovi formuláře nebo správci FormFlow se může stát, že právě dokončený formulář nebude po první instalaci do FormFlow pracovat správně a bude třeba hledat příčinu. Zde popíšeme prostředky, které jsou pro ladění k dispozici. Možnosti jsou v úpravě formuláře, použití ladicí funkce aplikace Software602 Form Filler a nastavení vyššího stupně výpisů (loglevel) ve FormFlow.

7.1 Formulář

7.1.1 Přidání kontrolních výpisů do formuláře

Nejsnadnější možností je přidání kontrolních výpisů některých elementů přímo do formuláře – zde se při práci s formulářem zobrazí hodnoty, podle kterých se např. přepínají sekce nebo které odcházejí zpět do FormFlow jako součást datové věty formuláře.

7.1.2 Kontrola obsahu XML souboru

Další možností je prohlížení obsahu formuláře jako XML dat. Ve formuláři jsou vidět elementy zobrazované na obrazovce a jednotlivé datové věty. Ve spodní části XML souboru snadno najdeme datovou větu formuláře a za ní i datovou větu připojenou FormFlow sloužící k předvyplnění hodnot v datové větě formuláře Form Fillerem – zde právě bývá nejčastěji problém, např. překlep ve jménech elementů, kvůli kterému se neprovede předvyplnění, případně chybná/neodpovídající struktura vět. Jak na to? Formulář je při stažení z FormFlow ve tvaru ZFO. Abychom ho mohli prohlížet jako XML data, je třeba koncovku přepsat na ZIP a soubor otevřít vhodným dekomprimačním programem. V tomto archívu bude soubor form.fo. Tento soubor zkopírujeme na disk a změníme jeho koncovku na XML. Potom je možné pouhým poklepáním soubor otevřít v programu pro přehledné zobrazení XML dat, kterým obvykle bude internetový prohlížeč, např. MSIE.

7.2 Software602 Form Filler

7.2.1 Ladicí konzole

Software602 Form Filler umožňuje zapnutí ladicí konzole (Ctrl+F5, resp. pro plugin Ctrl+0 (klávesa é/0, nikoliv 0 na numerické klávesnici)). Tato klávesová zkratka otevře ve spodní části okna programu podokno pro výpis zpráv programu.

7.2.2 Výpis datové věty

Stiskněte Ctrl+Shift+D kdykoliv z prostředí programu Software602 Form Filler nebo v pluginu z menu vyvolaného pravým tlačítkem myši v ladicí konzoli. Tím otevřete okno, kde je přehledným způsobem zobrazena XML datová věta a v ní obsažené hodnoty vypovídající o stavu formuláře v aplikaci Software602 Form Filler.

Ve Windows, kde není nainstalovaný vývojářský licenční certifikát užívaný pro provoz Software602 Form Designeru, je pouze možné v menu vyvolaném pravým tlačítkem myši v okně konzole zapnout logování provozu programu do souboru a konfigurovat výpisy o práci programu.

Po nainstalování licenčního certifikátu se zpřístupní ještě další volby menu:

- Řádka „**vyhodnocení výrazu**“ umožňuje přímo zapisovat XPath výrazy pracující s aktuálními daty v datové větě formuláře a po odklepnutí dojde ke zpracování a zobrazení výsledku. Tím je možné si vyzkoušet správný tvar výrazu nebo najít chybu v datech.
- Řádka „**spouštění aplikací**“ slouží ke spuštění událostí Software602 Form Filleru, např. odeslání dat podle odesílacího profilu.

7.3 FormFlow

7.3.1 Nastavení vyššího režimu logování (loglevel) ve FormFlow

V souboru config (bez koncovky) lze mj. nastavit úroveň výpisů serveru do log souboru. V poznámce u tohoto klíče je uveden výčet možných hodnot a jejich významu:

```
define('XMLGW_LOGFILE_LOGLEVEL',2); // 4=debug, 3=detailed, 2=extended, 1=normal, 0=none
```

Zbytečné zvýšení úrovně vede k nepřehlednosti výpisu, doporučujeme s tímto nastavením pracovat uvážlivě.

8 Archiv

8.1 Úvod

Vyplněné formuláře se mohou kromě standardního ukládání do složky uzavřených formulářů ukládat ještě do archivu. Archiv obsahuje složky vytvořené nezávisle na složkách s šablonami formulářů. Formulář se archivuje do složky archivu zvolené v konfiguraci formuláře ve FormFlow nebo do složky zvolené uživatelem ve formuláři. Archiv umožňuje archivaci dokumentů, zpracování jejich nových verzí a přímou URL na nejnovější verzi dokumentu. Pomocí spisových znaků a skartačních vlastností může FormFlow provádět řízenou skartaci formulářů.

8.2 Zapnutí archivu

Archiv je k dispozici pouze ve FormFlow, který obsahuje odpovídající licenci. Pro konkrétní formulář je třeba použití archivu zapnout v jeho Informacích u volby cílové složky archivu. Vypnout použití archivu pro tento formulář je možné, dokud nedošlo k prvnímu vyplnění tohoto formuláře.

8.3 Cílová složka pro archivaci formuláře v archivu

FormFlow archivuje formulář do složky nastavené v konfiguraci formuláře nebo do složky určené uživatelem ve formuláři. ID složky archivu zvolené uživatelem pro archivaci formulář uloží do elementu `gw_archive_id`.

Element `gw_archive_id` může uživatel naplnit hodnotou vybranou z datového číselníku `gw_grp_archive` s prvky `gw_archive_item`, které mají atributy `archive_id=""` a `archive_name=""`, kde v `archive_id` jsou ID složky archivu a v `archive_name` je název složky archivu. FormFlow tento datový číselník naplní jen těmi složkami archivu, které má uživatel právo „nastavovat“, a přidá ještě složku, která je již uvedena v `gw_archive_id`. Datový číselník se naplní jen tehdy, pokud se jedná o formulář pro archiv, tedy vzor formuláře má nastavenou výchozí složku archiv.

8.4 Archivace dokumentů v archivu

Dokument se v archivu ukládá jako příloha formuláře. Dokument může být součástí XML dat (zakódovaný do BASE64) nebo může být uložen v externím úložišti. Dokument nebo odkazy na něj se předávají do FormFlow v XML datech v elementu `gw_document`. Pro archivaci dokumentů každý formulář umožňuje vložení jedné binární přílohy, jejíž název se přenesení do elementu `doc_name` a mime type do elementu `mime_type`. Pokud je potřeba vložit více příloh, lze použít opakovací sekci. V elementu `gw_document` se vytvoří elementy či atributy: `doc_xpath_repe` (cesta na opakovací sekci), `doc_xpath_name` (název přílohy v opakovací sekci), `doc_xpath_mime_type` (mime type přílohy v opakovací sekci), `doc_xpath_content` (cesta na obsah přílohy v opakovací sekci). Ostatní XML elementy včetně případných dalších přiložených souborů ve formuláři slouží jako META DATA k uloženému dokumentu, podle kterých lze pak dokumenty následně vyhledávat pomocí fulltextového vyhledávání. Ostatní dokumenty uložené ve formuláři též mohou být přímo součástí XML věty nebo mohou být uloženy v úložišti vzdálených příloh. Formulář pro archivaci dokumentů může být „sběrový“ nebo „koloběhový (docflow)“ s libovolným počtem podepisovacích kroků.

Příklad:

```
<d:gw_document>
<d:doc_xpath_content>/d:root/d:prilohy_uloziste[1]/d:prilohy[1]/d:obsah_souboru[1]</d:doc_xpath_content>
<d:doc_xpath_mime_type>/d:root/d:prilohy_uloziste[1]/d:prilohy[1]/d:mime_type[1]</d:doc_xpath_mime_type>
<d:doc_xpath_name>/d:root/d:prilohy_uloziste[1]/d:prilohy[1]/d:nazev_souboru[1]</d:doc_xpath_name>
<d:doc_xpath_repe>/d:root/d:prilohy_uloziste[1]/d:prilohy[1]</d:doc_xpath_repe>
</d:gw_document>
```

8.5 Archivace dokumentů v archivu – LTV položky

Tyto položky slouží k zobrazení a práci s Externími podpisy pro účely LTV modulu. Použití je stejné, jako je popsáno v sekci Archivace dokumentů v archivu.

doc_xpath_ltv_accept – hodnotou bude cesta do té části opakovací sekce, která určuje, jestli se má dokument dlouhodobě udržovat. Přípustné hodnoty v této položce opakovací sekce jsou 0 nebo 1.

doc_xpath_cades_content – hodnotou bude cesta do té části opakovací sekce, jejímž obsahem je externí podpis.

doc_xpath_cades_mime_type – hodnotou bude cesta do té části opakovací sekce, jejímž obsahem je mime typ externího podpisu.

doc_xpath_cades_name – hodnotou bude cesta do té části opakovací sekce, jejímž obsahem je název externího podpisu.

8.6 Archivace dokumentů v archivu – LTA položky

LTA položky slouží k ukládání PDF/A přílohy vzniklé konverzí z jiného dokumentu.

V elementu **gw_document** je možné přidat atributy/elementy, které se následně budou ukládat do tabulky **xg_udat**. Obsahem těchto elementů/atributů jsou XPath výrazy na elementy/atributy, v nichž jsou uložena data o patřičné příloze.

doc_xpath_convert_pdfa – hodnotou je cesta do té části opakovací sekce, kde se určuje, jestli se dokument má konvertovat do PDF/A – možné hodnoty v položce opakovací sekce jsou 0 nebo 1.

doc_xpath_pdfa_ltv_accept – hodnotou je cesta do té části opakovací sekce, kde se určuje, jestli se dokument PDF/A má dlouhodobě udržovat – možné hodnoty v položce opakovací sekce jsou 0 nebo 1.

doc_xpath_pdfa_name – hodnotou je cesta do té části opakovací sekce, jejímž obsahem je název PDF/A souboru.

doc_xpath_pdfa_mime_type – hodnotou je cesta do té části opakovací sekce, jejímž obsahem je mime typ PDF/A souboru.

doc_xpath_pdfa_content – hodnotou je cesta do té části opakovací sekce, jejímž obsahem je nový soubor PDF/A.

8.7 Podpisová kniha

Podpisová kniha je nástroj na podepisování příloh ve formátu PDF. Podpis je vybrán ze seznamu uloženého ve FormFlow.

Datová vazba číselníku:

<code><d:gw_grp_sb_sign></code>	obalovací element pro výběr certifikátu
<code><d:gw_sb_sign_item</code>	číselník v datech
<code>sign_id=" "</code>	ID podpisu ve FormFlow
<code>sign_cn=" "</code>	jméno, na které byl certifikát vydaný
<code>sign_icn=" "</code>	vydavatel certifikátu
<code>sign_from=" "</code>	platnost certifikátu od
<code>sign_to=" "</code>	platnost certifikátu do
<code>sign_sn=" "</code>	sériové číslo certifikátu
<code></></code>	
<code><d:gw_sb_sign></code>	jeho hodnotou je @sign_id vybraného certifikátu

Rozšíření pro opakovací sekci s přílohami:

`<d:sb_sign_accept>` položka pro vyřazení/zařazení přílohy z podpisové množiny. Jedná se o hodnoty 0 (nepodepisovat) a 1 (podepisovat). Pokud bude element/atribut nevyplněný, zachází se s ním stejně, jako by obsahoval hodnotu 1.

Prvek slouží k určení, kterou část opakovací sekce podepsat a kterou ne. Tím lze nastavit podepisování například jen tří příloh z deseti.

`<d:sb_signed>` položka, kterou si FormFlow označuje, jestli podpis proběhl v pořádku (hodnota 1), nebo jestli nastala chyba (hodnota 0), příloha nebyla podepsána a při dalším pokusu bude opět odeslána k podpisu. V případě hodnoty 1 se tato příloha již nebude podepisovat.

Rozšíření pro vytahování XPath výrazů v sekci Správa – Správa formuláře – Verze – VerzeX – Přílohy:

`<d:doc_xpath_sb_sign_accept>` hodnotou je XPath cesta na element (`<d:sb_sign_accept>` položka pro vyřazení/zařazení přílohy z podpisové množiny) do opakovací sekce.

`<d:doc_xpath_sb_signed>` hodnotou je XPath cesta na element `<d:sb_signed>`.

8.8 Spisové znaky v archivu

Pro archivaci dokumentů jsou datové struktury FormFlow rozšířeny o tzv. spisové znaky:

- spisový znak: `gw_archive_ident_mark` (char[20])
- skartační znak: `gw_archive_shred_mark` (char[5])
- skartační lhůta: `gw_archive_shred_time` (integer) – počet roků

Spisové znaky jsou nastavovány ve správě formuláře, pak následně mohou být jejich hodnoty změněny v XML datech odeslaných do FormFlow. Formát spisových znaků FormFlow nekontroluje, v praxi je formát spisových znaků striktně řízen interními předpisy organizace.

8.9 Datum a čas archivace

Datum a čas archivace je uveden ve formuláři v připojených XML datech v elementech `gw_archive_date` a `gw_archive_time`. V datech se zobrazí pouze, pokud je pro formulář nastavena nějaká složka archivu. Zobrazeny jsou údaje ze sloupce `FILLED_TS_UFRM`.

8.10 Mazání archivovaných formulářů, kterým prošla skartační lhůta

FormFlow na pozadí maže archivované formuláře s prošlou skartační lhůtou.

Pokud bude nalezen formulář/dokument, který je starší než skartační lhůta, budou z FormFlow vymazány všechny verze formuláře. Formuláře budou vymazány definitivně, tedy nevhazují se do koše. K vymazání nedojde, pokud existuje rozpracovaná další verze dokumentu.

8.11 Verze

Formulář (pokud to má v konfiguraci povoleno) může mít v archivu verze. Číslo verze vkládá FormFlow do formuláře do elementu `gw_archive_version`. První uložený formulář=dokument má číslo verze 1.

8.12 Odkaz na dokument uložený ve formuláři

Uživatelé potřebují mít k dispozici odkazy na aktuální verze dokumentů uložených v archivu, které lze vystavit na webu. Odkaz, tedy URL adresa ukazující stále na nejnovější verzi dokumentu, je uvedena v informacích o formuláři uloženém v archivu.

Např.: <http://192.168.2.127/fs602/index.php?goto=sadoc&id=31>

Tato URL vyvolá přihlašovací stránku, po přihlášení uživatele je zahájeno stažení dokumentu (nikoliv formuláře!). Pokud je dokument přístupný i uživateli anonymous, pak se dokument stáhne hned, tedy bez přihlašování.

9 Převod původního typu formulářů na ASiC formulář

9.1 Úprava datové věty

V editaci zdroje vložit do datové věty následující XML:

```
<b:gw_form_signatures step="" substep="" code_form="" name_form="" id_form=""
  id_fmve="" id_layout="" relative_url_layout=""
  xmlns:b="http://www.software602.cz/forms/history">
<b:history>
<b:step relative_url="" step="" substep="" id_user="" login_user="" name=""
  surname="" pre_title="" job="" department="" fullname="" signtime="">
<b:signininfo>
<b:signtime/>
<dsig:KeyInfo xmlns:dsig="http://www.w3.org/2000/09/xmldsig#">
<dsig:X509Data>
<dsig:X509Certificate/>
</dsig:X509Data>
</dsig:KeyInfo>
</b:signininfo>
</b:step>
</b:history>
</b:gw_form_signatures>
```

9.2 Nastavení formuláře

Otevřete „Nastavení formuláře“ a do položky „Oddělené zpracování dat ve FormApps Serveru“ nastavte následující hodnoty:

- Oddělené zpracování dat ve FormApps Serveru: `true`
- Oddělené binární přílohy – jméno pluginu: `ASiC-SFLF`
- Oddělené binární přílohy – parametr pluginu: `sflf/put.php?pluginname=ASiC-SFLF`

Parametr pluginu není nutné zadávat přímo v tomto tvaru, je možné použít i vyhodnocování přes `$eval`.

Upozornění: Možnost ASiC-SFLF Designer nabízí jen v případě, že formulář má nastavené určení „Formulář určen pouze pro FormApps Server“. Filler oddělení dat nepodporuje.

Upozornění: U vzdálených příloh ve formuláři došlo ke změně formátu. Stávající APX link byl nahrazen XML strukturou `su:supplement`. Byl-li formát APX linku použit např. ve výpočtech, je třeba tyto výpočty změnit (nejspíše na úrovni backendu).

9.3 Úprava odesílacích skriptů

Pro plnou funkcionalitu služeb FormFlow (`url_send`, `url_release`, ...) je nutné změnit formát odesílaných dat následujícím způsobem:

```
var datapar = new DataParam();
var step = ele.XPath("/d:root/b:gw_form_signatures[1]/@step");
var substep = ele.XPath("/d:root/b:gw_form_signatures[1]/@substep");
datapar.workflowStep = step + "_" + substep;
datapar.requiredFormat = VarF602ASiCE;
```

9.4 Zobrazení historie

Pro zobrazení historie schvalovacích kroků je možné použít následující java script:

```
{
var doc = GetDocument();
var data = doc.TransformDataInstance("history");
doc.OpenSubform(data, "", "", "");
}
```

Transformace použitá ve skriptu vypadá následovně:

```
<xsl:stylesheet exclude-result-prefixes="xs" version="2.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:b="http://www.software602.cz/forms/history"
xmlns:d="http://software602.cz/sample">
<xsl:template match="/">
<xsl:apply-templates select="/d:root/b:gw_form_signatures/b:history"/>
</xsl:template>
<xsl:template match="@*|node()">
<xsl:copy>
<xsl:apply-templates select="@*|node()" />
</xsl:copy>
</xsl:template>
</xsl:stylesheet>
```

Uvedený skript zobrazí podformulář uložený v instalační složce FormApps Serveru.

Upozornění: Při použití jiné metody než výše uvedené transformace je třeba ohlídat, aby data odesílaná do podformuláře měla namespace: `http://www.software602.cz/forms/history`.

9.5 Nahrazení původního formuláře

Je-li původní formulář ve FormFlow nahrazen ASiC formulářem, je tato změna nevratná a dále již půjde nahradit jen dalším ASiC formulářem. Přejít zpět na původní typ formuláře již není možný.

10 Apendix A – Odkazy

Na formulář se lze odkazovat několika různými URL čili odkazy. Odkaz na formulář můžeme uvést na webové stránce, v obsahu informačního e-mailu, v jiném formuláři atd. Přístupová práva k šabloně formuláře řeší FormFlow, nikoliv aplikace, kde je odkaz uveden.

Uvedme zde varianty odkazů a vlastnosti:

10.1 Odkazy na šablonu formuláře

- Odkaz pro vyplnění šablony **přihlášeným uživatelem** FormFlow je uveden ve Správě formulářů, ve větvi Informace.

Např.: `/login.php?goto=fill&id=45`

- Odkaz pro vyplnění šablony formuláře **anonymním uživatelem** je uveden ve Správě formulářů, ve větvi Anonymní uživatel. Na tomto místě je také uvedena adresa `url_release` platná pro celý formulář, kterou je třeba doplnit do formuláře v případě, že bude k vyplňování distribuován jinou cestou než stažením z FormFlow (tedy e-mailem, stažením z webu nebo FTP atd.).

Např.: `/aforms.php?action=fill&id_form=45`

Tyto odkazy jsou nezávislé na verzi šablony formuláře.

10.2 Univerzální odkaz na formulář (rozpracovaný/uzavřený)

(platí pro FormFlow od verze 2.5.3 z listopadu 2009, případně s opravou z ledna 2010)

Odkaz má tento tvar:

`/login.php?goto=ushow&id=ID`

kde ID je ID částečně vyplněného formuláře (`id_ufrm`).

Tento odkaz ukazuje na částečně nebo zcela vyplněný formulář. Pro stažení formuláře uživateli postačí práva přístupu, není nutné, aby uživatel měl v daném okamžiku možnost pokračovat ve vyplňování či jiném zpracování tohoto formuláře.

10.3 URL pro zjištění informací o souboru v SFLF

Následující adresy lze použít pro získání informací o souborech (přílohách formulářů) uložených v externím úložišti.

- `sflf/get.php?id=<ID_FILE>&get_length=1` zjištění délky přílohy;
- `sflf/get.php?id=<ID_FILE>&get_hash=1` zjištění hashe přílohy;
- `sflf/get.php?id=<ID_FILE>&get_info=1` zjištění informací o souboru;
- `sflf/get.php?id=<ID_FILE>&get_part=1` část uloženého souboru od začátku pro rozlišení datové zprávy od formuláře zfo.