

# Binární XML

Marek Andrt

Katedra Informatiky, Fakulta elektrotechniky a informatiky, VŠB - Technická universita Ostrava, 17. listopadu 15, 708 33 Ostrava-Poruba, marek.andrt@vsb.cz

## Abstrakt

Reprezentace jazyka XML v binární formě je v současnosti stále aktuálnější problémem. Proto byla také poměrně nedávno konsorciem W3C sestavena pracovní skupina XBC (*XML Binary Characterization Working Group*)[1], jež by měla v budoucnu vytvořit standard pro uložení XML dat v binární podobě. Tento příspěvek se nejprve zabývá obecně důvody pro použití binárního formátu pro XML data a některými současnými technologiemi, jež tyto problémy řeší. Dále jsou zde popsány jednotlivé oblasti, ve kterých je binární formát XML dat potřebný a také vlastnosti binárního XML. Tyto vlastnosti jsou posuzovány z podstatné části z hlediska samotného formátu dat (serializace), ale i z hlediska algoritmického nebo z hlediska procesoru připravovaného formátu tak, jak je charakterizuje předběžně pracovní skupina XBC.

## ÚVOD

Značkovací jazyk XML[2] (*eXtensible Markup Language*) je dnes velmi rozšířeným jazykem pro reprezentaci dat. Je využíván a podporován v mnoha různých programovacích jazycích na odlišných počítačových platformách, v množství rozdílných aplikací a standardů. Jeho výhodou je jednotný způsob značkování dat, nezávislý na aplikační doméně. Důsledkem tohoto vzniklo mnoho obecně použitelných nástrojů, jako například parsery, generátory, API, validátory a také standardy jako XPath[3], XML Schema[4], XML Query[5] a jiné.

Avšak XML 1.x nemusí být vhodným formátem pro všechny účely, například v případě některých aplikací může režie potřebná při generování, přenosu, ukládání či načítání a syntaktické analýze být až příliš velká. Takové aplikace výhody XML nemohou využít. XML Binary Characterization Working Group se snaží konkrétně specifikovat tyto aplikace, nalézt jejich společné vlastnosti, z nichž poté odvodí potřeby, jež tyto aplikace kladou na binární formát XML dat. Pokud budou vlastnosti binárního formátu XML dat přesně specifikovány, bude možné navrhnout způsoby serializace a s pomocí metrik těchto jednotlivých žádoucích vlastností stanovit nejvhodnější kandidáty technologií serializace XML dat.

### 1.1 Základní výhody a zápory formátu XML

Hlavní výhody formátu XML dat vychází z jeho zavedení jako standardu, což umožňuje jednoduché vytváření, manipulaci a využití v různých programovacích jazycích a nástrojích. Další výhodou je popisný charakter tohoto formátu. Tato vlastnost je dána skutečností, že se jedná o textový dokument, jenž umožňuje získat informace o jeho obsahu jednoduše čtením textu dokumentu. Také však vychází z principu formátu, který vynucuje specifikovat informaci (hodnotu) jejím názvem a kontextem. Samozřejmě je tento popisný charakter relativní, ovšem výrazný oproti jiným datovým formátům a markantní oproti binárním formátům.

Na druhé straně tyto výhody přinášejí samozřejmě i zápory, jež mohou být klasifikovány do dvou kategorií a to z hlediska platformy, na které je daná aplikace provozována a velikosti daných XML dat [6].

V prvním případě se jedná o využití XML na omezených zařízeních, jako například mobilní

telefony a podobné, kde vytváří podstatnou roli omezení paměti, šíře přenosového pásma a také výpočetního výkonu, jenž souvisí se složitostí analýzy dat (tzv. parsing). Pro tuto třídu zařízení je ideální formát jednoduchý pro analýzu bez zbytečných datových redundancí.

V druhém případě je XML využíváno pro uložení velmi objemných dat na standardních platformách. Například jedná-li se o multimediální data či o velké databáze, kde jsou rovněž kladeny vysoké nároky na paměť, šíři přenosového pásma a výpočetní výkon. V této kategorii využití XML dat je také podstatnou nevýhodou absence možnosti náhodného přístupu k datům.

Většina záporů formátu XML figuruje ve vztahu k velikosti dat a výkonnosti dané platformy a mohou být relativně přijatelné v závislosti na konkrétní aplikaci, avšak jejich řešení navýšením výkonu je často drahé a neefektivní.

## 1.2 Využití komprese

V obou výše uvedených případech je při využití XML prvním problémem velikost dat. Tento problém lze částečně řešit jako v jiných případech použitím komprese. Nejjednodušší přístup je použití standardních kompresních algoritmů[6]. Například SVG[7] (*Scalable Vector Graphics*) standardizuje tento přístup specifikací SVG souboru tak, že může být uložen jako XML dokument nebo jako jeho komprimovaná verze.

XMill[8] je nástroj pro kompresi XML dosahující lepších výsledků nežli klasické kompresní algoritmy jako gzip a podobné a to pomocí rozdělení datových elementů do více datových proudů (int, char, string, base64, atd. ). Podobné hodnoty se samozřejmě komprimují lépe, analogie je komprese databází po sloupcích, jenž obsahují vždy podobnější hodnoty nežli řádky.

Formát WBXML[9] (*WAP Binary XML*) byl navržen pro komunikaci s mobilními zařízeními s užitím pomalého spojení pomocí formátu XML. Proto byla navržena kompaktní binární reprezentace XML dat, jež redukuje velikost přenášeného XML dokumentu. Formát je navržen tak, aby zachoval strukturu elementů XML a umožnil browseru přeskočit neznámé elementy či atributy. Binární formát kóduje zpracovanou formu XML dokumentu, jeho strukturu a obsah elementů. Příslušné elementy a atributy jsou nahrazeny číselnými hodnotami a tím je dosaženo komprese XML dokumentu. Přiřazení čísel názvům či hodnotám je uloženo v souboru, jenž může být použit u různých XML souborů stejného typu. Komprese je efektivní například v případě WML[10] (*Wireless Markup Language*).

## 1.3 Připojení binárních dat

Standardní formát XML neobsahuje binární data, ale pouze data textová a to v určeném specifikovaném kódování. Proto binární data, jako obrázky a podobné, nemohou být jednoduše přidána, protože každý znak v daném kontextu odpovídá deklarovanému kódování[6]. Aby binární data mohla být vložena do požadovaného kontextu XML souboru, používá se transformace těchto dat na text, tak aby vyhovovala užitému kódování. Nejběžnějším transformačním schématem je base64 mapující binární data na malý rozsah znaků, jenž je podmnožinou ASCII. V důsledku je však výsledná textová sekvence vždy větší nežli původní binární data. Načez poměr samozřejmě závisí na zvoleném schématu, může být například 4:3 pro base64 při kódování UTF-8, což je dosti nepříjemné jedná-li se o značné množství binárních dat.

V případě velkých dat je vhodné použít formát, jenž používá binární data v původní formě a zbytek dat zanechá ve formátu XML. Tohoto principu využívají technologie SWA[11] (*SOAP Messages with Attachments*), MTOM[12] (*SOAP Message Transmission Optimization Mechanism*) a XOP[13] (*XML Optimized Packagin*). Tyto přístupy jsou založeny na specifikaci MIME Multipart/Related Content-type[14] , jež popisuje mechanismus spojení dokumentů včetně souvisejících komponent do jednoho společného balíku, tak že jejich vazby na jiné dokumenty,

například pomocí URL, jsou zachovány. Každá část takto složeného balíku může obsahovat různé druhy dat jako XML, binární data a podobné. Specifikace *SOAP Messages with Attachments* popisuje jakým způsobem jsou vytvářeny zprávy protokolu SOAP, jež jsou složeny z více částí do jediného XML dokumentu. Kořenová část takového dokumentu obsahuje XML data. Naopak části, jež byly uloženy v kódování base64, jsou umístěny v odpovídajících přílohách. Tyto přílohy jsou poté odkazovány v původním XML dokumentu. Při přesunutí binárních dat do příloh nedochází ke zvětšení jejich velikosti a odpadá režie spojená s kódováním a dekódováním přes base64.

Jakmile jsou však binární data přesunuta do příloh, nejedná se již o korektní XML dokument a lze tento formát použít jen v případě, že odesílatel a příjemce souhlasí s výměnou zpráv složených z více částí. Pro zpracování těchto dokumentů, nemohou být poté použity standardní nástroje pro práci se základním formátem XML, jelikož použité nástroje musejí podporovat výše zmíněnou specifikaci MIME Multipart/Related Content-type. V případě modelů XOP a MTOM zůstávají binární data součástí XML dokumentu, avšak mohou být v případě přenosu pomocí SOAP zpráv dočasně dekódovány z base64 a přesunuty z dokumentu do jiné části zprávy, jež je pak složena z více částí.

Alternativou je samozřejmě reprezentace binárních dat pomocí XML a to tak, že trojici složek RGB představující barvu bodu reprezentujeme takto: `<bod r="256" g="128" b="0"/>`. Tato varianta však nevyhovuje v případě požadavku rozumné velikosti výsledného XML souboru, jelikož původní informace reprezentovaná v binární podobě třemi byty je expandována na 28 bytů v podobě textové. Výhodou tohoto přístupu je však možnost použití dotazovacích jazyků jako XPath, XQuery nebo také možnost použití transformací XSLT a dalších technologií souvisejících s XML.

#### **1.4 Využití schématu XML**

Schéma XML dokumentů je formální deklarace formátu množiny XML dokumentů, jenž vyhovují danému schématu. Toto schéma může být specifikováno s využitím XML DTD, XML Schema, Relax NG a dalších odpovídajících mechanismů. Některá schémata jako XML Schema popisuje např. který řetězec představuje celé číslo, datum, posloupnost znaků a jiné možné typy dat neboli také jaký typ elementu se může vyskytovat v daném kontextu.

Informace obsažená ve schématu daného XML dokumentu, může být využita při optimalizaci kódování tohoto dokumentu. Například pokud schéma ukládá, že daný řetězec reprezentuje 32 bitové číslo v pohyblivé řádové čarce, může být toto číslo kódováno přímo v binárním formátu, kde bude jeho velikost vždy 4 byty, narozdíl od jeho textové podoby, jež může zabírat i desítky bytů. Další výhodou je samozřejmě možnost přímého zavedení takového čísla do paměti a okamžité možnosti matematických operací bez nutnosti předchozího zpracování. Avšak jak již bylo řečeno, tyto optimalizace přinášející výše zmíněné výhody, mohou být použity pouze v případě XML dokumentů validních v daném schématu. Dané kódování se pak také nazývá kódování vědomé schématu (schema-aware encoding). Příkladem tohoto kódování může být formát Binary Format for MPEG-7, jenž efektivně reprezentuje metadata definované formátem MPEG-7.

## **2. AKTIVITY W3C KONSORCIA**

WWW Consortium (W3C) svolalo v září 2003 workshop, jenž měl rozhodnout zda, bude založena nová pracovní skupina za účelem vytvoření standardu pro binární XML. Na základě tohoto workshopu v březnu 2004 vznikla pracovní skupina XML Binary Characterization Working Group (XBC)[1] v jejímž plánu je shromáždění informací, jež pomohou na základě

potřebných oblastí využití binárního XML určit požadované vlastnosti tohoto formátu. Tyto materiály pak budou použity k rozhodnutí, zdali konsorcium W3C navrhne vývoj standardu pro binární XML.

Pracovní plán skupiny zatím počítá s vytvořením čtyř základních dokumentů, z nichž jsou tři již vydány v podobě draftů, tudíž se předpokládá jejich budoucí doplnění či změny. První dokument zabývající se oblastmi využití, v nichž je třeba binárního formátu XML, se nazývá XML Binary Characterization Use Case[15]. První draft tohoto dokumentu, byl vydána v červenci 2004. Na základě prvního dokumentu byly poté předběžně specifikovány požadavky na formát binárního XML a vydány v říjnu 2004 v dokumentu XML Binary Characterization Properties[16]. Třetí dokument, jenž navazuje na předcházející se snaží navrhnout metriky (způsoby měření) jednotlivých požadovaných vlastností nazývaný XML Binary Characterization Measurement Methodologies a byl vydán koncem února 2005. Z poznatků třetího dokumentu by následně mohl být vydán poslední ze čtveřice dokumentů charakterizující na základě příslušných metrik kandidátní technologie navržené skupinou XBC.

## 2.1 Případy využití binárního XML

Jak již bylo výše zmíněno, aby bylo možné formalizovat požadavky na binární formát XML, je nutné se nejprve zabývat konkrétními požadavky jednotlivých současných oblastí využití, ve kterých nyní vyvstává požadavek použít binární uložení XML dat. XML Binary Characterization Use Case popisuje 18 těchto oblastí (případů využití), analyzuje jejich požadavky a specifikuje jejich vlastnosti. Vlastnosti jsou děleny do tří skupin a to na vlastnosti jež pro danou oblast formát musí mít, na vlastnosti které by mohl mít a skupinu vlastností, jež by byly optimální.

Dále jsou uvedeny oblasti využití binárního formátu a vlastnosti, jež jsou nezbytně vyžadovány:

<b>Metadata v systému broadcastů</b> - robustnost, zpětná kompatibilita, obecnost, prostorová složitost, omezení velikosti procesoru formátu, podpora streamů, podpora fragmentace, neutralita platformy, kompaktnost, efektivita zpracování, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Pole reálných čísel užívaných v energetickém průmyslu</b> - zpětná kompatibilita, široké rozšíření, neutralita platformy, bezplatnost, kompaktnost, podpora integrace s XML technologiemi, efektivita zpracování, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Kompresa, serializace a přenos grafického modelu X3D</b> - akcelerovaný sekvenční přístup, podpora zapouzdření, identifikaceverze formátu, podpora šifrování, podpora roundtripu, podpora podpisu, široké rozšíření, podpora streamů, neutralita platformy, bezplatnost, kompaktnost, podpora integrace s XML technologiemi, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Webové služby pro malá zařízení</b> - omezení velikosti procesoru formátu, podpora streamů, bezplatnost, kompaktnost, podpora integrace s XML technologiemi, efektivita zpracování, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

<b>Webové služby uvnitř společnosti</b> - podpora šifrování, podpora podpisu, široké rozšíření, bezplatnost, podpora integrace s XML technologiemi, efektivita zpracování, neutralita vzhledem
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
<b>Elektronické dokumenty</b> - neutralita platformy, bezplatnost, podpora integrace s XML technologiemi, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
<b>FIXLM v oblasti bezpečnosti</b> - široké rozšíření, neutralita platformy, kompaktnost, nezávislost na přenosu.
<b>Infra/Inter podniková komunikace</b> - bezplatnost, kompaktnost, efektivita zpracování, nezávislost na přenosu, podpora přímého čtení a zápisu.
<b>XMPP komprese instantní výměny zpráv</b> - identifikace verze formátu, rozšíření a odchylky schématu, podpora roundtripu, podpora streamů, podpora fragmentace, neutralita platformy, bezplatnost, kompaktnost, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
<b>XML Dokumenty v perzistentním úložišti</b> - explicitní typování, obecnost, podpora podpisu, omezení velikosti procesoru formátu, podpora streamů, podpora fragmentace, neutralita platformy, nezávislost na přenosu, podpora přímého čtení a zápisu.
<b>Analýza procesů a znalostí</b> – delty, efektivní update, podpora zapouzdření, rozšíření a odchylky schématu, náhodný přístup, podpora fragmentace, neutralita platformy, podpora integrace s XML technologiemi, efektivita zpracování, podpora přímého čtení a zápisu.
<b>Směrování XML na základě obsahu</b> – náhodný přístup, efektivita zpracování, nezávislost na přenosu, podpora přímého čtení a zápisu.
<b>Směrování webových služeb</b> – náhodný přístup, bezplatnost, podpora integrace s XML technologiemi, efektivita zpracování, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
<b>Interoperabilita militárních informací</b> – obecnost, rozšíření a odchylky schématu, prostorová složitost, podpora roundtripu, omezení velikosti procesoru formátu, široké rozšíření, podpora fragmentace, neutralita platformy, kompaktnost, podpora integrace s XML technologiemi, efektivita zpracování, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.
<b>Komunikace a zpracování dat senzoru</b> - obecnost, prostorová složitost, podpora roundtripu, omezení velikosti procesoru formátu, široké rozšíření, kompaktnost, efektivita zpracování, podpora přímého čtení a zápisu.
<b>SyncML protokol pro synchronizaci dat</b> - podpora šifrování, kompaktnost, efektivita zpracování.
<b>Superpočítače a gridy</b> – rozšířitelné body, explicitní typování, samostatnost, podpora integrace s typem media, cena implementace, delty, efektivní update, podpora zapouzdření, identifikace verze formátu, rozšíření a odchylky schématu, náhodný přístup, široké rozšíření, podpora

fragmentace, neutralita platformy, bezplatnost, efektivita zpracování, neutralita vzhledem k jazykům, nezávislost na přenosu, podpora přímého čtení a zápisu.

## **2.2 Požadované vlastnosti binárního XML**

Na základě výše uvedených případů vyžadujících binární formát XML dat, byly formálně specifikovány obecné vlastnosti vycházející z těchto oblastí. Pracovní dokument XML Binary Characterization Properties dělí tyto vlastnosti do tří skupin a to na vlastnosti algoritmické, dále z hlediska formátu (serializace) a na vlastnosti procesoru, který by měl být schopný zpracovávat daný formát. Jsou zde také uvedeny požadavky na formát nesouvisející s vlastnostmi uvedených tří skupin. Tyto vlastnosti jsou však také velmi významné pro srovnávání jednotlivých návrhů formátu. Pracovní dokument, u něhož se předpokládají další dílčí změny a rozšíření, obsahuje v současném vydání 3 formalizace vlastností z algoritmického hlediska, dále 30 z hlediska formátu a 5 dalších podstatných požadavků.

Dále jsou uvedeny tyto výše zmíněné vlastnosti a požadavky:

### **1 Z hlediska algoritmického**

1.1 EFEKTIVITA ZPRACOVÁNÍ

1.2 OMEZENÍ VELIKOSTI PROCESORU FORMÁTU – velikost procesoru (programu) zpracovávající daný binární formát, je závislá na platformě a programovacím jazyku, avšak může být srovnávána s velikostí standardního procesoru zpracovávajícího XML.

1.3. PROSTOROVÁ SLOŽITOST

### **2 Z hlediska formátu**

2.1 AKCELEROVANÝ SEKVENČNÍ PŘÍSTUP

2.2 KOMPAKTNOST – závisí na velikosti použité paměti, neboli uložené reprezentace daných XML dat, pokud má být splněna, měl by formát obsahovat pouze informace nutné pro jeho kompletní a správné dekódování.

2.3 PODPORA INTEGRACE S TYPEM MEDIA

2.4 DELTY - jsou reprezentace libovolných změn určité instance dané báze rodičovského dokumentu, jenž mohou spolu s ním reprezentovat nový stav tohoto dokumentu.

2.5 PODPORA PŘÍMÉHO ČTENÍ A ZÁPISU

2.6 EFEKTIVNÍ UPDATE

2.7 PODPORA ZAPOUZDŘENÍ - je vlastnost umožňující instanci daného formátu obsahovat v sobě další soubory různých formátů.

2.8 PODPORA ŠIFROVÁNÍ

2.9 EXPLICITNÍ TYPOVÁNÍ - je vlastnost formátu, jenž přímo obsahuje informace o datovém typu svých položek.

2.10 ROZŠÍRITELNÉ BODY - je to metoda pro jednoduché rozšíření formátu a jeho implementace.

2.11 IDENTIFIKACE VERZE FORMÁTU

2.12 PODPORA FRAGMENTACE – podporuje možnost vytvořit instance formátu, které nereprezentují celý dokument společně s dostatečným kontextem k dekódování minimálně smysluplným způsobem.

2.13 OBECNOST – takový formát, jenž je použitelný pro širokou škálu aplikací a případů použití, jenž mají odlišné vlastnosti, například od velikostí 10B-10GB, strukturované či nestrukturované data a podobně.

2.14 NEUTRALITA VZHLEDEM K JAZYKŮM – Formát nebude optimální pouze pro určitou skupinu národních jazyků.

- 2.15 ČITELNOST A EDITOVATELNOST
- 2.16 PODPORA INTEGRACE S XML TECHNOLOGIEMI
- 2.17 LOKALIZOVANÉ ZMĚNY – je splněna pokud, se minimální změna obsahu položky projeví jako jedna změna malého rozsahu bytu formátu.
- 2.18 ABSENCE OMEZENÍ
- 2.19 NEUTRALITA PLATFORMY
- 2.20 NÁHODNÝ PŘÍSTUP
- 2.21 ROBUSTNOST – je splněna, pokud daný formát umožňuje jeho procesoru oddělit chybné sekce tak, že neovlivní zpracování dalších zbývajících sekcí.
- 2.22 PODPORA ROUNDTRIPU – *z/do XML* je splněna, pokud konverze *do/z* daného formátu/XML a zpět do XML/daného formátu, vytvoří shodný výstup původnímu vstupu. Nemusí se jednat o přesnou shodu ve smyslu srovnání bytů, např. u XML.
- 2.23 ROZŠÍŘENÍ A ODCHYLKY SCHÉMATU – je schopnost reprezentovat položky, jenž buď nejsou definovány schématem příslušného dokumentu a nebo se přímo neřídí příslušnou definicí daného schématu.
- 2.24 ODOLNOST PROTI ZMĚNÁM SCHÉMATU
- 2.25 SAMOSTATNOST – je splněna, pokud pouze informace potřebná k reprodukci datového modelu dané instance (i) je reprezentací datového modelu dané instance a (ii) je specifikací XML formátu.
- 2.26 PODPORA PODPISU
- 2.27 *SPECIALIZOVANÉ ROZŠÍŘENÍ PROCESORU* – umožňuje asociovat procesor formátu s rozšířením (jinak také pluginem či kodekem) se specifickou částí dokumentu a dekodovat tak danou část optimálněji, nežli pomocí původního přístupu procesoru.
- 2.28 PODPORA STREAMŮ
- 2.29 PODPORA KOREKČÍ CHYB
- 2.30 NEZÁVISLOST NA PŘENOSU – je splněna, pokud předpokládáme transportní službu bez možnosti zanesení chyb a neomezující délku přenášené zprávy.

### 3 Další požadavky

- 3.1 *ZPĚTNÁ KOMPATIBILITA*
- 3.2 *CENA IMPLEMENTACE*
- 3.3 *BEZPLATNOST*
- 3.4 *JEDINÁ ODPOVÍDAJÍCÍ TRÍDA IMPLEMENTACÍ* – je splněna, pokud daná implementace podporuje všechny vlastnosti definované specifikací, jenž ještě nemusí být potřebné ve všech případech použití.
- 3.5 *ŠIROKÉ ROZŠÍŘENÍ* – je splněno, pokud je formát využíván na širokém rozsahu zařízení, na různých platformách a na převážném počtu těchto zařízení je používán v širokém spektru aplikací. Pokud se jedná o nový formát, může tuto vlastnost splnit pokud splňuje jiné vlastnosti jako nízkou cenu implementace a podobné.

### 3. ZÁVĚR

Široké rozšíření formátu XML vedlo k jeho využití i v aplikacích, pro které nebyl návrh tohoto formátu nejoptimálnější. Tato situace vedla k navržení některých variant technologií pro binární data s kombinací formátu XML, jež byly výše zmíněny a které mají svým způsobem překonat toto omezení. Některé z těchto technologií jsou již poměrně rozšířené a ratifikované jako doporučení či standard. Konsorcium W3C se nyní snaží prosadit jednotné řešení, jež by se v budoucnu mohlo stát v tomto směru standardem. Avšak situace není jednoduchá vzhledem k širokému spektru požadovaných případů využití binárního XML, které vede k množství

požadovaných vlastností, jež jsou samozřejmě značně závislé.

## LITERATURA

- [1] WWW Consortium, BXC WG (XML Binary Characterization Working Group), March 2004, <http://www.w3c.org/TR/Binary/>
- [2] WWW Consortium, Extensible Markup Language (XML) 1.1, Feb 2004, <http://www.w3.org/TR/xml11/>
- [3] WWW Consortium, XML Path Language (XPath) 1.0, Nov 1999, <http://www.w3.org/TR/xpath>
- [4] WWW Consortium, XML Schema, Oct 2004, <http://www.w3.org/TR/xmlschema-0/>
- [5] WWW Consortium, XQuery 1.0: An XML Query Language, Feb 2005, <http://www.w3.org/TR/xquery/>
- [6] GOLDMAN, O. Binary XML. *Dr. Dobb's Journal*, Nov 2004.
- [7] WWW Consortium, SVG (Scalable Vector Graphics), 2000, <http://www.w3c.org/Graphics/SVG/>
- [8] XMill, 2003, <http://sourceforge.net/projects/xmill/>
- [9] WWW Consortium, WBXML (WAP Binary XML Content Format), 1999, <http://www.w3c.org/TR/wbxml/>.
- [10] Wireless Application Protocol Forum, WAP WML, Feb 2000, [http://www.openmobilealliance.org/release\\_program/docs/Browsing/V2\\_1-20021101-C/WAP-191-WML-20000219-a.pdf](http://www.openmobilealliance.org/release_program/docs/Browsing/V2_1-20021101-C/WAP-191-WML-20000219-a.pdf)
- [11] SOAP Messages with Attachments, Dec 2000, <http://www.w3.org/TR/SOAP-attachments>
- [12] SOAP Message Transmission Optimization Mechanism, Jan 2005, <http://www.w3.org/TR/soap12-mtom/>
- [13] XML-binary Optimized Packaging, Jan 2005, <http://www.w3.org/TR/xop10/>
- [14] MINE Multipart/Related Content-type, Aug 1998, <http://www.ietf.org/rfc/rfc2387.txt>
- [15] W3C, XML Binary Characterization Use Case, Feb 2005, <http://www.w3.org/TR/xbc-use-cases/>
- [16] W3C, XML Binary Characterization Properties, Feb 2005, <http://www.w3.org/TR/xbc-properties/>
- [17] W3C, XML Binary Characterization Measurement Methodologies, FEB 2005, <http://www.w3.org/TR/xbc-measurement/>