



Architektura WWW, první vydání

Český překlad doporučení W3C, 15. prosinec 2004

Originál této verze:

<http://www.w3.org/TR/2004/REC-webarch-20041215/>

Jeho poslední verze:

<http://www.w3.org/TR/webarch/>

Předchozí verze:

<http://www.w3.org/TR/2004/PR-webarch-20041105/>

Vydavatelé:

Ian Jacobs, W3C

Norman Walsh, Sun Microsystems, Inc.

Autoři:

Podívejte se na poděkování (§8, str. 43).

Autor překladu: Filip Rychetský

I přes veškerou snahu může tento dokument obsahovat chyby způsobené překladem. Pokud nějaké nepřesnosti objevíte, kontaktujte prosím autora na <rychetsky@atlas.cz>

Poslední verze překladu:

<http://www.kosek.cz/w3c/webarch/>

Překlad ve formátu PDF:

<http://www.kosek.cz/w3c/webarch/webarch.pdf>

Podívejte se prosím do **seznamu oprav** k originálnímu dokumentu, který obsahuje normativní opravy.

Viz též **překlady**.

Copyright © 2002-2004 W3C[®] (MIT, ERCIM, Keio), All Rights Reserved. W3C liability, trademark, document use and software licensing rules apply. Your interactions with this site are in accordance with our public and Member privacy statements.

Přehled

World Wide Web užívá relativně jednoduché technologie s dostatečnou rozšiřitelností, efektivitou a utilitami, které měly za následek vznik pozoruhodného informačního prostoru propojených zdrojů, spojující jazyky, kultury a média. Ve snaze uchovat tyto vlastnosti informačního prostoru související s vývojem technologií, diskutuje tento architekturní dokument o hlavních návrhových součástech webu, které jsou identifikacemi zdrojů, reprezentací stavu zdrojů, a o protokolech, které podporují interakci mezi agenty a zdroji v prostoru. Dokument líčí zejména hlavní návrhové součásti, omezení a dobré praxe k zásadám a vlastnostem, které podporují.

Statut tohoto dokumentu

Tento oddíl popisuje statut tohoto dokumentu v době jeho zveřejnění. Další dokumenty mohou tento dokument nahradit. Seznam aktuálních W3C publikací a poslední revize této technické zprávy může být nalezena na W3C technical reports index na <http://www.w3.org/TR/>.

Toto je doporučení z 15. prosince 2004 "Architektury World Wide Webu, první vydání." Tento dokument byl zhodnocen členy W3C, softwarovými vývojáři a jinými skupinami W3C a zájemci a je schválený ředitelem jakožto doporučení W3C. Je to "stabilní" dokument a může být užíván i jako odkazový materiál nebo být citován v jiném dokumentu. Rolí

W3C při vytváření doporučení je upozornit na specifikaci a podporovat její rozšíření. Tím se zvýší funkčnost a součinnost webu.

Tento dokument byl vyvinut skupinou W3C - Technical Architecture Group (TAG), která udržuje seznam architektonických problémů. Jeho rozsah je užitečnou podmnožinou těchto problémů a není záměrem oslovit všechny z nich. TAG se zamýšlí zabývat zbývajících (a budoucích) problémy právě zde v tomto prvním vydání, které je zveřejněno jako doporučení W3C. Kompletní historie změn je také k dispozici. Prosím pošlete komentáře k tomuto dokumentu na public-webarch-comments@w3.org (veřejný archiv veřejných komentářů k webové architektuře). Technická diskuze skupiny TAG se nachází na www-tag@w3.org (veřejný archiv [www-tag](http://www-tag.org)).

Tento dokument byl vytvořen v souladu s W3C IPR policy of the July 2001 Process Document. TAG udržuje veřejný seznam zveřejněných patentů týkající se tohoto dokumentu; můžete zde také nalézt instrukce pro zveřejnění patentu. Jednotlivec, který má faktickou znalost patentu a věří, že obsahuje základní podmínku(y) pro zveřejnění, s ohledem na tuto specifikaci, by měl odhalit informaci v souladu s odstavcem 6 Patentové politiky W3C.

Obsah

1. Úvod	7
1.1. O tomto dokumentu	8
1.1.1. Lidé, kterým je tento dokument určený	8
1.1.2. Rozsah dokumentu	8
1.1.3. Zásady, omezení a dobré praktické poznámky	9
2. Identifikace	9
2.1. Výhody URI	9
2.2. URI/Zdrojové vztahy	11
2.2.1. Kolize URI	11
2.2.2. Alokace URI	11
2.2.3. Nepřímá identifikace	12
2.3. Porovnání URI	12
2.3.1. Aliasy URI	12
2.3.2. Opětovné použití reprezentace	13
2.4. Schémata URI	13
2.4.1. Registrace schématu URI	14
2.5. Neprůhlednost URI	14
2.6. Identifikátory části	15
2.7. Budoucí směry pro identifikátory	15
2.7.1. Internacionalizované identifikátory	16
2.7.2. Tvrzení, že dva URI identifikují stejný zdroj	16
3. Interakce	16
3.1. Použití URI k zpřístupnění zdroje	17
3.1.1. Detaily o získávání reprezentace	17
3.2. Typy reprezentací a typy internetových médií	18
3.2.1. Typy reprezentace a sémantika identifikátoru části	18
3.2.2. Identifikátory části a domlouvání obsahu	19
3.3. Rozporuplnosti mezi reprezentacemi dat a metadat	19
3.4. Bezpečné interakce	21
3.4.1. Nebezpečné interakce a odpovědnost	21
3.5. Správa reprezentací	23
3.5.1. Stálost URI	22
3.5.2. Odkazování a řízení přístupu	23
3.5.3. Doprovodná navigace	23
3.6. Budoucí směry pro interakci	24
4. Formáty dat	24
4.1. Binární a textové formáty dat	24
4.2. Verzování a rozšiřitelnost	26
4.2.1. Verzování	25
4.2.2. Verzování a politika jmenného prostoru XML	25
4.2.3. Rozšiřitelnost	26
4.2.4. Skládání formátů dat	27
4.3. Oddělení obsahu, prezentace a interakce	27
4.4. Hypertext	28
4.4.1. Odkazy URI	29
4.5. Formáty dat založené na XML	33
4.5.1. Kdy použít formát dat založený na XML	30
4.5.2. Odkazy v XML	30
4.5.3. Jmenné prostory XML	30

4.5.4. Dokument jmenného prostoru	31
4.5.5. Kvalifikovaná jména v XML	32
4.5.6. Sémantika XML ID	32
4.5.7. Typy médií pro XML	33
4.5.8. Identifikátory části v XML	33
4.6. Budoucí směry pro formáty dat	34
5. Všeobecné architekturní zásady	34
5.1. Ortogonální specifikace	34
5.2. Rozšiřitelnost	34
5.3. Ošetření chyb	35
5.4. Součinnost založená na protokolu	36
6. Slovníček	37
7. Odkazy	41
7.1. Architektonické specifikace	42
8. Poděkování	43

Seznam zásad, omezení a užitečných praktických rad a poznámek

Následující zásady, omezení a praktické poznámky se probírají v tomto dokumentu a jsou zde uvedené pro usnadnění. Můžete se také podívat na samostatné hrnutí.

Identifikace

- Globální identifikátory (zásada, 2)
- Identifikování s URI (dobrá praxe, 2.1)
- URI identifikují jediný zdroj (omezení, 2.2)
- Vyhněte se aliasům URI (dobrá praxe, 2.3.1)
- Shodné užívání URI (dobrá praxe, 2.3.1)
- Opětovné používání schémat URI (dobrá praxe, 2.4)
- Neprůhlednost URI (dobrá praxe, 2.5)

Interakce

- Opětovné používání formátů reprezentace (dobrá praxe, 3.2)
- Neslučitelnost ve vztahu data - metadata (omezení, 3.3)
- Asociace metadat (dobrá praxe, 3.3)
- Bezpečné získání (zásada, 3.4)
- Dostupná reprezentace (dobrá praxe, 3.5)
- Odkaz neznamená získání (zásada, 3.5)
- Shodná reprezentace (dobrá praxe, 3.5.1)

Formáty dat

- Verze informace (dobrá praxe, 4.2.1)
- Politika jmenného prostoru (dobrá praxe, 4.2.2)
- Mechanismy rozšiřitelnosti (dobrá praxe, 4.2.3)
- Shoda rozšiřitelnosti (dobrá praxe, 4.2.3)
- Neznámá rozšíření (dobrá praxe, 4.2.3)
- Oddělení obsahu, prezentace, interakce (dobrá praxe, 4.3)
- Identifikování odkazu (dobrá praxe, 4.4)
- Odkazování na webu (dobrá praxe, 4.4)
- Obecné URI (dobrá praxe, 4.4)
- Hypertextové odkazy (dobrá praxe, 4.4)

- Přijetí jmenného prostoru (dobrá praxe, 4.5.3)
- Dokumenty jmenného prostoru (dobrá praxe, 4.5.4)
- Kvalifikovaná jména nerozeznatelná od URI (omezení, 4.5.5)
- Mapování kvalifikovaného jména (dobrá praxe, 4.5.5)
- XML a "text/*" (dobrá praxe, 4.5.7)
- XML a kódování znaků (dobrá praxe, 4.5.7)

Všeobecné architekturní zásady

- Ortogonalita (zásada, 5.1)
- Zotavení po chybě (zásada, 5.3)

1. Úvod

World Wide Web (WWW nebo jednoduše Web) je informační prostor, ve kterém položky zájmu, označované jako zdroje, jsou identifikovány globálními identifikátory, které se nazývají jednotné identifikátory zdroje (URI =Uniform Resource Identifier).

Příklady, jako je následující cestovní scénář, se užívají v celém tomto dokumentu k ilustraci typického chování webových agentů - lidí nebo softwaru jednajícího v tomto informačním prostoru. Uživatelský agent jedná v zastoupení uživatele. Softwaroví agenti zahrnují servery, servery proxy, vyhledávače, prohlížeče a multimediální přehrávače.

Příběh

Při plánování cesty do Mexika čte Nadia "Povětrnostní informace v Oaxaca: `http://weather.example.com/oaxaca`" v lesklém časopise cestovního ruchu. Nadia má dostatek zkušeností s webem, aby rozpoznala, že `http://weather.example.com/oaxaca` je URI, a že asi bude schopna vyhledávat přidružené informace pomocí jejího webového prohlížeče. Jakmile Nadia zadá URI do jejího prohlížeče:

1. Prohlížeč pozná že to, co Nadia napsala, je URI.
2. Prohlížeč vykoná požadavek k získání informací v souladu s jeho konfigurací pro zdroje identifikované přes "http" schéma URI.
3. Autorita odpovědná za "weather.example.com" poskytne informaci v odpovědi na základě požadované žádosti.
4. Prohlížeč interpretuje odezvu, identifikovanou serverem jako XHTML, a vykonává další dotazovací akce pro vložení grafiky a jiného obsahu podle potřeby.
5. Prohlížeč zobrazuje získané informace, které zahrnují hypertextové odkazy k ostatním informacím. Nadia může následovat tyto hypertextové odkazy k získání dalších informací.

Tento scénář ilustruje tři architektonické základy webu, o kterých se diskutuje v tomto dokumentu:

1. Identifikace (§2, str. 9). URI se používají k identifikaci zdroje. V tomto cestovním scénáři je zdroj pravidelně aktualizovaná zpráva o počasí v Oaxaca a URI je `http://weather.example.com/oaxaca`.
2. Interakce (§3, str. 16). Weboví agenti komunikují používáním standardizovaných protokolů, které umožňují interakci přes výměnu zpráv, které se řídí definovanou syntaxí a sémantikou. Zadáním URI do dotazovacího dialogu nebo vybráním hypertextového odkazu říká Nadia svému prohlížeči, aby vykonal požadavek k získání informací ze zdroje, který se rozpozná podle URI. V tomto případě pošle prohlížeč žádost HTTP GET (část protokolu HTTP) na server "weather.example.com", přes TCP/IP port 80 a server pošle zpět zprávu obsahující, co určuje reprezentaci zdroje v čase, v kterém byla reprezentace vygenerována. Všimněte si, že tento příklad je specifický pro hypertextové vyhledávání informací-další druhy interakce jsou možné, obojí uvnitř prohlížečů a za použití jiných druhů webových agentů; náš příklad má za cíl ilustrovat jednu běžnou interakci, nikoliv definovat řadu možných interakcí nebo omezit způsoby, jakým by mohli agenti používat web.
3. Formáty (§4, str. 24). Většina protokolů užívaných pro získávání reprezentace a (nebo) odeslání použije posloupnost jedné nebo více zpráv, které společně obsahují důležitý obsah reprezentace dat a metadat, a přenesou tak reprezentaci mezi agenty. Volba dorozumívacího protokolu stanoví meze na formáty reprezentace dat a metadat, které mohou být přeneseny. HTTP například typicky přenáší jednotlivé toky bytů plus metadatum a užívá v hlavičce "Content-Type" a "Content-Encoding" k další identifikaci formátu reprezentace. V našem scénáři je přenášená reprezentace v XHTML identifikována pomocí "Content-Type" v hlavičce HTTP obsahující registrované jméno typu internetového média "application/xhtml+xml". Toto jméno signalizuje, že zobrazení dat může být zpracováno podle specifikace XHTML.

Nadine prohlížeč je konfigurovaný a naprogramovaný tak, aby interpretoval výsledek zobrazení "application/xhtml+xml" jako instrukce k poskytnutí obsahu tohoto zobrazení podle stávajícího modelu XHTML, včetně nějakých vedlejších interakcí (jako žádost o externí styly nebo vložení obrázků), které jsou volány k zobrazení. Ve scénáři reprezentace dat XHTML přijatá od počáteční žádosti instruuje Nadine prohlížeč, aby našel a okamžitě poskytl povětrnostní mapy, přičemž každá je identifikována URI, a způsobí tak další dotazy mající

za následek další zajištění, která jsou zpracována prohlížečem podle jejich vlastních formátů dat (například, "application/svg+xml" signalizují SVG formát dat), a tento proces pokračuje dál, dokud nebudou poskytnuty všechny formáty dat. Výsledek všech těchto zpracování, jakmile prohlížeč dosáhl takového aplikačního stavu, který se shoduje s počáteční požadovanou akcí Nadi, je obvykle označován jako "webová stránka".

Následující ilustrace ukazuje vztah mezi identifikátorem, zdrojem a reprezentací.

URI

`http://weather.example.com/oaxaca`

Identifies

Resource

Oaxaca Weather Report

Represents

Representation

```
Metadata:  
Content-type:  
application/xhtml+xml
```

```
Data:  
<!DOCTYPE html PUBLIC "...  
    "http://www.w3.org/...  
<html xmlns="http://www...  
<head>  
<title>5 Day Forecaste for  
Oaxaca</title>  
...  
</html>
```

Ve zbytku tohoto dokumentu zdůrazníme důležité zásadní body architektury týkající se webových identifikátorů, protokolů a formátů. Budeme také hovořit o některých důležitých obecných architekturních zásadách (§5, str. 34) a jejich použití v rámci webu.

1.1. O tomto dokumentu

Tento dokument popisuje vlastnosti, které od webu požadujeme, a volby návrhu, kterými jsme tyto vlastnosti realizovali. Podporují opětovné použití existujících standardů, když se to hodí, a dává rady, jak provádět inovace ve způsobu shodném s architekturou webu.

Termíny MUSÍ, NESMÍ, MĚL BY, NEMĚL BY a MŮŽE jsou použity u zásad, omezení a u dobrých praxí v souladu s RFC 2119 [RFC2119].

Tento dokument nezahrnuje ustanovení o shodě z těchto důvodů:

- Předpokládá se, že přizpůsobovat software by bylo tak rozmanité, že by nemělo smysl odkazovat se na vyšší třídy vyhovujících softwarových agentů.
- Některé praktické rady se týkají lidí; specifikace obecně definují shodu pro software, ne pro lidi.
- Nevěříme, že přidáním sekce "shoda" bychom zvětšili užitečnost dokumentu.

1.1.1. Publikum tohoto dokumentu

Tento dokument má za cíl informovat o problémech Webové architektury. Oslovené publikum pro tento dokument zahrnuje:

1. Účastníky ve W3C Activities
2. Další skupiny a jedince navrhující technologie, které se dají začlenit do Webu
3. Implementátory specifikací W3C
4. Autory a vydavatele obsahu Webu

Poznámka: Tento dokument nerozlišuje nijak formálně termíny "jazyk" a "formát." Který termín je použit, to vyplývá z kontextu. Fráze "návrhář specifikace" v sobě zahrnuje jazykové, formátové a protokolové návrháře.

1.1.2. Rozsah dokumentu

Tento dokument představuje obecnou architekturu webu. Další skupiny uvnitř a vně W3C se také zabývají specializovanými aspekty webové architektury zahrnující přístupnost, záruku jakosti, internacionalizaci, nezávislost na zařízení a webové služby. Část Architektonické specifikace (§7.1, str. 42) zahrnuje odkazy na tyto související specifikace.

Tento dokument usiluje o rovnováhu mezi stručností a precizností, včetně názorných příkladů. Nálezy TAG jsou informační dokumenty, které doplňují stávající dokument tím, že poskytují více detailů o vybraných tématech. Tento dokument zahrnuje některé výňatky z těchto nálezů. Jelikož se nálezy vyvíjejí nezávisle, zahrnuje tento dokument odkazy na schválené nálezy TAG. Pro jiné záležitosti TAG pokryté tímto dokumentem, ale které by neměly schválený závěr, vedou odkazy k seznamu uveřejnění TAG.

Mnoho příkladů v dokumentu, které zahrnují lidskou aktivitu, předpokládají důvěrně známý model interakce webu (ilustrovaný na začátku kapitoly Úvod), kde osoba následuje odkaz přes uživatelského agenta, uživatelský agent získá a prezentuje data, uživatel následuje další odkaz atd. Tento dokument detailněji nediskutuje o dalších takovýchto modelech jako je hlasové ovládání (viz například [VOICEXML2]). Volba modelu interakce může mít dopad na očekávané chování agenta. Například když se grafický uživatelský agent běžící na laptopu nebo přenosném zařízení setká s chybou, tak může oznámit chybu přímo uživateli skrz vizuální a zvukové podněty, a prezentuje uživateli volby, jak vyřešit problém. Na druhé straně, když někdo brouzdá webem pomocí hlasového vstupu a jediným výstupem je zvukový výstup, pak, jestliže agent oznámí chybu zase jen hlasově, zastaví se dialog, čeká se na uživatelský vstup, a to může redukovat použitelnost, protože je tak snadné se ztratit při brouzdání s jediným zvukovým výstupem. Dokument nediskutuje o tom, jak by se zásady, omezení a dobré praxe, které jsou zde popsány, používaly ve všech interakčních souvislostech.

1.1.3. Zásady, omezení a dobré praktické poznámky

Důležité body tohoto dokumentu jsou rozříděné následujícím způsobem:

Zásada

Architektonická zásada je základní pravidlo, které platí pro velké množství situací a proměnných. Architektonické zásady zahrnují "oddělení zájmů", "obecné rozhraní", "samopopisnou syntaxi", "viditelnou sémantiku", "síťový efekt" (Metcalfov zákon) a Amdahlův zákon: "Rychlost systému je omezená jeho nejpomalejší součástí."

Omezení

Některé volby v designu webu jako jména p a li elementů v HTML, volba dvojtečky (:), znak v URI nebo sdružování bitů do osmibitových jednotek (oktetů), jsou poněkud libovolné; jestliže by se vybral paragraph namísto p nebo hvězdička (*) namísto dvojtečky, pak by byl výsledek ve velkém měřítku nejspíše stejný. Tento dokument se soustředí na základnější návrhové volby: návrhové volby, které vedou k omezení, tj. omezení v chování nebo v interakcích uvnitř systému. Omezení mohou být předepsána pro technické, politické nebo pro další důvody k dosažení žádoucích vlastností v systému, jako přístupnost, globální rozsah, relativní snadnost vývoje, efektivita a dynamická roztažnost.

Dobrá praxe

Dobré praxe softwarových vývojářů, autorů obsahu, správců webu a návrhářů specifikací, které zvyšují hodnotu webu.

2. Identifikace

Za účelem vnitřní komunikace se komunita shoduje (v rozumném rozsahu) na souboru podmínek a jejich významů. Jediným cílem webu od jeho začátku bylo vybudovat globální společenství, v kterém jakákoliv společnost může sdílet informaci s jinou společností. K dosažení tohoto cíle používá web jednotného globálního identifikačního systému: URI. URI jsou základním kamenem architektury webu, poskytující identifikaci, která je přes web běžná. Globální rozsah URI podporuje ve velkém měřítku "síťové efekty": hodnota identifikátoru se zvyšuje tím více, čím více je konzistentně používán (například, více je používán v hypertextových odkazech (§4.4, str. 28)).

Zásada: Globální identifikátory

Globální pojmenování vede ke globálním síťovým efektům.

Tento princip se datuje zpět přinejmenším až k seminární práci Douglase Engelbarta o otevřených hypertextových systémech; viz kapitola Every Object Addressable v [Eng90].

2.1. Výhody URI

Volba syntaxe pro globální identifikátory je poněkud libovolná; to je dáno jejich globálním rozsahem, který je důležitý. Jednotný identifikátor zdroje, [URI], byl úspěšně nasazen od vytvoření webu. Zahrnutí účasti v existující síti URI nese značné výhody obsahující odkazování, tvorbu záložek, ukládání do vyrovnávacích pamětí a indexování vyhledávacími stroji, a to také vyžaduje značné výdaje při vytvoření nového identifikačního systému, který má stejné vlastnosti jako URI.

Dobrá praxe: Identifikujte pomocí URI

Ke zvýhodnění a zvýšení hodnoty World Wide Webu BY MĚLI agenti pro zdroje poskytnout URI jako identifikátory.

Ke zdroji by měl být přidružený URI, jestliže by k němu další strana chtěla vytvořit hypertextový odkaz, nebo by o něm chtěla učinit nebo vyvrátit tvrzení, získat nebo ukrýt jeho reprezentaci, zahrnout celý nebo jen část odkazem do další reprezentace, poznamenat si jej nebo s ním vykonávat další operace. Softwaroví vývojáři by měli očekávat, že

sdílení URI aplikacemi bude užitečné, dokonce i když užitečnost není zpočátku očividná. Nález TAG "*URIs, Addressability, and the use of HTTP GET and POST*" diskutuje o dalších výhodách a předpisech adresovatelnosti URI.

Poznámka: Některá schémata URI (jako třeba specifikace schématu URI "ftp") používají termín "určený," ale tento dokument užívá "identifikovaný."

2.2. URI/Zdrojové vztahy

URI úmyslně identifikuje jeden jediný zdroj. Nemůžeme omezit rozsah toho, jak by mohl vypadat zdroj. Termín "zdroj" je používán v obecném smyslu pro cokoliv, co by mohlo být identifikované pomocí URI. Tradiční je v hypertextu webu popisovat jako "zdroje" webové stránky, obrázky, katalogy produktů atd. Rozlišující charakteristika pro zdroje je, že všechny jejich podstatné znaky mohou být zprostředkované ve zprávě. Tuto sadu znaků nazýváme jako "informační zdroj."

Tento dokument je příkladem informačního zdroje. Skládá se ze slov a interpunkčních symbolů, grafiky a dalších věcí, které mohou být zakódované s proměnnými stupni přesnosti do posloupnosti bitů. Nenajdou se zde takové informace, které by se nemohly poslat zprávou. V případě tohoto dokumentu je sdělením zprávy reprezentace dokumentu.

Nicméně naše použití termínu zdroj je úmyslně širší. Další věci jako auta a psi (a jestli jste tiskli dokument, věc, kterou držíte v ruce) jsou také zdroje. Nejsou to informační zdroje, protože jejich podstatou není informace. Ačkoli je možné zapsat velké množství informací o autě nebo psovi do posloupnosti bitů, součet těchto informací bude jen stálým přiblížením základního charakteru zdroje.

Definujeme termín "informační zdroj," protože jsme vysledovali, že to je užitečné v diskusích o technologii webu a také při vytváření specifikací zařízení vytvořených k používání na webu.

Omezení: URI identifikuje jediný zdroj

Přiřaďte přesný URI k přesným zdrojům.

Protože rozsah URI je globální, zdroj identifikovaný pomocí URI nezávisí na kontextu, v kterém se URI vyskytuje (viz také část o nepřímé identifikaci (§2.2.3, str. 12)).

[URI] je dohoda o tom, jak internetové společenství přidělí jména a spojuje je se zdroji, které identifikují. URI jsou rozděleny do schémat (§2.4, str. 13), která definují přes své specifikace schémat mechanismus, které specifické identifikátory jsou spojovány se zdroji. Například schéma URI "http" ([RFC2616]) používá HTTP servery založené na technologii DNS a TCP k účelu alokace a rozlišení identifikátoru. Výsledkem toho je, že identifikátor jako "http://example.com/somepath#someFrag" často nabývá významu dle zkušeností komunity vykonáváním HTTP GET požadavku na identifikátor, a jestli je odezva úspěšná, pak se odezva ukáže v podobě reprezentace identifikovaného zdroje. (viz také identifikátory části (§2.6, str. 15).) Samozřejmě, požadavek jako GET není jediný způsob, jak získat informaci o zdroji. Někdo by mohl také uveřejnit dokument, který by definoval význam konkrétního URI. Tyto další prameny zpráv mohou navrhnout význam pro takovéto identifikátory, ale je otázkou místního politického rozhodnutí, zda tyto návrhy brát v potaz.

Stejně tak by si někdo mohl přát odkázat se na osobu různými jmény (celým jménem, pouze křestním, sportovní přezdívkou a tak dále), proto architektura webu dovolí přidružení se zdrojem z více než jednoho URI. URI, které identifikují stejný zdroj, se nazývají aliasy URI. Kapitola Aliasy URI (§2.3.1, str. 12) diskutuje o některých potenciálních nákladech při vytváření vícenásobného URI pro stejný zdroj.

Několik kapitol z tohoto dokumentu nastíní otázky o vztahu mezi URI a zdroji včetně:

- Když prozkoumám URI, jak hodně mohu říci o zdroji, že je tímto URI identifikován? Viz kapitola Schémata URI (§2.4, str. 13) a Neprůhlednost URI (§2.5, str. 14).
- Kdo určí, jaký zdroj URI identifikuje? Viz kapitola Alokační URI (§2.2.2, str. 11).

- Může se zdroj identifikovaný URI změnit v průběhu doby? Viz kapitola Stálost URI (§3.5.1, str. 22) a Správa reprezentací (§3.5, str. 23).
- Jestliže více jak jeden URI může identifikovat stejný zdroj, jak mohu vědět který URI identifikuje stejný zdroj? Viz kapitola Porovnání URI (§2.3, str. 12) a Tvzení, že dva URI identifikují stejný zdroj (§2.7.2, str. 16).

2.2.1. Kolize URI

URI úmyslně identifikuje jeden zdroj. Používání stejného URI k přímému identifikování jiných zdrojů způsobuje kolizi URI. Kolize zvyšují náklady na komunikaci kvůli vynaloženému úsilí, které je potřebné k odstranění "dvojsmyslů."

Předpokládejme například, že jedna organizace využije URI, aby se odkázala na film *The Sting*, a další organizace používá stejný URI k odkázání na diskusní fórum o *The Sting*. U třetí osoby, která si je vědoma obou organizací, tato kolize způsobí zmatek o tom, co URI identifikuje, a tím se snižuje hodnota URI. Pokud by někdo chtěl mluvit o datu vyhotovení zdroje identifikovaného podle URI, tak by například nebylo jasné, zda to znamená "kdy byl vytvořený film" nebo "kdy bylo vytvořeno diskusní fórum o filmu."

Proto byla vymyšlena společenská a technická řešení k tomu, aby pomáhala se kolizím URI vyhnout. Nicméně úspěch nebo selhání těchto různých přístupů závisí na rozsahu trvale definujících specifikací, na kterých se shodlo internetové společenství.

Kapitola Alokace URI (§2.2.2, str. 11) probírá přístupy pro stanovení autoritativního pramene informací o tom, jaký zdroj je identifikován URI.

URI jsou někdy používány pro nepřímou identifikaci (§2.2.3, str. 12), což nutně nemusí vést ke kolizím.

2.2.2. Alokace URI

Alokace URI je proces asociace URI se zdrojem. Alokace může být vykonána oběma majiteli zdroje a i jinými stranami. Je důležité vyhnout se kolizím URI (§2.2.1, str. 11).

2.2.2.1. Vlastnictví URI

Vlastnictví URI je vztah mezi URI a společenskou jednotkou, jako je osoba, organizace nebo specifikace. Vlastnictví URI dává příslušné společenské jednotce jistá práva včetně:

1. předat vlastnictví některé nebo všech vlastněných URI dalšímu majiteli - Delegation; a
2. spojovat zdroj s vlastněným URI - Alokace URI.

Dle společné úmluvy se vlastnictví URI deleguje od IANA URI scheme registry [IANASchemes], což je sama společenská jednotka na specifikace schémat URI, které jsou registrované v IANA. Některé specifikace schémat URI dále delegují vlastnictví vedlejším registrům nebo jiným navrhnutým vlastníků, kteří mohou dále delegovat vlastnictví. V případě specifikace spočívá vlastnictví nakonec na komunitě, která specifikaci udržuje.

Přístup pokládaný za "http" schéma URI například následuje vzor, kterým internetová společnost deleguje oprávnění, prostřednictvím IANA registru schémat URI a DNS, přes soubor URI s běžnou předponou k jednomu vlastníku. Jeden důsledek tohoto přístupu je silné spolehnutí webu na centrální registr DNS. Odlišný přístup je převzatý z URN Syntax scheme [RFC2141], které deleguje vlastnictví z části prostoru URN k jmennému prostoru URN specifikace, které je zaregistrované v záznamech IANA-maintained registry of URN Namespace Identifiers.

Vlastníci URI jsou odpovědní vyvarovat se přiřazení ekvivalentních URI k více zdrojům. Tudiž, jestliže se specifikace schématu URI stará o poskytování delegace jedné nebo organizované skupiny URI, tak by se mělo usilovat o zajištění toho, aby vlastnictví nakonec zůstalo v rukou jediné společenské jednotky. Dovolení vícenásobného vlastnictví zvyšuje pravděpodobnost kolize URI.

Vlastníci URI mohou organizovat nebo rozmístit infrastrukturu, aby zajistili, že jsou dostupné reprezentace z přidružených zdrojů, a v případě potřeby je interakce se zdrojem možná výměnou reprezentací. Zde se nacházejí společenská očekávání

pro zodpovědnou správu reprezentací (§3.5, str. 23) vlastníky URI. Další společenské důsledky vlastnictví URI se zde neprobírají.

Podívejte se na uveřejnění TAG siteData-36, který se týká vyvlastnění jmenující autority.

2.2.2.2. Jiná schémata alokace

Některá schémata používají jiné techniky než je delegování vlastnictví, aby se vyhnuly kolizi. Například specifikace pro schéma URL data (sic) scheme [RFC2397] říká, že zdroj identifikovaný schématem URI data má jedinou možnou reprezentaci. Data reprezentace tvoří URI, který identifikuje zdroj. Tudíž specifikace sama určí, jak jsou alokována data URI; možnost nedelegovat je přípustná.

Další schémata (jako "news:comp.text.xml") spoléhá se na společenský proces.

2.2.3. Nepřímá identifikace

Výrok, že URI "mailto:nadia@example.com" identifikuje současně email a osobu Nadiu, představuje kolizi URI. Nicméně můžeme použít URI k nepřímému identifikování Nadi. Identifikátory se takto běžně používají.

Pokud budeme poslouchat vysílání zpráv, je možné slyšet zprávu o Británii, která začíná "dnes, 10 Downing Street oznámila sérii nových ekonomických měřítek." Obecně výraz "10 Downing Street" identifikuje oficiální sídlo britského ministerského předsedy. V kontextu reportér používá toto spojení (tak, jak to anglická mluva vyžaduje) k nepřímé identifikaci britské vlády. Podobně URI identifikují zdroje, ale také mohou být použité v mnoha pojmech nepřímě k identifikování dalších zdrojů. Podle globálně převzaté politiky jsou některé URI ustanoveny jako univerzální identifikátory. Místní politika stanoví, co budou nepřímo identifikovat.

Předpokládejme, že `nadia@example.com` je Nadina emailová adresa. Organizátoři konference, kterou Nadia navštěvuje, by se mohli na "mailto:nadia@example.com" na Nadiu nepřímo odvolávat (například použitím URI jako databázového klíče v databázi účastníků konference). Tímto způsobem nenastane kolize URI.

2.3. Porovnání URI

URI, které jsou identické znak po znaku, odkazují na stejný zdroj. Protože architektura webu dovolí asociaci více URI s daným zdrojem, dva URI, které nejsou identické znak po znaku, mohou stále odkazovat na stejný zdroj. Různé URI ještě nutně nemusí odkazovat na různé zdroje, ale určitě pak, že různé URI odkazují na stejný zdroj, je obecně náročnější.

K redukci rizika negativní chyby (tj. chybný závěr, že dva URI neodkazují na stejný zdroj) nebo pozitivní chyby (tj. nesprávný závěr, že dva URI odkazují na stejný zdroj) popisují některé specifikace navíc testy rovnocennosti k srovnání znak po znaku. Agenti, kteří dosáhnou závěrů založených na srovnáních, která nejsou definována významnými specifikacemi, přebírají odpovědnost za jakékoliv problémy, které by mohly z výsledků vyplývat; viz kapitola Ošetření chyb (§5.3, str. 35) pro více informací o zodpovědném chování při dosažení neoprávněných závěrů. Kapitola 6 [URI] poskytuje více informací o srovnávání URI a snižování rizika negativních a pozitivních chyb.

Viz také tvrzení, že dva URI identifikují stejný zdroj (§2.7.2, str. 16).

2.3.1. Aliasy URI

Ačkoli mají aliasy URI výhody (jako flexibilita pojmenování), mají i své nevýhody. Aliasy URI jsou škodlivé tehdy, když dělí web souvisejících zdrojů. Důsledek Metcalfova zákona ("síťový efekt") je to, že hodnota daného zdroje může být měřena číslem a hodnotou dalších zdrojů v jeho síťovém sousedství, to jest zdrojů, které jsou s ním propojeny.

Problém s aliasy je ten, že jestliže polovina sousedství ukazuje na jeden URI pro daný zdroj a další polovička ukazuje na druhý, jiný URI, ale pro ten samý zdroj, sousedství je rozdělené. Nejen že je aliasový zdroj podhodnocen kvůli tomuto rozštěpení, ale celé sousedství zdrojů ztrácí hodnotu kvůli chybějícím vztahům druhého řádu, které by měly existovat mezi doporučujícími zdroji z důvodu jejich odkazů na jinak zvaný zdroj.

Dobrá praxe: Vyhněte se aliasům URI

Vlastník URI BY NEMĚL spojovat libovolně různé URI se stejným zdrojem.

Spotřebitelé URI mají také úlohu v zajištění konzistence URI. Například při přepsání URI by agenti neměli bezdůvodně zakódovat znaky procentovou notací. Termín "znak" se odkazuje na znaky URI, jak byly definované v kapitole 2 [URI]; v kapitole 2.1 této specifikace se řeší toto kódování procentovou notací.

Dobrá praxe: Shodné užívání URI

Agent, který přijímá URI, BY SE MĚL odkázat na přidružený zdroj pomocí stejného URI, tedy stejné znak po znaku.

Když se alias URI stane obecnou pravdou, vlastník URI by měl používat protokolové techniky jako přesměrování na straně serveru, které se týká dvou zdrojů. Společenství prospívá, když vlastník URI podporuje přesměrování aliasu URI do odpovídajícího "oficiálního" URI. Pro více informací o přesměrování se podívejte na kapitolu 10.3, Redirection v [RFC2616]. Viz také [CHIPS] pro diskusi o některých praktikách a zásadách pro správce sítě.

2.3.2. Opětné použití reprezentace

URI aliasing nastane jen tehdy, když se více než jeden URI používá k identifikaci stejného zdroje. Skutečnost, že různé zdroje někdy mají stejnou reprezentaci, ještě nedělá k URI aliasy pro tyto zdroje.

Příběh

Dirk by si rád přidal odkaz ze své webové stránky na povětrnostní stránku v Oaxaca. Použije URI `http://weather.example.com/oaxaca` a tento odkaz si označí jako "zpráva o počasí v Oaxaca 1. srpna 2004". Nadia upozorní Dirka na to, že nastavil matoucí pojmenování pro URI, který použil. Politika stránky o počasí v Oaxaca je taková, že požadavek z URI identifikuje zpravodajství o aktuálním počasí v Oaxaca - na jakýkoliv daný den - a ne počasí na 1. srpna. Samozřejmě že prvního srpna 2004 bude Dirkův odkaz správný, ale pro zbytek času bude Dirk zmateným čtenářem. Nadia Dirka upozorní, že manažeři stránek o počasí v Oaxaca zpřístupní různé URI, permanentně přiřazené k zdroji zpravodajství o počasí na 1. srpna 2004.

V tomto příběhu nacházíme dva zdroje: "zpráva o aktuálním počasí v Oaxaca" a "zpráva o počasí v Oaxaca 1. srpna 2004". Manažeři stránek o počasí v Oaxaca přiřadí dva URI k těmto dvěma různým zdrojům. K 1. srpnu 2004 jsou reprezentace pro tyto zdroje totožné. Fakt, že získávání dvou odlišných URI produkuje identické reprezentace neznamená, že tyto dva URI jsou aliasy.

2.4. Schémata URI

V URI "`http://weather.example.com/`" zkratka "http", která se objeví před dvojtečkou (":"), se nazývá schéma URI. Každé schéma URI má určitou specifikaci, která vysvětluje specifické detaily o tom, jak jsou identifikátory schématu přiděleny a jak se přidružují ke zdroji. Syntaxe URI je tak federalizovaný a názvově rozšiřitelný systém, v němž každá specifikace schématu může dále omezit skladbu a sémantiku identifikátorů uvnitř schématu.

Příklady URI z různých schémat zahrnují:

- `mailto:joe@example.org`
- `ftp://example.org/aDirectory/aFile`
- `news:comp.infosystems.www`
- `tel:+1-816-555-1212`
- `ldap://ldap.example.org/c=GB?objectClass=one`
- `urn:oasis:names:tc:entity:xmlns:xml:catalog`

Zatímco architektura webu dovoluje definici nových schémat, uvedení nového schématu je drahé. Mnoho aspektů týkajících se zpracování URI je závislých na schématu a velké množství dislokovaného softwaru již postupuje podle známých schémat URI. Uvedení nového schématu URI vyžaduje vývoj a pokrytí nejen klientského softwaru, aby podporoval toto schéma, ale také vytvoření pomocných agentů jako jsou brány, servery proxy a vyrovnávací paměti. Podívejte se na [RFC2718], kde jsou další úvahy a náklady související s vytvářením schématu URI.

Kvůli těmto výdajům se doporučuje fakt, že jestliže existuje takové schéma URI, které splňuje potřeby aplikace, pak by návrháři měli raději použít toto schéma, než vynalézat jiné.

Dobrá praxe: Opětovné používání schémat URI

Specifikace BY MĚLA znovu použít existující schéma URI (než vytvořit nové), pokud poskytuje požadované vlastnosti identifikátorů a jejich vztahu ke zdrojům.

Uvažme náš cestovní scénář: Měl by agent poskytující informaci o počasí v Oaxaca registrovat nové schéma URI "weather" pro identifikaci zdrojů souvisejících s počasím? Pak by se mohl uveřejnit URI jako "weather://travel.example.com/oaxaca". Když softwarový agent získá takový URI, tak jestliže se pak stane, že HTTP GET je vyvolána k získání reprezentace tohoto zdroje, pak by "http" URI měl stačit.

2.4.1. Registrace schématu URI

The Internet Assigned Numbers Authority (IANA) udržuje registr [IANASchemes] mapování mezi jmény schémat URI a jejich specifikacemi. Například registr IANA ukazuje, že "http" schéma je definované v [RFC2616]. Proces pro registraci nového schématu URI je definován v [RFC2717].

Nezaregistrovaná schémata URI BY NEMĚLA BÝT používána z řady důvodů:

- Nikde není uveden žádný obecně uznávaný způsob, jak lokalizovat specifikaci schématu.
- Někdo jiný může používat schéma pro další účely.
- Nemělo by se očekávat, že univerzální software bude s URI daného schématu dělat něco užitečnějšího než porovnání URI.

Jedna zavádějící motivace pro registraci nového schématu URI je dovolit softwarovému agentovi spustit zvláštní aplikaci, když získává reprezentaci. Stejně věci může být dosaženo úsporněji, pokud se předání provede na základě typu reprezentace, a využijí se tak existující přenosové prokoly a jejich implementace.

Dokonce když agent nemůže zpracovat reprezentaci dat v neznámém formátu, může jí přinejmenším získat. Data mohou obsahovat dost informací k tomu, aby je mohl uživatel nebo uživatelův agent použít. Když agent neovládá nové schéma URI, nemůže získat reprezentaci.

Když se vytváří nový formát dat, preferovaný mechanismus k podpoření jeho rozmístění na web je typ internetového média (viz Typy reprezentací a typy internetových médií (§3.2, str. 18)). Typy médií poskytují také prostředky pro vybudování nových informačních aplikací, které jsou popsány v Budoucích směrech pro formáty dat (§4.6, str. 34).

2.5. Neprůhlednost URI

Je lákavé uhodnout povahu zdroje tím, že vyšetříme URI, který ho identifikuje. Nicméně web je navržený tak, že si agenti sdělují zdrojové informace prostřednictvím reprezentací, nikoliv pomocí identifikátorů. Obecně vzato nemůže nikdo určit typ zdrojové reprezentace vyšetřením URI daného zdroje. Například, ".html" na konci výrazu "http://example.com/page.html" neposkytuje žádnou záruku, že reprezentace identifikovaného zdroje budou obslouženy typem internetového média "text/html". Vydavatel má volnou ruku v přidělování identifikátorů a v definování jejich obsluhy. Protokol HTTP nemůže přinutit typ internetového média založeného na části cesty URI; Vlastník URI může klidně sám nastavit server tak, aby vracel reprezentaci, která používá PNG nebo jiný datový formát.

Stav zdroje se může vyvinout v průběhu doby. Vyžadovat po vlastníkově URI, aby uveřejnil vždy nový URI pro každou změnu ve stavu zdroje, by vedlo k výraznému počtu špatných odkazů. Kvůli robustnosti podporuje architektura webu nezávislost mezi identifikátorem a stavem identifikovaného zdroje.

Dobrá praxe: Neprůhlednost URI

Agenti využívající URI BY SE NEMĚLI pokusit o odvození vlastností zmíněného zdroje.

V praxi se mohou odvozování v malém počtu stát, jelikož jsou explicitně licencovaná příslušnými specifikacemi. Některá z těchto odvození naleznete v detailech o získávání reprezentace (§3.1.1, str. 17).

Příklad URI použitý v cestovním scénáři ("<http://weather.example.com/oaxaca>") nabádá čtenáře, že identifikovaný zdroj má něco společného s počasím v Oaxaca. Zpravodajství o počasí v Oaxaca by právě tak snadno mohlo být identifikované pomocí URI "<http://vjc.example.com/315>". A URI "<http://weather.example.com/vancouver>" může klidně odkazovat na zdroj "moje fotoalbum."

Na druhé straně URI "<mailto:joe@example.com>" signalizuje, že se URI odkazuje na poštovní schránku. Specifikace schématu URI "mailto" opravňuje agenty k tomu, aby mohli odvodit, že URI v této podobě identifikuje poštovní schránky.

Některá pravomocná oprávnění pro URI se dokumentují a publikují ve společných pravidlech přiřazování URI. Pro více informací o neprůhlednosti URI se podívejte na uveřejnění TAG `metaDataInURI-31` a `siteData-36`.

2.6. Identifikátory části

Příběh

Když Nadia prohlížela XHTML dokument, který obdržela jako reprezentaci zdroje identifikovaného "<http://weather.example.com/oaxaca>", zjistila, že URI "<http://weather.example.com/oaxaca#weekend>" odkazuje na část reprezentace, která zprostředkuje informaci o předpovědi na víkend. Tento URI zahrnuje identifikátor části "weekend" (řetěz za "#").

Identifikátor části v URI umožňuje nepřímou identifikaci sekundárního zdroje odkazem na hlavní zdroj a další identifikující informaci. Sekundární zdroj může být část nebo podmnožina primárního zdroje, určitý pohled na reprezentaci primárního zdroje nebo nějaký další zdroj definovaný nebo popsáný těmito reprezentacemi. Termíny "primární zdroj" a "sekundární zdroj" jsou definovány v kapitole 3.5 [URI].

Termíny "primární" a "sekundární" neomezují v tomto smyslu povahu zdroje—nejsou třídami. Slova primární a sekundární jednoduše značí, že je zde určitý vztah mezi zdroji k účelům jednoho URI: URI s identifikátorem části. Jakýkoliv zdroj může být identifikován jako sekundární. Může také být identifikován URI bez identifikátoru části a pak může být zdroj identifikovaný jako sekundární použitím rozmanitých URI. Účelem těchto termínů je umožnit diskusi o vztahu mezi takovými zdroji, nikoliv omezit povahu zdroje.

O identifikátorech části se také diskutuje v kapitole o typech médií a sémantice identifikátoru části (§3.2.1, str. 18).

Podívejte se na uveřejnění TAG `abstractComponentRefs-37`, které se týká používání identifikátorů části u názvů jmenných prostorů k identifikaci abstraktních komponent.

2.7. Budoucí směry pro identifikátory

Identifikátory webu v tomto smyslu zůstávají zatím otevřenou otázkou.

2.7.1. Internacionalizované identifikátory

Integrace internacionalizovaných identifikátorů (to znamená složených ze znaků mimo těch dovolených v [URI]) v architektuře webu je důležitou a otevřenou záležitostí. Viz uveřejnění TAG IRIEverywhere-27 pro diskusi o práci týkající se této oblasti.

2.7.2. Tvrzení, že dva URI identifikují stejný zdroj

Objevující se sémantické webové technologie, včetně "Web Ontology Language (OWL)" [OWL10], definují vlastnosti RDF jako `sameAs`, která tvrdí, že dva URI identifikují stejný zdroj nebo `inverseFunctionalProperty`, z které to také vyplývá.

3. Interakce

Komunikování se zdroji mezi agenty v síti zahrnuje URI, zprávy a data. Webové protokoly (zahrnující HTTP, FTP, SOAP, NNTP a SMTP) jsou založeny na výměně zpráv. Zpráva smí obsahovat data stejně jako metadata o zdroji (taková jako hlavičky HTTP "Alternates" a "Vary"), data zprávy a zprávu samotnou (jako je HTTP hlavička "Transfer-encoding"). A zpráva může dokonce obsahovat metadata o metadatech zprávy (například kvůli kontrole integrity zprávy).

Příběh

Nadia následuje hypertextový odkaz nazvaný jako "satellite image" očekávajíc, že získá satelitní fotografii regionu Oaxaca. Odkaz k tomuto snímku je odkaz XHTML zakódovaný jako `satellite image`. Nadiin prohlížeč analyzuje URI a určí, že její schéma je "http". Z nastavení prohlížeče se určí, jak se lokalizuje identifikovaná informace, která mohla být získána přes vyrovnávací paměť dřívějších požadavků kontaktováním prostředníka (jako je server proxy), nebo přímým přístupem k serveru identifikovaným v části URI. V našem příkladu otevře prohlížeč síťové spojení na portu 80 na serveru "example.com" a pošle zprávu "GET", která je specifikovaná protokolem HTTP, požadující reprezentaci zdroje.

Server pošle prohlížeči odezvu v podobě zprávy, opět dle protokolu HTTP. Zpráva se sestává z několika hlaviček a JPEG obrázku. Prohlížeč přečte hlavičky, dozví se z pole "Content-Type", že typ internetového média reprezentace je "image/jpeg", přečte posloupnost oktetů, které tvoří reprezentaci dat, a poskytne obrázek.

Tato kapitola popisuje principy architektury a omezení týkající se interakcí mezi agenty, včetně takových témat, jako jsou síťové protokoly a interakční styly, spolu s interakcemi mezi webem jakožto systémem, a mezi lidmi, kteří ho používají. Skutečnost, že web je vysoce rozšířený systém, ovlivňuje architektonická omezení a předpoklady interakcí.

3.1. Použití URI k zpřístupnění zdroje

Agenti mohou používat URI k zpřístupnění zmíněného zdroje; toto se nazývá získávání URI. Přístup může nabývat mnoha forem, zahrnující získávání reprezentace zdroje (například použitím HTTP GET nebo HEAD), přidáním nebo modifikováním reprezentace zdroje (například použitím HTTP POST nebo PUT, což může změnit v některých případech aktuální stav zdroje, jestliže předložené reprezentace jsou interpretovány jako instrukce sloužící tomuto účelu), a vymazáním některých nebo všech reprezentací zdroje (například použitím HTTP DELETE, což může mít v některých případech za následek zrušení zdroje samotného).

Může existovat více způsobů, jak zpřístupnit zdroj pro daný URI; z aplikačního kontextu vyplyne, jak se určí, kterou přístupovou metodu agent použije. Například by prohlížeč mohl použít HTTP GET k vyžádání reprezentace zdroje, zatímco aplikace pro kontrolu odkazů by mohla používat HTTP HEAD ke stejnému URI jednoduše proto, aby zjistila, zda je reprezentace k dispozici. Některá schémata URI předpokládají určité přístupové metody, jiná (jako je schéma URN [RFC 2141]) nikoliv. Kapitola 1.2.2 [URI] diskutuje o oddělení identifikace a interakce podrobněji. Pro více informací o vztazích mezi rozmanitými přístupovými metodami a adresovatelností URI se podívejte se na náleží TAG "URIs, Addressability, and the use of HTTP GET and POST".

Ačkoli je mnoho schémat URI (§2.4, str. 13) pojmenováno po protokolech, tak to ještě neznamená, že použití takového URI bude mít nutně za následek přístup ke zdroji pomocí pojmenovaného protokolu. Dokonce tehdy, když agent používá URI k získání reprezentace, jehož přístup by mohl vést skrz brány, servery proxy, vyrovnávací paměti a jmenné rozlišovací služby, které jsou nezávislé na protokolu přidruženým se jménem schématu.

Mnoho schémat URI definuje standardní komunikační protokol pro zkoušku přístupu k identifikovanému zdroji. Tento protokol je často základem pro přidělení identifikátorů tohoto schématu a právě tak "http" URI jsou definované v rámci HTTP serverů založených na technologii TCP. Nicméně to neznamená, že je celá interakce s takovými zdroji omezená pouze na standardní komunikační protokol. Například informační systémy na vyhledávání informací často používají služby, aby mohly komunikovat podle různých schémat URI, jako se HTTP servery proxy používají pro zpřístupnění jak k "ftp" tak k "wais" zdrojům. Servery proxy mohou také poskytnout rozšířené služby, jako anotující proxy servery, které kombinují standardní sběr informací s doplněnými metadaty k zajištění bezproblémovosti, multidimensionální pohled na zdroje použitím stejných protokolů a uživatelských agentů jako neanotovaný Web. Podobně by tak mohly být definovány budoucí protokoly, které obklopí naše současné systémy, používajíc úplně jiné interakční mechanismy, bez změny existujících schémat identifikátorů. Viz také principy ortogonálních specifikací (§5.1, str. 34).

3.1.1. Detaily o získávání reprezentace

Získávání URI zahrnuje posloupnost kroků, které jsou popsány v rozmanitých specifikacích a implementovány agentem. Následující příklad ukazuje sérii schémat, která řídí proces, když uživatelský agent následuje hypertextový odkaz (§4.4, str. 28), který je součástí SVG dokumentu. V tomto příkladu URI je "http://weather.example.com/oaxaca" a žádá uživatelského agenta, aby získal a poskytl reprezentaci identifikovaného zdroje.

1. Jelikož je URI součástí hypertextového odkazu v SVG dokumentu, první příslušná specifikace je SVG 1.1 Doporučení [SVG11]. Část 17.1 této specifikace importuje sémantiku odkazu definovanou v XLink 1.0 [XLink10]: "Vzdálený zdroj (cíl odkazu) je definovaný URI specifikovaným podle atributu XLink `href` v elementu 'a'." Specifikace SVG pokračuje dál v překladu elementu `a` zahrnující získání reprezentace zdroje, která se poznala podle atributu `href` ve jmenném prostoru XLink: "Jestliže se aktivuje spojení (kliknutím myši, vstupem z klávesnice, hlasovým příkazem atd.), pak mohou uživatelé navštívit tyto zdroje."
2. Specifikace XLink 1.0 [XLink10], která definuje atribut `href` v sekci 5.4, uvádí, že "Hodnota atributu `href` musí být odkaz URI definovaný v [IETF RFC 2396], nebo musí mít za následek odkaz URI po ukončení procedury aplikované níže."
3. Specifikace URI [URI] uvádí, že "Každý URI začíná jménem schématu, které se odkazuje na specifikaci kvůli přiřazení identifikátoru uvnitř schématu." Jméno schématu URI je v našem příkladu "http".
4. [IANASchemes] říká, že schéma "http" je definováno specifikací HTTP/1.1 (RFC 2616 [RFC2616], kapitola 3.2.2).
5. Ve smyslu SVG vyše agent žádost HTTP GET (podle kapitoly 9.3 [RFC2616]), aby získal reprezentaci.
6. Kapitola 6 [RFC2616] definuje, jak server zkonstruuje odpovídající odezvu, včetně pole 'Content-Type'.
7. Kapitola 1.4 [RFC2616] říká, že "HTTP komunikace se obvykle uskutečňuje přes spojení TCP/IP." Tento příklad se neprojevuje ani v tomto průběhu procesu ani v dalších krocích jako je určení adresy systémem DNS (Domain Name System).
8. Agent interpretuje vrácenou reprezentaci podle specifikace datového formátu, která odpovídá reprezentaci typu internetového média (§3.2, str. 18) (hodnota HTTP 'Content-Type') v příslušném IANA registru [MEDIATYPereg].

Která reprezentace je přesně získána závisí na několika faktorech, včetně toho:

1. Zda Vlastník URI vůbec zpřístupní nějaké reprezentace;
2. Zda agent posílající žádost má přístupová práva pro tyto reprezentace (viz kapitola o odkazování a řízení přístupu (§3.5.2, str. 23));
3. Jestliže vlastník URI poskytl více než jednu reprezentaci (v odlišných formátech jako HTML, PNG nebo RDF; v různých jazycích jako angličtina a španělština; nebo dynamicky převedené v závislosti na hardwarových nebo

softwarových schopností příjemce), tak výsledná reprezentace může záviset na dohodě mezi uživatelským agentem a serverem.

4. Čas žádosti; svět se mění v průběhu doby, takže reprezentace zdrojů se také pravděpodobně mohou změnit v průběhu doby.

Dejme tomu, že jsme reprezentaci úspěšně získali. Výrazná síla formátu reprezentace ovlivní, jak se přesně poskytovatel reprezentace dorozumívá s místem určení zdroje. Jestliže reprezentace komunikuje s místem určení zdroje nesprávně, tato nepřesnost nebo nejasnost může způsobit zmatek mezi uživateli o tom, co zdroj je. Jestliže různí uživatelé dosáhnou touto cestou odlišných závěrů, mohou tuto situaci interpretovat jako kolizi URI (§2.2.1, str. 11). Některé komunity, jako například ta, co pracuje na sémantickém webu, usilují o poskytnutí jakéhosi rámce pro přesné sdělování sémantiky zdroje do strojově čitelné podoby. Strojově čitelná sémantika může zmírnit některé nejasnosti, které jsou spojeny s přirozeným jazykovým popisem zdrojů.

3.2. Typy reprezentace a typy internetových médií

Reprezentace jsou data, která kódují informaci o stavu zdroje. Reprezentace nemusí nutně popisovat zdroj, nebo vykreslovat podobu zdroje, nebo reprezentovat zdroj v dalších smyslech slova "reprezentovat".

Reprezentaci zdroje můžeme poslat nebo přijmout prostřednictvím komunikačních protokolů. Tyto protokoly podle pořadí určí, jakým způsobem a které reprezentace se zprostředkují z webu. Například HTTP poskytuje přenos reprezentace jako proud oktetů (bajtů), jehož typ je určen internetovým typem média [RFC2046].

Stejně tak je důležité znovu použít existující schémata URI, kdykoliv je to možné, protože zde jsou podstatné výhody při použití oktetových bloků pro reprezentace, dokonce v neobvyklé situaci, kdy má být definované nové schéma URI a k němu přidružený protokol. Kdyby bylo například počasí v Oaxaca zprostředkované k prohlížeči Nadi jiným protokolem než HTTP, pak by software, který poskytuje formáty jako text/xhtml+xml a image/png byl stále ještě použitelný, za podmínky, že by nový protokol podporoval přenos těchto formátů. To je příklad podstaty ortogonálních specifikací (§5.1, str. 34).

Dobrá praxe: Opětovné používání formátů reprezentace

Nové protokoly určené pro web BY MĚLY přenášet reprezentace jako proud oktetů, jejichž typ je určen internetovým typem média.

Mechanismus typů internetových médií má některá omezení. Například typ média řetězec nepodporuje verzování (§4.2.1, str. 25) nebo další parametry. Podívejte se na uveřejnění TAG uriMediaType-9 a mediaTypeManagement-45, které se týkají aspektů mechanismu typu média

3.2.1. Typy reprezentace a sémantika identifikátoru části

Typ internetového média definuje syntaxi a sémantiku identifikátoru části (představený v kapitole Identifikátory části (§2.6, str. 15)), jestliže některý z nich může být použit ve spojení s reprezentací.

Příběh

Na jedné ze svých XHTML stránek vytváří Dirk hypertextový odkaz na obrázek, který Nadia publikovala na Webu. Vytvoří hypertextový odkaz `Nadiin klobouk`. Emma se ve svém prohlížeči dívá na Dirkovu XHTML stránku a následuje odkaz. HTML implementace v jejím prohlížeči odstraní identifikátor části z URI a žádá obrázek "http://www.example.com/images/nadia". Nadia vystavuje SVG reprezentaci obrázku (s typem internetového média "image/svg+xml"). Emmin internetový prohlížeč spustí SVG implementaci, aby se mohla na obrázek dívat, a předá původní URI včetně identifikátoru části "http://www.example.com/images/nadia#klobouk". To způsobí, že se zobrazí jen klobouk, nikoliv celý obrázek.

Všimněte si, že HTML implementace v Emmině prohlížeči nepotřebovala *rozumět syntaktické nebo sémantické* části SVG (ani SVG implementace nemusí rozumět HTML, WebCGM, RDF ... syntaktické nebo sémantické stránce; šlo jen o to poznat oddělovač # od syntaxe URI [URI] a odstranit zbytek při zpřístupnění zdroje). Tato ortogonalita (§5.1, str. 34) je důležitým rysem architektury webu; je to to, co dovolilo Emmině prohlížeči poskytnout vhodnou službu bez požadování aktualizace.

Sémantická stránka identifikátoru části se definuje množinou reprezentací, které by měly být výsledkem požadavku na primární zdroj. Rozlišení a formát identifikátoru části jsou proto závislé na typu potenciálně získané reprezentace, třebaže se takový požadavek pouze vykoná, když se získala URI. Jestliže žádná taková reprezentace neexistuje, potom se sémantika identifikátoru části považuje za neznámou a fakticky, unconstrained. Sémantika identifikátoru části je ortogonální podle schémat URI, a tak se nemusí předefinovat specifikaci schématu URI.

Interpretace identifikátoru části se vykoná výhradně agentem, který získává URI; identifikátor části během dotazu k dalším systémům dál neprojde. To znamená, že se někteří prostředníci architektury webu (jako servery proxy) nemohou dorozumět identifikátorem části, a přeměrování (například v HTTP [RFC2616]) neodpovídá za části za identifikátorem.

3.2.2. Identifikátory části a domlouvání obsahu

Domlouvání obsahu se odkazuje na praxi při tvoření dostupných rozmanitých reprezentací pomocí stejných URI. Domlouvání mezi žádajícím agentem a serverem určuje, která z reprezentací se použije (obvykle s cílem poskytnout tu "nejlepší", kterou agent může zpracovat). HTTP je příklad protokolu, který zpřístupní reprezentaci poskytovatele pomocí domlouvání obsahu.

Jednotlivé formáty dat si mohou definovat svá vlastní pravidla pro použití syntaxe identifikátoru části, kvůli specifikování různých druhů podmnožin, pohledů nebo externích adres, které jsou typem média identifikovatelné jako sekundární zdroje. Tudiž musí poskytovatelé reprezentace řídit pečlivě domlouvání obsahu, když se používá s URI, která obsahuje identifikátor části. Uvažme příklad, kdy majitel URI "http://weather.example.com/oaxaca/map#zicateľa" používá domlouvání obsahu, aby mohl používat dvě reprezentace identifikovaného zdroje. Mohou nastat tři situace:

1. Interpretace "zicateľa" je definována konzistentně oběma specifikacemi formátu dat. Poskytovatel reprezentace se rozhodne, kdy se definice sémantiky identifikátoru části dostatečně shodují.
2. Interpretace "zicateľa" je definována nekonzistentně i specifikacemi formátu dat.
3. Interpretace "zicateľa" je definována v jedné ze specifikací formátu dat, ale v žádné další.

V první situaci nenastane žádný problém.

V druhém případě server ohlásí chybu: poskytovatelé reprezentace nesmějí používat domlouvání obsahu k zobrazení formátů reprezentace, které mají nekonzistentní specifikace formátu dat. Tato situace také vede ke kolizi URI (§2.2.1, str. 11).

V třetím případě server chybu neohlásí. To je způsob, kterým Web může narůst. Protože web je natolik rozšířený systém, v kterém se formáty a agenti rozmísťují v nejednotném způsobu, architektura webu nepřinutí autory, aby používali jen formáty "nejbližšího společného jmenovatele". Autoři obsahu mohou využít nové formáty dat, ale může se stát, že ještě nebude zajištěna zpětná kompatibilita pro agenty, kteří formáty neimplementují.

V třetím případě by se chování přijímajícího agenta mělo měnit v závislosti na tom, zda dohodnutý formát definuje sémantiku identifikátoru části. Pokud jí přijatý formát nedefinuje, agent by neměl vykonat tiché zotavení po chybě dokud nedá uživatel souhlas; Viz [CUAP] pro další navrhované chování agenta týkající se tohoto případu.

Podívejte se na související uveřejnění TAG RDFinXHTML-35.

3.3. Rozporuplnosti mezi reprezentacemi dat a metadat

Úspěšná komunikace mezi dvěma stranami závisí na rozumně sdíleném porozumění sémantik z vyměněných zpráv, jak u dat tak u metadat. Občas se mohou vyskytnout neslučitelnosti mezi odesílatelskou zprávou dat a metadat. Příklady neslučitelností z praxe mezi reprezentací dat a metadat zahrnují:

- Aktuální kódování znaků reprezentace (například "iso-8859-1", specifikované v atributu `encoding` v deklarační části XML) je v rozporu s parametrem znakové sady v reprezentaci metadat (například "utf-8" specifikované polem 'Content-Type' v hlavičce HTTP).
- Jmenný prostor (§4.5.3, str. 30) kořenového elementu reprezentace XML dat (například, jak jsou specifikovány atributem "xmlns") je v rozporu s hodnotou pole 'Content-Type' v hlavičce HTTP.

Na druhé straně například není neslučitelnost v zobrazení obsahu HTML s typem média "text/plain", protože je tato kombinace povolena specifikacemi.

Přijímající agenti by měli zjistit neslučitelnosti protokolu a vykonat řádné zotavení po chybě.

Omezení: Neslučitelnost ve vztahu data - metadata

Agenti NESMĚJÍ ignorovat zprávu metadat bez souhlasu uživatele.

Tak například, jestli strana odpovědná za "weather.example.com" mylně označí satelitní fotografii Oaxaca jako "image/gif" namísto "image/jpeg", a jestliže Nadiin prohlížeč zjistí problém, nesmí Nadiin prohlížeč ignorovat problém (například jednoduše překladem obrázku JPEG) bez Nadiina souhlasu. Nadiin prohlížeč jí problém může oznámit, nebo oznámit a učinit opravné opatření.

Dále ještě mohou poskytovatelé reprezentace pomoci tím, že redukují riziko neslučitelnosti opatrným přiřazením reprezentace metadat (zvláště tam, kde se prochází skrz reprezentaci). Kapitola o typech médií pro XML (§4.5.7, str. 33) ukazuje příklad snížení rizika chyby neposkytnutím metadat o kódování znaků při zobrazování XML.

Přesnost metadat závisí na správci sítě, autorech reprezentací a na softwaru, který se používá. Schopnosti nástrojů a společenské vztahy mohou být prakticky omezujícími faktory.

Přesnost těchto a dalších metadat je stejně tak důležitá pro dynamické webové zdroje, kde trochu přemýšlení a programování může často zabezpečit správná metadata pro obrovské množství zdrojů.

Často dochází k oddělení kontroly mezi uživateli, kteří vytvářejí reprezentace zdrojů, a správci serveru, kteří udržují webově orientovaný software. Jestliže je zde obecně nějaký webově orientovaný software, který poskytuje metadata přidružená se zdrojem, tak z toho vyplývá, že se požaduje koordinace mezi správci serveru a tvůrci obsahu.

Dobrá praxe: Asociace metadat

Správci serveru BY MĚLI dovolit tvůrcům reprezentace kontrolovat metadata přidružená s jejich reprezentacemi.

Zvláště tvůrci obsahu potřebují, aby mohli ovládat formu obsahu (kvůli rozšiřitelnosti) a kódování znaků (pro řádnou internacionalizaci).

Nález TAG "*Authoritative Metadata*" řeší podrobněji, jak se manipuluje s daty/metadaty neslučitelně, a jak se tomu lze vyvarovat vhodnou konfigurací serveru.

3.4. Bezpečné interakce

Nadino získání povětrnostní informace (příklad read-only dotazu nebo vyhledání) je ohodnoceno jako "bezpečná" interakce; Bezpečná interakce je taková, kde si agent nezpůsobí žádný závazek z interakce. Agent si může závazek způsobit i dalšími prostředky (jako podepsáním kontraktu). Jestliže agent nemá závazek před bezpečnou interakcí, nemá ho ani později.

Další webové interakce se podobají spíše objednávkám než dotazům. Tyto nebezpečné interakce mohou způsobit změnu na úložišti zdroje a uživatel může nést odpovědnost za důsledky těchto interakcí. Nebezpečné interakce zahrnují předplacení informačního bulletinu, odeslání katalogu nebo modifikování databáze. **Poznámka:** V této souvislosti slovo "nebezpečný" nutně neznamená "životu nebezpečný"; Termín "bezpečný" se používá v kapitole 9.1.1 [RFC2616] a "nebezpečný" je přirozený opak.

Příběh

Nadia se rozhodne zamluvit si dovolenou v Oaxaca na "booking.example.com." Zadá vstupní data do políček online formuláře a nakonec je zažádána o číslo kreditní karty, aby si mohla koupit letenky. Číslo napíše do dalšího políčka formuláře. Jakmile stiskne tlačítko "koupit", její prohlížeč otevře další síťové spojení se serverem "booking.example.com" a pošle zprávu složenou z dat uvedených ve formuláři za pomoci metody POST. Toto je nebezpečná interakce; Nadia si přeje změnit stav tím, že vymění peníze za letenky.

Server si přečte požadavek POST a po vykonání transakce k rezervaci vrátí zprávu k Nadině prohlížeči, která obsahuje reprezentaci výsledků Nadině žádosti. Reprezentace dat je v XHTML, takže ji lze uložit nebo vytisknout, pokud by si to Nadia chtěla poznamenat.

Všimněte si, že ani data přenesená požadavkem POST ani přijatá data z odezvy nemusí nutně korespondovat s jakýmkoliv zdrojem identifikovaným URI.

Bezpečné interakce jsou důležité, protože jsou to interakce, kde uživatelé mohou brouzdat s důvěrou a kde agenti (včetně vyhledávacích strojů a prohlížečů, kteří zálohuji data pro uživatele) mohou bezpečně následovat hypertextové odkazy. Uživatelé (nebo agenti jednající v jejich zájmu) se sami nezavážou k něčemu, co by se ptalo na zdroj nebo co by mělo následovat hypertextový odkaz.

Zásada: Bezpečné získání

Agenti si nezpůsobí závazky získáním reprezentace.

Například je nesprávné uveřejnit URI který, když se následuje jako součást hypertextového odkazu, připiše uživatele do mailing listu. Pamatujte si, že vyhledávací stroje mohou takovéto odkazy následovat.

Skutečnost, že metoda HTTP GET, která se nejčastěji používá pro přístup při následování hypertextového odkazu, je bezpečná, ještě neznamená, že se na všechny bezpečné interakce musí použít HTTP GET. Občas se totiž mohou najít dobré důvody (jako důvěrnost požadavků nebo praktické omezení délky URI) provádět jinak bezpečný provoz použitím mechanismu, který se obecně používá pro nebezpečné operace (například HTTP POST).

Pro více informací o bezpečných a nebezpečných operacích za použití HTTP GET a POST a o zajišťování bezpečnostních zájmů kolem použití HTTP GET se podívejte se na nálezný TAG *"URIs, Addressability, and the use of HTTP GET and POST"*.

3.4.1. Nebezpečné interakce a odpovědnost

Příběh

Nadia platí za své letenky online (přes interakci POST, jak je popsáno výše). Obdrží webovou stránku s potvrzovací informací a přeje si vytvořit záložku, čímž se na tuto informaci může odkázat tehdy, až si bude vypočítávat své výlohy. Třebaže si Nadia může výsledky vytisknout, uložit si je do souboru, ráda by si je i chtěla uložit jako záložku.

Odeslané žádosti a výsledky transakcí jsou důležité zdroje a jako na všechny takovéto cenné zdroje je užitečné, abychom se na ně mohli odkázat stálým URI (§3.5.1, str. 22). Nicméně v praxi si Nadia nemůže do svých oblíbených položek založit ani svůj platební závazek (odeslaný přes požadavek POST) ani potvrzení letecké společnosti o zmluveném letu (přijatý jako výsledek této POST žádosti).

Zde jsou způsoby, jak poskytnout trvalé URI pro transakční žádosti a pro jejich výsledky. Pro transakční žádosti mohou uživatelští agenti poskytnout rozhraní pro řídicí transakce, kde uživatelský agent vytváří závazek jménem uživatele. Pro transakční výsledky HTTP dovoluje poskytovatelům reprezentace spojit URI s výsledkem žádosti HTTP POST za použití hlavičky "Content-Location" (popsaná v kapitole 14.14 [RFC2616]).

3.5. Správa reprezentací

Příběh

Od té doby, co Nadia považuje stránky o počasí v Oaxaca za užitečné, emailuje své dojmy příteli Dirkov, kde mu doporučuje, aby vyzkoušel 'http://weather.example.com/oaxaca'. Dirk klikne na výsledný hypertextový odkaz v emailu, který obdržel, a je frustrovaný zprávou 404 (nenalezeno). Dirk to zkusí znovu další den a přijímá reprezentaci se "zprávami", které jsou dva týdny staré. Zkusí to ještě jednou další den, kdy obdrží jen reprezentaci, která tvrdí, že počasí v Oaxaca je slunečné, třebaže mu jeho přátelé v Oaxaca řeknou do telefonu, že ve skutečnosti prší. Dirk a Nadia usoudí, že majitelé URI jsou nespolehliví nebo nepředvídatelní. Ačkoli si vlastník URI vybral Web jako komunikační médium, ztratil tak dva zákazníky kvůli neúčinné správě reprezentací.

Vlastník URI může nabízet nulu a více spolehlivých reprezentací zdroje identifikovaného URI. Což je výhoda pro veřejnost.

Dobrá praxe: Dostupná reprezentace

Vlastník URI BY MĚL poskytnout reprezentace zdroje, který se identifikuje.

Například vlastníci URI jmenného prostoru XML by jej měli používat k identifikování dokumentu jmenného prostoru (§4.5.4, str. 31).

Jen proto, že reprezentace jsou dostupné, ještě neznamená, že je vždy žádoucí je získat. Ve skutečnosti je v některých případech opak pravdou.

Zásada: Odkaz neznamená získání

Vývojář aplikací nebo autor specifikací BY NEMĚL vyžadovat síťové získání reprezentací, kdykoliv je o nich zmínka.

Získávání URI má (potenciálně významnou) výpočetní náročnost a nároky na přenosové pásmo, může způsobit bezpečnostní důsledky a může způsobit značné zpoždění při dotazování aplikace. Získávání URI bychom se měli vyhnout kromě situací, kdy je to nezbytné.

Následující kapitola diskutuje o některých aspektech správy reprezentací zahrnující propagaci stálosti URI (§3.5.1, str. 22), řízení přístupu ke zdrojům (§3.5.2, str. 23) a doprovodnou navigaci (§3.5.3, str. 23).

3.5.1. Stálost URI

Tak jako v případě lidských interakcí, důvěra v interakce na webu spočívá ve stabilitě a schopnosti předpovídat. Pro informační zdroj závisí stálost na shodě reprezentací. Poskytovatel reprezentace se rozhodne, kdy se reprezentace dostatečně shodují (ačkoli toto rozhodnutí obvykle bere uživatelská očekávání v úvahu).

Ačkoli se stálost dá v tomto případě vyzorovat následkem získání reprezentace, termín stálost URI se používá k popisu žádoucí vlastnosti, která říká, že jakmile se URI přidruží se zdrojem, tak by se měla na neurčitou dobu stále odkazovat na ten samý zdroj.

Dobrá praxe: Shodná reprezentace

Vlastník URI BY MĚL poskytnout reprezentace identifikovaného zdroje shodně a předvídatelně.

Stálost URI je věcí politiky a zavázání se ze strany vlastníka URI. Volba speciálního schématu URI neposkytuje žádnou záruku, že tyto URI budou nebo nebudou stálé.

Protokol HTTP [RFC2616] byl navržen k tomu, aby pomohl řídit stálost URI. Například přesměrování HTTP (používá 3xx kódy odezvy) povoluje serverům, aby řekli agentovi, že na další akci je potřeba agenta za účelem splnění žádosti (například, když je nový URI přidružený se zdrojem).

Navíc domlouvání obsahu také podporuje shodu, zatímco na správci webu se nepožaduje, aby definoval nové URI, když přidává podporu pro novou specifikaci formátu. Protokoly, které nepodporují domlouvání obsahu (jako FTP), vyžadují nový identifikátor, když se objeví nový datový formát. Nevhodné použití domlouvání obsahu může vést k rozporuplným reprezentacím.

Pro další diskusi o stálosti URI se podívejte na [Cool].

3.5.2. Odkazování a řízení přístupu

Je rozumné omezit přístup ke zdroji (například z komerčních nebo bezpečnostních důvodů), ale pouze identifikování zdroje je přesný návod, kde hledat ty správné informace, jako je odkázání se na knihu podle titulu. Za výjimečných okolností se mohou lidé dohodnout na držení titulu nebo URI v tajnosti (Například autor knihy a vydavatel mohou souhlasit s ponecháním URI stránky, která obsahuje další tajný materiál, který by neměl být zveřejněn, dokud kniha nevyjde), jinak je mohou volně zveřejňovat.

A obdobně: Majitelé budovy by mohli mít takovou politiku, že veřejnost může vstoupit do budovy pouze hlavním vchodem a jen v pracovních hodinách. Lidé, kteří pracují v budově a kteří doručují zásilky mohou přiměřeně užívat i jiné dveře. Taková politika by si vynutila kombinaci bezpečnostního personálu a mechanických zařízení jako jsou zámky a přístupové karty. Nikdo by tuto politiku neprosadil skrýváním některých vchodů do budovy, ani podle přání legislativy, která požaduje použít hlavní vchod a zakazovat komukoliv, kdo odhalí skutečnost, že do budovy vedou další dveře, aby to někomu říkal.

Příběh

Nadia pošle Dirkovu URI aktuálního článku, který právě čte. Se svým prohlížečem Dirk následuje hypertextový odkaz a je vyzván, aby zadal uživatelské jméno a heslo účastníka předplatitele. Od té doby, co je Dirk také předplatitelem služeb, které poskytuje "weather.example.com," má přístup ke stejným informacím jako Nadia. Tímto oprávněním pro "weather.example.com" se může omezit přístup jen schváleným stranám a stále ještě poskytovat výhody URI.

Web poskytuje několik mechanismů k ovládní přístupu ke zdrojům; tyto mechanismy se nespolehlivě spoléhají na skrývání nebo zamlčování URI pro tyto zdroje. Pro více informací se podívejte na náleží TAG *"Deep Linking' in the World Wide Web"*.

3.5.3. Doprovodná navigace

Silnou stránkou architektury webu je, že lze odkazy vytvořit a sdílet; A uživatel, který našel zajímavou část webu, může sdílet tuto zkušenost zase jen znovu publikováním URI.

Příběh

Nadia a Dirk chtějí navštívit Muzeum předpovědi počasí v Oaxaca. Nadia jde na adresu "http://maps.example.com", lokalizuje muzeum a pošle Dirkovu URI "http://maps.example.com/oaxaca?lat=17.065;lon=-96.716;scale=6". Dirk jde na "http://mymaps.example.com", lokalizuje muzeum a pošle Nadě URI "http://mymaps.example.com/geo?sessionID=765345;userID=Dirk". Dirk čte Nadin email a může následovat odkaz na mapu. Nadia čte Dirkův email, následuje odkaz a přijímá chybovou zprávu 'No such session/user' (nenalezen takovýto uživatel nebo jeho session ID). Nadia musí začít znovu z "http://mymaps.example.com" a nalézt umístění muzea ještě jednou.

Pro zdroje, které jsou vygenerované na požádání, je strojová generace URI běžná. Pro zdroje, které by mohly být založeny pro pozdější pročení, nebo sdílené s jinými lidmi, by se správci serverů měli vyhnout zbytečně omezující

znovupoužitelnosti takových URI. Jestliže záměrem je omezit informaci nějakému uživateli, jako by mohl být třeba případ vyřizování home banking aplikací, pak by měli návrháři používat vhodné mechanismy pro řízení přístupu (§3.5.2, str. 23).

Interakce zařízená HTTP POST (kde se mohl použít HTTP GET) také omezuje navigační možnosti. Uživatel nemůže vytvořit záložku nebo sdílet URI, protože transakce HTTP POST nemá typicky za následek různý URI tak, jak uživatel vzájemně působí se stránkou.

3.6. Budoucí směry pro interakci

Webové interakce v tomto smyslu zůstávají zatím otevřenou otázkou. TAG očekává budoucí verze tohoto dokumentu, které by oslovily podrobněji vztah mezi architekturou zde popisovanou, webovými službami, peer-to-peer systémy, instant messaging systémy (jako [RFC3920]), streamovaným audiem (jako RTSP [RFC2326]) a voice-over-IP (jako SIP [RFC3261]).

4. Formáty dat

Specifikace formátu dat (například pro XHTML, RDF/XML, SMIL, XLink, CSS a PNG) představují dohodu na správném výkladu reprezentace dat. První formát dat, který se na webu použil, byl HTML. Od té doby se formáty dat bezpečně rozrostly. Architektura webu nemůže přinutit poskytovatele obsahu, které formáty dat mají použít. Tato různorodost je důležitá, protože probíhá neustálý vývoj aplikací, což má za následek vzniku nových formátů dat a zdokonalování existujících formátů. Ačkoli architektura webu počítá s rozmístěním nových formátů dat, vytváření a rozmístování nových formátů (a schopnost agentů je ovládat) je náročná a drahá věc. Takže před vytvořením nového formátu dat (nebo "meta" formátu jako je XML) by návrháři měli pečlivě uvážit znovupoužití toho formátu, který je již dostupný.

U datového formátu, který by byl schopen užitečně spolupracovat mezi dvěma stranami, se musí strany dohodnout (v rozumném rozsahu) o jeho syntaxi a sémantice. Vzájemná shoda formátu dat podporuje kompatibilitu, ale neomezuje používání. Například odesílatel dat nemůže počítat s tím, že omezí chování příjemce.

Níže popisujeme některé charakteristické rysy formátu dat, které usnadňují integraci do architektury webu. Tento dokument neřeší obecně prospěšné charakteristické rysy specifikace jako čitelnost, jednoduchost, upozornění na programátorské cíle, potřeby uživatelů, přístupnost ani internacionalizaci. Kapitola architektonické specifikace (§7.1, str. 42) zahrnuje odkazy na další specifikační směrnice formátu.

4.1. Binární a textové formáty dat

Binární formáty dat jsou ty, v kterých jsou části dat zakódované pro přímé použití počítačovými procesory, například 32 bitový little-endian dvojkový doplněk a 64 bit IEEE double-precision floating-point. Takto reprezentované části dat zahrnují číselné hodnoty, ukazatele a komprimovaná data všech druhů.

Textový formát dat je takový, v kterém jsou data zakódovaná jako posloupnost znaků. HTML, e-mail a všechny formáty dat založené na XML (§4.5, str. 33) jsou textové. Internacionalizované textové formáty dat se stále více odkazují pro znakové definice na Unicode [UNICODE].

Jestliže je formát dat textový tak, jak je definován v této kapitole, tak to neznamená, že by měl být prezentován typem média se začátkem "text/". Ačkoli formáty dat založené na XML jsou textové, mnoho těchto XML formátů se primárně nesestává z frází přirozeného jazyka. Podívejte se na kapitolu o typech médií pro XML (§4.5.7, str. 33) kvůli problémům, které vyvstávají tehdy, když se používá "text/" ve spojení s formáty dat založenými na XML.

V zásadě se mohou všechna data reprezentovat pomocí textových formátů. V praxi se ale některé druhy obsahu (například audio a video) obecně reprezentují binárními formáty.

To, proč je někdy lepší použít, nebo že se upřednostní binární či textový formát dat, je komplexnější otázkou a také záleží na závislosti aplikace. Binární formáty mohou být podstatně více kompaktní zvláště pro komplexní datové struktury bohaté na ukazatele. Také se dají rychleji zpracovat agenty v těch případech, kde se mohou načíst do paměti

a použit za malé nebo žádné konverze. Nicméně si všimněte, že tyto případy nejsou většinou běžné, protože takové přímé použití může zpřístupnit bezpečnostní problémy, které se dají prozkoumat pouze důkladným rozbořením struktury dat.

Textové formáty jsou obvykle lépe přenosné a schopné spolupracovat. Mají také značnou výhodu, že je mohou lidé rovnou přečíst (a porozumět s dodanou dokumentací). Což může zjednodušit úkoly při vytváření a udržování softwaru a dovolí přímý zásah lidí, kteří na těchto úkolech pracují, aniž by se museli vracet k vývojářským nástrojům, které jsou daleko komplexnější než všudypřítomný textový editor. Konečně se také zjednodušuje nezbytný lidský úkol učít se nové formáty dat; toto se nazývá efekt "viděného zdroje".

Je důležité zdůraznit, že intuice není v případě velikosti a zpracování dat spolehlivým "průvodcem" v návrhu formátu dat; kvantitativní studie jsou nezbytné pro pochopení správného výběru formátu. Proto by měli návrháři specifikace datového formátu rozumě vybrat mezi návrhem binárního a textového formátu.

Viz uveřejnění TAG binaryXML-30.

4.2. Verzování a rozšiřitelnost

V dokonalém světě by jazykoví návrháři rádi vynalezli jazyky, které by vyhovovaly přesně jejich požadavkům), tyto požadavky by byly perfektním modelem světa, nikdy by nebylo potřeba je měnit a všechny implementace by byly schopné dokonale spolupracovat, protože specifikace by neměly žádnou proměnlivost.

Ve skutečném životě je vše naopak, tedy nic není tak perfektní a je potřeba provádět neustálé změny. V důsledku toho se návrháři musejí dohodnout s uživateli, dělají kompromisy a často zavádějí rozšiřitelné mechanismy, aby se problémy vyřešily v co nejkratší době. Dlouhodobě produkují rozmanité verze svých jazyků, podle toho jak se problém a jejich chápání problému vyvíjí. Vyplyvající proměnlivost v specifikacích, jazycích a v implementacích představuje součinné výdaje.

Rozšiřitelnost a verzování jsou strategie, které pomáhají řídit přirozený vývoj informací na webu a technologie, které se používají k reprezentaci těchto informací. Pro více informací o tom, jak tyto strategie představují proměnlivost a jak tato proměnlivost dopadá na součinnost, se podívejte na Variability in Specifications.

Podívejte se také na uveřejnění TAG XMLVersioning-41, který se týká "dobrých praxí" při návrhu rozšiřitelného jazyka XML, a pro zajištění verzování. Podívejte se také na "Web Architecture: Extensible Languages" [EXTLANG].

4.2.1. Verzování

Existuje typicky (dlouhé) přechodné období, během kterého se používají současně rozmanité verze formátu, protokolu nebo agenta.

Dobrá praxe: Verze informace

Specifikace formátu dat BY MĚLA poskytovat informace o verzi.

4.2.2. Verzování a politika jmenného prostoru XML

Příběh

Nadia a Dirk vytváří XML formát dat, aby zakódovali data o filmovém průmyslu. Poskytují rozšiřitelnost použitím jmenných prostorů XML a vytvořením schématu, které dovolí zahrnout v určitých místech elementy z nějakého jmenného prostoru. Kdy svůj formát korigují, Nadia navrhne nový nepovinný atribut `lang` elementu `film`. Dirk cítí, že taková změna vyžaduje přiřadit nový název jmenného prostoru, který by mohl vyžadovat změny rozmístěného softwaru. Nadia vysvětlí Dirkově, že jejich volba strategie rozšiřitelnosti spolu s jejich politikou jmenného prostoru připouští jisté změny, které neovlivní existující obsah a software, a tudíž není potřeba jakákoliv změna v identifikátoru jmenného prostoru. Vybrali si tuto politiku, aby jim pomáhala vyhovět jejich záměru snížit náklady ze změn.

Dirk a Nadia si vybrali zvláštní politiku změny jmenného prostoru, která jim umožní se vyhnout změně jmenného prostoru, kdykoli učiní změny, které neovlivní shodu rozmístěného obsahu a softwaru. Možná mohli vybrat jinou politiku, například, že nějaký nový element nebo atribut musí patřit jinému jmennému prostoru, než je ten originální. Kterákoliv zvolená politika by měla nastavit jasná očekávání pro uživatele formátu.

Obečně změna elementu jmenného prostoru kompletně mění jméno elementu. Jestliže "a" a "b" se vážou k dvěma různými URI, `a:element` a `b:element`, jsou stejně tak rozdílné jako `a:eieio` a `a:xyzy`. Prakticky řečeno to znamená, že rozmístěné aplikace se budou muset aktualizovat, aby mohly poznat nový jazyk; náklady této aktualizace mohou být velmi vysoké.

Z toho vyplývá, že je potřeba zdůraznit správnou volbu, když se rozhoduje o politice změn jmenného prostoru. Jestliže slovník neobsahuje žádné body rozšiřitelnosti (to jest, jestliže nedovoluje elementy nebo atributy z cizích jmenných prostorů nebo má mechanismus pro jednání s nerozpoznanými jmény ze stejného jmenného prostoru), může být absolutně nezbytné změnit název jmenného prostoru. Jazyky, které dovolí nějakou formu rozšiřitelnosti bez požadování změny jmenného prostoru, se pravděpodobněji budou vyvíjet "uhlazeněji."

Dobrá praxe: Politika jmenného prostoru

Specifikace formátu XML BY MĚLA zahrnovat informaci o politice změn pro jmenné prostory XML.

Jako příklad změněné politiky navržené k tomu, aby odrážela proměnnou stabilitu jmenného prostoru, uvažujme W3C namespace policy pro dokumenty W3C Recommendation track. Politika očekává, že "Working Group" odpovědná za jmenný prostor, ho může modifikovat jakýmkoliv způsobem až do jistého bodu ("Candidate Recommendation"), v kterém W3C vyžaduje soubor možných změn k jmennému prostoru, za účelem podporovat stabilní implementace.

Všimněte si, že názvy jmenných prostorů jsou URI, vlastník jmenného prostoru URI má oprávnění se rozhodnout změnit politiku pro jmenný prostor.

4.2.3. Rozšiřitelnost

Požadavky se mění v průběhu doby. Úspěšné technologie jsou přizpůsobeny a osvojeny novými uživateli. Návrháři mohou usnadnit přechodný proces tím, že vyberou pečlivé volby o rozšiřitelnosti během navrhování jazykové nebo protokolové specifikace.

Při výběru těchto voleb musejí zvážit kritéria výběru mezi rozšiřitelností, jednoduchostí a proměnlivostí. Jazyk bez mechanismů rozšiřitelnosti může být jednodušší a méně variabilní, zlepšující počáteční kompatibilitu. Nicméně je pravděpodobné, že změny v tomto jazyku budou obtížnější, možná více komplexní a více proměnné, než počáteční návrh poskytující takové mechanismy. Což může snížit kompatibilitu v dlouhém období.

Dobrá praxe: Mechanismy rozšiřitelnosti

Specifikace BY MĚLA poskytovat mechanismy, které dovolí jakékoliv straně vytvořit rozšíření.

Specifikace BY MĚLA poskytovat mechanismy, které dovolí jakékoliv straně vytvořit rozšíření. Rozšiřitelnost představuje proměnlivost, která má dopad na součinnost. Nicméně jazyky, které nemají žádné rozšiřovací mechanismy, mohou být rozšířeny ad hoc, což také dopadá na součinnost. Jedno klíčové kritérium mechanismů, které je poskytováno jazykovými návrháři je, že dovolí rozšiřitelným jazykům zůstat v souladu s originální specifikací, zvýšením pravděpodobnosti součinnosti.

Dobrá praxe: Shoda rozšiřitelnosti

Rozšiřitelnost NESMÍ kolidovat se souladem s originální specifikací.

Aplikační potřeby určí nejvíce vhodnou rozšiřující strategii pro specifikaci. Například aplikace určené k tomu, aby operovaly v izolovaném prostředí mohou dovolit návrhářům specifikaci definovat verzovací strategii, která by byla nepraktická v prostředí webu.

Dobrá praxe: Neznámá rozšíření

Specifikace BY MĚLA specifikovat chování agenta, který se setká tváří v tvář s neznámým rozšířením.

Jsou vyvinuty dvě strategie, které jsou zvláště užitečné:

1. "Musíš ignorovat": Agent ignoruje jakýkoliv obsah, který nerozpozná.
2. "Musíš rozumět": Agent zachází s nerozpoznaným označením jako s chybovým stavem.

Silná přístupová forma znamená pro jazyk buď poskytnutí způsobu rozšíření, nebo zřetelné rozeznání syntaxí mezi nimi.

Další strategie zahrnují nabádání uživatele k zadávání obsáhlejšího vstupu a k automatickému získávání dat z dostupného hypertextového odkazu. Jsou také možné i komplexnější strategie zahrnující smíšené strategie. Například jazyk může zahrnovat mechanismy pro hlavní standardní chování. Takže formát dat může specifikovat sémantiku "musíš ignorovat," ale také počítá s rozšířeními, která přehlíží tuto sémantiku vzhledem k potřebám aplikace (například sémantika "musíš rozumět" pro zvláštní rozšíření).

Rozšiřitelnost není volná. Poskytnutím míst pro rozšiřitelnost je jedním z mnoha požadavků, který ovlivňuje náklady jazykového návrhu. Zkušenost ukazuje, že i když jsou počáteční výdaje vyšší, dlouhodobé výhody z dobře navržených mechanismů rozšiřitelnosti obecně výdaje převáží.

Viz D.3 Extensibility and Extensions v [QA].

4.2.4. Skládání formátů dat

Mnoho moderních formátů dat zahrnuje mechanismy pro skládání. Například:

- Je možné vložit textové komentáře do některých obrázkových formátů, jako JPEG/JFIF. Ačkoli jsou tyto komentáře zaraženy v obsažených datech, není úmyslem ovlivnit ukázkou obrázku.
- Existují zásobníkové formáty jako SOAP, které plně očekávají obsah z rozmanitých jmenných prostorů, ale které poskytují celkový sémantický vztah obálky zprávy a jejího obsahu.
- Sémantika kombinovaných RDF dokumentů obsahujících přesně definované rozmanité slovníky.

V zásadě mohou tyto vztahy být libovolně smíšené a vložené. Například SOAP zpráva může obsahovat obrázek SVG, který obsahuje RDF komentář, který se odkazuje na slovník podmínek popisujících obrázek.

Nicméně si všimněte, že pro obecné XML není žádný sémantický model, který definuje interakce uvnitř XML dokumentů s elementy a/nebo atributy z různých jmenných prostorů. Každá aplikace musí definovat, s jakým jmenným prostorem vzájemně působí a jaký účinek elementu jmenného prostoru mají jeho předci, sourozenci a potomci.

Viz uveřejnění TAG mixedUIXMLNamespace-33 (vztahující se k významu dokumentu složeného z obsahu z různorodých jmenných prostorů), xmlFunctions-34 (vztahující se k jednomu přístupu pro řízení transformací a skládání XML) a RDFinXHTML-35 (vztahující se k výkladu RDF, který je zařazen do XHTML dokumentu).

4.3. Oddělení obsahu, prezentace a interakce

Web je heterogenní prostředí, kde široká pestrost agentů poskytuje přístup uživatelům k obsahu s celou škálou různorodých schopností. Dobrou praxí autorů je vytvářet obsah, který může oslovit nejširší možné publikum, zahrnující uživatele s grafickými stolními počítači, přenosnými zařízeními a s mobilními telefony, uživatele s postižením, kteří mohou vyžadovat syntetizátory řeči, a zařízení, která si zatím neumíme představit. Kromě toho autoři nemohou v některých případech předpovídat, jak agent zobrazí nebo zpracuje obsah. Zkušenosti ukazují, že oddělením obsahu,

prezentace a interakce se podporuje u obsahu znovupoužitelnost a nezávislost na prostředí; vyplývá to z podstaty ortogonálních specifikací (§5.1, str. 34).

Toto oddělení také usnadňuje znovuvyužití autorova zdrojového obsahu přes rozmanité doručovací souvislosti. Někdy, dle uživatelských funkčních zkušeností, které jsou vhodné k nějaké doručovací souvislosti, mohou být vygenerovány použitím adaptačního procesu aplikovaným na reprezentaci, která nezávisí na přístupovém mechanismu. Pro více informací o principech nezávislosti na prostředí se podívejte na [DIPRINCIPLES].

Dobrá praxe: Oddělení obsahu, prezentace, interakce

Specifikace BY MĚLA dovolit autorům oddělit obsah od prezentace a interakce.

Všimněte si, že když obsah, prezentace a interakce jsou podle návrhu odděleny, agenti je potřebují znovu zkompletovat. Pak existuje rekombinační spektrum s tím, že "klient dělá vše" na jednom konci a "server dělá vše" na druhé straně.

Výhody jsou u každého z přístupů. Například když klient (jako mobilní telefon) sdělí schopnosti zařízení serveru (například použitím CC/PP), server může dodat obsah šitý na míru klientovi. Server může například umožnit rychlejší stahování tím, že přizpůsobí odkazy k zobrazení nižšího rozlišení obrázků, zmenšením videa nebo video nepustí vůbec. Podobně, když byl obsah napsaný s paralelními větvemi, server může odstranit nepoužité větve před doručením. Navíc tím, že připraví obsah odpovídajícím vlastnostem cílového klienta, může server pomoci redukovat vedlejší výpočetní nároky klienta. Nicméně specializující se obsah v tomto stylu redukuje výkonost vyrovnávací paměti.

Na druhé straně vytváření takového obsahu, který může být znovu zkompletován na klientovi, také inklinuje k tomu, že obsah bude vhodný pro širší škálu zařízení. Tento návrh také zlepšuje výkonost vyrovnávací paměti a nabízí uživateli více prezentačních voleb. Styly závislé na médiích se mohou použít pro přizpůsobení obsahu na straně klienta k určitým skupinám cílových zařízení. Pro textový obsah s pravidelným a opakujícím se uspořádáním je sdružení objemu textového obsahu se stylem typičtější než plná kompletace obsahu; uspořádaní se pak dále zlepšuje, pokud je styl opětovně použit jinými stránkami.

V praxi se často užívá kombinace obou přístupů. Rozhodnutí o tom, kam by se měla aplikace umístit závisí na síle klienta, síle a zatíženosti serveru a na přenosovém pásmu média, které je spojuje. Jestli je počet možných klientů neohraničený, aplikace bude vážit lépe, když bude větší zatížení na klientech.

Samozřejmě, že nemusí být žádoucí dosáhnout nejširšího možného publika. Návrháři by měli vzít v úvahu vhodné technologie pro omezení jako šifrování a řízení přístupu (§3.5.2, str. 23).

Některé formáty dat jsou navrženy k tomu, aby popisovaly prezentaci (zahrnující SVG a XSL Formatting Objects). Formáty dat, jako tyto, demonstrují, že někdo může zatím jen oddělit obsah od prezentace (nebo interakce); v některých případech se stává nevyhnutelným mluvit o prezentaci. Ze zásady ortogonální specifikace (§5.1, str. 34) by tyto formáty dat měly *jen* oslovit prezentační problémy.

Podívejte se uveřejnění TAG formattingProperties-19 (vztahující se ke kompatibilitě v případě formátování vlastností a jmen) a contentPresentation-26 (vztahující se k oddělení značení sémantiky a prezentace).

4.4. Hypertext

Definující charakteristikou webu je, že dovoluje začlenit odkazy na další zdroje pomocí URI. Jednoduchost při vytváření hypertextových odkazů používáním absolutních URI (``) a relativních odkazů URI (`` a ``) je částečně (možná velkou měrou), jak už v současnosti víme, odpovědná za úspěch hypertextu webu.

Když reprezentace jednoho zdroje obsahuje odkaz na další zdroj, vyjádřený URI, který ho identifikuje, tak to představuje odkaz mezi dvěma zdroji. Další metadata také mohou tvořit část odkazu (například viz [XLink10]). **Poznámka:** V tomto dokumentu se termínem "odkaz" obecně myslí "vztah", ne "fyzické spojení".

Dobrá praxe: Identifikování odkazu

Specifikace BY MĚLA poskytovat způsoby, jak identifikovat odkazy k ostatním zdrojům, včetně sekundárních (pomocí identifikátorů části).

Formáty, které dovolují autorům obsahu použít URI namísto lokálních identifikátorů, podporují síťový efekt: hodnota těchto formátů roste s velikostí rozmístění webu.

Dobrá praxe: Odkazování na webu

Specifikace BY MĚLA dovolit celosvětové odkazování, nikoli jen vnitřní dokumentové spojování.

Dobrá praxe: Obecné URI

Specifikace BY MĚLA dovolit autorům obsahu, aby mohli používat URI bez omezení na limitovaný soubor URI schémat.

Kteří agenti pracují s hypertextovými odkazy, není architekturou webu omezeno a může záviset na aplikaci. Uživatelé hypertextových odkazů očekávají, že se budou moct pohybovat mezi reprezentacemi následováním odkazů.

Dobrá praxe: Hypertextové odkazy

Formát dat BY MĚL začleňovat hypertextové odkazy, pokud se očekává, že uživatelské rozhraní bude fungovat jako hypertext.

Formáty dat, které nedovolují autorům obsahu vytvořit hypertextové odkazy, vedou k vytvoření "koncových uzlů" webu.

4.4.1. Odkazy URI

Odkazy se obvykle vyjadřují používáním odkazů URI (definovaných v kapitole 4.2 [URI]), které mohou být kombinovány se základním URI pro získání využitelného URI. Kapitola 5.1 [URI] vysvětluje různé způsoby, jak ustanovit základní URI pro zdroj, a stanoví mezi nimi přednost. Například základní URI může být URI pro zdroj nebo specifikovaná v reprezentaci (viz elementy `base` poskytované HTML, a XML a HTTP hlavička 'Content-Location'). Podívejte se také na kapitolu odkazech v XML (§4.5.2, str. 30).

Agenti rozluští odkaz URI před použitím výsledného URI, aby mohli společně působit s dalším agentem. Odkazy URI pomáhají při správě obsahu tím, že dovolí autorům obsahu navrhnout reprezentaci lokálně, to znamená bez starosti, který globální identifikátor se později použije pro odkázání se na přidružený zdroj.

4.5. Formáty dat založené na XML

Mnoho formátů dat je založených na XML, tj., že se přizpůsobí podle syntaktických pravidel definovaných ve specifikaci XML [XML10] nebo [XML11]. Tato kapitola diskutuje o problémech, které jsou charakteristické u takových formátů. Kdokoliv, kdo zde hledá rady a vodítka, se bude muset poradit s "Guidelines For the Use of XML in IETF Protocols" [IETFXML], která obsahuje důkladnou diskusi o úvahách, které obsahují, kdy se má XML používat a kdy ne, stejně jako specifické směrnice, jak by se mělo XML použít. Zatímco je zaměřena na internetové aplikace se specifickým odkazováním na protokoly, je tato diskuse obecně vhodná také pro webové scénáře.

Naše debata by se měla spíše chápat jako pomocná debata k obsahu [IETFXML]. Můžete se odkázat také na "XML Accessibility Guidelines" [XAG] pro pomoc při navrhování formátů XML, které snižují bariéry přístupnosti k webu pro postižené lidi.

4.5.1. Kdy použít formát dat založený na XML

XML definuje textové formáty dat, které se přirozeně hodí pro popis datových objektů, které jsou hierarchické a zpracované ve voleném sledu. Toto je široce ale ne všeobecně platné pro formáty dat; například zvukový formát nebo video pravděpodobně nebudou vhodné pro vyjádření v XML. Návrhová omezení, který by doporučovala použití XML zahrnují:

1. Požadavek na hierarchickou strukturu.
2. Potřeba širokého okruhu nástrojů kvůli různorodosti platform.
3. Potřeba dat, která mohou přežít aplikace, které je právě zpracovávají.
4. Možnost podpořit internacionalizaci samopopisujícím způsobem, který téměř vylučuje zmatek způsobený kódováním.
5. Včasná detekce kódovacích chyb bez nutnosti obcházení takovýchto chyb.
6. Vysoký podíl lidsky čtivého textového obsahu.
7. Potenciální složení formátů dat s ostatními formáty, které se kódují stejně jako XML.
8. Touha po datech, která se snadno analyzují lidmi i stroji.
9. Touha po slovnících, které by se mohly vynalézt tak, aby byly snadno distribuovatelné a daly se pružně kombinovat.

4.5.2. Odkazy v XML

Sofistikované odkazovací mechanismy byly vymyšleny pro formáty XML. XPointer dovolí odkazům oslovit obsah, který nemá explicitně vyjádřenou kotvu. [XLink] je vhodná specifikace pro reprezentaci odkazů v hypertextu (§4.4, str. 28) XML aplikací. XLink dovolí odkazům mít vícenásobné konce, a aby byly vyjádřeny buď v řadě za sebou nebo v "databázi odkazů" uložené externě k některým nebo ke všem zdrojům identifikovaným odkazy, které databáze odkazů obsahuje.

Návrháři formátů založených na XML mohou vzít v úvahu používání XLink a pro definování syntaxe identifikátorů části používáním systému XPointer a schéma element() jazyka XPointer.

XLink není jen propojení návrhu, který byl navržen pro XML a byl by všeobecně přijat za dobrý návrh. Podívejte se také na uveřejnění TAG xlinkScope-23.

4.5.3. Jmenné prostory XML

Účelem jmenného prostoru XML (definovaného v [XMLNS]) je umožnit rozmístění slovníků XML (v kterých se definují názvy elementů a atributů) do globálního prostředí a redukovat riziko jmenných kolizí v daném dokumentu, kde se slovníky zkombinovaly. Například obě specifikace MathML a SVG definují element `set`. Ačkoli se mohou zkombinovat data XML z odlišných formátů jako MathML a SVG v jednom dokumentu, tak by se v tomto případě mohla vyskytnout nejednoznačnost toho, k jakému úmyslu element `set` slouží. Jmenné prostory XML redukuje riziko jmenných kolizí využitím existujících systémů pro přidělování globálně působných jmen: systémů URI (viz také kapitola alokace URI (§2.2.2, str. 11)). Při používání jmenných prostorů XML je každé lokální jméno ve slovníku XML spárováno s URI (zvané jmenný prostor URI), aby se rozlišilo lokální jméno od lokálních jmen z dalších slovníků.

Použitím URI se prokazují další výhody. Za prvé, že každý pár URI/ lokálních jmen může být mapován k dalšímu URI. Tyto termíny mohou být důležitými zdroji, a tudíž je vhodné URI s nimi přidružovat.

[RDFXML] užívá jednoduché slučování jmenného prostoru URI a lokálního jména, aby vytvořil URI pro identifikovaný termín. Další mapování budou pravděpodobně více vhodná pro hierarchické jmenné prostory; viz související uveřejnění TAG abstractComponentRefs-37.

Návrhářům formátů založených na XML, kteří deklarují jmenné prostory, je tak umožněno znovu použít tyto formáty dat a spojovat je tak za pomoci nových metod, jak si doposud nepředstavovali. Opomenutí deklarace jmenných prostorů takovéto opětovné použití ztěžuje a v některých případech je to dokonce nepraktické.

Dobrá praxe: Přijetí jmenného prostoru

Specifikace, která zakládá slovník XML, BY MĚLA umístit všechny názvy elementů a globálních atributů do jmenného prostoru.

Atributy se vždy vztahují k elementu, ve kterém se objevují. Atribut který je "globální, " to jest ten, který by se mohl smysluplně objevit u mnoha elementů, zahrnující elementy v dalších jmenných prostorech, by měl být explicitně umístěn ve jmenném prostoru. Lokální atributy, ty co jsou přidruženy jen s určitým typem elementu, se nemusejí do jmenných prostorů zahrnovat, jelikož jejich význam bude vždy jasný ze souvislosti poskytnuté konkrétním elementem.

Atribut `type` z jmenného prostoru W3C XML Schema Instance "`http://www.w3.org/2001/XMLSchema-instance`" ([XMLSCHEMA], kapitola 4.3.2) je příkladem globálního atributu. Může být použit autory jakéhokoliv slovníku k učinění tvrzení na příklad o datovém typu elementu, na který se vztahuje. Jelikož je to globální atribut, musí být vždy přesně stanovený. Atribut `frame` pro tabulku v HTML je příkladem lokálního atributu. Nemá žádný význam jej umístit do jmenného prostoru, jelikož pravděpodobně nebude užitečný v jiném elementu než v tabulce HTML.

Aplikace, které se spoléhají na zpracování DTD, musejí zavádět další omezení na použití jmenných prostorů. DTD vykonávají ověření platnosti založené na slovní formě jména elementu a atributu v dokumentu. To vytváří prefixy, které jsou syntakticky významné ve věcech, které nejsou předpokládány [XMLNS].

4.5.4. Dokument jmenného prostoru

Příběh

Nadia přijímá reprezentaci dat z "`weather.example.com`" v neznámém datovém formátu. Ví dost o XML, aby poznala, který jmenný prostor XML elementům patří. Jelikož je jmenný prostor identifikovaný URI "`http://weather.example.com/2003/format`", požádá svůj prohlížeče o získání reprezentace identifikovaného zdroje. Dostane některá užitečná data, která jí umožní dozvědět se více o tomto formátu dat. Nadin prohlížeč je také schopen vykonávat některé operace automaticky (to znamená bez dozoru člověka) za předpokladu, že dostane data, která byla optimalizována pro softwarové agenty. Například by prohlížeč mohl v její prospěch stáhnout další agenty, kteří by zpracovali a poskytli formát.

Další výhodou použití URI při budování jmenných prostorů XML je, že se jmenný prostor URI může používat k identifikování informačního zdroje, který obsahuje užitečné, strojově použitelné a/nebo lidsky použitelné informace o termínech ve jmenném prostoru. Tento typ informačního zdroje se nazývá dokument jmenného prostoru. Když vlastník jmenného prostoru URI poskytne dokument jmenného prostoru, tak je tento dokument směrodatný pro jmenný prostor.

Existuje mnoho důvodů poskytovat tento dokument. Někdo by mohl chtít:

- porozumět účelu jmenného prostoru,
- učit se, jak použít značkovací slovník ve jmenném prostoru,
- zjistit, kdo tyto věci řídí, a s nimi přidružené politiky,
- žádat oprávnění k zpřístupnění schémat nebo doplňujících materiálů nebo
- podat zprávu o chybě nebo situaci, která se může považovat za chybu v nějakém doplňujícím materiálu.

Procesor by mohl chtít:

- získat schéma pro ověření platnosti,
- získat šablonu pro prezentaci nebo
- získat ontologie pro stanovení závěrů.

Obecně neexistuje žádná ustanovená nejlepší praxe pro vytvářející reprezentaci dokumentu jmenného prostoru; Aplikační očekávání také ovlivní, jaký se použije formát nebo formáty dat. Dále je také ovlivní, zda se důležitá informace objeví přímo v reprezentaci, nebo se jen na takovou informaci reprezentace odkazuje.

Dobrá praxe: Dokumenty jmenného prostoru

Vlastník jmenného prostoru XML BY MĚL zpřístupnit materiál určený lidem, aby si jej přečetli, a materiál optimalizovaný pro softwarové agenty, za účelem splnit potřeby těch, kdo použijí slovník jmenného prostoru.

Následující příklady jsou příklady formátů dat pro dokumenty jmenných prostorů: [OWL10], [RDDL], [XMLSCHEMA] a [XHTML11]. Každý z těchto formátů se setkává s různými požadavky (popsanými výše) pro uspokojování potřeb agenta, který chce více informací o jmenném prostoru. Nicméně si všimněte problémů souvisejících s identifikátory části a s domlouváním obsahu (§3.2.2, str. 19), jestliže se používá domlouvání obsahu.

Podívejte se na uveřejnění TAG namespaceDocument-8 (který se dotýká požadovaných charakteristických rysů dokumentu jmenného prostoru) a abstractComponentRefs-37 (vztahující se k použití identifikátorů části s názvem jmenného prostoru k identifikování abstraktní součásti).

4.5.5. Qnames v XML

Kapitola 3 v "Namespaces in XML" [XMLNS]] poskytuje syntaktický pojem známý jako QName pro kompaktní vyjádření kvalifikovaných jmen (qualified names = Qnames) v XML dokumentech. Kvalifikované jméno je dvojice URI a lokálního názvu, která pojmenovává jmenný prostor a lokální název umístěný uvnitř tohoto jmenného prostoru. "Namespaces in XML" umožňují používat QNames jak pro jména elementů, tak pro atributy XML.

Další specifikace, počínaje [XSLT10], upotřebily nápad používání QNames v jiné souvislosti než u jmen elementů a atributů, například u hodnot atributu a obsahu elementu. Nicméně hlavní procesory XML nemohou spolehlivě poznat QNames v takových případech, jako u hodnot atributu a obsahu elementu; syntaxe QNames se například překrývá s URI. Zkušenosti také odhalily další omezení QNames jako ztráta vazby jmenného prostoru po kanonizaci XML.

Omezení: Qnames nerozeznatelná od URI

Nedovoďte QNames a URI v hodnotách atributu nebo v obsahu elementu, kde nejsou rozeznatelná.

Pro více informací se podívejte na nálezy TAG *"Using QNames as Identifiers in Content"*.

Protože jsou QNames kompaktní, někteří návrháři specifikací přijali stejnou syntaxi jako prostředky k identifikování zdrojů. Ačkoli jsou vhodná jako těsnopisný záznam, jejich užívání přináší i svou cenu. Není žádný jednotně přijatý způsob, jak převést QName na URI nebo naopak. Ačkoli jsou QNames šikovná, nezamění URI jakožto poznávací systém webu. Použití QNames k identifikování webového zdroje bez poskytnutí mapování URI je v rozporu s architekturou webu.

Dobrá praxe: Mapování QName

Specifikace, v které QNames slouží jako identifikátory zdroje, MUSÍ poskytnout mapování na URI.

Viz jmenné prostory XML (§4.5.3, str. 30) pro příklady mapujících strategií.

Podívejte se také na uveřejnění TAG rdfsQnameUriMapping-6 (vztahující se k mapování QNames na URI), qnameAsId-18 (vztahující se k použití QNames jako identifikátorů v obsahu XML) a abstractComponentRefs-37 (vztahující se k použití identifikátorů části s názvem jmenného prostoru k identifikování abstraktní komponenty).

4.5.6. Sémantika XML ID

Uvažme následující kousek kódu XML: `<section name="foo">`. Má být element `section` podle doporučení XML identifikován pomocí ID `foo` (znamená to, že se "foo" nesmí objevit v okolním XML dokumentu více než jednou)? Nikdo nemůže odpovědět na tuto otázku zkoumáním osamoceneného elementu a jeho atributů. V XML je vlastnost "být ID" přidružená s typem atributu, nikoliv s jeho jménem. Nalezení ID v dokumentu vyžaduje další zpracování.

1. Zpracování dokumentu procesorem, který pozná v DTD seznam deklarací atributů (v externí nebo interní podmnožině), by mohlo odhalit deklaraci, která identifikuje atribut `name` jako ID. **Poznámka:** Toto zpracování není nutně částí validace. Nevalidní procesor podporující DTD může rozpoznat ID.
2. Zpracování dokumentu s W3C XML schématem by mohlo odhalit deklaraci elementu, který identifikuje atribut `name` jako W3C XML Schema ID.
3. V praxi by zpracování dokumentu s dalším schématem jazyka jako je RELAX NG [RELAXNG] mohlo odhalit atributy deklarované jako ID ve smyslu XML Schema. Mnoho moderních specifikací začíná zpracování XML na úrovni Infoset [INFOSET] a normativně nespecifikuje, jak je Infoset zkonstruován. Pro tyto specifikace jakýkoliv postup, který zavádí typ ID do Infoset (a do Post Schema Validation Infoset (PSVI) definované v [XMLSCHEMA]) může užitečně identifikovat atributy typu ID.
4. V praxi mohou mít aplikace nezávislé prostředky (jako ty definované ve XPointer specifikaci, [XPTRFR] kapitola 3.2) k lokalizaci identifikátorů uvnitř dokumentu.

Dále se ještě situace komplikuje tím, že DTD zavádí typ ID do Infoset zatímco W3C XML Schema produkuje PSVI, ale nemodifikuje originální Infoset. Což nechává otevřenou možnost, že by se procesor mohl podívat pouze do Infoset, a následkem toho by chybně rozeznal schématu přiřazená ID.

Podívejte se uveřejnění TAG xmlIDSemantics-32 pro hlubší informace a [XML-ID] pro řešení při vývoji.

4.5.7. Typy médií pro XML

RFC 3023 definuje typy internetových médií "application/xml" a "text/xml" a popisuje úmluvu, kterou formáty dat založené na XML užívají typy médií s "+xml" příponou, například "image/svg+xml".

Jsou zde dva problémy spojené s typem médií "text": Zaprvé, pro data identifikovaná jako "text/*" mají weboví prostředníci povoleno "překódování", čili přeměnit jedno kódování znaků na druhé. Překódování se může podařit špatně nebo může způsobit dokument nečitelným.

Dobrá praxe: XML a "text/*"

Obecně BY poskytovatel reprezentace NEMĚL přiřazovat typům internetových médií začátek "text/" pro XML reprezentace.

Zadruhé, reprezentace, jejichž typy internetových médií vyžadují začínat výrazem "text/", a pokud nemají specifikovaný parametr `charset`, tak očekávají kódování US-ASCII. Od té doby, co je syntaxe XML navržena k tomu, aby učinila dokumenty samopopisnými, je dobrou praxí vynechávat parametr `charset`, a jelikož XML velice často není zakódovaný v US-ASCII, použití výrazu "text/" u typů internetových médií účinně zamezí této dobré praxi.

Dobrá praxe: XML a kódování znaků

Obecně BY poskytovatel reprezentace NEMĚL specifikovat kódování znaků pro data XML v hlavičkách protokolu, jelikož jsou data samopopisná.

4.5.8. Identifikátory části v XML

Kapitola zaměřená na typy médií a sémantiku identifikátoru části (§3.2.1, str. 18) probírá o výklad identifikátoru části. Návrháři specifikací formátů dat založených na XML by měli definovat sémantiku identifikátoru části v tomto formátu. XPointer Framework [XPTRFR] poskytuje vhodné východisko.

Když je typ média, přiřazený k reprezentaci dat, "application/xml", tak zde není žádná sémantika, která by byla definovaná pro identifikátory části, a autoři by v takovýchto datech používat tyto identifikátory neměli. To samé nastane, jestliže má přiřazený typ média příponu "+xml" (definovanou v "XML Media Types" [RFC3023]), a specifikace formátu dat nespecifikuje sémantiku identifikátoru části. To, že již víme, že je obsah v XML, ještě neznamená, že poskytuje informaci o sémantice identifikátoru části.

Mnoho lidí předpokládá, že identifikátor části #abc, při odkazování na data XML, identifikuje element v dokumentu s ID "abc". Nicméně pro tento předpoklad není žádná normativní podpora. Očekává se, že oprava RFC 3023 toto vyřeší.

Podívejte se na uveřejnění TAG fragmentInXML-28.

4.6. Budoucí směry pro formáty dat

Formáty dat umožní vytvoření nových aplikací, které budou moci použít infrastrukturu informačního prostoru. Sémantický web je jedna z takových aplikací, postavený na prvcích RDF [RDFXML]. Tento dokument sémantický web podrobněji neprobírá; TAG očekává že se napíše budoucí vydání tohoto dokumentu. Podívejte se na související uveřejnění TAG httpRange-14.

5. Všeobecné architekturní zásady

Několik všeobecných architekturních zásad se aplikuje na všechny tři základy architektury webu.

5.1. Ortogonální specifikace

Identifikace, interakce a reprezentace jsou ortogonální koncepty, které znamenají, že technologie používané pro identifikaci, interakci a reprezentaci se mohou vyvíjet nezávisle. Například:

- Zdroje jsou identifikovány pomocí URI. URI mohou být zveřejněné bez vybudování jakékoliv reprezentace zdroje nebo bez určení, zda jsou některé reprezentace dostupné.
- Syntaxe obecného URI dovoluje agentům fungovat v mnoha případech bez znalosti specifik schémat URI.
- V mnoha případech může někdo změnit reprezentaci zdroje bez přerušení odkazů na zdroj (například použitím domlouvání obsahu).

Když jsou dvě specifikace ortogonální, tak jedna může změnit druhou bez vyžadování změn té další, dokonce i když je ta první závislá na druhé. Například přestože specifikace HTTP závisí na specifikaci URI, obě se mohou vyvíjet nezávisle. Tato ortogonalita zvyšuje flexibilitu a robustnost webu. Například se někdo může odkazovat pomocí URI na obrázek bez jakékoliv znalosti formátu, který je vybrán k reprezentaci tohoto obrázku. Toto usnadňuje představit si formáty obrázku jako PNG a SVG bez přerušení existujících odkazů na zdroje obrázků.

Zásada: Ortogonalita

Ortogonalní abstrakce těží z ortogonálních specifikací.

Zkušenosti ukazují, že problémy vznikají, když se ortogonální koncepty vyskytují v jedné specifikaci. Představme si například specifikaci HTML, která zahrnuje ortogonální specifikaci x-www-form-urlencoded. Softwaroví vývojáři (například vývojáři [CGI] aplikací)by mohli rychleji nalézt specifikaci, kdyby byla publikovaná odděleně, a potom citovaná ze specifikací HTTP, URI a HTML.

Problémy také vyvstávají, když se specifikace pokouší modifikovat ortogonální abstrakce popisované jinde. Historická verze specifikace HTML přidala hodnotu "Refresh" k atributu http-equiv elementu meta. Byl definován jako ekvivalentní k hlavičce HTTP stejného jména. Autoři specifikace HTTP se nakonec rozhodli neposkytovat tuto hlavičku, která by těmto dvěma specifikacím ztížila výhody mezi sebou. W3C HTML Working Group nakonec hodnotu "Refresh" odstranila.

Specifikace by měla zřetelně ukazovat, které rysy překrývá s rysy řízenými dalšími specifikacemi.

5.2. Rozšiřitelnost

Informace na webu a technologie, které je zobrazují se v průběhu doby neustále mění. Rozšiřitelnost je vlastnost technologie, která podporuje vývoj bez obětování kompatibility. Některé příklady úspěšně vytvořených technologií, které umožňují změnu za minimálního počtu poruch zahrnují:

- fakt, že jsou schémata URI specifikovaná ortogonálně;
- použití volně dostupné (open) sady typů internetových médií v emailech a HTTP, které specifikují interpretaci dokumentu;
- oddělení obecné gramatiky XML a otevřené sady jmenných prostorů XML pro jména elementů a atributů;
- modely rozšiřitelnosti v kaskádových stylech (CSS), XSLT 1.0 a v SOAP;
- plug-iny uživatelských agentů.

Příklad neúspěšného rozšiřujícího mechanismu je direktivní rozšíření HTTP [HTTPEXT]. Společenství hledalo mechanismy k rozšíření HTTP, ale zřejmě náklady závazně rozšiřujícího návrhu (náklady znatelné ve složitosti) přesáhly výhody, a tak brzdil přijetí.

Níže diskutujeme o vlastnictví "rozšiřitelnosti" ukázaného na URI, některých formátech dat a některých protokolech (během začlenění nových zpráv).

Zúžený jazyk: Jeden jazyk je podmnožinou (nebo "profil") druhého jazyka, jestliže nějaký dokument v prvním jazyce je také platný dokument ve druhém jazyce a má stejný výklad ve druhém jazyce.

Rozšířený jazyk: Jestliže jeden jazyk je podmnožinou jiného, ten druhý "nadmnožinový" se nazývá rozšířený jazyk; rozdíl mezi jazyky se nazývá rozšíření. Je jasné, že rozšířený jazyk je lepší pro kompatibilitu, než vytvoření nekompatibilního jazyka.

Teoreticky může být mnoho příkladů "nadmnožinových" jazyků bezpečně a užitečně zpracováno jakoby byly v zúženém jazyce. Jazyku, který se může takto vyvíjet, dovolujícímu aplikacím poskytnout nové informace, když je potřeba, zatímco je stále zaručena kompatibilita s aplikacemi, které zatím jen rozumí podmnožině aktuálního jazyka, se říká, že je "rozšiřitelný." Jazykoví návrháři mohou usnadnit rozšiřitelnost definováním výchozího chování neznámých rozšíření - například, že budou ignorována (nějakým definovaným způsobem) nebo mohou být považována za chyby.

Například v počátcích webu HTML agenti ignorovali podle dohody neznámé tagy. Tato volba nechala prostor pro inovaci (tím se myslí nestandardní elementy) a povzbuzovala rozšíření HTML. Nicméně s tím vyvstaly i problémy s kompatibilitou. V tomto typu prostředí je nevyhnutelné napětí mezi krátkodobou kompatibilitou a touhou po rozšiřitelnosti. Zkušenosti ukazují, že návrhům, které nastolí pravou rovnováhu mezi dovolováním měnit a zachováním kompatibility, se pravděpodobněji bude dařit se a pravděpodobněji méně naruší webové společenství. Ortogonální specifikace (§5.1, str. 34) pomáhají redukovat riziko narušení.

Pro další diskuzi se podívejte na kapitolu o verzování a rozšiřitelnosti (§4.2, str. 26). Podívejte se také na uveřejnění TAG xmlProfiles-29 a HTML Dialects.

5.3. Ošetření chyb

Chyby se v propojené síti informačních systémů objevují. Chybový stav může být dobře rozpoznatelný (například syntaktické chyby v XML nebo chyby typu 4xx v klientovi v HTTP) nebo vyvstávají nepředvídatelně. Oprava chyby znamená, že agent opraví stav tak, že uvnitř systému to bude vypadat, jakoby chyba nikdy nastala. Jeden příklad opravy chyby vyžaduje opětovný přenos dat jako odpověď na dočasné síťové selhání. Zotavení po chybě znamená, že agent neopraví chybový stav, ale pokračuje ve zpracování s oznámením skutečnosti, že chyba nastala.

Agenti často *opraví* chyby bez uživatelského vědomí, aby ušetřily uživatele detailů komplexní síťové komunikace. Na druhé straně je důležité, aby agenti *obnovili* stav způsobem, který je evidentně uživatelem vidět, jelikož agenti jednají v jeho prospěch.

Zásada: Zotavení po chybě

Agenti, kteří zvolí volbu obnovení stavu bez uživatelského souhlasu, nejednají v jeho prospěch.

U agenta se nevyžaduje přerušit uživatele (například, že vyskočí okno pro potvrzení), aby získal souhlas. Uživatel může svůj souhlas již přednastavit vhodnou konfigurační volbou, režimem nebo jiným rozhraním s příslušným

oznámením uživatelovi, jakmile agent detekuje chybu. Vývojáři agentů by neměli ignorovat otázky týkající se použitelnosti, když navrhují chování při zotavování se po chybě.

Kvůli podporování kompatibility by návrháři specifikací měli identifikovat předpověditelné chybové stavy. Zkušenosti vedly k následujícím pozorováním o přístupech při ošetřování chyb.

- Návrháři protokolů by měli poskytnout dostatek informací o chybovém stavu tak, že agent může oznámit chybový stav. Například stavový kód HTTP 404 (soubor nenalezen) je užitečný, protože dovolí uživatelským agentům prezentovat důležitou informaci k uživatelům, a je jim umožněno v případě problému kontaktovat poskytovatele reprezentace.
- Zkušenosti vyplývající z úsilí budování uživatelského klienta, který by ovládal různorodé formy špatně utvořeného obsahu HTML, přesvědčily návrháře specifikací XML, aby vyžadovali, aby agenti selhali při načítání chybového obsahu. Protože uživatelé neradi tolerují takovéto poruchy, tato volba návrhu tlačila na všechny strany, aby respektovali omezení XML, k uspokojení všech.
- Agent, který se setká s neznámým obsahem, by tento obsah mohl zpracovat různými způsoby, včetně považování situace za chybu; viz také kapitola rozšiřitelnost a verzování (§4.2, str. 26).
- Chybové chování, které je patřičné pro člověka, nemusí být patřičné pro software. Lidé jsou schopni tříbit si svůj úsudek způsoby, které aplikace nezvládnou. Neformální oznámení chyby může stačit člověku ale ne procesoru.

Podívejte se na uveřejnění TAG contentTypeOverride-24, který se týká zdroje o směrodatných metadatech.

5.4. Součinnost založená na protokolu

Web následuje tradici internetu v tom směru, že jsou jeho důležitá rozhraní definována v rámci protokolů, specifikováním syntaktických, sémantických a sekvenčních omezení z vyměněných zpráv. Protokoly navržené tak, aby byly flexibilní tváří v tvář široce proměnnému prostředí, pomohly rozsahu webu a usnadnily posun komunikace k novým hraničním důvěry. Tradiční aplikační programovací rozhraní (API) neberou vždy tato omezení v úvahu, ani kdyby se požadovala. Jedním účinkem návrhu založeného na protokolu je to, že sdílení technologie mezi agenty často trvá déle než agenti sami.

Je běžné pro programátory, kteří pracují s webem, aby psali kód, který generuje a rozebere tyto zprávy přímo. Je méně běžné, ale nikoliv neobvyklé, pro koncové uživatele mít přímý přístup k těmto zprávám. Často je žádoucí poskytnout uživateli přístup k detailům o formátu a o protokolu: umožnit jim "vidět zdroj," kterým mohou získat odborné znalosti při práci se základním (ve smyslu s nejspodnějším) systémem.

6. Slovníček

Domlouvání obsahu

Praxe poskytování rozmanitých reprezentací dostupných přes stejný URI. Která reprezentace se použije, závisí na domluvě mezi žádajícím agentem a agentem, který obsluhuje reprezentace.

Získávání URI

Zpřístupnění reprezentace zdroje identifikovaného podle URI.

Oprava chyb

Agent opraví stav tak, že uvnitř systému to bude vypadat, jakoby chyba nikdy nenastala.

Zotavení po chybě

Agent se dovolává výjimečného chování, protože chybu neopraví.

Rozšířený jazyk

Jestliže je jeden jazyk podmnožinou druhého, tak ten druhý se nazývá rozšířený jazyk.

Identifikátor části

Část URI, která umožňuje identifikaci sekundárního zdroje.

Informační zdroj

Zdroj, který má takovou vlastnost, že všechny jeho podstatné znaky mohou být zprostředkovány ve zprávě.

Odkaz

Vztah mezi dvěma zdroji, kdy se jeden zdroj (reprezentace) odkazuje na další zdroj přes URI.

Zpráva

Jednotka komunikace mezi agenty.

Dokument jmenného prostoru

Informační zdroj identifikovaný XML jmenným prostorem URI, který obsahuje užitečné informace, strojově použitelné a/nebo lidsky použitelné, o podmínkách v jednotlivých jmenných prostorech XML. Je užitečné, nikoliv však povinné, že URI užívaný jako název jmenného prostoru, identifikuje dokument jmenného prostoru.

Reprezentace

Data, která zakódují informaci o stavu zdroje.

Zdroj

Cokoli, co by mohlo být identifikované pomocí URI.

Bezpečná interakce

Interakce se zdrojem, kde si agent nezpůsobí žádný závazek z interakce.

Sekundární zdroj

Zdroj související s dalším zdrojem hlavního zdroje s další identifikující informací (identifikátor části).

Zúžený jazyk

Jeden jazyk je podmnožinou druhého jazyka, jestliže nějaký dokument v prvním jazyce je také platný dokument ve druhém jazyce a má stejný výklad ve druhém jazyce.

URI

Zkratka pro jednotný identifikátor zdroje.

Alias URI

Dva nebo více různých URI, které identifikují stejný zdroj.

Kolize URI

Použití stejného URI k odkázání se na více než jeden zdroj ve smyslu protokolů a formátů webu.

Vlastnictví URI

Vztah mezi URI a společenskou jednotkou, jako je osoba, organizace nebo specifikace.

Stálost URI

Společenské očekávání, že jeden URI identifikuje specifický zdroj, by mělo pokračovat neurčitě dlouho stálým odkazováním se na tento zdroj.

Odkaz URI

Pracovní zkratka pro URI.

Jednotný identifikátor zdroje (URI)

Globální identifikátor, v tomto smyslu pro World Wide Web.

Nebezpečná interakce

Interakce se zdrojem, který není bezpečnou interakcí.

Uživatelský agent

Jeden typ webového agenta. Část softwaru jednající jménem osoby.

WWW

Zkratka pro World Wide Web.

Web

Zkrácená forma pro World Wide Web.

Webový agent

Osoba nebo část softwaru jednající v informačním prostoru jménem osoby, entity nebo procesu.

World Wide Web

Informační prostor, ve kterém jsou položky zájmu identifikovány jednotnými identifikátory zdroje.

Formát založený na XML

Takový, který se přizpůsobí syntaktickým pravidlům definovaných ve specifikaci XML.

7. Odkazy

CGI

Common Gateway Interface/1.1 Specification. Dostupné na <http://hoohoo.ncsa.uiuc.edu/cgi/interface.html>.

CHIPS

Common HTTP Implementation Problems, O. Théreaux, leden 2003. Toto W3C Team Submission je dostupné na <http://www.w3.org/TR/chips/>.

CUAP

Common User Agent Problems, K. Dubost, LEDEN 2003. Toto W3C Team Submission je dostupné na <http://www.w3.org/TR/cuap>.

Cool

Cool URIs don't change T. Berners-Lee, W3C, 1998 Dostupné na <http://www.w3.org/Provider/Style/URI>. Všimněte si, že titul je poněkud zavádějící. Nejsou to URI, které se mění, mění se to, co identifikují.

Eng90

Knowledge-Domain Interoperability and an Open Hyperdocument System, D. C. Engelbart, červen 1990.

HTTPEXT

Mandatory Extensions in HTTP, H. Frystyk Nielsen, P. Leach, S. Lawrence, 20. leden 1998. Tento vysloužilý internetový koncept je dostupný na <http://www.w3.org/Protocols/HTTP/ietf-http-ext/draft-frystyk-http-mandatory>.

IANASchemes

IANA's online registry of URI Schemes je dostupný na <http://www.iana.org/assignments/uri-schemes>.

IETFXML

IETF Guidelines For The Use of XML in IETF Protocols, S. Hollenbeck, M. Rose, L. Masinter, eds., 2. listopad 2002. Tento vysloužilý internetový koncept je dostupný na <http://www.imc.org/ietf-xml-use/xml-guidelines-07.txt>. Pokud tento dokument již není déle dostupný, odkazujte se prosím na ietf-xml-use mailing list.

INFOSET

XML Information Set (Second Edition), R. Tobin, J. Cowan, Editors, W3C Recommendation, 4. únor 2004, <http://www.w3.org/TR/2004/REC-xml-infoset-20040204>. Poslední verze dostupná na <http://www.w3.org/TR/xml-infoset>.

IRI

IETF Internationalized Resource Identifiers (IRIs), M. Dürst, M. Suignard, Eds., listopad 2004. V prohlášení z 8. prosince 2004 IESG schválilo draft-duerst-iri-11 jakožto navrhovaný standard IETF. Odkazy na specifikaci IRI - první vydání *Architektury World Wide Web* jsou k navrhovanému standardu. Jelikož IETF zveřejňuje

číslo Request For Comments (RFC) pro specifikaci, toto doporučení W3C může být aktualizováno a odkazovat se na toto RFC. Podívejte se také na poslední informace o specifikaci IRI specification.

MEDIATYPereg

IANA's online registry of Internet Media Types je dostupný na <http://www.iana.org/assignments/media-types/index.html>.

OWL10

OWL Web Ontology Language Reference, M. Dean, G. Schreiber, Editors, W3C Recommendation, 10. únor 2004, <http://www.w3.org/TR/2004/REC-owl-ref-20040210/>. Poslední verze dostupná na <http://www.w3.org/TR/owl-ref/>.

P3P10

The Platform for Privacy Preferences 1.0 (P3P1.0) Specification, M. Marchiori, Editor, W3C Recommendation, 16. duben 2002, <http://www.w3.org/TR/2002/REC-P3P-20020416/>. Poslední verze dostupná na <http://www.w3.org/TR/P3P/>.

RDDL

Resource Directory Description Language (RDDL), J. Borden, T. Bray, eds., 1. červen 2003. Tento dokument je dostupný na <http://www.tbray.org/tag/rddl/rddl3.html>.

RDFXML

RDF/XML Syntax Specification (Revised), D. Beckett, Editor, W3C Recommendation, 10. únor 2004, <http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/>. Poslední verze dostupná na <http://www.w3.org/TR/rdf-syntax-grammar>.

RELAXNG

The RELAX NG schema language project.

REST

Representational State Transfer (REST), kapitola 5 "Architectural Styles and the Design of Network-based Software Architectures", doktorská disertace R. T. Fieldinga, 2000. Návrháři protokolových specifikací by měli obzvlášť investovat čas do pochopení modelu REST a do závažnosti principů daného návrhu. Tyto principy zahrnují státní bezpříslušnost, jasné přiřazení rolí stran, jednotný adresový prostor a omezený, jednotný soubor sloves. Dostupné na http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm.

RFC2045

IETF RFC 2045: Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies, N. Freed, N. Borenstein, listopad 1996. Dostupné na <http://www.ietf.org/rfc/rfc2045.txt>.

RFC2046

IETF RFC 2046: Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types, N. Freed, N. Borenstein, listopad 1996. Dostupné na <http://www.ietf.org/rfc/rfc2046.txt>.

RFC2119

IETF RFC 2119: Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, březen 1997. Dostupné na <http://www.ietf.org/rfc/rfc2119.txt>.

RFC2141

IETF RFC 2141: URN Syntax, R. Moats, květen 1997. Dostupné na <http://www.ietf.org/rfc/rfc2141.txt>.

RFC2326

IETF RFC 2326: Real Time Streaming Protocol (RTSP), H. Schulzrinne, A. Rao, R. Lanphier, duben 1998. Dostupné na: <http://www.ietf.org/rfc/rfc2326.txt>.

RFC2397

IETF RFC 2397: The "data" URL scheme, L. Masinter, srpen 1998. Dostupné na: <http://www.ietf.org/rfc/rfc2397.txt>.

RFC2616

IETF *RFC 2616: Hypertext Transfer Protocol - HTTP/1.1*, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, červen 1999. Dostupné na <http://www.ietf.org/rfc/rfc2616.txt>.

RFC2717

IETF *Registration Procedures for URL Scheme Names*, R. Petke, I. King, listopad 1999. Dostupné na <http://www.ietf.org/rfc/rfc2717.txt>.

RFC2718

IETF *RFC 2718: Guidelines for new URL Schemes*, L. Masinter, H. Alvestrand, D. Zigmond, R. Petke, listopad 1999. Dostupné na: <http://www.ietf.org/rfc/rfc2718.txt>.

RFC2818

IETF *RFC 2818: HTTP Over TLS*, E. Rescorla, květen 2000. Dostupné na: <http://www.ietf.org/rfc/rfc2818.txt>.

RFC3023

IETF *RFC 3023: XML Media Types*, M. Murata, S. St. Laurent, D. Kohn, leden 2001. Dostupné na: <http://www.ietf.org/rfc/rfc3023.txt>

RFC3236

IETF *RFC 3236: The 'application/xhtml+xml' Media Type*, M. Baker, P. Stark, leden 2002. Dostupné na: <http://www.ietf.org/rfc/rfc3236.txt>

RFC3261

IETF *RFC 3261: SIP: Session Initiation Protocol*, J. Rosenberg, H. Schulzrinne, G. Camarillo, et. al., červen 2002. Dostupné na: <http://www.ietf.org/rfc/rfc3261.txt>

RFC3920

IETF *RFC 3920: Extensible Messaging and Presence Protocol (XMPP): Core*, P. Saint-Andre, Ed., říjen 2004. Dostupné na: <http://www.ietf.org/rfc/rfc3920.txt>

RFC977

IETF *RFC 977: Network News Transfer Protocol*, B. Kantor, P. Lapsley, únor 1986. Dostupné na <http://www.ietf.org/rfc/rfc977.txt>.

SOAP12

SOAP Version 1.2 Part 1: Messaging Framework, J. Moreau, N. Mendelsohn, H. Frystyk Nielsen, et. al., Editors, W3C Recommendation, 24. červen 2003, <http://www.w3.org/TR/2003/REC-soap12-part1-20030624/>. Poslední verze dostupná na <http://www.w3.org/TR/soap12-part1/>.

SVG11

Scalable Vector Graphics (SVG) 1.1 Specification, J. Ferraiolo, D. Jackson, Editors, W3C Recommendation, 14. leden 2003, <http://www.w3.org/TR/2003/REC-SVG11-20030114/>. Poslední verze dostupná na <http://www.w3.org/TR/SVG11/>.

UNICODE

The Unicode Consortium, The Unicode Standard, Version 4, ISBN 0-321-18578-1, čas od času aktualizováno publikováním nových verzí. Viz <http://www.unicode.org/unicode/standard/versions> pro nejnovější verzi Unicode a pro další informace verzí standardní a Unicode Character databáze.

URI

IETF *Uniform Resource Identifiers (URI): Generic Syntax*, T. Berners-Lee, R. Fielding, L. Masinter, Eds., září 2004. V prohlášení z 18. října 2004 IESG schválilo draft-fielding-uri-rfc2396bis-07 jako "Full Standard" IETF. Odkazy na specifikaci URI v prvním vydání *Architektury World Wide Webu* jsou k tomuto "Full Standartu". Jelikož IETF zveřejňuje číslo Request For Comments (RFC) pro specifikaci, toto doporučení W3C může být aktualizováno a odkazovat se na toto RFC. Podívejte se také na poslední informace o specifikaci URI.

UniqueDNS

IAB Technical Comment on the Unique DNS Root, B. Carpenter, 27. září 1999. Dostupné na <http://www.icann.org/correspondence/iab-tech-comment-27sept99.htm>.

VOICEXML2

Voice Extensible Markup Language (VoiceXML) Version 2.0, B. Porter, A. Hunt, K. Rehor, *et. al.*, Editors, W3C Recommendation, 16. březen 2004, <http://www.w3.org/TR/2004/REC-voicexml20-20040316/>. Poslední verze dostupná na <http://www.w3.org/TR/voicexml20>.

XHTML11

XHTML™ 1.1 - Module-based XHTML, S. McCarron, M. Altheim, Editors, W3C Recommendation, 31. květen 2001, <http://www.w3.org/TR/2001/REC-xhtml11-20010531>. Poslední verze dostupná na <http://www.w3.org/TR/xhtml11/>.

XLink10

XML Linking Language (XLink) Version 1.0, E. Maler, S. DeRose, D. Orchard, Editors, W3C Recommendation, 27. červen 2001, <http://www.w3.org/TR/2001/REC-xlink-20010627/>. Poslední verze dostupná na <http://www.w3.org/TR/xlink/>.

XML-ID

xml:id Version 1.0, D. Veillard, J. Marsh, Editors, W3C Working Draft (work in progress), 7. duben 2004, <http://www.w3.org/TR/2004/WD-xml-id-20040407>. Poslední verze dostupná na <http://www.w3.org/TR/xml-id/>.

XML10

Extensible Markup Language (XML) 1.0 (Third Edition), F. Yergeau, J. Paoli, C. M. Sperberg-McQueen, *et. al.*, Editors, W3C Recommendation, 4. duben 2004, <http://www.w3.org/TR/2004/REC-xml-20040204>. Poslední verze dostupná na <http://www.w3.org/TR/REC-xml>.

XML11

Extensible Markup Language (XML) 1.1, J. Paoli, C. M. Sperberg-McQueen, J. Cowan, *et. al.*, Editors, W3C Recommendation, 4. únor 2004, <http://www.w3.org/TR/2004/REC-xml11-20040204/>. Poslední verze dostupná na <http://www.w3.org/TR/xml11/>.

XMLNS

Namespaces in XML 1.1, R. Tobin, D. Hollander, A. Layman, *et. al.*, Editors, W3C Recommendation, 4. duben 2004, <http://www.w3.org/TR/2004/REC-xml-names11-20040204>. Poslední verze dostupná na <http://www.w3.org/TR/xml-names11/>.

XMLSCHEMA

XML Schema Part 1: Structures, D. Beech, M. Maloney, H. S. Thompson, *et. al.*, Editors, W3C Recommendation, 2. květen 2001, <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>. Poslední verze dostupná na <http://www.w3.org/TR/xmlschema-1/>.

XPTRFR

XPointer Framework, E. Maler, N. Walsh, P. Grosso, *et. al.*, Editors, W3C Recommendation, 25. březen 2003, <http://www.w3.org/TR/2003/REC-xptr-framework-20030325/>. Poslední verze dostupná na <http://www.w3.org/TR/xptr-framework/>.

XSLT10

XSL Transformations (XSLT) Version 1.0, J. Clark, Editor, W3C Recommendation, 16. listopad 1999, <http://www.w3.org/TR/1999/REC-xslt-19991116>. Poslední verze dostupná na <http://www.w3.org/TR/xslt>.

7.1. Architektonické specifikace

ATAG10

Authoring Tool Accessibility Guidelines 1.0, C. McCathieNevile, I. Jacobs, J. Trevisanus, *et. al.*, Editors, W3C Recommendation, 3. únor 2000, <http://www.w3.org/TR/2000/REC-ATAG10-20000203>. Poslední verze dostupná na <http://www.w3.org/TR/ATAG10>.

CHARMOD

Character Model for the World Wide Web 1.0: Fundamentals, R. Ishida, M. J. Dürst, M. Wolf, *et. al.*, Editors, W3C Working Draft (work in progress), 25. únor 2004, <http://www.w3.org/TR/2004/WD-charmod-20040225/>. Poslední verze dostupná na <http://www.w3.org/TR/charmod/>.

DIPRINCIPLES

Device Independence Principles, R. Gimson, Editor, W3C Note, 1. září 2003, <http://www.w3.org/TR/2003/NOTE-di-princ-20030901/>. Poslední verze dostupná na <http://www.w3.org/TR/di-princ/>.

EXTLANG

Web Architecture: Extensible Languages, T. Berners-Lee, D. Connolly, 10. únor 1998. Tyto "W3C Note" jsou dostupné na <http://www.w3.org/TR/1998/NOTE-webarch-extlang-19980210>.

Fielding

Principled Design of the Modern Web Architecture, R.T. Fielding and R.N. Taylor, UC Irvine. R.T. Fielding and R.N. Taylor, UC Irvine. Ze zpráv Mezinárodní konference o softwarovém inženýrství z roku 2000. (ICSE 2000), Limerick, Ireland, červen 2000, strany 407-416. Tento dokument je dostupný na http://www.ics.uci.edu/~fielding/pubs/webarch_icse2000.pdf.

QA

QA Framework: Specification Guidelines, D. Hazaël-Massieux, L. Rosenthal, L. Henderson, *et. al.*, Editors, W3C Working Draft (work in progress), 30. srpen 2004, <http://www.w3.org/TR/2004/WD-qaframe-spec-20040830/>. Poslední verze dostupná na <http://www.w3.org/TR/qaframe-spec/>.

RFC1958

IETF *RFC 1958: Architectural Principles of the Internet*, B. Carpenter, červen 1996. Dostupné na <http://www.ietf.org/rfc/rfc1958.txt>.

SPECVAR

Variability in Specifications, L. Rosenthal, D. Hazaël-Massieux, Editors, W3C Working Draft (work in progress), 30. srpen 2004, <http://www.w3.org/TR/2004/WD-spec-variability-20040830/>. Poslední verze dostupná na <http://www.w3.org/TR/spec-variability/>.

UAAG10

User Agent Accessibility Guidelines 1.0, J. Gunderson, I. Jacobs, E. Hansen, Editors, W3C Recommendation, 17. prosinec 2002, <http://www.w3.org/TR/2002/REC-UAAG10-20021217/>. Poslední verze dostupná na <http://www.w3.org/TR/UAAG10/>.

WCAG20

Web Content Accessibility Guidelines 2.0, W. Chisholm, J. White, B. Caldwell, *et. al.*, Editors, W3C Working Draft (work in progress), 30. červenec 2004, <http://www.w3.org/TR/2004/WD-WCAG20-20040730/>. Poslední verze dostupná na <http://www.w3.org/TR/WCAG20/>.

WSA

Web Services Architecture, D. Booth, F. McCabe, E. Newcomer, *et. al.*, Editors, W3C Note, 11. únor 2004, <http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/>. Poslední verze dostupná na <http://www.w3.org/TR/ws-arch/>.

XAG

XML Accessibility Guidelines, S. B. Palmer, C. McCathieNevile, D. Dardailler, Editors, W3C Working Draft (work in progress), 3. říjen 2002, <http://www.w3.org/TR/2002/WD-xag-20021003>. Poslední verze dostupná na <http://www.w3.org/TR/xag>.

8. Poděkování

Tento dokument byl napsán skupinou W3C Technical Architecture Group, která zahrnovala následující účastníky: Tim Berners-Lee (co-Chair, W3C), Tim Bray (Antarctica Systems), Dan Connolly (W3C), Paul Cotton (Microsoft Corporation), Roy Fielding (Day Software), Mario Jeckle (Daimler Chrysler), Chris Lilley (W3C), Noah Mendelsohn (IBM), David Orchard (BEA Systems), Norman Walsh (Sun Microsystems) a Stuart Williams (co-Chair, Hewlett-Packard).

TAG si cení mnoha příspěvků na veřejném mailing listu TAG's public mailing list, www-tag@w3.org (archív), který pomáhal zlepšit tento dokument.

Za další příspěvky velice děkujeme i těmto lidem: David Booth, Erik Bruchez, Kendall Clark, Karl Dubost, Bob DuCharme, Martin Duerst, Olivier Fehr, Al Gilman, Tim Goodwin, Elliotte Rusty Harold, Tony Hammond, Sandro Hawke, Ryan Hayes, Dominique Hazaël-Massieux, Masayasu Ishikawa, David M. Karr, Graham Klyne, Jacek Kopecky, Ken Laskey, Susan Lesch, Håkon Wium Lie, Frank Manola, Mark Nottingham, Bijan Parsia, Peter F. Patel-Schneider, David Pawson, Michael Sperberg-McQueen, Patrick Stickler a Yuxiao Zhao.

