

Záv re ná zpráva k projektu MathMLCan - David Formánek

Jako vedoucí týmu jsem koordinoval práci, rozd loval úkoly ostatním a kontroloval jejich provedení. Krom toho jsem se zabýval následujícími oblastmi:

Návrh architektury nástroje

T ídou poskytující aplika ní rozhraní pro kanonizaci je *MathMLCanonicalizer*. Kanonizátor je možné inicializovat s výchozím nastavením, s využitím konfigura ního XML souboru nebo ru ním p ídáním nakonfigurovaných instancí modul pomocí metody **addModule**. Samotná kanonizace proudu dat je pak provedena spušt ním metody **canonicalize**. Aplika ní rozhraní využívá t ída *MathMLCanonicalization*, která poskytuje rozhraní p íkazového řádku pro dávkové zpracování a jednoduché grafické rozhraní pro snadné použití.

Moduly zajiš ují jádro kanonizace, každý je zodpov dný za ur itou oblast. Modul je t ída rozši ující rozhraní *Module*, resp. v sou asnosti jedno z rozhraní *StreamModule* nebo *DOMModule*. Moduly prvního typu zpracovávají p ímo vstupní proud, ostatní pracují nad JDOM2 dokumentem. Kanonizátor vstup nejprve transformuje pomocí proudových modul , poté p evede výsledek na DOM reprezentaci, nad kterou provedou kanonizaci postupn ostatní moduly, a výstup serializuje do výstupního proudu. Jednotlivé moduly obsahují vlastní konfigura ní volby, jejichž výchozí hodnoty jsou zapsány v textovém souboru. Tato modulární architektura umož ũje použití kanonizátoru pro r zné ũly a okamžitou možnost p ízp sobení p íp ehodnocení požadavk ũ na výstup kanonizace.

Rozhraní pro testování modul

P í vývoji modul , p ídávání dalších konfigura ních voleb a optimalizaci je d ležité mít pro každý modul sadu jednotkových test ũ. S využitím knihovny *XMLUnit* bylo vytvo eno rozhraní umož ũjící snadno testovat funk nost daného modulu pouze zavoláním metody s názvem souboru (bez p ípony), který v p ípad p ípony *.original.xml* obsahuje testovaný MathML kód a v p ípad p ípony *.canonical.xml* o ekávaný výstup kanonizace. T ída *AbstractModuleTest* za ídí transformaci prvního souboru testovaným modulem (a ũž využívá proudové zpracování nebo DOM) a provede porovnání s druhým souborem na úrovni struktury XML (bílé znaky jsou ignorovány). Rovn ũž provede op tovné zpracování již kanonizovaného výstupu a zkontroluje, zda nedošlo k dalším zm ũnám (modul by m l kanonizaci provést již p í prvním pr chodu, tuto kontrolu lze ale vypnout). Lze též nastavit vypisování výstupu v pr b hu test ũ pro ru ní kontrolu.

Implementace modulu *MfencedReplacer*

Úkolem modulu *MfencedReplacer* typu DOM je vhodn nahradit výskyt všech element ũ **mfenced** v MathML dokumentu. Potomci t chto element ũ jsou ve výchozím nastavení odd leny separátory, obaleny elementem **mrow**, obklopeny závorkami a znovu obaleny **mrow**. Separátory a závorky jsou umíst ny v elementu **mo** a specifikovány v attributech nahrazovaného elementu **mfenced**. Nejsou-li p ítomny, použity jsou výchozí hodnoty. V konfiguraci je možné vypnout vkládání vnit ních a vn jších **mrow** element ũ, specifikovat vlastní výchozí hodnoty závorek a separátor ũ a také vynutit použití výchozích hodnot bez ohledu na aktuální hodnoty atribut ũ elementu **mfenced**.