DEPARTMENT OF COMPUTER SCIENCE

Supporting Website Transformation with cssML

CS-49996-199 Individual Study **Student:** Denisse Andrade **Instructor:** Michael L. Collard

CELEBRATE CENTENNIAL KENT STATE UNIVERSITY 1910·2010

Problem

• Websites get (technically) old fast. They must continually adapt to new standards and move to new technologies

• Typically consist of files in multiple formats: HTML, CSS, and JavaScript

• Currently, there is no unified format for the representation of all these file types, making transformation difficult

Solution

•A lightweight, document-oriented approach to website transformation and refactoring by means of a new format, cssML

• Supports a unified XML view for all types of website files: HTML/ CSS/JavaScript

• Based on success of srcML platform for transformation of source code

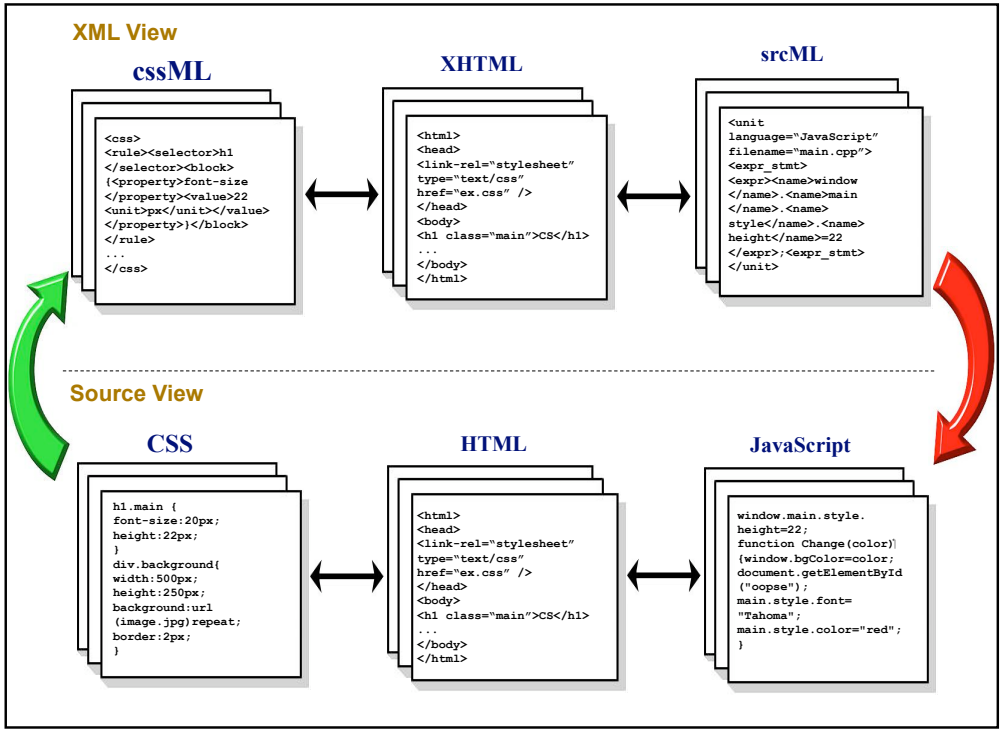
Process

• Website consists of multiple source CSS and HTML files

• Convert entire website to unified XML format using cssML

• Transform using XML transformation technologies, e.g., XSLT, DOM, SAX, etc.

• Convert from unified XML format back to source HTML and CSS



Transformations/Refactorings

• Extract CSS embedded tags to CSS external files

• Convert to new CSS standards, e.g., from CSS 2.0 to CSS 3.1

• Rename class attribute in both CSS and HTML

• Find the subset of CSS rules that affect a particular HTML page

• Remove any CSS that is not used on a set of HTML pages

Implementation

• cssML toolkit includes *css2cssml* and *cssml2css* to convert to/from CSS and cssML

• Source code is kept to the last detail: comments and whitespace are preserved when translating CSS source to cssML

• Allows leveraging of XML technologies: XPath, XQuery, XSLT, RelaxNG, XSchema, DOM, JDOM, SAX

• Preparing for submission to 11th IEEE International Symposium on Web Systems Evolution

Other Approaches

• cssxml – data-oriented format to serialize CSS. Cannot handle even minor CSS validation problems. Loses developer style, i.e., formatting, comments, etc.

• Model-driven approach based on DMS (Ricca, Tonella, Baxter)

• Abstract model-driven approach (Rossi, Urbieta)

• Modernizing JavaServer pages - (Xu, Dean)

• Refactoring Websites to a Controller Centric Architecture (Ping, Kontogiann)

Additional Applications

• Calculating website metrics, e.g., complexity, usage

• Impact analysis and cost analysis of website changes

• Validation and use of CSS according to specific CSS standards, e.g., 3.1, or to a website specific standard

• Customizing CSS for browser irregularities, e.g., IE, Firefox, Chrome

• Automatic fixes for validation errors

• Combined HTML/CSS/JavaScript transformations using srcML

• Reverse-engineering to web component technologies, e.g., XAML, XBL