

// Bolded items are private APIs

```
function extractParameter(value, parameterName) {
    // Maybe we should extend ProcessingInstruction.idl to let you get at the parameters?
    var regexp = new RegExp(parameterName + "\\s*=\\s*[\"']?([^\"]+)[\"']?");
    var match = value.match(regexp)
    if (!match)
        return null;
    return match[1];
}
```

```
function findTransformURL(node) {
    // node.nodeType should be Node.PROCESSING_INSTRUCTION_NODE
    if (node.nodeName != 'xml-stylesheet')
        return null;
    var value = node.nodeValue;
    var type = extractParameter(value, 'type');
    if (!type.match(/^text\/xsl$/i))
        return null;
    return extractParameter(value, 'href');
}
```

```
function transformDocument(stylesheet) {
    var processor = new XSLTProcessor();
    processor.importStylesheet(stylesheet);
    var transformed = processor.transformToDocument(document);
    window.replaceDocument(transformed.innerHTML, transformed.contentType);
}
```

```
function onProcessingInstructionAvailable(node) {
    var url = findTransformURL(node);
    if (!url)
        return;
    // TODO(abarth): We need to do something here to prevent a flash of unstyled content.
    // Maybe we can remove the documentElement from the document? The parser should
    // continue to build DOM under it, but the whole tree will be detached from the window.
    var xhr = new XMLHttpRequest();
    xhr.responseType = "document";
    xhr.open("GET", url);
    xhr.send();
    xhr.addEventListener("load", function() {
        // TODO(abarth): We actually need to wait for DOMContentLoaded too.
        transformDocument(xhr.responseXML);
    });
}
```

```
    }  
};
```