

# XZES40-Transformer

Zixun Lu (luzi), Shuai Peng (pengs), Elijah Voigt (voigte)

CS 461 — CS Senior Capstone — Group #40

Fall 2016

# Program Purpose and Goals



Figure: Source: Wikimedia Commons [2]



Figure: Source: Apache Software Foundation [1]

# Program Purpose and Goals

The purpose of the XZES40-Transformer is:

- ▶ Transform XML and XSLT documents into a new XML document.
- ▶ The XZES40-transformer can transform multiple documents in parallel.

# Program Purpose and Goals

The goals of XZES40-Transformer are:

- ▶ Transform XML documents faster than older XML/XSLT document transformers; achieved by adding caching and parallel computation to the transformation pipeline.
- ▶ Accessed via a web browser or CLI for end-user convenience.
- ▶ Will be compatible with multiple OS platforms (i.e., Linux, BSD, Windows).

# Program Purpose and Goals

The following technologies will be used:

- ▶ Document transformation: Apache Xalan-C++ and Xerces-C++
- ▶ Document encoding: International Components of Unicode (ICU) C++
- ▶ Web API and Website Server: Apache webserver
- ▶ CLI and API Implementations: Python
- ▶ Website interface: Bootstrap
- ▶ Platform Packaging: FPM and WiX

# Shuai Peng

The part that I responsible for:

- ▶ Transformer
- ▶ Cache
- ▶ Web interface
- ▶ Windows installation package

The part that I responsible for:

- ▶ Research and Benchmarking
- ▶ XML/XSLT Document parallel computation
- ▶ Linux installation package

# Elijah C. Voigt

I am responsible for:

- ▶ Application infrastructure and basic functionality
- ▶ Web API
- ▶ User Command Line Interface (CLI)
- ▶ BSD installation package



# Project State

- ▶ A 25GiB Debian Linux virtual machine, for software development, has been created with the following packages, libraries, and tools:
  - ▶ The Xerces, Xalan, and ICU libraries.
  - ▶ Common C/C++ Build Dependencies.
  - ▶ Git, Text Editors, Gnome and common tools.
  - ▶ The TeXLive series of packages.
- ▶ Design Documents Completed

# Problems Encountered

- ▶ Not on the same page understanding at the beginning of the term.
- ▶ Confused about the format of some documents.
- ▶ Our client has not always been as available as would be convenient.

# Retrospective; Weeks 1-3

- ▶ We met our client to discuss our project.
- ▶ We need to start working on the problem statement
- ▶ We would meet our sponsor to obtain require tools for development.

## Retrospective Week 4

- ▶ The client provided a Debian Linux development virtual machine.
- ▶ The Client Requirement document needed to be written and signed.
- ▶ Our team needed to complete the Client Requirement document and get signature from our clients.

# Retrospective Week 5

- ▶ We understood what the project is after we discussed.
- ▶ We need to met each other to understand what the project is.
- ▶ We would refactor the client requirement next week.

# Retrospective Week 6

- ▶ Our team scheduled the daily meeting and worked together on documents.
- ▶ The client was busy to contact so we get signed digitally.
- ▶ We needed to start work on the technology review.

# Retrospective Week 7

- ▶ We gave each people work and start writing.
- ▶ Some elements in technology review did not fully understand
- ▶ We met TA later for clearing the problem.

# Retrospective Week 8

- ▶ We completed the technology review and began working on the Design document.
- ▶ Our team misunderstood the IEEE format so we had to re-write the design document.
- ▶ We would try finish the rough draft of the design document the following week.



# Retrospective Week 9

- ▶ Because of Thanksgiving, we had a break.
- ▶ We finished the rough draft for design documents.
- ▶ We would meet TA to improve our design documents.

# Retrospective Week 10

- ▶ We completely restructured our Design document and turned in an unsigned version.
- ▶ We need to contact our client to get signature.
- ▶ We needed to completely write, finish, and record the progress report/presentation.

# Conclusion

Looking forward we will:

- ▶ Begin development early and make assessments often to complete an Alpha version on time.
- ▶ Benchmark competing software to understand our target performance requirements.
- ▶ Communicate more with our client in development checkups.

# Reference



*ASF Press Kit: Apache Software Foundation Logo.* URL:  
<https://www.apache.org/foundation/press/kit/>.



*Wikimedia Commons: Oregon State University Logo.* URL:  
[https://commons.wikimedia.org/wiki/File:Oregon\\_State\\_Un](https://commons.wikimedia.org/wiki/File:Oregon_State_Un)