

OpenExiProposal

- [Abstract](#)
- [Proposal](#)
- [Background](#)
- [Rationale](#)
- [Initial Goals](#)
- [Current Status](#)
 - [Meritocracy](#)
 - [Community](#)
 - [Core Developers](#)
 - [Alignment](#)
- [Known Risks](#)
 - [Orphaned products](#)
 - [Inexperience with Open Source](#)
 - [Homogenous Developers](#)
 - [Reliance on Salaried Developers](#)
 - [Relationships with Other Apache Products](#)
 - [A Excessive Fascination with the Apache Brand](#)
- [Documentation](#)
- [Initial Source](#)
- [Source and Intellectual Property Submission Plan](#)
- [External Dependencies](#)
- [Cryptography](#)
- [Required Resources](#)
 - [Mailing lists](#)
 - [Subversion Directory](#)
 - [Issue Tracking](#)
 - [Subversion Directory](#)
 - [Other Resources](#)
- [Initial Committers](#)
- [Affiliations](#)
- [Sponsors](#)
 - [Champion](#)
 - [Nominated Mentors](#)
 - [Sponsoring Entity](#)

Abstract

Efficient XML Interchange (EXI) is a forthcoming W3C Recommendation for compression and high performance decompression of XML. This standard has wide applicability to all forms of XML documents and consistently beats zip/gzip in terms of compactness. Multiple software implementations are beginning to emerge.

This work will establish a high performance open source codebase in both Java and C++ that can immediately be used in bandwidth-limited environments and other software applications that are not currently well served by XML. It may later may integrated into http servers and clients.

Proposal references:

- [Proposal Guide](#)
- [Enter The Incubator](#) (draft)
- [Jakarta Subproject Proposals](#) has interesting example information

Proposal

This proposal seeks to create a project within the Apache Software Foundation to develop an implementation of the current EXI Candidate Recommendation, and to track changes to the Candidate Recommendation as it progresses to an approved W3C standard. The initial implementation will be in Java, and a subsequent C++ implementation will follow. Once implemented the EXI standard could be used in many other Apache projects, such as the web server, web services, etc.

- The [EXI specification](#) is available at the [EXI Working Group Public Page](#).
- A [Primer on EXI](#) is available there, as are an [evaluation of the likely impacts](#) and [best practices](#).
- An [evaluation](#) and [measurement note](#) are available; these notes are a product of the [test framework results](#).

Background

Since the inception of XML, it has been noticed that a good number of data exchange application scenarios seemed to fit the use of XML very appealing, only to find XML inhibitive given its sometimes very costly inefficiency of inherent verbosity. Legacy applications involving data exchange, for example, typically use non-XML data formats (e.g. ASN.1 PER) that predate XML, are often far more efficient and in some cases hand-optimized to achieve the best performance result. When such applications attempt to harness the numerous benefits of XML, it is not unusual that they find XML helplessly bulky to adopt given the bandwidth constraints of the existing communication infrastructures that were designed with the currently used format in mind. Another example is a data-intensive mobile application for which bandwidth is at a premium and the use of XML is not very realistic due to its substantive disadvantage at bandwidth conservation. While there are some other use cases that address the bloated message size issue with general-purpose compression methods such as GZip, the application of such methods unfortunately more often than not compound the efficiency issue for those use cases aforementioned because GZip usually degrades the processing efficiency dramatically and has little or no impact on the message size when individual message is short.

Over the years, there have been developed numerous file formats purported to serve as alternative, efficient representation of XML data. W3C's (World Wide Web Consortium) XBC WG (XML Binary Characterization Working Group) in 2005 found that most, if not all of those formats are not very general in the sense that they had been each designed to target a particular problem domain and do not serve well use cases of other domains. In 2006, W3C launched the EXI (Efficient XML Interchange) WG with the charter to conduct study and formulate a single alternative format that provides utmost efficiency better than the customarily used formats (e.g. ASN.1 and GZip) do and even competes with hand-optimized formats, with broadest coverage of use cases and platforms including those that had not been well served by XML, and yet is compatible with XML and integrates well with existing XML family of standards and applications without major disruption.

As of this writing, EXI is a W3C Candidate Recommendation, and is well on its way towards becoming the W3C Recommendation around mid-2010. The status of Candidate Recommendation indicates that W3C calls for implementations of the specification in order to foster interoperability between various implementations before the technology becomes a W3C Recommendation.

Rationale

Apache, a free Web server application, is, and has been the dominant market shareholder of Web servers in the world.

The primary motivational goal for EXI is to bring to the WWW and other networks a better XML interchange to further XML Web penetration, specifically to small mobile and handheld devices. Making an EXI solution non-viral OSS encourages adoption by both individual developers and well-established corporations due to the reduced development overhead, "take this working source-code and use it as you see fit," without having to invest extensive time and effort into development. Using a license that encourages broad use can help meet the goals of EXI to make it an adopted and utilized industry binary XML standard.

The OPENER-EXI solution is best fitted with an open and free license (such as Apache) to increase the expected likelihood of widespread adoption. At the same time this grants corporations the right to customize the OPENER-EXI solution and package it into their existing products, as they see fit, for profit. Placing a non-viral free license on the OPENER-EXI code allows it to be used without restrictions with proprietary source, which should encourage the corporations to adopt the solution into their codebase. This in turn helps to deliver a wider dissemination of EXI solutions.

Initial Goals

A series of deliberate steps are needed to accomplish these important outcomes. Project goals are listed for the various planned milestones of the project:

Initial configuration and setup

- Donate existing codebases from initial contributors.
- Set up the incubation infrastructure (svn repository, build scripts, test document corpus, measurements suite, regular working group resources, etc.) to prepare for continuous development, testing and releases.

Initial integration of Java build

- Integrate the two initial codebases (schema-less implementation and schema-informed implementation) into a single consolidated codebase.
- Add core format capabilities that are missing in the existing codebases. These include support for EXI header options, built-in datatype codecs, compression options and XML Schema regular expressions.
- Make sure all core features pass the interoperability test suite already developed by W3C EXI Working Group. TODO add links at W3C and NPS
- Produce an initial release that demonstrates the core features of EXI.
- Add more format capabilities to achieve complete coverage of EXI specification. These include support for XML fragments, datatype representation map, etc. Again validate the implementation by running the interoperability test suite.

Correctness and optimization of Java build

- Produce the second major release that provides a complete implementation of all EXI features in Java.
- Measure, document and profile codebase performance using the already-created JAPEX testing framework. Optimize the codebase for compaction efficiency and decompression performance.
- Continue releases of the Java codebase until working group consensus is achieved that the implementation is well-structured, efficient and high-performance.

Create and test corresponding C++ build

- Create a corresponding C++ codebase that matches the architecture of the Java codebase. Shared improvements to the common architecture may also be valuable at this point.
- Perform testings and optimizations as necessary to achieve comparable or superior performance.
- Create an Apache HTTP module that plugs in the C++ implementation and provides all configuration settings needed to ensure proper HTTP support for EXI.
- Continue codebase development to add EXI utility packages providing common APIs similar to SAX DOM StAX etc., for both Java and C++ codebases.

- Ensure that all documentation and examples are completing, matching high quality of other Apache work

Current Status

We are collaboratively editing and discussing this proposal. Next steps:

- We are ready to discuss this incubator proposal with the [Apache Software Foundation \(ASF\)](#) on the [Apache Incubator list](#) to begin following the Apache process.
- Please contact [Stephen Williams](#) to discuss who on the Apache team might sponsor and mentor this project.
- We will also move this proposal to [Sourceforge openexi project](#), and update the website pages there to describe this new work.
- Our next teleconference for discussing this work is
 - Monday 20 December 2010 (1500 pacific GMT-8)
 - Dial +1.831.656.6500, Code 831.656.2149#

Completed progress:

- Finish draft proposal 10 November 2010 - complete
- Invitation sent to Siemens and [W3C EXI Working Group](#) members to consider participating or sponsoring - complete
- Proposal briefing and discussion planned for the [W3C EXI Working Group](#) 17 November 2010 teleconference - complete, positive response received
- Progress with Apache outreach was discussed on our 24 November 2010 teleconference
- Based on discussion on the [Apache Incubator list](#) this proposal was moved to the [Apache Incubator Wiki](#) as the [OpenEXIProposal](#) during our 6 December 2010 teleconference

Meritocracy

The people who have developed the codebases for initial contribution have ample experience with meritocracy-based engineering in multiple projects including [W3C EXI Working Group](#) and [Web3D Consortium](#) activities. In each case, standards development and deployment have been driven by open software development in partnership with commercial software development.

Meritocracy succeeds and flourishes when individual motivation and commitment are honored. People rise to the best possible levels of performance and effort when given opportunities to contribute and govern. We plan to use the principles of meritocracy so that the OpenEXI project can build the best possible results out of the community, continuously evolving to become a successful Apache project.

Community

One of the primary motivations behind the making of EXI is the desire to expand the reach of XML. As the reach extends into more applications and devices, the community's interest in OpenEXI will grow. We expect the the rate of such growth to accelerate as the community become well acquainted with EXI and starts to help promote EXI, which may enlist more people into the community. We plan to actively communicate the project with wide audience by leveraging every opportunity to engage with the public.

A sustainable community is especially important for the EXI Apache Incubator for two reasons: we want to co-evolve extremely high-performance similar implementations in C++ and Java, plus we want to achieve code that is sufficiently robust that it be used in Apache http servers everywhere. Long-term contributions, innovation and stability will be the key to such success.

Core Developers

The core developers worked on original implementations first developed independently at Fujitsu and NPS.

- Taki Kamiya
- Don [McGregor](#)
- Don Brutzman
- Stephen Williams
- Sheldon Snyder

Other candidate developers will be invited to join this effort as the incubator proposal proceeds.

Alignment

[Guide](#): "Describe why Apache is a good match for the proposal. An opportunity to highlight links with Apache [projects](#) and [development philosophy](#)."

EXI is an XML technology that integrates into the XML stack at the very bottom just below the XML Information Set, right beside XML. The primary motivation behind the notion of EXI is to help XML expand its reach further beyond its traditional application areas. Both XML and EXI are forms of representing XML Information Set, and the two are exchangeable and technically equal though it is not the intention of EXI to take the place of XML; EXI complements XML, on the contrary. OpenEXI is to EXI what Xerces has been to XML, therefore, OpenEXI and Xerces need to work in tandem and the best way to facilitate that is for OpenEXI to be incubated under the auspices of Apache to which Xerces belongs. Besides this conceptual link, OpenEXI already uses Xerces to read in XML Schemas and get access to the schema component model. With OpenEXI to work seamlessly with Xerces, the users of EXI and XML both will get benefit out of the other, the combination will allow Apache to fortify its position as the venue to provide the most useful set of technologies supporting XML foundations. We also conceive the goal of extending the Apache http server to include the EXI encoding as a high-performance alternative to XML itself.

Known Risks

The only significant known risk might be that the full amount of time needed to achieve these ambitious goals for Apache and the Web might be hard to predict. Even so, any uncertainty about overall timing is no impediment to making steady progress on OpenEXI.

Orphaned products

All the initial contributors are active members of W3C EXI Working Group, therefore have strong commitment to the success of OpenEXI project. Even in the very unlikely hypothetical case that the project had lost all initial contributors, the project will undoubtedly sustain and flourish because the community's interest in EXI will not dwindle.

EXI is a W3C Candidate Recommendation which has completed Last Call. The next phase of review is W3C Proposed Recommendation. These steps are detailed in the [W3C Process Document](#). No major unresolved technical problems are currently identified and EXI Working Group efforts are ongoing.

Inexperience with Open Source

The initial committers from NPS have an excellent track record of leading an open source project to a success. This experience will be valuable for OpenEXI project especially because the project NPS has led was also concerned with a data format. Others have varying degrees of experience with open source projects though admittedly not very extensive, however, they are all committed to the success of OpenEXI leveraging the power of Apache community and the virtue of meritocracy.

Homogenous Developers

The list of initial committers includes developers from Fujitsu and NPS. Though the two set of developers have known each other for several years, the collaboration was only through the activity of the W3C EXI Working Group. Therefore, each party should have its peculiar background that the other either runs short of or is not as proficient in. The initial contributors are based in California, U.S. Our plan is to solicit help and enlist developers from a variety of locations, backgrounds and skills.

Reliance on Salaried Developers

All the initial committers are paid by their employer to contribute to this project. The initial employers (i.e. NPS and Fujitsu) have been the members of W3C EXI Working group from its inception and remain committed to its success. Their commitment to OpenEXI is part of the broader commitment to EXI, therefore, it is expected funded proposals and salaried time will continue to be invested into OpenEXI for a long time. The individual developers, on the other hand, each have strong sense of code ownership, and their commitment to the code can be considered to transcend a single employment. In addition, our plan is to gradually morph the OpenEXI development community into a good mixture of salaried and volunteer developers to extend the longevity of the project even further and more secure.

Relationships with Other Apache Products

EXI can integrate well with many other Apache projects, and a native Apache implementation could reduce problems integrating Apache XML efforts with EXI. XML permeates many Apache projects, so a number of other connections may be possible.

A Excessive Fascination with the Apache Brand

Although we expect the Apache brand may help attract more contributors as a natural consequence of its reputation, our primary interest in starting this project is based on the factors mentioned in the Rationale section. Note that the status of EXI technology as a W3C Candidate Recommendation is independent from any affiliation with the Apache brand, and EXI is well on its way towards becoming W3C Recommendation. However, we will be sensitive to inadvertent abuse of the Apache brand and will work with the Incubator PMC and the PRC to ensure the brand policies are fully respected.

Documentation

TODO: list and link EXI specification documents here.

- Sheldon L. Snyder, *Efficient XML Interchange (EXI) Compression and Performance Benefits: Development, Implementation and Evaluation*, Master's Thesis, Naval Postgraduate School, Monterey California USA, March 2010. References: [Thesis online](#), [thesis poster](#) and [Sourceforge openexi project](#)

TODO:

- Fujitsu javadoc
- NPS OpenEXI Javadoc

Initial Source

Initial source contributions:

- Fujitsu codebase (currently private, release authorization under review)
- NPS codebase: [Open EXI](#) on Sourceforge under Apache Software License (ASL)

Other resources for comparison and testing include

- [EXI test corpus](#) of [example XML documents](#)
- [EXI Japex](#) test framework

Other EXI implementations can be used for interoperability and round-trip comparison testing. Such implementations include

- [Exificient](#) is an independent Java implementation of EXI under the Gnu Public License (GPL)
- [AgileDelta](#) produces commercial implementations in C++ and Java

Source and Intellectual Property Submission Plan

- Fujitsu codebase will be placed under the Apache Software License (ASL) v2.0
- NPS codebase is under ASL v2.0
- EXI test corpus of example XML documents is under the W3C software license
- EXT Japex test framework license?

TODO integrate links

TODO precautions about not using other open source code that might contain patented algorithms

External Dependencies

- [xsdregex](#) from Thai Open Source Software Center (BSD license)

Cryptography

No cryptography code is directly associated with the EXI codebase.

Usage of EXI compression has been tested in conjunction with XML Encryption and XML Signature Recommendations using the corresponding Apache libraries and Bouncy Castle cryptographic libraries.

- EXI Likely Impacts
- Snyder thesis
- Williams thesis

TODO add further details and links.

Required Resources

Mailing lists

We request that an apache mailing list be created for this project.

Other lists of interest:

- A sourceforge mailing list already exists for the NPS Opener-EXI sample implementation.
- The EXI working group has a members-only and public mailing list.

TODO proposed name, links

Subversion Directory

We request that an apache subversion directory be created for this project.

Other version-control directories of interest:

- A sourceforge subversion directory already exists for the NPS Opener-EXI sample implementation as part of the [Sourceforge openexi project](#).
- The EXI working group has a members-only cvs directories for the XML examples test corpus and also for the japex text framework.

TODO proposed name, links

Issue Tracking

We request that an apache issue tracker be created for this project.

Other issue trackers of interest:

- A sourceforge issue tracker already exists for the NPS Opener-EXI sample implementation.
- The W3C EXI working group has a members-only issue tracker for the XML examples test corpus and also for the japex text framework.

TODO proposed name, links

Subversion Directory

We request that an apache issue tracker be created for this project.

Other issue trackers of interest:

- A sourceforge issue tracker already exists for the NPS Opener-EXI sample implementation.
- The W3C EXI working group has a members-only issue tracker for the XML examples test corpus and also for the japex text framework.

TODO name, links

Other Resources

Initial Committers

- Taki Kamiya
- Don [McGregor](#)
- Don Brutzman
- Stephen Williams
- Sheldon Snyder

Affiliations

[Fujitsu](#)

- Taki Kamiya

[Naval Postgraduate School \(NPS\), U.S. Navy](#)

- Don [McGregor](#)
- Don Brutzman
- Sheldon Snyder U.S. Navy (NPS graduate, probably observer role)

[OptimaLogic](#)

- Stephen Williams

Sponsors

NPS is actively soliciting sponsorship for further programming work. Please contact [Don Brutzman](#) if you or your company are interested in helping support these efforts.

Champion

TODO: we need to identify an [Apache Champion](#).

Please contact [Stephen Williams](#) to discuss who on the Apache team might sponsor and mentor this project.

Nominated Mentors

TODO: The Apache Sponsor will need to identify [Nominated Mentors](#) for this incubator.

Please contact [Stephen Williams](#) to discuss who on the Apache team might sponsor and mentor this project.

Sponsoring Entity

TODO: we expect that our initial [Sponsoring Entity](#) is the Apache Incubator project.