Todo List

Table of contents

1. Long Term

- **[xml] Compressed DOM: Huge document support** We need to create a virtualized byte array class to drive the compressed DOM system. Right now, there's a single byte array that represents the entire compressed document, but if that byte array is huge, and the document doesn't need to be fully traversed, then most of it is hanging around in memory for no reason. # open
- **[xml] Basic Document-level transactions** We need to provide the ability to lock a node (and recursively everything beneath it) to support node-level transactions. This locking should allow for both single writer/many reader and exclusive locking. # open
- **[xml] Scheduling** An Application-level scheduler needs to be developed for performing automated management tasks. Events will fall into several classes, including XMLObject method calls and external command execution. # open
- **[xml] The Query Engine** The query engine has basic functionality right now. Indexing and XPath query work against a Collection, but no unified cross-Collection query system currently exists. # open
- **[xml] Grouped resource-level transactions** A system needs to be designed that allows transactions across a set of resources to be performed. First step towards implementing transaction support is implementation of the document locking mechanism. # open
- **[xml] XQuery support** The Xindice projects needs to work with the Xalan project in order to produce a common Apache Foundation XQuery implementation. # open

2. High Priority

- **[code] WebAdmin** Integrate new WebDAV capable WebAdmin into the main codebase. # open
- **[code] Caching** Revisit implementation of documents cache in the core Collection class. Clarify caching semantics, implement caching for all use cases (compressed, uncompressed, binary objects). Ensure dirty data can not be placed in the cache. # open
- **[code] XML-RPC** Extend Xindice XML-RPC API to optimize transmission of collection's symbol table between server and the client for reduced traffic and improved performance. # open

3. Medium Priority

- **[code] Entity Catalogue** Xindice should have internal XML entity catalogue and utilize it. Catalogue management operations should be made available via Collection API. # open
- **[code] Authentication** Xindice server should support collection level authentication as per XML:DB Database.getCollection method. # open

- **[code] Authorization** Xindice should support collection level and document level authorization of users and groups. # open
- **[code] Administration** Xindice should provide administration tools for database consistency check, recovery, export/import. # open
- **[code] Specifications** We need to continue fleshing out our technical specifications for the core. # open
- **[code] Javadocs** We need to continue fleshing out our technical specifications for the core. # open