PolyVersion

Package in package 'PolyVersion'

<div align="right">

PolyVersion
Version 1.0 Phase 1.0 Proposed
rozho created on 2019-04-03. Last modified 2019-04-03

</div>

Visitor diagram

Class diagram in package 'PolyVersion'

<div align="right">

Visitor
Version 1.0
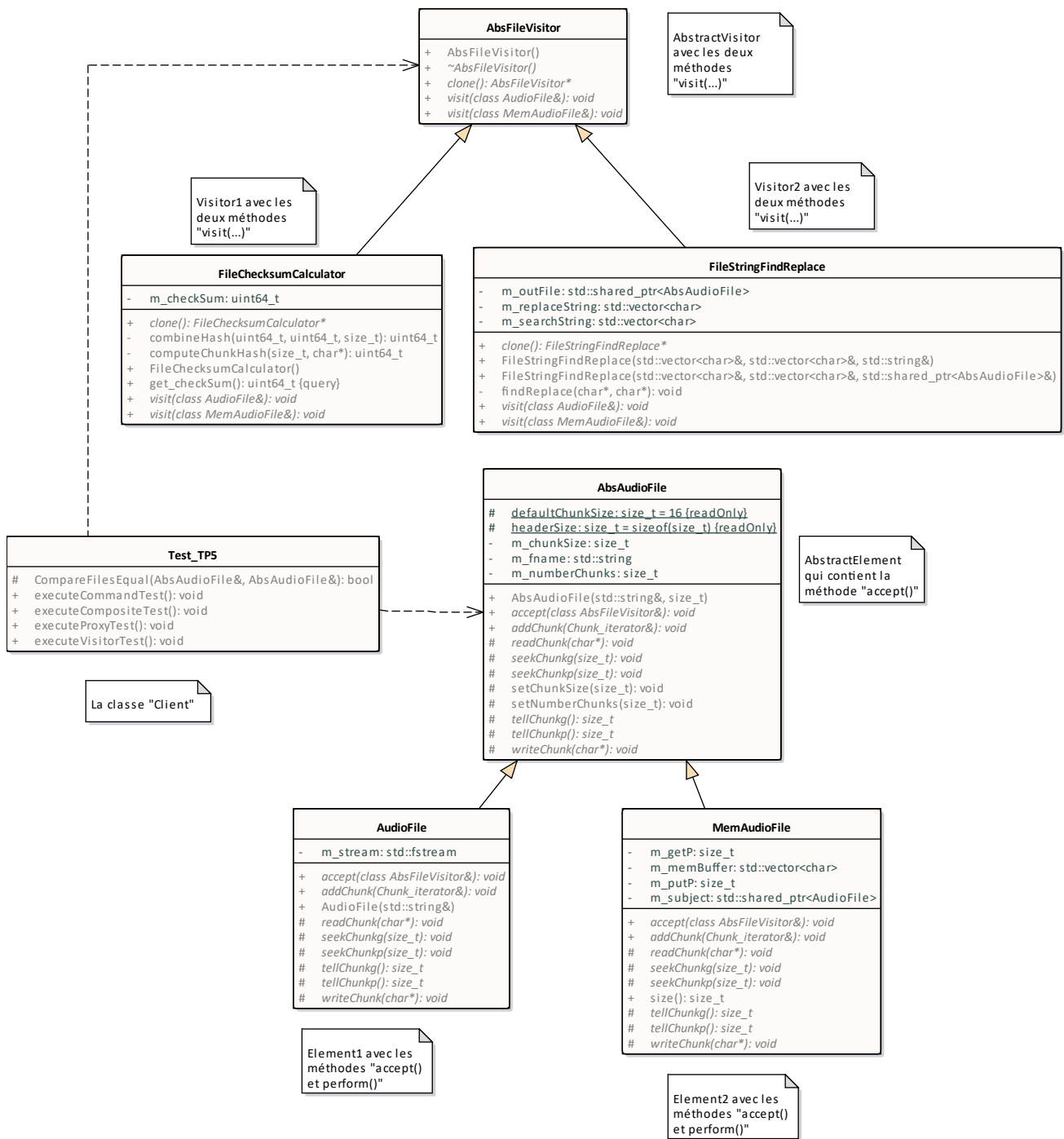rozho created on 2019-04-03. Last modified 2019-04-03

</div>

PolyVersion

Package in package 'PolyVersion'

AbsFileVisitor
+   AbsFileVisitor()
+   ~AbsFileVisitor()
+   clone(): AbsFileVisitor*
+   visit(class AudioFile&): void
+   visit(class MemAudioFile&): void

AbstractVisitor avec les deux méthodes "visit(...)"

Visitor1 avec les deux méthodes "visit(...)"

Visitor2 avec les deux méthodes "visit(...)"

FileChecksumCalculator
-    m_checkSum: uint64_t
+   clone(): FileChecksumCalculator*
-    combineHash(uint64_t, uint64_t, size_t): uint64_t
-    computeChunkHash(size_t, char*): uint64_t
+   FileChecksumCalculator()
+   get_checkSum(): uint64_t {query}
+   visit(class AudioFile&): void
+   visit(class MemAudioFile&): void

FileStringFindReplace
-    m_outFile: std::shared_ptr<AbsAudioFile>
-    m_replaceString: std::vector<char>
-    m_searchString: std::vector<char>
+   clone(): FileStringFindReplace*
+   FileStringFindReplace(std::vector<char>&, std::vector<char>&, std::string&)
+   FileStringFindReplace(std::vector<char>&, std::vector<char>&, std::shared_ptr<AbsAudioFile>&)
findReplace(char*, char*): void
+   visit(class AudioFile&): void
+   visit(class MemAudioFile&): void

AbsAudioFile
defaultChunkSize: size_t = 16 {readOnly}
headerSize: size_t = sizeof(size_t) {readOnly}
-    m_chunkSize: size_t
-    m_fname: std::string
-    m_numberChunks: size_t
+   AbsAudioFile(std::string&, size_t)
+   accept(class AbsFileVisitor&): void
+   addChunk(Chunk_iterator&): void
#   readChunk(char*): void
#   seekChunkg(size_t): void
#   seekChunkp(size_t): void
#   setChunkSize(size_t): void
#   setNumberChunks(size_t): void
#   tellChunkg(): size_t
#   tellChunkp(): size_t
#   writeChunk(char*): void

AbstractElement qui contient la méthode "accept()"

Test_TP5
CompareFilesEqual(AbsAudioFile&, AbsAudioFile&): bool
+   executeCommandTest(): void
+   executeCompositeTest(): void
+   executeProxyTest(): void
+   executeVisitorTest(): void

La classe "Client"

AudioFile
-    m_stream: std::fstream
+   accept(class AbsFileVisitor&): void
+   addChunk(Chunk_iterator&): void
+   AudioFile(std::string&)
#   readChunk(char*): void
#   seekChunkg(size_t): void
#   seekChunkp(size_t): void
#   tellChunkg(): size_t
#   tellChunkp(): size_t
#   writeChunk(char*): void

Element1 avec les méthodes "accept() et perform()"

MemAudioFile
-    m_getP: size_t
-    m_memBuffer: std::vector<char>
-    m_putP: size_t
-    m_subject: std::shared_ptr<AudioFile>
+   accept(class AbsFileVisitor&): void
+   addChunk(Chunk_iterator&): void
#   readChunk(char*): void
#   seekChunkg(size_t): void
#   seekChunkp(size_t): void
+   size(): size_t
#   tellChunkg(): size_t
#   tellChunkp(): size_t
#   writeChunk(char*): void

Element2 avec les méthodes "accept() et perform()"

Figure 1: Visitor

ResultatTest

Enumeration «Enumeration» in package 'PolyVersion'