CRevision.cpp

Fa felállítása, melyik melyikből öröklődik.

**interface/IBitList.h**

**data/CRevision.h**

**IBitList.h**-->CSoDAIO, CBitReader, CBitWriter, Sodalibdefs

-->io/csodaio.h -->io/cbinaryio.h, data/Sodalibdefs.h

io/cbinaryyio.h -->data/sodalibdefs.h PIPA

-->io/CBitReader.h-->io/CBinaryio.h-->PIPA

-->io/cbinaryyio.h -->data/sodalibdefs.h PIPA

-->io/CbitWriter.h-->io/CBinaryio.h-->PIPA

--/data/sodalibdefs-->PIPA

**CRevision.h**-->io/CSoDAio, data/SodaLibDefs.h PIPA

///////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

**CRevision.h-->used functions() and etc:**

Ebbe mindent bennehagyam (nemtudtam melyik nem kell), ez végett a CPP-ben is mindent bennehagytam pl(writeInt4() stb)

**CRevision.cpp**

**-header alapján itt is kitörölni ami nem kell, ami megmaradt azokat a megvalósításokat kellett megvizsgálni, hogy melyik rész, honnan jön, melyik headerből, odaírni melléjük is kommentbe.**

**így egyszerűen átláthatóvá válik a lebontott projekt. Ez alapján:**

CRevision.cpp-->vectors, etc. -->SoDALibDefs.h

CRevision.cpp-->count()--> IBitList.h

CRevision.cpp--> at()---> IBitList.h

CRevision.cpp--> erase() -🡪 IBitList.h

CRevision.cpp-->push\_back() -🡪IBitList.h

CRevision.cpp-->begin(),end(),size(),clear()-🡪IBitList.h

CRevision.cpp-->writeInt4() , writeULongLong8(), writeUInt4(), writeData(), readUInt4(), readData()🡪 CBinaryIO

**Ezekbe a headerekbe belelépve, kikommentelni azokat amik nem kellenek ezekhez a functionokhoz (lehet hogy később mégis kelleni fog így elég csak a kommentelést visszavonni), csökkentve így a headerek méretét, ezáltal az interfaceba is kevesebb dolgot kell felülvizsgálni.**

**CBitReader, CBitWriter --- ezt kivettem nem tudtam hogy kell –e**

**A SoDALibDefs mindenképp kell**