

However, I have some minor concerns:

- > ...from a database/object-oriented standpoint (normalization), there
- > should be no circular references.

This could even be a general thesis. But it is rather a theoretic, "nice to have" rule with not much practical background.[...]

I still would agree with your thesis but not as a strict rule, rather in a "recommendational", nice-to-have kind.

I agree. And if I may go off topic just a little bit: In some cases de-normalization (circular references are one example of that, redundant storage of data another) makes a lot of sense. Normalization often increases the complexity of the structures, increasing runtime and making them difficult to understand. And anybody who has been a victim of "overnormalization" -- in a database context it usually means waiting a very long time for some operation to finish, then calculating that it will take X million years to do so, and then ask the developers more or less politely what they have been thinking when designing this :razz: -- will agree that normalization is more a guideline and not an axiom.

Having said this, in this specific I think normalization makes sense and has no drawbacks.

---