C++ Programming

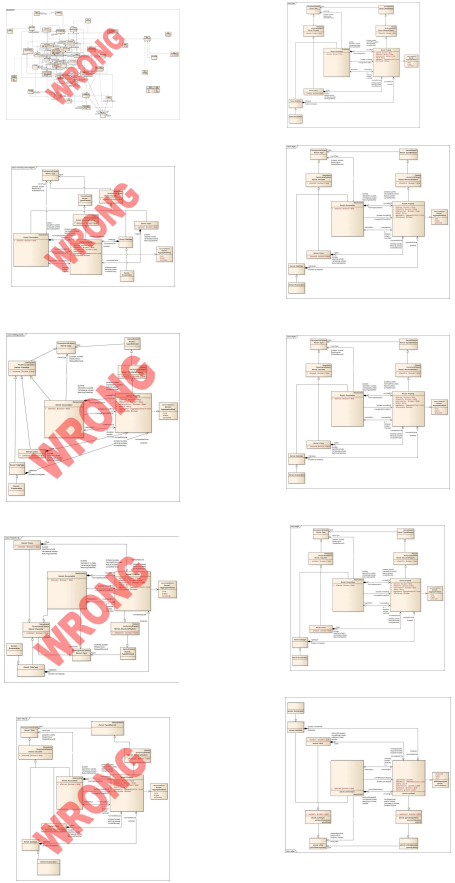
UML in Practice

Mostafa S. Ibrahim
Teaching, Training and Coaching since more than a decade!

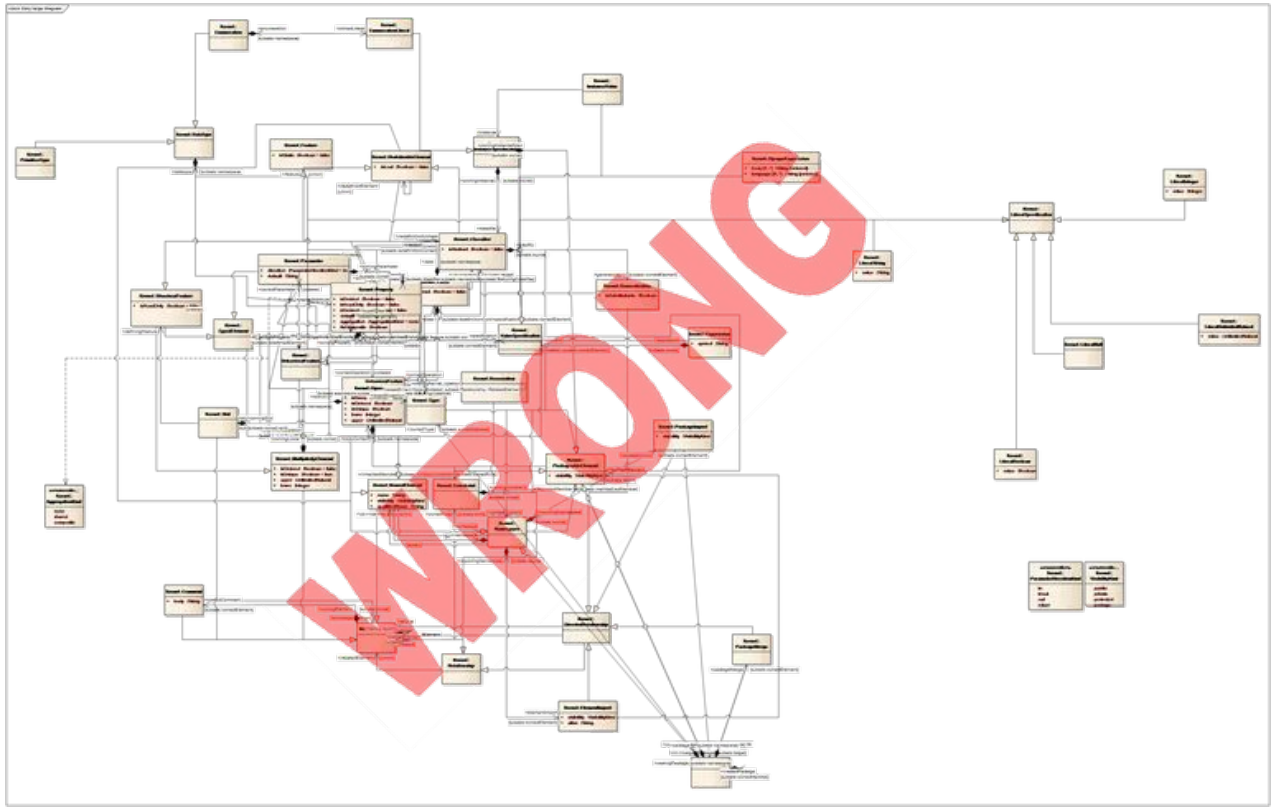
Artificial Intelligence & Computer Vision Researcher
PhD from Simon Fraser University - Canada
Bachelor / Msc from Cairo University - Egypt
Ex-(Software Engineer / ICPC World Finalist)

Class Diagram: UML Best Practice

- Many diagrams can be so useless / hard to get
- Future: reading reading
- 5 Tips
  - Less is more (Providing a lot just confuse)
  - No Crossings (Don’t cross lines)
  - Orthogonality (All lines vertical or horizontal)
  - Parents Up (Inheritance parent always above)
  - Tidy Up (Clean view, e.g. alignments)

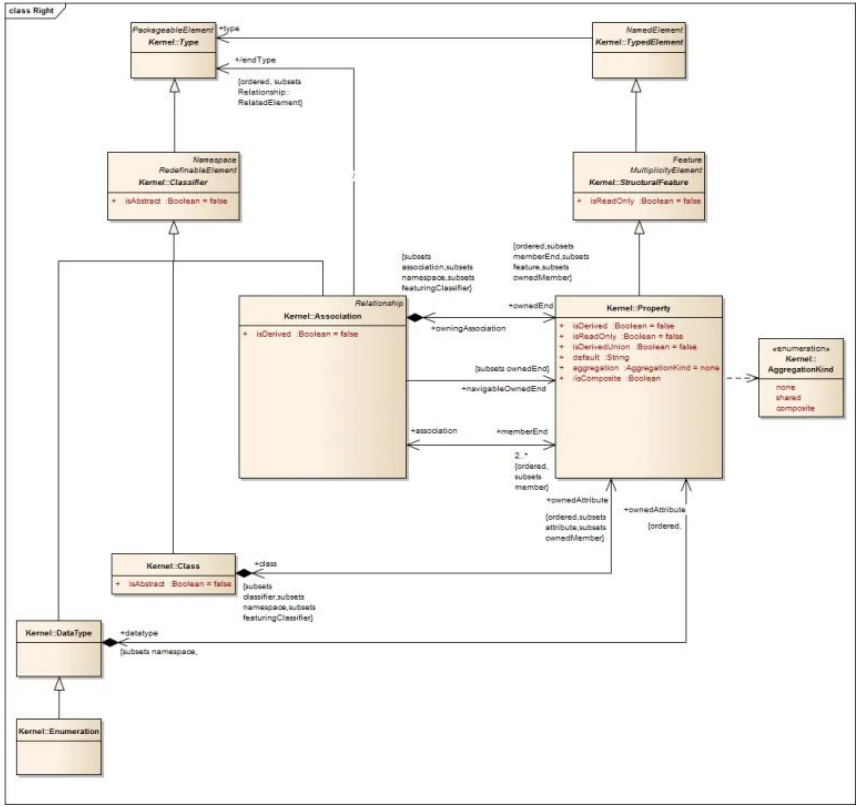


Rule #1: Less is more



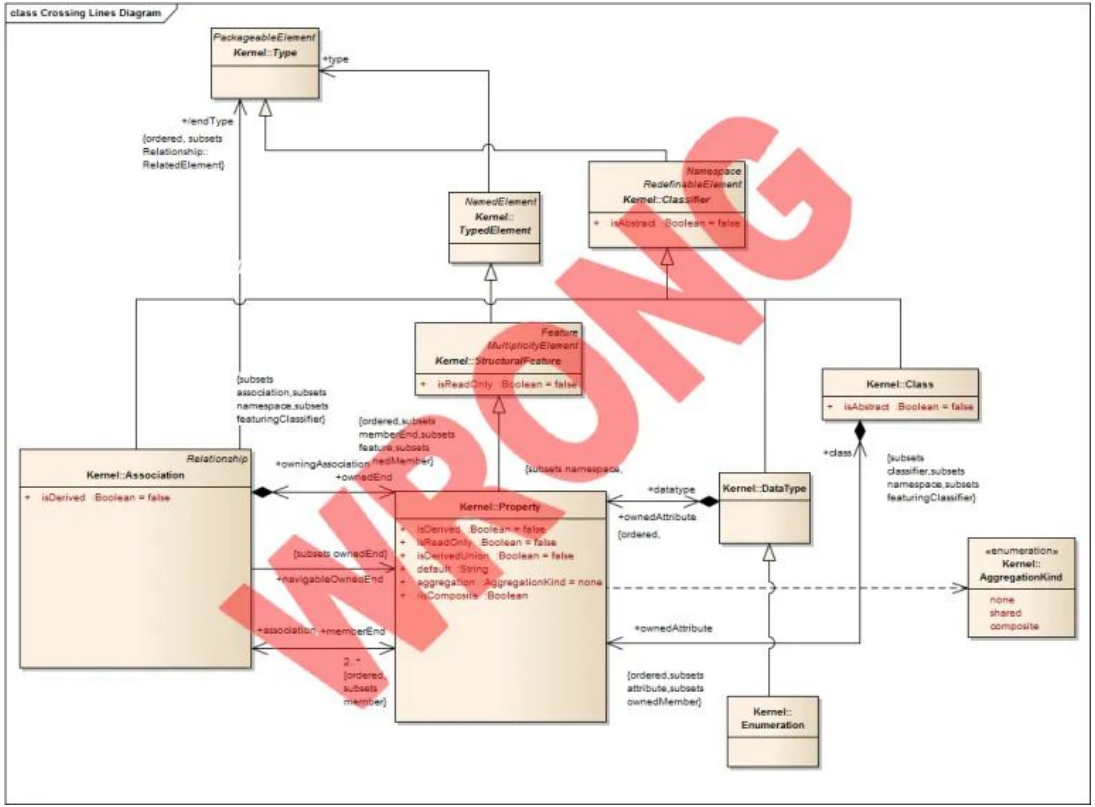
Img Src

Rule #1: Less is more



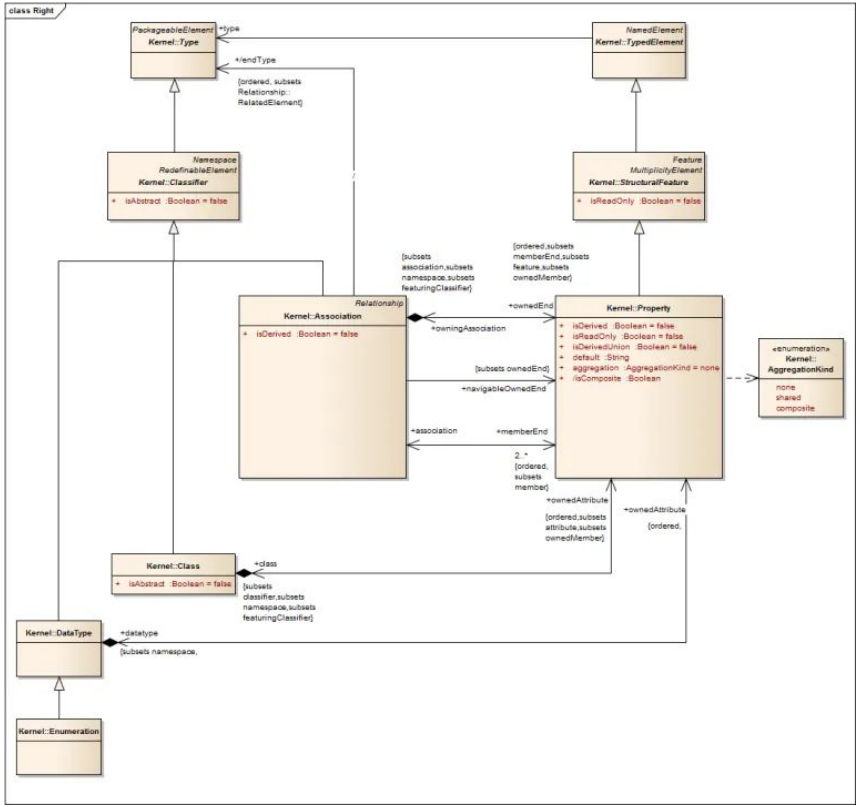
Img Src

Rule #2: No Crossings



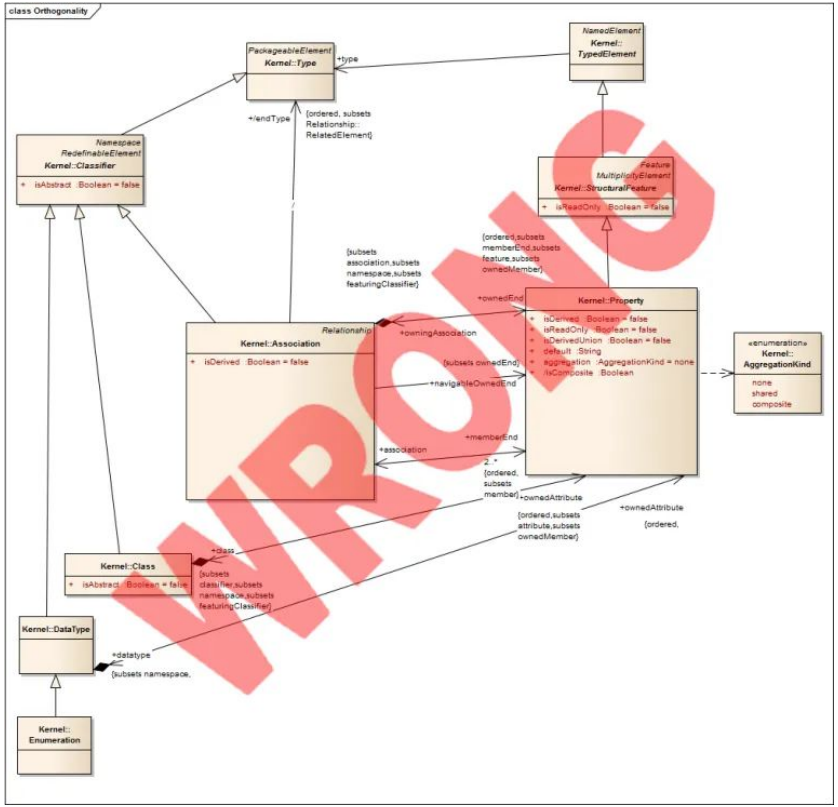
Img Src

Rule #2: No Crossings



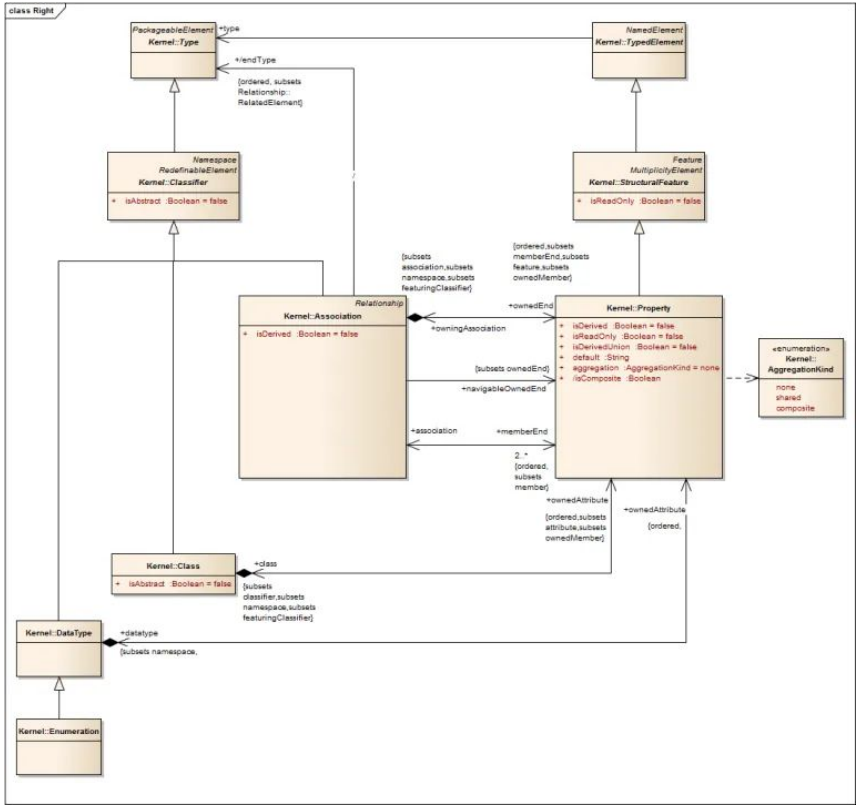
Img Src

Rule #3: Orthogonality



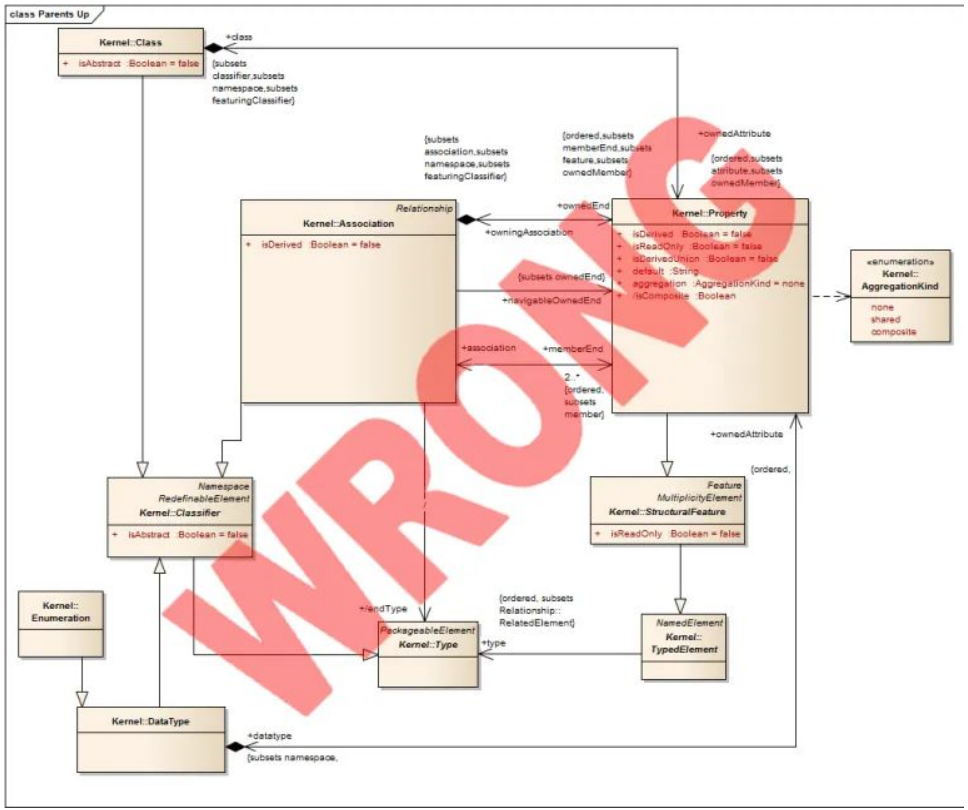
Img Src

Rule #3: Orthogonality



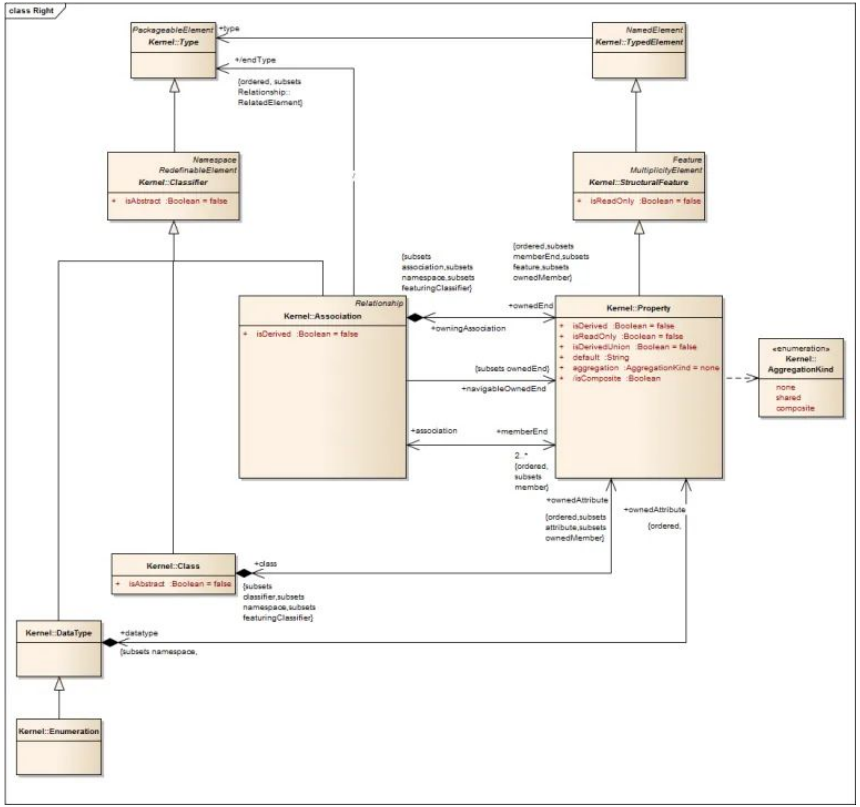
Img Src

Rule #4: Parents Up



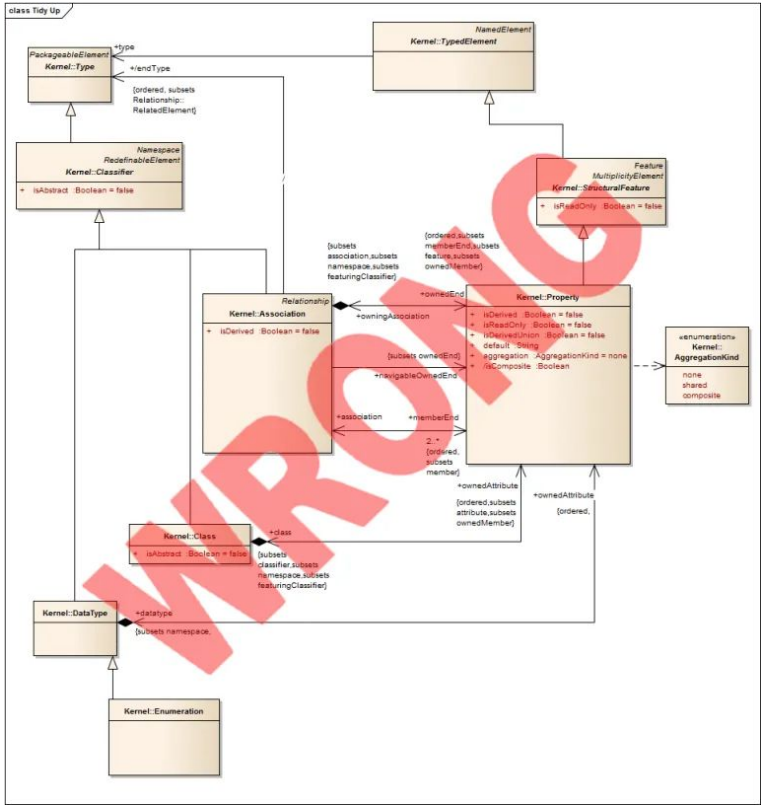
Img Src

Rule #4: Parents Up



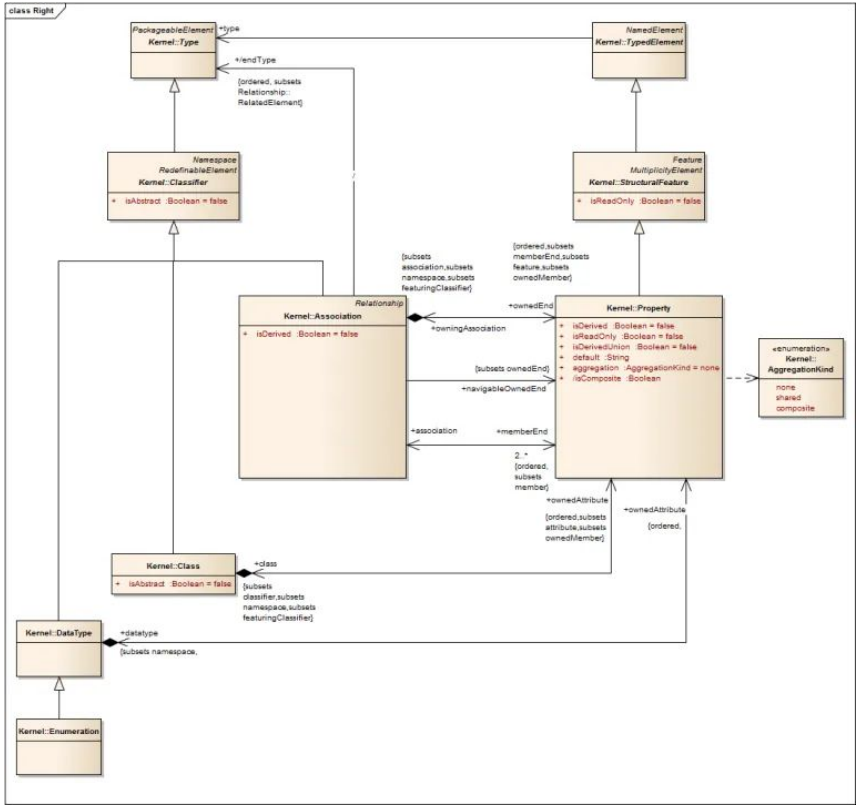
Img Src

Rule #5: Tidy Up



Img Src

Rule #5: Tidy Up



Img Src

UML in Practice

- Some companies never/rarely use
  - Just use a **whiteboard**. Team brainstorms and draws to communicate thoughts
  - New employee? A bit trouble. Some high level explanation + code deep dive
- In many small projects (3-6 month), maybe no diagrams
- Big projects: create some **high level** diagrams
  - Most important: [Class - Sequence - State - Activity] **Diagrams**
- Agile challenge: Diagrams will be outdated soon due to rapid changes
- Tips
  - Learn the notations & diagrams. Think twice before creating diagram. Focus on high level
- Future readings
  - Reading reading reading

"Acquire knowledge and impart it to the people."

"Seek knowledge from the Cradle to the Grave."