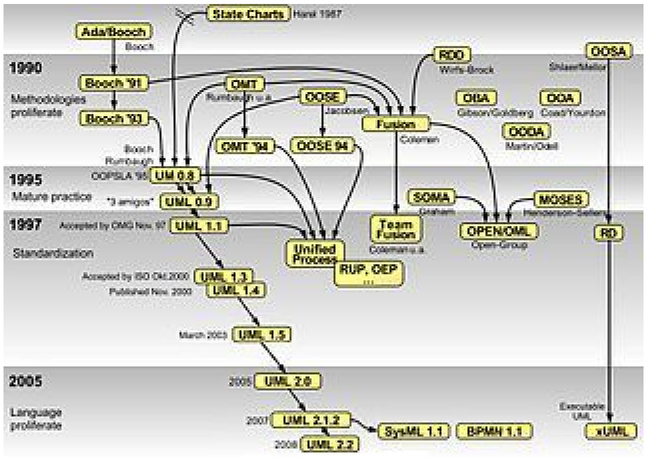
Dylan Brown
10/25/19
CS-250
Rosasco

Why UML?

Dear Boss,

Below on my research on UML and why we should be using it.

Unified modelling language, or for short UML, is a “standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems” (Visual Paradigm). UML was created to help combine large sets of industry standards. Before UML, the standard for modelling looked something like this:



As you can see engineers, software developers, and even other professions all need to model solutions to their complex problems. Before UML was created, we had a hodgepodge of different solutions. These different models were not compatible or universal. If two companies needed to communicate, there was an extremely large barrier in terms of their models. UML was created to solve this problem. UML combines some of the best parts of the diverse modelling solutions, and adds some of its unique ideas to modeling. We use UML because of this standardization. It's something that communicates between companies, industries, and even languages.

Our company needs to be using UML for a multitude of reasons. First, our company is large and we have so many diverse minds on our teams. Without a standard, there is no way for us to communicate effectively. UML has layouts that all of our employees can find their great ideas into. These layouts include, but are not limited to: class diagrams, object diagrams, timing diagrams, state machines, and use case diagrams (smartdraw). This way, when our projects are going from development to testing to deployment, we can always track how our product works.

Second of all, UML makes large-scale decisions less risky. When we model our products before deploying new and improved features, w e can very easily see the permutations. It allows people from other departments to look at a piece of the product and break down how it is going to work. Not only can they catch logical errors, they can also add new ideas and improvements for our design. When we model our product out with great detail, we create a culture of creativity. While doing this, we can easily manage our risks (Thomas). We can model changes to our software, write the testing, and complete the task. If something was to escape our model, going back to the old version would be easy with our model and the tools of Github.

I want to end this email with a quote from one of the inspirational members behind the creation of UML. Grady Booch, author of the second edition of the Unified Modeling Language User Guide, once said

> “Modeling is a central part of all the activities that lead up to the deployment of good software. We build models to communicate the desired structure and behavior of our system. We build models to visualize and control the system's architecture. We build models to better understand the system we are building, often exposing opportunities for simplification and reuse. And we build models to manage risk" (Visual Paradigm).

I thank you for reading this, and I hope you understand why we should use UML.

Dylan Brown

References

Thomas, Dunstan. “Why Use UML? Why Deploy the Unified Modelling Language.” *Dunstan Thomas Consulting*, dthomas-software.co.uk/resources/frequently-asked-questions/why-use-uml-2/.

“UML Diagram.” *Smartdraw*, www.smartdraw.com/uml-diagram/#WHYUseUML.

“What Is UML.” *Unified Modeling Language*, Object Management Group, www.uml.org/what-is-uml.htm.

“What Is Unified Modeling Language (UML)?” *Visual Paradigm*, www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/.