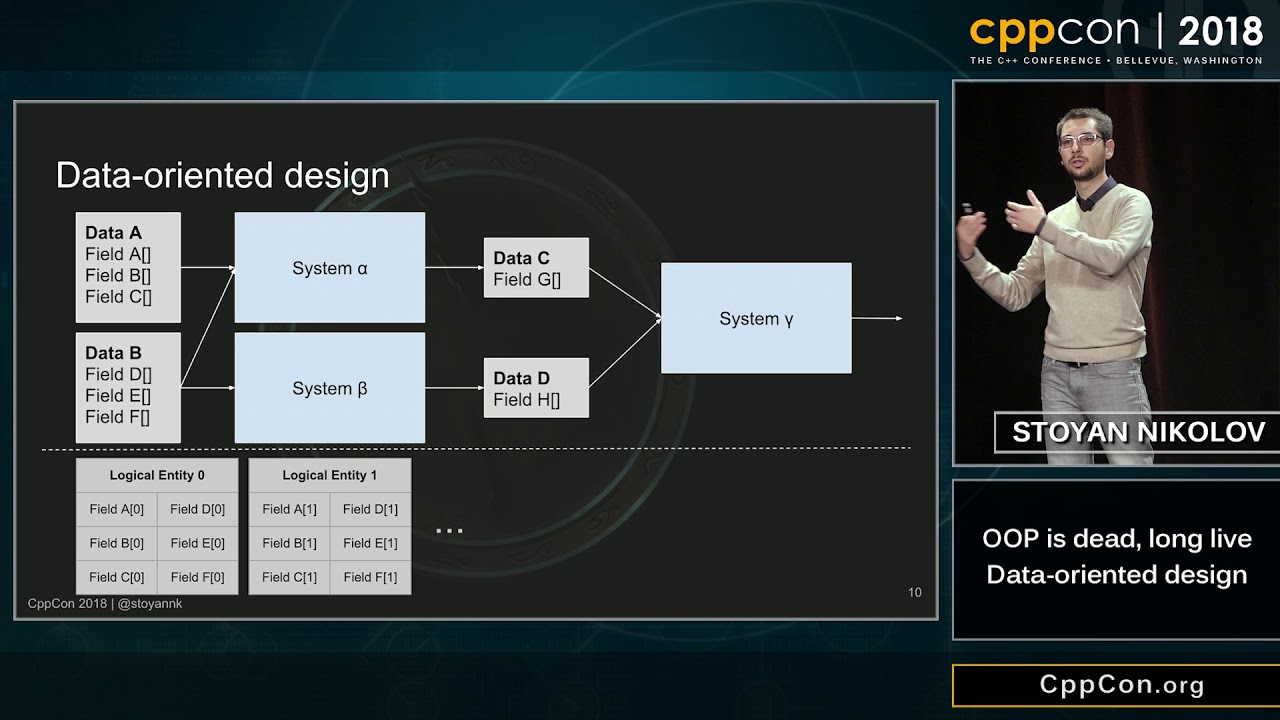
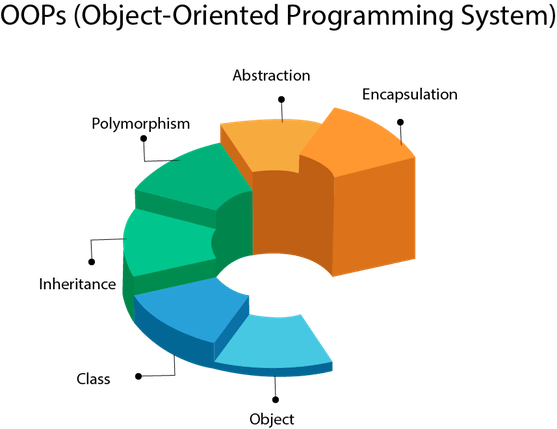
Alternatives to OOP

By Kimon Togrou



Contents

[Introduction 2](#_Toc7883845)

[Problem Definition 2](#_Toc7883846)

[Method 3](#_Toc7883847)

[Planning 3](#_Toc7883848)

[What is Data-Oriented-Design (DoD)? 4](#_Toc7883849)

[Preface: 4](#_Toc7883850)

[DoD: 4](#_Toc7883851)

[What are the benefits of DoD? 5](#_Toc7883852)

## Introduction

First off, I choose to write this in English because of all the technical terms that I’m going to be dealing with and the fact that everything I’ll be studying on the subject is in English.  
I have chosen this subject because I feel that we get taught to think in a very specific way when it comes to developing software (Object-Oriented), and if I look on the web there is an increasing amount of very prominent and very experienced engineers that are speaking against this way of writing software so I thought that it would be an interesting area to look into why that is if it’s cultural or some other reason. I want to figure out what, we can learn from these people and if what they’re saying holds any water so to say.

## Problem Definition

Although this is a very hard and technical subject, I am hoping to have answered all these questions by the end of these weeks:

* What is Data-Oriented-Design (DoD)?
* What are the benefits of DoD vs OOP?
* Can you write Data-Oriented in C#?
* Which languages suit themselves best to DoD?
* What kind of project/company should use which design method?
* What is the difference between Array of Structs (AoS) and Structs of Arrays (SoA) and why is it relevant?
* Does DoD have to compromise structure and maintainability for performance?

## Method

As this is a very theoretical and heavy subject and there isn’t a lot of readily available information on it, therefore most of my work will consist of research although I am hoping to produce at least 1 or 2 code examples.

* I will watch all the talks I can find on the subject (which are only about a handful)
* I’ll be making a code example to test some SoA vs AoS and see what runs the fastest
* Read forums and study the few practical examples I can find
* I would like to have a program developed in OOP and then make another version where I will try to apply DoD principles to look at any perceivable differences and code metrics

## Planning

I have four weeks to hand in this assignment, but I have spent first 3 days trying to come up with an angle for this subject and looking for sources so I will be planning the remaining days:

First Week:

* May 1st – 3rd: Watch talks and general reading/research fundamentals

Second Week:

* Write about the fundamentals
* Research benefits of DoD
* Research on DoD within C# and other languages

Third Week:

* Write about benefits of DoD
* Write about which languages support DoD best
* Write about which kind of projects/companies fits DoD the best
* Research AoS vs SoA

Fourth Week

* Write a code example for AoS vs SoA
* Write a bigger code example if possible and compare metrics

## What is Data-Oriented-Design (DoD)?

### Preface:

Let’s start by lightly defining Object-Oriented-Programming (OOP) first so we have something a point of comparison.  
OOP was established as a way of writing software, that could be written from world modeling. It’s the presumption, that if we have different kinds objects like three types of “chairs” in our program there must be some defining general characteristics, which we perceive, when we look at it in the real world, that make a “chair” a “chair”, and therefore we should make a base class that hold the general info that all “chair” share and all the individual objects should be derived from that base class. Basically, software/code is the platform, and it’s putting more importance on the fact, that we as people can find the ideal way to abstractly represent the data than on the fact that computers and more specifically CPU’s want the data laid out in a specific way. There is more to it than this of course, but this will do for now.

### DoD[[1]](#footnote-1):

Most adherents to this philosophy have a very good point. Their point is that, since the purpose of all programs is to transform data from one form to another, the way we write programs should reflect the most efficient ways of doing this. It’s the presumption that because all we are doing is transforming data from one form to another, that if we don’t understand the cost of solving that specific problem, we don’t understand the problem. In other words; the hardware is the platform. We have a finite range of hardware constraints, and so we should make the most of it. From their point of view world modeling leads to huge programs that are so separated and abstract with unrelated data structures and transforms that tries to idealize the problem rather than solving the very specific data transformation that is required in the best/fastest way. That the programmer’s responsibility is for the data and not the code, the code is simply a tool to transform one set of data to another.

Many of them believe that OOP leads to:

* Poor Performance
* Poor Stability
* Poor Testability
* Encourages AoS (Array of Struct) pattern which is said to be slow

And more but these 4 negatives seem to be universally talked about.

## What are the benefits of DoD?

* It’s all about the data, which means you’re know you’re dealing with the actual problem of transforming data efficiently
* Cache utilization, DoD makes you lay out the data in a contiguous and homogeneous way which is what the CPU ultimately wants
* Parallelization becomes much easier because you localize the problems with small functions that have input and output data which is easy to get running on multiple threads
* Testing becomes a lot simpler too for the same reason. When you are largely dealing with low level data transformations like small functions that have an input and an output, it’s easy to check if the data transformation went as it should
* Modularity is a big bonus as there aren’t as many dependency problems, like classes that are interconnected in very complicated and abstract ways, this makes it so you can move pieces around more freely if you need to experiment

## Which languages are most suited to DoD?

So before trying to answer this question I wanted to know if I could use C# to write in a DoD manner, so was it possible?  
What I quickly found out by reading some forum posts was that yes you can, but the language is really not encouraging it, but neither are any other currently popular languages. The truth is that it is mostly a way of reasoning about the problem of efficient data transformation. The biggest change in the way you write code other than avoiding the world modeling approach seems to be the change from laying out your data in AoS to SoA.

The only language I have been able to find that focuses on DoD is a new programming language currently in closed beta. It’s call “jai” and it’s being developed by a very prominent developer in the games industry called Jonathan Blow

1. This is mostly based on Mike Actons talk at CppCon 2014 as he, as far as I can tell was the one that coined the term. [↑](#footnote-ref-1)