

## Participation Summary

### Ariane 5

One of real word problems that was presented in lecture was the Ariane 5 rocket failure. The system didn't recognize that the values being read was not supposed to be processed and rather ignored. Specifically, the value was being read and passed as a 64 bit floating point and then converted to 16 bit integer. However, the trouble was the sensor value was too big to convert to 16 bit integer and caused an overflow error. This caused the system to read a value that wasn't supposed to be used since it was not the conversion or velocity of the rocket. It was instead just a numeric value that is a diagnostic code indicating overflow error. So, the conversion process recognized an invalid value that was too big, and caused an overflow exception and returned an integer indicating the exception. Since it was an integer, the rocket system thought that was the actual conversion value. Overall the problem was that the rocket system didn't account for an error happening in the sensor and conversion and didn't assess the appropriateness of the values.

This problem is connected to the benefits of service orientation and designing for reliability. The thing with the rocket is that it probably used a reusable code component to read in the sensor value. However, it didn't have a valid exception management. This goes into the topic of reusable exception management and reuse certification. A reusable module in this case, should assess the appropriateness of a value read from the sensor. In no way should that module make an inappropriate value appropriate. An example of this would be adding a minimum and maximum constant value, say for the velocity of the rocket. If the value passed in was outside the min and max values, then it would deem that value to be inappropriate and not process it. This is just one of the ways to manage exceptions and make sure no critical error occurs. For mission critical systems such as this, designing for reliability is a requirement. However, there are costs and benefits of achieving reliability and those must be assessed. Various benefits from reliability includes system performance and error free environments. However, the less errors in a system, the higher the costs of finding them. So, there's a balance between pursuing reliability and costs. In the case of service orientation, that could also have been done in Ariane 5. In that case, it separated the concerns in the software to more manageable components. The system would then be composed of mostly reusable components. These components are non-context specific, composable, encapsulated and has its own independent deployment and versioning each designed to solve an individual concern. In the case of Ariane 5, one would look at the component that made the error. That single component may have been the only faulty component in the whole system. In that sense, the people that maintained and verified that component is at fault.

### IIT Crosswalk

One of the personal experiences brought up was from using the crosswalk in IIT on State Street. The crosswalks only contained signs that highlights a pedestrian and cars should stop for pedestrians and let them cross. This can cause a major problem since drivers are not obligated to yield for pedestrians, or at the very least some drivers don't. So, as a pedestrian you still have to be really careful even with all these signs saying drivers should yield and stop. Drivers also have to be wary of pedestrians and take precaution as they might be speeding through or letting the cars go by. This makes the flow of

both the pedestrians and cars all over the place. It's not very efficient or very safe for both cars and pedestrians. One solution that makes crosswalks such as these more safe is to make an autonomous system that is more apparent and helpful. This would involve getting sensors to detect cars and pedestrians and showing signals to indicate who should keep going, the car or pedestrian. The system in a way would behave like a traffic light but less intrusive and result in more flow for both cars and pedestrians in a crosswalk resulting in a more safe space.

This problem is connected to assessing awareness and common design patterns. This system should be assessing awareness of pedestrians and cars in some way because if not, it would simply behave like a traffic light. It would disrupt the flow a lot more if awareness wasn't looked at. In this case, the awareness detected doesn't have to be relatively complex. Awareness could mean the pedestrian stopping, walking, running and cars normal speed, slowing down, stopped, etc. These are valid state of awareness of the people. Let's say a sensor senses a pedestrian is going near a crosswalk and puts a light on for the cars indicating a pedestrian is near. The system can determine the awareness of a car if it starts slowing down when that light is on. If it determines that the car is slowing down, then it is probably yielding to the pedestrian so it can give a green light to the pedestrian. It is crucial that automated systems such as this has this awareness because there can be certain conditions in which the system doesn't know what to do. In this specific case, the amount of possibilities are a lot less, but it never hurts to have this awareness as it further helps increase the safety of the pedestrians and cars. Now, at the topic of common design patterns. There's a reason why all traffic lights and crosswalks look the same and have the same functionalities. It is important that architectural patterns, both software and physical, are present in our lives. Imagine if each time you cross a street, the system to cross the street is different each time. You would have to remember and learn new things just to cross the street each time. Having reused traffic light architecture means, once a user experiences it, they will know what to do for each time they encounter it in the future. So "common" is not only appealing to software engineers, but also to everyone. These "common" patterns in our everyday lives is what makes it easy to navigate and understand the world and its surroundings.