Design-patterns-matrix

Indicators in analysis: There are different commonalities whose implementations are coordinated with each other. M Indicators in design: A class needs to instantiate a derivation of another class, but doesn’t know which one. Factory method e allows a derived class to make this decision. t Field notes: The Factory method is often used with frameworks. It is also used when the different implementations of one class h hierarchy requires a specific implementation of another class hierarchy. Note that a factory method pattern is not simply a method that serves as a factory.

The pattern specifically involves the case where the factory is varied polymorphically. Factory Method is o also very useful when unit testing with Mock Objects. d Principle manifested: Encapsulate the relationship between class hierarchies. Indicators in analysis: There are rules about how to manage a set of objects. These can relate to the number, the lifespan, handling error conditions, load balancing and more. P o Indicators in design: There are rules about creating and managing your objects. o Field notes: The object pool is a great way to defer the management of the lifespan of your objects.

Use some trivial to implement l rule, but encapsulate it in the object pool. Once you have the system working and you better understand the rules for object creation and management, implement these rules in the encapsulated methods. Indicators in analysis: There are prototypical instances of things. Indicators in design: When objects being instantiated need to look like a copy of a particular object. Allows for dynamically specifying what our instantiated objects look like. Indicators in analysis: There exists only one entity of something in the problem domain that is used by several different things.

Indicators in design: Several different client objects need to refer to the same thing and we want to make sure we don’t have more than one of them. You only want to have one of an object but there is no higher object controlling the instantiation of the object in questions. Field notes: You can get much the same function as Singletons with static methods, however using static methods eliminates the possibility of handing future change through polymorphism, and also prevents the object from being passed by reference, serialized, remoted, etc… in general, statics are to be avoided if possible.