

# Design Patterns

**November 20, 2014**

# Design Patterns

- **Goals**
- **Background**
- **Example**

# Design Patterns - Goals

## Today

- **provide a template for future tech talks about specific design patterns**
- **present an example**

# Design Patterns - Goals

## Series

- **Share instances of where specific design patterns are used**
- **Demonstrate implementation of design patterns in differing languages**

# Design Patterns - Background

**What are design patterns?**

*"The elements of this language are entities called patterns. Each pattern describes a problem that occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice."*

**Christopher Alexander, et al 'A Pattern Language' - 1977**

# Design Patterns - Back Ground

***'Design Patterns'* - 1995**

**Erich Gamma**

**Richard Helm**

**Ralph Johnson**

**John Vlissides**

**-The Gang of Four**

# Design Patterns - Essential Elements

- **Name**
- **Problem**
- **Solution**
- **Consequences**



# Design Patterns - Essential Elements

- **Name**

- A simple name makes it easier to add the pattern to your pattern language

- **Problem**

- A description of the problem and its context

# Design Patterns - Essential Elements

- **Solution**

- An abstract description of the elements of the design that solves the problem.

- **Consequences**

- The pros and cons of the solutions.
- Things to consider
- Alternatives
- Related patterns

# Related Concepts

- **Idioms**
- **Antipatterns**

# Design Patterns - Idioms

## Idioms

- Language specific expressions of programming constructs
- Examples:
  - while(true) infinite loops - (c, c++, java)
  - for(;;) infinite loops - (c, c++, java)
  - self executing functions (javascript)

## Design Patterns - Antipatterns

- If design patterns show how to go from a problem to a good solution, then antipatterns show how to go from a problem to a bad solution
- Examples:
  - Spaghetti code
  - Blob class
  - Vendor Lockin

# Design Pattern Example

## **Name:**

- Singleton

## **Problem:**

- Ensure that a class only has a single instance and provide global access

## **Solution:**

- Hidden constructor
- Instance operator that lets client access the single instance of the class

# Design Pattern Example

## **(Singleton Continued)**

### **Consequences:**

Benefits:

- Controlled access
- Reduced namespace compared to a global variable instance

Things to consider

- Thread safety

# Design Pattern Example

**(Singleton Continued)**

## **Implementation**



# Design Patterns Future

- **More talks about software:**
  - Specific Design Patterns
  - Idioms
  - Antipatterns
- **Talks about Project Management**
  - Specific Design Patterns
  - Idioms
  - Antipatterns