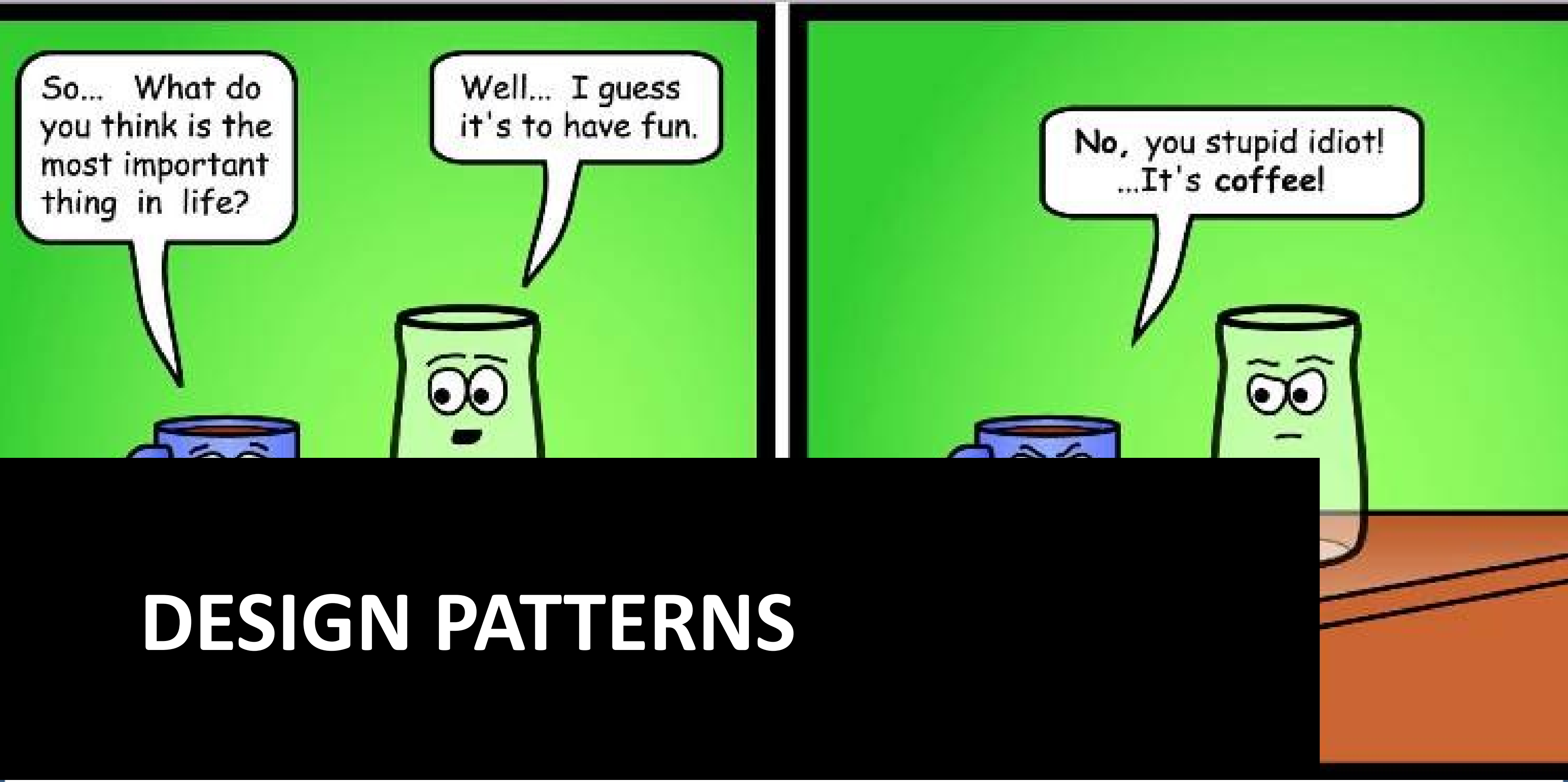

So... What do you think is the most important thing in life?
Well... I guess it's to have fun.
No, you stupid idiot! ...It's coffee!

DESIGN PATTERNS

http://www.flickr.com/photos/digitalcolony/2725932075/sizes/o/

<Patterns>

- You need to hang a poster on the wall…
  - What do you do?

- You need to write a term paper…
  - How should it be organized?

Design Patterns

- Design patterns (DPs) are…
  - Strategies for your “toolkit of ideas”
  - Templates for solutions
  - Codified best practices
  - Ideas for how to organize code

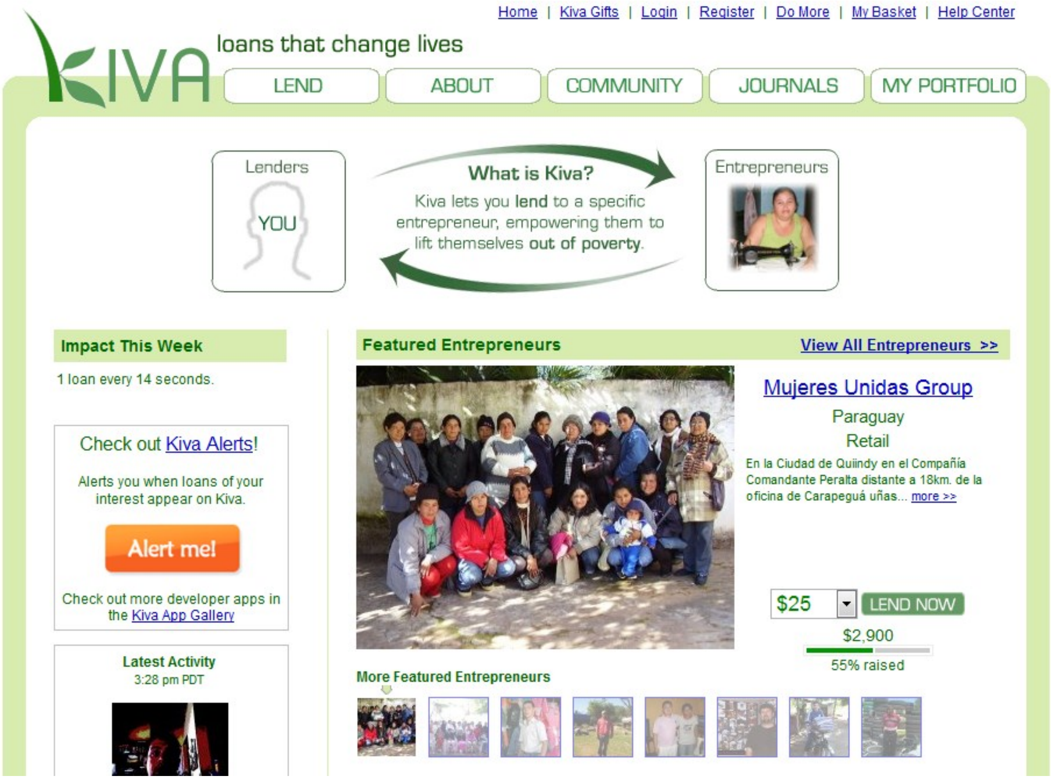
- Design patterns are not…
  - Architectural styles (DPs are too low level)
  - Code libraries (DPs are ideas, not code)

Design Patterns

- Primary goals of DPs
  - To help maintainability, flexibility, other quality attributes
  - To help system designers make good decisions
- There are a few dozen very common OO patterns
  - Patterns exist for other kinds of non-OO systems.
  - Patterns are recognizable based on their *structure and their purpose.*

Example system

- Kiva system to connect lenders with borrowers
  - How could we use DPs to implement Kiva?
  - How could we use DPs to implement a better Kiva???



loans that change lives

What is Kiva?

Kiva lets you lend to a specific entrepreneur, empowering them to lift themselves out of poverty.

Impact This Week

1 loan every 14 seconds.

Check out Kiva Alerts!

Alerts you when loans of your interest appear on Kiva.

Check out more developer apps in the Kiva App Gallery

Latest Activity
3:28 pm PDT

Featured Entrepreneurs

View All Entrepreneurs >>

Mujeres Unidas Group

Paraguay

Retail

En la Ciudad de Quiindy en el Compañía Comandante Peralta distante a 18km. de la oficina de Carapeguá uñas... more >>

$2,900

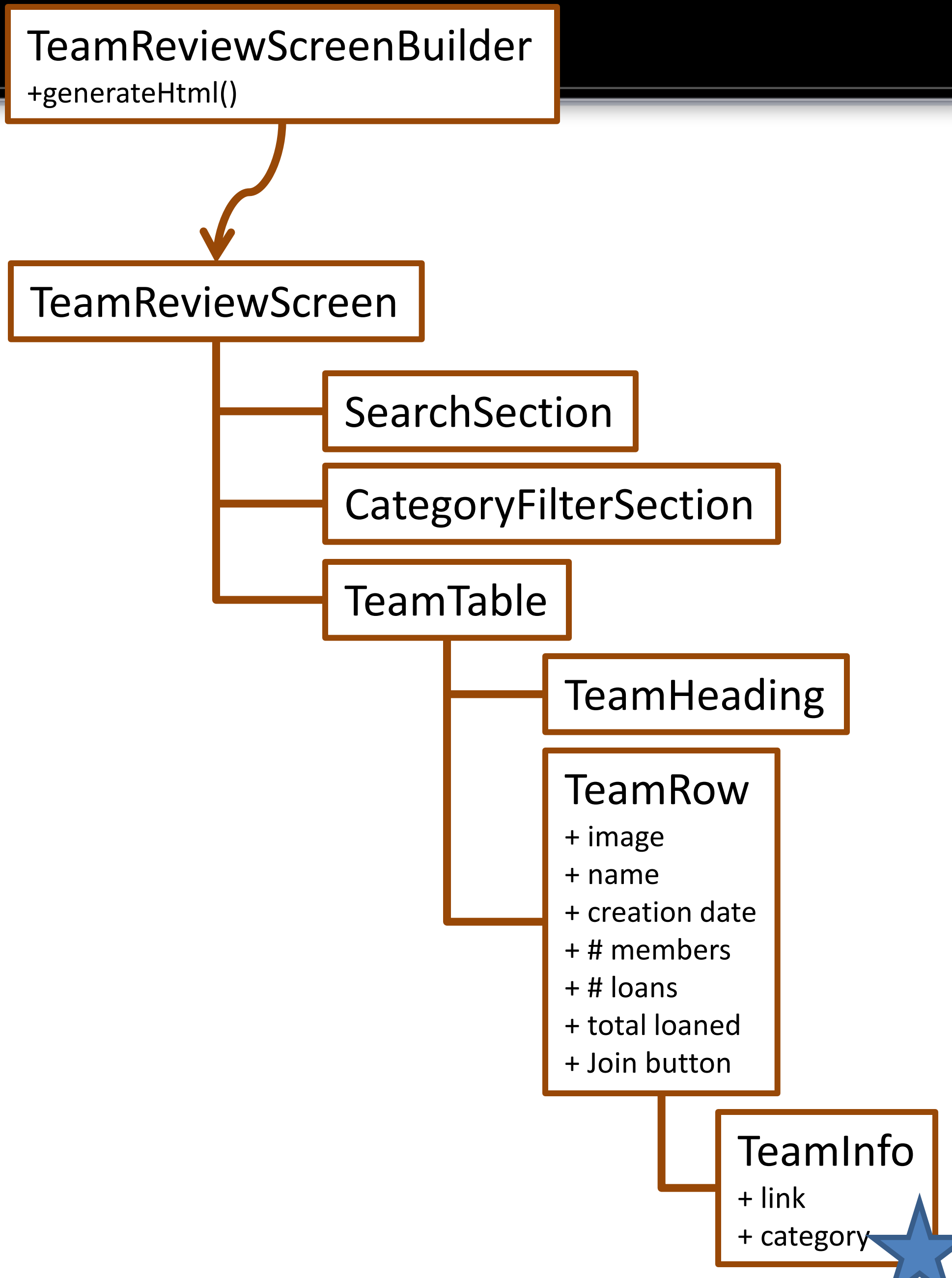
55% raised

More Featured Entrepreneurs

Builder

Knows how to create a complex object

Use when instantiating an object requires filling it with parts or otherwise lengthy configuration



TeamReviewScreenBuilder
+generateHtml()

TeamReviewScreen

SearchSection

CategoryFilterSection

TeamTable

TeamHeading

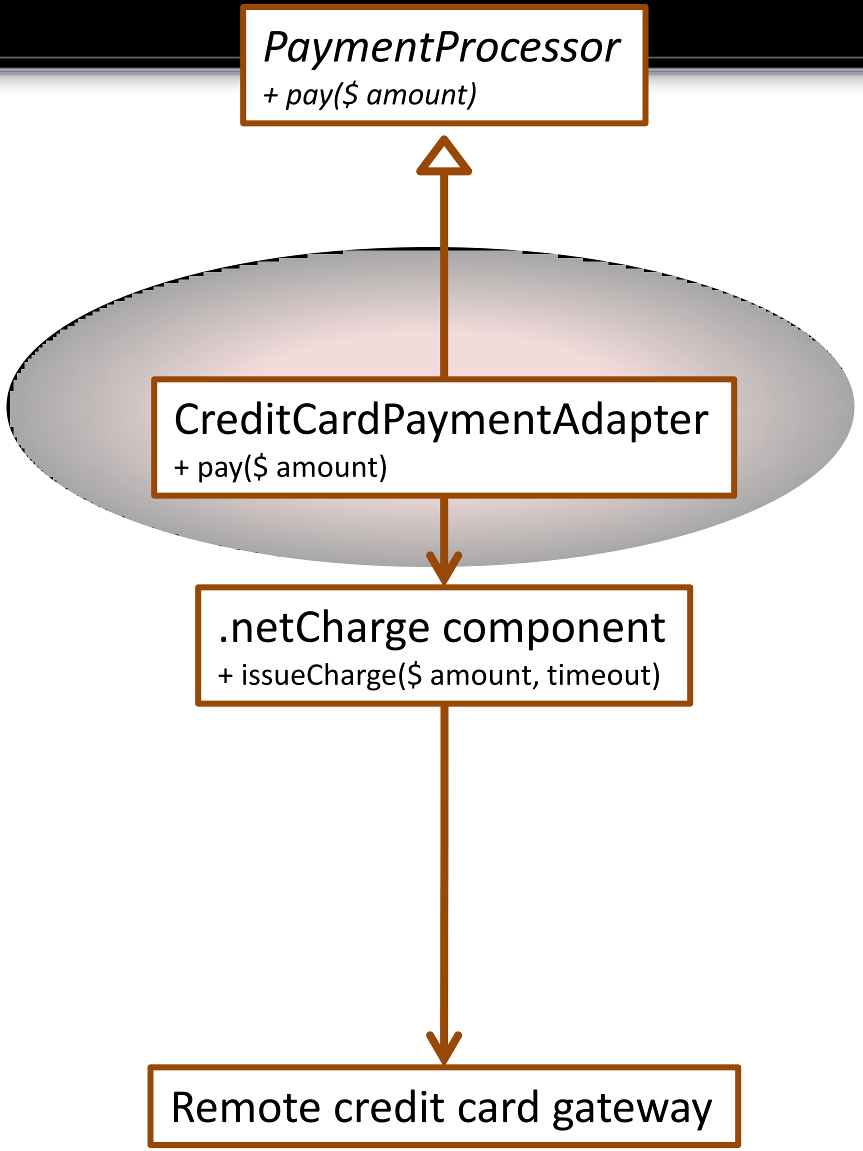
TeamRow
+ image
+ name
+ creation date
+ # members
+ # loans
+ total loaned
+ Join button

TeamInfo
+ link
+ category

Adapter

Converts one interface to another by wrapping

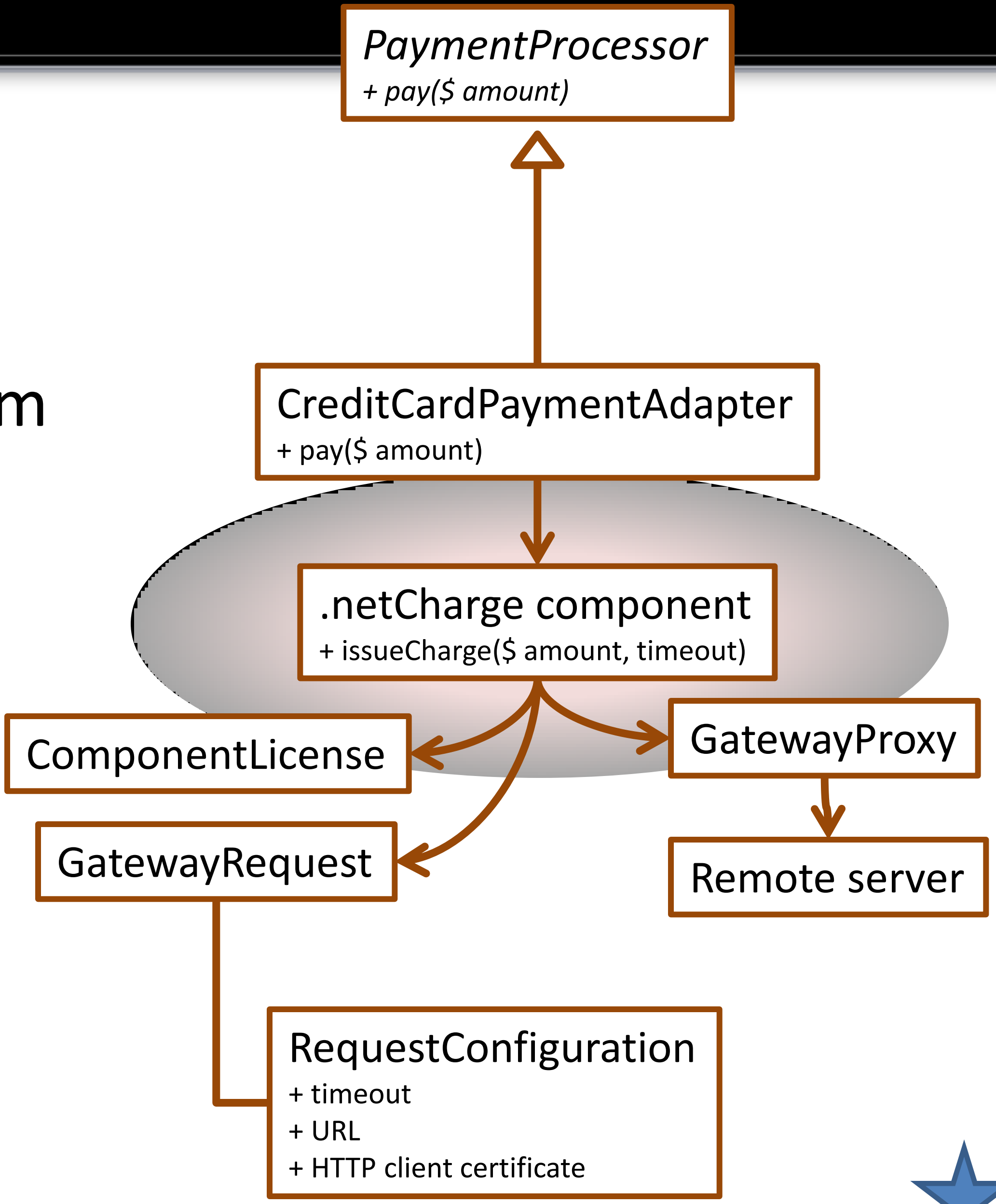
Use to overcome incompatibility



Facade

Object that provides a unified, high-level interface to a subsystem

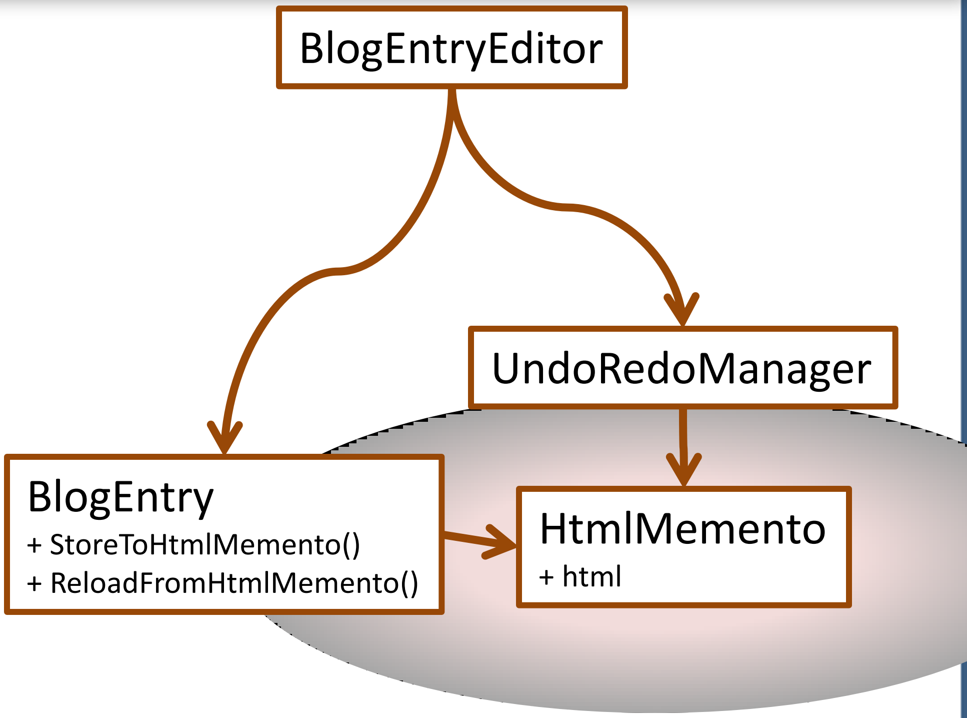
Use when calling a subsystem requires a frequent series of complex lines of code



Memento

Encapsulate state in an object

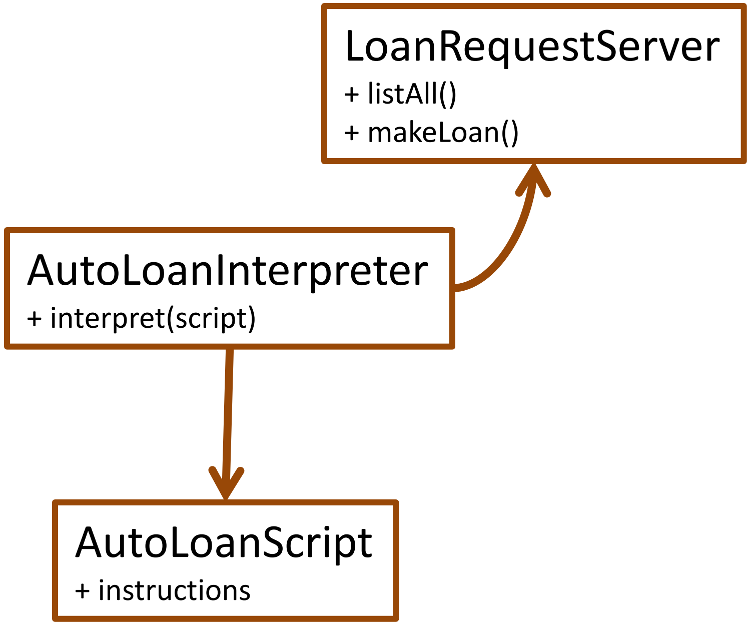
Use if you might want to return to a certain state later



Interpreter

Parses and acts on instructions written in a certain syntax

Use to add scriptability



LoanRequestServer
+ listAll()
+ makeLoan()

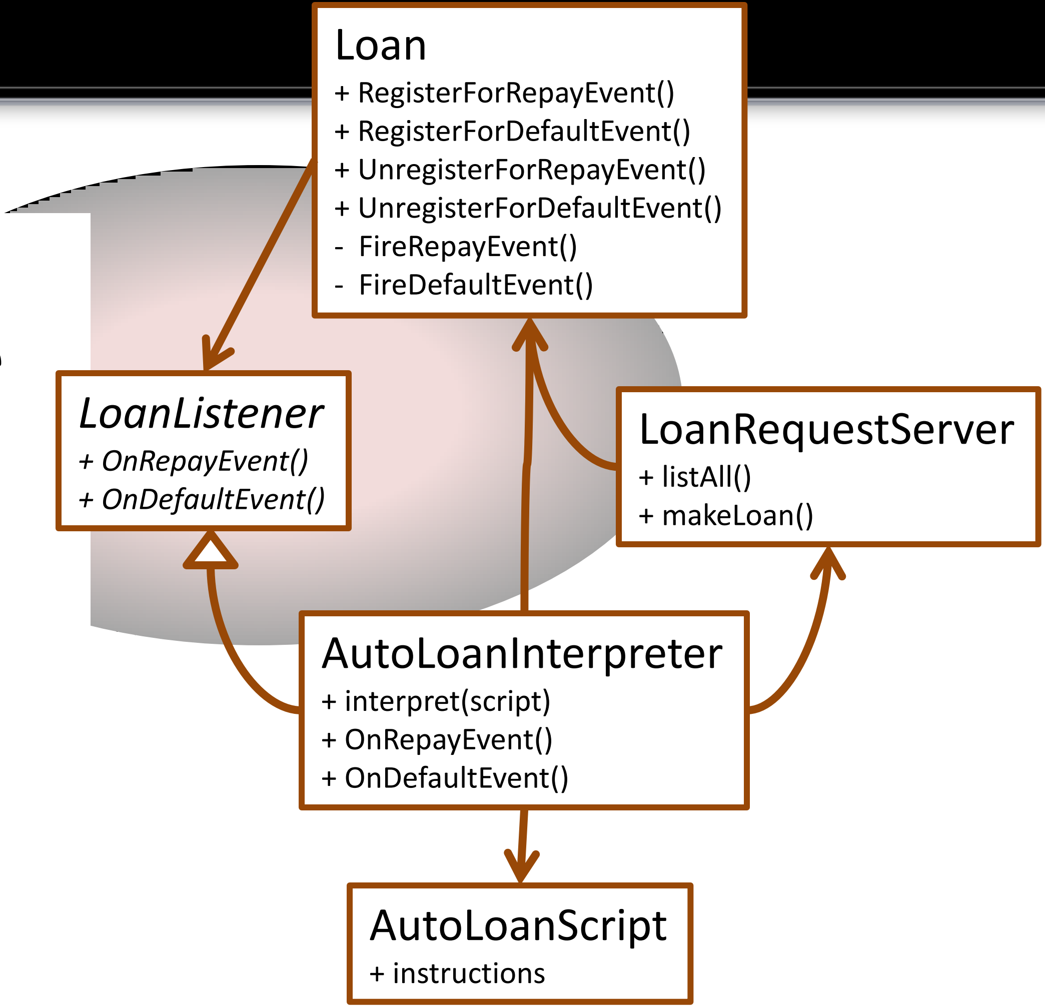
AutoLoanInterpreter
+ interpret(script)

AutoLoanScript
+ instructions

Observer

Watching for another object to change state

Use in any event-driven design



Which pattern

would you use?

- You are building a cool 3D game. Your company licenses a big, ugly library that implements the 3D mathematics.

Builder

Adapter

Façade

Memento

Interpreter

Observer

Which pattern

would you use?

- Your application might crash at any time. You want your application to save its state so that if it crashes, then it can auto-recover.

Builder	Memento
Adapter	Interpreter
Façade	Observer

Which pattern

would you use?

- Your application should run after any student uploads a homework to Blackboard.

Builder

Adapter

Façade

Memento

Interpreter

Observer

Which pattern

would you use?

- Your application needs to generate PDF files (from scratch).

Builder

Adapter

Façade

Memento

Interpreter

Observer

Which pattern

would you use?

- You want to let users create and run macros inside your application.

Builder	Memento
Adapter	Interpreter
Façade	Observer

Which pattern

would you use?

- Your company already implemented a component that almost implements the interface that you need, but not quite.

Builder	Memento
Adapter	Interpreter
Façade	Observer

Which pattern

would you use?

- You have a component that needs to implement three very slightly different interfaces.

Builder

Adapter

Façade

Memento

Interpreter

Observer

Which pattern

would you use?

- Your program has to create some big, ugly record objects before inserting them into a database.

Builder	Memento
Adapter	Interpreter
Façade	Observer

Now go read about these

(They are usually on the final exam!)

- **Use the textbook and/or Wikipedia**
  - Template method
  - Factory method
  - Strategy
  - Decorator
  - Composite
  - Visitor
- **Ask yourself: What is each of these patterns good for?**
  - If you can’t figure it out, then talk with your instructor!