JMock with GUI testing

VMFrame

«create»

«AdHoc Driver» VMFrameDriver

KeyPad

«Mock» VendingManager

«create»

Component Under test

CoinBox

«create»
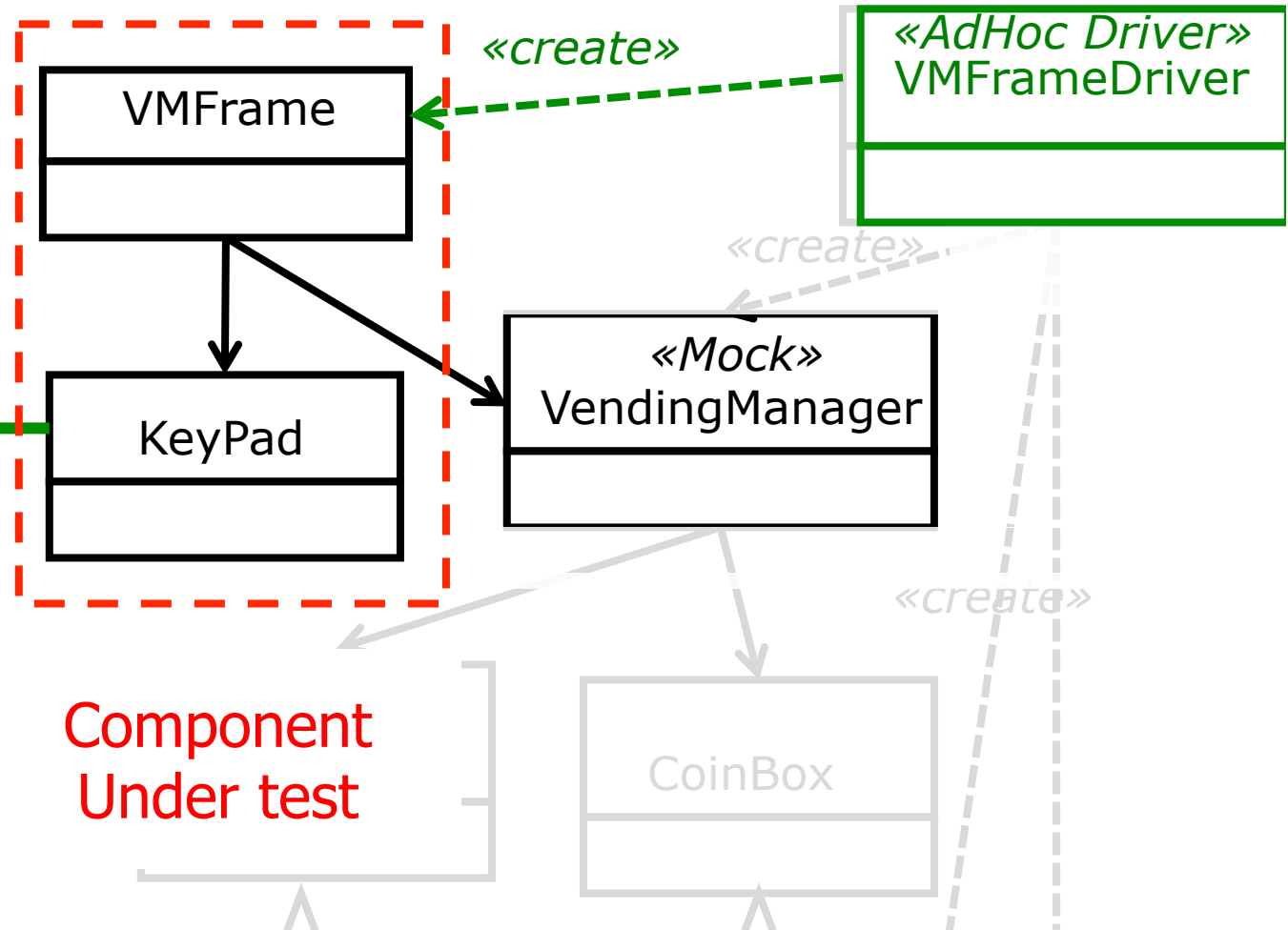
We need a mock for the VendingMachine
And a driver that instantiates the VMFrame.
We used the stereotype AdHoc Driver to highlight
That this driver is programmatically implemented
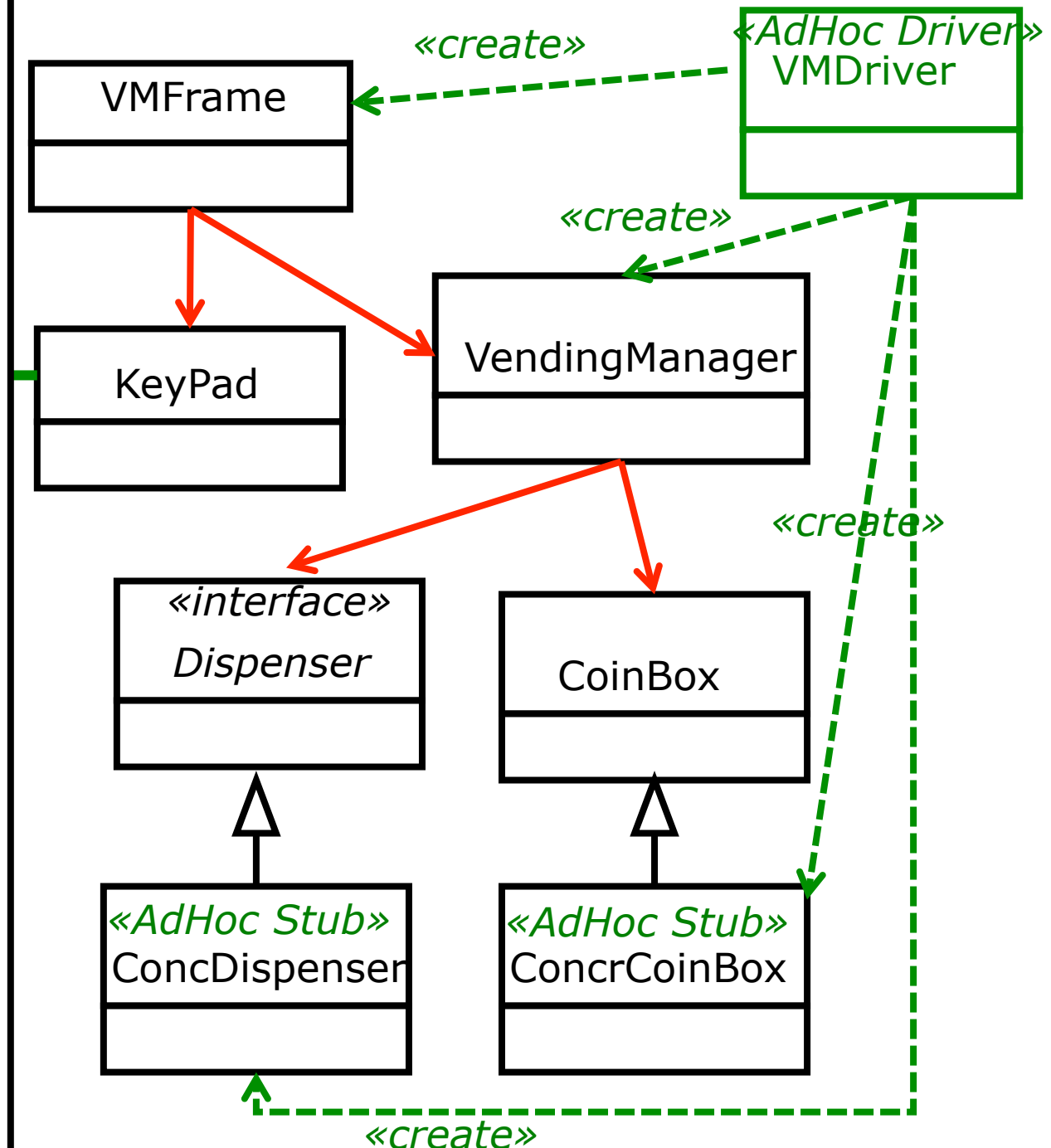as a java class, and is not provided by a framework
like JUnit

«create»

If we want to test the **integration** of the real components, while simulating the hardware, we may need **more scaffholding**.

In our example we decided to build:
*) an Ad-Hoc GUI stub that simulates the dispenser.
*) an Ad-Hoc GUI stub that simulates the coinbox.
*) a driver that starts up the components

Different choices might be taken. Simulating real-hardware with graphical stubs might be useful to perform alpha testing (or exploratory testing) without the real hardware.

VMFrame

«AdHoc Driver»
VMDriver

«create»

«create»

KeyPad

VendingManager

«create»

«interface»
Dispenser

CoinBox

«AdHoc Stub»
ConcDispenser

«AdHoc Stub»
ConcrCoinBox

«create»

If we want to test the **integration** of the real components, while simulating the hardware, we may need **more scaffholding**.

In our example we decided to build:

*) an Ad-Hoc GUI stub that simulates the dispenser.

*) an Ad-Hoc GUI stub that simulates the coinbox.

*) a driver that starts up the components

Different choices might be taken. Simulating real-hardware with graphical stubs might be useful to perform alpha testing (or exploratory testing) without the real hardware.

VMFrame

«create»

«AdHoc Driver»
VMDriver

KeyPad

«create»

VendingManager

«create»

«interface»
Dispenser

CoinBox

In our example this integration test allows to detect that the VendingManager never sets the prices displayed in the dispenser.

«AdHoc Stub»
ConcDispenser

«create»