Dr Strangebug or: How I Learned to Stop Worrying and Love the Exceptions

mobile application design

week 7: RSS, RMS. Lots of code touring.

michael sharon

Menu

- **Tips'n'tricks:**
  - **#1 Return of the JADi**
  - **#2 Know your SDKs**
  - **#3 Deployment shortcuts**
- **Exceptions, Threads**
- **Presentation:**
- **RSS example**
- **RMS**

Tips'n'tricks: Return of the JADi

Tips’n’tricks: Know your SDKs

Tips'n'tricks: Deployment

- Bluetooth
  - Drag + Drop on OS X
- OTA - upload to a server
- Email the URL
  - [phone_number]@carrier_email_host.com
  - Get list here: http://www.hiptools.net/sms/

Tips’n’tricks: Deployment

Exceptions

- What are exceptions?
  - “an event that occurs in a program that disrupts the normal flow of operations”
- Use try, catch, finally to handle
- 1 - Checked exceptions - IllegalArgumentException
- 2 - Errors - serious errors
- 3 - Runtime exception - NullPointerException

Why we love Exceptions

- Separate error handling code
- Grouping and differentiating error types
- Aids in debugging

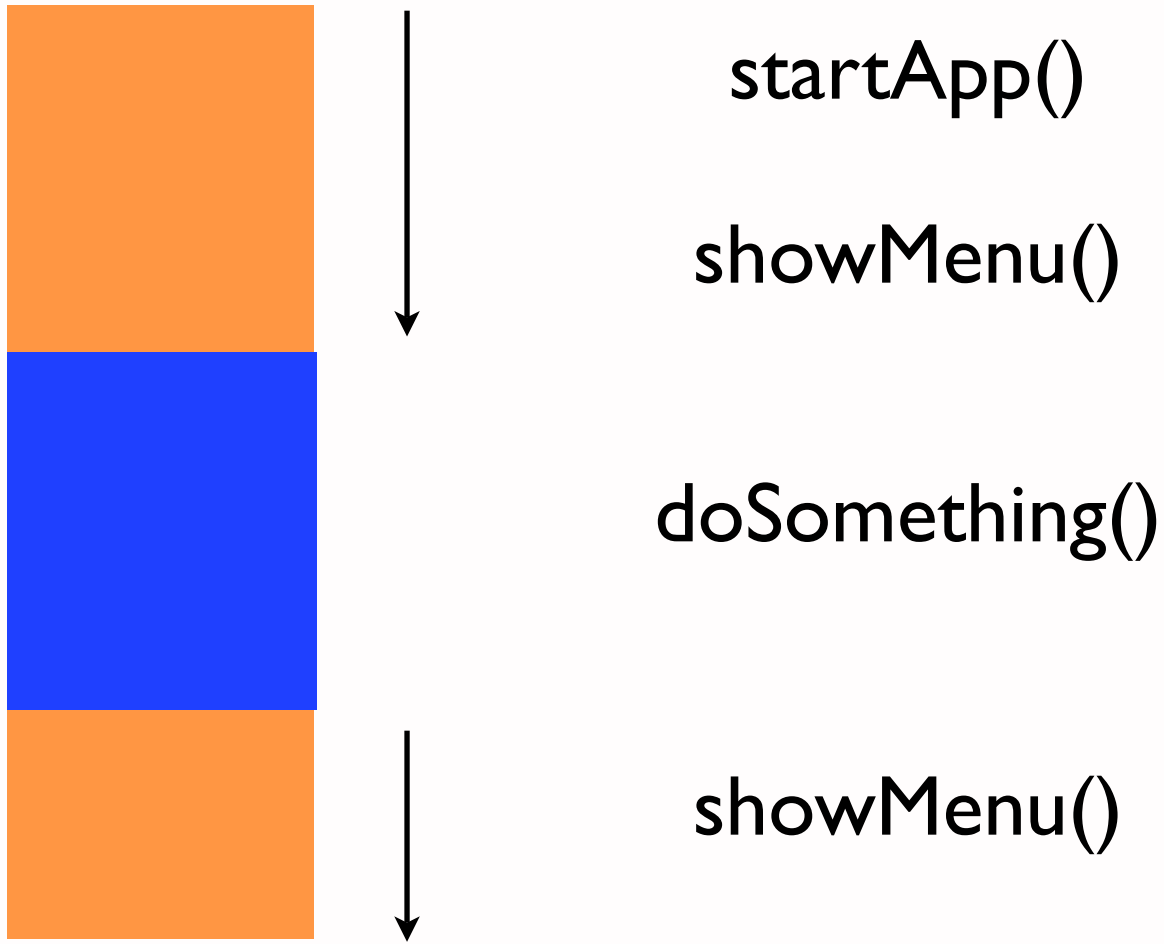
Threads

- What are threads?
  - "Single sequential flow of control"
- AKA sub processes which will do things for you
- Application = System thread + other threads
- J2ME is **multithreaded**

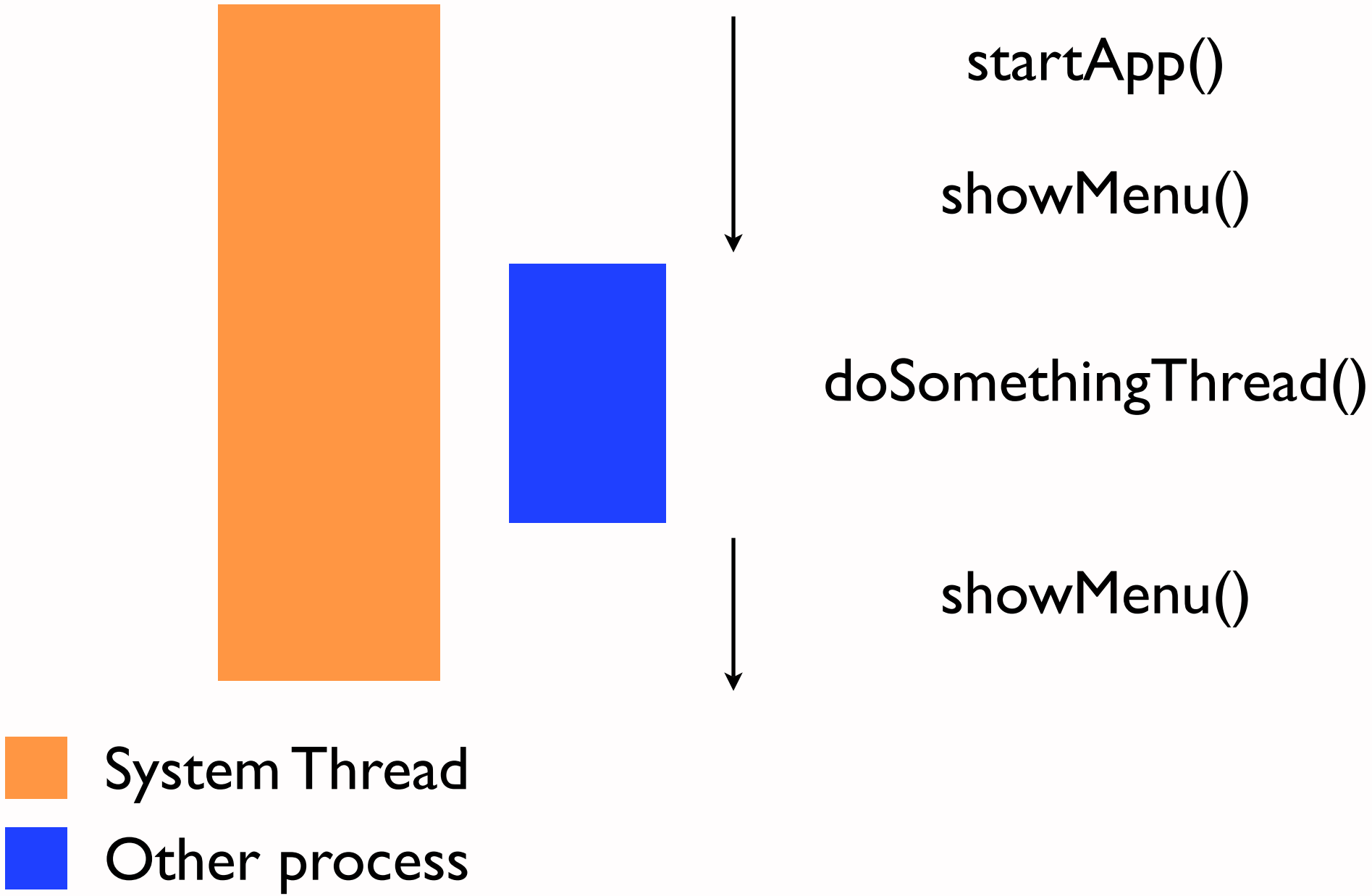
Threads (2)

- Every thread has a context
  - **Running** thread is executing code
  - **Ready** thread is ready to execute code
  - **Suspended** thread is waiting on an external event.
  - **Terminated** thread has finished executing code.
- Threads are scheduled (usually by OS)

Unthreaded MIDlet



Threaded MIDlet



Threads (3)

- To use threads you need:

1. an instance of java.lang.Thread
2. an object that implements java.lang.Runnable

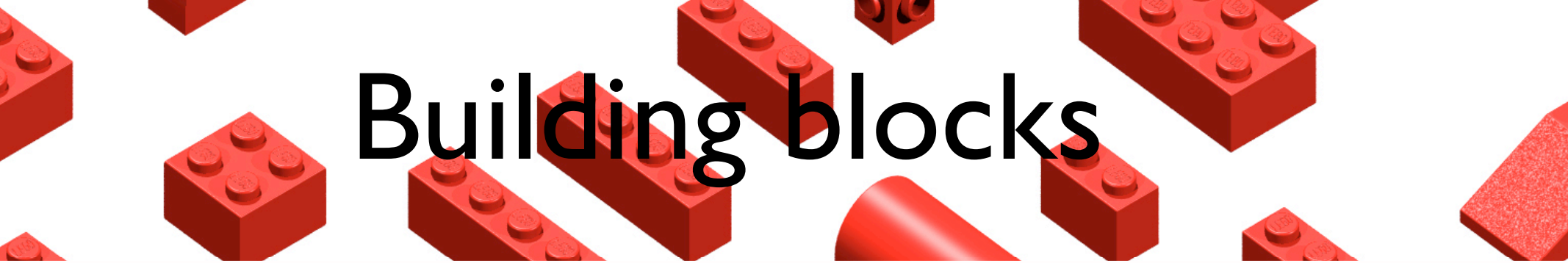
Using Threads

```
public class DoSomething implements Runnable {
    public void run(){
        // here is where you do something
    }
}
```

```
DoSomething doIt = new DoSomething();
Thread myThread = new Thread( doIt );
myThread.start();
```

Presentation: Gilad, Chunxi + Greg

Building blocks



Building blocks

Week 1 - HelloMIDlet (Form + Canvas skeleton)
Week 2 - ZombieDetector (Form, User Input)
Forms-A-Go-Go (UI components)
Week 3 - CanvasDemo (Threads, Canvases, Separate Files
NameThatKeyMidlet (Key Events)
TastingCanvas (Canvas drawing examples)
VideoMidlet (Camera Canvas MMAPI)
Week 4 - ImageMonkey (HTTP Image downloads)
MediaInformationMidlet (MIME-Type support)
Week 5 - XMLDemon (HTTP + XML parsing)
Jesse Boyes Game (GameCanvas + Threads)
Week 6 - PosDemo - (GPS Positioning)

MMAPI Architecture

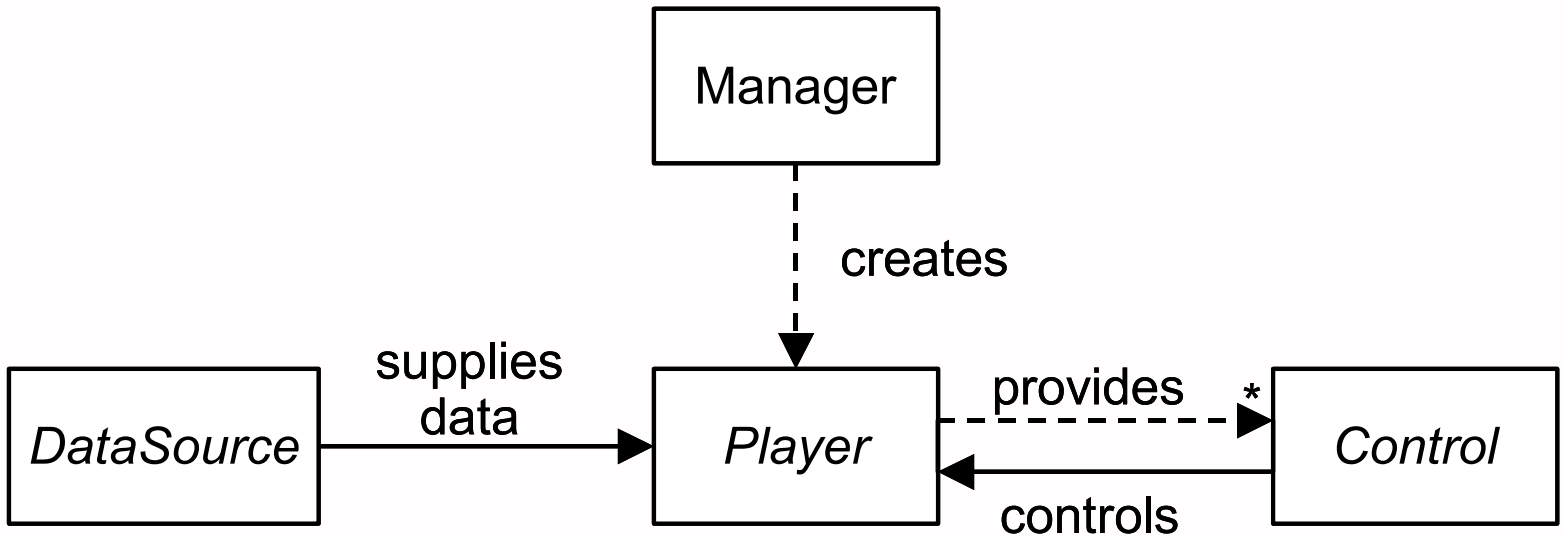


Figure 1: MMAPI architecture

Player State Model