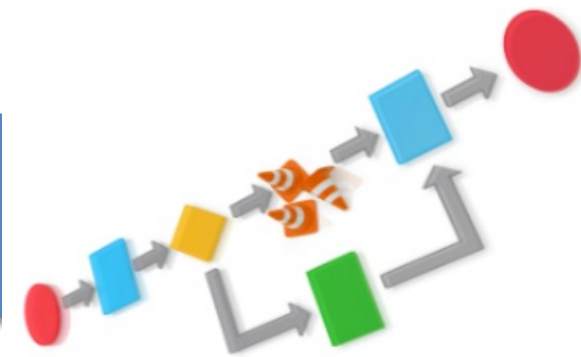


Java RMI

Jiří Šebek

b4m36swa



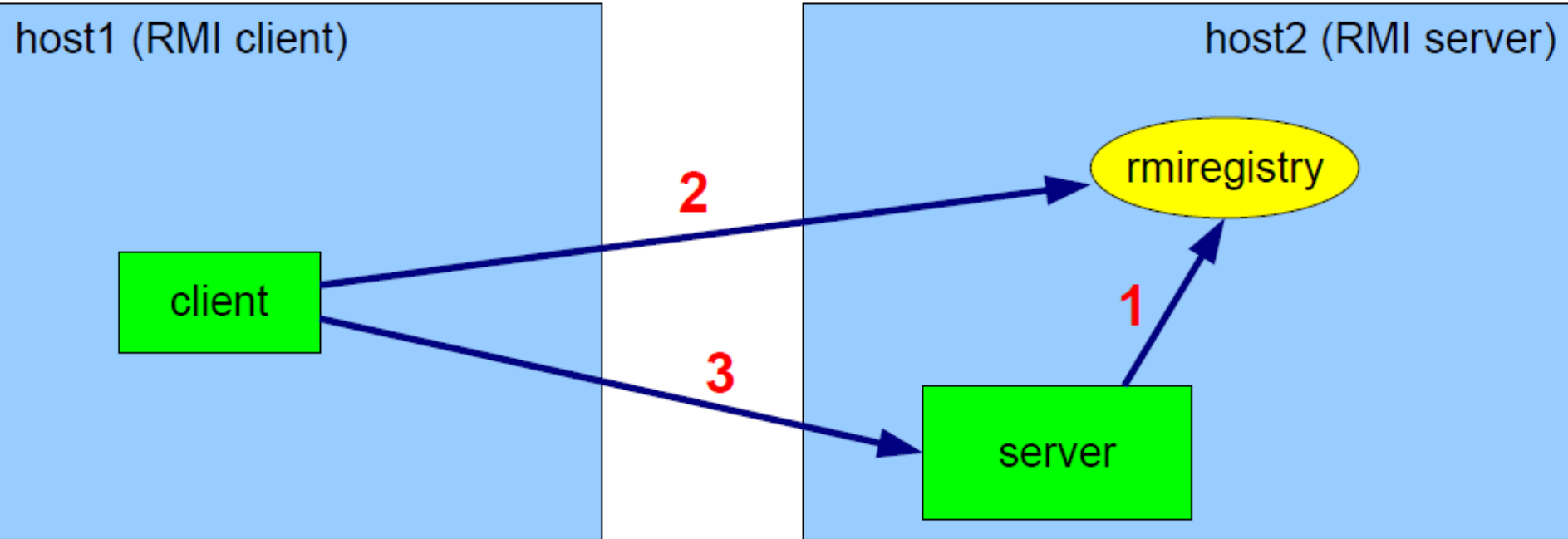
```
public final void onSensorChanged(SensorEvent event)
{
    m_flightIntensity = event.values[0];
    m_etAmblight.setText("" + m_flightIntensity + " lx");
}

... resume()
... light, ... NORMAL);
```

Popis

- Vyvolání metody z jiné JVM
 - princip vzdálených objektů
 - (deklarace + komunikace)
 - přenos objektu v bytecode
- Příklad ze cvičení
 - Klient – server architektura
 - Point to point (p2p)

Java RMI princip



- 1 – registrace objektu (svázání objektu se jménem) - `Registry.(re)bind`
- 2 – dotaz na objekt příslušící danému jménu- `Registry.lookup`
- 3 – práce se vzdáleným objektem - `Remote_object.remote_method`

Jednotlivé části

- vzdálený interface
 - deklarace metod vzdáleného objektu vyvolávaných klientem
- vzdálený objekt
 - implementace vzdáleného interface
- klient
- komunikace
 - stub, skeleton

Remote části

- remote methods
 - metody volané mezi JVM
- remote objects
 - objekty s remote metodami
- remote interface
 - určuje remote objekty, rozšiřuje interface *java.rmi.Remote*
 - všechny remote metody musí deklarovat *java.rmi.RemoteException*
- remote stub
 - nahrazuje remote objekt lokální reprezentací (proxy)
 - stejná množina remote interface, marshaling, unmarshaling

Vytvoření aplikace pomocí Java RMI

- remote interface
- implementace remote objektů
- implementace klienta
- překlad zdrojových kódů
- vytvoření stubu
- distribuce aplikace
- start aplikace

Implementace

- viz Math Implementace na cvičení