

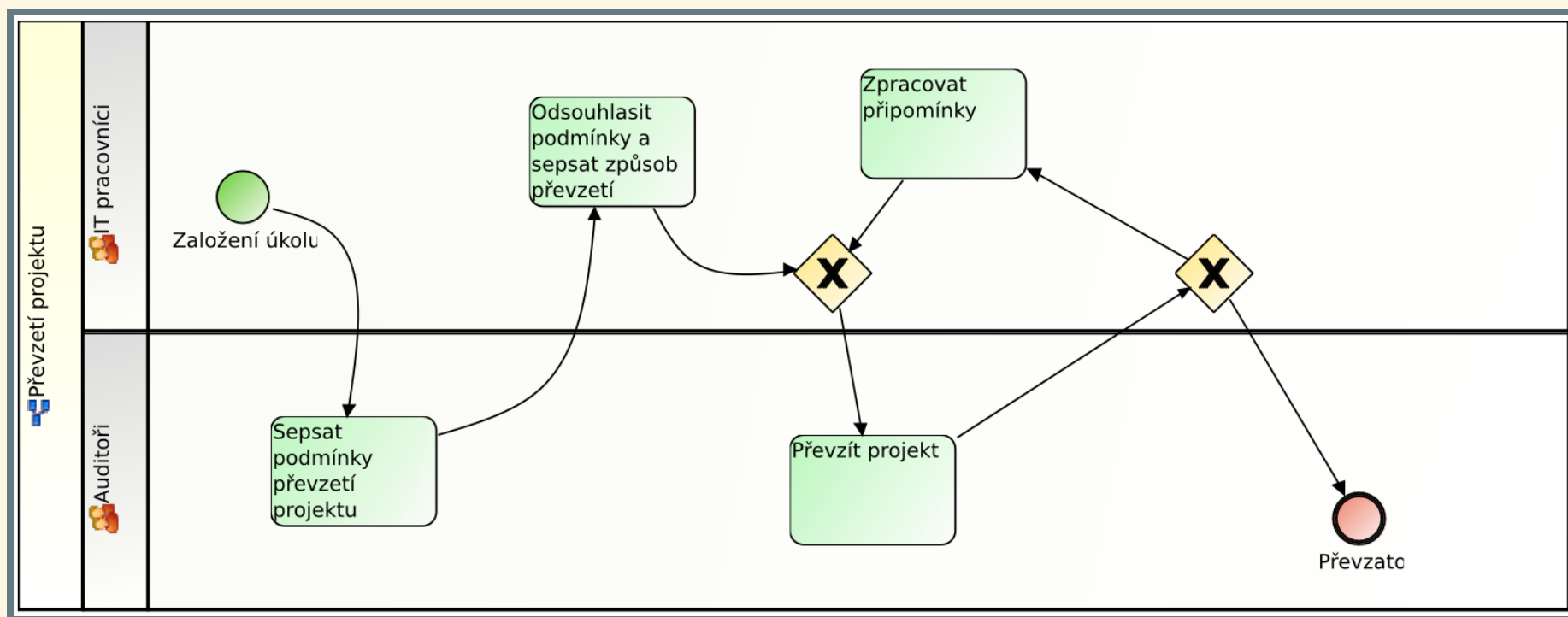
APACHE CAMEL INTRO

JOSEF LUDVÍČEK, 26. 2. 2015

BP AUTOMATICKÉ ZAKLÁDÁNÍ PRACOVNÍCH POSTUPŮ Z E-MAILŮ

- dostupné z [IS MU](#)
- cíl: zjednodušit zakládání procesů v systému MAP
- `java -jar router-1.0-SNAPSHOT.jar`
- monitorování pomocí webové konzole [Hawt.io](#)

"Proces je po částech uspořádaná množina činností, jež na základě jednoho nebo více vstupů tvoří opakovatelným způsobem požadovaný výstup."



Bonita BPM

The screenshot displays the Bonita BPM console interface. At the top, the browser address bar shows the URL: localhost:8080/bonita/console/homepage?tenant=1#?_p=tasklistinguser&_pf=1&_id=8&_f=available. The Bonitasoft logo is on the left, and the user 'William Jobs' is logged in. The main navigation bar includes 'Tasks', 'Cases', and 'Apps'. A sidebar on the left lists navigation options: 'To do', 'My tasks', 'Available tasks', 'Hidden', 'Done', and 'Apps' (with sub-items 'Help Desk' and 'Travel Request').

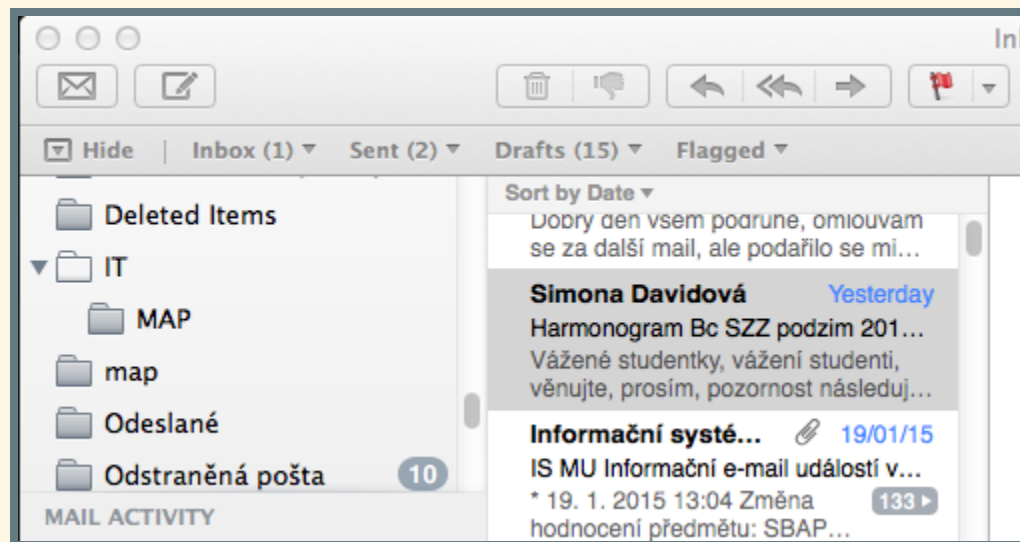
The 'Tasks' section shows a list of tasks with columns for 'Name' and 'Due date'. The selected task is 'Validate Travel Request' (Travel Request), which is due in 51 minutes. Action buttons 'TAKE', 'RELEASE', and 'HIDE' are visible above the list. A 'DO IT' button is highlighted in green at the top right of the task details panel.

The task details panel for 'Validate Travel Request' includes:

- Description: No description.
- App: Travel Request
- Due date: in 51 min
- Subtasks: No data, with an 'ADD SUBTASK' button.
- Done subtasks: No data.
- Technical details: Ready since May 22, 2013.
- Comments: A text input field for 'Type new comment' and an 'ADD COMMENT' button.

POUŽITÍ CAMEL APLIKACE

- Naskenováním dokumentu dojde k zahájení procesu
 - z kancelářských tiskáren lze odesílat skeny e-mailem
- Operátor přesunutím e-mailu zahájí proces



APACHE CAMEL

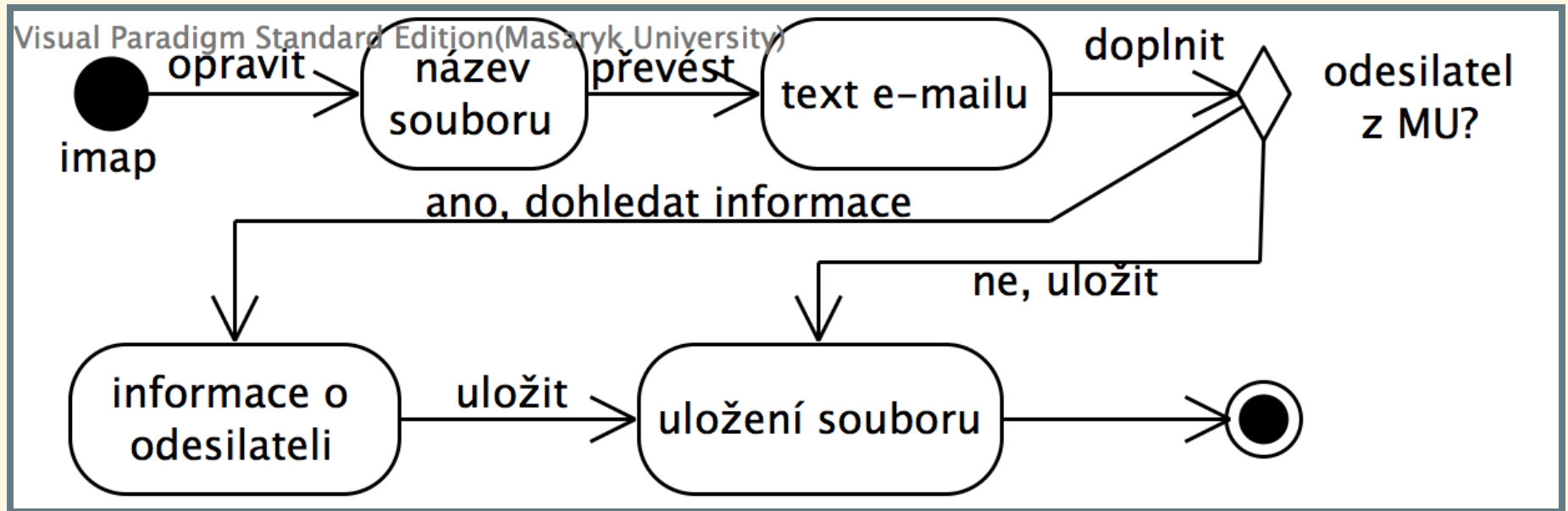
- oficiální web camel.apache.org
- kniha Camel in Action, (Manning Publications)
- kapitola 3 zmíněné bakalářské práce

OSNOVA

- koncept cesty
- Camel DSL
- transformace přenášených dat
- Camel & Spring
- Camel komponenty
- chybové řízení
- produkční nasazení Camel aplikace
- monitorování

KONCEPT CESTY

- posloupnost operací nad přenášenými daty



ZÁPIS CESTY

Java DSL

```
from("imaps://imap.gmail.com?" +
    "username=LOGIN&password=PASSWORD" +
    "delete=false&unseen=true")

.id("gmail-to-file")

.setHeader("CamelFileName")
    .groovy("resource:classpath:script.groovy")
.convertBodyTo(String.class)
.choice()
    .when(header("sender").contains("@mail.muni.cz"))
        .to("language://python:resource:classpath:is_muni.py")
        .to("file:files")
    .otherwise()
        .to("file:files");
```

Spring DSL

```
<camel:route id="gmail-to-file">
  <camel:from uri="imaps://imap.gmail.com?
    username=LOGIN&password=PASSWORD&
    delete=false&unseen=true"/>

  <camel:setHeader headerName="CamelFileName">
    <camel:groovy>
      resource:classpath:script.groovy
    </camel:groovy>
  </camel:setHeader>
  <camel:convertBodyTo type="java.lang.String"/>
  <camel:choice>
    <camel:when>
      <camel:simple>
        ${header.sender} contains '@mail.muni.cz'
      </camel:simple>
      <camel:to uri="language://
        python:resource:classpath:is_muni.py"/>
      <camel:to uri="file:files"/>
    </camel:when>
    <camel:otherwise>
      <camel:to uri="file:files"/>
    </camel:otherwise>
  </camel:choice>
</camel:route>
```

CAMEL A SPRING FRAMEWORK

```
<bean id="amq"  
  class="org.apache.activemq.camel.component.ActiveMQComponent">  
  <property name="brokerURL" value="tcp://localhost:61616"/>  
  <property name="userName" value="jack"/>  
  <property name="password" value="sparrow"/>  
</bean>  
  
<!-- send message to ActiveMQ in Camel Route -->  
<to uri="amq:queue:TEST.FOO"/>
```

ActiveMQ Hello World

Zasílání zpráv z obyčejné třídy (Bean Binding).

```
import org.apache.camel.EndpointInject;
import org.apache.camel.ProducerTemplate;

public class SomeClass {

    @EndpointInject(uri="amq:queue:TEST.FOO")
    ProducerTemplate template;

    public void sendMessage(String message) {
        template.sendBody(message);
    }
}
```

Volání Java třídy z camel cesty

```
from("direct:something")
    .bean(SomeClass.class);
```

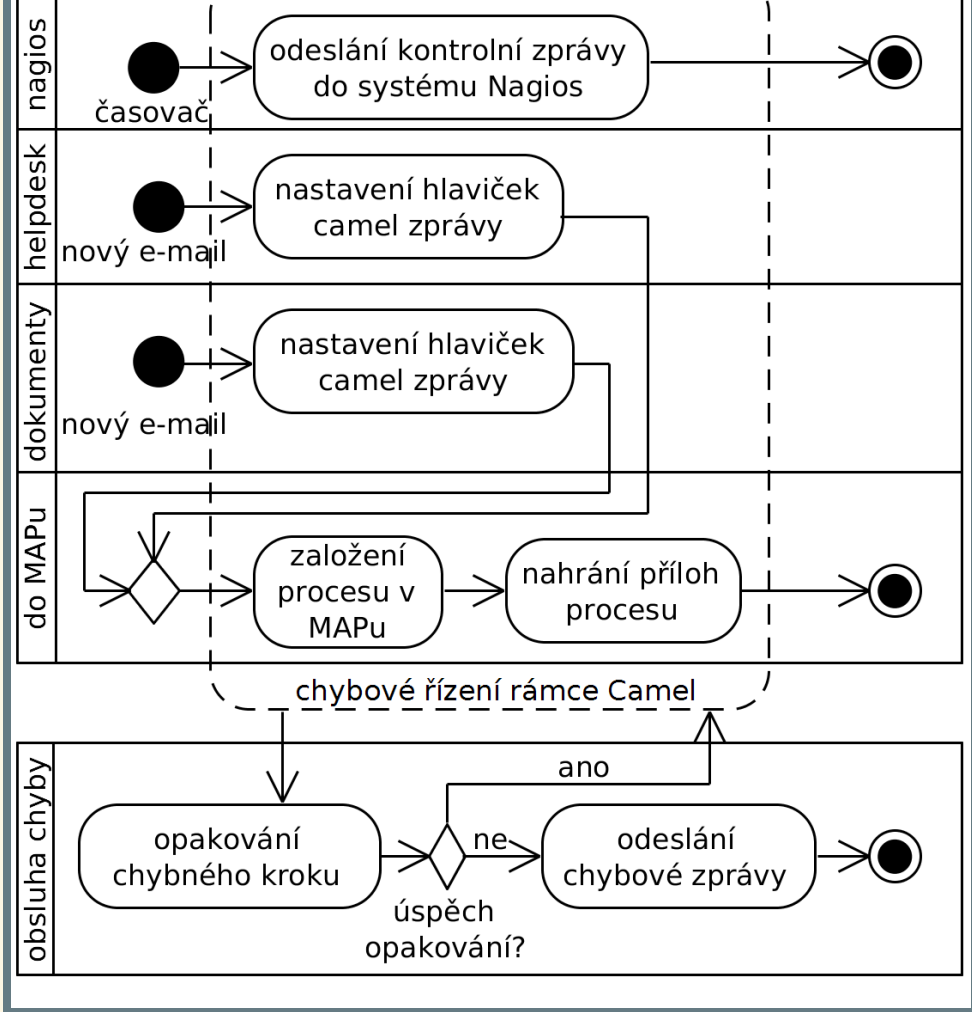
CAMEL KOMPONENTY

camel.apache.org/components

CHYBOVÉ ŘÍZENÍ

```
<camel:errorHandler id="deadEH" type="DeadLetterChannel"
    deadLetterUri="file:problematic">
  <camel:redeliveryPolicy maximumRedeliveries="10"
    logStackTrace="true" logHandled="true"
    delayPattern="1:1000;5:5000;10:20000"
    retryAttemptedLogLevel="WARN"
    retriesExhaustedLogLevel="ERROR"/>
</camel:errorHandler>

<onException>
  <exception>java.io.IOException</exception>
  <handled><constant>true</constant></handled>
</onException>
```

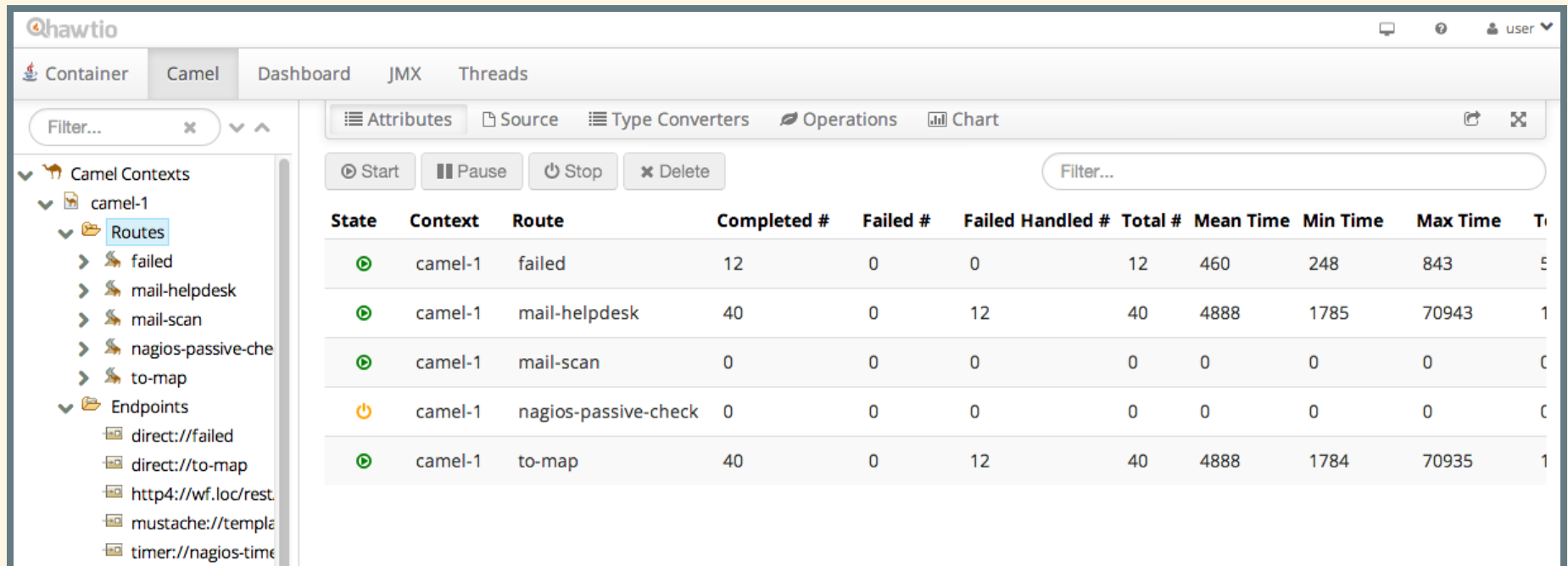


PRODUKČNÍ NAsAZENÍ

- spustitelná aplikace
 - java -jar app.jar
 - Spring Boot
- webový kontejner / aplikační server
 - JBoss Wildfly
 - Apache Tomcat
- OSGI kontejner
 - JBoss Fuse
 - Apache Karaf

MONITOROVÁNÍ

- JConsole
- Webová konzole [Hawt.io](https://hawtio.org/)



The screenshot displays the Hawtio web console interface for monitoring Camel routes. The interface includes a navigation menu on the left, a top navigation bar, and a main content area with a table of route statistics.

Navigation: Container, Camel, Dashboard, JMX, Threads

Left Panel (Camel Contexts):

- Camel Contexts
 - camel-1
 - Routes
 - failed
 - mail-helpdesk
 - mail-scan
 - nagios-passive-che
 - to-map
 - Endpoints
 - direct://failed
 - direct://to-map
 - http4://wf.loc/rest
 - mustache://temple
 - timer://nagios-time

Table of Route Statistics:

State	Context	Route	Completed #	Failed #	Failed Handled #	Total #	Mean Time	Min Time	Max Time	Ti
🟢	camel-1	failed	12	0	0	12	460	248	843	5
🟢	camel-1	mail-helpdesk	40	0	12	40	4888	1785	70943	1
🟢	camel-1	mail-scan	0	0	0	0	0	0	0	0
🟡	camel-1	nagios-passive-check	0	0	0	0	0	0	0	0
🟢	camel-1	to-map	40	0	12	40	4888	1784	70935	1

DEMO

?

Josef Ludvíček

is.muni.cz/auth/osoba/ludvicek

ludvicek@mail.muni.cz