

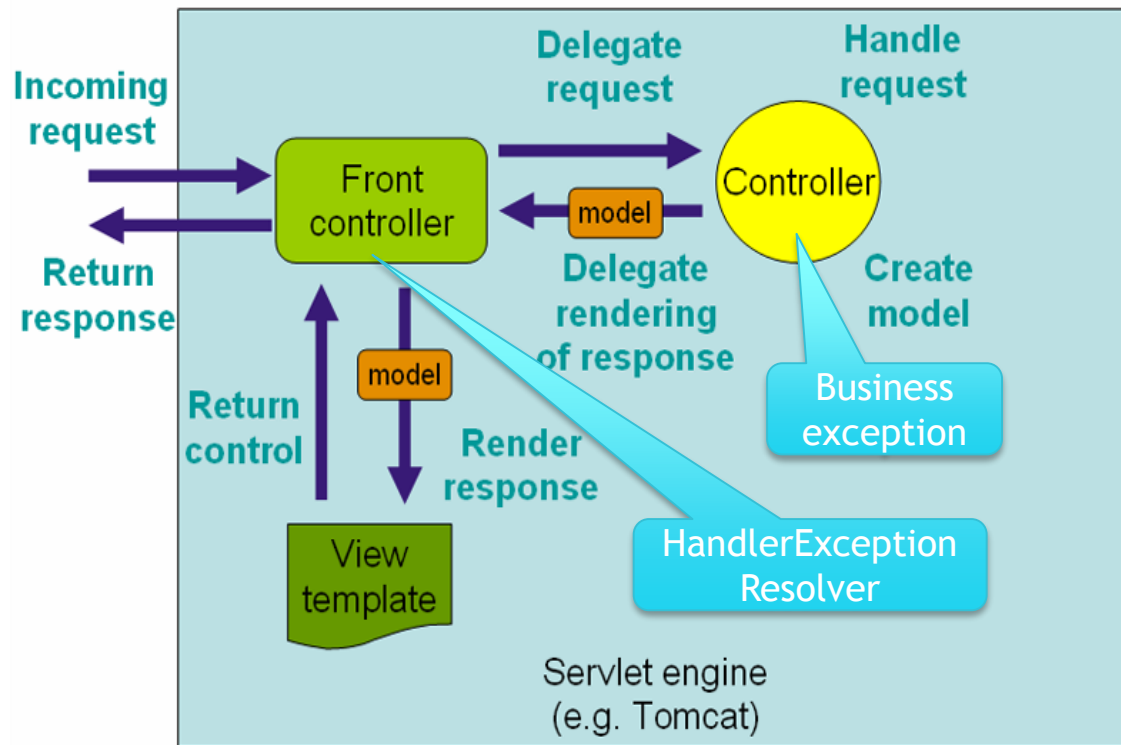


SPRING MVC

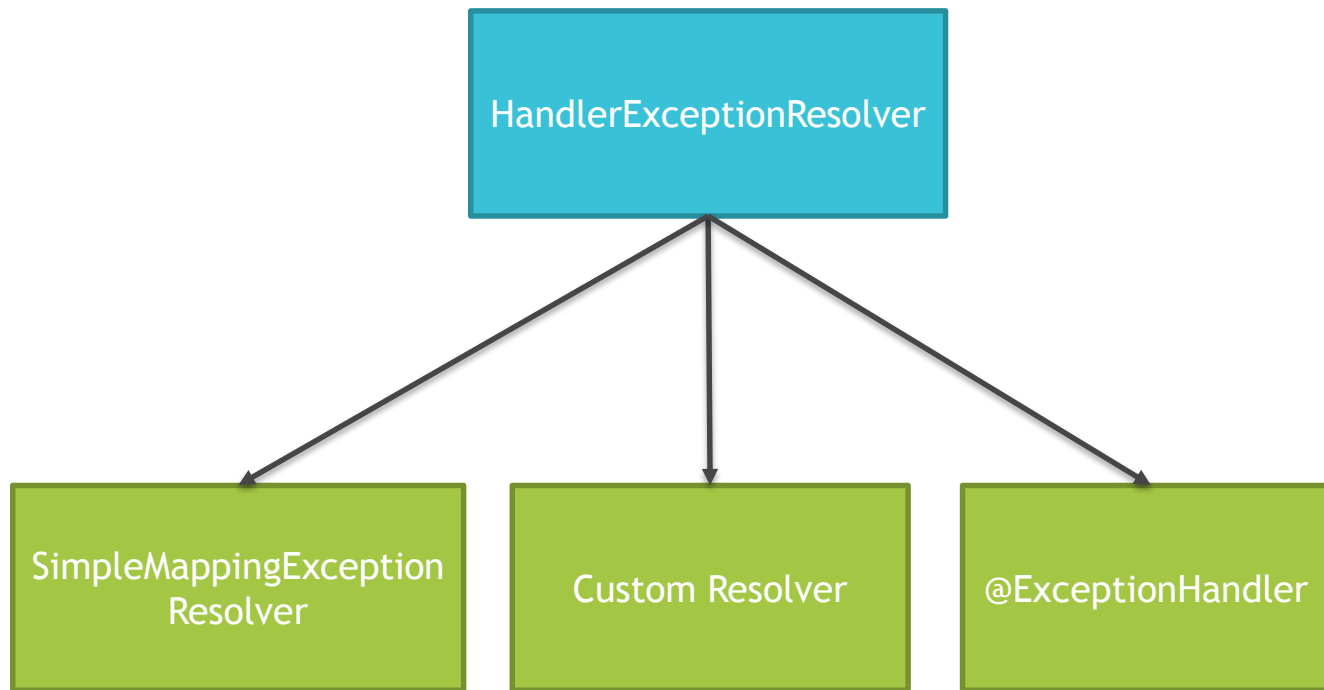
EXCEPTION HANDLING

MARCH, 2015

EXCEPTION HANDLING



EXCEPTION HANDLING APPROACHES



SimpleMappingExceptionHandler

```
<bean id="simpleMappingExceptionHandler"
      class="org.springframework.web.servlet.handler.SimpleMappingExceptionHandler">
  <property name="exceptionMappings">
    <map>
      <entry key="DatabaseException" value="databaseError"/>
    </map>
  </property>

  <property name="defaultErrorView" value="error"/>
  <property name="exceptionAttribute" value="ex"/>

  <!-- Name of logger to use to log exceptions. Unset by default, so logging disabled -->
  <property name="warnLogCategory" value="com.epam.example.MvcLogger"/>
</bean>
```

CUSTOM EXCEPTION RESOLVER

```
public interface HandlerExceptionResolver {  
    ModelAndView resolveException(  
        HttpServletRequest request,  
        HttpServletResponse response,  
        Object handler,  
        Exception ex);  
}
```

```
<?xml version="1.0" encoding="UTF-8"?>  
  
<beans>  
  
    <bean class="com.epam.example.CustomExceptionHandler">  
  
</beans>
```

ANNOTATION BASED CONFIGURATION

```
@Controller
public class TicketController {
    ...
    @ExceptionHandler(Exception.class)
    @ResponseStatus(HttpStatus.INTERNAL_SERVER_ERROR)
    public ModelAndView onServerException(Exception e) {

        LOGGER.error("Internal server error while processing request", e);
        return createResponse(HttpStatus.INTERNAL_SERVER_ERROR.toString(), e.getMessage());
    }
}
```

P.S. `@ExceptionHandler` can also be defined in `@ControllerAdvice`



THANK YOU!

MAKSYM_GOVORISCHEV@EPAM.COM

MARCH, 2015