



Doing More With PASOE and Spring



Chad R. Thomson

Senior Principal Consultant, Progress Services

May, 2019

Agenda

- Session Introduction
- PASOE Architecture Review
- Use Case and Demonstration
- What Else Can Spring Do?
- Q and A



Session Introduction

DISCLAIMER: Unsupported
techniques lay ahead.

Here be dragons!

Session Introduction

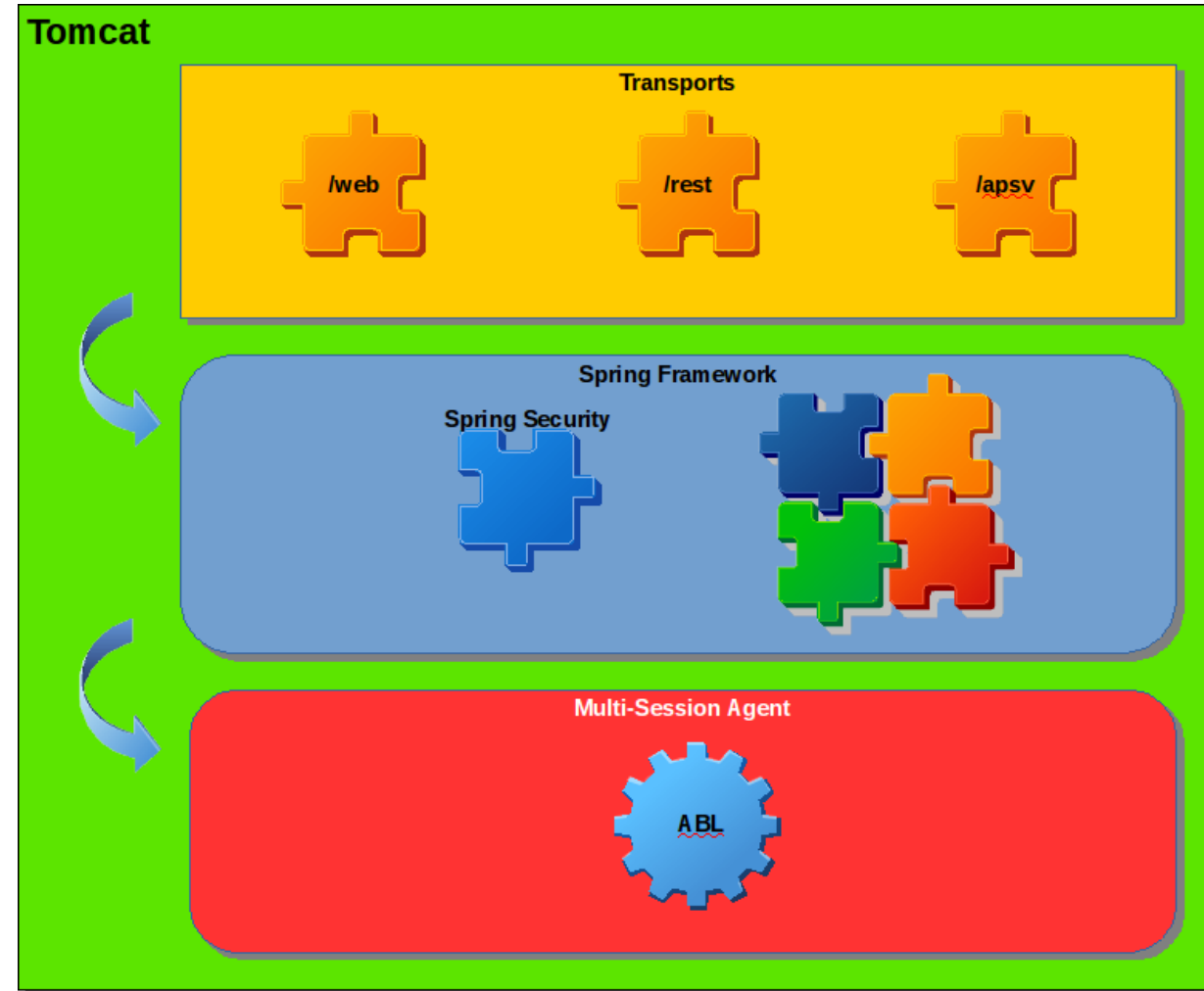
- The PASOE platform has yes-to-be-discovered possibilities
 - e.g. Spring Framework and its dependencies
 - All Spring Framework functions are accessible to PASOE webapps
- You say, “Wow! Can any of it be used for our own purposes?”
 - *Hint – YES!*
- Deep Technical Aspects
 - Java, Eclipse IDE, Docker
 - OpenEdge Java OpenClient
 - Will not go into details of installation, setup, or configuration



PASOE Architecture Review

PASOE Architecture Review

- Tomcat
 - <https://tomcat.apache.org/>
- Spring Framework
 - <https://spring.io>
 - Used for **security** components
 - Pay attention to library **version**
 - Two styles of Spring configuration
 - **<xml>** (PASOE)
 - **@Annotations** (Java)
- OpenEdge Multi-Session Agent
 - OpenEdge ABL

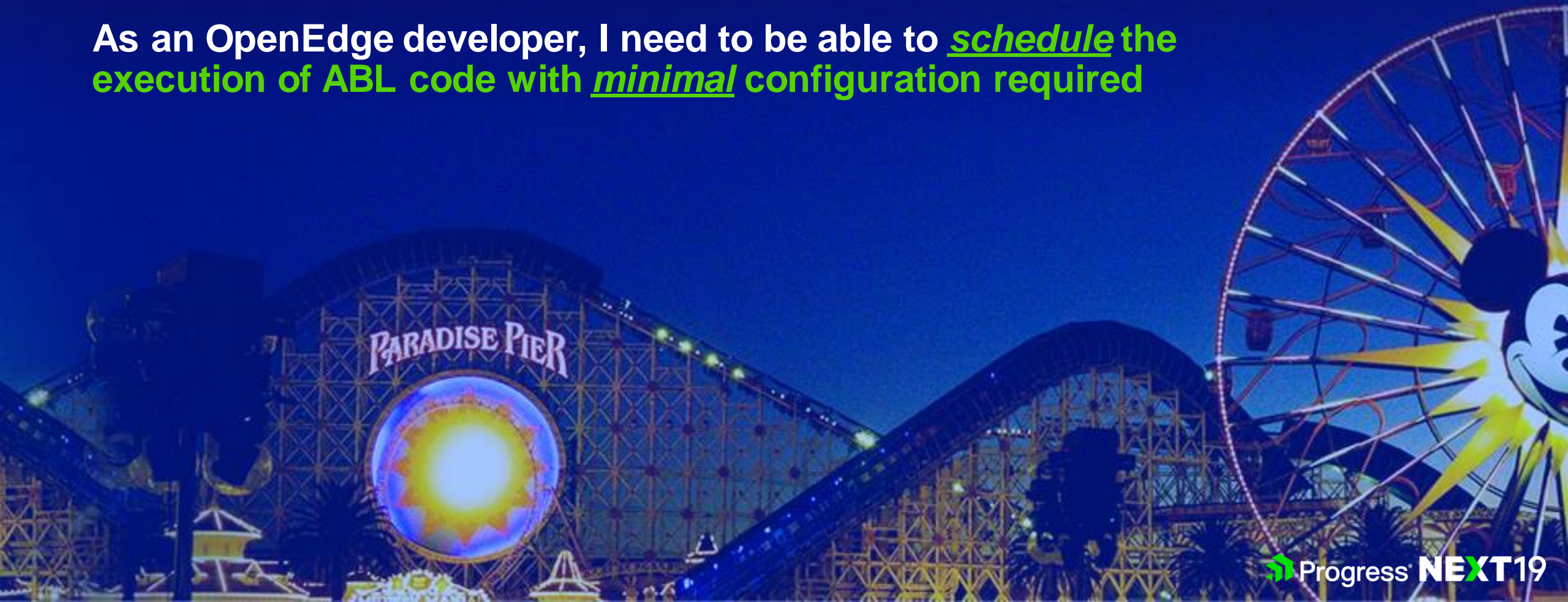




Use Case and Demonstration

Use Case Statement:

As an OpenEdge developer, I need to be able to schedule the execution of ABL code with *minimal* configuration required



Use Case: Spring TaskScheduler

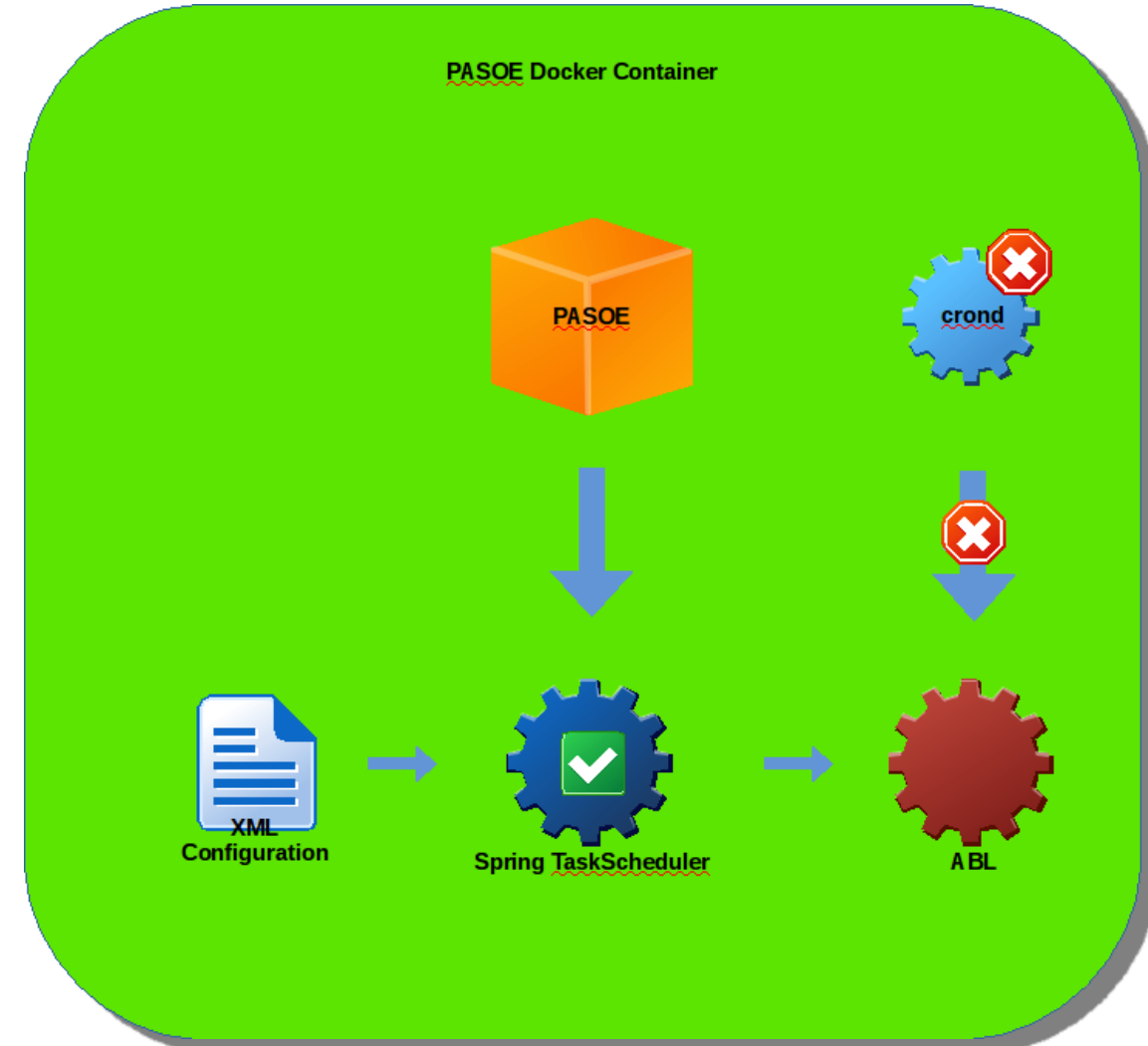
■ PASOE Container Use Case

- Docker is not *required*
 - Approach also applies to non-container PASOE
- Docker vs Crontab (crond)
 - *single process per-container*

■ Spring TaskScheduler to the rescue!

■ Process

- Implement ABL classes
- Write Spring (Java) task interface once
- Simple configuration via XML



Demonstration: Spring STS Project

■ Spring STS

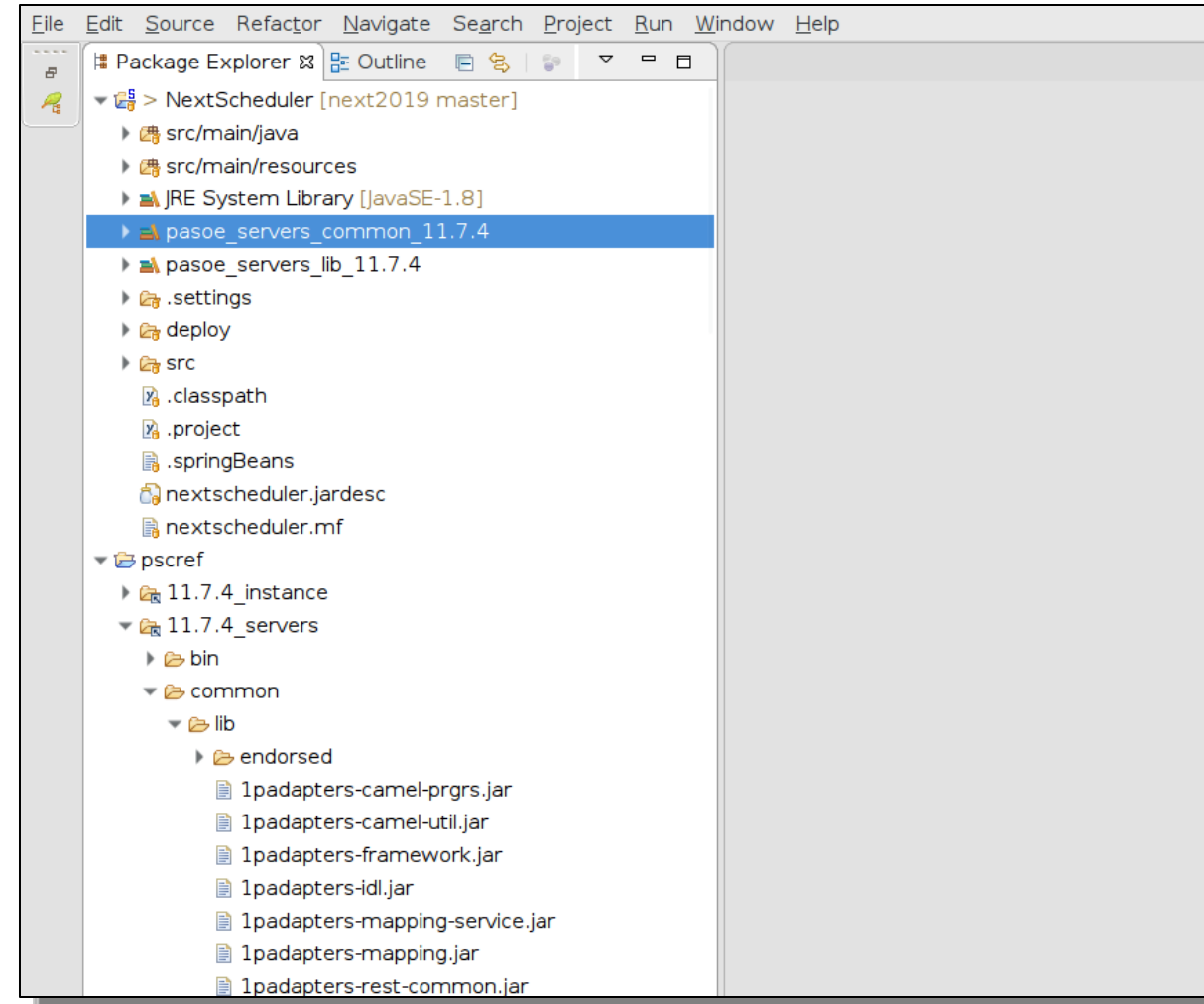
- Eclipse IDE, Java EE
 - *PDSOE should work*
- <https://spring.io/tools>

■ Reference all PASOE libraries

- Consider creating a Maven project
- <http://maven.apache.org/guides/getting-started/maven-in-fix-minutes.html>

■ Build a JAR file

- Make JAR and assets available for deployment to PASOE



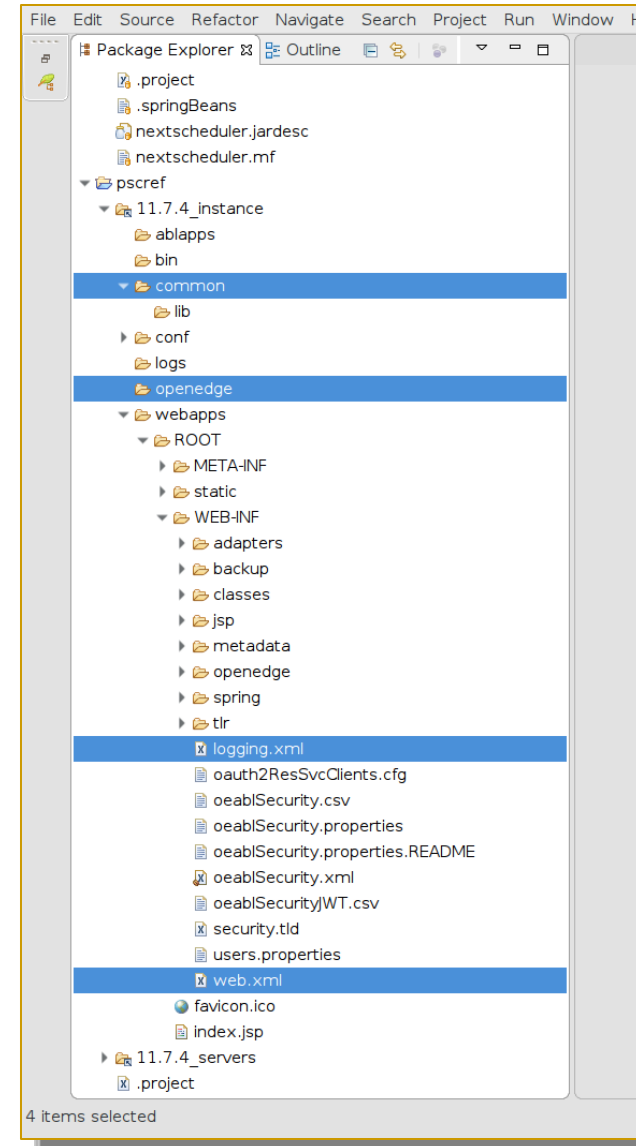
Demonstration: PASOE Key Locations

■ Instance

- **common/lib**
 - Classes and assets shared by webapps
- **openedge**
 - ABL code shared by ABLapps

■ WebApps

- webapps/<webapp>/WEB-INF
 - logging.xml - tweak verbosity
 - web.xml - implement customization



Demonstration: Build ABL Classes

- Design an interface
- Develop class implementation(s)
 - *Could also use procedures (.p)*

```
ITaskProcessor.cls
1 /*-----*/
2 File      : ITaskProcessor
3 Purpose   :
4 Syntax    :
5 Description :
6 Author(s) : chthomso@progress.com
7 Created   :
8 Notes     :
9 -----*/
10
11 interface psc.tasks.ITaskProcessor:
12     method public integer runAllScheduledTasks().
13 end interface.
```

```
QueueProcessor.cls
1 /*-----*/
2 File      : QueueProcessor
3 Purpose   :
4 Syntax    :
5 Description :
6 Author(s) : chthomso@progress.com
7 Created   :
8 Notes     :
9 -----*/
10
11 block-level on error undo, throw.
12
13 using OpenEdge.Logging.ISupportLogging from propath.
14
15 class psc.queue.QueueProcessor
16     implements ITaskProcessor,ISupportLogging
17     :
18     define public property Logger as OpenEdge.Logging.ILogWriter no-undo
19     get():
20         if not valid-object(Logger) then
21             assign Logger = OpenEdge.Logging.LoggerBuilder:GetLogger(this-object:GetClass()).
22             return Logger.
23         end get.
24     set.
25
26 /*-----*/
27 Purpose:
28 Notes:
29 -----*/
30
31 constructor public QueueProcessor ( ):
32     super ().
33
34     logger:trace('New QueueProcessor instance created').
35 end constructor.
36
37 method public integer runAllScheduledTasks( ):
38     define variable retcount as integer no-undo.
39
40     assign
41         retcount = random(1,5000).
42
43     logger:Debug(subst('Processed all requests: &1',retcount)).
44
45     return retcount.
46 end method.
47 end class.
```


Demonstration: Java Implementation

```
taskexec.xml
5  xmlns:task="http://www.springframework.org/schema/task"
6  xmlns:p="http://www.springframework.org/schema/p"
7  xsi:schemaLocation="http://www.springframework.org/schema
8      http://www.springframework.org/schema/beans http://www
9
10  <!-- see: https://docs.spring.io/spring/docs/4.3.23.RELEASE
11
12  <!-- Configure the scheduler/executer -->
13  <!-- Defines a ThreadPoolTaskScheduler instance with config
14  <task:scheduler
15      id="myscheduler"
16      pool-size="2" />
17
18  <!-- load the class (bean) that contains the task methods
19  <bean
20      id="taskExecutorExample"
21      class="psc.services.pasoescheduler.PasOEScheduler"
22      p:asURL="internal://nxgas"
23      p:oeClsName="psc.queue.QueueProcessor" />
24
25  <!-- maintain a list of scheduled tasks; associate with a
26  <task:scheduled-tasks scheduler="myscheduler">
27      <!-- exec every 15 seconds -->
28      <task:scheduled
29          ref="taskExecutorExample"
30          method="runOETasks"
31          cron="*/15 * * * * " />
32
```

```
PasOEScheduler.java
44  public String getObjectClassName() {
45      return m_clsName;
46  }
47
48  public void setObjectName(String oeClsName) {
49      this.m_clsName = oeClsName;
50  }
51
52  /**
53   * This method will exec OE Tasks on the AppServer
54   *
55   * @throws Exception
56   */
57  public void runOETasks() throws Exception {
58      // TODO: Assert that all required variables have values
59
60      log.debug("Running with ASURL:[{}], clsName:[{}]", m_asURL, m_clsName);
61
62      /**
63       * Call 04glrt/Proxy code
64       */
65      if (null == m_client) {
66          log.trace("Building Client");
67          createClient();
68      }
69
70      // collect parameters : if any
71      ParamArray params = new ParamArray(0);
72
73      // our implementation returns how many tasks were executed
74      params.setReturnType(Parameter.PRO_INTEGER);
75
76      // create classObject and invoke "runAllScheduled" method
77      this.m_client.getOpenClassObject().invokeMethod("runAllScheduledTasks", params);
78
79      // collect return value
80      Integer retval = (Integer) params.getReturnValue();
81      log.debug("Returnvalue from runAllSchedule: {}", retval);
82
83      // release class object
84      this.m_client.releaseClassObject();
85  }
86
```

- Xml Configuration
- Java OpenClient (essentially)
 - OpenClient development best-practices become **vital**

Demonstration: Implement Customization

```
web.xml
47 <!-- BEGIN:Spring security.definition -->
48 <!--
49 - Location of the Spring Security XML configuration file that
50 defines the which security model this web application should
51 employ. The configuration is loaded and applied by the
52 ContextLoaderListener.
53 -->
54 <context-param>
55   <param-name>contextConfigLocation</param-name>
56
57   <!-- All previous Spring Security configuration templates
58   have been retired and replaced by the single definition
59   below. All Spring Security configuration, including
60   the client login model and user account source is now
61   configured in the WEB-INF/oeablSecurity.properties file's
62   http.all.authmanager & client.login.model properties
63
64   NOTE: Please keep the param value in this format to enable
65   external administration
66
67   <param-value>/WEB-INF/oeablSecurity.xml</param-value>
68
69   -->
70   <param-value>
71     /WEB-INF/oeablSecurity.xml
72     classpath:taskexec.xml
73   </param-value>
74
75 </context-param>
76
77 <!-- Environment and PropertySource Abstraction -->
78 <context-param>
79   <param-name>contextInitializerClasses</param-name>
80   <param-value>com.progress.appserv.services.security.OESpringPropertySource</param-value>
81 </context-param>
82
83
```

- Add reference to custom XML in web.xml

Demonstration: Deploy and Run

- Deploy changes to PASOE environment normally
- pasoe.<datestamp>.log
- pasoe.agent.log

```
1 16:50:51.969/1427 [localhost-startStop-1] INFO o.s.s.c.ThreadPoolTaskScheduler - Initializing ExecutorService 'myscheduler'
2 16:50:52.023/1481 [localhost-startStop-1] DEBUG p.s.pasoescheduler.PasOEScheduler - New PasOEScheduler created
3 16:50:54.085/3543 [localhost-startStop-1] WARN com.progress.appserv.Prop - Development Server is limited to 1 agent
4 16:50:54.085/3543 [localhost-startStop-1] WARN com.progress.appserv.Prop - Development Server is limited to 5 concurrent requests
5 16:50:54.315/3773 [localhost-startStop-1] WARN c.p.appserv.IdleResourceWatchdog - Idle resource watchdog disabled
6 16:51:00.000/9458 [myscheduler-1] DEBUG p.s.pasoescheduler.PasOEScheduler - Running with ASURL:[internal://nxgas], clsName:[psc.queue.QueueProcessor]
7 16:51:00.001/9459 [myscheduler-1] TRACE p.s.pasoescheduler.PasOEScheduler - Building Client
8 16:51:00.005/9463 [myscheduler-1] TRACE p.s.pasoescheduler.PasOECClient - Attempting to Connect: isConnected: false
9 16:51:00.006/9464 [myscheduler-1] DEBUG p.s.pasoescheduler.PasOECClient - Creating new Connection to: internal://nxgas
10 16:51:00.277/9735 [myscheduler-1] DEBUG p.s.pasoescheduler.PasOEScheduler - Returnvalue from runAllSchedule: 829
11 16:51:15.000/24458 [myscheduler-1] DEBUG p.s.pasoescheduler.PasOEScheduler - Running with ASURL:[internal://nxgas], clsName:[psc.queue.QueueProcessor]
12 16:51:15.006/24464 [myscheduler-1] DEBUG p.s.pasoescheduler.PasOEScheduler - Returnvalue from runAllSchedule: 834
13 16:51:30.000/39458 [myscheduler-1] DEBUG p.s.pasoescheduler.PasOEScheduler - Running with ASURL:[internal://nxgas], clsName:[psc.queue.QueueProcessor]
14 16:51:30.007/ 28 @16:50:54.189-0400] P-000555 T-1942955776 2 AS-7 AS MSAS Session Startup. (5473)
15 16:51:45.000/ 29 @16:50:54.198-0400] P-000555 T-1942955776 1 AS-Aux-0 MSAS Worker Thread exiting. Number: 5, Status: 0
16 16:51:45.007/ 30 @16:50:54.309-0400] P-000555 T-1943484160 2 AS-Admin AS Application Server connected with connection id: . (8358)
17 16:51:46.206/ 31 @16:51:00.269-0400] P-000555 T-1941796608 1 AS-7 LogMgrWrtr QueueProcessor psc.queue.QueueProcessor TRACE: New QueueProcessor instance created
18 16:51:46.359/ 32 @16:51:00.270-0400] P-000555 T-1941796608 1 AS-7 LogMgrWrtr runAllScheduledTasks psc.queue.QueueProcessor DEBUG: Processed all requests: 829
19 16:51:46.368/ 33 @16:51:15.006-0400] P-000555 T-1941796608 1 AS-7 LogMgrWrtr runAllScheduledTasks psc.queue.QueueProcessor DEBUG: Processed all requests: 834
20 34 @16:51:30.006-0400] P-000555 T-1941796608 1 AS-7 LogMgrWrtr runAllScheduledTasks psc.queue.QueueProcessor DEBUG: Processed all requests: 3110
35 @16:51:45.006-0400] P-000555 T-1941796608 1 AS-7 LogMgrWrtr runAllScheduledTasks psc.queue.QueueProcessor DEBUG: Processed all requests: 1035
36 @16:51:46.204-0400] P-000555 T-2040751872 1 AS-Listener MSAS Agent Shutting Down. Status: 4020
37 @16:51:46.204-0400] P-000555 T-1941796608 1 AS-Aux-0 MSAS Worker Thread exiting. Number: 6, Status: 0
38 @16:51:46.206-0400] P-000555 T-2040751872 1 AS-Listener MSAS Agent Shutdown Complete.
```


What Else Can Spring Do?

Spring Integration

Enterprise Integration Patterns – messaging.
Simple XML configuration.

<https://spring.io/projects/spring-integration>

Spring Cloud

Cloud-based service configuration – Spring Consul.

<https://consul.io>

Secret management service – Spring Vault

<https://vaultproject.io>

<https://spring.io/projects/spring-cloud>

Spring Actuator

Administration and insights to health and security of Spring services.

Not a replacement for OEE, OEM or PASOE Health scanner

<https://spring.io/guides/gs/actuator-service/>

Spring Boot

Minimalistic configuration to bootstrap fully-functioning applications.

<https://spring.io/projects/spring-boot>



Where will you go NEX19 with Spring?



Questions?

Thank You!

 chthomso@progress.com

 <https://www.linkedin.com/in/chadrthomson/>

 <https://github.com/ChadThomsonPSC>

You might also like to attend

**Deploying Applications with the Docker Container
for Progress Application Server for OpenEdge**

Edsel Garcia

3:45 PM – 4:30 PM

