Quality-driven Multi-objective Optimization of Software Architecture Design: Method, Tool, and Application

Etemadi Idgahi, R.

Appendix A

AQOSA IR Sample Model

In this appendix, a sample of AQOSA IR (described in Chapter 4.3) is presented. Listing A.1 shows the source of the AQOSA IR model for the case study mentioned in Section 5.3. It is encoded in the Eclipse EMF [Ecl] format. This model is also accessible via this URL: http://goo.gl/nio8u8.

Listing A.1: *AQOSA IR EMF model for Saab Instrument Cluster system*

```
<?xml version="1.0" encoding="UTF-8"?>
<aqosa.ir:AQOSAModel xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:xsi="
    http://www.w3.org/2001/XMLSchema-instance" xmlns:aqosa.ir="http://se.liacs.
    nl/aqosa/ir">
  <assembly>
    <component name="ReadWheelSpeedSensors">
      <service name="ReadWheelSpeedSensors"/>
      <inport name="ReadWheelSpeedSensors-in"/>
      <outport name="ReadWheelSpeedSensors-out"/>
    </component>
    <component name="ControlWheelSpeed">
      <service name="CalculateWheelRotation"/>
      <inport name="ControlWheelSpeed-in"/>
      <outport name="ControlWheelSpeed-out"/>
    </component>
    <component name="EngineVehicleInterface">
      <service name="ObtainEngineSpeed"/>
      <service name="ObtainVehicleSpeed"/>
      <service name="ObtainCoolantTemp"/>
      <inport name="EngineVehicleInterface-in_Engine"/>
      <inport name="EngineVehicleInterface-in_Vehicle"/>
      <inport name="EngineVehicleInterface-in_Coolant"/>
      <outport name="EngineVehicleInterface-out_Engine"/>
      <outport name="EngineVehicleInterface-out_Vehicle"/>
      <outport name="EngineVehicleInterface-out_Coolant"/>
    </component>
    <component name="ProvidePowerModeInfo">
      <service name="PowerModeInfo"/>
      <outport name="PowerModeInfo-out"/>
    </component>
```

```
29      <component name="ControlEngineSpeedGauge">
30        <service name="DisplayEngineSpeed"/>
31        <inport name="ControlEngineSpeedGauge-in"/>
32        <outport name="ControlEngineSpeedGauge-out"/>
33      </component>
34      <component name="ControlVehicleSpeedGauge">
35        <service name="DisplayVehicleSpeed"/>
36        <inport name="ControlVehicleSpeedGauge-in"/>
37        <outport name="ControlVehicleSpeedGauge-out"/>
38      </component>
39      <component name="Gauge_Engine">
40        <service name="CalculateNeedlePosition"/>
41        <inport name="Gauge_Engine-in"/>
42      </component>
43      <component name="TransmissionVehicleInterface">
44        <service name="ReadLeverPstn"/>
45        <outport name="TransmissionVehicleInterface-out"/>
46      </component>
47      <component name="ControlGearSelectedIndication">
48        <service name="GearDisplayValue"/>
49        <inport name="ControlGearSelectedIndication-in"/>
50        <outport name="ControlGearSelectedIndication-out"/>
51      </component>
52      <component name="Display_Engine">
53        <service name="IndicateGearPstn"/>
54        <service name="DisplayOAT"/>
55        <service name="DisplayOdometer"/>
56        <service name="IndicateLowWasher"/>
57        <inport name="Display_Engine-in_Gear"/>
58        <inport name="Display_Engine-in_OAT"/>
59        <inport name="Display_Engine-in_Odometer"/>
60        <inport name="Display_Engine-in_Washer"/>
61      </component>
62      <component name="ReadOATSensor">
63        <service name="ObtaionOAT"/>
64        <outport name="ReadOATSensor-out"/>
65      </component>
66      <component name="ControlOutsideAirTemp">
67        <service name="CalculateOAT"/>
68        <inport name="ControlOutsideAirTemp-in"/>
69        <outport name="ControlOutsideAirTemp-out"/>
70      </component>
71      <component name="ControlCoolantTempGauge">
72        <service name="DisplayCoolantTemp"/>
73        <inport name="ControlCoolantTempGauge-in"/>
74        <outport name="ControlCoolantTempGauge-out"/>
75      </component>
76      <component name="ReadDriverDoorAjarSwitch">
77        <service name="ReadDriverDoorAjarSwitch"/>
78        <outport name="ReadDriverDoorAjarSwitch-out"/>
79      </component>
80      <component name="ControlOdometer">
81        <service name="OdometerValue"/>
82        <inport name="ControlOdometer-in"/>
83        <outport name="ControlOdometer-out"/>
84      </component>
85      <component name="ReadTripStemButton">
86        <service name="ReadTripStemButton"/>
87        <outport name="ReadTripStemButton-out"/>
88      </component>
```

```
89      <component name="ReadLowWasherLevel">
90        <service name="ReadLowWasherLevel"/>
91        <outport name="ReadLowWasherLevel-out"/>
92      </component>
93      <component name="ControlWasherLevelIndication">
94        <service name="ControlWasherLevelIndication"/>
95        <inport name="ControlWasherLevelIndication-in"/>
96        <outport name="ControlWasherLevelIndication-out"/>
97      </component>
98      <flow name="Ignition_to_EngineSpeed">
99        <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.3
              /@service.0"/>
100       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.3
              /@outport.0" destination="//@assembly/@component.2/@inport.0"/>
101       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.2
              /@service.0"/>
102       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.2
              /@outport.0" destination="//@assembly/@component.4/@inport.0"/>
103       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.4
              /@service.0"/>
104       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.4
              /@outport.0" destination="//@assembly/@component.6/@inport.0"/>
105       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.6
              /@service.0"/>
106     </flow>
107     <flow name="Ignition_to_VehicleSpeed">
108       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.3
              /@service.0"/>
109       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.3
              /@outport.0" destination="//@assembly/@component.0/@inport.0"/>
110       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.0
              /@service.0"/>
111       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.0
              /@outport.0" destination="//@assembly/@component.1/@inport.0"/>
112       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.1
              /@service.0"/>
113       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.1
              /@outport.0" destination="//@assembly/@component.2/@inport.1"/>
114       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.2
              /@service.1"/>
115       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.2
              /@outport.1" destination="//@assembly/@component.5/@inport.0"/>
116       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.5
              /@service.0"/>
117       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.5
              /@outport.0" destination="//@assembly/@component.6/@inport.0"/>
118       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.6
              /@service.0"/>
119     </flow>
120     <flow name="GearIndication">
121       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.7
              /@service.0"/>
122       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.7
              /@outport.0" destination="//@assembly/@component.8/@inport.0"/>
123       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.8
              /@service.0"/>
124       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.8
              /@outport.0" destination="//@assembly/@component.9/@inport.0"/>
125       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.9
              /@service.0"/>
```

```
126     </flow>
127     <flow name="VehicleSpeedIndication">
128       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.0
              /@service.0"/>
129       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.0
              /@outport.0" destination="//@assembly/@component.1/@inport.0"/>
130       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.1
              /@service.0"/>
131       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.1
              /@outport.0" destination="//@assembly/@component.2/@inport.1"/>
132       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.2
              /@service.1"/>
133       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.2
              /@outport.1" destination="//@assembly/@component.5/@inport.0"/>
134       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.5
              /@service.0"/>
135       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.5
              /@outport.0" destination="//@assembly/@component.6/@inport.0"/>
136       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.6
              /@service.0"/>
137     </flow>
138     <flow name="EngineSpeedIndication">
139       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.2
              /@service.0"/>
140       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.2
              /@outport.0" destination="//@assembly/@component.4/@inport.0"/>
141       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.4
              /@service.0"/>
142       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.4
              /@outport.0" destination="//@assembly/@component.6/@inport.0"/>
143       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.6
              /@service.0"/>
144     </flow>
145     <flow name="OATCalculation">
146       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.10
              /@service.0"/>
147       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.10
              /@outport.0" destination="//@assembly/@component.11/@inport.0"/>
148       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.11
              /@service.0"/>
149       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.11
              /@outport.0" destination="//@assembly/@component.9/@inport.1"/>
150       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.9
              /@service.1"/>
151     </flow>
152     <flow name="EngineCoolantTemp">
153       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.2
              /@service.2"/>
154       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.2
              /@outport.2" destination="//@assembly/@component.12/@inport.0"/>
155       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.12
              /@service.0"/>
156       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.12
              /@outport.0" destination="//@assembly/@component.6/@inport.0"/>
157       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.6
              /@service.0"/>
158     </flow>
159     <flow name="DriverDoor_to_Odometer">
160       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.13
              /@service.0"/>
```

```
161       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.13
              /@outport.0" destination="//@assembly/@component.14/@inport.0"/>
162       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.14
              /@service.0"/>
163       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.14
              /@outport.0" destination="//@assembly/@component.9/@inport.2"/>
164       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.9
              /@service.2"/>
165     </flow>
166     <flow name="StemButton_to_Odometer">
167       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.15
              /@service.0"/>
168       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.15
              /@outport.0" destination="//@assembly/@component.14/@inport.0"/>
169       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.14
              /@service.0"/>
170       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.14
              /@outport.0" destination="//@assembly/@component.9/@inport.2"/>
171       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.9
              /@service.2"/>
172     </flow>
173     <flow name="LowWasherIndication">
174       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.16
              /@service.0"/>
175       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.16
              /@outport.0" destination="//@assembly/@component.17/@inport.0"/>
176       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.17
              /@service.0"/>
177       <action xsi:type="aqosa.ir:CommunicateAction" source="//@assembly/@component.17
              /@outport.0" destination="//@assembly/@component.9/@inport.3"/>
178       <action xsi:type="aqosa.ir:ComputeAction" service="//@assembly/@component.9
              /@service.3"/>
179     </flow>
180   </assembly>
181   <scenarios>
182     <flowset name="Average" completionTime="10000.0" missedPercentage="0.05">
183       <flowinstance instance="//@assembly/@flow.0" start="300.0" trigger="5000.0"
              deadline="75.0"/>
184       <flowinstance instance="//@assembly/@flow.1" start="400.0" trigger="5000.0"
              deadline="75.0"/>
185       <flowinstance instance="//@assembly/@flow.2" start="500.0" trigger="5000.0"
              deadline="50.0"/>
186       <flowinstance instance="//@assembly/@flow.3" start="1000.0" trigger="100.0"
              deadline="50.0"/>
187       <flowinstance instance="//@assembly/@flow.4" start="1030.0" trigger="100.0"
              deadline="35.0"/>
188       <flowinstance instance="//@assembly/@flow.5" start="100.0" trigger="1000.0"
              deadline="50.0"/>
189       <flowinstance instance="//@assembly/@flow.6" start="1070.0" trigger="100.0"
              deadline="30.0"/>
190       <flowinstance instance="//@assembly/@flow.7" start="2000.0" trigger="5000.0"
              deadline="250.0"/>
191       <flowinstance instance="//@assembly/@flow.8" start="2500.0" trigger="5000.0"
              deadline="250.0"/>
192       <flowinstance instance="//@assembly/@flow.9" start="3000.0" trigger="5000.0"
              deadline="125.0"/>
193     </flowset>
194   </scenarios>
195   <repository>
196     <componentinstance id="ReadWheelSpeedSensors_Instance" compatible="
```

```
          //@assembly/@component.0" variancePercentage="0.05">
197       <service instance="//@assembly/@component.0/@service.0" cycles="600" networkUsage="
              4000.0">
198         <provide connects="//@assembly/@component.0/@outport.0"/>
199         <depend>
200           <require external="//@repository/@externalport.4"/>
201           <require internal="//@assembly/@component.0/@inport.0"/>
202         </depend>
203       </service>
204     </componentinstance>
205     <componentinstance id="ControlWheelSpeed_Instance" compatible="//@assembly/@component
            .1" variancePercentage="0.05">
206       <service instance="//@assembly/@component.1/@service.0" cycles="500" networkUsage="
              4000.0">
207         <provide connects="//@assembly/@component.1/@outport.0"/>
208         <depend>
209           <require internal="//@assembly/@component.1/@inport.0"/>
210         </depend>
211       </service>
212     </componentinstance>
213     <componentinstance id="EngineVehicleInterface_Instance" compatible="
            //@assembly/@component.2" variancePercentage="0.05">
214       <service instance="//@assembly/@component.2/@service.0" cycles="500" networkUsage="
              2000.0">
215         <provide connects="//@assembly/@component.2/@outport.0"/>
216         <depend>
217           <require external="//@repository/@externalport.3"/>
218           <require internal="//@assembly/@component.2/@inport.0"/>
219         </depend>
220       </service>
221       <service instance="//@assembly/@component.2/@service.1" cycles="500" networkUsage="
              2000.0">
222         <provide connects="//@assembly/@component.2/@outport.1"/>
223         <depend>
224           <require internal="//@assembly/@component.2/@inport.1"/>
225         </depend>
226       </service>
227       <service instance="//@assembly/@component.2/@service.2" cycles="500" networkUsage="
              1000.0">
228         <provide connects="//@assembly/@component.2/@outport.2"/>
229         <depend>
230           <require internal="//@assembly/@component.2/@inport.2"/>
231         </depend>
232       </service>
233     </componentinstance>
234     <componentinstance id="ProvidePowerModeInfo_Instance" compatible="
            //@assembly/@component.3" variancePercentage="0.05">
235       <service instance="//@assembly/@component.3/@service.0" cycles="400" networkUsage="
              1000.0">
236         <provide connects="//@assembly/@component.3/@outport.0"/>
237         <depend>
238           <require external="//@repository/@externalport.2"/>
239         </depend>
240       </service>
241     </componentinstance>
242     <componentinstance id="ControlEngineSpeedGauge_Instance" compatible="
            //@assembly/@component.4" variancePercentage="0.05">
243       <service instance="//@assembly/@component.4/@service.0" cycles="2850" networkUsage=
              "2000.0">
244         <provide connects="//@assembly/@component.4/@outport.0"/>
```

```
245         <depend>
246           <require internal="//@assembly/@component.4/@inport.0"/>
247         </depend>
248       </service>
249     </componentinstance>
250     <componentinstance id="ControlVehicleSpeedGauge_Instance" compatible="
            //@assembly/@component.5" variancePercentage="0.05">
251       <service instance="//@assembly/@component.5/@service.0" cycles="2950" networkUsage=
              "2000.0">
252         <provide connects="//@assembly/@component.5/@outport.0"/>
253         <depend>
254           <require internal="//@assembly/@component.5/@inport.0"/>
255         </depend>
256       </service>
257     </componentinstance>
258     <componentinstance id="Gauge_Engine_Instance" compatible="//@assembly/@component.6"
            variancePercentage="0.05">
259       <service instance="//@assembly/@component.6/@service.0" cycles="500">
260         <depend>
261           <require internal="//@assembly/@component.6/@inport.0"/>
262         </depend>
263       </service>
264     </componentinstance>
265     <componentinstance id="TransmissionVehicleInterface_Instance" compatible="
            //@assembly/@component.7" variancePercentage="0.05">
266       <service instance="//@assembly/@component.7/@service.0" cycles="100" networkUsage="
              1000.0">
267         <provide connects="//@assembly/@component.7/@outport.0"/>
268         <depend>
269           <require external="//@repository/@externalport.5"/>
270         </depend>
271       </service>
272     </componentinstance>
273     <componentinstance id="ControlGearSelectedIndication_Instance" compatible="
            //@assembly/@component.8" variancePercentage="0.05">
274       <service instance="//@assembly/@component.8/@service.0" cycles="2500" networkUsage=
              "1000.0">
275         <provide connects="//@assembly/@component.8/@outport.0"/>
276         <depend>
277           <require internal="//@assembly/@component.8/@inport.0"/>
278         </depend>
279       </service>
280     </componentinstance>
281     <componentinstance id="Display_Engine_Instance" cost="55.0" compatible="
            //@assembly/@component.9" variancePercentage="0.05">
282       <service instance="//@assembly/@component.9/@service.0" cycles="500" networkUsage="
              1000.0">
283         <depend>
284           <require internal="//@assembly/@component.9/@inport.0"/>
285         </depend>
286       </service>
287       <service instance="//@assembly/@component.9/@service.1" cycles="500" networkUsage="
              1000.0">
288         <depend>
289           <require internal="//@assembly/@component.9/@inport.1"/>
290         </depend>
291       </service>
292       <service instance="//@assembly/@component.9/@service.2" cycles="500" networkUsage="
              1000.0">
293         <depend>
```

```
294             <require internal="//@assembly/@component.9/@inport.2"/>
295           </depend>
296         </service>
297         <service instance="//@assembly/@component.9/@service.3" cycles="500" networkUsage="
                1000.0">
298           <depend>
299             <require internal="//@assembly/@component.9/@inport.3"/>
300           </depend>
301         </service>
302       </componentinstance>
303       <componentinstance id="ReadOATSensor_Instance" compatible="//@assembly/@component.10"
              variancePercentage="0.05">
304         <service instance="//@assembly/@component.10/@service.0" cycles="1000" networkUsage
              ="1000.0">
305           <provide connects="//@assembly/@component.10/@outport.0"/>
306           <depend>
307             <require external="//@repository/@externalport.6"/>
308           </depend>
309         </service>
310       </componentinstance>
311       <componentinstance id="ControlOutsideAirTemp_Instance" compatible="
              //@assembly/@component.11" variancePercentage="0.05">
312         <service instance="//@assembly/@component.11/@service.0" cycles="2744" networkUsage
              ="1000.0">
313           <provide connects="//@assembly/@component.11/@outport.0"/>
314           <depend>
315             <require internal="//@assembly/@component.11/@inport.0"/>
316           </depend>
317         </service>
318       </componentinstance>
319       <componentinstance id="ControlCoolantTempGauge_Instance" compatible="
              //@assembly/@component.12" variancePercentage="0.05">
320         <service instance="//@assembly/@component.12/@service.0" cycles="1500" networkUsage
              ="1000.0">
321           <provide connects="//@assembly/@component.12/@outport.0"/>
322           <depend>
323             <require internal="//@assembly/@component.12/@inport.0"/>
324           </depend>
325         </service>
326       </componentinstance>
327       <componentinstance id="ReadDriverDoorAjarSwitch_Instance" cost="1.0" compatible="
              //@assembly/@component.13" variancePercentage="0.05">
328         <service instance="//@assembly/@component.13/@service.0" cycles="100" networkUsage=
              "1000.0">
329           <provide connects="//@assembly/@component.13/@outport.0"/>
330           <depend>
331             <require external="//@repository/@externalport.0"/>
332           </depend>
333         </service>
334       </componentinstance>
335       <componentinstance id="ControlOdometer_Instance" compatible="//@assembly/@component
              .14" variancePercentage="0.05">
336         <service instance="//@assembly/@component.14/@service.0" cycles="2440" networkUsage
              ="4000.0">
337           <provide connects="//@assembly/@component.14/@outport.0"/>
338           <depend>
339             <require internal="//@assembly/@component.14/@inport.0"/>
340             <require external="//@repository/@externalport.7"/>
341           </depend>
342         </service>
```

```
343     </componentinstance>
344     <componentinstance id="ReadTripStemButton_Instance" compatible="
            //@assembly/@component.15" variancePercentage="0.05">
345       <service instance="//@assembly/@component.15/@service.0" cycles="100" networkUsage=
              "1000.0">
346         <provide connects="//@assembly/@component.15/@outport.0"/>
347         <depend>
348           <require external="//@repository/@externalport.1"/>
349         </depend>
350       </service>
351     </componentinstance>
352     <componentinstance id="ReadLowWasherLevel_Instance" compatible="
            //@assembly/@component.16" variancePercentage="0.05">
353       <service instance="//@assembly/@component.16/@service.0" cycles="100" networkUsage=
              "1000.0">
354         <provide connects="//@assembly/@component.16/@outport.0"/>
355         <depend>
356           <require external="//@repository/@externalport.8"/>
357         </depend>
358       </service>
359     </componentinstance>
360     <componentinstance id="ControlWasherLevelIndication_Instance" compatible="
            //@assembly/@component.17" variancePercentage="0.05">
361       <service instance="//@assembly/@component.17/@service.0" cycles="300" networkUsage=
              "1000.0">
362         <provide connects="//@assembly/@component.17/@outport.0"/>
363         <depend>
364           <require internal="//@assembly/@component.17/@inport.0"/>
365         </depend>
366       </service>
367     </componentinstance>
368     <processor id="cpu066-h" clock="66.0" cost="100.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
369     <processor id="cpu066-l" clock="66.0" cost="140.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.01" upperFail="0.025"/>
370     <processor id="cpu100-h" clock="100.0" cost="125.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
371     <processor id="cpu100-l" clock="100.0" cost="175.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.01" upperFail="0.025"/>
372     <processor id="cpu133-h" clock="133.0" cost="150.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
373     <processor id="cpu133-l" clock="133.0" cost="210.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.01" upperFail="0.025"/>
374     <processor id="cpu166-h" clock="166.0" cost="175.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
375     <processor id="cpu166-l" clock="166.0" cost="245.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.01" upperFail="0.025"/>
376     <processor id="cpu200-h" clock="200.0" cost="200.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
377     <processor id="cpu200-l" clock="200.0" cost="280.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.01" upperFail="0.025"/>
378     <processor id="cpu233-h" clock="233.0" cost="225.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
379     <processor id="cpu233-l" clock="233.0" cost="315.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.01" upperFail="0.025"/>
380     <processor id="cpu266-h" clock="266.0" cost="250.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.02" upperFail="0.035"/>
381     <processor id="cpu266-l" clock="266.0" cost="350.0" internalBusBandwidth="1024.0"
            internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
382     <processor id="cpu300-h" clock="300.0" cost="275.0" internalBusBandwidth="1024.0"
```

```
        internalBusDelay="0.1" lowerFail="0.02" upperFail="0.035"/>
    <processor id="cpu300-l" clock="300.0" cost="385.0" internalBusBandwidth="1024.0"
        internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
    <processor id="cpu333-h" clock="333.0" cost="300.0" internalBusBandwidth="1024.0"
        internalBusDelay="0.1" lowerFail="0.02" upperFail="0.035"/>
    <processor id="cpu333-l" clock="333.0" cost="420.0" internalBusBandwidth="1024.0"
        internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
    <processor id="cpu366-h" clock="366.0" cost="325.0" internalBusBandwidth="1024.0"
        internalBusDelay="0.1" lowerFail="0.02" upperFail="0.035"/>
    <processor id="cpu366-l" clock="366.0" cost="455.0" internalBusBandwidth="1024.0"
        internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
    <processor id="cpu400-h" clock="400.0" cost="350.0" internalBusBandwidth="1024.0"
        internalBusDelay="0.1" lowerFail="0.02" upperFail="0.035"/>
    <processor id="cpu400-l" clock="400.0" cost="490.0" internalBusBandwidth="1024.0"
        internalBusDelay="0.1" lowerFail="0.015" upperFail="0.03"/>
    <bus id="CAN-HS" bandwidth="500.0" delay="0.002" cost="100.0"/>
    <bus id="CAN-MS" bandwidth="125.0" delay="0.008" cost="50.0"/>
    <bus id="CAN-LS" bandwidth="33.3" delay="0.016" cost="25.0"/>
    <bus id="LIN" bandwidth="10.0" delay="0.05" cost="10.0"/>
    <externalport id="ajar-switch" lowerFail="0.01" upperFail="0.05"/>
    <externalport id="stem-button" lowerFail="0.01" upperFail="0.05"/>
    <externalport id="ignition-switch" lowerFail="0.01" upperFail="0.05"/>
    <externalport id="crankshaft-sensor" lowerFail="0.01" upperFail="0.05"/>
    <externalport id="wheel-sensor" lowerFail="0.01" upperFail="0.05"/>
    <externalport id="gear-sensor" lowerFail="0.01" upperFail="0.05"/>
    <externalport id="oat-sensor" lowerFail="0.01" upperFail="0.05"/>
    <externalport id="odometer-storage" lowerFail="0.01" upperFail="0.05"/>
    <externalport id="lowwasher-switch" lowerFail="0.01" upperFail="0.05"/>
  </repository>
  <objectives>
    <settings noRun="3" noSampling="50" noDuplicate="1" minCost="200.0" maxCost="10000.0"
        >
      <evaluations>ResponseTime</evaluations>
      <evaluations>CPUUtilization</evaluations>
      <evaluations>BusUtilization</evaluations>
      <evaluations>Safety</evaluations>
      <evaluations>Cost</evaluations>
    </settings>
  </objectives>
</aqosa.ir:AQOSAModel>
```

List of Figures

List of Tables

Bibliography

[ABG+14] Aldeida Aleti, Barbora Buhnova, Lars Grunske, Anne Koziolek, and Indika Meedeniya. Software architecture optimization methods: A systematic literature review. In Wilhelm Hasselbring and Nils Christian Ehmke, editors, *Software Engineering*, volume 227 of *LNI*, pages 77–78. GI, 2014. URL: http://eprints.uni-kiel.de/23752/. (cited on page 99).

[ABGM09] Aldeida Aleti, Stefan Björnander, Lars Grunske, and Indika Meedeniya. ArcheOpterix: An extendable tool for architecture optimization of AADL models. In *MOMPES*, pages 61–71. IEEE Computer Society, 2009. doi:10.1109/MOMPES.2009.5069138. (cited on page 18).

[ABM+13] Michal Antkiewicz, Kacper Bak, Alexandr Murashkin, Rafael Olaechea, Jia Hui (Jimmy) Liang, and Krzysztof Czarnecki. Clafer tools for product line engineering. In *SPLC Workshops*, pages 130–135. ACM, 2013. doi:10.1145/2499777.2499779. (cited on page 117).

[AGHIR12] Silvia Abrahão, Javier Gonzalez-Huerta, Emilio Insfrán, and Isidro Ramos. Software evolution in model-driven product line engineering. *ERCIM News*, 2012(88), 2012. (cited on page 24).

[Akk] Akka Framework: an open-source toolkit and runtime simplifying the construction of concurrent and distributed applications on the JVM. http://akka.io/. URL: http://akka.io/. (cited on page 139).

[ALBG99] Daniel Amyot, Luigi Logrippo, Raymond J. A. Buhr, and Tom Gray. Use case maps for the capture and validation of distributed systems requirements. In *RE*, page 44. IEEE Computer Society, 1999. doi:10.1109/ISRE.1999.777984. (cited on page 56).

[Axe09] Jakob Axelsson. Evolutionary architecting of embedded automotive product lines: An industrial case study. In *WICSA/ECSA*, pages 101–110. IEEE, 2009. doi:10.1109/WICSA.2009.5290796. (cited on page 82).

[Bäc96] Thomas Bäck. *Evolutionary algorithms in theory and practice - evolution strategies, evolutionary programming, genetic algorithms*. Oxford University Press, 1996. URL: http://dl.acm.org/citation.cfm?id=229867. (cited on pages 7 and 8).

[BBL10] Michael Bowman, Lionel C. Briand, and Yvan Labiche. Solving the class responsibility assignment problem in object-oriented analysis with multi-objective genetic algorithms. *IEEE Trans. Software Eng.*, 36(6):817–837, 2010. doi:10.1109/TSE.2010.70. (cited on page 22).

[BCdW06] Egor Bondarev, Michel R. V. Chaudron, and Peter H. N. de With. A Process for Resolving Performance Trade-Offs in Component-Based Architectures. In Ian Gorton and et al., editors, *CBSE*, volume 4063 of *LNCS*, pages 254–269. Springer, 2006. doi:10.1007/11783565_18. (cited on pages 17 and 27).

[BCK03] Len Bass, Paul Clements, and Rick Kazman. *Software Architecture in Practice*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2 edition, 2003. URL: http://dl.acm.org/citation.cfm?id=773239. (cited on page 14).

[BKLW95] Mario Barbacci, Mark H. Klein, Thomas A. Longstaff, and Charles B. Weinstock. Quality Attributes. Technical Report CMU/SEI-95-TR-021, Carnegie Mellon, 1995. URL: http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=12433. (cited on pages 14 and 43).

[BKR09] Steffen Becker, Heiko Koziolek, and Ralf Reussner. The Palladio component model for model-driven performance prediction. *Journal of Systems and Software*, 82(1):3–22, 2009. doi:10.1016/j.jss.2008.03.066. (cited on pages 13, 17 and 27).

[BNE07] Nicola Beume, Boris Naujoks, and Michael Emmerich. Sms-emoa: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research*, 181(3):1653–1669, 2007. doi:10.1016/j.ejor.2006.08.008. (cited on pages 34 and 52).

[BR03] Christian Blum and Andrea Roli. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3):268–308, 2003. doi:10.1145/937503.937505. (cited on page 2).

[BS14] Maxim Buzdalov and Anatoly Shalyto. A provably asymptotically fast version of the generalized jensen algorithm for non-dominated sorting. In Thomas Bartz-Beielstein, Jürgen Branke, Bogdan Filipic, and Jim Smith, editors, *Parallel Problem Solving from Nature - PPSN XIII - 13th International Conference, Ljubljana, Slovenia, September 13-17, 2014. Proceedings*, volume 8672 of *Lecture Notes in Computer Science*, pages 528–537. Springer, 2014. doi:10.1007/978-3-319-10762-2_52. (cited on page 48).

[CFD+11] Andrea Ciancone, Antonio Filieri, Mauro Luigi Drago, Raffaela Mirandola, and Vincenzo Grassi. Klapersuite: An integrated model-driven environment for reliability and performance analysis of component-based systems. In Judith Bishop and Antonio Vallecillo, editors, *TOOLS (49)*, volume 6705 of *LNCS*, pages 99–114. Springer, 2011. doi:10.1007/978-3-642-21952-8_9. (cited on page 19).

[CFL+10] Marco Ceriani, Fabrizio Ferrandi, Pier Luca Lanzi, Donatella Sciuto, and Antonino Tumeo. Multiprocessor systems-on-chip synthesis using multi-objective evolutionary computation. In Martin Pelikan and Jürgen Branke, editors, *Genetic and Evolutionary Computation Conference, GECCO 2010, Proceedings, Portland, Oregon, USA, July 7-11, 2010*, pages 1267–1274. ACM, 2010. doi:10.1145/1830483.1830710. (cited on page 20).

[Clo] CloudBees. CloudBees: The Java PaaS (Platform-as-a-Service). URL: http://www.cloudbees.com/. (cited on page 47).

[CMT10] Vittorio Cortellessa, Antinisca Di Marco, and Catia Trubiani. Performance antipatterns as logical predicates. In Radu Calinescu, Richard F. Paige, and Marta Z. Kwiatkowska, editors, *ICECCS*, pages 146–156. IEEE Computer Society, 2010. doi:10.1109/ICECCS.2010.44. (cited on page 102).

[CN01] Paul C. Clements and Linda Northrop. *Software Product Lines: Practices and Patterns*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2001. URL: http://dl.acm.org/citation.cfm?id=501065. (cited on page 2).

[Col12] Thelma Elita Colanzi. Search based design of software product lines architectures. In Martin Glinz, Gail C. Murphy, and Mauro Pezzè, editors, *ICSE*, pages 1507–1510. IEEE, 2012. doi:10.1109/ICSE.2012.6227050. (cited on page 23).

[CPSV08] Vittorio Cortellessa, Pierluigi Pierini, Romina Spalazzese, and Alessio Vianale. MOSES: MOdeling Software and platform architEcture in UML 2 for Simulation-based performance analysis. In Steffen Becker, Frantisek Plasil, and Ralf Reussner, editors, *QoSA*, volume 5281 of *LNCS*, pages 86–102. Springer, 2008. doi:10.1007/978-3-540-87879-7_6. (cited on page 19).

[CV12] Thelma Elita Colanzi and Silvia Regina Vergilio. Applying search based optimization to software product line architectures: Lessons learned. In Gordon Fraser and Jerffeson Teixeira de Souza, editors, *SSBSE*, volume 7515 of *Lecture Notes in Computer Science*, pages 259–266. Springer, 2012. doi:10.1007/978-3-642-33119-0_19. (cited on page 23).

[DAPM02] Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002. doi:10.1109/4235.996017. (cited on pages 33 and 52).

[Depa] Department of Computer Science at the University of Erlangen-Nuremberg. Opt4J: A Modular Framework for Meta-heuristic Optimization. URL: http://opt4j.sourceforge.net/. (cited on pages 33 and 47).

[Depb] Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology. Alloy: a language and tool for relational models. URL: http://alloy.mit.edu/alloy/index.html. (cited on page 118).

[DG04] Jeffrey Dean and Sanjay Ghemawat. Mapreduce: Simplified data processing on large clusters. In Eric A. Brewer and Peter Chen, editors, *6th Symposium on Operating System Design and Implementation (OSDI 2004), San Francisco, California, USA, December 6-8, 2004*, pages 137–150. USENIX Association, 2004. URL: http://www.usenix.org/events/osdi04/tech/dean.html. (cited on pages 135 and 136).

[DW00] Stefan Droste and Dirk Wiesmann. Metric based evolutionary algorithms. In Riccardo Poli, Wolfgang Banzhaf, William B. Langdon, Julian F. Miller, Peter Nordin, and Terence C. Fogarty, editors, *EuroGP*, volume 1802 of *Lecture Notes in Computer Science*, pages 29–43. Springer, 2000. URL: http://dl.acm.org/citation.cfm?id=646808.703953. (cited on page 100).

[EBN05] Michael Emmerich, Nicola Beume, and Boris Naujoks. An EMO algorithm using the hypervolume measure as selection criterion. In Carlos A. Coello Coello, Arturo Hernández Aguirre, and Eckart Zitzler, editors, *Evolutionary Multi-Criterion Optimization, Third International Conference, EMO 2005, Guanajuato, Mexico, March 9-11, 2005, Proceedings*, volume 3410 of *Lecture Notes in Computer Science*, pages 62–76. Springer, 2005. doi:10.1007/978-3-540-31880-4_5. (cited on page 34).

[Ecl] Eclipse open source community and Eclipse Foundation. Eclipse Modeling Framework (EMF). URL: http://www.eclipse.org/modeling/emf/. (cited on pages 30 and 151).

[EF11] Michael T. M. Emmerich and Carlos M. Fonseca. Computing hypervolume contributions in low dimensions: Asymptotically optimal algorithm and complexity results. In Ricardo H. C. Takahashi, Kalyanmoy Deb, Elizabeth F. Wanner, and Salvatore Greco, editors, *Evolutionary Multi-Criterion Optimization - 6th International Conference, EMO 2011, Ouro Preto, Brazil, April 5-8, 2011. Proceedings*, volume 6576 of *Lecture Notes in Computer Science*, pages 121–135. Springer, 2011. doi:10.1007/978-3-642-19893-9_9. (cited on page 48).

[EG13] Ulrik Eklund and Håkan Gustavsson. Architecting automotive product lines: Industrial practice. *Sci. Comput. Program.*, 78(12):2347–2359, 2013. doi:10.1016/j.scico.2012.06.008. (cited on page 120).

[EGS01] Michael Emmerich, Monika Grötzner, and Martin Schütz. Design of graph-based evolutionary algorithms: A case study for chemical process networks. *Evolutionary Computation*, 9(3):329–354, 2001. doi:10.1162/106365601750406028. (cited on page 100).

[EM08] Leire Etxeberria and Goiuria Sagardui Mendieta. Variability driven quality evaluation in software product lines. In *SPLC*, pages 243–252. IEEE Computer Society, 2008. doi:10.1109/SPLC.2008.37. (cited on page 23).

[FGH06] Peter H. Feiler, David Gluch, and John J. Hudak. The Architecture Analysis & Design Language (AADL): An Introduction. Technical Report CMU/SEI-2006-TN-011, Carnegie Mellon, 2006. URL: http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=7879. (cited on pages 18 and 27).

[Fie10] Tony Field. *JINQS: An Extensible Library for Simulating Multiclass Queueing Networks*, 2010. URL: http://www.doc.ic.ac.uk/~ajf/Research/manual.pdf. (cited on page 43).

[FS10] Marc Förster and Daniel Schneider. Flexible, Any-Time Fault Tree Analysis with Component Logic Models. In *ISSRE*, pages 51–60. IEEE Computer Society, 2010. doi:10.1109/ISSRE.2010.47. (cited on page 41).

[FT09] Marc Förster and Mario Trapp. Fault Tree Analysis of Software-Controlled Component Systems Based on Second-Order Probabilities. In *ISSRE*, pages 146–154. IEEE Computer Society, 2009. doi:10.1109/ISSRE.2009.22. (cited on page 44).

[GH11] Javier Gonzalez-Huerta. Integration of Quality Attributes in Software Product Line Development: Master en Ingenieria del Software Metodos Formales y Sistemas de Informacion. Master's thesis, UPV, Spain, 2011. (cited on pages 15, 55 and 89).

[GLHT10] Michael Glaß, Martin Lukasiewycz, Christian Haubelt, and Jürgen Teich. Lifetime Reliability Optimization for Embedded Systems: A System-Level Approach. In *Proceedings of RASDAT '10*, pages 17–22, 2010. URL: https://www12.informatik.uni-erlangen.de/publications/pub2010/GLHT10.pdf. (cited on page 20).

[Gooa] Google Inc. Google App Engine: Platform as a Service (PaaS) cloud computing platform for developing and hosting web applications in Google-managed data centers. URL: https://developers.google.com/appengine/. (cited on page 47).

[Goob] Google Inc. Google Chart: rich gallery of interactive charts and data tools which are powerful, simple to use. URL: https://developers.google.com/chart/. (cited on page 47).

[HBS73] Carl Hewitt, Peter Bishop, and Richard Steiger. A universal modular actor formalism for artificial intelligence. In Nils J. Nilsson, editor, *IJCAI*, pages 235–245. William Kaufmann, 1973. URL: http://dl.acm.org/citation.cfm?id=1624775.1624804. (cited on pages 135 and 138).

[HC01] George T. Heineman and William T. Councill, editors. *Component-based Software Engineering: Putting the Pieces Together*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2001. URL: http://dl.acm.org/citation.cfm?id=379381. (cited on page 13).

[HE13] Iris Hupkens and Michael Emmerich. Logarithmic-time updates in sms-emoa and hypervolume-based archiving. In *EVOLVE-A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation IV*, pages 155–169. Springer International Publishing, 2013. (cited on page 48).

[Hei12] Werner Heijstek. *Architecture Design in Global and Model-Centric Software Development*. PhD thesis, Leiden Institute of Advanced Computer Science (LIACS), Faculty of Science, Leiden University, 2012. URL: https://openaccess.leidenuniv.nl/handle/1887/20225. (cited on page 11).

[HF07] John J. Hudak and Peter H. Feiler. Developing AADL Models for Control Systems: A Practitioner's Guide. Technical Report CMU/SEI-2007-TR-014, Carnegie Mellon University, 2007. URL: http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=8437. (cited on pages 55 and 89).

[HHRV11] Ábel Hegedüs, Ákos Horváth, István Ráth, and Dániel Varró. A model-driven framework for guided design space exploration. In Perry Alexander, Corina S. Pasareanu, and John G. Hosking, editors, *ASE*, pages 173–182. IEEE, 2011. doi:10.1109/ASE.2011.6100051. (cited on page 19).

[HKP10] George T. Heineman, Jan Kofron, and Frantisek Plasil, editors. *Research into Practice - Reality and Gaps, 6th International Conference on the Quality of Software Architectures, QoSA 2010, Prague, Czech Republic, June 23 - 25, 2010. Proceedings*, volume 6093 of *Lecture Notes in Computer Science*. Springer, 2010. doi:10.1007/978-3-642-13821-8. (cited on page 176).

[Inf03] Information Technology for European Advancement (ITEA) Project. Robust Open Component Based Software Architecture for Configurable Devices Project (ROBOCOP), 2003. URL: http://www.hitech-projects.com/euprojects/robocop/. (cited on page 27).

[ISO01] ISO. ISO/IEC 9126-1:2001, Software engineering – Product quality – Part 1: Quality model. Technical report, International Organization for Standardization, 2001. URL: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=22749. (cited on page 14).

[ISO11] ISO/IEC/IEEE. Systems and software engineering – architecture description. *ISO/IEC/IEEE 42010:2011(E) (Revision of ISO/IEC 42010:2007 and IEEE Std 1471-2000)*, pages 1 –46, 1 2011. doi:10.1109/IEEESTD.2011.6129467. (cited on page 11).

[KCH+90] K. Kang, S. Cohen, J. Hess, W. Novak, and S. Peterson. Feature-Oriented Domain Analysis (FODA) Feasibility Study. Technical Report CMU/SEI-90-TR-21, Software Engineering Institute, Carnegie Mellon University, 1990. URL: http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=11231. (cited on page 15).

[KCV02] Nattavut Keerativuttitumrong, Nachol Chaiyaratana, and Vara Varavithya. Multi-objective co-operative co-evolutionary genetic algorithm. In Juan J. Merelo Guervós, Panagiotis Adamidis, Hans-Georg Beyer, José Luis Fernández-Villacañas Martín, and Hans-Paul Schwefel, editors, *PPSN*, volume 2439 of *Lecture Notes in Computer Science*, pages 288–297. Springer, 2002. doi:10.1007/3-540-45712-7_28. (cited on page 149).

[KKR11] Anne Koziolek, Heiko Koziolek, and Ralf Reussner. Peropteryx: automated application of tactics in multi-objective software architecture optimization. In Ivica Crnkovic, Judith A. Stafford, Dorina C. Petriu, Jens Happe, and Paola Inverardi, editors, *QoSA/ISARCS*, pages 33–42. ACM, 2011. doi:10.1145/2000259.2000267. (cited on page 21).

[KM06] Ronny Kolb and Dirk Muthig. Architecture-centric quality engineering form software product lines. In *SPLC*, page 226. IEEE Computer Society, 2006. URL: http://dl.acm.org/citation.cfm?id=1158337.1158710. (cited on page 25).

[Koz11] Anne Koziolek. *Automated Improvement of Software Architecture Models for Performance and Other Quality Attributes.* PhD thesis, Karlsruhe Institute of Technology, 2011. http://d-nb.info/1017321884. URL: http://digbib.ubka.uni-karlsruhe.de/volltexte/1000024955. (cited on pages 13 and 14).

[Kru95] Philippe Kruchten. The 4+1 view model of architecture. *IEEE Software*, 12(6):42–50, 1995. doi:10.1109/52.469759. (cited on page 11).

[Kru12] Johannes W. Kruisselbrink. *Evolution Strategies for Robust Optimization*. PhD thesis, Leiden Institute of Advanced Computer Science (LIACS), Faculty of Science, Leiden University, 2012. URL: https://openaccess.leidenuniv.nl/handle/1887/18931. (cited on pages 8 and 9).

[LCL10] Rui Li, Michel R. V. Chaudron, and René C. Ladan. Towards automated software architectures design using model transformations and evolutionary algorithms. In Martin Pelikan and Jürgen Branke, editors, *GECCO (Companion)*, pages 2097–2098. ACM, 2010. doi:10.1145/1830761.1830880. (cited on pages 17 and 27).

[LGHT08] Martin Lukasiewycz, Michael Glaß, Christian Haubelt, and Jürgen Teich. Efficient symbolic multi-objective design space exploration. In *Proceedings of the 13th Asia South Pacific Design Automation Conference, ASP-DAC 2008, Seoul, Korea, January 21-24, 2008*, pages 691–696. IEEE, 2008. doi:10.1109/ASPDAC.2008.4484040. (cited on page 20).

[LGRT11] Martin Lukasiewycz, Michael Glaß, Felix Reimann, and Jürgen Teich. Opt4J: a modular framework for meta-heuristic optimization. In Natalio Krasnogor and Pier Luca Lanzi, editors, *GECCO*, pages 1723–1730. ACM, 2011. doi:10.1145/2001576.2001808. (cited on page 33).

[LH12] Kenneth Lind and Rogardt Heldal. Automotive system development using reference architectures. In Jonathan P. Bowen, Huibiao Zhu, and Mike Hinchey, editors, *SEW*, pages 42–51. IEEE Computer Society, 2012. doi:10.1109/SEW.2012.11. (cited on page 82).

[LSBB03] G. Latif-Shabgahi, S. Bennett, and J.M. Bass. Smoothing voter: a novel voting algorithm for handling multiple errors in fault-tolerant control systems. *Microprocessors and Microsystems*, 27(7):303–313, 2003. URL: http://www.sciencedirect.com/science/article/pii/S0141933103000401, doi:10.1016/S0141-9331(03)00040-1. (cited on pages 88 and 102).

[MAK+10] Anne Martens, Danilo Ardagna, Heiko Koziolek, Raffaela Mirandola, and Ralf Reussner. A Hybrid Approach for Multi-attribute QoS Optimisation in Component Based Software Systems. In Heineman et al. [HKP10], pages 84–101. doi:10.1007/978-3-642-13821-8_8. (cited on page 21).

[MBAG10] Indika Meedeniya, Barbora Buhnova, Aldeida Aleti, and Lars Grunske. Architecture-Driven Reliability and Energy Optimization for Complex Embedded Systems. In Heineman et al. [HKP10], pages 52–67. doi:10.1007/978-3-642-13821-8_6. (cited on page 18).

[Mit98] Melanie Mitchell. *An introduction to genetic algorithms*. MIT Press, 1998. URL: http://dl.acm.org/citation.cfm?id=522098. (cited on page 8).

[MKBR10] Anne Martens, Heiko Koziolek, Steffen Becker, and Ralf Reussner. Automatically improve software architecture models for performance, reliability, and cost using evolutionary algorithms. In Alan Adamson, Andre B. Bondi, Carlos Juiz, and Mark S. Squillante, editors, *WOSP/SIPEW*, pages 105–116. ACM, 2010. doi:10.1145/1712605.1712624. (cited on pages 2, 17, 52, 53 and 86).

[NHAH04] Masanori Natsui, Naofumi Homma, Takafumi Aoki, and Tatsuo Higuchi. Topology-oriented design of analog circuits based on evolutionary graph generation. In Xin Yao and et al., editors, *PPSN*, volume 3242 of *Lecture Notes in Computer Science*, pages 342–351. Springer, 2004. doi:10.1007/978-3-540-30217-9_35. (cited on page 100).

[OMGa] OMG (Object Management Group). UML Superstructure Specification, v2.0. URL: http://www.omg.org/cgi-bin/doc?formal/05-07-04. (cited on page 13).

[OMGb] OMG (Object Management Group). UML/MARTE (UML Profile for MARTE: Modeling and Analysis of Real-Time Embedded Systems). URL: http://www.omg.org/spec/MARTE/1.1/. (cited on page 27).

[PB99] Vreda Pieterse and Paul E. Black. *Algorithms and Theory of Computation Handbook*. CRC Press LLC, "Levenshtein distance", in Dictionary of Algorithms and Data Structures [online], 1999. URL: http://www.nist.gov/dads/HTML/Levenshtein.html. (cited on page 119).

[PEP06] Andy D. Pimentel, Cagkan Erbas, and Simon Polstra. A systematic approach to exploring embedded system architectures at multiple abstraction levels. *IEEE Trans. Computers*, 55(2):99–112, 2006. doi:10.1109/TC.2006.16. (cited on page 21).

[PHS+04] Hector Posadas, Fernando Herrera, Pablo Sánchez, Eugenio Villar, and Francisco Blasco. System-level performance analysis in systemc. In *2004 Design, Automation and Test in Europe Conference and Exposition (DATE 2004), 16-20 February 2004, Paris, France*, pages 378–383. IEEE Computer Society, 2004. doi:10.1109/DATE.2004.1268876. (cited on pages 63 and 81).

[Pim08] Andy D. Pimentel. The Artemis workbench for system-level performance evaluation of embedded systems. *International Journal of Embedded Systems*, 3(3):181–196, 2008. doi:10.1504/IJES.2008.020299. (cited on page 19).

[PSZ05] Gianluca Palermo, Cristina Silvano, and Vittorio Zaccaria. Multi-objective design space exploration of embedded systems. *J. Embedded Computing*, 1(3):305–316, 2005. URL: http://iospress.metapress.com/content/1yd2pya4avy0799x/. (cited on page 18).

[RHRR12] Per Runeson, Martin Höst, Austen Rainer, and Björn Regnell. *Case Study Research in Software Engineering - Guidelines and Examples*. Wiley, 2012. URL: http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1118104358.html. (cited on pages 81 and 131).

[RKM11] Outi Räihä, Kai Koskimies, and Erkki Mäkinen. Multi-objective genetic synthesis of software architecture. In Natalio Krasnogor and Pier Luca Lanzi, editors, *GECCO (Companion)*, pages 249–250. ACM, 2011. doi:10.1145/2001858.2001998. (cited on page 20).

[Rot06] Octavian Paul Rotaru. Caching patterns and implementation. *Leonardo Journal of Sciences*, 8(8):61–76, January-June 2006. URL: http://ljs.academicdirect.org/A08/61_76.htm. (cited on page 100).

[SELC12] Oliver Schütze, Xavier Esquivel, Adriana Lara, and Carlos A. Coello Coello. Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization. *IEEE Trans. Evolutionary Computation*, 16(4):504–522, 2012. doi:10.1109/TEVC.2011.2161872. (cited on page 109).

[SIMA13] Abdel Salam Sayyad, Joseph Ingram, Tim Menzies, and Hany Ammar. Scalable product line configuration: A straw to break the camel's back. In *ASE*, pages 465–474. IEEE, 2013. doi:10.1109/ASE.2013.6693104. (cited on page 24).

[SMA13] Abdel Salam Sayyad, Tim Menzies, and Hany Ammar. On the value of user preferences in search-based software engineering: a case study in software product lines. In David Notkin, Betty H. C. Cheng, and Klaus Pohl, editors, *35th International Conference on Software Engineering, ICSE '13, San Francisco, CA, USA, May 18-26, 2013*, pages 492–501. IEEE / ACM, 2013. URL: http://dl.acm.org/citation.cfm?id=2486853. (cited on page 24).

[SPG10] Christopher L. Simons, Ian C. Parmee, and Rhys Gwynllyw. Interactive, evolutionary search in upstream object-oriented class design. *IEEE Trans. Software Eng.*, 36(6):798–816, 2010. doi:10.1109/TSE.2010.34. (cited on page 22).

[STT+08] Guido Sand, Jochen Till, Thomas Tometzki, Maren Urselmann, Sebastian Engell, and Michael Emmerich. Engineered versus standard evolutionary algorithms: A case study in batch scheduling with recourse. *Computers & Chemical Engineering*, 32(11):2706–2722, 2008. doi:10.1016/j.compchemeng.2007.09.006. (cited on page 100).

[Sun] Sun Microsystems (merged with Oracle Corporation). Java: a set of several computer software products and specifications from Sun Microsystems (which has since merged with Oracle Corporation), v6. URL: http://java.sun.com/. (cited on pages 43 and 47).

[Szy98] Clemens A. Szyperski. *Component software - beyond object-oriented programming*. Addison-Wesley-Longman, 1998. URL: http://dl.acm.org/citation.cfm?id=515228. (cited on page 12).

[Thea] The Apache™ Hadoop® project. http://hadoop.apache.org/. URL: http://hadoop.apache.org/. (cited on page 136).

[Theb] The Distributed ASCI Supercomputer 4. http://www.cs.vu.nl/das4/. URL: http://www.cs.vu.nl/das4/. (cited on page 139).

[TK11] Catia Trubiani and Anne Koziolek. Detection and solution of software performance antipatterns in palladio architectural models. In Samuel Kounev, Vittorio Cortellessa, Raffaela Mirandola, and David J. Lilja, editors, *ICPE*, pages 19–30. ACM, 2011. doi:10.1145/1958746.1958755. (cited on pages 22 and 99).

[TMD10] Richard N. Taylor, Nenad Medvidovic, and Eric M. Dashofy. *Software Architecture - Foundations, Theory, and Practice*. Wiley, 2010. URL: http://eu.wiley.com/WileyCDA/WileyTitle/productCd-EHEP000180.html. (cited on pages 10, 11 and 13).

[Tru11] Catia Trubiani. *Automated generation of architectural feedback from software performance analysis results*. PhD thesis, Universita di L'Aquila, 2011. URL: www.di.univaq.it/catia.trubiani/phDthesis/PhDThesis-CatiaTrubiani.pdf. (cited on page 103).

[vdLSR07] Frank van der Linden, Klaus Schmid, and Eelco Rommes. *Software product lines in action - the best industrial practice in product line engineering*. Springer, 2007. doi:10.1007/978-3-540-71437-8. (cited on pages 2 and 14).

[WTVL06] Ernesto Wandeler, Lothar Thiele, Marcel Verhoef, and Paul Lieverse. System architecture evaluation using modular performance analysis: a case study. *International Journal on Software Tools for Technology Transfer (STTT)*, 8(6):649–667, 2006. doi:10.1007/s10009-006-0019-5. (cited on page 41).

[WW04] Xiuping Wu and C. Murray Woodside. Performance modeling from software components. In Jozo J. Dujmovic, Virgílio A. F. Almeida, and Doug Lea, editors, *WOSP*, pages 290–301. ACM, 2004. doi:10.1145/974044.974089. (cited on page 52).

[YCAM+11] Chantal Ykman-Couvreur, Prabhat Avasare, Giovanni Mariani, Gianluca Palermo, Cristina Silvano, and Vittorio Zaccaria. Linking run-time resource management of embedded multi-core platforms with automated design-time exploration. *IET Computers & Digital Techniques*, 5(2):123–135, 2011. doi:10.1049/iet-cdt.2010.0030. (cited on page 18).

[YD10] Adel Younis and Zuomin Dong. Trends, features, and tests of common and recently introduced global optimization methods. *Engineering Optimization*, 42(8):691–718, 2010. doi:10.1080/03052150903386674. (cited on pages 31 and 33).

[ZLT02] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. Spea2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In *Evolutionary Methods for Design, Optimisation, and Control*, pages 95–100. CIMNE, Barcelona, Spain, 2002. URL: http://www.tik.ee.ethz.ch/file/97b63e1661aedd98422e486b7c888ad7/ZLT2002a.pdf. (cited on pages 33 and 52).

[ZT98] Eckart Zitzler and Lothar Thiele. Multiobjective optimization using evolutionary algorithms - a comparative case study. In A. E. Eiben, Thomas Bäck, Marc Schoenauer, and Hans-Paul Schwefel, editors, *PPSN*, volume 1498 of *Lecture Notes in Computer Science*, pages 292–304. Springer, 1998. doi:10.1007/BFb0056872. (cited on page 10).