3 Architectural Tactics for Cyber-Foraging

3.1 Introduction

3.2 Functional Architectural Tactics for Cyber-Foraging

3.2.1 Computation Offload

3.2.2 Data Staging

3.2.2.1 Pre-Fetching

3.2.2.2 In-Bound Pre-Processing

3.2.2.3 Out-Bound Pre-Processing

3.2.3 Surrogate Provisioning

3.2.3.1 Pre-Provisioned Surrogate

3.2.3.2 Surrogate Provisioning from the Mobile Device

3.2.3.3 Surrogate Provisioning from the Cloud

3.2.4 Surrogate Discovery

3.2.4.1 Local Surrogate Directory

3.2.4.2 Cloud Surrogate Directory

3.2.4.3 Surrogate Broadcast

3.3 Non-Functional Architectural Tactics for Cyber-Foraging

3.3.1 Resource Optimization

3.3.1.1 Runtime Partitioning

3.3.1.2 Runtime Profiling

3.3.1.3 Resource-Adapted Computation

3.3.2 Fault Tolerance

3.3.2.1 Local Fallback

3.3.2.2 Opportunistic Mobile-Surrogate Data Synchronization

3.3.2.3 Cached Results

3.3.2.4 Alternate Communications

3.3.2.5 Eager Migration

3.3.3 Scalability/Elasticity

3.3.3.1 Just-in-Time Containers

3.3.3.2 Right-Sized Containers

3.3.3.3 Surrogate Load Balancing

3.3.4 Security

3.3.4.1 Trusted Surrogates

3.4 Summary and Conclusions
### 5.3.4 Mapping of Architectural Design Decisions to Architectural Tactics

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.3.4.1 Out-Bound Pre-Processing</td>
<td>160</td>
</tr>
<tr>
<td>5.3.4.2 Pre-Provisioned Surrogate</td>
<td>160</td>
</tr>
<tr>
<td>5.3.4.3 Local Surrogate Directory</td>
<td>163</td>
</tr>
<tr>
<td>5.3.4.4 Client-Side Data Caching</td>
<td>165</td>
</tr>
</tbody>
</table>

### 5.3.5 Analysis

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.3.5.1 Mapping between Tactics and Requirements</td>
<td>167</td>
</tr>
<tr>
<td>5.3.5.2 Discussion of Tactics for System Enhancements</td>
<td>172</td>
</tr>
<tr>
<td>5.3.5.3 Findings</td>
<td>173</td>
</tr>
</tbody>
</table>

### 5.3.6 Threats to Validity

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.4 Conclusions</td>
<td>175</td>
</tr>
<tr>
<td>5.5 Acknowledgments</td>
<td>176</td>
</tr>
</tbody>
</table>

### 6 Case Study 3: AgroTempus — Using Architectural Tactics for Cyber-Foraging Systems Development

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Introduction</td>
<td>177</td>
</tr>
<tr>
<td>6.2 Case Study Design</td>
<td>178</td>
</tr>
<tr>
<td>6.2.1 Research Questions</td>
<td>178</td>
</tr>
<tr>
<td>6.2.2 Data Collection Procedure</td>
<td>178</td>
</tr>
<tr>
<td>6.2.3 Analysis Procedure</td>
<td>179</td>
</tr>
<tr>
<td>6.3 Results</td>
<td>180</td>
</tr>
<tr>
<td>6.3.1 System Context</td>
<td>180</td>
</tr>
<tr>
<td>6.3.2 System Requirements</td>
<td>181</td>
</tr>
<tr>
<td>6.3.2.1 Functional Requirements</td>
<td>181</td>
</tr>
<tr>
<td>6.3.2.2 Non-Functional Requirements</td>
<td>182</td>
</tr>
<tr>
<td>6.3.2.3 Constraints and Assumptions</td>
<td>185</td>
</tr>
<tr>
<td>6.3.3 Mapping of System Requirements to Architectural Tactics</td>
<td>185</td>
</tr>
<tr>
<td>6.3.3.1 Computation Offload</td>
<td>186</td>
</tr>
<tr>
<td>6.3.3.2 Out-Bound Pre-Processing</td>
<td>187</td>
</tr>
<tr>
<td>6.3.3.3 Pre-Fetching</td>
<td>187</td>
</tr>
<tr>
<td>6.3.3.4 Pre-Provisioned Surrogate</td>
<td>187</td>
</tr>
<tr>
<td>6.3.3.5 Surrogate Broadcast</td>
<td>188</td>
</tr>
<tr>
<td>6.3.3.6 Cached Results</td>
<td>188</td>
</tr>
<tr>
<td>6.3.3.7 Client-Side Data Caching</td>
<td>188</td>
</tr>
<tr>
<td>6.3.3.8 Just-in-Time Containers</td>
<td>189</td>
</tr>
<tr>
<td>6.3.4 System Architecture and Design</td>
<td>189</td>
</tr>
<tr>
<td>6.3.5 Mapping of Architectural Components to System Requirements</td>
<td>191</td>
</tr>
</tbody>
</table>
6.3.6 Mapping of Architectural Components to Identified Architectural Tactics

6.3.6.1 Computation Offload
6.3.6.2 Out-Bound Pre-Processing
6.3.6.3 Pre-Fetching
6.3.6.4 Pre-Provisioned Surrogate
6.3.6.5 Surrogate Broadcast
6.3.6.6 Cached Results
6.3.6.7 Client-Side Data Caching
6.3.6.8 Just-in-Time Containers

6.3.7 System Implementation

6.3.8 Analysis

6.3.8.1 System Evaluation
6.3.8.2 Developer Observation and Feedback
6.3.8.3 Findings

6.3.9 Threats to Validity

6.4 Conclusions

6.5 Acknowledgments

7 Characterization of Cyber-Foraging Usage Contexts

7.1 Introduction

7.2 Analysis

7.3 Cyber-Foraging Usage Contexts

7.4 Computation Offload Usage Contexts

7.4.1 Usage Context 1: Computation-Intensive Mobile Applications (Short Operations)

7.4.2 Dynamic Environments

7.4.2.1 Usage Context 2: Mobile Applications in Low Coverage Environments

7.4.2.2 Usage Context 3: Computation-Intensive Mobile applications (Long Operations)

7.4.2.3 Usage Context 4: Computation-Intensive Mobile Applications in Hostile Environments

7.4.2.4 Usage Context 5: Public Surrogates

7.5 Data Staging Usage Contexts

7.5.1 Usage Context 6: Sensing Applications

7.5.2 Usage Context 7: Data-Intensive Mobile Applications

7.6 Summary and Conclusions
8 Decision Model for Cyber-Foraging Systems

8.1 Introduction ................................. 243
8.2 Mapping the Problem Space to the Solution Space ................................. 244
8.3 How to Use the Decision Models ................................................. 246
8.4 Decision Models for Cyber-Foraging Systems ................................. 248
  8.4.1 Data Staging .......................................... 248
  8.4.2 Surrogate Provisioning ........................................ 251
  8.4.3 Surrogate Discovery ..................................... 253
  8.4.4 Resource Optimization ..................................... 256
  8.4.5 Fault Tolerance ........................................ 258
  8.4.6 Scalability and Elasticity ................................... 262
  8.4.7 Security .................................................. 264
    8.4.7.1 Credential Exchange .................................. 264
    8.4.7.2 Credential Validation ................................. 266
8.5 Validation ................................................. 266
8.6 Related Work ............................................... 268
8.7 Conclusions ................................................. 269

9 Conclusions ................................................. 271

  9.1 Contributions ............................................... 271
    9.1.1 RQ1: What Software Architecture Design Decisions for Cyber-Foraging Systems can be Identified in the Literature? ........................................... 272
    9.1.2 RQ2: What Architectural Tactics can be Derived from the Identified Architectural Design Decisions? ........................................... 273
    9.1.3 RQ3: What are the Usage Domains and Contexts (Defined in Terms of Functional and Non-Functional Requirements) that Benefit from Cyber-Foraging? ........................................... 274

  9.2 Future Research ................................................. 275
    9.2.1 Extension of the Tactics Catalog .................................. 275
    9.2.2 Quantitative Analysis of the Impact of Tactics Selection ........................................... 276
    9.2.3 Tools for the Development and Analysis of Cyber-Foraging Systems ........................................... 277
    9.2.4 Architecture Patterns for Cyber-Foraging Systems ........................................... 278