ABSTRACT

Mobile sensing denotes the use of mobile devices and their integrated sensors for sensing and learning physical and social phenomena, and to use derived information for sharing, informing, and persuading humans. From the perspective of software architecture, mobile sensing bears several design challenges regarding, e.g., use of battery powered mobile devices, and collection and processing of sensor data. In this paper, we present tactics to address these architecture design challenges. We discuss the two architectural qualities energy efficiency and resource adaptability, and describe them using general scenario-generation tables to support the systematic specification of architecture requirements. Furthermore, we develop a catalog of architectural tactics distilled from literature to enable developers to systematically apply proven methods. For each tactic, we provide examples to relate the respective tactics to particular cases illustrating their use in practice. Our preliminary findings show that the tactics are beneficial to provide a guideline and to create awareness of the special challenges of energy-efficient and resource-adaptable architecture design.

Categories and Subject Descriptors

D.2.11 [Software Engineering]: Software Architectures

General Terms

Design, Experimentation

Keywords

software architecture; mobile sensing; architecture tactics

1. INTRODUCTION

Mobile sensing denotes the use of mobile devices and their embedded sensors for sensing and learning physical and social phenomena, and using derived information to inform, share, and persuade humans [21]. Due to the increasing penetration of mobile devices and easy application distribution, mobile sensing systems can potentially reach very large user populations. Thereby mobile sensing systems can have a major impact on human behavior and our means for making informed decisions. For instance, mobile sensing enables new tools for gathering temporal-spatial data about humans to inform building facility planning [42], and for monitoring human crowds to improve safety and logistics at large events [18]. From the viewpoint of software architecture, mobile sensing brings attention to a number of design challenges, e.g., device mobility, battery limitations, and continuous collection and processing of sensor data.

Problem Statement. To address the general challenges, and energy efficiency and resource adaptability in particular, several contributions can be found in literature reporting lessons learned regarding the design and development of individual mobile sensing solutions, e.g., CreekWatch [12], IntraTime [37], CenceMe [29], and EV transportation mapping [53]. Furthermore, complex middleware systems can be found, e.g., EnTracked [19] and METIS [39], to support architects and developers addressing the aforementioned challenges. However, so far, the proposed solutions do not sufficiently provide lessons learned in a structured manner, which allows software architects to precisely specify architecture quality attributes, and to reuse the gathered knowledge.

Objective. We aim to improve the understanding of the challenges of mobile sensing systems and how these challenges affect software architecture. Apart from (standard) requirements regarding architecture quality attributes, e.g., performance, interoperability, availability, security, usability, and testability [4], we discuss the special challenges for mobile sensing systems. In particular, we focus on those challenges related to the two architecture quality attributes energy efficiency and resource adaptability. We consider energy efficiency the ‘amount’ of energy required to provide and/or deliver a (mobile) service at a given quality of service (QoS). Resource adaptability is the ability of services to flexibly adapt to changing environments given by fluctuating availability of sensing and communication resources with different QoS levels, context-dependent quality of sensing data, and fluctuating wireless communication options and network availability/accessibility.

Contribution. In this paper, we apply the concepts proposed by Bass et al. [4] to capture architectural knowledge for mobile sensing systems in the form of general scenario-generation tables for specifying requirements (1) as quality...

© ACM, 2015. This is the author’s version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution. The definitive version was published in

QoSA’15, May 4–8, 2015, Montréal, QC, Canada.

Copyright © 2015 ACM 978-1-4503-3470-9/15/05 ...$15.00.

http://dx.doi.org/10.1145/2737182.2737196.
attribute scenarios and (2) as tactics describing solutions for how to control a particular quality attribute. Using general scenario-generation tables, we describe the two quality attributes energy efficiency and resource adaptability in detail to aid architects and developers to precisely specify architecture requirements for mobile sensing systems. Based on literature, we provide an initial catalog of proven architecture tactics complemented with rationale and examples. This catalog aims to help architects and developers to systematically apply proven approaches to ease design and implementation tasks. The proposed catalog was initially validated in a case study with student teams. So far, the initial results indicate that the tactics are beneficial for guiding software design and development and, moreover, help to build awareness of the challenges of energy-efficient and resource-adaptable architecture design.

Outline. The remainder of the paper is structured as follows: In Sect. 2, we introduce the two quality attributes energy efficiency and resource adaptability, provide a detailed description of the attributes, and present the general scenario-generation tables. Section 3 introduces tactics for mobile sensing, and provides the developed catalog (including the context-specific related work). In Sect. 4, we present the initial validation of the tactics and discuss the outcomes, before concluding the paper in Sect. 5.

2. QUALITY ATTRIBUTES

In this section, we introduce the two quality attributes energy efficiency and resource adaptability, we provide a general scenario-generation table each to support requirement specification, and we relate them to other qualities.

2.1 Energy Efficiency

With the quality attribute energy efficiency, we denote the ‘amount’ of energy required to provide and/or deliver a (mobile) service at a given QoS. The energy consumption of a mobile service is mainly impacted by heavy computation loads, and the use of high-consuming sensors, screens, and communication options. Furthermore, energy consumption is amplified in scenarios in which continuous use of computation, sensing, and communication resources is required, e.g., transportation mode detection from inertial sensors, place detection from position traces, and information sharing of sensor data among different devices.

Services are requested from an unknown number of clients (e.g., mobile devices, data servers, and web-based dashboards). The QoS highly depends on the accuracy and timeliness of the information provided by a service. Response measures for energy efficiency focus on the energy spent for serving service requests in relation to the QoS. An important consideration when describing scenarios for energy efficiency is the number and capacity of available resources on a device, since a system might take different actions based on available energy. Table 1 presents the general scenario generation table for energy efficiency.

2.2 Resource Adaptability

With the quality attribute resource adaptability, we denote the ability of services to flexibly adapt to changing environments given by, e.g., fluctuating availability of sensing and communication resources with different QoS levels, fluctuating communication options and network availability, and context-dependent quality of sensing data.

Resources include local as well as remote sensing options, e.g., accelerometers, gyroscopes, barometers, temperature sensors, humidity sensors, GPS chips, WiFi chips, microphones, light sensors and cameras, and communication options (e.g., Cellular (2G, 3G, 4G), Bluetooth, WiFi, Zigbee, and NFC), which might appear, disappear, or change QoS. This can happen in an environment with low/high quality of resources or few/many resources. Response measures capture any degradation of service levels in both the spatial and temporal dimensions. Table 2 presents the general scenario generation table for resource adaptability.

2.3 Further Qualities for Mobile Sensing

Beyond the aforementioned two quality attributes, further quality attributes contribute to high-quality architectures for mobile sensing systems, e.g., modifiability, availability,
testing, and security. In the following, we list some exemplary contributions addressing these quality attributes. For modifiability, [6] describes an easy-to-customize middleware for mobile sensing. In [10], availability for distributed processing of sensor data is addressed. For testing, [35] presents debugging tools for energy-related bugs, and [25] describes a testing framework, which allows for automatically replicating heterogeneous sensor data. Finally, [34] addresses security by means for privacy protection. However, further detailing scenarios and tactics for these attributes in the context of mobile sensing is subject to future work.

3. TACTICS FOR MOBILE SENSING

In this section, we introduce the tactics for the two considered quality attributes energy efficiency (Sect. 3.1) and resource adaptability (Sect. 3.2). To systematize the presentation, we first give an overview by providing a categorization, before providing detailed descriptions of the tactics as tables for the respective tactics including rationale and complementing examples from related work.

3.1 Energy Efficiency

Energy efficiency tactics address how to handle service requests in order to improve energy efficiency. Figure 1 gives an overview of the proposed tactics.

![Tactics for energy efficiency](image)

Figure 1: Tactics for energy efficiency.

Figure 1 also presents the chosen categorization of the considered tactics: sensing deals with scheduling of sensors, processing addresses the processing of sensor data, sharing addresses the distribution of data and control, and energy awareness deals with considerations regarding the balance of quality and energy-efficiency in general.

However, the tactics are non-exclusive and may also affect each other, and different tactics can be embedded into complex distributed scenarios that require the combination of different tactics. For instance, sensing and processing may happen on different computational entities and each step can involve several entities at the same time, which are coordinated by a sharing mechanism. Furthermore, for mobile sensing, QoS highly depends on the quality and quantity of sensor data, and the sophistication of processing. Additionally, energy consumption of sensors and communication options, which may significantly deviate on different platforms [13], influence the tactics’ application. For example, a simple approach to improve energy efficiency might be to just reduce QoS. However, this “horse-trading” approach is not straightforward and, hence, energy efficiency needs always to be considered in relation to the requested QoS, e.g., required accuracy of the information provided by a service. That is, the different tactics need to be balanced in order to provide an energy-efficiency profile appropriate for a particular context. The quantification of energy efficiency requires monitoring using internal and/or external tools [17, 7] in different stages of the product life cycle.

To support the systematic design of mobile sensing systems, we provide 12 tactics in 4 categories.

3.1.1 Category: Sensing

An important step to improve energy efficiency is to consider how sensing is implemented by sampling sensors to collect sensor data and to serve service requests. To improve the energy efficiency in the context of sensing, Table 3 presents the following five tactics complemented with examples: demand driven, static duty cycling, dynamic duty cycling, sensor selection, and sensor replacement.

3.1.2 Category: Energy Awareness

As batteries discharge, the energy available at a mobile device changes over time. Users might value if the “last drop” of energy of their device is spent as valuable as possible. The urgency of energy usage at near battery depletion might also depend on the forecasting of recharge options [41]. To improve energy efficiency using energy awareness, Table 4 presents the quality for efficiency tactic and gives examples.

3.1.3 Category: Processing

Services often involve the processing of sensor data to extract information or to relate sensor data to historical data, user data, social networking data, or physical models. For some kinds of data, e.g., sound, images, and video processing data, mobile devices require more energy for processing than for sampling plain sensor data, e.g., geographic location. To improve the energy efficiency of processing sensor data, Table 5 presents the following two tactics with examples: demand-driven and event-based.

3.1.4 Category: Sharing

A computational entity requesting a service is often not the only consumer. Furthermore, multiple entities can be involved in serving requests for other consumers. Therefore, energy efficiency can be improved by optimizing how data is shared among entities when serving a request and delivering responses to requesters. To improve the energy effi-
Table 3: Mobile sensing tactics for energy efficiency, category: sensing

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMD Demand Driven</td>
<td>This tactic aims to improve sensor scheduling by providing sensor data on demand/request which is efficient if request for sensing is rare. A variation is aggregating close-in-time requests to avoid multiple sensor-scheduling within a short period of time. <strong>Examples:</strong> Kim et al. [12] propose the participatory sensing system CreekWatch to schedule sensors on demand for sporadic user-provided reports, and Zhuang et al. [55] optimize demand-driven requests for positioning serving close-in-time requests by piggybacking on previous requests.</td>
<td></td>
</tr>
<tr>
<td>SDC Static Duty Cycling</td>
<td>If requests arrive with high frequency or if serving requests require continuous sensing over a period of time, improvements can be achieved by optimizing sensor duty cycling. This tactic aims to select a static duty cycling interval minimizing the duty cycle, yet, still providing the minimal required temporal and spatial resolution of sensor data. The tactic assumes that the relationship between QoS and minimal temporal and spatial resolution is known or predictable. <strong>Examples:</strong> Sun et al. [49] propose CenceMe—a sensor-based social networking system that apply SDC for sampling all types of sensors.</td>
<td></td>
</tr>
<tr>
<td>DDC Dynamic Duty Cycling</td>
<td>The DDC tactic can be improved by dynamically controlling sensor duty cycling, i.e., by continuously calculating the duty cycling interval based on the quality of sensor data required to serve requests, e.g., if low quality is suitable the interval is increased, and for high quality intervals shortened. For instance, DDC can be adapted to patterns of human behavior, i.e., short intervals during daytime, and long ones at night. This tactic requires a model for relating service quality requirements to sampling frequency. However, implementing dynamic schemes is more complex than implementing simpler ones (DMD, SDC). <strong>Examples:</strong> LiKamWa et al. [26] propose methods to apply DDC to CMOS sensors for image capturing by controlling stand-by time and clock scaling. Rachuri et al. [40] present SociableSense—a social sensing system applying DDC to acceleration sensors, which continuously estimates the probability of observing events. Kjærgaard et al. [19] present EvTracked, which implements DDC for positioning based on estimates of movement speed. Lu et al. [28] present the jigsaw engine utilizing DDC to judiciously triggering power-hungry processing respecting mobility and behavioral patterns of users to reduce energy costs.</td>
<td></td>
</tr>
<tr>
<td>SES Sensor Selection</td>
<td>Selecting the least energy-consuming sensor from all available resources, or balancing sensor data quality with energy cost, is another approach toward energy-efficiency. This tactic aims to compare sensor options to fulfill a request and use the least costly sensor to deliver data with the needed QoS. <strong>Examples:</strong> Lin et al. [27] present a-Loc implementing SES among several options for positioning a mobile device. Rachuri et al. [39] present METIS applying SES with mobile and static sensors to capture, inter alia, the social context of person. Lee et al. [23] propose CoMon that allows for SES taking into account options residing on nearby mobile devices. Brouwers et al. [8] present a SES mechanism to configure WiFi scanning based on the predicted sensor data quality to minimize energy consumption by minimizing the scanned channels.</td>
<td></td>
</tr>
<tr>
<td>SER Sensor Replacement</td>
<td>Points in time where a sensor is sampled might correlate with the output of a less energy-consuming sensor. This tactic aims at using the least costly sensor to trigger another (more energy-consuming) sensor to collect data. <strong>Examples:</strong> Wang et al. [51] propose a framework for specifying rules for SER, e.g. a rule to wake-up position sensing when recognizing movement via an accelerometer. Nath [31] proposes ACE using correlations to infer SER relations by quantifying the cost of deriving an attribute. For Headio, Sun et al. [48] apply SER for a camera by steering camera activations by the smartphone orientation sensed using the accelerometer.</td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Mobile sensing tactics for energy efficiency, category: energy awareness

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QFE Quality for Efficiency</td>
<td>Due to urgency of battery depletion, energy efficiency can be further improved by lowering required QoS. This tactic proposes to adaptively trade quality for energy efficiency by limiting the types of requests served and the QoS delivered for each request. <strong>Examples:</strong> Nirjon et al. [32] present Auditor—a platform that, depending on the level of available energy, adaptively selects what features are recognised from audio streams. As part of a framework for position sensing, Zhuang et al. [55] adapt sensing quality parameters based on available energy.</td>
<td></td>
</tr>
</tbody>
</table>

3.2 Resource Adaptability

Resource adaptability tactics address how to make mobile sensing systems responsive to fluctuations in available resources and their QoS to avoid delivery of degraded services. Figure 2 presents the chosen categorization of the considered tactics: resource availability addresses the adaptation to different levels of resource availability, processing deals with resource adaptation in the context of sensor data processing, and quality awareness deals with considerations regarding the balance of quality of the delivered service and available resources in general.

The primary challenge addressed by resource adaptability tactics is to avoid the delivery of degraded services respecting fluctuating availability of sensing and communication resources, and their respective QoS. To evaluate the successful
To support the systematic design of mobile sensing systems, we provide 7 tactics in 3 categories.

3.2.1 Category: Resource Availability

An important step toward improving the resource adaptability is to optimize the use of available resources. To improve resource adaptability using resource availability Table 7 presents the following three tactics with complementing examples: resource selection, resource prediction, and increase resources.

3.2.2 Category: Quality Awareness

If it is impossible to avoid delivering a degraded service, consequences can be circumvented if the requester is made aware of the level of degradation, e.g., limitations regarding the level of degradation, e.g., limitations regarding

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDR</td>
<td>Demand-driven</td>
<td>If processing requests occur only sporadically, the most energy-efficient way to implement processing is on demand. To provide requested information, this tactic aims to schedule processing of sensor data on a per-request base. Furthermore, aggregating processing steps further improves DDR if, i.e., particular processing steps serve several types of requests, and similar requests occur close-in-time. This tactic works in parallel with DMD, and can also use buffered and/or continuously-sensed data. <strong>Examples:</strong> Shin et al. [47] present FindingMiMo utilizing continuous sensing to track objects, yet, only processing data on demand (if users request functionality to find a missing object). Iqbal et al. [11] propose a method for configuration folding to combine repetitive processing steps for sensor data, and prototype the idea for speech detection.</td>
</tr>
<tr>
<td>EVB</td>
<td>Event-based</td>
<td>If sensing requests demand low latency (time span from event recognition to requester notification), energy efficiency can be improved by scheduling processing when a simple threshold on raw sensor data indicate an event. This tactic aims to continuously inspect sensor data, and to only use complex processing if simple thresholds are violated. The tactic assumes that relevant thresholds, which indicate sensor data contain event-related data, can be chosen properly. <strong>Examples:</strong> For Headio, Sun et al. [48] show indoor heading estimation utilizing heavy feature processing of images using simple quality metrics on images, e.g., extreme brightness. Kjærgaard et al. [16] guard processing for trajectory simplification by significant position changes. Lu et al. [28] present the yoga engine adaptively throttling depth and sophistication of sensing pipelines if input data is low-quality or uninformative.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RQC</td>
<td>Requester-centric</td>
<td>If the sender of requests is the main trigger of data sharing actions, energy efficiency can be improved by limiting the request rate of the sender. This tactic proposes to limit the requests from a sender by temporally restricting the issuing of new requests. Denied requests can be served by, e.g., caching previous responses, and aggregating several requests. <strong>Examples:</strong> Baier et al. [2] propose to limit sensing service requests for public sensing systems by only requesting services from those mobile devices that are most likely to return valuable sensor data.</td>
</tr>
<tr>
<td>PVC</td>
<td>Provider-centric</td>
<td>If entities serving requests are the main trigger of data sharing actions (e.g., in continuous sensing scenarios) energy efficiency can be improved by limiting response rates of request-serving entities. This tactic aims to limit the responses from request-serving entities by temporally limiting issuing of new responses. Requests can be scheduled only if significant/new information is available, or responses can be aggregated. <strong>Examples:</strong> Bhattacharya et al. [5] discuss protocols for limiting position updates for continuous sensing. Musolesi et al. [30] discuss optimization of uploading strategies for continuous sensing. Baier et al. [3] show how to decrease communication overhead by opportunistically updating position information when communication channels are opened for other background data transfer.</td>
</tr>
<tr>
<td>RPC</td>
<td>Requester-Provider-centric</td>
<td>For some observations, models are available to predict future quantities based on historical data and, thus, to improve energy-efficiency by limiting data sharing activities. This tactic proposes to use models for requesters and providers to minimize data sharing activities by only sending responses if the model’s quality of predictions violates a quality threshold. The model can be a generically applicable model for predicting quantitative values or a domain-optimized one taking into account known correlations and periodicities of values. <strong>Examples:</strong> Kjærgaard et al. [13] use models based on dead-reckoning to minimize position updates. Rachuri et al. [40] propose methods for distributing mobile sensing processing tasks between computational entities in the most energy-efficient way. Philipp et al. [36] use models of the phenomena under observation to minimize the rate of sensing requests and responses.</td>
</tr>
<tr>
<td>CSL</td>
<td>Communication selection</td>
<td>Since mobile devices move, options for communication change over time and different energy footprints of the communication options allow for opportunities to improve energy efficiency. This tactic aims to improve energy efficiency by selecting the least energy-consuming communication option to exchange data between requesters and providers. <strong>Examples:</strong> Ra et al. [38] propose methods to select the most energy-efficient option to transmit data, and they discuss methods for estimating when to delay data transmission until a less costly option is available.</td>
</tr>
</tbody>
</table>

Table 5: Mobile sensing tactics for energy efficiency, category: processing

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDR</td>
<td>Demand-driven</td>
<td>If processing requests occur only sporadically, the most energy-efficient way to implement processing is on demand. To provide requested information, this tactic aims to schedule processing of sensor data on a per-request base. Furthermore, aggregating processing steps further improves DDR if, i.e., particular processing steps serve several types of requests, and similar requests occur close-in-time. This tactic works in parallel with DMD, and can also use buffered and/or continuously-sensed data. <strong>Examples:</strong> Shin et al. [47] present FindingMiMo utilizing continuous sensing to track objects, yet, only processing data on demand (if users request functionality to find a missing object). Iqbal et al. [11] propose a method for configuration folding to combine repetitive processing steps for sensor data, and prototype the idea for speech detection.</td>
</tr>
<tr>
<td>EVB</td>
<td>Event-based</td>
<td>If sensing requests demand low latency (time span from event recognition to requester notification), energy efficiency can be improved by scheduling processing when a simple threshold on raw sensor data indicate an event. This tactic aims to continuously inspect sensor data, and to only use complex processing if simple thresholds are violated. The tactic assumes that relevant thresholds, which indicate sensor data contain event-related data, can be chosen properly. <strong>Examples:</strong> For Headio, Sun et al. [48] show indoor heading estimation utilizing heavy feature processing of images using simple quality metrics on images, e.g., extreme brightness. Kjærgaard et al. [16] guard processing for trajectory simplification by significant position changes. Lu et al. [28] present the yoga engine adaptively throttling depth and sophistication of sensing pipelines if input data is low-quality or uninformative.</td>
</tr>
</tbody>
</table>

Table 6: Mobile sensing tactics for energy efficiency, category: sharing

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RQC</td>
<td>Requester-centric</td>
<td>If the sender of requests is the main trigger of data sharing actions, energy efficiency can be improved by limiting the request rate of the sender. This tactic proposes to limit the requests from a sender by temporally restricting the issuing of new requests. Denied requests can be served by, e.g., caching previous responses, and aggregating several requests. <strong>Examples:</strong> Baier et al. [2] propose to limit sensing service requests for public sensing systems by only requesting services from those mobile devices that are most likely to return valuable sensor data.</td>
</tr>
<tr>
<td>PVC</td>
<td>Provider-centric</td>
<td>If entities serving requests are the main trigger of data sharing actions (e.g., in continuous sensing scenarios) energy efficiency can be improved by limiting response rates of request-serving entities. This tactic aims to limit the responses from request-serving entities by temporally limiting issuing of new responses. Requests can be scheduled only if significant/new information is available, or responses can be aggregated. <strong>Examples:</strong> Bhattacharya et al. [5] discuss protocols for limiting position updates for continuous sensing. Musolesi et al. [30] discuss optimization of uploading strategies for continuous sensing. Baier et al. [3] show how to decrease communication overhead by opportunistically updating position information when communication channels are opened for other background data transfer.</td>
</tr>
<tr>
<td>RPC</td>
<td>Requester-Provider-centric</td>
<td>For some observations, models are available to predict future quantities based on historical data and, thus, to improve energy-efficiency by limiting data sharing activities. This tactic proposes to use models for requesters and providers to minimize data sharing activities by only sending responses if the model’s quality of predictions violates a quality threshold. The model can be a generically applicable model for predicting quantitative values or a domain-optimized one taking into account known correlations and periodicities of values. <strong>Examples:</strong> Kjærgaard et al. [13] use models based on dead-reckoning to minimize position updates. Rachuri et al. [40] propose methods for distributing mobile sensing processing tasks between computational entities in the most energy-efficient way. Philipp et al. [36] use models of the phenomena under observation to minimize the rate of sensing requests and responses.</td>
</tr>
<tr>
<td>CSL</td>
<td>Communication selection</td>
<td>Since mobile devices move, options for communication change over time and different energy footprints of the communication options allow for opportunities to improve energy efficiency. This tactic aims to improve energy efficiency by selecting the least energy-consuming communication option to exchange data between requesters and providers. <strong>Examples:</strong> Ra et al. [38] propose methods to select the most energy-efficient option to transmit data, and they discuss methods for estimating when to delay data transmission until a less costly option is available.</td>
</tr>
</tbody>
</table>
Table 7: Mobile sensing tactics for resource adaptability, category: resource availability

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSL</td>
<td>Resource Selection</td>
<td>In order to avoid delivering a degraded service, an important element is to select those resources that deliver the highest QoS. This tactic supports selecting resources with the highest QoS for delivering requested services. The perceived QoS of a resource can be based on a-priori calculated statistics for each resource type, or QoS can be estimated at the time of use. Examples: Schuhmann et al. [46] introduce methods for adaptive RSL in heterogeneous environments considering services and resources. Friedman et al. [9] consider RSL for communication—in particular—tradeoffs between using WiFi and Bluetooth communication.</td>
</tr>
<tr>
<td>RPD</td>
<td>Resource Prediction</td>
<td>The prediction of resource availability allows for improving resource adaptability. This tactic recommends using predictions of resource availability to schedule sensing and communication when the right resources with the highest possible QoS are available. Moreover, the tactic minimizes the chance of delivering a degraded service. Examples: Xu et al. [54] propose methods for predicting the QoS and communication resource availability. Kjærgaard et al. [20] present PosQ to map and predict GPS availability in urban areas to, inter alia, predict whether a GPS positioning request will succeed and whether other positioning resources should be utilized to avoid wasting time on a potentially unsuccessful GPS request.</td>
</tr>
<tr>
<td>IRS</td>
<td>Increase Resources</td>
<td>An increase in the amount of available resources makes it easier to avoid delivering a degraded service. This tactic proposes to increase the number of available resources by deploying devices with more resources or enabling better resource utilization among mobile devices. Examples: Sani et al. [45] present Rio for sharing inputs and outputs between mobile systems while increasing the resource availability. Rachuri et al. [39] present METIS utilizing mobile as well as static sensors to increase available resources.</td>
</tr>
</tbody>
</table>

Figure 2: Tactics for resource availability.

Table 8: Mobile sensing tactics for resource adaptability, category: quality awareness

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAD</td>
<td>Application Adaptation</td>
<td>If requesters are aware of degraded QoS, requesters can use this information for resource adaptation. This tactic aims to use estimates of service degradation to adapt application logic and user interfaces to compensate degradation. If a degraded service is available, application logic can limit actions and user interfaces to make users aware of the degradation to, i.e., take correcting actions, and to build awareness of service limitations. Examples: Lemelson et al. [24] propose methods to estimate the service degradation level for indoor positioning, and Langdal et al. [22] present a middleware for handling quality adaptation for degraded services.</td>
</tr>
</tbody>
</table>

4. TACTICS VALIDATION

We analyze the tactics for their feasibility by conducting an explorative case study with students of the Software System Design and Technologies course taught as part of the University of Southern Denmark (SDU) “Master in Software Engineering” program. Based on the structure proposed by Runeson and Höst [44], in the following, we provide a brief overview of the case study, initial results from the internal validation, and a discussion of the feasibility.

4.1 Research Design

In order to validate the tactics, we conducted an explorative case study in the course Software System Design and Technology...
A catalog of tactics and development of mobile applications?

Do the architecture design tactics improve awareness of architectural challenges of mobile sensing?

RQ1: Do the architecture design tactics improve awareness of architectural challenges of mobile sensing? This question aims at investigating the general perception of the design tactics and if the proposed tactics help to create awareness of the associated architectural challenges.

RQ2: Do the architecture design tactics improve the design and development of mobile applications? A catalog of tactics can be considered helpful or limiting. This question aims to study if the students perceived improvements in the design and development tasks or if they experienced, e.g., restrictions, increased complexity, and development overhead.

4.2 Case Description

Before presenting the results, we provide some insights into the case. The students developed different types of mobile sensing smart phone apps including an app for mapping crowdedness levels by recording the level of noise, and an app for collecting location traces of building occupants. All apps were developed in Java on the Android platform. In the following, we exemplarily describe which tactics were applied by the groups in the stage 2 of the study to improve energy efficiency, and which improvements were achieved by applying the tactics.

The group developing the crowdedness level mapping app used the DDC and SES tactics to improve energy efficiency, and the MSV tactic to improve resource adaptability. To utilize the SES tactic, the group used the smart phone’s proximity sensor to guard the sampling and processing of sound. For DDC, they applied spatial-temporal geofencing. Using the internal power-monitoring tool PowerTutor2 [1], they collected data over a 5-minute test run in which they found a drop in power consumption from 79.8 Joule (app from stage 1) to 0.36 Joule (app from stage 2).

The group developing an app for collecting location traces of building occupants, applied the SDC and DDC tactics for energy efficiency utilizing spatial-temporal geofencing. For SER, the group used built-in movement detection capabilities of the smart phones. This group also used PowerTutor2 to collect data over a 1-hour test run, and found a drop in energy consumption from 13.3 Joule (app from stage 1) to less than 1 Joule (app from stage 2).

Based on these results, so far, applying the proposed tactics showed beneficial to improve energy efficiency for mobile sensing apps.

4.3 Case Study Results

In summary, 9 students filled out the questionnaire of which 8 are enrolled in the SDU SE Master’s program, and one student is in the Bachelor’s program. Six of the participants have 2-5 years of programming experience in Java, two students have more than 5 years of experience, and one student has about one year of experience.

In the projects, students utilized different tactics, as described in Sect. 4.2 and illustrated in Figure 3. Furthermore, students were asked if they used different tactics in combination. They responded that they combined, e.g., SES, SER, and QFE. In the following, we investigate the research questions based on the survey results.

4.3.1 RQ1—Improved Awareness

Since students were asked to implement a mobile application in two stages, we were interested into improvements of energy efficiency and/or resource adaptability while developing their apps. Therefore, we were, notably, interested into improved awareness regarding the special energy-efficiency requirements, and the respective realization strategies. In Table 10, we give the results on a 5-point Likert scale, and we present the mode and mean values.

Our initial validation shows that students strongly agreed on an improved general awareness of energy-efficiency requirements. However, regarding the awareness of consequences of decisions made to realize such requirements, we could yet not find major improvements.

---

Table 9: Mobile sensing tactics for resource adaptability, category: processing

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSV</td>
<td>Mask Variability</td>
<td>A resource variant might deviate in the delivered QoS or in the type of sensor data provided. This tactic recommends developing algorithms handling the variability of resources to mask their impact. <strong>Examples:</strong> Kjærgaard et al. [14] present methods for MSV for the heterogeneity of WiFi sensor data for positioning. Nirjon et al. [33] present methods for MSV to improve GPS reception indoors by better processing of indoor measurements with low quality.</td>
</tr>
<tr>
<td>DMM</td>
<td>Domain Modeling</td>
<td>Given a lack of resources, appropriate domain models can help improving QoS. Assuming a lack of communication or sensor resources, this tactic recommends using domain models to extrapolate sensor data and other information to deliver a rudimentary QoS. <strong>Examples:</strong> Toftkjaer et al. [50] suggest applying particle filters with building models to the indoor sensor data and other information to deliver a rudimentary QoS.</td>
</tr>
<tr>
<td>RSF</td>
<td>Resource Fusion</td>
<td>If different resources are available, QoS can be improved by fusing resources to work in symbiosis. This tactic recommends combining several kinds of resources to deliver a service with an improved QoS. Since resources are heterogeneous, a composition method must explicitly address the creation of an integrated service. <strong>Examples:</strong> Toftkjaer et al. [50] present methods to improve QoS for indoor positioning by fusing input from inertial sensors, GPS sensors, and building models using particle filters. Parate et al. [34] present CQue fusing output of individual classifiers and sensors to derive an individual’s habitual patterns and associated correlations with context.</td>
</tr>
</tbody>
</table>

---

Technologies in which the participating students were asked to develop a mobile sensing application in groups of three students and in two stages (Sect. 4.2). Initially, the students developed the application in the “classic” way. In step 2, students were introduced into the design tactics, and were asked to develop the application again explicitly considering energy efficiency and/or resource adaptability. Having finished the second run, students were surveyed for which tactics did they use and combine. In particular, we were interested into answering the following two research questions:

**RQ1:** Do the architecture design tactics improve awareness of architectural challenges of mobile sensing? This question aims at investigating the general perception of the design tactics and if the proposed tactics help to create awareness of the associated architectural challenges.

**RQ2:** Do the architecture design tactics improve the design and development of mobile applications? A catalog of tactics can be considered helpful or limiting. This question aims to study if the students perceived improvements in the design and development tasks or if they experienced, e.g., restrictions, increased complexity, and development overhead.

---
4.3.2 RQ2—Improved Design & Development

Our second research question aims at investigating whether the students perceived the tactics beneficial to design and implement a mobile application utilizing the tactics for energy efficiency. Figure 4 shows the results (modes and mean) of the questionnaire presented on a 5-point Likert scale. So far, the students agreed on a simplification of design and implementation tasks. Furthermore, students agreed on the straightforward use and the ease of application in general—“Overall they make perfect sense and can break the software implementation processes to make it more manageable.” (Participant 9, more than 5 years of experience). Furthermore, students consider the overhead coming along with applying the tactics critical—“More time consuming for small projects.” (Participant 2, more than 5 years of experience). Finally, students did not experience restrictions in the measurability of energy efficiency by applying the tactics.

4.4 Discussion

The initial validation analyzed if the proposed tactics were perceived beneficial to develop energy-efficient and resource adaptable mobile apps. From the technical point of view, applying the tactics showed beneficial, since improvements in energy efficiency could be achieved (Sect. 4.2). Utilizing the tactics significantly improved energy consumption with (relatively) little overhead. However, these improvements could also be a consequence of the students’ learning curve. Since the observation of this particular effect was not the main focus of the presented case study, in future investigations, further means to address the threats to validity must be used to improve data quality.

Regarding the perception of the tactics, so far, we found improvements of the general awareness, and of the design and implementation tasks (Sect. 4.3). However, this case study has to be considered a first step toward an in-depth investigation of the benefits of the proposed tactics. In the course in which we conducted the case study, only 9 students participated in the study. Furthermore, only 8 out of 12 energy-efficiency tactics were applied, only 3 out of 7 resource-adaptability tactics were applied, and only few tactics were combined to better achieve energy-efficiency goals. That is, conclusions drawn so far provide only initial indication toward improved awareness.

In future investigations, scenarios need to be created in which more tactics can be utilized. Moreover, since context plays an important role, the case study design should be improved in order to address multiple cases providing more setups to apply different (combinations of) tactics. Thus, the next step is to conduct a controlled experiment according to the guidelines described by Wohlin et al. [52] to create reference data and a research design that allows for independent replication. Furthermore, a major threat to validity of the study is its execution in a university course using students as subjects [43]. Future studies thus have to involve industry to provide insights into practically relevant settings.

5. CONCLUSIONS

In this paper, we presented tactics to address architecture design challenges for mobile sensing systems. In particular, we discussed the two architectural qualities energy efficiency and resource adaptability. The description of these qualities was given by general scenario-generation tables to support the systematic specification of architecture requirements. Furthermore, in order to enable architects and developers to systematically apply proven methods, we proposed a catalog of architectural tactics distilled from literature. In summary, we provided 12 tactics (in 4 categories) to ad-
dress the energy efficiency attribute, and 7 tactics (in 3 categories) to address the resource adaptability attribute. For each tactic, we provided rationale and examples to relate the respective tactics to particular cases illustrating their use in practice.

In order to study the feasibility of the developed catalog, we provided a first case study, which was conducted with student teams. Our initial findings are two-fold: First, from the technical perspective, applying the tactics showed significant drops in energy consumption and, second, initial findings indicate that the tactics provide a proper guideline, and improve awareness of the special challenges of energy-efficient and resource-adaptable architecture design.

However, the contributions of the paper have some limitations. The paper at hand needs to be considered a first step toward a more systematic approach of engineering energy-efficient and resource-adaptable mobile sensing systems. So far, as a first step, we provided the systematization for only two selected quality attributes. Since the proposed tactics showed beneficial, extending the catalog with the remaining quality attributes is subject to future work. Furthermore, so far, we only provided an initial validation of the tactics in a university-hosted case study. That is, further investigations—especially in an industry context—are required to improve the initial data base, which allows for better rating the tactics tradeoffs and benefits.

Acknowledgment
We would like to thank the students that participated in the case study on the tactics’ validation, and that spent a vast amount of time to realize their mobile sensing projects.

6. REFERENCES