1

Introduction

This chapter first introduces the motivation behind the developments described in this book. Then, it discusses the main objectives of GAMBAS and describes the two motivating scenarios in the domain of mobility and environmental monitoring. Based on these scenarios, the chapter derives the overall vision and identifies the innovative characteristics. Finally, the chapter closes with a discussion of the state of the art that is used to highlight the primary innovations realized by the development of the GAMBAS middleware.

1.1 Motivation

With the advent of powerful personal mobile devices such as smart phones, digital assistants and tablet computers, an ever-increasing number of people has constant access to the wealth of information stored on the millions of servers connected via the Internet. Over the last years, the availability of such devices has caused a paradigm shift in the way people deal with information. Instead of collecting and printing potentially relevant documents in advance, using a personal computer that is only available at particular locations, they now access information on-demand and on-the-go.

Yet, despite this significant change in behavior, the technical means to access information have only changed marginally. As depicted in Figure 1.1, in most cases, information is accessed via the web, which requires users to memorize long URLs, click through sequences of web pages or browse irrelevant search results. Alternatively, if they are frequently accessing the same service, they may install an app or application that provides more convenient access. However, such an installation requires advance planning and does not provide suitable support for services that are primarily useful in a particular environment. Moreover, even if they are using a local proxy,

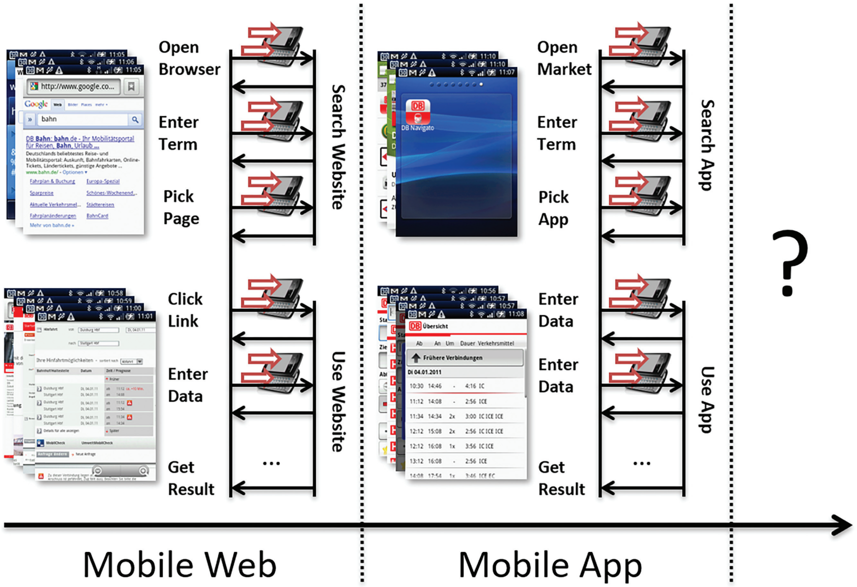

Figure 1.1 The Challenge.

the utilization of a more complex service, for example, to book a train ticket, requires users to specify numerous inputs such as destination, time, etc. using miniaturized and often, inadequate peripherals. As a consequence, the state of the art puts a natural limit on the complexity of the software and thus on the level of support that can be gained from existing services.

In contrast, ubiquitous computing [Wei91] envisions services which provide seamless and distraction-free support for simple and complex everyday tasks of their users. In order to realize this vision, the set of services available and the services themselves must be adapted to the users situation, behavior and to varying user intents. Thereby, adaptation must be performed autonomously in order to ensure that it does not conflict with the goal of providing a distraction-free user experience. This, in turn, requires services to gather a broad range of characteristics of the user's context at runtime. Examples for these characteristics include the user's location, activity, plans and goals.

Personal mobile devices such as smart mobile phones and personal digital assistants provide a promising basis for determining user context in an automated manner on a large scale. The reasons for this are manifold. First and foremost, personal mobile devices are self-contained and do not require additional infrastructure support, but existing cellular and wireless local area networks can provide the backbone for device interaction if needed. Secondly, though these devices are resource-constrained, newer generations are designed to support more complex tasks such as displaying a high-resolution movie. As a consequence, the devices are often not utilized to their fullest capacity, leaving enough resources to perform context recognition. Thirdly, with a variety of on-board sensor, personal mobile devices have access to both physical and virtual data sources, which allows multi-modal context recognition with high precision. Lastly, since the devices are carried by and owned by a single user continuously, the device's context is tightly correlated to the user's context and the recognition alone does not invade privacy.

In the past, these characteristics have contributed to the development of a number of context recognition systems for personal mobile devices. The recognition methods applied by existing systems are usually fine-tuned for specific requirements in order to provide reasonably accurate results while requiring limited resources. Although these methods are suitable for accurately detecting desired characteristics, they cover only a narrow set that can be detected by one device. Moreover, due to the resource-constrained nature of personal mobile devices, developers have usually concentrated on providing solutions for a concrete service.

The vision of ubiquitous computing, however, extends beyond the boundaries of a single service as it envisions seamless support for everyday tasks. As a consequence, achieving the overall vision of ubiquitous computing raises a number of challenges which include:

- the development of concepts to support the automated recognition of a broad range of context information types to support a variety of application scenarios in a generic fashion,
- the development of context recognition methods that are able to cope with the limited resource availability and energy constraints of personal mobile devices,
- the development of novel data acquisition and distribution protocols to share context information in order to increase the recognition accuracy without endangering privacy,
- the definition of an interoperable data representation model for context information and associated query models to support machine-to-machine communication,
- the design of a scalable data infrastructure to share and aggregate possibly frequently changing context information gathered by a large number of devices,

- the development of tools to reduce the required amount of manual configuration of policies and the mechanisms to validate them in order to protect the privacy of users,
- the design of new context-based human computer interaction techniques that are able to incorporate user goals and intents.

1.2 GAMBAS Objectives

The main objective of the GAMBAS project was to develop an innovative and adaptive data acquisition and processing middleware to enable the privacy-preserving and automated use of behavior-driven services that are able to adapt autonomously to the context of their users. Towards this end, GAMBAS was set up to address the complete set of challenges listed in the previous section in order to provide a truly integrated solution, thus closing a significant gap between the systems that were in use at the time and the vision of ubiquitous computing. The primary result of the project was the design, implementation and validation of a **G**eneric **A**daptive **M**iddleware, i.e. a set of application-independent services, to support the development and utilization of **B**ehavior-driven **A**utonomous **S**ervices.

As depicted in Figure 1.2, the GAMBAS middleware enables the development of novel applications and Internet-based services that utilize context information in order to adapt to the behavior of the user autonomously. To do this, the middleware provides the means to gather context in a generic, yet resource-efficient manner and it supports the privacy-preserving sharing of the acquired data. Thereby, it applies interoperable data representations which support scalable processing of data gathered from a large number of devices. In order to make the resulting services accessible to the user, the middleware supports intent-aware interaction, e.g., by providing recommendations for services, which minimizes the need for user inputs.

The realization of this middleware accompanied the development and integration of a flexible context recognition framework that is able to capture the context of users (e.g. location, activity, plans, intents), an interoperable data model to represent context information, a scalable data processing infrastructure to query and aggregate context information and to integrate context into services, a suite of security protocols to enforce the user's privacy when sharing context information and last but not least, a system to largely automate the discovery and selection of relevant services available to the user. In addition, it encompassed the development of tools to simplify the configuration of privacy policies, which ensures that the user's

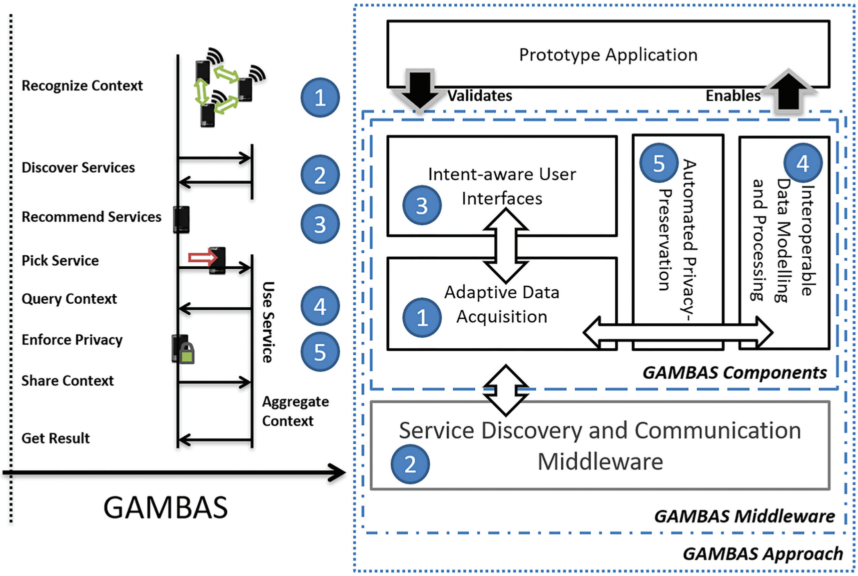

Figure 1.2 Approach and Components.

privacy expectations are met to improve the user experience and to increase user acceptance.

As a consequence, the implementation of this middleware by the members of the GAMBAS consortium resulted in a number of innovations in the research areas of context recognition, data modeling and processing and privacy preservation.

1.3 Application Scenarios

To define the scope of the vision addressed by the GAMBAS consortium, we first introduce the two scenarios that motivated the project. Thereafter, we discuss how they fit to the overall vision.

1.3.1 Mobility Scenario

John has just arrived to a new city. At the airport, he receives a message on his mobile phone by Bluetooth broadcast welcoming him to the city and inviting him to download an application on his smart phone in order to make his life in the city easier and to make his visit more enjoyable.

He follows the link proposed by the welcome message and downloads the application. When booting the application for the first time, he is requested to provide some data that does not affect his privacy. From that moment on, the application begins to capture the context related to John's interests including the change of positions, used transport modes, visited shops, etc.

The interface requests John to select which type of information he is interested in. John can choose from different sources of information and services. For this short visit to the city, John selects the mobility, events and shopping layers. The selection of "events" invites John to refine his selection and choose among different kinds of events: sports, theater, exhibitions, conferences, etc. John selects sports and theater.

The interface also suggests John to connect his smart phone application with social networks such as FourSquare, Facebook and Twitter. John selects Foursquare in order to publish his "check-in" events and share them with his friends in the city.

As it is the first time John visits the city, and he has just downloaded the application, the application is not able to predict the targeted destination of John. His city behavior profile has been just created and the phone's calendar is empty.

Thus, the application asks John: **What do you want to do?**

John responds via voice **I want to go to the hotel Astoria**. The smart interface of the application detects and recognizes the semantics of the phrase **go to** and **hotel Astoria** and suggests this destination. John confirms this selection with a simple gesture on his smart phone.

The application then shows John the route through public transport means to reach his destination. John begins his trip first by metro and then continues by bus. The application on his phone is able to detect at any time where John is and alerts John shortly before he has to leave the metro. Thereby, it notifies him about which bus to take next.

If he decides to leave the recommended route, he can do so at any point and at any time. If he decides to go for a walk in the city, he can leave the route and get updated route recommendations. At any point, he can look up information on the bus stops and metro stations or other points of interest (POI) making use of speech recognition combined with semantic services.

During the journey, the application informs John of the sport and theater events taking place in the next days in the city.

When he is close to his destination and since it is already lunch time, his smart phone suggests three restaurants nearby his hotel. At any point during his visit to the city, John can identify locations with a voice tag. At the location of the selected restaurant, he can use the voice recognition system to tag the location **Luigi's restaurant** or **good pasta**. Later on, the voice recognition will be able to use this information to lead John back to the restaurant.

After lunch, the application suggests buying in **Cortefiel** next to his hotel that has a two-for-one offer on spring shirts.

Once arrived at his destination, the application detects his **check-in** and suggests John to publish the event on his enabled social networks. John accepts the suggestion and according to his settings, his location is published in **Foursquare**. Once the goal is achieved, i.e. arrival at the destination – the application returns to its initial state, **What do you want to do?**

This time, John ignores his smart phone however. While John is in the city, the application keeps analyzing his behavior and suggesting information and services based on his position and preferences.

The application can notify John about shopping deals depending on his position and the proximity of the shops. Thus, the interaction with the user becomes more efficient and the GAMBAS framework is capable of filtering the offers, resulting in distraction-free support for the user's tasks.

1.3.2 Environmental Scenario

Paul is a regular user of the smart city application on his mobile phone. He uses it often to find the best options to get around in the city. For this, he is always subscribed to the mobility layer.

Today, he has decided to do some sport around the city, and his friend Ringo has explained him how to make use of the smart city application to obtain a jogging route through the less polluted areas of the city (CO2 levels). He indicates the number of kilometers he wants to run, and for how long, and he also specifies that if possible he would like to run with a friend.

As a result, the smart city application offers him a route with Ringo. Paul observes that in order to have a reasonable route, the mobile application is proposing to take first a bus to the starting point of his jogging route.

At the same time, Ringo, who was already planning to go jogging, receives an alert asking him if he wants to share a route with Paul. He accepts and both friends receive a confirmation on the appointment in their agendas.

Ringo is not as concerned when it comes to environmental issues as Paul, so he does not use the public transport. Instead its smart city application proposes him a route by car through an urban tolling area. He is though quite concerned about costs, and by default he is subscribed to the mobility layer offering him a car pooling services. The urban tolling in the city depends on a number of factors such as type of vehicle used, number of passengers in the car and level of pollution in the city. Ringo receives a proposal from the application to share the trip with his friend George.

When activating the environmental layer on his mobile phone – in order to access the levels of CO2 in the city – Paul has accepted to join the group of users collaborating with the municipality to study the noise levels in the city. Without any further intervention from his side, its mobile phone records and processes measurements of noise level each time Paul is outdoors and changes his position. At the end of the day, Paul can access the city pollution map application and check the noise levels in the route he has being following, including the jogging activity. Moreover, he obtains his environmental footprint due to the trip on public bus.

1.4 Overarching Vision

Given the advances in computer technology and the proliferation of wireless communication and sensing technologies, GAMBAS envisions the realization of major parts of the ubiquitous computing vision by means of a cloud of intelligent services, which provides adaptive and predictive information to people.

The basis for providing this information is the ability to automatically capture the state of the physical world by means of personal mobile devices as well as other sensing-enabled devices integrated in stationary or mobile Internet-connected objects. Given a variety of observations made by these sensors, the devices of a person can observe parts of its behavior which, in turn, can then be used to estimate and possibly predict parts of the person's behavior by means of a profile.

Upon request of the person, different views on this profile can be exposed (in a tightly controlled fashion) to different services such that they can adapt themselves not only to the person's current situation but also to some of the person's future intents. Thereby, the adaptive services might have to interact with other services as well as the personal mobile devices of other persons.

This creates dynamic mashups of services that share and integrate the information managed by them. To allow the ad hoc creation of such mashups, the information managed by each services and the information available on personal mobile devices must be discoverable. In addition, in order to seamlessly combine the information provided by different sources, it must be possible to easily link different pieces of information. This requires the use of a common, extensible and interoperable data representation to allow data processing that extends beyond the boundaries of a single service or device.

Based on the data provided by these dynamic service mashups, GAMBAS envisions new types of user interaction paradigms that transform the reactive information retrieval that is commonly applied by most Internet services into a proactive information provisioning that emerges from this system of Internet-connected objects and services.

1.4.1 Smart Cities

By employing the overarching vision described previously to the context of smart cities, it is possible to further detail the vision without narrowing its general applicability. GAMBAS envisions a smart city as a cloud of intelligent digital services that provides adaptive and predictive information to citizens. GAMBAS foresees a variety of services that manage different types of information that relates to the city as depicted in Figure 1.3.

Conceptually, these services and their data can be grouped into the so-called layers that cover different aspects of people's life in the city. A shopping layer, for example, might encompass services that manage store

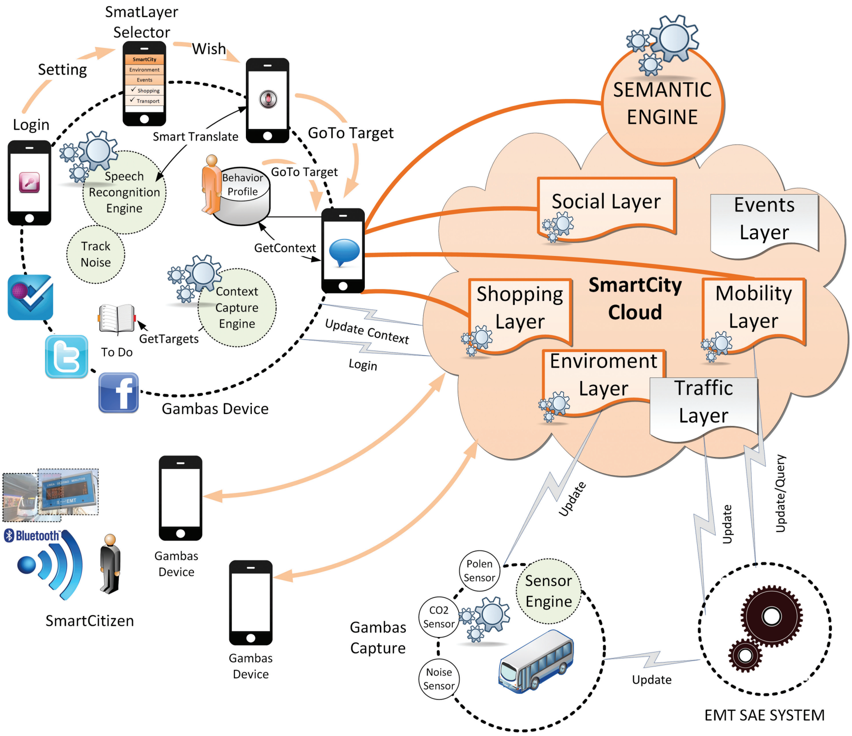

Figure 1.3 Smart City Vision.

locations and special offers or recommendations on products and experience reports on different stores. Similarly, a mobility layer might encompass services that manage taxi locations, bus routes, subway stations or traffic information. A social layer might manage relationships between citizens, events that take place in the city, bar and restaurant locations, recommendations, etc. An environmental layer might manage information related to water or air quality in the city or it might capture the noise levels at different places. Clearly, some of the services found in these layers can apply to multiple layers as some pieces of information and some of the services might be applicable to multiple aspects. As a simple example, both the shopping and the mobility layer may rely on generic geographic information about the smart city.

In order to enable the creation of dynamic mashups of services, the services export (parts of) their information. The information is then represented using an interoperable data representation that allows automatic linking of different pieces of information. This makes the information accessible to other services which can then add additional value by providing, for example, a better experience for a specific group of citizens. In order to simplify the integration of services, a distributed query processing system enables the execution of queries across different information sources.

To provide up-to-date information and adaptive information to users, the layers capture information from different sensors embedded in various Internet-connected objects. The objects may belong either to a particular service provider or to a citizen. The devices in the first category, may, for example, encompass sensors embedded in a taxi or a bus or they may be deployed at specific positions such as a bus stop or a metro station. The devices in the second category may encompass the personal mobile devices of the citizens such as their smart phones but they also may contain traditional systems such as their desktops at home, for example.

To protect the privacy of the citizens, they can control the collection and sharing of data with the services in different layers. Towards this end, behavioral data is stored and processed on the devices that belong to the citizen. Optionally, in order to access additional services, they may share their information with specific service providers or other citizens. In order to avoid the expensive task of manually controlling the sharing process, automatic proposals for different settings can be computed based on social relationships that are formalized by means of existing policies that the citizens created for similar contexts.

To access the information from services and to perceive their current context, citizens will run a special application on their personal devices. The application performs predictions based on the citizens' past behavior. Given these predictions, the application is able to proactively retrieve information from different layers that are interesting for a particular citizen. Furthermore, it automatically determines appropriate times to notify the citizen about important events. For example, when traveling in a bus, the application may notify the citizen shortly before the bus stops at the target destination such that the citizen does not miss the bus stop. In cases where the citizen is exploring new terrains that cannot be predicted, a natural user interface based on speech recognition technology allows the citizen to specify alternative goals.

1.4.2 Characteristics

Based on this smart city architecture, it is possible to identify two key characteristics that differentiate the basic idea from other approaches in this application domain:

- **Adaptive Acquisition and Presentation:** In terms of data acquisition, the GAMBAS vision foresees citizens not only as consumers of digital services, but also as an important source of information that can provide feedback to different stakeholders. This feedback can then be used to adapt services, which results in mutual benefits for both the citizens and the providers of services.
- **Dynamic and Distributed Processing:** In terms of data processing, the GAMBAS vision foresees high dynamics that depend on the individual behavior of citizens as well as the results of their aggregation. This enables novel services that go beyond the possibilities of today's service infrastructures as they are typically focusing on isolated operation (often referred to as "data silos") or they solely combine a few data sources that are determined statically.

1.5 State of the Art

Although it has not been realized so far, overall, the vision of ubiquitous computing as defined by Mark Weiser [Wei91] is not new. Ever since its formulation in 1991, researchers and practitioners have focused on closing different research gaps. With respect to middleware issues, a significant amount of research has been performed in the area of enabling seamless device interaction as well as application adaptation. Furthermore, there have been considerable efforts in the area of enabling context management, which is an important basis for context-adaptive applications. Lately, the availability of results in these areas has led to the development of a number of large-scale sensing applications. In the following, we briefly review the state of the art in each of these fields, but before we do this, we quickly review the available hardware technologies.

1.5.1 Hardware Technologies

As for any other software project, the execution environment for the GAMBAS middleware is defined by a subset of the existing hardware platforms. Due to the specific focus of GAMBAS on enabling adaptive data acquisition with Internet-connected objects, in the following, we briefly discuss the available hardware technologies with respect to devices, communication and sensing. The focus is not to provide a comprehensive list of available technologies. Instead, we take a more high-level perspective that uses current technology as examples but, in principle, is independent of the concrete implementation. Based on the resulting discussion of device, communication and sensing technology, we then introduce a classification of device types that are the basis for GAMBAS. Thereby, it is noteworthy to mention that not all features of the GAMBAS middleware are realized for all types of devices. However, it enables their integration into a single system.

1.5.1.1 Devices

The devices forming the Internet of Things are heterogeneous. For example, besides traditional personal computer systems, a significant number of devices are either mobile or integrated. When analyzing the different types of devices, we can categorize them with respect to several orthogonal axes.

1. **Specialization:** Naturally, we can classify devices on the degree of specialization. This degree may range from general purpose devices such as PCs or laptops to special purpose devices such as micro-controllers that are integrated into all kinds of objects. Although, in principle, the concepts developed by GAMBAS are applicable to all kinds of Internet-connected objects, the GAMBAS middleware does not focus on the latter. The reason for this is that highly specialized devices are often closed systems that cannot be programmed easily. However, given the rapid advances of technology, we can expect that many closed devices will open up in the future.
2. **Resources:** Independent of the degree of specialization, we can classify devices on the available resources. On the one end of the spectrum, the set of devices forming the Internet of Things may encompass resource-rich devices such as mainframes or clusters of workstations. On the other end, they may contain resource-poor devices such as simple sensor

nodes. In between, there are devices such as laptops or devices with less resources such as mobile phones or tablets.

3. **Mobility:** Another important axis is the degree of mobility. Here, we can distinguish stationary devices and mobile devices. In contrast to stationary devices, mobile devices are usually equipped with batteries and thus, their energy is a limited resource that needs to be managed appropriately. This is especially true, when using mobile devices for long-running tasks such as the continuous monitoring of the environment.
4. **Interaction:** Last but not least, the devices can also be classified based on their capability of supporting immediate interaction with a user. Here, we can distinguish devices that support user inputs, e.g., by means of graphical or audible interfaces, and devices that are invisibly integrated other objects. This axis is particularly relevant since only devices that support the interaction with a user can be configured manually by the user. Due to the invisible integration, the remaining devices can solely be configured indirectly through other devices.

1.5.1.2 Communication

Existing communication technologies can be broadly classified into wired and wireless. Due to the success of mobile devices, the latter ones have become main stream over the last couple of years. At the present time, there are several technologies that are widely available and frequently integrated into mobile devices. They cover the complete spectrum from low to high speeds and low to high range. At the same time, they exhibit vastly different energy profiles.

- **Near-Field Communication** is a set of standards to enable radio communication between devices by bringing them in close proximity. NFC is based on existing standards on radio frequency identification (RFID). In contrast to other technologies in that family, it enables bi-directional communication between two devices. However, it offers only low transmission speeds and it is only applicable to very close range communication (i.e. few centimeters). At the present time, it is mostly used for mobile payment systems or in order to bootstrap connections with other communication technologies (e.g., Bluetooth).
- **ZigBee** is a standard for short-range communication. ZigBee is specifically designed for low-power devices with low data rate and short-range communication capabilities. The IEEE standard 802.15.4 defines the physical and the MAC layer for ZigBee. The devices in a ZigBee setup can be categorized into ZigBee coordinators, ZigBee routers and ZigBee

devices. The ZigBee coordinator is the central entity that keeps record of the devices in the network as well of the other ZigBee coordinators. The ZigBee router is responsible for routing messages and associating devices with each other. Devices that are not ZigBee coordinators or ZigBee routers are classified as ZigBee devices.

- **Bluetooth** is another popular short-range communication standard. Bluetooth modules are commonly available for standard computers and various peripherals. These modules support low-bandwidth and short-range communication. Depending on the communication range and energy consumption, Bluetooth devices are divided into three classes. Class 1 Bluetooth devices consume around 100 mW and support approximately 100 m. Class 2 Bluetooth devices consumes up to 2.5 mW and support communication range of approximately 10 m. Class 3 consumes the minimal power (1 mW) but also provide the shortest communication range (approximately 1 m).
- **Wi-Fi** is probably the most popular communication standard for connecting various devices such as laptops or mobile phones wirelessly. Wi-Fi certification is given to the devices with wireless capabilities that implement IEEE 802.11 standards. There exist several 802.11 standards that include 802.11a, 802.11b, 802.11g and 802.11n. Wi-Fi-supported routers cover approximately 100 m in outdoors. Since the clients in the Wi-Fi network do not require wire, the network can be easily extended. Wi-Fi-enabled devices can move in a limited area but they have relatively short range. A possible shortcoming of Wi-Fi-certified devices is that they have comparatively higher energy requirements.
- **UMTS** (Universal Mobile Telecommunications System) is the successor of GSM and designed to support third-generation telephone technology. UMTS is specifically designed to support advanced services. It is developed to support 14 Mbps data transfer rate and UMTS support is now commonly available in most smart phones. Compared to its predecessor (GSM), it consumes more power. However, in terms of speed and service capabilities, it is a significant improvement over GSM.

For stationary devices, wired communication technologies are still an important alternative to wireless technology. Many stationary general-purpose devices such as servers are usually connected with Ethernet.

- **Ethernet** is based on IEEE 802.3 specification and is a very popular LAN technology. The specification defines standard for physical layer as well as data link layer of the OSI model. Starting with 10 Mbps, it has evolved to support 100 Mbps (fast Ethernet) and later 1000 Mbps

(Gigabit Ethernet) speed. Currently, the fastest speed standard supported by Ethernet is 10 Gbps, although we can assume that there will be further progress on connection speeds.

1.5.1.3 Sensing

Besides device and communication technologies, the third hardware pillar of GAMBAS is sensing technology. Over the last couple of years, device manufacturers have started to integrate various sensors into different types of devices. At the present time, current mobile devices such as smart phones and tablets commonly exhibit the following combination of sensors:

- **Accelerometer:** Accelerometers are used to measure the acceleration that a device experiences. In most cases, they are able to differentiate acceleration along three axes. Usually, they are used to adapt the screen orientation of the device according to the way the user is holding it. However, researchers have also used accelerometers for various other tasks such as classifying the mode of locomotion or detecting potholes.
- **Gyroscope:** More recently, device manufacturers have started to add gyroscopes to the set of standard sensors that are available on mobile phones. Gyroscopes are used to measure the orientation of a device. Advanced applications include inertial navigation systems, for example. However, at the present time, they are mostly used to support gaming.
- **Microphone:** As a natural consequence of their function, all mobile phones are equipped with microphones that allow them to record and transmit voice during a call. However, in addition to that, most devices nowadays exhibit multiple microphones (e.g. to enable automatic noise reduction) that can also be used to capture and analyze ambient sound.
- **Proximity:** In order to activate and deactivate the screen automatically during a call, many mobile phones are equipped with proximity sensors that can measure the distance between the phone and another object (typically in front of the screen) in a course-grained scale (e.g. far or close).
- **GPS:** To support location-based services and to support user navigation, many mobile devices are equipped with GPS receivers. Although they cannot be used reliably in indoor environments, outdoors they provide reliable localization with 5 to 10 m accuracy.
- **Camera:** Similar to microphones, nowadays, most mobile phones and tablets are equipped with cameras which can be used to record videos as well as still images. In addition to simply taking pictures or recording videos, they can also be used to recognize visual tags (e.g. QR-Tags) and

they can be used for different types of context recognition applications (e.g. to automatically detect gas station prices).

- **RF:** Although they are mostly intended for communication, RF-based communication technologies such as Wi-Fi or GSM can also be used to extend the capabilities of other sensors such as GPS, for example. The use of these technologies as sensors usually requires special maps that model the signal propagation in a certain area. Using these maps, a course-grained but energy-efficient localization can be supported.

Besides mobile devices, researchers have also developed a number of sensing platforms such as Berkeley Mica2 or UCLA iBadge, etc., mostly in the area of wireless sensor networks. Typically, these platforms can be extended with different types of sensors, but most of them contain at least the following combination of sensors.

- **Light:** Light sensors typically measure the light level received at a particular point of the device. In many cases, light sensors are directly built into the sensor node or they can be added by attaching a sensor board.
- **Temperature:** Temperature sensors typically measure the ambient temperature of the sensor node. In many cases, the sensors are not calibrated and the raw values need to be converted programmatically to the usual Celsius or Fahrenheit scale.
- **Pressure:** Pressure sensors typically measure the barometric pressure, and thus, they can be used to compute the altitude.
- **Humidity:** Humidity sensors typically measure the humidity using capacitive measurements. In many cases, they are bundled with temperature sensors.

In addition to mobile devices and sensor nodes, there are numerous application-specific sensors. Due to their great variety, it is not possible to provide a comprehensive list here. To name some examples that may be relevant in the context of GAMBAS, using the OBD unit of a modern car, it is possible to capture various engine-related values. These include, for example, the current fuel consumption or the current state of the catalytic converter.

1.5.1.4 Classification

Based on the previous discussion of device, communication and sensing technologies, we can identify four broad classes of devices that are forming the hardware platform for services developed with the GAMBAS middleware.

Intuitively, based on the capabilities of the device, the support provided by the middleware differs.

- **Back-end computer system (BCS):** Back-end systems usually consist of one (or more) general-purpose computer that is connected to the Internet via a wired and often high-speed connection. Usually, they exhibit high storage and processing capacities and they are shared by multiple users remotely, i.e. through the Internet. Consequently, to most of their users, they do not expose a physical interface that would enable interaction. Instead, they are accessed through web-browsers or via custom applications that are performing some form of remote call (e.g. RPC, RMI, etc.). The GAMBAS middleware uses these systems for data storage, aggregation and processing.
- **Traditional computer system (TCS):** Traditional computer systems encompass workstations, desktops and laptops. If they are stationary, they are typically connected via wired connections. If they are mobile, like laptops, the predominant communication technology is Wi-Fi. In some cases, they are equipped with a few sensors (e.g. microphones, cameras, accelerometers for hard disk protection). Usually, they are accessed and used by a single user (e.g. personal desktop/laptop) or a small group (e.g. shared workstation). Although, they have fewer resources than most back-end computer systems, when considering that they are not shared between many users, the ratio of resources to number of users may be equally high. The GAMBAS middleware uses these systems primarily to perform similar tasks as back-end systems (although on a smaller scale). However, it also enables their usage as sensing devices.
- **Constrained computer system (CCS):** Constrained computer systems include mobile devices such as smart phones and tablets. Furthermore, they include stationary devices such as set top boxes or industrial PCs. When compared with traditional computer systems, they exhibit a significantly lower amount of computing resources with less capable processor architectures (e.g. ARM instead of X64) and less amount of memory (e.g. MB instead of GB). In many cases, they are equipped with a multitude of built-in sensors (e.g. mobile phone) or they can be attached to application-specific sensors (e.g. industrial PC). Consequently, they provide the primary basis for data acquisition in GAMBAS. In addition, they are also used as a personal data storage that can be accessed remotely.

- **Embedded computer system (ECS):** Embedded computer systems include highly specialized micro-controllers or ASICs that are built into existing products such as a dishwasher or a car. Furthermore, they include less specialized sensor platforms that may be programmable such as a SunSPOT or a Mica2 node. Usually, these systems are not directly connected to the Internet. Instead, they can be connected through some gateway device that mediates the interaction. Although such embedded devices outnumber the other classes, the GAMBAS middleware does not focus on the use of these devices as a primary processing platform. The reason for this is that usually, these devices cannot provide their function without additional computing infrastructure. Furthermore, in many cases, they are not equipped with easily accessible interfaces or they do not provide sufficient computing resources to implement additional services. However, the GAMBAS middleware supports their use as data sources when combined with a more capable device such as a constrained computer system or a traditional computer system.

1.5.2 Communication Middleware

Regarding device interaction, researchers and practitioners have developed a number of communication middleware systems to enable the seamless and trustworthy cooperation of a heterogeneous set of possibly resource-poor connected objects. Examples for past and present research projects in this area are the 3PC [3PC12], GAIA [RJH02] and AURA [GSSS02] projects or the PECES FP7 project [PEC12], to name a few. Traditionally, the resulting systems either focused on enabling the interaction of devices at a specific geographic location (i.e. the so-called smart spaces) or focused on enabling the interaction of devices that are in close proximity (i.e. the so-called smart peer groups). More recently, systems such as the PECES middleware have integrated and extended these two concepts by enabling the interaction within a smart space that is formed by devices in close proximity and beyond smart spaces by enabling device interaction across the Internet in a peer-to-peer fashion. Since the resulting concepts provide a higher degree of flexibility, the GAMBAS middleware will use PECES as its underlying communication middleware.

Besides device interaction, research on communication middleware also addressed the development of new programming paradigms to support the development of adaptive applications, for example, on the basis of goals as done by O2S [PPS$^+$08] or components as done by PCOM [Han09] or flows as done in the ALLOW FP7 project [ALL12]. While these abstractions are interesting to support the development of adaptive applications, the GAMBAS project does not primarily target the development of new abstractions to support application adaptation. Instead, it focuses on the acquisition of context information as well as the processing of environmental information in a privacy-preserving way, which usually provides an important basis for adaptation that is independent of the concrete abstraction that performs the adaptation. Consequently, from a high-level perspective, the goal of GAMBAS is a more fundamental one that will enable the use of such abstractions at a later point in the development process.

1.5.3 Context Management

The importance of context information for the realization of ubiquitous computing has been recognized very early after the formulation of the vision [SAW94]. Over the course of several years, researchers have developed a number of middleware systems to acquire and leverage context information, e.g. [HKL$^+$99], [SDA99], [Bar05]. Traditionally, these systems have either focused on the scalability issues that arise from providing context awareness in an application-independent way using a federated system [HKL$^+$99] or focused on the actual distributed acquisition and usage when developing applications with a limited scale such as a room or a house [SDA99], [Bar05]. In addition to that, specialized context management systems have been integrated into all kinds of middleware systems for smart environments such as GAIA [RJH02] and AURA [GSSS02], to name a few. Similar to [SDA99] and [Bar05], these systems focused on a rather restricted execution environment.

Besides that, the active research in the area of sensor networks and cooperating objects has spawned a number of initiatives to acquire context information from a heterogeneous set of networked sensors that is deployed in an environment. Project at the European level include, for example, the PLANET FP7 project [PLA12] which works on concepts to deploy and operate large-scale sensor networks to capture environmental information. However, usually these systems focus on low-level networking aspects of various sensors or they solve high-level data management aspects resulting from a large number of sensors. Thereby, these systems do not have to consider the resulting privacy implications when moving from environmental context – such as temperature or animal population – to personal context – such as human location, activity and plans.

1.5.4 Sensing Applications

In the recent past, the advances with respect to middleware, device and sensing technologies have led to the development of a number of large-scale sensing applications that are often summarized as participatory sensing [BEH$^+$06] or people-centric sensing [CEL$^+$06] applications. Similar to the goals of GAMBAS, these applications leverage the personal Internet-connected objects of users to capture relevant sensor information. The type of information typically depends heavily on the application area. To give some examples, DietSense [RSB$^+$09] tries to collect diet-related information about the user through photos and sound samples. PEIR [MRS$^+$09] provides an estimate of the environmental impact of a user trip by determining the mode of locomotion. BikeNet [EML$^+$10] captures the biking experience by means of measuring the location and speed and providing an estimate over the used calories. Haze Watch [CYCS12] captures pollution information by attaching external sensors to a mobile phone.

Usually, these and other similar types of applications capture the sensor information at some central application server where it is then processed and analyzed. Furthermore, although they are very similar, they are often built completely from scratch without adequate middleware support. Finally, in most cases, the applications merely inform the user about the collected data by providing some aggregated view on it. The GAMBAS middleware simplifies the development of such applications by providing a scalable, interoperable basis. In contrast to collecting all data at some trustworthy central server, however, the GAMBAS middleware provides configurable sharing that enables users to protect their privacy, if that is desired, which allows users to balance the potential loss of privacy with the potential gaining in service quality. Furthermore, instead of merely aggregating and visualizing the information, the middleware enables the behavior-driven adaptation of services.

1.6 Innovations

Building upon the existing work, the GAMBAS middleware specifically targets the acquisition of personal context information. Consequently, it shares similar goals with several of the existing large-scale sensing applications. However, in contrast to existing applications, GAMBAS also can enforce the user's privacy goals. Towards this end, the acquisition is performed primarily with personal Internet-connected objects. This empowers the user to limit the sharing of the acquired context. In order not to overwhelm the user, the GAMBAS middleware contains a framework to automate the sharing in a privacy-preserving manner. Furthermore, to directly use the acquired context on the connected object, the middleware provides concepts to implement intent-aware user interfaces, which allows the user to have full control over the use of the GAMBAS software via a fine granular system to enable and disable features as needed. Finally, in order to use the shared context effectively in enterprise business processes, the middleware makes use of an interoperable data representation with the associated processing infrastructure that supports a large number of sensors. This provides the basis for efficient object–object interactions and thus, it enables the development of services that can autonomously adapt to the user's behavior.