Chapter 3
Towards a Framework for the Levels and Aspects of Self-aware Computing Systems

Peter Lewis, Kirstie L. Bellman, Christopher Landauer, Lukas Esterle, Kyrre Glette, Ada Diaconescu and Holger Giese

Abstract Increased self-awareness in computing systems can be beneficial in several respects, including a greater capacity to adapt, to build potential for future adaptation in unknown environments, and to explain their behaviour to humans and other systems. When attempting to endow computing systems with a form of self-awareness, it is important to have a clear understanding of what that form looks like. This chapter therefore first introduces the general concept of self-awareness and its various facets. Second, we provide an overview of the range of efforts to interpret the concept of self-awareness in computing. Third, we provide a structured conceptual

---

P. Lewis (✉)
Aston Lab for Intelligent Collectives Engineering (ALICE),
Aston University, Birmingham, UK
e-mail: p.lewis@aston.ac.uk

K.L. Bellman · C. Landauer
Aerospace Integration Science Center,
The Aerospace Corporation, Los Angeles, CA, USA
e-mail: Kirstie.L.Bellman@aero.org

C. Landauer
e-mail: topcycal@gmail.com

L. Esterle
Vienna University of Technology, Vienna, Austria
e-mail: lukas.esterle@tuwien.ac.at

K. Glette
University of Oslo, Oslo, Norway
e-mail: kyrrehg@ifi.uio.no

A. Diaconescu
Equipe S3, Departement INFRES, Telécom ParisTech,
46 rue Barrault, 75013 Paris, France
e-mail: ada.diaconescu@telecom-paristech.fr

H. Giese
Hasso Plattner Institute for Software Systems Engineering
at the University of Potsdam, Prof.-Dr.-Helmert-Str. 2-3, 14482 Potsdam, Germany
e-mail: holger.giese@hpi.de

This chapter is part of the book: Kounev, S., Kephart, J.O., Milenkoski, A., Zhu, X. (Eds.)
Self-aware Computing Systems. Springer 2017.
pp 51-85. https://link.springer.com/chapter/10.1007/978-3-319-47474-8_3

framework that organizes this variety of different forms of self-awareness. This provides a broad set of concepts and a language that can be used to describe and reason about self-aware computing systems.

3.1 Introduction

In this book, we are aiming to implement the self-awareness capabilities, that is to say, capabilities resembling and inspired by human self-awareness, in computing systems. This chapter, and indeed this book as a whole, is based on the idea that increased self-awareness in computing systems *can* be beneficial in several respects. These benefits are centred around systems' increased capacity to adapt, build potential for future adaptation in unknown environments, and explain their behaviour. Such capacity is derived from their ability to learn about themselves, including aspects such as their state, their goals and progress towards them, their behaviour and interactions with other systems, their environment, their perspective, and their evolution over time.

But increased self-awareness will also come at a cost, whether it be in terms of more processing power, increased storage, or the need for further data gathering and communication. In designing self-aware systems, *we* must have the concepts and tools readily available to be able to reason about the benefits and costs increased self-awareness will bring in its various forms. And in realising self-aware computing systems, *the systems themselves* will also require concepts and tools to enable them to perform meta-reasoning about the benefits and costs of their own self-awareness.

The ability of self-aware computing systems to reflect on their their own self-awareness properties and behaviour, termed *meta-self-awareness*, leads to the ability to model and reason about changing trade-offs during the system's lifetime. Meta-self-awareness enables advanced adaptation and explanatory behaviour at multiple levels within a system, including why it has spent resources on self-awareness related activities.

An important distinction to make from the outset is that in the process of doing this, we are not aiming to explain human self-awareness or consciousness. There are those (e.g., [57]) who build intelligent systems with the primary aim of developing a better understanding of human minds and human self-awareness. Instead, we aim to learn and gain inspiration from self-awareness in order to build better computational systems, following the tradition of a range of efforts in computer science that have sought inspiration elsewhere, such as bio-inspired computing, socially inspired computing, and economics-inspired computing. In order to be clear, and in line with a previous work [34], we refer to the forms of self-awareness implemented in a computing system as *computational self-awareness*.

As highlighted by many (e.g., as reviewed Lewis et al. [35]), self-awareness is *not* a binary property; it makes little sense to characterize a system (biological or computational) as either being self-aware or not. Indeed, the term self-awareness encompasses a broad range of related processes and capabilities, which include various types of explicit self-knowledge, an understanding of the subjectivity and

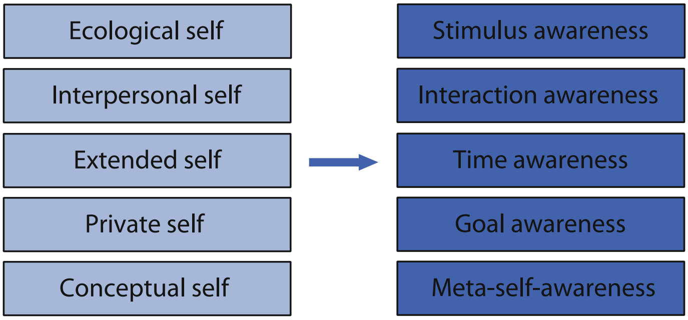

Fig. 3.1 Lewis et al.'s [34] levels of computational self-awareness (*right*) describe various different aspects of self-knowledge and are derived from those by Neisser [44] (*left*). These are discussed in Sects. 3.2.1.4 and 3.2.2.2

context of that knowledge, and processes for its acquisition. Some of these aspects concern situated knowledge acquisition processes (e.g., subjectively context-aware learning, reasoning, and acting), while others are related to the architecture of these learning processes, and how they reflect on and are influenced by each other. In much of the self-awareness literature, particularly in psychology (e.g., [42]), these varieties of self-awareness are characterized as *levels of self-awareness*. Recent work on self-aware computing [34] has borrowed from this, developing the notion of levels of computational self-awareness. Lewis et al.'s levels of computational self-awareness are illustrated in Fig. 3.1.

Our aims in this chapter are as follows. First, we review these treatments of self-awareness and computational self-awareness in the existing literature. Second, we extend initial characterisations of different levels and aspects of computational self-awareness, in order to provide a broad set of concepts and language that can be used to describe and reason about self-aware computing systems. Third, we organize these concepts into a coherent framework that provides a set of related considerations for designers of computing systems who are interested in incorporating self-awareness concepts into their systems. Fourth and finally, we position the definition of self-aware computing systems proposed in Chap. 1 in the context of existing work in self-aware computing. The framework developed in this chapter is then explored in a variety of contexts, throughout the rest of the book.

3.1.1 Why Consider Types of Self-awareness in Computing Systems?

As described above, in this chapter, our aim is to characterize various levels of self-awareness as they might pertain to current and future computing systems. However, first we motivate the need to do this.

As a starting point, since we are claiming that endowing computing systems with increased self-awareness can have benefits, it is clearly important to begin to at least sketch, if not define, what we mean by this. It should stand as obvious that one should not begin implementing a form of computational self-awareness in a system, unless one has a clear understanding of what that form looks like. Based on the multifarious nature of self-awareness briefly discussed above, and elaborated on later in this chapter, we argue that an explicit understanding of the sort of self-awareness targeted is necessary. Indeed, as outlined in recent work on how to translate self-awareness to the engineering domain and apply it in specific scenarios, Lewis et al. [34] highlight the importance of beginning with an understanding of the plethora of different forms self-awareness may take, and the wide range of methods and approaches that can be used to implement self-awareness to different extents, as needed. Chen et al.'s handbook [12] for engineering self-aware computing systems in particular describes a structured process for determining and evaluating the inclusion (or not) of various levels of self-awareness in a system's architecture, and approaches to realising them.

Pragmatically, this is important since as part of the engineering design process, one would want to understand the role and benefits of different levels and aspects of self-awareness, along with different approaches and mechanisms for realising them, before implementing. This supports the ability to choose the most effective methods and the desired level of self-awareness for a given application, its context, and requirements.

We expect that in working with self-aware computing systems, one will often be in situations where there are a number of agents or components within the system that possess self-awareness capabilities, and potentially of different types. Understanding different approaches and levels of self-awareness, and how they interact, will help developers to institute the appropriate communication mechanisms, supporting interfaces and desirable policies among these different entities, for a given purpose. By considering forms of computational self-awareness at an abstract level, our aim is that generic interfaces and communication mechanisms between levels of self-awareness may be specified. If this is achieved, then self-awareness capabilities may be able to be provided in reusable "context-free" ways, to enable the development and evolution of arbitrarily complex architectures appropriate to varying applications.

In this chapter, we organize this plethora of different approaches into a structured conceptual framework, based around *levels of self-awareness*. In doing so, we aim to eventually create a foundation for strategies that consider forms of self-awareness included and excluded by our definition in Chap. 1. This will support the engineering process by enabling the automation of design decisions relating to self-awareness, thereby not only supporting the efforts of engineers, but also of systems themselves, that reason about their own self-awareness.

Our aim is therefore to make progress towards understanding how different application and environmental requirements map to the need to implement different levels of self-awareness (or not), in different ways. Such an understanding should incorporate explicit knowledge of the trade-offs implicit in implementing higher levels of self-awareness including meta-self-awareness. This type of self-knowledge, and techniques by which a system can acquire it itself, will then enable strategies by which transitions between levels of self-awareness in evolving systems may occur.

We would also like to make clear to the reader that in laying out the conceptual framework that we do in this chapter, we are not claiming to provide a definitive characterisation; indeed, we believe it cannot be so, and this framework is one more step along the road to mapping out important considerations in self-aware computing. Therefore, we aim at developing a conceptualisation of relevant and important aspects, and in doing so provide a tool by which engineers and systems may structure their thoughts, considerations, arguments, and decisions. We target the provision of a range of concepts, with an eye to helping the researchers adopt the notions most suitable to their needs and bring out the issues they need to consider as they pertain to self-awareness concerns. As suggested above, we fully expect many systems to ignore some of the aspects presented here, but to do so based on a sound understanding that the costs outweigh the potential benefit in that case. To interpret the framework as a comprehensive list of requirements would be incorrect. Instead, engineers may use the concepts presented to orient themselves, and to ask questions about what the benefits of different kinds of self-awareness would be, and what the costs are of implementing them.

3.1.2 Summary of This Chapter

In summary, this chapter:

- Introduces and briefly discusses the general concept of self-awareness and its various aspects, thereby providing context for the explorations in the rest of the book.
- Briefly surveys the state of the art in interpreting the concept of self-awareness in computing, putting this book in its wider context.
- Discusses the similarities and differences in the computational self-awareness approaches taken so far, with a particular focus on the scope of the approach, with respect to notions of self-awareness.
- Presents a conceptual framework for describing and comparing various levels and aspects of self-awareness, which may be desirable or present in computing systems, including their potential benefits and costs.

3.2 Fundamentals, Inspiration, and Interpretations in Computing

In this book, we are interested in taking inspiration from self-awareness across natural systems, in order to design the future computing systems that are better able to learn, reason, and ultimately adapt and explain themselves. Some of the mechanisms used to achieve the self-awareness in computing systems will also be inspired by naturally occurring ones; others will not. We expect the benefits of self-awareness in computing systems will be most apparent when systems inhabit a world characterized by unfolding situations, particularly when changes are unforeseen at design time. In this section, we first ask the fundamental question: *what is self-awareness?* In other words, what inspiration can we gain, by looking at self-awareness in biological and social systems? Second, we expand on Chap. 2, describing the efforts in computer science and engineering that have directly and explicitly attempted to translate the concepts from self-awareness in natural systems to the computing domain.

3.2.1 What Is Self-awareness?

Self-awareness is a concept long studied in philosophy, and personal and social psychology. One recent example is a work by Morin [42], which defines self-awareness as *"the capacity to become the object of one's own attention."* Many modern theories of human self-awareness stem from work of psychologists since the 1960s, and trace their roots back to notions proposed at the closing of the nineteenth century. Smith [6] and Tawney [58] wrote on the self, but perhaps the most influential was James's [27] distinction between the *me self* and the *I self*. James's *me self* describes the parts of a person that are objects within the world. By contrast, the *I self* is to James not the objects experienced, but the *experiencer*. The *me self* is now often referred to in psychology literature as the *explicit* or *objective* self, or as the *self-as-object*. Conversely, the *implicit* or *subjective* self, sometimes also referred to as the *self-as-subject*, is the self which is the subject of experiences. While we will present a more comprehensive review of work that explores self-awareness in a computational context later in this section, it is helpful to highlight a few links here as motivation, as we introduce the foundational concepts. For example, as argued by some who have considered computational forms of self-awareness (e.g., Lewis et al. [34]), both self-as-subject and self-as-object are important concepts to include in a consideration of how computing systems may be made more self-aware.

3.2.1.1 Reflective Self-awareness

We have found that many researchers who are unfamiliar with the notion of self-awareness find that the *explicit* self, or self-as-object lends itself to a reasonably intuitive interpretation in a computational setting. This form of self-awareness is concerned with things that "belong" to the system itself, objects that comprise the *me self*. These can be learnt about and modeled, becoming concepts in the system's knowledge. This process is often referred to as *reflection*, and when language is used as the modeling tool, reflection can also be referred to as *self-reference* [45]. The aspects of the self in explicit reflective self-awareness are things that can therefore be recognized, monitored, modeled, and reasoned about, including in relation to other objects in the world.

In addition to self-as-object, we may also consider an individual that is aware of its own thinking and experiencing of the world, i.e., its *self-as-subject*. This is also a form of reflection, in which a system has the capability to monitor and conceptualize its own (mental) processes. For example, I may be aware of how much I have been thinking about my parents recently, or in a computational setting, a system may be aware of the algorithm being used to perform a particular interaction with the world. In this way, aspects such as the algorithm's efficiency, memory usage, or time-to-completion can be modeled, and the algorithm be replaced with an alternative version, if required.

This form of self-awareness is powerful, since it permits recursion: the processes of which the system is aware may be its own self-awareness processes. This is the form of self-awareness discussed in much of the literature on meta-reasoning or meta-cognition, which overlap significantly with, and rely on the notion of meta-self-awareness introduced in this chapter.

3.2.1.2 Pre-reflective Self-awareness

The notion of self-as-subject also draws attention to the idea that a system making observations and taking actions in its environment can be thought of as the subject of those experiences and that, despite any similarity to other systems, those experiences are unique to that system. Observations (through sensors, for example) are necessarily from that system's own point of view, and as such are not only influenced, but determined by factors such as their sensing apparatus, their situation within the world, and their own state. This explicit acknowledgement of the perspective from which (self-)knowledge exists is a core theme running through self-awareness literature, highlighted for example by Metzinger [39], and by Newen and Vogeley [46], and acknowledges a *pre-reflective* component in self-awareness.

The assumption that subjectivity is associated with a system's experience of its world underlies the elicitation of reflections of both self-as-subject and self-as-object. We might call this subjectively known self the *self-as-experienced*. In practical terms, this provides one of the key differences between context-aware systems (e.g., [10, 15]) and self-aware systems. The former typically assumes a ground-truth-based environmental context, which holds as true for all entities. Systems can learn more or less complete and accurate models of the context, which are then used to inform things such as actions. In a self-aware system, by contrast, it is explicitly acknowledged that systems will have different experiences of a shared environment (which includes themselves), since they observe things from different perspectives, using different apparatus.

This form of self-awareness is not reflection but is *pre-reflective*. It underlies an individual's ability to develop a reflective self-awareness, in either self-as-object or self-as-subject form. Indeed, the presence of this implicit subjectivity does not require that any reflection takes place afterwards: self-awareness may simply stop at this *pre-reflective* stage.

3.2.1.3 Notions of Self and Collectivity

While self-awareness is most commonly and perhaps most intuitively thought of as a property of a single "mind", self-awareness can also be a property of a collective system. What, in any case, is the self that possesses the self-awareness?

In recent years, we have come to develop an understanding of the role of collectivity and emergence in complex systems. Rather than require the existence of a CPU-like entity within the brain, interactions between many distributed components can give rise to a range of complex mind-like entities, which have no single central physical presence. Mitchell [41] provides some excellent examples of apparent self-awareness in collective systems, highlighting the absence of a single controlling mind-like thing in each case. In addition to the human brain, she explores the human immune system and ant colonies, where self-awareness appears at the level of the collective, even though this property may not be present at the level of the individual components.

Mitchell describes this form of *collective self-awareness* as one in which information about the global state of the system is distributed throughout the components, and builds up statistically, in a bottom-up manner, through interactions. Despite its distributed nature, this information feeds back to enable the control of the system's lower level components. The right information ends up where it needs to be, and the system achieves a form of parallelized self-awareness, implemented across decentralized hardware. Furthermore, when viewed from the outside, the entire system appears to have a sense of its own state that is both "coherent and useful" [41].

When we talk of self-aware computing systems, we may therefore be referring to several different types of self. Firstly, it should by now be clear that self-awareness may be a property of an autonomous agent, which is capable of obtaining and representing knowledge concerning itself and its experiences. Indeed, much of the literature on autonomous and intelligent agents is concerned with techniques for agent learning, knowledge acquisition and representation, and architectures to support these capabilities. Secondly, according to this notion of collective self-awareness, self-awareness may be present at the level of a collective system. The boundary of a self-aware entity, indeed the "self", is therefore not limited to encapsulating a single agent with a central knowledge-gathering and decision-making process.

3.2.1.4 Notions of Levels of Self-awareness

In Sect. 3.2.1, we established a distinction in *levels of self-awareness* between pre-reflective and reflective self-awarenesses, and within reflective self-awareness, between reflections on the self-as-subject and self-as-object. Morin [42] dives further into a stratified explanation of self-awareness, reviewing various sets of more fine-grained levels of self-awareness in the psychology literature.

Of these, Neisser's [44] have received the most attention to date in computational treatments of self-awareness. His five levels range from basic awareness of environmental stimuli through awareness of interactions and time, up to awareness of one's own thoughts and one's own self as a concept. Unusually, among the sets of levels presented in psychology, Neisser's model also includes "lower-level" aspects of implicit self-awareness.

Rochat takes a developmental angle on the question of levels of self-awareness, asking how self-awareness competences develop in human children. His experiments take inspiration from and extend the well-known classical "test" for self-awareness, the *mirror test* [19]. While the mirror test is flawed as a method for establishing the presence of some supposed binary notion of self-awareness [21],[1] Rochat makes good use of observations of the behaviour of children in front of a mirror, to draw more general conclusions.

Newen and Vogeley [46] argue that sets of levels such as those discussed so far, while helpful, are based upon a more fundamental set of self-awareness forms. They draw attention to different levels of complexity associated with human self-awareness or self-consciousness, derived from developmental and linguistic psychology. They argue that there are at least five of these levels, describing increasingly complex forms of first-person representations, from nonconceptual representations, through conceptual, sentential, meta, and iterative meta-representations.

While there are important differences in both emphasis and content in different sets of self-awareness levels, there are also key themes. Acknowledging this, Morin [42] attempts an integration of these and other sets of levels, into broad themes reflective of the overarching levels presented earlier. Morin labels these *consciousness*, *self-awareness* and *meta-self-awareness*.

3.2.1.5 Agency, Ownership and Models of a Minimal Self

It is hopefully clear by now that when considering the self-awareness properties of an individual, rather than thinking of self-awareness as a binary property that can be present or not, there is a wide range of notions to consider. It is then reasonable to ask what is the minimal requirement for self-awareness?

Examples of such explorations include Gallagher's [18] *minimal self* and Dennetts *narrative self*. These highlight that even in systems without agency, we can begin to consider the self-awareness question.

In practice, for the design of self-aware computing systems, either strong single-entity-based or emergent approaches to providing a unity and consistency of self

---

[1] As discussed earlier, the binary notion of self-awareness is in any case a mischaracterisation. Further, Haikonen [21] showed that very little sophistication can enable a computational system to "pass the test".

could be beneficial, though we believe that at this early stage in research into self-aware computing, it is important not to limit subsequent thinking to either one.

A synthesising role for self-awareness able to integrate implicit and explicit sources of (self-)knowledge now comes into play. In general, a system may (a) intend to take actions and fail to realize them, (b) intend to take actions and realize them, or (c) not intend to take actions, and involuntarily take them. Self-awareness levels that include models of causality (i.e., forms of interaction awareness) should account for these different cases, as the distinctions between them will be as relevant for computers as they are for humans.

3.2.2 Interpretations and Applications

While self-awareness has been studied extensively in natural systems, including in psychology, philosophy, and sociology, it has also more recently been explicitly considered in computational systems. Lewis et al. [35] provided an early survey on self-awareness and its application in computing systems. Recent books [37, 51, 62] also provide excellent overviews of a range of work incorporating the self-awareness concepts into computer science and engineering, and a recent special issue of *Computer* [59] highlights the state of the art in various self-awareness-related efforts.

In this book, Chap. 2 gave an introduction to the broad range of efforts in computing which relate to self-awareness. In this section, however, we expand on Chap. 2, focussing on efforts in computer science and engineering which directly and explicitly attempt to translate concepts from self-awareness in natural systems to computing. This is not a full survey; however, we aim to provide an overview of how some of the key areas where self-awareness has been explicitly interpreted and applied in computer science and engineering.

3.2.2.1 Self-awareness and Meta-cognition

Some of the earliest literature that began to explicitly address self-awareness in computing, emerged around 2004. A DARPA workshop on “Self-Aware Computer Systems” [4] drew together contributions from a range of researchers, many of whom provided position papers proposing key challenges to move beyond the initial stages of understanding about what self-aware computing might come to mean.

Other works to follow soon after would consider those aspects of self-awareness that form part of a *meta-cognitive loop* [5], where the key challenges are associated with knowledge representation and logical reasoning to provide self-awareness. Schubert [56], in expectation that self-awareness will “push the AI envelope”, proposes requirements for both knowledge representation and reasoning to support explicit self-awareness. Meanwhile, Cox [14] highlights that such a feedback loop is inherent to self-awareness, arguing that being aware of oneself is not merely about possessing information. He argues that self-awareness includes the ability to use that information in order to generate goals that may in turn lead to the information being modified. Both Schubert [56] and Cox [13] argue that a key expected benefit of self-awareness is the possibility of resultant self-explanation, the ability of a system to draw on its self-awareness in order to explain or justify itself to an external entity such as a human. It is easy to see how in the face of ever more complex *black box* AI, e.g., as provided by many machine learning techniques, comprehensibility would be facilitated by this approach.

Providing helpful intuition, Cox also suggests [14] that meta-cognition is similar to the algorithm selection problem, wherein the task is to choose the most efficient algorithm from a set of possibilities. This notion of an ability to select one's own method of collection and processing of information, according to goals which may themselves be modified by the individual, has much in common with the conceptual self described by Neisser and discussed in Sect. 3.2.1.4. Cox further considers the differences between cognition and meta-cognition and argues, in line with Maes [38], that a meta-cognitive system is one whose domain is itself. Such a meta-cognitive system can therefore reason about its knowledge, beliefs, and own reasoning process, as opposed to merely *using* knowledge about itself.

3.2.2.2 Computational Self-awareness

Lewis et al. [34] and Faniyi et al. [17] provide an attempt at characterising the self-awareness in computational systems, in a framework inspired by and translated from psychology. They propose the notion of *computational self-awareness*, arguing that human self-awareness can serve as a source of inspiration for equivalent concepts in computing. They describe a taxonomy of forms of computational self-awareness, based on (i) notions of *public* versus *private* self-awareness, (ii) *levels of computational self-awareness*, and (iii) collectivity in and emergence of self-awareness. Their levels of computational self-awareness [17, 34] are inspired by Neisser's levels for humans, introduced in Sect. 3.2.1.4 above, but translated appropriately for describing the capabilities of computer systems. By translating the concepts such as this to the computing domain, it is argued that designers are then able to adopt a common language in considering the various self-awareness capabilities that their systems may or may not possess. While "full-stack" computational self-awareness may often be beneficial, with several processes responsible for one or more levels of self-awareness, there are also cases where a more minimal approach is appropriate.

In a recent book, Lewis et al. [37] present their framework for computational self-awareness more fully. The book further describes a set of derived architectural patterns and a collection of engineering case studies, where explicit consideration of the patterns has been beneficial.

Several engineering efforts have built on Lewis et al.'s [34] conceptual framework and reference architecture for computational self-awareness, including notably as part of the EU-funded EPiCS Project.[2] In one line of work, self-awareness has been used to improve user experience in interactive music systems [49]. As demonstrated through case studies in rhythm [47], chord progression [48], and the sharing of solos around a music group [11], self-awareness enables musical devices to provide varying degrees of control to the user, based on their behaviour and preferences at run-time.

Meanwhile, a work on heterogeneous multi-core nodes [3] illustrates how self-awareness can support the dynamic vertical function migration between the hardware and software at run-time. At the network level, an FPGA-based self-aware network node architecture [28] supports the autonomous configuration of dynamic network protocol stacks, reducing the communication overhead in terms of sent packets, when compared to static stacks. Incorporating the self-awareness in smart camera networks [16, 36] has facilitated the move from static design-time network calibration to run-time management of the network, enabling adaptation to changing and unforeseen deployment conditions. These systems share the characteristics of being large, decentralized, dynamic, uncertain, and heterogeneous and have benefited from the explicit consideration of self-awareness properties and additional online learning compared to "classic" designs.

3.2.2.3 Reflective Architectures

While many of the examples discussed in the self-aware computing literature include algorithms inspired by the biological systems (e.g., evolutionary algorithms, reinforcement learning, neural networks), approaches to implementing architectures for reflection have also been based on inspiration from biology. Indeed, such systems have many characteristics that support their rich reflective processing [7, 9, 42, 50]. The multilayered architectures of biological systems and the biological style of using both opposing processes and the combination of global and local processing for control lead to two main results in terms of reflection: First, there are a variety of direct and indirect sources of information and control available for reflection and adaptation. Second, these direct and indirect control points can be used as "entry points" for monitoring (instrumentation) and reasoning about the data collected and adjusting effects (reflective processes). Bellman, Landauer and colleagues have explored this over several years (e.g., [9, 32, 33]).

The *Wrappings* approach is one way to implement computational reflection and self-modeling systems. In continuous development since 1988, the Wrappings approach grew out of work in conceptual design environments for space systems which had hundreds of models and computational components [8].

---

[2] http://www.epics-project.eu/.

3.2.2.4 Self-awareness in Computer Engineering

While meta-cognition, supported by meta-self-awareness, is concerned with higher order reasoning abilities and of particular interest to the artificial intelligence community, at a more practical level, efforts exist to engineer systems which explicitly consider knowledge about themselves. Project Angstrom [1, 2] is one such effort, where Agarwal and others propose a new way of approaching the system design. The core idea is to move from a procedural design methodology wherein the behaviour of the computing system is pre-programmed or considered at design time, towards a self-aware system where the system adapts to its context at run time. As a result, many decisions can be delayed from design time at the expense of providing the system with the possibility of taking these decisions during run-time. For example, this avoids the need to consider what resources will be available to the system once it is operational. Instead, the system discovers resources and makes decisions about how to allocate them during its operation.

The intention is that this will lead to reduced programming effort, since if a system can automatically discover how to meet its goals at run-time, based on what it finds available, then designers are no longer required to determine how to satisfy resource constraints themselves. In describing this vision, Agarwal [2] proposes five properties that self-aware computers should possess:

- Introspective: they can observe and optimize their own behaviour,
- Adaptive: they can adapt to changing needs of applications running on them,
- Self-healing: they can take corrective action if faults appear whilst monitoring resources,
- Goal-oriented: they attempt to meet user application goals,
- Approximate: they can automatically choose the level of precision needed for a task to be accomplished.

More recently, Hoffman [23], Santambrogio [54], and others have extended Agarwal's work, developing systems that, based on an Observe-Decide-Act (ODA) loop, seek to automatically adapt to meet high-level goals online. The monitoring aspects of self-awareness are facilitated by technology called *application heartbeats* [24]. The aim is to define a general method for monitoring the behaviour of an application against high-level goals. Typically, machine learning techniques are then used to adapt, in order to continue satisfying the goals [25, 40].

3.2.2.5 Self-awareness in Complex IT Systems

With a focus on the engineering of IT systems and services, Kounev [31] proposes self-awareness as an extension to the autonomic computing architecture, in which systems possess built-in self-models. Such models include aspects such as the system's own architecture, and its interactions with its environment, enabling these things to be reasoned about at run-time. Kounev et al. further argue [30] that self-reflection (awareness of hardware and software infrastructure, execution environment, and operational goals), self-prediction (the ability to predict the effects of environmental changes and of actions) and self-adaptation are key characteristics of self-aware systems. Self-awareness itself is then considered from a fundamentally pragmatic perspective, as concerning the nature of (self-)models required for more effective predictive and adaptive behaviour. In taking this approach in software and systems engineering, it is argued that quality of service requirements can be met despite changes in the environment. A challenge is highlighted around the need for systematic engineering methodologies for self-aware systems.

3.2.2.6 Self-awareness in Collective Systems

Zambonelli et al. [63] consider the self-awareness properties of collective systems they term *ensembles*. An example of such an ensemble is a robot swarm. In their work, Zambonelli et al. consider self-awareness to be the ability to recognize the situations of their current operational context that require self-adaptive actions. Hence, self-awareness is here closely tied to knowledge supporting the need for resulting actions. In particular, they envisage the utility of self-awareness to be targeted towards radical run-time modification of structure, at both the individual and ensemble level. In order to reason about this, they highlight that self-awareness is not only concerned with what is currently happening (in terms of oneself and the state of the world), but also what could happen. This includes what the individual or collective could become by adapting, how the world could change, or how these things may affect each other. Hence, the notions of *time awareness*, as will be discussed in Sect. 3.3.2, are important, motivating the development of models that have a predictive or anticipatory power, and those that capture causality in complex collective systems.

This line of research was pursued in the EU-funded ASCENS[3] project, primarily through the application of formal methods to reason about the knowledge encapsulated in ensemble-based systems. In particular, the *general ensemble model (GEM)*, is proposed [26], as a common integrated system model for describing components and their interactions in mathematical terms. A recent book [62] discusses progress made during the ASCENS project.

Lewis et al. [34] also explicitly discuss systems with collective self-awareness. Building on Mitchell [41], they describe components *"within a collective that interact with each other locally as part of a bigger system"*, which *"might not individually possess knowledge about the system as a whole. Although global knowledge is distributed, each system within the collective can work with other systems, giving rise to the collective itself obtaining a sense of its own state and thus being self-aware at one or more of the five self-awareness levels."*

---

[3] http://ascens-ist.eu/.

3.2.2.7 Systems with Minimal Self-awareness

One example of where self-awareness has been applied in a small somewhat simple system is in *cognitive radio*, where devices control their own capabilities and communicate with other devices to monitor theirs. This enables them to improve the efficiency of communication by negotiating changes in parameter settings [60]. Another area where a comparatively simple self-awareness loop is used to great practical effect, is in cognitive packet networks (CPNs) [52]. Here, nodes on a network are able to monitor the effect of choosing different routes through the network, enabling them learn during run-time to adapt the route between a source and a destination on an ongoing basis. The approach is able to naturally deal with changing quality of service requirements of different nodes and has been demonstrated to be highly resilient to denial of service attacks [20].

3.2.2.8 Self-awareness in Robotics

In robotics, research is also concerned with replicating forms of self-awareness that *appear to be human*, such as the robot, *Nico* [22], which learns about its own body, deliberately attempting to mimic an infant. Meanwhile at the collective level, Schmickl et al. [55] showed that a group of robots with simple behavioural rules and local interactions may achieve collective awareness of a global state, distributed across the individual units.

Winfield has highlighted a crucial role for self-awareness in future robotics. He proposes [61] that many robots operating in the world should possess internal self-models, arguing that such models are essential for safe and ethical robot behaviour. The key idea is that by constructing and evaluating a model of the robot and its surroundings, unsafe and unethical outcomes can be avoided. A consequence evaluator predicts the future events through the use of self-models, and as a result moderates the actual robot controller.

3.2.2.9 Self-awareness Elsewhere and in Future Systems

In this section, we have provided an overview of the major research efforts over the last decade or so, in self-aware computing. However, there are many other areas of computer science and engineering research, where self-awareness is mentioned. Two notable ones are autonomic computing [29] and organic computing [43]. Both of these visions have spawned rich literatures, and self-awareness is frequently mentioned as a desirable property. However, in neither case has the notion been further developed. Similarly, in the self-adaptive software community, the so-called self-* properties are organized in levels where *self-adaptiveness* is emphasized as being the most general level, and self-awareness and context-awareness are considered only as primitive levels [53]. Here too, concepts of self-awareness have not been explicitly explored.

Finally, it is clear that there are many computer systems available today, and dating way back before the previous decade, that exhibit the sorts of self-awareness capabilities that we describe in this chapter. These systems, along with autonomic and organic ones, also have the potential to offer crucial insights into the computational self-awareness. In the vast majority of cases, however, there has simply not yet been any explicit engagement with self-awareness concepts. In the report on the 2004 DARPA workshop on self-aware computing, Amir et al. noted [4] that there were even then already systems in existence that exhibited what they called "features of self-aware computer systems." On the evidence presented in this and the previous chapters, we agree. It is our hope that by providing a principled and clear conceptual framework for the description of different self-awareness types, it will become easier for others to engage in discussing and reasoning about the self-awareness properties of their own systems, and how they might be extended.

3.3 A Conceptual Framework

In this section, we present an inclusive conceptual framework for describing computational self-awareness. Taking inspiration from the varieties of self-awareness present in human and animal systems, we outline a structured set of technical concepts, which describe the types of capabilities that self-aware computing systems might possess, and how they might be present in such systems. As such, capabilities described using our framework might form building blocks for self-aware system architectures, which will be elaborated on in Chaps. 6 and 7. We also provide axes on which computational self-awareness may vary, and may indeed be quantitatively or qualitatively compared, in anticipation of Chap. 15, where this topic is revisited.

The framework described here is inspired by, and builds on, the conceptual framework presented by Lewis et al. [34]. In building on that prior work, we highlight two important differences. First, this framework takes a more inclusive and extensible approach, where levels of self-awareness are not directly mapped to psychological levels (as was done with Neisser's levels by Lewis et al.), and thus may be adapted to suit a particular application context. Second, we include an explicit treatment of the concepts of pre-reflective, reflective and meta-reflective self-awareness processes, and how they relate to one another. In this section, we do not however yet discuss techniques or approaches to realising the concepts introduced, except by means of illustrative example. Therefore, our framework aims to serve as a sort of space of possible conceptual requirements of a computationally self-aware system.

Our proposed framework is based on the following high-level concepts:

- Levels of self-awareness,
- Aspects of reflective self-awareness, and
- The domain of self-awareness.

These are elaborated upon below, and can together be used to characterize the form of self-awareness processes that a system might possess. In general, self-aware computing systems will possess many such processes, which interact, for example in reflective relationships, and by acting on one another. We distinguish three overarching levels of self-awareness: *pre-reflective*, *reflective*, and *meta-reflective*. We then characterize various aspects of reflective models, which may be present in a self-aware system. These include time, interactions, identity, goals, etc. The *domain* of self-awareness includes the *span* of self-awareness (that is the subject of the self-awareness, i.e., who is self-aware in this case), as well as the *scope* of self-awareness (the objects of the systems sensing and reflection).

3.3.1 Overarching Levels of Self-awareness

As discussed in Sect. 3.1, an important concept in self-awareness is the existence of different levels of self-awareness. These range from basic subjective awareness of stimuli, from both the system itself and its environment, through an awareness of how knowledge can represent concepts such as social interaction, causality, and time, up to an awareness of one's own thoughts. Advanced organisms engage in meta-self-awareness [42], an awareness that they themselves are self-aware. Meta-self-awareness may also take many forms, including awareness of self-awareness processes, awareness of reflective knowledge, and iterative meta-reflection on meta-self-aware processes and meta-representational knowledge [46]. Computational self-aware systems will also similarly vary a great deal in their complexity, and levels of computational self-awareness are also needed to express this variety.

Taking a coarse grained view, we can distinguish three overarching levels of self-awareness: pre-reflective self-awareness, reflective self-awareness, and meta-reflective self-awareness. These are elaborated upon below.

3.3.1.1 Pre-reflective Self-awareness

Pre-reflective self-awareness is the subjective awareness a system has from its sensory input - its ability to perceive and make observations. This includes environmental stimuli, internal stimuli from the system itself, the sensing of carrying out an action, receiving signals from the outside world, etc. This level aligns with Lewis et al.'s [34] *stimulus awareness*, based on Neissers ecological self, and describes the presence of subjective knowledge being obtained by the system before (or perhaps without) any reflective modeling of this knowledge takes place.

At this level, the system has no knowledge of historical data, and no models of interaction or causality. In terms of actions possible by a system with only pre-reflective self-awareness, these are limited to stimulus-action rules, since any reasoning (if one can still call it that) is limited to that based on perception, rather than any pre-existing internal models (since none exist). As such the system is able to adapt only to immediate observations. Indeed, at this level, the system has no conceptualisation of its experiences, it merely senses for itself, and possibly acts.

Pre-reflective self-awareness describes the minimal form of computational self-awareness. While in computing systems, it describes nothing beyond a typical system able to interact with its environment, it is a prerequisite for all later forms of self-awareness. A system cannot learn a reflective model, conceptualize its experiences, reason, and act based on them, if there is no experience present in the first place, represented in the system. From a pragmatic point of view, the inclusion of pre-reflective self-awareness in our taxonomy of computational self-awareness, invokes an important question: for a given application, is pre-reflective self-awareness sufficient to achieve the desired behavior? Or, are reflective models (and possibly more) also needed? Thus, the explicit consideration of pre-reflective self-awareness enables the engineers to justify the use of more advanced forms of self-awareness for their systems.

Pre-reflective self-awareness is however more than simply *data*: it is data that has been sensed, that has come from somewhere according to the ability of the system to obtain it (e.g., through physical sensors). Therefore, this level of self-awareness also implicitly includes the notion of subjectivity. Two different systems with different sensors are in different locations would sense the same phenomenon differently, and thus end up with *different data*. In formalising any notion of pre-reflective self-awareness, it is crucial to include the subjective source of the data itself. In turn, this means that this “meta”-data, if we should indeed call it that, can be used in the reflective processes, elsewhere in the system. Ultimately, this may lead to radically different outcomes in terms of knowledge or action.

3.3.1.2 Reflective Self-awareness

Reflective self-awareness is the process of producing a conceptualisation (i.e., a model) of a one’s knowledge and experiences. In the first instance, the reflective model will be of a pre-reflective self-aware experience, i.e., a reflective model of a sensory input, observation etc., as described in Sect. 3.3.1.1. In other words, as the system goes through life, experiencing its environment through its sensors, reflective self-awareness processes build models of the empirical and subjective data gathered by these sensors, as well as (potentially) models of the subjectivity with which they were obtained. We call the specific observation (or pre-reflective representation of a phenomenon) captured by the modeling process the *object* of the reflective self-awareness process.

As with human self-awareness, such models or conceptualisations can also capture various different aspects of an experience (of an object). For example, a reflective self-awareness process may build a model that conceptualizes an experience over time. Alternatively, it may build a model that conceptualizes the causality present in an experience, for example, based on modeling how the experienced environment responded to an action. In general, there are many such *aspects* to reflective self-awareness; these will be discussed in Sect. 3.3.2. In our framework, we may therefore describe a reflective self-awareness process according to its *aspect(s)* and *object*.

3.3.1.3 Meta-Self-Awareness

Meta-reflective self-awareness, or meta-self-awareness is essentially reflective self-awareness where the object of the reflection is a reflective self-awareness process. Hence, we may describe the meta-reflective processes in the same ways that we describe the reflective processes (since they are a subset of them). It is important to note, however, that there are at least two high level classes of object that a meta-reflective process could be concerned with: the underlying *process* that forms the object of the meta-reflection (e.g., a learning process), and/or the *output* of the underlying process, (i.e., the models produced by the process being reflected upon). In practice, both are likely to be valuable, and indeed integrating reflection upon both process and output will provide the necessary knowledge to be able to reason about how the two relate to each other. As an example, a meta-reflective process could model the behavior (e.g., memory usage) of another learning process, while also judging the fidelity of its learnt models. Thus, the meta-reflective process has the ability to reason about when it might be advantageous to switch to a different learning method that better balances this trade-off. In general, there are many types of processes that could be learnt and reasoned about, such as reasoning processes, decision processes, problem-solving approaches, and other meta-reflective processes.

There is a strong link between this level and the meta-cognition or meta-reasoning literature discussed in Sect. 3.2.2.1. Indeed, our distinction above echoes Cox's [14] and Maes's [38] arguments that the difference between cognition and meta-cognition is that a meta-cognitive system is one whose domain is itself. Hence, a meta-cognitive system can reason about its knowledge, beliefs, and reasoning processes, as opposed to merely using knowledge about itself as sensed.

In practical terms, our experience with meta-reflection has so far focussed these types of meta-self-awareness processes on resource management, trade-off management, and integration in complex systems. However, while meta-reflection is certainly useful in these regards, we expect there to be additional areas where meta-reflection can help. Examples might include in building higher-order models of causality, or in reflecting on the self-awareness of others in a social system, or on the emergent collective self-awareness of a group of which the individual is a part. However, there will be many additional challenges when we move beyond our current emphasis on resource and trade-off management, and system integration. One challenge is that part of the reason for meta-self-awareness is to gain a perspective on the overall goals for the whole system (much like the role of a central nervous system). One new type of role for this meta-self-awareness would be to decide what type of decision processes should be in control when there are several alternatives possible, or indeed simultaneously running in the system. This type of meta-self-aware reasoning would have to deal with the role of attention in addition to run-time trade-offs. This brings up more new challenges in terms of what actions and what types of information will be needed for such processes and also how to become increasingly liberated from a particular domain and set of experiences to something that begins to look more like general intelligence. This of course greatly changes the nature of the self-models and the models of the situation.

For the avoidance of doubt, it is worth emphasising that given that the object of a meta-reflective process is a reflective process, and meta-reflective processes are themselves reflective processes, then this permits the iterative meta-reflection. This layering permits arbitrarily complex internal reasoning about the behavior of different self-awareness processes with respect to their aspects, goals, expectations, conceptualisation, etc., and also with respect to the systems continuing pre-reflective perceptions (i.e., empirical data taken from observations in the world).

3.3.2 Aspects of Reflective and Meta-reflective Self-awareness

We have so far introduced three overarching levels of self-awareness: pre-reflective, reflective and meta-reflective self-awareness. We further discussed that (meta-) reflective processes, those that build conceptual models of their objects, might capture different *aspects* of the object, or the experience of the object, in their modeling. In this section, we discuss *some* of these aspects of *reflective* self-awareness processes, and how they may combine and build upon each other to form complex self-aware behavior.

In much of the psychological literature, the aspects discussed here are often described as part of a more fine-grained set of *levels* of self-awareness. In this chapter, we have restricted the term "levels" to refer to capabilities and processes that build upon each other, for example as discussed in Sect. 3.3.1. As this section highlights, reflective self-awareness processes in a system can and typically will comprise a rich variety of learning, reasoning and acting behaviors, each of which models or focuses on a particular aspect of self-awareness. We now proceed to describe those aspects of reflective self-awareness that we at present identify as important to self-aware computing systems.

3.3.2.1 Identity Awareness

Identity awareness in a reflective process concerns the modeling of experiences such that they contribute towards a conceptualisation of a coherent identity, possibly over time. This describes the ability to recognize and model the identity of entities, such as other systems, objects in the world, and humans. Without identity awareness, experiences form isolated stimuli, as part of a noisy "soup" of an environment. Identities as modeled may be unique, such as the identities of specific other systems with which to interact (e.g., through unique IDs, such as MAC addresses or DOIs), or simply by affixing one's own label to an encountered object to track it over time (e.g., the third client I saw join the network). Alternatively, identities may be modeled at the level of roles (e.g., a web server, a small robot in my environment). Ultimately, the level of expressiveness and uniqueness required of identities in a reflective model will depend on the application requirements and context. Finally, identity awareness also extends to oneself. Neisser's highest level of self-awareness, the self-concept [44], concerns the conceptualisation of the whole self as a unified entity within the world. An awareness of the identity of oneself (or one of its many identities) is a prerequisite for self-concept. Extending this notion of self-identity to the external sphere, identity awareness also includes the concepts such as one's own role(s) within a wider system.

3.3.2.2 State Awareness

State awareness builds upon the knowledge captured in any and all other aspects of self-awareness, by providing the ability to model and recognize the identity of states of oneself, the world or other entities within it. For example, a system may have a variety of knowledge present in its various conceptual models, associated with various other aspects. With the addition of state awareness, the system would then be able to use this knowledge to characterize the state of the world. Examples might include "winter," "in low power mode," or "waiting for a reply," although there is no requirement for states to have intuitive names or meanings outside of the individual itself. Indeed, unsupervised learning presents a promising technique for the realisation of state awareness, since experiences might be clustered to identify particular emergent states.

3.3.2.3 Time Awareness

Time awareness describes the aspect of reflective models that are concerned with historical knowledge, or knowledge of potential future phenomena. In its simplest form, this may include knowledge of past or potential future basic stimuli. Temporal aspects of models may of course be expressed in a wide range of ways, from precise associations (e.g., timestamping) to ordinal relationship information (e.g., x happened before y). When combined with other conceptual aspects, awareness of the temporal nature of these may be also be modeled. Examples of these may include historic interactions, future states or previous identities.

3.3.2.4 Interaction Awareness

In *interaction awareness*, run-time models are used to take into account patterns of interactions between entities. There are various sub-aspects here, which build on each other. Most obviously, the system must be able to recognize that some actions form part of interactions, such that they are in some way causally connected. An example of this includes message passing, such as is used in a communications protocol, where one message may be a response to another, rather than an isolated action. There may be, in simple interaction awareness, simply a model of the flow of actions over time (e.g., action b typically occurs after action a), or there may be additional semantics associated with the actions or the combination of them (e.g., actions b is a response and action a is a query).

As a prerequisite to the above form of interaction awareness, there must be some form of identity awareness, at least insofar that the system can identify messages, as apart from the general noise in the environment. It may also be important to be able to identify individuals as those engaged in an interaction, if not in terms of their unique identity, then perhaps in terms of role.

Interaction awareness can also build upon state awareness, since models may encapsulate knowledge of causality such as "when action a is taken in state s, this leads to state t." Markovian approaches may be effective choices for modeling state-based interaction awareness.

Finally, the interactions need not be external. Causality of internal processes may also be modeled in meta-reflective processes. For example, a system might model the behavior of one of its own decision-making processes, when other parts of the system are either operational or not. In the former case, the system may learn a model of how decision-making degrades (due to more stringent resource constraints), when load elsewhere is high. Such models of internal causality may then be used to provide adaptive internal re-architecting, or perhaps more effective scheduling of tasks.

3.3.2.5 Behavior Awareness

In *behavior awareness*, run-time models represent the internal behavior of the system or behavior of external entities. Behavior here is taken to mean an action, or more normally a group of actions, taken by an individual. Models of behavior may, at the simplest end of the spectrum, comprise a representation of an observed action. More useful forms of behavior awareness, however, will link these representations together, over time, and with awareness of states. For behavior awareness to be particularly meaningful, it would usually be coupled with some identity awareness, i.e., knowledge of the identities of individuals or roles carrying out the behavior, or subject to it. Behavior awareness can apply also to oneself, by modeling either ones externally facing behavior in the world or one's own internally facing behavior. Behavior awareness can be applied in a meta-reflective context, where models are built that describe self-awareness processes in terms of their behavior.

3.3.2.6 Appearance Awareness

In human self-awareness, one's *public self*, the image an individual presents to the world, is often considered important. This may also be true of computational systems, for example where the environment or individuals within it might respond differently depending on the appearance of the individual to which they are responding. Thus, *appearance awareness* is concerned with how the individual appears, or may appear, to others in the environment. At one end of the spectrum, this might simply concern an awareness of physical properties (i.e., a robot may be aware of its height, or if it is dirty). At the other end, a system may learn models of how it presents itself, including its own knowledge and self-awareness. In this way, appearance awareness is linked to notions such as self-explanation, acknowledging that a system's explanation or justification for its actions form part of the way it presents itself, and may go on to have subsequent implications for how the environment responds to it. The reflective processes that deal with the aspect of appearance awareness are able to explicitly learn and reason about these factors.

3.3.2.7 Goal Awareness

Goal awareness, in a general sense, includes the ability to conceptualize the internal factors that drive the behavior, such as a system's goals, objectives, and constraints. In some cases, goals will be explicitly available and formally specified. Note that this is not the same as a goal being *implicitly* present in the system, due to the advance decisions of designers. Goal awareness implies the presence of goal knowledge at run-time, in such a way that it may be reasoned about at run-time. Further, this presence of explicit goal knowledge at run-time permits the acknowledgement of and adaptation to changes in goals, either internally generated, or due to external forces (e.g., a user's changing needs).

Not all goals will be expressed in this way. In some cases, goals will be implicit in the environment or need to be derived from higher-level more abstract goals (e.g., the goal to replenish energy may be derived from a higher-level goal to survive). In other cases, goals will take the form of motivations: impulses and drives that are not tied to formally specified objectives or goal states. In these cases, goals may be less obvious, and only intentions or even actions will be observable. Thus, learning will need to be employed to model those goals implicitly in the architecture of the system.

Note that the goal awareness may apply to oneself or to others. In the latter case, a system may possess a model of another system's goals, or those of a human or organisation. Further, the goal awareness may be combined with other aspects, such as time awareness, where models of how goals change over time might be one benefit.

3.3.2.8 Belief Awareness

Belief awareness is concerned with, in general, things believed to be true by a system and differs from expectation awareness since beliefs do not, in general, need to capture the notion of time. In this way, belief awareness provides a generic mechanism for reflecting on other aspects of self-awareness, in both oneself and in others, with degrees of uncertainty. In simpler cases, belief awareness models will capture the knowledge that the system itself believes something in terms of one of the aforementioned aspects to be the case (e.g., I believe that I have an expectation that an apple will fall when thrown into the air). This may appear superfluous; however, the added benefit here is that belief awareness provides a *generic mechanism* for arbitrary meta-reflection on one's own awareness. A belief awareness process may be added to reflect on any other process, to express knowledge of the level of belief associated with the model. In terms of arbitrary reflective architectures, processes that are concerned only with belief awareness allow us to add models of uncertainty, as well as sources of knowledge and uncertainty, in arbitrary arrangements.

Belief awareness also applies in the public sphere, since awareness of others' beliefs may also be modeled. This provides the necessary machinery for iterative meta-representational self-awareness, for example cases such as system x believing that system y believes system z to be aware of system x's origin.

3.3.2.9 Expectation Awareness

Expectation awareness combines belief awareness and time awareness, to form models that express what the system or others believe about how the world will unfold over time. This includes, for example, awareness of how the laws of physics act on objects over time. For example, the idea that "when I throw an apple in the air in front of me, I expect that I will see it come down" can be modeled as a belief over the state of an observation of an object over time. A further example is that of knowledge of social conventions (e.g., when a person greets me, he expects me to respond with a greeting within a short time window). In this way, expectations in fact can form part of a requirements model (requirements awareness might be seen as a sub-aspect of expectation awareness), where the expectations are those of a user, client machine, or interaction partner in a collective system.

At the meta level, this may be used to model the requirements or expectations over other aspects. A system may, for example, be watching two individuals engaged in a conversation. In this case, a model may capture the expectations over the sequence of messages being passed in the interaction (see interaction awareness, above): there may be a model that the response b is *expected after* query a. Subsequent reasoning may concern the implications of this expectation being broken and appropriate actions that may be triggered.

3.3.2.10 Applicability and Extensibility

In the above characterisation, we have presented the aspects of reflective self-awareness that we anticipate will be most relevant to self-aware computing systems. However, as in psychology, we expect that various different set of aspects will be used, as appropriate for the particular context or system design process. Therefore, the above list should *not* be considered our proposal for a complete list, but instead as a starting point for further consideration as to the aspects captured in reflective and meta-reflective self-awareness.

3.3.3 Domain of Self-awareness

In this section, we are concerned with two important questions when considering a self-aware computing system: *what is it that is self-aware*, and *of what is it aware?* By now it should be clear that in a given self-aware computing system, there are various entities that are involved, or potentially involved, in a particular aspect of self-awareness that is present. These entities exist at various levels of abstraction and aggregation, and include the system that *is aware*, the parts of that system that are involved in providing the awareness, and the things it is *aware of*. Without limiting this collection of entities, we refer to them together as the *domain* of self-awareness. In this section, we define various concepts, chief among them are the *span* and *scope* of self-awareness, that comprise the domain. These provide a helpful basis for discussions that rely on these notions, in the rest of the book.

3.3.3.1 Scope of Self-awareness

In *reflective self-awareness*, we are primarily concerned with a relationship between the *subject* of the awareness, that is, the entity doing the reflecting, and the *object* of the awareness, that is, the entity being reflected upon. A self-aware system will be aware of a broad range of objects over its lifetime. These include, for example, externally sensed things, the system's sensing apparatus, internal reflective processes, a sense of self-concept, and indeed these things at different points in time, and in other individuals. Collectively, we call the objects of self-awareness that system's *scope of self-awareness*. The scope comprises all the entities observed, or able to be observed, by the system.

3.3.3.2 Span of Self-awareness

In general terms, we refer to the extent of the entity that has the awareness, the entity at which the knowledge is available, as the *span* of self-awareness. We must refine the concept of span somewhat; however, since it is not sufficient to base the definition of span on the notion of an entity that is *doing the reflecting*. This is too ambiguous to be particularly helpful. In general, reflecting on a particular entity will be carried out by one *or more* reflective self-awareness processes within a system, and candidates for being the *span* in this case could be any one of these processes, all of them combined, or the system of which they are part. Coming back to the discussion of what constitutes *a self* (as begun in Sect. 3.2.1), we may accept that all of these interpretations are potentially valid answers to the question: *what is doing the reflecting?*

3.3.3.3 Complexities in Span and Scope

As an example, consider a system that processes sensor data and passes messages to other neighbouring systems. In the system, a number of self-awareness processes reflect on different aspects of what the system's sensors observe. One process that captures a time-awareness aspect may, through its learning, become aware of a pattern in what is being observed, corresponding, albeit noisily, to the time of day. A second self-awareness process does not reflect on time aspects of the observation, but instead provides some interaction awareness. This process develops a model of potential cause and effect, as it becomes aware of a certain type of incoming messages triggering a shift in sensor readings. A third self-awareness process observes outgoing and incoming messages, reflects on the correlations between these, and becomes aware that the type of incoming messages noticed by the second process is received shortly after the system sends a reset signal to its neighbouring systems. It observes that the reset signal is generated once per hour. Finally, a meta-self-awareness process reflects on the knowledge obtained by the second and third self-awareness processes, and is therefore able to develop an awareness that hourly reset messages precede a change in sensor readings.

There are now two *different and coexisting* models that explain the shifts in sensor readings observed by the system. The first, provided by the first self-awareness process, correlates shifts with regular time intervals. The second, provided by the meta-self-awareness process, through its reflection on the knowledge acquired by the second and third processes, associates the timing of the shifts with the system's own reset messages.

It is reasonable to ask at this stage, what is the system aware of, in terms of how its environment changes, and its effect on those changes?

This small example serves to illustrate that the span of reflective self-awareness, the *entity doing the reflecting*, is both several things at once, and specific to the particular reflective self-awareness aspect, and role of that self-awareness process within the wider system.

3.3.3.4 Hierarchical Reflection View of Span and Scope

If we take a view of self-awareness, only as it arises from hierarchical subject-object views of reflection, then we can consider that there might be several different *spans* at work here, each with their own different (but overlapping) *scope*:

- A span comprising reflective process 1, which has the system's observations through its sensors, some aspects of the entities observed, and the time of day, as its scope.
- A span comprising reflective process 2, which similarly has the system's observations through its sensors, some (partially different) aspects of the entities observed, and incoming messages as its scope.
- A span comprising reflective process 3, which has the observation of system's incoming and outgoing messages as its scope.

If we, however, consider a span comprising the meta-self-aware process, then the scope includes the models and learning contained within processes 2 and 3. However, in doing so, the span must be expanded to also include those processes, since without them being part of the *entity doing the reflecting*, there would be no possibility for the sensor observations or messages to be part of the scope of the meta-self-aware process—and they clearly are. Hence, our fourth option is a span comprising self-awareness processes 2 and 3 and the meta-self-aware process.

In this case, we see an overlap between both the span and the scope of the same self-awareness phenomenon. Even in this simple example, as the self-awareness processes are broken down into individual reflective processes, the system, as defined here by the span, is aware of (as defined by its scope) part of its own self-awareness. This is in addition to its awareness of the external things being observed by the sensors.

3.3.3.5 Collectivity in Span and Scope

The one potential span missing from the above discussion, which is perhaps more intuitive, is to consider the system-as-a-whole (as comprising the three reflective, one meta-reflective process and pre-reflective observing apparatus) as a single span. It would follow that the associated scope of such a span consists of everything that is known to it.

Since in the example system there is no single reflective process that has a whole-system view, we must instead return to the notion of collectivity in self-awareness, as first discussed in Sect. 3.2.1.3. The unity of the system in the example, which provides us with the intuition of such a system-as-a-whole span, arises from considering the system as a “well-defined” entity within its broader environment. This is in much the same way that we often consider a human, a dog, a smart phone, or a daffodil as a single system, even though there is no omniscient component within them. Of course, this does not come without the requirement for someone, another observer perhaps, to do the “well defining”.

One way of reasoning about such a well-defined entity is based on behavior. In our example system, and coming back to the fact that there are now two different and coexisting models describing what is occurring, we can of course allow for either or both to trigger observable behaviors. In such a case, we may observe a behavior that suggests that the system has a coherent view integrating both models, even though none is present. Is the “system-as-a-whole” then aware of both models? From this external behaviorist observer, it is the case, and this is clearly sufficient for behavior to be generated based on both hypotheses.

This *collective span* can indeed be a highly useful concept, when concerned with the actions driven by self-awareness. It requires us to accept that a span may not be bound only by the presence of subject-object relations in a reflective hierarchy, but can be enabled through collectivity. In this case, this collectivity arose from a unity of behavior, as observed. A final point to make on the unity of behavior and its reliance on being observed is that there is no requirement for the observer to be an entity apart from the system itself. Indeed, most humans have a sense of their own self, which according to this discussion is of the collective type. This arises from that person's own self-awareness and not (directly at least) the awareness of others.

3.3.3.6 Types of Self-awareness Objects and Scopes

We have indicated above that a system's span, which is the subject of self-awareness, is aware, according to some descriptions based on its levels and aspects, of a variety of objects, collectively termed the system's scope. The systems reflective processes (part of the span) produce various conceptualisations, or models, of internal and/or external entities (part of the scope), which in turn may feature pre-reflective, reflective, or meta-reflective self-awareness.

In general, the object of a reflective self-awareness process is the answer to the question what is the thing the system is paying attention to? A reflective process produces a conceptualisation (i.e., a model) of any pre-reflective or reflective self-awareness. Let us first consider the initial case, where the object is a sensed experience, i.e., an observation. As with model characteristics, the list of possible object types cannot be exhaustive and will to a large degree be domain dependent. However, we provide here a suggested list of important types of objects.

Let us therefore take a look at some of the main *types of objects* that a system can be aware of. From these, we can also define corresponding *scope types*. The list of possible object types cannot be exhaustive, and will, to a large degree, be domain-specific. However, we provide here a suggested list of important types of objects.

Generally, we consider that a system and its environment consist of various entities (or resources) and that the system can be (self-)aware of the existence and various characteristics of these entities. Such characteristics would notably include:

- Property characteristics: include both the parameters that define an entity's characteristics or state, and the actual values of those parameters at different times. Also this includes potential states and their characteristics;
- Observability capabilities: specify the measures that can be taken from an entity, given the available sensing apparatus. These may additionally provide a more detailed description, including recommended observation frequencies, expected resource overheads, and accuracy of measurements;
- Action capabilities: specify the actions that are being or can be performed on an entity; these may additionally provide a more detailed description, including the expected effects within various contexts, quality parameters, and necessary resources;
- Interconnections among objects: represent any kind of relation between objects of any of the types listed above, including links among computing entities (e.g., system architecture); dependencies between state characteristics; and conflicts among action capabilities. At higher levels of abstraction, this may include groups, organizations, or sets of and relations between objects.

We may, as such, derive several types of a system's self-awareness scopes. An *action scope* includes all entities that the system may act upon. These will typically include not only internal entities, but also external entities (if we consider communications with other systems as some sort of actions). We may also find useful the notion of *influence scope* to refer to entities upon which the system may only act indirectly (e.g., via direct communication or indirect effects via a shared environment). The *self-awareness scope* includes all "entities" the system can observe, either directly (e.g., through sensors) or indirectly (e.g., through acquired knowledge obtained from elsewhere). The entities in the scope may include both internal and external entities.

Crucially for all but the simplest of self-aware systems, the entities that are the objects of reflection can themselves be self-awareness processes, including their own acquired knowledge. These observed processes may operate over different scopes and at different resolutions and self-awareness levels, than the reflecting process. We may now use the terminology of the domain of self-awareness to establish that when the scope of self-awareness includes (at least partially) the span of self-awareness, we have meta-self-awareness. Namely, by virtue of its role and connectivity within the concert of self-awareness processes, the subject of a reflective self-aware process is a meta-self-awareness process. In general, we envisage that self-aware systems will have many such meta-self-awareness processes, operating over different objects and with respect to different aspects (goals, interactions, time, etc.). Furthermore, it is likely that reflecting on these further still, meta-meta-self-awareness processes will provide the system with knowledge about the role and impact of operating such an array of self-awareness processes. There are many architectural alternatives for structuring reflection relationships in a self-aware system, as will be explored in Part II of this book. At one extreme however, we may consider a particular type of overarching meta-self-awareness process, able to reflect on the entire system and its self-awareness, providing the system with a complete self-concept.

3.3.3.7 Overlapping Spans and Scopes

In summary, we refer to the subject(s) of reflective self-awareness as the span, comprising all entities that contribute to the formation of self-awareness, and to the object(s) of self-awareness as the scope, comprising all the entities observed by the subject's self-awareness. We refer to the domain of self-awareness as the combination of subjects (span) and objects (scope).

Finally, in practice in many self-aware systems, we find and expect that the span and scope will not be distinct from each other. As in the above example, there may be substantial overlap, through meta-self-awareness. For a fully meta-self-aware system, where every reflective process was itself reflected upon by the system, then the scope necessarily includes the span.

3.3.4 Putting It All Together

In summary, we have in this section sketched a conceptual framework for describing and comparing qualitative aspects of self-awareness in computing systems. Our framework is based on three tenets:

- Levels of self-awareness,
- Aspects of reflective self-awareness,
- The domain of self-awareness.

Using the intersection of all three tenets, we may produce descriptions of the specific self-awareness of a system. For example, we might construct a sentence as follows: Peter (span) is aware of Ada's (scope) goal (aspect) to reduce the power usage (object). Further, Ada (span) is aware of her own reasoning (meta-self-awareness) about what to do (act) about it.

3.4 Self-awareness and Goals

The notion of domain can also be associated with goal definitions. Typically, the goal domain of a self-aware computing system would be included in its self-awareness domain. Exceptions would represent cases where the system achieved its goal without being aware of it and without making use of any of its reflective self-awareness capabilities for this purpose.

A goal's span identifies the entity or entities that are responsible for achieving the goal, from the perspective of the entity that requested the goal. For instance, if a system is required, by an external entity, to achieve a goal, then that system represents the goals span. If a collective of systems is required to achieve a goal, then the entire collective of systems represents the goal's span.

We can also relate a span and scope to the goals of a self-aware computing system where the span refers to the group of entities that is responsible for achieving the goals, while the scope is the group of entities the goal refers to in its specification of what has to be achieved.

Usually, we would expect that if self-awareness is employed to realize a goal the span of the self-awareness must be a subset of the span of the goal (as otherwise the self-awareness would include entities that do not share the related goal and therefore will also not contribute to it) and that the scope of the self-awareness must include the scope of the goal (as otherwise the span is not able to judge whether the goal has been achieved for the scope of the goal).

A goal's scope represents the set of resources over which the constraints defined by the goal should be attained. The goal's achievement can then be evaluated by taking measures from the resources in the goal's scope.

As indicated above, if a system is aware of a goal that it must achieve, then the goal's domain will impact the systems self-awareness domain which is necessary for achieving the goal.

Of course, the considerations discussed here do not apply to cases where the system reaches a goal without being aware of that goal (i.e., lack of goal awareness).

Let us now take a closer look at way in which a goals scope impacts the different types of objects in the targeted systems self-awareness scope, provided that the system is goal-aware. Here, the targeted system is the goals span.

Firstly, the systems self-awareness scope must include the resources (including potentially other systems) upon which the system can act to achieve its goal. These are part of the system's action-awareness scope.

Secondly, the systems' self-awareness scope must include the aspects of the environment (or context) that are relevant to the reasoning process involved in achieving the goal. These aspects are part of the systems context-awareness scope. For instance, the system must be aware of the outside temperature in order to decide whether or not to open the window shutters for achieving an inside temperature.

Thirdly, in some cases, it may be helpful if the system's self-awareness scope included the resources upon which the system could not act directly but could influence by indirect means. These would be part of the system's interaction awareness. For instance, if the smart home aimed to achieve a power prosumption goal—e.g., not consume more than the local production—then the smart home might try to ask a local producer to produce more (rather than asking its own devices to consume less).

3.5 Challenges

In this chapter, we began by reviewing what is understood by self-awareness, both as it pertains to humans, and to computers. We provided a brief overview of relevant work in psychology and computing, on the topic of self-awareness. The main contribution of this chapter is a new conceptual framework for computational self-awareness, extending a characterisation by Lewis et al. [34], in order to (i) provide inclusion and extensibility, (ii) distinguish between reflective levels of self-awareness and aspects of reflective modeling, and (iii) introduce the notion of the domain of self-awareness, in terms of span and scope. In doing so, we provided the language to engage with a broad set of concepts that can be used to describe and reason about self-aware computing systems. The characterisations presented here are concerned with how they might pertain to both current and future computing systems.

There are many challenges that need to be tackled in further developing the notion and practicalities of self-aware computing.

Firstly, the formalisation of the framework sketched in this chapter would provide engineers with the ability to make use of more rigorously processes for the production of self-aware computing systems. Further, such a formalisation could be used by self-aware systems themselves, in order to better reason about their own self-awareness capabilities in a principled and comprehensible way.

Secondly, though we have described a wide range of capabilities, which rely heavily on learning, modeling, instrumentation, and more, we have not here touched on methods or techniques for implementing such capabilities. Some of these are addressed in the later chapters in this book, but there is as yet no fundamental understanding of the linkage between algorithms and self-awareness levels, or if new algorithms are needed in some cases. Such a linkage would again prove valuable to designers of self-aware systems.

Thirdly, how might we structure self-awareness processes with respect to each other, such that the right sort of learning and reasoning is operating on the right internal and external objects (including self-awareness processes themselves)? In other words, what should architectures for self-aware systems look like? Further, how might and should such structures change over time, as the needs of the system change, or more is learnt about the system, its environment, and about what is needed to be learnt?

Finally, there is no assumption that at any given moment, all forms of self-awareness are being used for the same problem or context. Systems may engage multiple parallel self-awareness processes in related or unrelated tasks. How should we organize and manage this? Both architectures and meta-management processes must acknowledge that attention is a limited resource, reflective of the limited computational power, memory, and time for running self-awareness processes on real systems. Another way of phrasing this challenge is to consider how a system itself should decide how to engage and disengage different self-awareness processes dynamically, as relevant or beneficial, given its experience of a changing world.

Acknowledgements Peter Lewis acknowledges the teams of the EPiCS and AWARENESS EU projects, for their contributions to many discussions in the development of the notion of computational self-awareness.

References

1. Anant Agarwal and Bill Harrod. Organic computing. Technical Report White paper, MIT and DARPA, 2006.
2. Anant Agarwal, Jason Miller, Jonathan Eastep, David Wentzlaff, and Harshad Kasture. Self-aware computing. Technical Report AFRL-RI-RS-TR-2009-161, MIT, 2009.
3. Andreas Agne, Markus Happe, Achim Lsch, Christian Plessl, and Marco Platzner. Self-awareness as a model for designing and operating heterogeneous multicores. *ACM Trans. on Reconfigurable Technology and Systems (TRETS)*, 7(2), 2014.
4. Eyal Amir, M. L. Anderson, and Vinay K. Chaudhri. Report on darpa workshop on self-aware computer systems. Technical Report UIUCDCS-R-2007-2810, UIUC Comp. Sci., 2007.
5. Michael L. Anderson and Don R. Perlis. Logic, self-awareness and self-improvement: The metacognitive loop and the problem of brittleness. *Journal of Logic and Computation*, 15(1):21–40, 2005.
6. Smith Baker. The identification of the self. *Psyc. Rev.*, 4(3):272–284, 1897.
7. K. L. Bellman and D. O. Walter. Biological processing. *American Journal of Physiology*, 246, 1984.

8. Kirstie. L. Bellman, April Gillam, and Christopher Landauer. Challenges for conceptual design environments: the VEHICLES experience. *Revue Internationale de CFAO et dInfographie*, 8(2):185–217, 1993.
9. Kirstie. L. Bellman, Christopher Landauer, and Phyllis R. Nelson. *Systems engineering for organic computing: The challenge of shared design and control between OC systems and their human engineers*, chapter 3. Understanding Complex Systems Series. Springer, 2008.
10. Cristiana Bolchini, Carlo A. Curino, Elisa Quintarelli, Fabio A. Schreiber, and Letizia Tanca. A data-oriented survey of context models. *SIGMOD Rec.*, 36(4):19–26, 2007.
11. Arjun Chandra, Kristian Nymoen, Arve Volsund, Alexander Refsum Jensenius, Kyrre Glette, and Jim Torresen. Enabling participants to play rhythmic solos within a group via auctions. In *Proc. Int. Symp. on Computer Music Modeling and Retrieval (CMMR)*, page 674689, 2012.
12. Tao Chen, Funmilade Faniyi, Rami Bahsoon, Peter R. Lewis, Xin Yao, Leandro L. Minku, and Lukas Esterle. The handbook of engineering self-aware and self-expressive systems. Technical report, EPiCS EU FP7 project consortium, 2014.
13. Michale T. Cox. Metareasoning, monitoring, and self-explanation. In Michael T. Cox and A. Raja, editors, *Metareasoning: Thinking about thinking*, pages 131–149. MIT Press, Cambridge, MA, USA, 2011.
14. M.T. Cox. Metacognition in computation: A selected research review. *Artificial Intelligence*, 169(2):104–141, 2005.
15. Anind K. Dey. Understanding and using context. *Personal Ubiquitous Computing*, 5(1):4–7, 2001.
16. Lukas Esterle, Peter R. Lewis, Xin Yao, and Bernhard Rinner. Socio-economic vision graph generation and handover in distributed smart camera networks. *ACM Transactions on Sensor Networks*, 10(2):20:1–20:24, January 2014.
17. Funmilade Faniyi, Peter R. Lewis, Rami Bahsoon, and Xin Xao. Architecting self-aware software systems. In *Proc. IEEE/IFIP Conf. on Software Architecture (WICSA)*, pages 91–94. IEEE, April 2014.
18. Shaun Gallagher. Philosophical conceptions of the self: implications for cognitive science. *Trends in Cognitive Sciences*, 4(1), 2000.
19. Gordon G Gallup. Chimpanzees: self-recognition. *Science*, 1970.
20. E. Gelenbe and G. Loukas. A self-aware approach to denial of service defence. *Computer Networks*, 51:1299–1314, 2007.
21. Pentti OA Haikonen. Reflections of consciousness: The mirror test. In *Proceedings of the 2007 AAAI Fall Symposium on Consciousness and Artificial Intelligence*, pages 67–71, 2007.
22. Justin W. Hart and Brian Scassellati. *Robotic Self-Modeling*, chapter 14. Imperial College Press/World Scientific Book, 2014.
23. H. Hoffmann, J. Eastep, M.D. Santambrogio, J.E. Miller, and A. Agarwal. Application heartbeats for software performance and health. In *ACM SIGPLAN Notices*, volume 45, pages 347–348. ACM, 2010.
24. Henry Hoffmann, Jim Holt, George Kurian, Eric Lau, Martina Maggio, Jason E. Miller, Sabrina M. Neuman, Mahmut Sinangil, Yildiz Sinangil, Anant Agarwal, Anantha P. Chandrakasan, and Srinivas Devadas. Self-aware computing in the Angstrom processor. In *Proceedings of the 49th Annual Design Automation Conference*, DAC '12, pages 259–264, New York, NY, USA, 2012. ACM.
25. Henry Hoffmann, Stelios Sidiroglou, Michael Carbin, Sasa Misailovic, Anant Agarwal, and Martin Rinard. Dynamic knobs for responsive power-aware computing. In *Proceedings of the Sixteenth International Conference on Architectural Support for Programming Languages and Operating Systems*, ASPLOS XVI, pages 199–212, New York, NY, USA, 2011. ACM.
26. Matthias Hölzl and Martin Wirsing. Towards a system model for ensembles. In *Formal Modeling: Actors, Open Systems, Biological Systems*, pages 241–261. Springer, 2011.
27. William James. *The principles of psychology*. Henry Holt & Co., 1890.
28. Ariane Keller, Daniel Borkmann, Stephan Neuhaus, and Markus Happe. Self-awareness in computer networks. *Int. Journal of Reconfigurable Computing (IJRC)*, 2014.

29. Jeffrey O. Kephart and David M. Chess. The Vision of Autonomic Computing. *IEEE Computer*, 36(1):41–50, 2003.
30. Samuel Kounev. Engineering of Self-Aware IT Systems and Services: State-of-the-Art and Research Challenges. In *Proceedings of the 8th European Performance Engineering Workshop (EPEW'11), Borrowdale, The English Lake District, October 12–13*, 2011. (Keynote Talk).
31. Samuel Kounev. Self-Aware Software and Systems Engineering: A Vision and Research Roadmap. In *GI Softwaretechnik-Trends, 31(4), November 2011, ISSN 0720-8928*, Karlsruhe, Germany, 2011.
32. Christopher Landauer and Kirstie L. Bellman. Knowledge-based integration infrastruc ture for complex systems. *International Journal of Intelligent Control and Systems*, 1(1):133–153, 1996.
33. Christopher Landauer and Kirstie L. Bellman. Self-modeling systems. In H. S. R. Laddaga, editor, *Self-Adaptive Software*, Lecture Notes in Computer Sci ence, pages 238–256. Springer, 2002.
34. Peter R. Lewis, Arjun Chandra, Funmilade Faniyi, Kyrre Glette, Tao Chen, Rami Bahsoon, Jim Torresen, and Xin Yao. Architectural aspects of self-aware and self-expressive systems: From psychology to engineering. *Computer*, 48(8), August 2015.
35. Peter R. Lewis, Arjun Chandra, Shaun Parsons, Edward Robinson, Kyrre Glette, Rami Bahsoon, Jim Torresen, and Xin Yao. A Survey of Self-Awareness and Its Application in Computing Systems. In *Proceedings of the International Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW)*, pages 102–107, Ann Arbor, MI, USA, October 2011. IEEE Computer Society.
36. Peter R. Lewis, Lukas Esterle, Arjun Chandra, Bernhard Rinner, Jim Torresen, and Xin Yao. Static, dynamic, and adaptive heterogeneity in distributed smart camera networks. *ACM Trans. Auton. Adapt. Syst.*, 10(2):8:1–8:30, 2015.
37. Peter R. Lewis, Marco Platzner, Bernhard Rinner, Jim Torresen, and Xin Yao, editors. *Self-Aware Computing Systems: An Engineering Approach*. Springer, 2016.
38. P. Maes. Introspection in knowledge representation. *Advances in Artificial Intelligence II*, pages 249–262, 1987.
39. T. Metzinger. *Being no one*. MIT Press, Cambridge, MA, USA, 2003.
40. Sasa Misailovic, Stelios Sidiroglou, Henry Hoffmann, and Martin Rinard. Quality of service profiling. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*, pages 25–34. ACM, 2010.
41. Melanie Mitchell. Self-awareness and control in decentralized systems (Tech Report SS-05-04). In *AAAI Spring Symposium on Metacognition in Computation*, Menlo Park, 2005. AIII Press.
42. Alain Morin. Levels of consciousness and self-awareness: A comparison and integration of various neurocognitive views. *Consciousness and Cognition*, 15(2):358–71, 2006.
43. Christian Müller-Schloer, Hartmut Schmeck, and Theo Ungerer. *Organic computing: a paradigm shift for complex systems*. Springer, 2011.
44. Ulric Neisser. The Roots of Self-Knowledge: Perceiving Self, It, and Thou. *Annals of the NY AoS.*, 818:19–33, 1997.
45. Albert Newen. The logic of indexical thoughts and the metaphysics of the self. In In W. Künne, A. Newen, and M. Anduschus, editors, *Direct reference, indexicality and propositional attitudes*, pages 105–131. CSLI publications, Stanford, USA, 1997.
46. Albert Newen and Kai Vogeley. Self-representation: Searching for a neural signature of self-consciousness. *Consciousness and Cognition*, 12:529–543, 2003.
47. Kristian Nymoen, Arjun Chandra, Kyrre Glette, and Jim Torresen. Decentralized harmonic synchronization in mobile music systems. In *Proceedings of the International Conference on Awareness Science and Technology (ICAST)*, 2014.
48. Kristian Nymoen, Arjun Chandra, Kyrre Glette, Jim Torresen, Arve Voldsund, and Alexander Refsum Jensenius. Pheromusic: Navigating a musical space for active music experiences. In *Proc. Int. Computer Music Conference (ICMC) joint with the Sound and Music Computing Conference*, page 17151718, 2014.

49. Kristian Nymoen, Arjun Chandra, and Jim Torresen. *Self-Awareness in Active Music Systems*, chapter 14.
50. L. I. Perlovsky and R. Kozma, editors. *Neurodynamics of Cognition and Consciousness*. Springer, Berlin, 2007.
51. Jeremy Pitt, editor. *The Computer After Me: Awareness and Self-awareness in Autonomic Systems*. Imperial College Press, 2014.
52. Georgia Sakellari. The cognitive packet network: A survey. *The Computer Journal*, 53, 2010.
53. Mazeiar Salehie and Ladan Tahvildari. Self-adaptive software: Landscape and research challenges. *ACM Trans. Auton. Adapt. Syst.*, 4(2):1–42, 2009.
54. M.D. Santambrogio, H. Hoffmann, J. Eastep, and A. Agarwal. Enabling technologies for self-aware adaptive systems. In *Adaptive Hardware and Systems (AHS), 2010 NASA/ESA Conference on*, pages 149–156. IEEE, 2010.
55. Thomas Schmickl, Ronald Thenius, Christoph Moslinger, Jon Timmis, Andy Tyrrell, Mark Read, James Hilder, Jose Halloy, Alexandre Campo, Cesare Stefanini, et al. CoCoRo–The Self-Aware Underwater Swarm. In *Proceedings of the International Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW)*, pages 120–126, Ann Arbor, MI, USA, October 2011. IEEE Computer Society.
56. Lenhart Schubert. Some knowledge representation and reasoning requirements for self-awareness. 2005.
57. Aaron Sloman. Prospects for ai as a general science of intelligence. In *Prospects for Artificial Intelligence: Proceedings of AISB93*, Amsterdam, 1993. IOS Press.
58. G. A. Tawney. Feeling and self-awareness. *Psyc. Rev.*, 9(6):570–596, 1902.
59. Jim Torresen, Christian Plessl, and Xin Yao. Self-aware and self-expressive systems. *Computer*, 48(7).
60. J. Wang, D. Brady, K. Baclawski, M. Kokar, and L. Lechowicz. The use of ontologies for the self-awareness of the communication nodes. In *Proceedings of the Software Defined Radio Technical Conference SDR*, volume 3, 2003.
61. Alan F. T. Winfield. Robots with internal models: a route to self-aware and hence safer robots. In Jeremy Pitt, editor, *The Computer After Me*, chapter 16. Imperial College Press/World Scientific Book, 2014.
62. Martin Wirsing, Matthias Hölzl, Nora Koch, and Philip Mayer. *Software Engineering for Collective Autonomic Systems: The ASCENS Approach*, volume 8998 of *Lecture Notes in Computer Science*. Springer, 2015.
63. F. Zambonelli, N. Bicocchi, G. Cabri, L. Leonardi, and M. Puviani. On self-adaptation, self-expression, and self-awareness in autonomic service component ensembles. In *Proceedings of the Fifth IEEE Conference on Self-Adaptive and Self-Organizing Systems Workshops (SASOW)*, pages 108–113, 2011.