Pragmatic Research on Context Modeling and Use 1

Patrick Brézillon, Roy Turner

1

Pragmatic Research on Context Modeling and Use[1]

Pragmatic approaches to solving problems in artificial intelligence arise due to fast technological evolution as well as the complexity of real-world tasks. These operational, bottom-up approaches are based on representing how to behave in specific contexts. In this chapter, pragmatic approaches are discussed from the perspective of agents (human and AI) using data, information, and knowledge in context. Information is what is interpreted by an agent from data about the world, and what is exchanged with other agents. Information is highly context-dependent and under the control of the knowledge that is in the agent's "mind". Context plays a pivotal role in the development of information and knowledge. Setting data, information and knowledge in context allows new approaches to modeling (human and AI) reasoning. Based on its contextual knowledge (the part of its knowledge concerning the current focus), a reasoner interprets and uses information and produces new information and knowledge. Concretely, the pragmatic approach is shown in three examples of context-based research addressing applications in the real world. Operational knowledge in these approaches leads to contextualization of reasoning and opens the door to context-based AI systems.

1.1. Introduction

As artificial intelligence (AI) has matured from its early roots in research focused on "weak", or general-purpose, methods, the field has become increasingly focused on "strong" (knowledge-based), pragmatic approaches to solving problems (Turner & Brézillon, 2022). While research on formal, theoretical topics is still very important, it has become apparent that in order to solve complex real-world problems, a "top-down" approach from theory to system is impractical.

---

[1] Patrick Brézillon & Roy M. Turner

AI research has over time also become increasingly aware of the role context and contextual knowledge plays in problem solving. This has evolved from tentative approaches such as the situation calculus (McCarthy & Hayes, 1969) and dividing rules in expert systems and knowledge-based systems into context-specific packets (e.g., Chandrasekaran *et al*., 1979, Brézillon, 1989) to formal approaches focused on context per se (e.g., McCarthy, 1993; Giunchiglia, 1993) to an entire community focused on modeling and using context beginning with a workshop at the 1993 International Joint Conference on Artificial Intelligence (Brézillon & Abu-Hakima, 1995). This community has grown since them, centered around the CONTEXT (International and Interdisciplinary Conference on Modeling and Using Context) biennial conference series. As with AI in general, this community has become increasingly focused on pragmatic approaches to solve real-world problems.

In this chapter, we discuss the pragmatic approach to context modeling and use in intelligent systems, then examine three approaches we consider to be examples of pragmatic context-based systems and the lessons that can be learned from them for such systems in general. A companion paper (Turner & Brézillon, 2022) takes a closer look at the requirements for pragmatic context-based systems going forward.

1.2. Pragmatic research on context

While natural language researchers have been concerned with context in a narrow sense for many years, context first appeared as an important challenge in AI itself at the International Joint Conference on AI in 1993 (IJCAI-93). In his seminal paper, McCarthy (1993) introduced the relation *ist(c,p)*, which states that a proposition *p* is true in context *c*, and concluded that: a context is always relative to another context; contexts have an infinite dimension; contexts cannot be described completely; and when several contexts occur in a discussion, there is a common context above all of them into which all terms and predicates can be *lifted*. As a consequence, one cannot speak of context outside of its own context.

Context is a concern not only for AI theorists and practitioners, but also for a wide range of disciplines, including psychology and cognitive science, linguistics and natural language processing, neuroscience, anthropology, sociology, organizational behavior, and philosophy. This community of researchers was fostered and made concrete by the CONTEXT series of biennial conferences, beginning in 1997 and continuing to the present. Other specialized research communities interested in context have arisen over the years (e.g. context-aware computing, ubiquitous computing, etc.), but CONTEXT remains the backbone of the interdisciplinary context community.

However, even though it is critical to the success of the interdisciplinary CONTEXT community, there is still no universal definition of context, as pointed out by Bazire and Brézillon (2005), who collected 166 definitions of context in the literature;

as of 2021, the number had grown to 268. Similarly, there is no single definition of contextual knowledge, that is, the knowledge an agent needs about a context and about how to behave while in it. There is also the problem of determining which knowledge is contextual (related to the task focus) and which is not (i.e. external to the task focus). And this is not a static determination, since a piece of external knowledge may become contextual or vice versa as the situation changes.

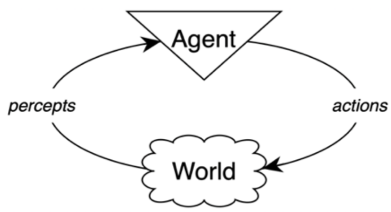
There has been a shift in emphasis and concerns over time. At the first CONTEXT conference in 1997, roughly 40% the papers had to do with communication (including NLP and most cognitive science approaches), 16% with reasoning (formal aspects), and 44% with what might be called “activity” (applications, user interfaces, etc.). Twenty years later, in 2017, this had changed to 25% in communication, 10% in reasoning, and 65% in “activity”. This was driven in large part by the rapidly-growing importance of information and communication technology (ICT) over this time. Since ICT relies on innovations, immediate implementations, and adoption by end-users, it is not concerned with formal aspects of reasoning or cognitive plausibility.

If we focus on a *pragmatic* view of context—how context and contextual knowledge should be modeled in and affect real-world applications and agents—then the definitions at least begin to converge. Here, we can view the agent’s knowledge as its *mental model* of how to realize its task in the local environment, a model that is developed by accumulated experiences, i.e., contextualized knowledge or operational knowledge. Such a mental model assembles, organizes and structures all the contextual elements encountered by the actor during his past experience with this task realization, decision making or problem solving. This is different from the traditional view of a knowledge base or axiom set divorced from how that knowledge is used. A mental model implies a focus on knowledge’s operational and dynamical organization for the current context, changing as the context changes. It is different from agent to agent, and it is built from the agent’s experience performing tasks in different contexts.

1.3. Role of context in AI systems

1.3.1. Data, information, and knowledge

We can see the role of context in AI systems perhaps best from the *agent-based* perspective (Russell & Norvig, 2020), to which much of modern AI subscribes. An *agent* is an entity that repeatedly accepts a percept from the environment, makes a decision, then takes action (see Figure 1.1). A *percept* can be sensor data, a message from other agents, communication from a user, etc., or it can be null (so the



agent can be proactive as well as reactive). Except for the most rudimentary reflex agents, the decision-making process involves updating the agent's own internal knowledge based on the new percept, and the agent's decisions are based on this updated internal knowledge. Humans, animals, and virtually all AI systems can be usefully conceptualized as agents.

What counts as percepts, decisions, and actions will vary from agent to agent. For example, a typical deep-learning vision system has an image as its percept; its "decision process" is just the extremely complex function it embodies; and its action is the result of applying that function to the input image to classify it. An intelligent autonomous underwater vehicle (AUV) derives its percepts from its sensors; its decision process is one or more AI techniques, such as automated planning, behavior-based control, etc.; and its output is via its effectors and their effects on the world. An intelligent assistant system receives inputs from the world and its user; its decision process is comprised of one or more AI techniques; and its outputs are its recommendations, warnings, etc., to the user as well as any direct control it has over the world.

We can think of the information used by an agent as being of three types. The first, simplest, kind is *data*: information that is agent- and context-independent, i.e., objective. Examples of this are raw or filtered sensor signals (e.g., a voltage from a temperature sensor, a pixel's RGB values, etc.) and even some symbolic information ("temperature is 24 °C").

When data are interpreted, processed, organized, structured or presented in a given context, it becomes *information* that is meaningful to the agent who receives it. Information in this sense is data that has been converted into a more useful or intelligible form for helping humans or AI agents in their decision-making process. Context-dependent information has meaning that may vary with the context or the individual reasoning about it. For example, "it is warm" is an assessment based on data (the raw temperature, the feeling of the air on one's skin, etc.), but relative to a particular context and agent; it could refer to 16 ℃ in Paris in March or −12 ℃ in January in Maine.

The third kind of information is *knowledge*. We consider this to be information that is both context-dependent *and* integrated into the rest of the body of knowledge possessed by an agent, often as the result of reasoning about it in relation to the rest of the knowledge. Knowledge is information containing wisdom, so to speak. For example, "386" is data, "your marks are 386" is information, and "You have done well because of your hard work" is knowledge. It is knowledge upon which an agent bases its decisions.

As shown in Figure 1.2, the three kinds exist in a hierarchy of sorts, with complexity, agent-centricity, and integration with other information increasing as one moves up the hierarchy. Data, which can come from either sensors or communication (from another agent or the user) is transformed into information by a process of interpretation. Information, which can also come from communication, is in turn transformed into knowledge by reasoning and by integrating it into the agent's other knowledge. As is also shown, information can come from interpreting other information, and new knowledge can also be produced by reasoning (i.e. by inferences) about existing knowledge.

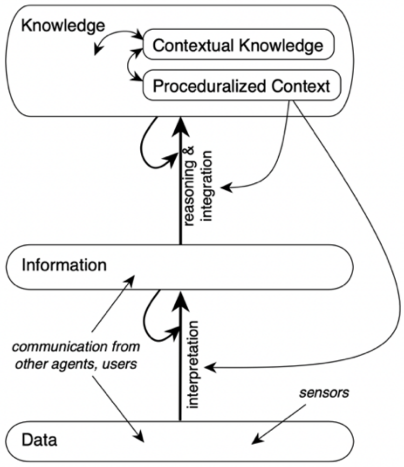


Figure 1.2. Information hierarchy.

The problem with managing knowledge is that it is largely contextual with a strong experiential bias, which makes identifying, acquiring, and replicating is difficult. Knowledge is of two inter-dependent types:

– Fact-based or information-based: This comes from fundamental science, derived from experiments, rules, and principles commonly agreed upon by experts. This type of knowledge is found in models and software.

– Heuristic: This is knowledge of good practice, experience and good judgment. It is the knowledge underlying "expertise", rules of thumb or hypotheses about what usually works to achieve desired results, but it does not guarantee them. This type of knowledge pertains to humans and AI agents (see Turner and Brézillon, 2022).

Many kinds of software use the bottom tiers in the hierarchy. For example, signal processing, data mining and databases, and some kinds of deep learning all take as input data or information and produce other data or information, usually for use and interpretation by a human. However, only humans and intelligent systems can be said to use knowledge in the sense of actually understanding the knowledge itself and integrating it into their own model of the world and ongoing processing.

1.3.2. Contextual knowledge

Context impacts the activities of all humans and real-world AI systems, and, consequently, contextual knowledge must be used by them to function properly within their contexts. As shown in Figure 1.2, contextual knowledge is only one type of knowledge an agent has, and whether or not a piece of knowledge is contextual can change over time as the problem-solving focus changes. Knowledge not relevant to the current context has been referred to as *external knowledge* (Brézillon, 2022). Of contextual knowledge, there are two types: knowledge that is relevant to the current context, but that has not (yet) been instantiated with the specifics of the situation because it is not directly being used (but may have an indirect influence); and *proceduralized context* (Brézillon, 2022), that is, the part of contextual knowledge that has been instantiated based on the task and particular situation at hand and that to some extent directs the agent's processing.[2] A piece of knowledge may move between proceduralized context and non-proceduralized context as the focus of problem solving and/or the situation changes.

Intelligent agents are influenced by contextual knowledge in several ways. It can help a reasoner interpret information it has, producing new information. Simple examples of this would be using context-specific rules to infer from the fact that "depth is 10 m" and from the context of an agent's location (and corresponding data from undersea depth maps) that it is currently at an altitude of 30 m from the sea floor, or to infer from a machine being turned on and the context that the user is present that the user turned the machine on and, thus, meant for it to be on. Context can also help an agent interpret information to produce subjective knowledge by suggesting how to integrate it into the rest of the agent's knowledge.

For both humans and artificial agents, context limits the possibilities for interpretation, reasoning, decision making and acting.[3] For incident solving on a subway line, the operator looks first at the contextual knowledge (status of the entire line, rush hour, how things are done on the other subway lines in correspondence with his line, etc.) in order to select the best contextual knowledge and means to the ultimate goal of ensuring the security of travelers. For a given agent, a change in mood implies a change of context, therefore possibly a different decision.

Context is relevant to Simon's (1960) key contributions to understanding the decision-making process. According to Simon and his later work with Newell (Newell & Simon 1972), decision-making is a process with three distinct phases:

– **Intelligence Phase:** The decision-maker identifies/detects the problem or opportunity. A problem is anything that is not according to the plan, rule or standard, while an opportunity is any promising circumstance that might lead

[2] This reflects work by one of us (Brézillon, 2022); work by the other (Turner, 2022) does not use the term PC, but there are similarities in how contextual knowledge is used in that work.

[3] This is true for larger ensembles of people, too; for example, a *paradigm* in Kuhn's (1970) view of scientific revolutions strictly limits the perception and reasoning of large swaths of the scientific community.

to better results. Information is gathered about a problem, formatted into a useful structure, then factored into the information about the context in which the problem has occurred. The context has a major role in decision making, and information is required both about the problem and about the context in which the problem occurred. Without information about a problem or opportunity, the decision-making process cannot even start. Without information about the context in which the problem has occurred, no correct decision can be made. Information about the context is integral to the complete definition of the problem.

– **Design Phase:** The design phase involves the search for alternative solutions to solve a problem. Information is a key ingredient in the generation of alternatives for decision-making. Each alternative solution is evaluated after gathering data about it. The evaluation is done on the basis of criteria to identify the positive and negative aspects of each solution. Alternative solutions may be difficult to identify in the case of insufficient information and may require a return to the intelligence phase.

– **Choice Phase:** In this stage, alternative solutions are compared to find the most suitable one. This is not as easy as it seems, because each solution presents a scenario, and the problem itself may have multiple objectives. The lack of a satisfactory solution requires a return to the design phase. Based on the information about the suitability of the alternatives, a choice is made to select the best alternative.

Each of these phases of decision making requires knowledge about the context in which it is occurring. Problems or opportunities can only be understood in context, and the information needed to generate alternatives during the design phase comes in large part from knowledge about the context for the solution. Evaluation and choice of alternative is context-dependent as well, since an alternative that is perfectly fine in one context (going into the kitchen to get a glass of water at home) may be completely inappropriate in another (when in a restaurant).

1.4. Three examples of pragmatic research on context

1.4.1. Introduction

In this section, we compare three examples of pragmatic research in context-sensitive reasoning: Context-Based Reasoning (CxBR), Contextual Graphs (CxG), and Context-Mediated Behavior (CMB). These three context-driven approaches were developed in different contexts for addressing initially different objectives. The discussion here comparing them is an extension of work by Gonzalez and Brézillon (2008), who searched for a unifying vision of CxBR and CxG, or at least to identify either their overlap or the specificity of their domains of application. Here, we include CMB in the evaluation to support generalizing conclusions about properties and abilities needed for any pragmatic context-based reasoning approaches. CMB, as is true also of CxBR, is concerned with AI agents, while CxG systems interact with a human or simulated agent (see Turner and Brézillon, 2022).

The initial comparison was based on ten different criteria, with some indication of which one excels at each particular facet of performance. It focused the comparison on how each would represent human or agent tactical behavior, either in a simulation or in the real world.

Conceptually, these three context-driven approaches are not at the same decisional level. This could provide an opportunity in the future to combine them synergistically for a more global approach to context-based reasoners. The inclusion of CMB here broadens the range of application of the systems studied, since CxBR concerned specifically human behavior in a tactical situation, while CMB was developed for AI agents acting autonomously or in multiagent systems at the tactical and strategic levels, and CxG is, on the one hand, at the tactical level with an experience base containing all the practices developed for realizing a given task, and, on the other hand, at an operational level for developing the specific practice of the contextual graph corresponding to the working context at hand. Apart from that, the three context-based approaches focus on task realization, decision-making and problem solving. Context serves to help prune the search for solutions and/or to influence the solution selected or the decision made.

These context-based approaches aim to attack the most challenging problems in modeling human and artificial agent behavior through:

1. Identifying a context and its relevant features. The classical frame problem is closely related to this issue.
2. Deciding what is the most appropriate behavior (actions, practices) for the current context.
3. Knowing when a change in context has taken place. Humans know this instinctively. However, it is difficult to generalize rules that dictate when to switch to another context as a result of environmental and/or internal events.

While these approaches share similarities, they are also quite different in how they model contexts and use contextual knowledge. Therefore, they address the above problems from different perspectives.

1.4.2. Contextual Graphs (CxGs)

A *contextual graph* (Brézillon *et al*., 2000) is a directed, acyclic, series-parallel graph in which a path represents a task realization where the actions are undertaken in accordance with the current context. Actions aim to achieve a goal while *contextual* *nodes* describe the possible contextual issues of a given focus (e.g. a task realization) in different situations. The contextual graph contains the contextual elements more or less related to the focus, and, among them, the *proceduralized context*, that is, the ordered set of instantiated contextual elements corresponding to the given practice development at hand. The proceduralized context is the specific context *chunks* ready to be used in the actions selected for the focus. The proceduralization of the contextual knowledge along the path makes the links explicit, especially the causal and consequential links, between contextual knowledge chunks and as such, the links become a part of the proceduralized context. Thus, the proceduralized context appears as a kind of compiled (i.e. contextualized) knowledge that the system can easily "decompile" to explain its reasoning. Consequently, one can regard a contextual graph as representing the proceduralized context for any path (i.e. practices developed for the focus in different contexts), because for each context represented by a sequence of instantiated contextual elements, the implicit reasoning about causes and consequences implies that the action to undertake is defined without ambiguity. In other words, each sequence of contextual nodes along a branch triggers some actions as a result of rationales that are not represented on the graph but are generally known and compiled in the user's mind (contextualized knowledge). Thus, the CxG explicitly represents the reasoning involved jointly with its proceduralized context.

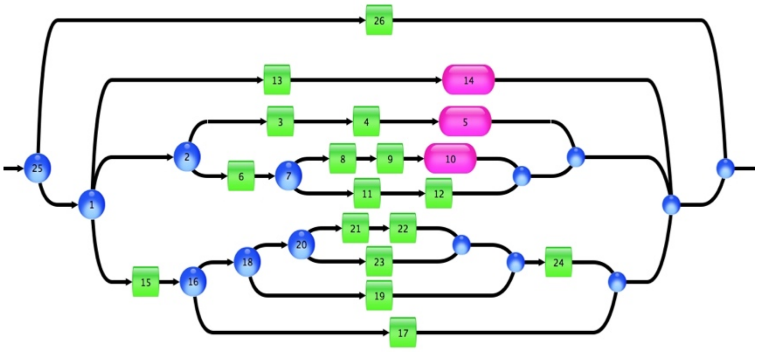


Figure 1.3. Contextual Graphs organization

Figure 1.4 depicts a Contextual Graph and its organization of contextual elements (CE). A contextual element is a pair consisting of a *contextual node* and a *recombination node*. In the figure, the circular nodes with a number are the contextual nodes while the square ones are the action nodes and the elongated ovals represent the activities (independent contextual subgraphs). The contextual nodes require a decision or an information input generally given by the user (but a system may retrieve the required information too), while the action or activity nodes perform an action or describe an action to be performed by the agent. It may also modify the environment and, by extension, the context. The smaller unnumbered circular nodes are the recombination nodes where a particular contextual element ends. Each recombination node is associated with a contextual node, and both correspond to a contextual element.

The purpose of contextual graphs is to represent along each branch of the graph an increasingly refined state of the problem that the users try to identify in order to better target a solution. For example, operators in control rooms for subways want to work with the certainty of having entered the proper branch (i.e. the correct practice to develop), and they generally postpone actions until after the collection of contextual elements. The proceduralized context evolves continuously with the practice development. Some contextual elements enter while others leave the proceduralized context along the process of reasoning.

Recall that the organization of a contextual graph makes explicit the context and its dynamics for decision-making. This representation is more compact than decision trees and was accepted well by the operators in a subway line management application (Pasquier et al., 2000). The representation in a contextual graph of all the known ways for a task realization or decision making as practices allows a contextual graph to behave as an experience base, a basic attribute of context-based humans and AI agents.

Although the CxG formalism was used to model mainly expert's reasoning in task realization, decision making and problem solving, the collection of the instantiation of the contextual elements, the generation of the proceduralized context, and the execution of the actions and activities in the focus may be automated in an AI agent (as implemented in a CxG-based simulator).

1.4.3. Context-Based Reasoning (CxBR)

Context-Based Reasoning (Gonzalez et al., 2008) is based on three ideas:

1. A *situation* corresponds to a set of actions, procedures and expectations that properly address the situation. These sets of actions, procedures and expectations are called *contexts*.
2. Evolution of the situation requires a new set of actions, procedures and expectations to manage the newly emerging situation. Therefore, a *transition* to a new context must be done successfully.
3. A future situation is identified supposing that things in the current situation are limited by the current situation itself. This facilitates *situational awareness*.

CxBR encapsulates into contexts knowledge about appropriate actions, procedures, and expectations as well as knowledge about possible new situations. CxBR is based on concepts such as: not all inputs are relevant; a situation may evolve towards a finite number of next situations, decisions are made in contexts with a defined set of tools; and context changes depend on external events.

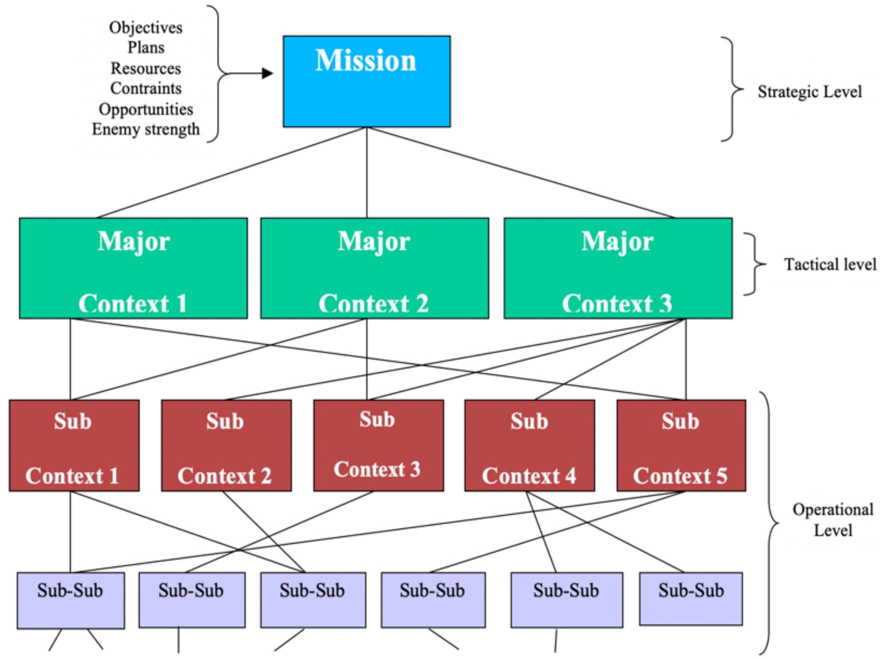


Figure 1.4. CxBR hierarchical organization for a particular mission

The hierarchical organization of CxBR is shown in Figure 1.4. CxBR contexts are organized as a graph, with each context having possibly multiple parents. The hierarchy consists of three levels, the Mission Context, the Major Contexts and the Minor Contexts (see Figure 1.4). There is only one Mission Context, and it serves to define rather than control the agent. Major Contexts are the main control element, and while there may be several assigned to a particular mission, one and only one actively controls the agent at any one time. This is referred to as the *active context*. The Minor Contexts play a supporting role, and there can be several levels of minor contexts, starting with the Sub-Contexts.

A full description of CxBR can be found in Gonzalez and Ahlers (1998).[4]

1.4.4. Context-Mediated Behavior (CMB)

Context-mediated behavior (Turner, 1998, 2017, 2022) is a pragmatic approach to using contextual knowledge that springs from the "scruffy", more cognitively-inspired side of AI (Turner and Brézillon, 2022), in particular from case-based reasoning (Kolodner, 1993). Context is defined somewhat differently in CMB than in many other approaches. An agent's *situation* is considered to be comprised of all of the features of the world, the agent, the mission, etc., that currently exist. As such, its full extent is in general unknown and unrepresentable in most realistic domains and may, in fact, be infinite (cf. McCarthy's [1993] view of contexts). In contrast, a *context* in CMB is defined as a class of similar situations that have similar implications for the agent's behavior. Thus, a context represents a generalization of these situations and so refers to the *salient* features of the world, agent, problem, etc., that impact how the agent should behave. In contrast to a situation, a context in CMB is finite and representable.

Contexts are not necessarily monolithic in this view, but can often be seen as the merger of other contexts. For example, an AUV may find itself on a search mission in a shallow harbor when there is a storm overhead. It may never have encountered this particular kind of situation before, but it may know about contexts such as "in a storm", "in a harbor", "in shallow water", and "on a search mission". Since the current, novel context can be thought of as a combination of these other known contexts, the agent can combine its knowledge about them to create a representation for the novel context.

CMB is part of a reasoning style called *schema-based reasoning* (Turner, 1994), in which most knowledge is stored as packets of related knowledge (*schemas*). *Contextual schemas* (c-schemas) represent contextual knowledge, and *procedural schemas* store knowledge about plans, goal decompositions, scripts, rules, etc. A c-schema contains descriptive and prescriptive knowledge about a context. *Descriptive knowledge* includes facts about the context, both those that must be true for the agent to be considered actually to be in the context as well as predictions about things that may not yet have been encountered by the agent in the current situation. Descriptive knowledge also includes any context-dependent concept meanings. Prescriptive knowledge, on the other hand, tells the agent how it should behave in the context. It includes goal-related knowledge, such as context-dependent goal priorities and which procedural schemas are appropriate to use to achieve goals in the context, and it also includes event-handling knowledge, such as signs that an event may be about to

[4] The interested reader should see the ISL web site at isl.ucf.edu/CxBR_Appendix_A.doc, where a detailed description of CxBR, with formal definitions, can be found.

happen, how to diagnose an event given those signs, how important an event is or what will happen if it is ignored, and what to do in response (activate a new goal, perform a particular action, etc.). Prescriptive knowledge also includes "standing orders"[5] for the context, for example, parameter settings, goals to automatically activate, etc.

In CMB, context assessment is modeled after work on abductive medical diagnostic reasoners (Miller *et al*., 1982). A dynamic memory (Kolodner, 1984; Lawton *et al.,* 1999*)* watches the evolving situation and provides c-schemas that are "evoked" by it to the context manager, which then does differential diagnosis to determine which one(s) best match the situation. These are merged to represent the context. The process repeats as the situation changes.

CMB is meant to incorporate learning. When used with a dynamic memory, simple inductive learning is automatically supported, as c-schemas are updated as new cases of problem solving are organized relative to them. Other types of learning can be added to more intelligently modify and create c-schemas, although to date they have not been implemented.

CMB does not make a clear distinction, as does CxG, between contextual knowledge and proceduralized contextual knowledge (PC). All knowledge in the c-schema(s) representing the current context is considered for use by the agent in the context. However, the actual *procedural schemas* (plan-like structures arising from problem solving in context) used in a situation can be seen as similar to PC. P-schemas, though linked to c-schemas in which they arose or are useful, are stored in their own portion of dynamic memory. When a c-schema suggests one as the way to achieve a goal, the memory is traversed from it as a starting point to find one that fits the context even better. Thus, both the initial p-schema suggested for the context and especially any specialization found can be seen as very similar in spirit to *practices* in CxG.

1.4.5. Conclusions and lessons learned

CxBR, CxGs, and CMB are very similar in some ways, yet still have significant differences. In this section, we look at the similarities and differences, both to explore the properties of pragmatic approaches to contextual reasoning as developed in the approaches as well as a step toward our ultimate goal of developing a unified approach that incorporates their strengths and avoids their weaknesses.

We base our analysis on work by Gonzalez and Brézillon (2008) comparing CxG and CxBr, which included identifying dimensions important for tactical behavior, in

[5] Thanks to the late D. Richard Blidberg for this term.

particular, human tactical behavior. Although these were motivated by the needs of context-based intelligent assistant systems (CIASs), they are general enough to also be important for autonomous real-world AI agents and form the basis for our dimensions, which are:

– Basis in human reasoning
– Task-based or goal-based
– Type of formalism
– Scope of application
– Context granularity
– Representation of context dynamics
– Representation of activity
– Environment interface
– Representation of time
– Knowledge acquisition and learning
– Representation of uncertainty and unpredictability
– Provision of explanations

A summary of the comparison is shown in Table 1.1 below. In the rest of this section, we discuss the major similarities and differences in more detail.

All three approaches generally attempt to emulate human-like intelligence by using context as the basic modeling element. CxG is designed to work with context-based intelligent assistant systems (CAISs), where the agent needs to work like a human and be understandable to a human. CxMB was developed in part to control simulated agents in training simulations, and so it is important that it, too, have behavior that is understandable by and similar to humans. CMB has been mostly used for autonomous systems where human-like behavior is not as important as effectively accomplishing the mission, but it, too, is based on cognitively-plausible principles, and in at least two implementations (in a medical diagnostic assistant (Turner, 1994) and in a virtual human (Wilson, 2022, this volume), it needs human-like behavior.

CxBR and CxGs both are task-based, in sharp contrast to several popular cognitive architectures such as SOAR (Laird *et al.*, 1987), ACT-R (Anderson *et al.*, 1997) and other goal-based approaches that rely on the agent setting a goal, with the system trying to find a production or some other knowledge element that allows the agent to achieve it. In CxG, goals are implicit in the context being experienced. CxBR introduces its mission goals in the separate, non-controlling Mission Context, of which there is only one. The intermediate goals of the agent in the mission are defined by a plan consisting of a sequence of Major Contexts with the transition criteria defined. However, the control contexts themselves contain the functionality to perform the tasks appropriate to that context. In contrast, CMB is more agnostic on the distinction between being goal- or plan- based; it can be either depending on the agent in which it is embedded. If used with a planner, then it will provide context-appropriate goal information, including how to achieve goals in the context; when used with other reasoners (e.g., a deep-learning system), then will provide non-goal-based guidance (e.g., appropriate net topology, weights, and hyperparameters).

Table 1.1. Comparison of three pragmatic context-based approaches

Criterion	CxBR	CxG	CMB
Basis in human reasoning	Yes	Yes	Cognitively motivated
Task- vs goal-based	Task-based	Task-based	Either
Type of formalism	Graphs with multiple parents and shared children	Series-parallels & acyclic directed graphs	Schemas organized along similarities and differences
Scope of application	Hierarchical organization of missions	Real-time modeling of practice development	Hierarchical mission organization
Granularity	Strategic → operational, but esp. tactical	Tactical → operational	Strategic → operational
Representation of context dynamics	Discrete	Continuous	Discrete
Representation of activity	N/A	At the core of the formalism	Included
Environment interface	Situation-driven	None	Present or absent
Representation of time	Explicit	Implicit	Mostly implicit
Knowledge acquisition and learning	None	Incremental acquisition and learning when system fails	From humans and/or via induction as cases are stored
Uncertainty/unpredictability management	Yes	N/A	Context-specific unanticipated event handling
Explanation Provision	None	Practices come from human experts	None

With respect to formalism, none of CxBR, CxG, or CMB are languages, as are, e.g., SOAR and ACT-R, nor true representational paradigms, but rather conceptual modeling approaches where knowledge is organized by context. How a model is defined or is actually implemented is unspecified. This gives the developer great flexibility in adapting to the specific needs of the computing environment available. For example, CxBR could be easily modeled through finite state machines, Petri nets, or

Markov chains. The action knowledge within a Major or Minor context (Sub-context, Sub-Sub-context, etc.) can be likewise represented through functional programming, rule-based systems, constraint networks, or neural networks, among others. Transition knowledge, while traditionally represented by production rules (i.e. *transition rules*), have also been incorporated through neural networks (Stensrud, 2005). Problems represented by contextual graphs in CxGs have likewise been implemented in various different ways, such as Petri nets, influence graphs, production rules and others. It is interesting to note that while a contextual graph can be expressed fully in terms of productions, a set of productions cannot be expressed in a unique way as a contextual graph. CMB's contextual schemas be implemented as frames, semantic networks, description logic, etc., and its schema memory, although intended to be implemented as a dynamic memory, can be anything reasonable. Transitions between contexts have been implemented by an active dynamic memory noticing changes or by predictions directly contained in c-schemas (Wilson, 2022).

The three approaches differ with respect to the kinds of applications they have been used for. CxBR has been used for the hierarchical (strategic, tactical, and operational) organization of missions for real and simulated agents. CxG has been used extensively in modeling practice development for real-world CIASs in several disciplines. CMB has been used in the domains of a medical diagnosis assistant, hierarchical organization of missions for autonomous agents, for organization of multiagent systems, and most recently in the domain of context-dependent deep learning.

The approaches can be compared by the level, or granularity, of reasoning they control: strategic, tactical, or operational. CxG exists primarily at the operational level, concerned as it is with practices in particular contexts, but also can be at the tactical level. CxBR is primarily at the tactical level, but also extends to the strategic and operational levels. CMB similarly extends across all three levels, but makes no hard distinction between contexts existing at different levels; use of a particular contextual schema at one or another of the levels depends on the needs of the reasoner and on its situation. The focus by the three approaches on different levels of granularity promises to allow work that has been done on the strategic and tactical levels (CxBR, CMB) to inform future work at those levels and work focusing on the operational level (CxG) to inform future work at the operational, practice-based level, which was not a focus of the other two.

The approaches can also be compared based on their treatment of the dynamic nature of contexts. All three take context dynamics seriously, with provision for representing the changing context as the situation or the context evolves. CxBR and CMB treat contexts as discrete and relatively large-scale. The context is tracked and represented as a sequence of representations of entire contexts, even though any particular context may be represented by several knowledge structures. In contrast, CxG takes a more continuous approach, with the context changing smoothly as the agent progresses through the graph when a contextual element is instantiated differently.

In summary, the three approaches, developed contemporaneously for the particular needs of their applications, resulted in some important differences, but remarkably also many commonalities. This speaks, we feel, to the common needs of pragmatic, context-based reasoning systems.

In particular, we see that all three approaches developed explicit, rich representations of contexts centered around the salient features of similar problem-solving situations and how to behave appropriately in them. Contexts in each are represented as first-class entities about which the system can reason, not just symbols or a collection of axioms. All approaches also explicitly recognize the changing nature of context and have developed different, though potentially compatible or even synergistic, ways to transition from context to context. All three highlight the benefit of organizing knowledge by context for acquiring knowledge from humans.

In addition, in their differences, the three approaches still point out potentially important characteristics of pragmatic context-based systems. For example, two of the approaches highlight the importance of the tactical and strategic levels of reasoning, while another shows the importance of context in developing and using operational knowledge. Thus, we can see that context plays an important role at all three levels. Two of the approaches highlight the need to acquire context-related knowledge from humans (experts and especially users), while the third shows a route to automatic contextual knowledge acquisition and update via a process borrowed from memory update in case-based reasoning.

1.5. Conclusion

During the last 25 years, technological evolution has relied on a very dynamic development of bottom-up approaches in AI in general, and in context research in particular. This took a more pragmatic, operational form than a formal, top-down one. This is important when trying to solve real-world problems. Any parent who has tried to teach their child how to ride a bike knows the virtue of a pragmatic method (e.g. pushing the child on the bike) rather than explaining to the 5-year old child the mathematical model of riding a bike (as in the 10-page paper of Fitton and Symons 2018).

The need to tackle operational knowledge is now acknowledged as requiring contextualized knowledge, by opposition to decontextualized knowledge (e.g. procedures or some formal approaches) that looks to represent general behavior that is robust for any situation. Pragmatic research concerns mainly context-based systems, and it looks to represent how to behave in specific contexts. This is not to say that formal systems are not important. After all, although there were undoubtedly many pragmatic engineering practices used, getting a designing and getting a spacecraft to Mars required a hefty dose of formal methods.

This paper is situated against the backdrop of the CONTEXT community. This community is international and interdisciplinary precisely because context is a concern found in a number of domains in which humans solve problems. A major objective of the community is a sharing of results across the domains. The main problem is the lack of a unifying model of context, as shown by the large number of definitions found in the literature. This chapter presents successful context-based approaches in different technological domains, and proposes an operational definition of context for at least one of them.

We have found it useful to consider an agent-based approach and to distinguish between data (low-level, objective, agent- and context-independent), information (higher-level, context-dependent, somewhat agent-dependent) and knowledge (high-level, integrated, highly context- and agent-dependent). Based on its contextual knowledge (the part of its knowledge concerning the current focus), a reasoner interprets and use information, and produces new information. Interpretation, reasoning, decision making and acting are strongly context-based.

The final contribution of this chapter is an illustration of what a pragmatic, context-based intelligent system looks like by presenting three examples addressing applications in the real world: Context-Based Reasoning (CxBR), Contextual Graphs (CxG) and Context-Mediated Behavior (CMB). These have been used in different domains (mission management for the army for CxBR; medical diagnosis and autonomous agents for CMB; and several different domains for CxG). We show that they share many commonalities and, though there are differences, there are no incompatibilities. They all put context front stage for their use of data, information, knowledge and reasoning in a uniform representation. They use operational knowledge as previous approaches used formal knowledge. The main interest in operational knowledge is to adapt formal knowledge to each situation in which it is used and for each specific problem to which it is applied. This thus leads to a contextualization of the reasoning and to pragmatic, context-based intelligent (assistant or autonomous) systems, which is the subject of a companion chapter (Turner & Brézillon, 2022) in this book. The comparison leaves us hopeful that an integrated approach can be devised that combines the approaches' strengths while avoiding their weaknesses.

1.6. References

Anderson, J.R., Matessa, M., Lebiere, C. (1997). ACT-R: A theory of higher level cognition and its relation to visual attention. *Human–Computer Interaction*, *12*(4), 439–462.

Brézillon, P. (1989). Interpretation and rule packet in expert systems. Application to the SEPT expert system. International Conference on Knowledge Based Computer Systems, 78–87.

Bazire, M., Brézillon, P. (2005). Understanding context before to use it. In A. Dey, B. Kokinov, D. Leake, & R. Turner (Eds.), *Modeling and using context: Proceedings of the international and interdisciplinary conference* (Vol. 3554, 29–40). Springer Verlag.

Brézillon, P. (2022). Modeling and using context: 25 years of lessons learned. In P. Brézillon & R.M. Turner (Eds.), *Modeling and Using Context in Action*. ISTE Science Publishing Ltd.

Brézillon, P., Abu-Hakima, S. (1995). Using knowledge in its context: Report on the IJCAI-93 Workshop. *AI Magazine*, *16*(1), 87–87.

Brézillon, P., Pasquier, L., Pomerol, J.-C. (2000). Representing operational knowledge by contextual graphs. In *Advances in artificial intelligence*, 245–258. Springer.

Chandrasekaran, B., Gomez, F., Mittal, S., Smith, J. (1979). An approach to medical diagnosis based on conceptual structures. *Proceedings of the Sixth International Joint Conference on Artificial Intelligence*.

Fitton, B., Symons, D. (2018). A mathematical model for simulating cycling: applied to track cycling. *Sport Engineering*, 21, 409-418 (https://doi.org/10.1007/s12283-018-0283-0).

Gonzalez, A.J., Ahlers, R. (1998). Context-based Representation of Intelligent Behavior in Training Simulations. *Transactions of the Society of Computer Simulation* 15 (4), 153-166.

Gonzalez, A.J., Brézillon, P. (2008). Comparing two context-driven approaches for representation of human tactical behavior. *Knowledge Engineering Review*, *23*(2), 1-21.

Gonzalez, A.J., Stensrud, B.S., Barrett, G. (2008). Formalizing context-based reasoning: A modeling paradigm for representing tactical human behavior. *International Journal of Intelligent Systems*, *23*(7), 822–847.

Kolodner, J.L. (1984). *Retrieval and organizational strategies in conceptual memory*. Lawrence Erlbaum Associates.

Kolodner, J.L. (1993). *Case-based reasoning*. Morgan Kaufman.

Kuhn, T.S. (1970). *The structure of scientific revolutions* (second edition). The University of Chicago Press.

Laird, J.E., Newell, A., Rosenbloom, P.S. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence*, *33*(1), 1–64.

Lawton, J.H., Turner, R.M., Turner, E.H. (1999, July). A unified long-term memory system. *Proceedings of the International Conference on Case-Based Reasoning (ICCBR'99)*.

McCarthy, J. (1993). Notes on formalizing context. *IJCAI*, 555–560.

McCarthy, J., Hayes, P.J. (1969). Some philosophical problems from the standpoint of artificial intelligence. In B. Meltzer & D. Michie (Eds.), Machine intelligence 4. Edinburgh University Press.

Miller, R.A., Pople, H.E., Myers, J.D. (1982). INTERNIST–1, an experimental computer-based diagnostic consultant for general internal medicine. *New England Journal of Medicine*, *307*, 468–476.

Newell, A., Simon, H.A. (1972). *Human Problem Solving*, Englewood Cliffs, NJ., Prentice Hall (summary paper).

Pasquier, L., Brézillon, P., Pomerol, J.-C. (2000). From representation of operational knowledge to practical decision making in operations. *Decision Support Through Knowledge Management*, 301–320.

Russell, S., Norvig, P. (2020). *Artificial intelligence: A modern approach* (4th ed.). Pearson.

Simon, H.A. (1960). *The new science of management decision*. Harper & Brothers. https://doi.org/10.1037/13978-000

Stensrud, B.S. (2005). *FAMTILE: An algorithm for learning high-level tactical behavior from observation*. University of Central Florida.

Turner, R.M. (1994). *Adaptive reasoning for real-world problems: A schema-based approach*. Lawrence Erlbaum Associates.

Turner, R.M. (1998). Context-mediated behavior for intelligent agents. *International Journal of Human–Computer Studies*, *48*(3), 307–330.

Turner, R.M. (2017). Context-mediated behavior. *Modeling and Using Context*, *17*(1).

Turner, R.M. (2022). Pragmatic reasoning in context: Context-Mediated Behavior. In P. Brézillon & R.M. Turner (Eds.), *Modeling and Using Context in Action*. ISTE Science Publishing Ltd.

Turner, R.M., Brézillon, P. (2022). Toward pragmatic context-based systems. In P. Brézillon & R.M. Turner (Eds.), *Modeling and Using Context in Action*. ISTE Science Publishing Ltd.

Wilson, C.L. (2022). Context-driven behavior: A proactive approach to contextual reasoning. In P. Brézillon & R. M. Turner (Eds.), *Modeling and Using Context in Action*. ISTE Science Publishing Ltd.