Ethics and Information Technology (2007) 9:49–62
DOI 10.1007/s10676-006-9128-9

Ethical protocols design

Matteo Turilli
Oxford University Computing Laboratory, University of Oxford, Wolfson Building, Oxford, OX1 3QD, UK
IEG, University of Oxford, Oxford, UK
E-mail: matteo.turilli@oucl.ox.ac.uk

Abstract. The paper offers a solution to the problem of specifying computational systems that behave in accordance with a given set of ethical principles. The proposed solution is based on the concepts of *ethical requirements* and *ethical protocols*. A new conceptual tool, called the *Control Closure* of an operation, is defined and used to translate ethical principles into ethical requirements and protocols. The concept of *Generalised Informational Privacy* (GIP) is used as a paradigmatic example of an ethical principle. GIP is defined in such a way as to (i) discriminate specific cases in which an individual's GIP can be infringed without accessing the individual's data; (ii) separate unauthorised accesses to data that do not respect the right to GIP from access that do; and (iii) distinguish different degrees of GIP. Finally a camera phone is used to illustrate the proposed solution.

Key words: ethical protocols, ethical requirements, informational privacy, protocols design, system specification

Introduction

The responsibilities of a system designer are growing and expanding in fields that only 10 years ago were the exclusive realms of philosophy, sociology or jurisprudence. Nowadays, a system designer must have a deep understanding not only of the social and legal implications of what he is designing but also of the ethical nature of the systems he is conceptualising. These artefacts not only behave autonomously in their environments, embedding themselves into the functional tissue or our society but also 're-ontologise'[1] part of our social environment, shaping new spaces in which people operate.

It is in the public interest that automated systems minimise their usage of limited resources, are safe for users, and integrate ergonomically within the dynamics of every-day life. For instance, one expects banks to offer safe, multifunction ATMs, hospitals to ensure that electro-medical instruments do not electrocute patients, and nuclear plants to employ redundant, formally specified control systems.

It is equally important to the public interest that artificial autonomous entities behave correctly. Autonomous and interactive systems affect the social life of millions of individuals, while performing critical operations such as managing sensitive information, financial transactions, or the packaging and delivery of medicines. The development of a precise understanding of what it means for such artefacts to behave in accordance with the ethical principles endorsed by a society is a pressing issue.

The first section of this paper presents the definitions of the concepts of 'actor', 'agent', 'individual' and 'heterogeneous organisation'. Actors, agents and individuals are entities that operate within environments that will be referred to as 'heterogeneous organisations'. Other concepts introduced in the paper are to be understood in terms of this underlying ontology.

The second section introduces a specific type of actor, namely Information and Communication Technology (ICT) actors. These are autonomous and interactive technologies that may have an ethical impact on the environment on which they operate. Two concrete examples of ICT actors are introduced: Automated Pharmacy Systems (APSs) and Business

[1] Re-ontologise is a neologism introduced in Luciano Floridi. The Ontological Interpretation of Informational Privacy. *Ethics and Information Technology*, 7(4): 185–200, 2006. "Computers and ICTs are [...] ontologizing devices because they engineer environments that the user is then enabled to enter through (possibly friendly) gateways."

Process Management Systems (BPMSs). The social risks of automation are introduced by recalling the classic example of the Wall Street Black Monday.

The Ethical Consistency Problem (ECP) is introduced in the third section. This is the core problem investigated in this paper. The identity theft felony/crime is used as a paradigmatic example of ECP. The analysis of the problem is furthered by looking at what types of ethical principles are used to constrain individuals and by discussing a prototypical abstraction of a software design process.

The forth section offers a solution for the Ethical Consistency Problem. Initially, the concepts of Ethical Requirement and Ethical Protocol are defined. The proposed solution is then described by suggesting a two-step translation, from ethical principles to ethical requirements and then from ethical requirements to ethical protocols. A new formal tool, called *Control Closure* of an operation (CCop), is defined and used to represent the normative constraints expressed by an ethical principle in terms of ethical requirements and then ethical protocols.

The fifth section contains an explicative example of the solution offered for the ECP. The example is composed of the definition of the ethical principle of Generalised Informational Privacy (GIP), the translation of this principle into an ethical requirement and the description of an ethical protocol for camera phones.

In the conclusion, some of the possible developments of the approach supported in this paper are briefly outlined.

Actors, agents, individuals in heterogeneous organisations

Heterogeneous organisations, whether society as a whole, private and public companies or research institutions, are populated by *actors, agents* and *individuals* that share resources to coordinate their activities effectively, in order to achieve common goals.

Actors[2] are autonomous entities capable of interacting with the environment by receiving inputs, producing outputs and performing operations. Actors have some degree of control over their internal state, which enables them to perform their operations autonomously, i.e. without the direct intervention of other entities.

The definition of actor has a minimal ontological commitment and is ontologically neutral. There is no one-to-one correspondence between actor and entity as an actor is defined by the operations it performs. At a given level of abstraction (LoA)[3] an operation that defines an actor may be performed by the overall activity of a distributed system. In this case, the whole system would instantiate one actor. At a different LoA, the same operation could be performed by different independent processes. In this case, every process would instantiate a different actor. The autopilot of an airplane, for example, is an actor, as it autonomously cruises an airplane while interacting with the environment. Depending on the LoA adopted, the autopilot can be considered as a single actor that performs the operation of flying an airplane or as a set of interacting actors that execute the subtasks of that operation.

Actors are ontologically neutral as there is no assumption about the nature of the entities that compose an actor. Any autonomous and reactive transition system can be an actor. Computational processes, mechanical artefacts, biological entities, ICTs, distributed systems, control systems, trading programs may be all good examples of actors.

The concept of agent is widely used in different research fields. Since there is no general agreement on its definition, a minimalist, and hence conceptually safer, definition of agent is to be preferred.[4] Agents are not only interactive and autonomous, like actors, but also have the distinctive property of being adaptive.[5] Adaptation is the ability of the agent to change the rules that it follows in order to perform its operations. A typical example of an artificial agent is a thermostat endowed with machine learning algorithms. The thermostat interacts with its environment and autonomously adjusts the heater, but it is also capable of adapting by machine learning to distinguish warm and cold seasons as well as the preferences of the tenants. So agents are a special kind of actors.

The definitions of agent and actor share the same minimal ontological commitment and ontological neutrality. Whole companies, computational systems capable of machine learning, human societies and single human beings can be regarded as agents.

[2] Terence Hawkes. *Structuralism and Semiotics*. 2nd ed. Routledge, London, 2003; Carl Hewitt, Peter Bishop, and Richard Steiger. A Universal Modular Actor Formalism for Artificial Intelligence. *IJCAI3*, pp. 235–245. Stanford, CL, 1973.

[3] Luciano Floridi and Jeff W. Sanders. The Method of Abstraction. In M. Negrotti, editor, *Yearbook of the Artificial. Nature, Culture and Technology*, pp. 177–220. P. Lang, Bern, 2004.

[4] Gian Maria Greco, Gianluca Paronitti, Matteo Turilli, and Luciano Floridi. How to Do Philosophy Informationally. *Lecture Notes in Artificial Intelligence*, 3782: 623–634, 2005.

[5] Luciano Floridi and Jeff W. Sanders. On the Morality of Artificial Agents. *Minds and Machines*, 14(3): 349–379, 2004.

Individuals are the traditional entities that perform operations in an organisation. Individuals are not only autonomous, interactive and adaptive but are also endowed (at least) with semantic capacities. The frame problem provides a good criterion to discriminate between the two kinds of agents: truly semantically enabled agents (i.e. individuals) are not affected by it, whereas ordinary agents cannot overcome it. The upper boundary of an individual's complexity is open-ended. Intelligence, intuition, sensibility and artistic expressions are all properties that can be added to the previous definition.

The definition of individual is strongly ontologically committed and weakly ontologically neutral. Individuals map with single entities, for example costumers, employees, managers or owners of an organisation. While in principle it is plausible to imagine artefacts endowed with truly semantic capabilities, at the present time only thinking biological entities are exempt from the frame problem.

ICT actors in heterogeneous organisations

Having defined the entities that perform the activities of a heterogeneous organisation, the next step it to clarify how each entity operates. The ubiquitous adoption of ICT infrastructures is increasingly affecting the way in which tasks are performed inside heterogeneous organisations. These changes are of at least two main types.

First, the instrumental use of ICT systems augments the individuals' capabilities to manage the whole life cycle of information (creation, collection, storage, manipulation, transmission, etc.). This increases the productivity of the organisation without affecting which individuals can perform an operation, targeting instead *how* the operations are executed. Faxes tend to perform faster than pigeons and pocket calculators tend to be more efficient than paper and pencil. Neither faxes nor pocket calculators are actors as they are not autonomous. They are tools that allow actors, agents and individuals to perform their operations better.

Second, the development of ICT instruments that offer growing degrees of *autonomy* in performing their operations leads to the automation of parts of the activities of an organisation. These instruments become actors and agents of the organisation in which they are deployed. Stock market exchanges, identification procedures, billing and payments, taxation, call centres, emergency management, data storage and duplication and data mining are all examples of activities or processes that have been fully or partially automated, with a corresponding degree of outsourcing and delegation.

The autonomy of ICT actors and agents consists in the routine execution of one or more operations, whenever a given set of parameters holds. Operations and parameters are defined (by individuals in the organisation) in such a way as to guarantee that, given a particular situation, the outcome of the performance of the ICT actors and agents may be as good as, or even better than, that of individuals placed in analogous conditions. Two examples of actors deployed to automate part of the activity of heterogeneous organisations are Automated Pharmacy Systems (APSs)[6] and Business Process Management Systems (BPMSs).[7]

APSs are robotic systems capable of automating the delivery of drugs in hospital pharmacy dispensaries. Research conducted at the pharmacy of the Royal Wolverhampton Hospitals NHS Trust (RWHT)[8] documents how APSs dramatically reduce time and errors (16% less) in drug delivery, thereby maximising staff efficiency and storage space for medicines.

HSBC, a world wide bank network with 9500 offices in 76 countries, will update its BPMS in the next 2 years (by 2008) in order to achieve a higher level of automation in answering customer queries.[9] The new system will automatically generate the documentation relative to the status of the transactions performed either domestically or cross-borders in the whole HSBC global network. The automation of these operations will reduce individual intervention, minimising the time spent answering clients' queries.

The coherence of the operations performed by different actors, agents and individuals of an organisation is pivotal to the consistency of the overall conduct of that organisation. For example, earlier APS systems were unable to manage packages produced by the pharmaceutical companies for the usual distribution. This functional deficiency was a limiting factor in the adoption of APSs, as it produced an incoherent drug delivery system. Drugs had to be delivered by APS actors and individuals following an automated and a manual procedure. The hospital had to address this inconsistency by developing hybrid procedures,

[6] Rachel Graham. Robots Benefit Patients and Staff in Hospitals and Community Pharmacies. *The Pharmaceutical Journal*, 273: 534, 2004.

[7] Wil M.P. van der Aalst. Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management. *Lecture Notes in Computer Science*, 3098: 1–65, 2004.

[8] Ray Fitzpatrick, Peter Cooke, Carol Southall, Kelly Kauldhar, and Pat Waters. Evaluation of an Automated Dispensing System in a Hospital Pharmacy Dispensary. *The Pharmaceutical Journal*, 274: 763–765, 2005.

[9] Steve Ranger. Bank Automates to Boost Customer Service. Case Study: HSBC Speeds up Queries with Workflow Automation. *Silicon.com*, Monday 06 February 2006.

which therefore increased organisational complexity, decreased efficiency and led to higher costs. Modern APSs are not similarly restricted and can be coherently introduced into the drug delivery workflow. APSs and individuals can collaborate consistently in order to achieve the goal of an efficient drug delivery procedure.

One of the main criteria used by the HSBC ICT staff, in choosing the new BPMS, has been the possibility of its integration with other actors operating in the bank network. The absence of compatibility between different actors deployed in the workflow of the HSBC customer care would be the source of potential inconsistency issues analogous to those faced by the earlier adopters of the APSs systems.

Automation can easily prove to be more problematic when subjective parameters are involved. The infamous Black Monday of Wall Street in 1987[10] is a striking example of the catastrophic effects that can be produced by automated actors that are not bound by an appropriate combination of economical, psychological and sociological factors. The causes that led to the devastating events of Black Monday are still debated and the relevance of the use of automated trading procedures has often been reconsidered. Nonetheless, it is generally acknowledged that the over-simplified automation of trading programs was one of the major factors that contributed to the vertical fall of the market.[11] The problem was caused by the inability of the trading actors to perform according to the complex set of rules that determine the trading strategies in unusual situations. Trading actors did not consider crucial factors of the market and produced an inconsistency in the behaviour of the trading organisation. That inconsistency contributed to the serious social and economical consequences of Black Monday.

Since 1987, the growth of automation has been propelled by the continuous spread and evolution of ICT actors and by the expansion of the global market.[12] A crucial factor in the evolution of automation is the massive process of parallelisation and distribution of computational and communication resources. Research into distributed systems is radically changing both what computational systems can do and how they do it. As a direct effect of this progress, distributed databases[13] and ubiquitous communication and computation networks[14] – internet and grids – are becoming the foundations for the deployment of more and increasingly complex actors.

The ethical consistency problem (ECP)

Distributed ICT actors collaborate alongside individuals in performing operations on sensitive data in, for example, banks, hospitals, public offices and private companies. They control, among other things, high volumes of economic transactions, sensitive industrial machineries, customer care systems and medical devices. These operations may have critical impact both socially and economically. Basic human rights can be affected as, equally, the business image of the organisations. Individuals that perform such critical operations are usually bound by a set of ethical principles that normatively constrain their behaviours. It is crucial to use an analogous set of ethical principles to bind ICT actors. This problem can be referred to as the Ethical Consistency Problem (ECP). Here is a definition:

Given:

1. a heterogeneous organisation composed of actors, agents and individuals, and
2. a set of ethical principles constraining the individuals,

the ECP consists in:

> how to constrain actors, agents and individuals with the same set of ethical principles so that the overall output of the organisation is ethically consistent.

The ECP is a tangible problem. Consider identity theft, for example. This is a general label for any crime perpetrated when sensitive personal information is stolen. With this information, the felon is able to assume the identity of his victim and gain access to bank accounts, obtain credit cards, loans or even more reserved information, sometime with devastating consequences for the victim. In 2001, an article on the BBC[15] reported that, with an increase rate of 500% a year, identity theft was Britain's fastest-growing white-collar crime. In the same article, it was estimated that American figures for identity theft stood in the region of hundreds of thousands. Two years later, in 2003, the Federal Trade Commission released a survey[16] in

[10] Avner Arbel and Albert E. Kaff. *Crash: Ten Days in October. Will It Strike Again?* Longman Financial Services, Chicago, 1989.

[11] M. Mitchell Waldrop. Computers Amplify Black Monday. *Science* 238(4827): 602–604, 1987.

[12] Daniel Gross. Attack of the Machines. Is Your Stockbroker a Robot? *Slate*, Jan. 18, 2005.

[13] M. Tamer Özsu and Patrick Valduriez. *Principles of Distributed Database Systems*. 2nd ed. Prentice Hall London, 1999.

[14] Josâe C. Cunha and Omer Rana. *Grid Computing: Software Environments and Tools*. Springer, London, 2006.

[15] John Penycate. Identity Theft: Stealing Your Name. *BBC News*, Monday, 18 June 2001.

[16] Synovate. *Identity Theft Survey*. Federal Trade Commission, 2003.

which it was estimated that, between 1997 and 2002, 27.3 million Americans had been victims of identity theft, 9.9 million in 2002 alone. The survey reported losses of $48 billion for businesses and financial institutions and $5 billion for individual consumer victims.

Poor identification procedures and sensitive data scattering are the main causes of identity theft. Biometric identification seems the path chosen by governments and private companies to secure identification procedures. It is information scattering, however, that presents a much more elusive problem. Thus far, the main method of preventing information scattering is the avoidance of disclosure of sensitive data. Unfortunately, this is essentially impossible. In an increasingly digitalised society, people do not have full control over their sensitive data. Sensitive data are digitalised and stored in computational devices that are not under the direct control of the data owner. These data are given away for entirely legitimate reasons, such as opening a bank account, paying taxes, buying goods, or simply paying with a credit card at a restaurant. Once digitalised, sensitive data become fodder for (distributed) ICT actors. These actors make limited distinctions between the quality of information they manipulate. They store, duplicate, manipulate and exchange information with few, if any, constraints. Regulations that normatively constrain the handling of sensitive data by individuals do not affect the ICT actors that perform operations on the very same data set. Identity theft is a clear instance of the ECP.

ECP is a problem involving the design of dynamic systems. Specifically, in this paper, the ECP refers to the design of distributed ICT actors. The first step towards a solution to the ECP is to understand how individuals and distributed ICT actors are or can be ethically constrained. The second step will be to propose a solution to the ECP and the third step to illustrate its application using a modelled example.

Actions performed by individuals employed by organisations are constrained by a set of ethical principles. In the most general case, these principles may derive from the knowledge and information available to the individual as much as his beliefs, education and culture. They may influence the individual's behaviours consciously or unconsciously, affecting different spheres of his activity, for example interpersonal relationships, choices, attitude and evaluation of working situations. The refusal to work on weapon-related projects is a typical example of how personal ethical principles may affect the individual's choices and job-related activities.

An organisation can openly commit itself to a set of ethical principles.[17] These principles may be endorsed in terms of codes of conduct, stakeholder statutes and values statements. Similar documents define a wide range of company responsibilities, ranging from the quality of products and services to a commitment to respect the environment. They also delineate the appropriate conduct among employees, the principles that stakeholders must respect, and how the employees can use the organisation's properties. These principles normatively constrain individuals that opt to become members of the organisation. For example, members of an organisation might have to avoid racial or gender discrimination or may have to promote team-work. Stakeholders might be committed to principles of fairness, transparency and honesty.

Finally, individuals and organisations may be subject to state or international laws and regulations. For example, the manager of a company is expected to obey the laws of the country in which he is developing his business, or to adhere to international laws during operations involving more than one country.

Once it is understood how ethical principles can affect the behaviour of single individuals and whole organisations, the following step is to examine how the behaviour of ICT actors is defined. The principles that constrain the behaviours of distributed ICT actors are generally defined in the phases of the development process called "requirements elicitation" and "design specification". These phases are creative efforts, made by system designers, to produce computational systems – for example distributed ICT actors – that correctly and efficiently perform the operations required by the user. There are many different approaches to the development of a computational system, including many different methods for software specification and requirements elicitation,[18] but their review is beyond the scope of this paper. For our purposes, it is sufficient to outline only the salient properties of requirement elicitation and specification processes.

At a very general level of abstraction, requirements define the properties of the system from the users' points of view. Requirements may be divided

[17] Muel Kaptein. Business Codes of Multinational Firms: What Do They Say? *Journal of Business Ethics*, 50(1): 13–31, 2004; Simon Webley and Martin Le Jeune. *Corporate Use of Codes of Ethics: 2004 Survey*. IBE, 2005.

[18] Matthew Bickerton and Jawed Siddiqi. The Classification of Requirements Engineering Methods. In Stephen Fickas and Anthony Finkelstein, editors, *Requirements Engineering '93*, pp. 182–186. IEEE Computer Society Press, 1993.

into functional or non-functional requirements, and constraints. Functional requirements describe the behaviours of the system independently of any particular implementation. For example, the functional requirement for an APS is that it must refrigerate the medicines. How it does it is a matter of implementation and obviously there can be several, different implementations for the same functional requirement. Non-functional requirements usually refer to properties of the system that are visible to its users and are not related to the functional behaviours of the system. For example, APSs must be protected from external intrusions so as to avoid dangerous contamination of the medicines they contain. Finally, constraints are pseudo-requirements imposed by the environment in which the system will be deployed. For example, a hospital policy could mandate the obligatory encryption of every patient's data used by the APSs to prevent breaches of the patients' informational privacy.

The process of specification refines the elicited requirements.[19] In this phase, the system is generally decomposed into interconnected components that communicate through well defined interfaces. The specification describes the behaviours performed by these components. For example, a specification of the APS system will define its functional components – unpacking unit, refrigerator, dispenser, waste collector, information management unit, labeller – and how they behave. Informally, a behaviour could describe the unpacking units to take as inputs boxes of dimensions between X and Y, to read the information from the package and to communicate them to the information management unit; and then to discard the package material and send the medicines to the dispenser unit.

Specifications are progressively refined until they are translated into implementations. Following the previous example, materials, software, scanners and all the other components of the APS are actually built and assembled. At this level, decisions are taken following principles of efficiency. Materials, specific algorithms and programming languages are chosen as they are economically feasible, or because they are faster and more durable than others in performing the operations defined by the previous specification.

Finally, the implementation is tested to verify that it correctly and efficiently implements the specifications.

A solution for the ECP

The previous description of how ethical principles constrain individuals and organisations indicates the type of ethical principles that must be used to constrain distributed ICT actors as well, ensuring that the ECP is avoided. Considering the phases of a prototypical system design process, it is clear that the normative constraints, expressed by ethical principles must be introduced at the stage of requirement elicitations and system specification. It is in these phases that the characteristics of the behaviours of the system are defined.

The solution proposed for the ECP assumes:

1. a heterogeneous organisation (i.e. composed by individuals, actors and agents)
2. one or more ethical principles to which this organisation is committed.

The solution is divided into three steps.

1. Translating the normative constraints expressed by the given ethical principles into terms of ethical requirements. An ethical requirement constrains the functionalities of a computational system, thus guaranteeing that one or more properties are maintained during its execution;
2. translating the ethical requirements into an ethical protocol. An ethical protocol specifies the operations performed by the system so that their behaviours match the condition posed by the ethical requirements;
3. refining the specification of the system into executable algorithms.

The translation process described by this solution to the ECP is visually depicted in Figure 1. The schema outlines the ethical consistency of the ethical principles that constrain individuals, actors or agents. Functional requirements and ethical requirements proceed in parallel to converge into a single specification that can then be refined into an implementation.

There is an important difference in how individuals and actors or agents may be bound by ethical principles. Individuals are normatively constrained by ethical principles. These principles indicate how an individual ought to act in specific circumstances, discriminating a right action from a wrong one. For example, in a cultural and social context in which the ethical principle of respect is in place, an individual is normatively bound to act in ways that are considered respectful in that group. Ethical principles are not physically wired into individuals. Individuals can act in disagreement with ethical principles facing the consequences, if any, of acting in a wrong manner.

[19] Usually there is no clear cut division among the different phases of the development process. Requirements tend to evolve during the whole process of development and, analogously, specifications can be revised during the implementation of the system.

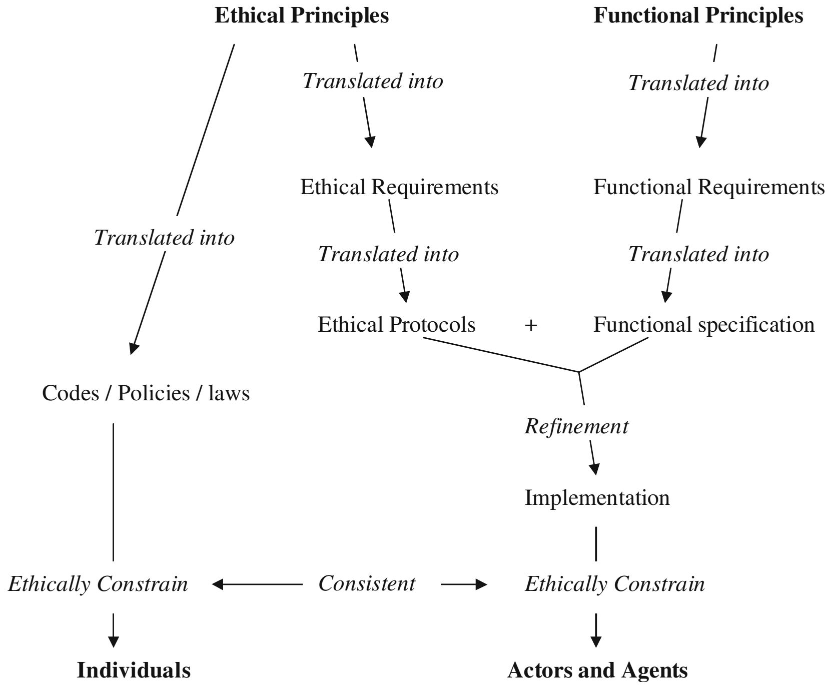

Figure 1. Visual representation of the proposed solution for the ECP.

As it relates to ICT actors (or agents), ethical principles are translated into ethical protocols used to specify actors alongside their functional properties. The normative constraints, expressed by ethical principles, become a precondition of the execution of the operations. If the preconditions of an operation are not matched by the actor's state then it cannot perform that operation in that state. It follows that actors cannot perform their operations outside the boundaries imposed by the ethical principles. This constraint is much stronger than the normative one applied to individuals.

This difference necessitates the careful analysis of the ethical principles adopted to constrain ICT actors. The goal of this analysis is to produce the proper ethical integration among individuals and actors, guaranteeing that the exception-free, ethical behaviours of the former will integrate properly with the 'good will' of the latter. Adam,[20] for example, clearly points out the need of balancing the delegation of morality to ICT actors with the distribution and deletion of sensitive data.

The proposed solution to the ECP requires a tool to translate the ethical principle into ethical requirements and ethical requirements into ethical protocols. The concepts of distributed system and degree of control are defined so as to introduce the Control Closure of an operation (CCop). The CCop is the new tool required to operate the translation.

A distributed system is modelled as a collection of autonomous entities that communicate and coordinate their actions via message passing.[21] Intercommunicating means that the entities of the system are capable of exchanging information. Autonomous means, as in the definition of actor, that the entities have some degree of control over their own actions. The main characteristics of distributed systems are:

1. the concurrency among their components;
2. the lack of a global clock; and
3. the independent failures of their components.

The entities that compose a distributed system can be actors, agents and individuals. However, in this paper, entities refer mainly to the (computational) processes of a software system as, for example, a distributed ICT actor. Every process is identified by a set of variables, called state variables.

The degree of autonomy of a process is proportional to the degree of control that process has over its operations. A process has full control over an operation if it is fully autonomous and then performs its operations without interacting with any other process of the system. For example, an individual is fully autonomous when writing a paper if the paper

[20] Alison Adam. Delegating and Distributing Morality: Can We Inscribe Privacy Protection in a Machine? *Ethics and Information Technology*, 7(4): 233–242, 2005.

[21] George Coulouris, Jean Dollimore, and Tim Kindberg. *Distributed Systems: Concepts and Design*. 4th ed. Addison–Wesley Harlow, 2005.

has no other authors. In this case, the individual has full control of the writing process. A process that is not fully autonomous shares its control, as it needs to coordinate itself with other processes of the system in order to carry out its operations. In a paper with multiple authors, all the authors must agree upon the content of the paper. In this case, all the authors share the control over the content of the paper and are therefore not fully autonomous.

We are now ready to define the control closure of an operation and of a process:

> Control Closure of an operation:
>
> at a given level of abstraction, the control closure of an operation in a distributed system is the set of processes whose state variables are needed to perform the operation. The state variables of these processes provide the values that determine the operation as a relation from the initial to the final state of the system.
>
> The Control Closure of a process:
>
> at a given level of abstraction, the control closure of a process is the union of the control closures of each of its operations.

Assuming an operation OP and a set of processes $\{p_n\}$ the control closure of OP is written as:

$$\mathrm{CCop} = \{p_1, p_2, \ldots, p_n\}$$

The control closure of an operation models the concept of control. It may contain all the processes of the system, as the state variables of all the processes are needed to perform the operation. In this case, all the processes share control over the operation. Alternatively, the control closure of an operation may contain only one process, as the operation can be performed by the process independently, without accessing the state variables of other processes. In this case, the process has full control over that operation. For example, the control closure of the operation of lecturing a classroom performed by a teacher has in its control closure all the individuals of the system. The teacher is needed to explain while the students listen and pose questions. Students and teacher share the control over the operation. Conversely, every student that performs a written exam in a classroom has individual control over that operation. In this case, the control closure of the operation contains only the student himself.

The control closure of an operation supports discrimination between two different specifications of the same operation. The same operation may be performed by a single process in some conditions and by all the system's processes in others. In the former case, the control closure of the operation contains only one process; in the latter, all the system's processes. Consider, for example, cycling. Cycling can be performed either alone or in a team. When cycling is performed in a team, the control closure of the operation contains the state variables of all the cyclists of that group. Every cyclist must be aware of the position of the others. The control closure of cycling contains only one cyclist when it is performed in isolation.

The control closure of an operation and of a process can now be used to discriminate between the degree of distribution (or centralisation) of an operation.

An operation is fully distributed if and only if its control closure is a singleton and the control closure of every process of the system contains the process itself (Figure 2).

An operation is fully centralised if and only if its control closure and that of every process of the system equals the set of all the processes in the system (Figure 3).

An operation is partially decentralised (or not fully centralised) if and only if its control closure contains at least two processes but not all the processes of the system (Figure 4).

There are three limit cases to be taken into account. The first is a system that contains only one process. In this case, the system is centralised and the control closure of every operation may contain only the process itself. The second and third cases involve the empty set. An operation with an empty control closure is performed without access to any state variables. This type of operation results from global constants or inputs that can occur, for example, in specifications with insufficiently elaborated details. For example, as in a clock that outputs its own states but relates to no variables to update them. A process

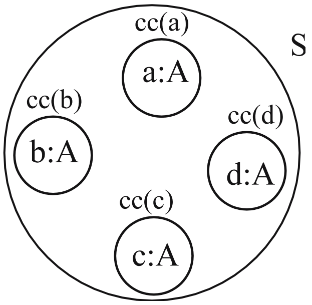

Figure 2. Given a system S with set $A = \{a, b, c, d\}$ of processes, the control closure of the operation OP is a singleton. OP is performed individually by every processes and $\forall\ x{:}A \cdot cc(x) = \{x\}$.

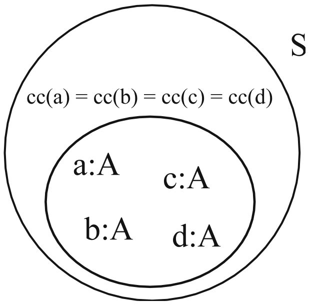

Figure 3. Given a system S with set $A = \{a, b, c, d\}$ of processes, the control closure of the operation OP is $cc(OP) = A$ and $\forall x{:}A \cdot cc(x) = cc\{OP\}$.

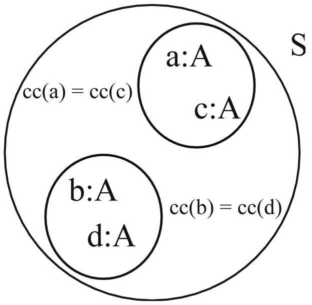

Figure 4. Given a system S with set $A = \{a, b, c, d\}$ of processes, the control closure of the operation OP contains two processes. The processes are divided in two pairs each performing the operation OP.

with an empty control closure is not a process, as it does not perform operations in the system. It can then be considered a constant of the system.

Clearly, the control closure of an operation depends on the level of abstraction that the designer or observer adopts in describing the system. The same system can be considered as a single-process system or as composed of a set of processes. For example, an APS can be described by a single process that takes boxed medicines as input and delivers prescriptions as output. At a different level of abstraction, the same APS can be described by a set of interacting processes that unpack, refrigerate, label, record, and dispense medicine. The control closure of the operation 'deliver a prescription' contains the state variables of one process in the former case, while in the latter case it contains the state variables of the processes of unpack, refrigerate, label, record, and dispense medicine.

The control closure is a particularly useful concept as it can be used at the inception of the design process of the actors and agents. The solution proposed in this paper clearly demonstrates how the ECP ought to be addressed from the early stages of system design.

It is now possible to describe how the solution to the ECP is applied. The solution to the ECP assumes one or more ethical principles to which a heterogeneous organisation is committed. An ethical principle can be decomposed into one or more types of observables, a class of operations and a normative constraint. The decomposed ethical principle may express different types of normative constraints such as the necessity to perform or not perform operations or the exclusive or shared right to perform them. For example, individuals who hold a copyright on one or more products have the right to exclude other individuals from exploiting those products. The set of observables contains the types 'copyright holders', 'products' and 'individuals other than copyright holders'. The class of operations contains all the operations that lead to an exploitation of one instance of the type 'products'. The normative constraint is expressed by 'exclude other individuals from'. Another example is the principle for which no one has the right to kill anyone else. In this case, 'individuals' is the only type of observable and the class of operations contains all the operations that kill an individual. The normative constraint states that none of the operations that belong to that class must be performed by an individual.

Note that no assumptions are made on how the ethical principles and their normative constraints are obtained. They could be derived with a descriptive process, as in the case of descriptive ethics, or they could be formulated inside any theory of conduct. There are also no assumptions about the nature of the ethical principles and normative constraints endorsed. They could be grounded in either universal or relative beliefs, dependant on a situation or derived from observing individuals' behaviours and best practices.

The translation of an ethical principle into an ethical requirement maintains the same types of observables and class of operations and redefines the normative constraint on the operations in terms of control closure. The control closure imposes a precondition on the execution of the operation. This means that the operation can be performed by the system only if the normative constraint is matched. So, in the example of the copyright case, the constraint is translated into a control closure such that, for every operation that exploits a product covered by copyright, the control closure of this operation must contain only the copyright holder. Analogously, in the case of the principle for which no one has the right to kill, the control closure of every killing operation is empty (it imposes an always false precondition) so that no killing operation can be legally performed.

Eventually, the ethical requirement is translated into an ethical protocol. An ethical requirement applies to every system that contains the types of observables and the class of operations singled out by the ethical requirement. On the contrary, an ethical protocol specifies a set of observables and the operations performed by a specific system. These observables, operations, and their control closures belong to types and classes defined by the ethical requirement. In this way, the operations of the ethical protocol are constrained by the same normative constraint translated from an ethical principle into an ethical requirement. For example, consider a system with the musician 'Sergio Mendes', his album 'Timeless' and myself as observables and the operation of receiving money from selling Timeless. The ethical protocol for this system implements the ethical requirement of the copyright, if the control closure of the operation of receiving money from selling the album contains only Sergio Mendes and not myself. This control closure indeed constrains the operation so that only the copyright holder exploits his original product.

Finally, the ethical protocol is refined into an implementation. This process of refinement can have different degrees of formalisation, depending on the chosen process of development. The implementation of an ethical protocol may also require new formalisms and conceptual tools in order to cope with the needs of an executable system. For example, Wiegel, Hoven and Lokhorst,[22] propose to use deontic epistemic action logic to model the information itself as an intentional agent so to preserve the integrity of the information and to regulate its dissemination.

A simple illustration: generalised informational privacy and ethical camera phones

It might be useful to introduce an example to clarify the solution proposed to the ECP. The example is divided into three steps. The first defines the ethical principle to be used in the example, the second translates the ethical principle into ethical requirements and the third translates the ethical requirement into ethical protocols.

First step: the definition of the ethical principle. The analysis of identity theft felony has exposed the crucial role that information may play in jeopardising the economical and social status of individuals. In particular, identity theft is based on malicious access to data regarding the identity of the victim. This access should not be malicious but when this occurs it represents a breach of the right of Informational Privacy (IP) of the victim.[23]

The contemporary debate has produced a definition of IP based on the concept of access as opposed to that of control.[24] IP is the right of the individual of having, in specific circumstances, portions of information relative to herself or himself inaccessible to other individuals. As explained by Floridi,[25] the accessibility of information is an epistemic factor that depends on how actors, agents and individuals interact among themselves and with their environment. The ontological characteristics of entities and environment produce a measure of the 'ontological friction' that allows some exchange of information and therefore varying degrees of informational privacy.

The access to information is disjoint from the access to data. Data *per se* do not yet constitute information, as they need to be interpreted by a subject to produce information. It is possible to access data without extracting information and, vice versa, it is possible to extract information without accessing all the available data. A typical example consists in appropriately encrypted messages that are exchanged through an insecure channel. Whoever intercepts the encrypted messages cannot decrypt it without the corresponding encryption key. So encrypted data accessed without the key cannot be used to extract information. Conversely, given a database and a correlation rule among its data, it is possible to derive information without accessing the whole database. A minimalist example consists of a database that contains three logical propositions correlated by the connectives of the propositional logic. The three propositions are: $\neg A$, $A \vee B$ and B. By accessing $\neg A$ and $A \vee B$ it is possible to infer B without accessing it. More complex examples can be found in problems of distributed epistemic logic as the classic 'muddy children' in which a child that interacts with other children is able to infer whether his forehead is muddy without looking at it.

[22] Vincent Wiegel, Jeroen van den Hoven, and Gert-Jan Lokhorst. Privacy, Deontic Epistemic Action Logic and Software Agents. *Ethics and Information Technology*, 7(4): 251–264, 2005.

[23] IP constitutes only one type of privacy. Psychological privacy is concerned with the effects that intrusions in physical spaces or personal affairs produce on individuals' psyche. Privacy can also refer to the right of privacy as it is codified into laws. For example, in the USA, the constitutional right to privacy establishes that some portion of individuals' life, like the social institution of marriage and the sexual life of married people, are protected zones of privacy. See Judith DeCew. Privacy. *Stanford Encyclopedia of Philosophy*.

[24] Herman T. Tavani and James H. Moor. Privacy Protection, Control of Information, and Privacy-Enhancing Technologies. *SIGCAS Comput. Soc.*, 31(1): 6–11, 2001.

[25] Floridi. The Ontological Interpretation of Informational Privacy.

A comprehensive definition of IP must therefore take into account the distinction between data and information.[26] The right to IP must then be based on limiting the extraction of information instead of the access of data. Thus, the Generalised Information Privacy (GIP) is defined as:

> the right of an individual not to have his or her own data used to extract information without his or her own consent.

This definition (i) discriminates specific cases in which an individual's GIP can be infringed without accessing the individual's data; (ii) separates the unauthorised access to data that do not respect the right to GIP from access that do; and (iii) considers accessing data as one among other operations that allow the extraction of information from data. The GIP is the ethical principle assumed to illustrate the solution to the ECP.

The second step is to translate the GIP into an ethical requirement. The GIP definition assumes two observables – an owner O and his data DS_o – and a set of operations $\{OP_n\}$ such that any operation takes as input DS_o and returns as output the informational content of DS_o.

The normative constraint expressed by the GIP definition is embedded in the statement 'without her or his own (i.e. of the owner) consent'. This constraint can be translated into an ethical requirement using the control closure on the operations of the set $\{OP_n\}$.

> GIP-ER1
>
> For every operation OP_n in $\{OP_n\}$, the owner O of DS_o must belong to the control closure of OP_n, $CC(OP_n)$.

A typical application of this constraint would be to make it impossible for an ICT actor to extract information from customers' data without their explicit consent. The operation 'to extract information' would have a control closure also containing the customer; so a precondition for the execution of the operation would be to have the customer's Boolean variable 'consent' set to one.

This ethical requirement is very strong, as it makes no qualitative distinction about the type of information that is extracted. Every operation that extracts information from DS_o must depend, also but not only, on one or more state variables of the owner O.

A more relaxed ethical requirement for the GIP can be obtained by qualifying the type of information that cannot be extracted from DS_o.

> GIP-ER2
>
> For every operation OP_n in $\{OP_n\}$ that extracts *relevant* information DS_o, the owner O of DS_o must belong to the control closure of OP_n, $CC(OP_n)$.

Consider for example the system that reads from supermarket loyalty cards the customer's identity and associates it to every item that has been purchased. The ethical requirement ER2 could be useful to distinguish between the extraction of information relative to the identity of a customer and that relative to his shopping behaviours. The extraction of the former would require the customer's consent, while the extraction of the latter would be accessible without his explicit consent. This distinction of the quality of the information extracted is not allowed by the ethical requirement ER1.

The third and last step of the proposed example is to derive an ethical protocol from the GIP ethical requirement. Camera phones serve as useful examples of how ethical protocols might be implemented when the operation of taking a picture is constrained in accordance with the ethical requirement of GIP. Note that this is just an example used to illustrate the proposed solution for the ECP, not a blueprint of how to design a real camera phone.

Camera phones are very popular and most mobile telephones are equipped with a camera to take pictures and to record videos. Invented in 2000, in 2003 already 84 million camera phones were sold worldwide, exceeding the numbers of stand-alone cameras.

Camera phones offer an unprecedented overlap between capturing images and videos and distributing them. Moreover, mobile phones are becoming increasingly functional today, with the majority of them capable of accessing the Internet. They can therefore be used as massive, virtually immediate publication tools for pictures and videos taken wherever and whenever mobiles can be used.

Horror stories of privacy breaches and camera phones abound. Clandestine lovers spotted in foreign cities, a teenager acting in revenge sending private pictures of his ex-girlfriend to all his social network, thus making her life very difficult, an unhappy customer who embarrasses the retailer of a used mobile phone by publishing all the private information found in the memory of the phone on the internet. There are good exceptions too. Men arrested because filmed roughing up a man on the street, journalist scoops taken on the spot thanks to camera phones, emergencies handled thanks to the information gathered from pictures taken and sent by victims.

[26] Luciano Floridi. Is Semantic Information Meaningful Data? *Philosophy and Phenomenological Research*, 70(2): 351–370, 2005.

Camera phones are forbidden in many places in order to protect the identity of customers and employers or to protect secrets that could be stolen by a picture. Gyms, night clubs, government offices and high tech companies are all places where the use of a camera phone is occasionally forbidden.

The debate about the difficulty in maintaining privacy as it relates to taking pictures is as old as photography. In 1890 Warren and Brandeis[27] lamented that 'instantaneous photographs [...] have invaded the sacred precincts of private and domestic life'. In more recent years, there have been proposals to change the design of camera phones so as to protect the privacy of those photographed better. The journey towards an ethical camera phone has officially begun.

The system in which a camera phone operates can be modelled by two individuals and an operation. One individual *I* operates a camera phone making it perform the operation *TP* of taking a picture. The other individual *O* is the subject of the picture or, in informational terms, the owner of the information that is recorded by the phone into its memory.

Three different versions ($TP_{1\text{–}3}$) of the operation of taking a picture can be used to illustrate how to evaluate whether the ethical requirement of the GIP has been correctly translated into an ethical protocol. Recall the ethical requirement derived from the ethical principle of the GIP. The observables and operations present in that definition can be directly translated into the system of the camera phone:

> CP-GIP-ER
>
> Given the operation TP that extracts *relevant* information DS_o, the owner O of DS_o must belongs to the control closure of *TP*, *CC*(*TP*).

1st version of *TP*

- Description: The operation TP_1 is constrained only in terms of efficiency and correctness. A picture can be taken by pressing a specific button B of the phone.
- Protocol: Pressing the button B, a picture is saved in the phone's memory.
- Control closure of TP_1: $CC(TP_1) = \{phone, user\}$

This is the minimalist design of the operation *TP*. The control closure of TP_1 does not contain the subject of the picture *O* as the operation does not depend on any of *O*'s state variables. This implementation of the operation *TP* is functionally correct but does not match the ethical requirement *CP-GIP-ER*. It follows that it does not implement an ethical protocol for the operation of taking pictures that respects the GIP.

2nd version of *TP*

- Description: The operation TP_2 is more constrained than TP_1. When a picture is taken the phone emits an audible noise.
- Protocol: on pressing button B, an audible noise is produced and a picture is saved into the phone's memory.
- Control closure of TP_2: $CC(TP_2) = \{phone, user\}$

The emission of a camera-like noise is a mandatory function for camera phones sold in Korea and Japan. Nonetheless, the control closure of TP_2 reveals that this design does not respect yet the ethical requirement of GIP. In this case too, the control closure of TP_2 does not contain the subject of the picture *O* as the operation does not depend on any of *O*'s state variables. Nothing has changed from TP_1. The respect of the privacy is fully outsourced to *O*, who has to react to the noise and defend his right to GIP.

3rd version of *TP*

- Description: The operation TP_3 is always constrained by the subject being photographed. The phone takes a picture only if the subject allows it to do so.
- Protocol: Pressing button B connects to the subject's policy device, if policy = allow, a picture is saved in the phone's memory.
- Control closure of TP_2: $CC(TP) = \{\text{phone, user, subject}\}$

This is a similar solution to the one proposed by Iceberg Systems in 2003 for inhibiting the use of camera phone in areas protected by privacy devices. In this scenario, the phone becomes an actor that communicates autonomously with the subject's privacy device (for example another phone) and reacts to it by taking a picture or not. The control closure of TP3 contains the subject wearing the policy device, as the execution of the operation depends also on its state. This approach respects the ethical requirements as the subject can activate the privacy device only in situations in which relevant information could be photographed. TP3 implements an ethical protocol for the GIP.

Conclusions

Recall our initial concern. Nowadays, a system designer is the demiurge of the artefacts he produces.

[27] Samuel Warren and Louis D. Brandeis. The Right to Privacy. *Harvard Law Review*, 4(5): 193–220, 1890.

When he decides to design autonomous and reactive artefacts, it gives them an ethical dimension. The solution to the ECP, proposed in this paper, is an analytical tool meant to facilitate the ethical design of artificial actors. The tool can be used to analyse existing systems – so as to evaluate whether they effectively implement a given set of ethical principles – or it can be used in designing new systems from scratch – in order to guarantee the effective implementation of a given set of ethical principles.

The solution proposed for the ECP is especially useful in designing distributed systems for large organisations with explicit ethical commitments. Typical examples could be hospital and bank systems, in which rights such as confidentiality, anonymity and privacy are crucial both for the user and for the institution's image.

This paper offers a normative complement to the greater endeavour of research into ethical requirements elicitation. Ongoing work will delve into how best the descriptive elicitation of ethical requirements may be conjugated with the necessity of assuming a set of ethical principles to derive a normative analysis of how the system would have to behave.

The problem of integrating ethical actors and ethical individuals deserves further investigation. In this paper, it has been assumed that actors have to behave as individuals would behave in the same situation. However, this is only one form of consistency and several questions might easily be raised. What are the criteria to decide which ethical principles can be implemented in artificial actors and which ones must be left to the good will of individuals? How can it be decided when it is preferable to have ethically neutral actors, thereby leaving the ethical dimension completely to individuals? And, conversely, are there situations in which it would be better to have ethically neutral individuals, leaving any ethical accountability to ethical actors? Finally, is it possible to ascribe to artificial actors or agents some form of morality and/or the capability of moral reasoning?[28] These are all important questions that will need to be addressed in the close future.

Acknowledgments

This research has been partially funded by a PRIN project through the Università degli Studi di Salerno, and I wish to thank Roberto Cordeschi for his constant support. A previous version of this paper was presented at the *European Conference Computing and Philosophy* (E-CAP, NTNU, Trondheim, 22–24 June, 2006). I wish to thank all the participants, and especially Dan Burk, for their useful feedback. I also wish to thank all the members of my research group (IEG) for their comments and Luciano Floridi and Jeff Sanders for their crucial help in developing some of the ideas presented in this paper. Of course, I remain the only person responsible for any of its shortcomings.

References

W.M.P. Alast. Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management. *Lecture Notes in Computer Science*, 3098: 1–65, 2004.

A. Adam. Delegating and Distributing Morality: Can We Inscribe Privacy Protection in a Machine? *Ethics and Information Technology*, 7(4): 233–242, 2005.

C. Allen, I. Smit and W. Wallach. Artificial Morality: Top–Down, Bottom–up, and Hybrid Approaches. *Ethics and Information Technology*, 7(3): 149–155, 2005.

A. Arbel and A.E. Kaff, *Crash: Ten Days in October. Will It Strike Again?* Longman Financial Services, Chicago, 1989.

M. Bickerton and J. Siddiqi. The Classification of Requirements Engineering Methods. In S. Fickas and A. Finkelstein, editors, *Requirements Engineering '93*, pp. 182–186. IEEE Computer Society Press, 1993.

G. Coulouris, J. Dollimore and T. Kindberg, *Distributed Systems: Concepts and Design*. 4 ed. Addison–Wesley, Harlow, 2005.

J.C. Cunha and O. Rana, *Grid Computing: Software Environments and Tools*. Springer, London, 2006.

J. DeCew. Privacy. In E. N. Zalta, editor, *Stanford Encyclopedia of Philosophy,* 2006.

R. Fitzpatrick, P. Cooke, C. Southall, K. Kauldhar and P. Waters. Evaluation of an Automated Dispensing System in a Hospital Pharmacy Dispensary. *The Pharmaceutical Journal*, 274: 763–765, 2005.

L. Floridi. Is Semantic Information Meaningful Data? *Philosophy and Phenomenological Research*, 70(2): 351–370, 2005.

L. Floridi. The Ontological Interpretation of Informational Privacy. *Ethics and Information Technology*, 7(4): 185–200, 2006.

L. Floridi and J.W. Sanders. The Method of Abstraction. In M. Negrotti, editor, *Yearbook of the Artificial Nature, Culture and Technology*, pp. 177–220. P. Lang, Bern, 2004.

L. Floridi and J.W. Sanders. On the Morality of Artificial Agents. *Minds and Machines*, 14(3): 349–379, 2004.

R. Graham. Robots Benefit Patients and Staff in Hospitals and Community Pharmacies. *The Pharmaceutical Journal*, 273: 534, 2004.

[28] Colin Allen, Iva Smit, and Wendell Wallach. Artificial Morality: Top–Down, Bottom–up, and Hybrid Approaches. *Ethics and Information Technology*, 7(3): 149–155, 2005; Floridi and Sanders. On the Morality of Artificial Agents.

G.M. Greco, G. Paronitti, M. Turilli and L. Floridi. How to Do Philosophy Informationally. *Lecture Notes in Artificial Intelligence*, 3782: 623–634, 2005.

D. Gross. Attack of the Machines. Is Your Stockbroker a Robot? *Slate*, Jan. 18, 2005.

T. Hawkes, *Structuralism and Semiotics*. 2 ed. Routledge, London, 2003.

C. Hewitt, P. Bishop and R. Steiger. A Universal Modular Actor Formalism for Artificial Intelligence. *IJCAI3*, pp. 235–245. Stanford, CL, 1973.

M. Kaptein. Business Codes of Multinational Firms: What Do They Say? *Journal of Business Ethics*, 50(1): 13–31, 2004.

M.T. Özsu and P. Valduriez, *Principles of Distributed Database Systems*. 2 ed. Prentice Hall, London, 1999.

J. Penycate. Identity Theft: Stealing Your Name. *BBC News*, Monday, 18 June, 2001.

S. Ranger. Bank Automates to Boost Customer Service. Case Study: HSBC Speeds up Queries with Workflow Automation. *Silicon.com*, Monday 06 February, 2006.

Synovate. *Identity Theft Survey*. Federal Trade Commission, 2003.

H.T. Tavani and J.H. Moor. Privacy Protection, Control of Information, and Privacy-Enhancing Technologies. *SIGCAS Computer Society*, 31(1): 6–11, 2001.

M.M. Waldrop. Computers Amplify Black Monday. *Science*, 238(4827): 602–604, 1987.

S. Warren and L.D. Brandeis. The Right to Privacy. *Harvard Law Review*, 4(5): 193–220, 1890.

S. Webley and M. Le Jeune. *Corporate Use of Codes of Ethics: 2004 Survey*. IBE, 2005.

V. Wiegel, J. Hoven and G.-J. Lokhorst. Privacy, Deontic Epistemic Action Logic and Software Agents. *Ethics and Information Technology*, 7(4): 251–264, 2005.