Normative Multi-agent Approach for Socio-physical Computing - An Application to Distributed Tangible Interaction

Fabien Badeig, Catherine Garbay, Valentin Valls

NORMATIVE MUTLI-AGENT APPROACH FOR SOCIO-PHYSICAL COMPUTING

An Application to Distributed Tangible Interaction

Fabien Badeig[1], Catherine Garbay[1] and Valentin Valls[2]
[1]*AMA*
[2]*MultiCom*
first_name.last_name@imag.fr

Keywords: Ambient Intelligence, Tangible Distributed Interfaces, Activity Theory, Norms, Multi-agent system

Abstract: We propose a new vision of normative multi-agent systems for computer-supported collaboration in the framework of socio-physical computing. An example application to the RISK game, in the context of the TangiSense distributed tangible environment, is provided to support the proposed design. Our work is driven under three complementary views: (i) a systemic view, to integrate in a single modelling a wide range of levels, from the very physical infrastructure level to the higher social level of human coordination, (ii) a normative view, in which norms are introduced to mediate human activity, in accordance with the principles of activity theory, and (iii) a trace-based view, in which traces, left in the information space by an action, are made to evolve under the application of norms. In this way, norms do not act as a prerequisite, or as a way to apply a priori constraints on action. Rather, they are meant to situate action, by evaluating norm-dependent properties that may then be considered by other agents, in proper contexts. We further distinguish between application-dependent and communication-dependent norms, which respectively check the correctness of action and delimitate proper viewpoints in the information space. We show through simple examples the potential richness of the proposed concept.

1 INTRODUCTION

This research is conducted in the framework of a project called IMAGIT (Lepreux et al., 2011), whose objective is the management of distant interactive surfaces supporting tangible and virtual objects (the TangiSense table). Human activity (e.g. musical creation as well as crisis mangement) involves the handling of tangible objects, when considering a single table. Communication between distant tables is managed via virtual objects displaying the current status of the tangible original objects. Human collaboration in this context is mediated rather than assisted by complex computerized systems; it is regulated rather than constrained. Our guiding principle is therefore to preserve the spontaneity of human action by designing ecological working environments. Our work is driven under (i) a systemic view, integrating in a single modelling the very physical infrastructure level as well as the higher social level of human coordination, (ii) a normative view, in which application-dependent and communication-dependent norms are introduced to regulate human activity, and (iii) a trace-based view, to record and transcribe human activity according to the various systems of norms under consideration. These traces are made to evolve under the application of these norms. In accordance with the principles of activity theory, norms do not act as a prerequisite, or as a way to apply a priori constraints on action. Rather, they are meant to situate action, by evaluating properties that may then be considered by other agents, in proper contexts. An example application is used, all along the paper, to illustrate the proposed concepts. The Risk game is a strategic board game, whose objective is to dominate the world, that is to occupy territories and eliminate other players. Players control armies with which they attempt to capture territories from other players, with results determined by dice rolls. We first of all present the application framework, in terms (i) of the infrastructure at hand and (ii) of the precise collaborative activity (RISK game) that is considered as example. We then present the proposed approach, in the line of both normative agents and activity theory state of the art. We present the architecture model and discuss some implementation details in section 4 and finally conclude in section 5.

2 APPLICATIONFRAMEWORK

2.1 The TangiSense Infrastructure

The TangiSense table (Kubicki et al., 2009) may be seen as a magnetic retina (Figure 1), which is able to detect and locate tangible objects equipped with RFID tags. The table consists of 25blocks containing 64antennas and readers of one square inch each. Each block of antennas further contains two microprocessors, one dedicated to the management of the RFID readers and another one to the handling of Ethernet UDP communications. The density of antennas allows a spatial and temporal resolution compatible with the real-time detection of moving objects by users. The blocks are driven by the Communication and Interface software layer (CI) running on the PC host. The role of this Java layer is to send and receive UDP datagram's, to filter potentially unstable tags IDs and positions, or as well to aggregate information in case a tag is crossing between two blocks. Each tangible object, when put on the table, is associated a dedicated Java object instance.

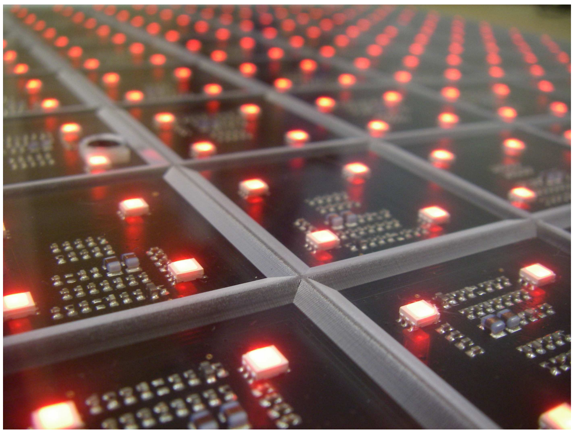

Figure 1: The TangiSense table: a magnetic retina made of 25 blocks containing each 64 antennas and readers.

When RFID tags are pasted underneath tangible objects, it is possible (i) to detect their presence on the surface of the table, (ii) to proceed to their identification, since RFID tags are unique and (iii) to store relevant information directly in the object tags memory (for example their last position, or their last owner). Each RFID antenna is further equipped with 4 multicolor light emitting diodes (leds). These diodes are associated to Java object instances at the software level. When lit, they may be considered as virtual objects displayed on the table (as in the example of Figure 2). The primary role of these diodes is to have feedback or to react to tangible objects positioning and moves, assessing for the user their effective detection by the table. Their secondary role is to provide additional information to the user, from the mere assessment of action correctness, to the display of color pattern enriching the feedback to the user by indicating places where events occur. Virtual objects may also be projected onto the table using a video projector placed vertically above the table if necessary for full display. Several applications have been designed based on this table, involving the recognition and learning of colors by children (Figure 2), musical creation, road traffic simulation and crisis management.

Figure 2: The TangiSense table used in an application to the recognition and learning of colors by children.

2.2 The distributed Risk Game

Initially Risk game is a strategic board game for two to six players. Risk is a turn-based game. The standard version is played on a board depicting a political map of the Earth, divided into forty-two territories, which are grouped into six continents. Each player begins with a set of territories where he positions their armies. The player objective is to occupy every territory on the board and in so doing, eliminate all other players. Players control armies with which they attempt to capture territories from other players, with results determined by dice rolls.

In a distributed numeric view, the board is a interactive table where the map is cloned. Several persons can play around the table and each player has their own armies identified by a specific color. Players are known by all the other distant players. The virtual images on a table shows the armies of a distant player. The difficulty is to enable an intuitive game process despite the distributed context and to keep a consistent vision of the game for all players.

3 STATE OF THE ART

3.1 Normative agents

To design collaborative support systems, a major issue is to preserve the spontaneity and fluidity of human activity while ensuring the consistency and proper coordination action (Pape and Graham, 2010). COIN (Coordination, Organization, Institutions and Norms in Agent Systems) community introduces the notion of norm in a complex agent organizations as a way to cope with the conceptual antagonism between autonomy and control; it allows to approach coordination as a social paradigm. Behaviour in such agents is not only guided by their mere individual objectives but also regulated by norms specifying which actions are considered as "legal" or not by the group. Norms are specified in a declarative way, they may be adopted or not by the agents according to their role in the organization, and adapted to cope with the evolution of context (Boella and Torre, 2006; Boissier et al., 2011).

In multi-agent system, a multitude of agents interact, usually with some intended individual or collective purpose. Such a view usually assumes structures that articulate or restrain interactions in order to make them more effective in reaching those goals, more certain for participants or more predictable. One major issue (Boissier et al., 2011) is to cope with the various interaction modes in the complex organization according to its structural and functional specifications. The organizational model MOISE (Hubner et al., 2002; Hubner et al., 2010) has been extended to specify the interaction modes of agents belonging to the same organization (Boissier et al., 2011). The goal of such specification is to allow the checking by the organization that interaction modes are used appropriately and to allow the agents to reason on these interaction modes like they do with norms.

A second major issue is how to maintain consistency, especially in contexts where human actors do not know each other, are communicating from distant places, and may display with opposite or conflicting goals. An application to the control of multiplayer computer games has been studied in (Gâteau et al., 2006). The purpose is to constrain players and their avatars to adopt a team behaviour and to respect rules, while allowing some autonomy to keep the game appealing. One further requirement concerns the evolution of the game, since rules change according to rounds of the game. The proposed design articulates two layers using a normative organisational model: the multi-agent interactive game in which avatars evolve as autonomous agents, and an institutional multi-agent middleware called SYNAI (SYstem of Normative Agents for Institution) dedicated to the management of the organisation and to the arbitration. The role of this arbitration system consists in rewarding or punishing agents when they respect or not their agreements. Finally mention is made in (Okuyama et al., 2008) of a distributed and situated approach to normative design. The proposed normative infrastructure is composed of normative objects and normative places, and further allows the spatial contextualisation of norms. We are in someway in line with that, since we have norms attached to tangible objects, with different types of norms depending on the type of object.

3.2 Activity theory

Activity theory articulates within a single dynamics the individual, the group and its organization on one hand, the subject, the object and the tool, on the other hand. According to this theory, the tool supports and limits activity, it mediates its structure and objectives, it carries the history of the relationship between the subject and the object (Bourguin et al., 2001). In turn, it is transformed and built along the activity and therefore keeps track of the user experience. Individual and group activity co-evolve in a context-dependent way, they are driven according to certain rules, norms and conventions, and depend on the actors roles and resources, as well as their organisation. The object and motive reveal themselves only in the process of doing. The "tool" is at the same time both enabling and limitating: it empowers the subject in the transformation process with the historically collected experience and skill "crystallized" to it but it also restricts the interaction to be from the perspective of that particular tool or instrument only - other potential features of object remain "invisible" to subject (Kuutti, 1995). The object is seen and manipulated not as such but within the limitations set by the instrument (Engestrom, 1991).

"A central tenet of the situated action approach is that the structuring of activity is not something that precedes it but can only grow directly out of the immediacy of the situation" (Nardi, 1996). The involvement in action create circumstances that could not anticipated in advance (). According to (Engestrom, 1987), an activity system includes participants of that activity (subject), their goals and intentions, and objects or products that are being transformed, the tools that are used to this end), the rules and norms that circumscribe activity, the larger community in which the activity occurs, and the negotiation of roles and responsibilities (division of labor). Activity theory recognizes the dynamic nature of context where the components of activity such as tools, goals, norms, and rules are constantly changed, constructed, and transformed in relation to the outcome of the activity system (Cole, 1996; Greenberg, 2001). In activity theory, a tool can be physical, mental, or semiotic in that it can be a physical object that the individual can use to transform another object, it can be an heuristic that one follows to transform an object, or it can be a speech act that transforms a situation (Cole, 1996).

3.3 Normative approach based on the traces

A normative view is proposed, to preserve the spontaneity and fluidity of human activity while ensuring the consistency and proper coordination of action, but also as a way to approach the social dimension of action. Such view has to be adapted to the presence of (i) tangibility, with consequences as regards the proper regulation and coordination of action and (i) physical distribution, with consequences in terms of distant communication and privacy.

Human activity in the context of the proposed design involves the handling of physical, tangible objects that is performed under specific application- (or game-) dependent rules - with the limitation that human gesture obeys intentions that may not be circumscribed within a set of norms. Physical distribution implies that human action be transmitted to the distant actors of the game: such transmission, depending of the application, may be performed under control of privacy rules. It implies that human activity be "traced", registered and displayed for the distant observer. Finally, the distant aspect of communication implies robust coordination for the sake of consistency; the spontaneity of gameplay implies that coordination may not be formalized by a priori rules; rather, like turn-taking in a conversation, we propose that it be materialized by dedicated tangible objects called tangigets and digitized as numerical traces.

Human action is therefore mediated via tangible objects, registered via numerical traces and regulated under three normative universe governed by separate rules: the one of the game (or application), the one of communication, and the one of coordination.

We propose that norms be activated and operated through traces deposited in the system environment and reflecting the handling of tangible objects. Traces of activity have been proposed by several researchers as a way to represent, share and visualize human experience in its interaction with numerical platforms. Interaction traces have further been explored to enhance synchronous collaboration, and sharing traces at a group level has been advocated to support group awareness (Clauzel et al., 2011).

Traces may further be considered as reflecting beyond human activity its relationship to the norms under considerations. More precisely, the idea is to enrich the trace by its relation to the norm. In this way, the result of applying a norm is made visible not only as a compliance to obligations or permissions but as a set of signs, revealing potentially complex relationships and facets, that will afterwards be interpreted, possibly in different contexts by different actors. We therefore allow multi-actor expression and more flexibility. In addition, it is a way to preserve modularity and independence, since the norm is specified as operating on traces rather than on actions.

In this way, norms are not a prerequisite to action or a way to prevent action but rather a prolongation, an extension. It reveals in the course of action and enriches the traces at hand by various indicators. According to activity theory, the various levels of action are considered as a dynamics involving a broadening scope of action rather than isolated units obeying separate laws. The idea is to equip tangible objects, that are used under normal conditions, with norms that will sustain (prolonger) their actions.

La norme est porte par objects tangibles. Elle se rvle doublement, localement par retour au joueur (alerte locale) et au niveau social par transmission ou non. On enrichit la trace et on rgule laccs aux traces (fenłtre glissante sur la trace). Linscription de lactivit est une trace dobjet. On joue au niveau perception, sur ce qui est transmis, on ne lgifre pas (?).

4 PLATFORM SPECIFICATIONS

4.1 Layered architecture of the distributed platform

Figure 3 shows the general functional description of the distributed platform architecture. The current architecture is a 3-layered architecture : Hardware, Middleware and Software layer.

The first one is composed of two sublayers : communication interface layer and the toolkit table layer. The 25 networked blocks are driven by the Communication Interface middleware layer (CI) running on the computer host. It deals with the RFID events. The role of this Java layer is to send and receive UDP datagram's, to filter potentially unstable tags IDs and positions, or as well to aggregate information in case a tag is crossing between two blocks. Each tangible object, when put on the table, is associated a dedicated Java object instance. The CI layer is also designed

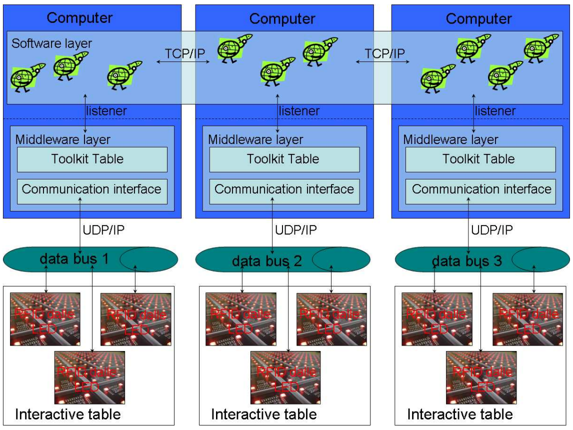

Figure 3: The functional description of the distributed platform architecture.

to configure the blocks at table power up. TCP network transport protocol is implemented but still not used now, since UDP reveals faster and no conflict issues are to be faced, with the table blocks connected to the PC host on a local private network. Moreover, this layer ensures the control of diodes on each block. These diodes are associated to Java object instances at the middleware level. When lit, they may be considered as virtual objects displayed on the table or a land for the risk game. Moreover, an tangible object can be a composition of several tag. Therefore, an layer is necessary to make the object representation on the table. This is the objective of the TT layer. This layer access to a structural object database which details each object of the application with the tag and landmark representation. Four tag event is identified by the middleware layer : the new tag apparition (corresponding to a new object on the table), the enter of a detected tag on the table (corresponding to an object on the table), the leave of a detected tag on the table (corresponding to an leave of an object on the table) and the move of a detected tag on the table. The middleware layer is implemented thanks to JAVA programming language.

The second main layer is the software layer based on the multi-agent programming. Each tangible object is equiped of an agent model instanciation. So the software layer specifies an event behavior for each agent model instanciation. We use JADE programming to develop and run our multi-agent system. JADE () is a software Framework fully implemented in Java language simplifying the integration with the middleware. JADE is particulary adapted for the distributed environment. Indeed JADE have primitives for the inter/intra platform communication. Thus, the distributed software layer can be seen like a *continuous layer* where each agent communicates indifferently with the other local or distant agents.

4.2 Agents

The agent organization reflects the structure of the environment in which the human actors are meant to evolve, as shown in Table 1. In our case, this environment is made of interactive surfaces and tangible objects. We therefore distinguish between Interactive Surface Agents (ISA): these agents may be application-dependent (overall management of the game for example) or communication-dependent (led enlightenment for example); agents at the level of a set of surfaces may further be introduced, if necessary to regulate the whole application at hand; Tangible Object Agents (TOA): tangible objects agents may be application-dependent (a given game token, the avatar of a player, for example), or communication-dependent (a way to ensure turn-taking for example). Application-dependent agents may obey a further social organization, if needed to formalize the relationships between tangible objects and human actors;

Agents	Application	Communication
Surfaces	ISA	ISA
Objects	TOA	TOA

Figure 4: The agent organization.

These agents are activated via filters they deposit in the normative space. They process the corresponding trace information depending on their role. The result of such process is in the form of information projected on given tables and/or trace modification (enrichment of the trace, updating of property values). The level (eg infrastructure, local or global) at which they operate merely depend on the level of the trace they manage.

4.3 Traces

Like agents, traces reflect the structure of the human environment. We therefore distinguish between interactive surface traces (IST) and tangible object traces (TOT). Any trace is considered as a set of tuples (property, value), the precise syntax depending on its role. We propose that properties be typed, to account (i) for their private or public character and (ii) for the status of the trace, i.e. new, modify, valid or invalid property status. As a consequence, a trace is expressed as follows: $trace = \{(p,v)\}$ with $p = name : status : privacy$

- $status \in \{new, modify, valid, invalid\}$. The value *new* means the property does not exist before and it has to check its consistency; The value *modify* indicate a property modificaton by a extern source; the values *valid* and *invalid* results to the validation trace process.

- *privacy* $\in \{private, public\}$. When the slot privacy is *private*, the property is not accessible by other distant agents. Otherwise, the property is readable.

4.4 Filters

Filters formalize norms that may be application or communication-dependent: as a simple example, a communication-dependent filter will regulate turn-taking and ensure that no moves are made that would not respect this rule; an application-dependent filter will ensure that the player army is moved according to proper game rules. To this end, filters operate on dedicated trace properties (privacy properties in particular). Since tangible object handling may not be prohibited, the solution is that the privacy property be moved to private in case of a violation, so that the action is kept private to the player. In addition, a specific display (led enlightenment) may provide feedback to this player. More generally, filters may be seen as regulating the spheres of perception and activity of agents by modulating the trace content and launching agents on dedicated parts of the traces.

4.5 Overall Architecture

The proposed design is finally characterized by:

- dedicated relationships between traces, agents and filters; filters mediate the relation between agents and traces (Figure 5); traces are modified for normative as well as application or communication-dependent reasons;
- a seemless integration between elements of the physical, numerical and human world

4.6 An Application to the RISK Game

We present a risk scenario to illustrate the previous concepts. The scenario consist of identifying the new player on the interactive table. When a new player wants to join a game, he has to put the player tangible card on the table surface. In our case study, we distinguish between 5 agents types : Interface (IA), Gameplay (GA), Player (PA), Tangiget (TGA), Tangible object (TOA). The role of these various agents is as follows :

- Gameplay Agent: to manage the gameplay in its various phases (initialization, game, strategy study, end)
- Interface Agent: to maintain a representation of visual information on the table

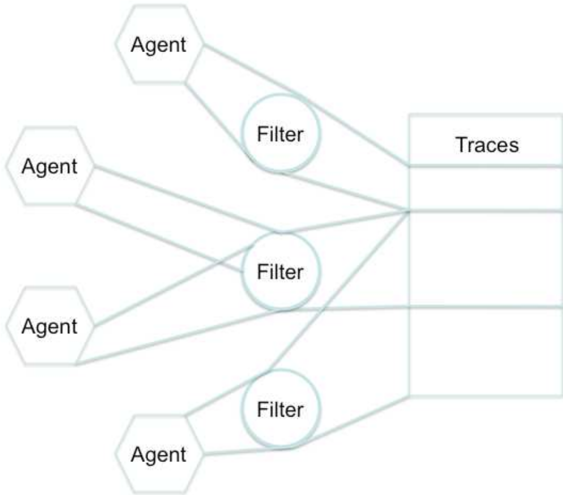

Figure 5: Filters as mediating the relationship between agents and activity traces.

- Player Agent: to maintain information about the player (id, turn taking, resources, points, ...)
- Tangiget Agent: to maintain information about a given tangiget (id, position, status, history of manipulations)
- Tangible Object Agent : to maintain information about a given tangible object (id, position, role, parent player, ...)

In the following, we depict the identification process of a new player. When a player dropes a card with player information on an interactive table, the card detection generates a tangible object trace in the local information space, noted:

$$
\begin{aligned}
t_{player} = \{ \quad & (surname : new : private, Dupont), \\
& (name : new : private, gamer1), \\
& (tag : new : private, 1), \\
& (position : new : private, (2,3)), \\
& (table : new : private, \{1\}) \\
& (type : new : private, (tot, player)) \\
\}
\end{aligned}
$$

The fiel *name* and *surname* is the player description in game; the field *position* is the card position on the table and the field *table* is the table situation of player. The field *tag* is specific to the RFID detection and its value is unique. At the generation time, all properties of trace is initialized with the extension type *new* and *private*. For a property, the slot *new* involves the property is just created and the slot *private* deals with the property accessibility.

When a new tangible object is detected on the table or is modified, a verification process is triggered to check the trace consistency. Two validation filters (f_{valid}, $f_{invalid}$) manage this process. Here, a new player can join the game if the number of players is less than 6. A role of GA is to keep game information in a specific trace written:

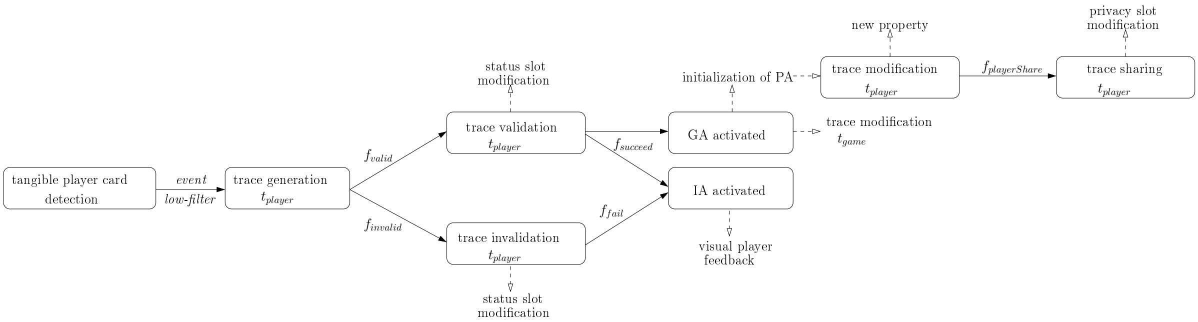

Figure 6: Flexible process between the trace evolution and the filter triggering.

$$
\begin{aligned}
t_{game} = \{ \quad & (id : valid : public, 1),\\
& (number_{player} : valid : public, 3)\\
& (color_{available} : valid : public, \{blue, green, red\})\\
& (type : valid : public, (ist, game))\\
\}
\end{aligned}
$$

The field $number_{player}$ shows the number of players in the game. Like the slot of properties is to $public$, this trace is accessible for all agents. The filter f_{valid} is written:

$$
\begin{aligned}
f_{valid} \quad = \quad & [type.status(trace_1) = new]\\
\wedge \quad & [type(trace_1) = (tot, player)]\\
\wedge \quad & [type(trace_2) = (ist, game)]\\
\wedge \quad & [number_{player}(trace_2) < 6]\\
\Rightarrow \quad & \forall\ property\ p_i\ of\ trace_1,\\
& p_i.status(trace_1) \leftarrow valid
\end{aligned}
$$

When this filter is triggering, the associated action is the slot update of trace properties from *new* to $valid$. Thus, the new trace properties are well validated.

After the process of trace validation , several game process occur like visual player feedback, creation of player agent, ... Each process is activated thanks to a filter. For example, the visual feedback process is ensured by two filters $f_{succeed}$ and f_{fail} which activate tha agent IA in a specific context. The filter $f_{succeed}$ is written:

$$
\begin{aligned}
f_{succeed} \quad = \quad & [position.status(trace) = valid]\\
\wedge \quad & [position(trace) = ?pos]\\
\wedge \quad & [type(trace) = (tot, player)]\\
\wedge \quad & [state(trace) = null]\\
\wedge \quad & [table(trace) = ?t]\\
\Rightarrow \quad & inform(IA, ?t, succeed, ?pos)\\
\& \quad & inform(GA, ?t, trace)
\end{aligned}
$$

When the filter $f_{succeed}$ is triggered, tha agent IA is activated in a specific context which is the successful to join the game for a player. The result is a green circle under the tangible player card. Moreover, the second action of $f_{succeed}$ is to activate the agent GA to inform that the trace t_{player} is valid. Then, GA create a player agent that equipes the tangible player card link to the trace t_{player}. When PA is initialized, GA increments the value of property $number_{player}$ of t_{game} and PA adds a property $state$ with value $initialization$ to the trace t_{player}. Thus, t_{player} is written:

$$
\begin{aligned}
t_{player} = \{ \quad & (surname : valid : private, Dupont),\\
& (name : valid : private, gamer1),\\
& (tag : valid : private, 1),\\
& (position : valid : private, (2,3)),\\
& (table : valid : private, \{1\})\\
& (type : valid : private, tot)\\
& (state : valid : private, initialization)\\
\}
\end{aligned}
$$

When PA is initialized and the associated trace t_{player} is updated, a part of information trace has to be shared. This process is ensured by a filter $f_{playerShare}$. The condition to share information is that the trace is valid and the agent activity is on initialization (property $state$). The filter is written:

$$
\begin{aligned}
f_{playerShare} \quad = \quad & [type.status(trace) = valid]\\
\wedge \quad & [type(trace) = (tot, player)]\\
\wedge \quad & [state.status(trace) = valid]\\
\wedge \quad & [state(trace) = initialization]\\
\wedge \quad & [surname.status(trace) = valid]\\
\wedge \quad & [table.status(trace) = valid]\\
\Rightarrow \quad & surname.privacy \leftarrow public\\
\& \quad & table.privacy \leftarrow public\\
\& \quad & type.privacy \leftarrow public\\
\& \quad & state.privacy \leftarrow public
\end{aligned}
$$

When the filter is triggered, the associated action is the modification of privacy slot of several t_{player} trace properties. Then t_{player} is updated to:

$$
\begin{aligned}
t_{player} = \{ \quad & (surname : valid : public, Dupont),\\
& (name : valid : private, gamer1),\\
& (tag : valid : private, 1),\\
& (position : valid : private, (2,3)),\\
& (table : valid : public, \{1\})\\
& (type : valid : public, tot)\\
& (state : valid : public, initialization)\\
\}
\end{aligned}
$$

All public slots are readable by the other user. Thus the information share runs agent/filter process to transmit the useful information to anthor agent. The consequence in the risk game is to inform the player by visual situated table lightening. The filter $f_{playerVisual}$ which activates distant IA, is written:

$$
\begin{aligned}
f_{playerVisual} \quad = \quad & [type.status(trace) = valid] \\
\wedge \quad & [type(trace_1) = (tot, player)] \\
\wedge \quad & [state.status(trace_1) = valid] \\
\wedge \quad & [state(trace_1) = initialization] \\
\wedge \quad & [table.status(trace_1) = valid] \\
\wedge \quad & [table(trace_1) = ?n] \\
\wedge \quad & \exists([table(trace_2) = ?t] \\
\wedge \quad & [table.status(trace_2) = valid] \\
\wedge \quad & [?n \neq ?t]) \\
\Rightarrow \quad & inform(GA, ?t, trace_1)
\end{aligned}
$$

The figure 6 shows the flexible process between the trace evolution and the filter triggering.

5 CONCLUSION

This paper presents a original trace/normative-based approach for social-physical computing. The objective of the system is to enable the user learning and norm appropriation. Our system is centred on the trace expression where the trace evolution draws conjointly the norm evolution.

ACKNOWLEDGEMENTS

This work was supported by Agence Nationale de la Recherche (ANR) under grant IMAGIT (Environnement multi-acteurs multi-tables interactives à objets tangibles et virtuels).

REFERENCES

Boella, G. and Torre, L. (2006). Introduction to normative multiagent systems. In *Computational and Mathematical Organization Theory*, volume 12, pages 71–79.

Boissier, O., Balbo, F., and Badeig, F. (2011). Controlling multi-party interaction within normative multi-agent organisations. In *Coordination, Organization, Institutions and Norms in Agent Systems VI (LNAI 6541)*. Springer-Verlag.

Bourguin, G., Derycke, A., and Tarby, J. (2001). Beyond the interface: co-evolution inside interactive systems a proposal founded on activity theory. In *IHM-HCI 2001 conference, People and computer XV - Interactions without Frontiers*, pages 297–310. Springer Verlag.

Clauzel, D., Sehaba, K., and Prié, Y. (2011). Enhancing synchronous collaboration by using interactive visualisation of modelled traces. In *Simulation Modelling Practice and Theory*, volume 19, pages 84–97.

Cole, M. (1996). *Cultural Psychology : A once and future discipline*. Cambridge, MA.: Harvard University Press.

Engestrom, Y. (1987). Learning by expanding: An activity-theoretical approach to developmental research. Helsinki, Orienta-Konsultit.

Engestrom, Y. (1991). Developmental work research: reconstructing expertise through expansive learning. In *Human jobs and computer interfaces*, pages 265–290.

Gâteau, B., Boissier, O., Khadraoui, D., and Dubois, E. (2006). Controlling an interactive game with a multi-agent based normative organizational model. In *Coordination, Organization, Institutions and Norms in agent systems workshop (COIN) at the 17th European Conference on Artificial Intelligence (ECAI)*, Italy.

Greenberg, S. (2001). Context as a dynamic construct. *Human-Computer Interaction*, 16:257–268.

Hubner, J. F., Boissier, O., Kitio, R., and Ricci, A. (2010). Instrumenting multi-agent organisations with organisational artifacts and agents. *Journal of Autonomous Agents and Multi-Agent Systems*, 20(3).

Hubner, J. F., Sichman, J. S., and Boissier, O. (2002). A model for the structural, functional, and deontic specification of organizations in multiagent systems. In Bittencourt, G. and Ramalho, G. L., editors, *Proceedings of the 16th Brazilian Symposium on Artificial Intelligence (SBIA'02)*, volume 2507 of *LNAI*, pages 118–128, Berlin. Springer.

Kubicki, S., Lepreux, S., Lebrun, Y., Santos, P., Kolski, C., and Caelen, J. (2009). New human-computer interactions using tangible objects: Application on a digital tabletop with rfid technology. In *Human-Computer Interaction (LNCS 5612)*, pages 446–455. Springer.

Kuutti, K. (1995). Activity theory as a potential framework for human-computer interaction research. In *Context and Consciousness: Activity Theory and Human Computer Interaction*, pages 17–44. Cambridge: MIT Press.

Lepreux, S., Kubicki, S., Kolski, C., and Caelen, J. (2011). Distributed interactive surfaces using tangible and virtual objects. In *Workshop DUI'2011 Distributed User Interfaces, at CHI'2011*, pages 65–68, Canada.

Nardi, B. (1996). Activity theory and human-computer interaction. In *Context and consciousness: activity theory and human-computer interaction*, pages 69–103. Cambridge and London, MIT Press.

Okuyama, F., Bordini, R., and da Rocha Costa, A. (2008). A distributed normative infrastructure for situated multi-agent organisations. In *Sixth International Workshop on Declarative Agent Languages and Technologies (DALT) to held as part of AAMAS-2008*, Portugal.

Pape, J. and Graham, T. (2010). Coordination policies for tabletop gaming. In *Graphics Interface*, pages 24–25.