GESTALT: a framework for redesign of educational software

M. Puustinen*, M. Baker† & K. Lund‡

*University of Poitiers, CNRS, France
†IUFM Nord Pas-de-Calais, France
‡LEAPPLE Laboratory, CNRS, France
§University Descartes Paris, France

Abstract
Design of educational multimedia rarely starts from scratch, but rather by attempting to reuse existing software. Although redesign has been an issue in research on evaluation and on learning objects, how it should be carried out in a principled way has remained relatively unexplored. Furthermore, understanding how empirical research on information and communication technologies (ICT) should feed back into redesign remains difficult. The present paper addresses these problems from the viewpoint of carrying out pedagogical expert evaluations, in the absence of empirical studies of target learners, in order to generate recommendations for redesign. Firstly, redesign proposals should be based on a coherent reconstruction of pedagogical foundations of educational ICT (software, documentation). Secondly, redesign proposals should result from dialogue between stakeholders, such as future users, pedagogical experts, software designers, and deciders. To these ends, we propose a framework, called GESTALT (Goals, (E)SiTuations, Actions, Learners, Tools), as a ‘boundary object’ for dialogical redesign. Within an activity theory approach, GESTALT is based on analysis of available tools, the actions they support, the characteristics of learners who perform actions, and pedagogical goals that could be achieved in specific situations. An illustrative GESTALT analysis of educational software is provided, principally from the viewpoint of pedagogical experts. Finally, the strengths and limits of GESTALT are discussed.

Keywords
boundary objects, educational multimedia, educational redesign, expert evaluation, simulation, qualitative evaluation.

Introduction
There can be several reasons for evaluating educational multimedia. For example, a teacher might want to evaluate different multimedia software in order to decide which is most adapted for use in teaching a particular subject in a particular class. This might be done using a checklist of important criteria, relating to, for example, ease of use of the interface and relevance of the knowledge taught. A second reason for evaluation can be to give feedback on the effectiveness of educational software during the process of its development. This might be done by experimentation, for example comparing a class of students using the software, and another class learning (supposedly) the same content, but using a pen-and-paper medium.

A third reason concerns redesign of educational multimedia. Given the demands of flexibility and distant access for higher education (Littlejohn 2003), and the considerable investment required for producing digital learning resources, several initiatives have investigated their possible reuse on an interna-
tional scale. One goal would be to produce and disseminate (via Web-based repositories) modular ‘learning objects’ that could be reused in new contexts to produce new courses. However, as Wiley (2003) has pointed out, this approach requires existing resources to be pre-tailored (in terms of software and pedagogy) for being recomposed with others. He (p. 1) proposes the striking metaphor of using ‘mortar holding together blocks of a variety of shapes and sizes’, rather than providing learning objects (LO) with ‘knobs and notches so that they could fit together perfectly’. The problem therefore remains as to how to reuse and/or redesign educational software that was not initially designed with the current aim in view.

In this paper, we describe a framework for educational multimedia redesign that was developed in the European Research and Development project VIRTU@LIS. The project brought together specialists in information technology, sustainable development, environmental modelling, public policy and governance, learning psychology, and open learning, to develop computer-based learning tools on ecosystems and natural resources. Among these learning tools, personal barometers, allowing quantification of environmental impacts of individual lifestyles, were developed, for the domains of soil pollution, emission of greenhouse gases, the depletion or sustaining of fisheries, and preservation of water resources.

As is the case with many such R&D projects, the software to be developed was to be based on previous software, whether already developed within the consortium or available elsewhere. As this software had been largely produced by content-experts (e.g. simulations of fish stocks in the North Sea, developed by biologists), the role of the present authors was perceived to be that of pedagogical experts who would provide recommendations for redesign. We would claim that most existing evaluation methods have not really been developed for addressing this problem. In attempting to solve it, we developed a framework for educational multimedia design, as a means for carrying out what we term a pedagogical (re)design critique. By a framework, we mean a set of theoretical principles, together with a ‘boundary object’ (Star 1989) for guiding critical analysis, rather than an explicit set of ordered procedures.

In what follows, we begin by discussing the redesign problem in relation to existing evaluation methods and research on reusing software, in the specific context of the VIRTU@LIS project. Following a presentation of the framework itself, the paper centres on a case study in which the framework is applied to redesign of software.

Evaluation and redesign in relation to the VIRTU@LIS project

In order to begin the process of making pedagogical recommendations to multimedia software within the VIRTU@LIS project, we reviewed existing research on educational software that was closely related to future VIRTU@LIS tools (principally, educational simulations) together with existing methods for educational software evaluation. Below, we briefly discuss the possible contributions to our redesign problem of each of these two areas of research, beginning with a short presentation of the GAS software that is analysed in a later section of the paper.

The GAS ‘personal barometer’

GAS (version 1.0; Guimaraes Pereira 2001) is a personal barometer on climate change. Its goal is to help European citizens to make connections between their individual lifestyles, modes of consumption, and their contribution to the emission of three greenhouse gases: carbon dioxide (CO₂), methane (CH₄), and nitrous oxide (N₂O). GAS is composed of three modules: a GAS Meter, a GAS Explorer, and a GAS Day Planner. In the GAS Meter, users introduce their personal data on energy consumption, and observing the effects of their own energy choices on greenhouse gas emissions, they are assumed to make the connection between energy use, pollutant emissions, and their own habits. The GAS Explorer, on the other hand, consists of a scenario generator with which users may explore ways towards sustainability by playing with alternative lifestyles or alternative regional settings. The GAS Day Planner, finally, allows users to calculate their daily greenhouse gas emissions. Figure 1 shows two screens of the GAS Meter, on the left, for data entry and on the right, a screen for showing output, calculated on the basis of data entry.
GAS is in fact built on a scientific simulation, incorporating up-to-date information on gas emissions in different countries. Its specificity lies in the fact that the input data are to be derived from the users’ own lifestyles (rather than from an experimental situation, for example). Therefore, we first looked to the literature on educational simulations, as a means of inspiring redesign. Our aim was to uncover sets of relations between characteristics of simulations, users’ prior knowledge, and specific domains, that would enable us to propose modifications of GAS that would lead to different types of learning for different user populations.

**From empirical research results to recommendations for redesign**

Educational simulations are usually built on (more or less ‘transparent’) models of a real or natural system (e.g. flight deck or laws of physics). Learning with educational simulations is commonly associated with discovery learning, with active involvement in the construction of meaning by hypothesis generation and testing.

Although simulation-based discovery learning has been claimed to be motivating, thus leading to deeper, better-anchored, and more intuitive learning than traditional lecturing (e.g. de Jong 1991; Kanselaar et al. 2000), studies having compared simulation-based discovery learning with more traditional kinds of instruction (i.e. classroom, computer tutorial) show no clear results in favour of simulations. Such inconclusive results can at least partly be explained by the difficulties that learners encounter during the different phases or processes related to discovery type of learning: hypothesis generation, design and running of experiments, interpreting graphical output, self-regulation, etc. (de Jong & van Joolingen 1998).

Given these difficulties, it is evident that some kind of instructional support is required (e.g. de Jong 1991), including goal decomposition tools (Singley 1990) or graphing tools supporting prediction making (de Jong & van Joolingen 1998). Some results have shown that model progression may be beneficial to learning if the target model is sufficiently complex (Kanselaar et al. 2000). Finally, the level of prior knowledge is an important factor to consider when providing learners with support in discovery learning situations (e.g. de Jong & van Joolingen 1998). Therefore, an important supportive element consists of allowing access to extra information, in the form of hypertext or hypermedia systems for example.

How should these research findings be applied to redesign (of GAS)? From the body of research (very) briefly summarised above, here are three examples of the recommendations that were developed, relating specifically to educational simulations:

- **Prior knowledge**: Consider learners’ everyday conceptions of the environment in design of interface terminology and explanatory texts (e.g. Andersson & Wallin 2000).
- **Transparency**: Augment the degree of transparency of the simulation by exposing some aspects of the
simulated domain to the learners via instructional measures (e.g. explanations; e.g. Leutner 1993).

- Critical reflection: Incite learners to reflect on the learning experience, to look for relationships between phenomena (e.g. Kolb 1984).

However, regardless of the relevance of these specific recommendations, it remains difficult to move from disparate experimental research results to design recommendations. Can we assume that recommendations are general, definitive, and certain, given the considerable human and material resources that must be invested in any new implementation? The experimental results on which our recommendations were drawn on had been derived from a limited yet relatively extensive set of situations, domains, and software implementations, involving manipulation of limited sets of variables. However, the extent to which such results can be extrapolated to new real-life learning situations has not yet been determined. For example, to what extent do results obtained in secondary school biology classes, with a mechanics simulation, transfer to everyday citizens’ understanding the emission of greenhouse effect gases?

Another issue concerned the necessity for understanding underlying scientific models. Although this may be an explicit learning objective for secondary school children, is such understanding required of every European citizen? Will such users not be simply content with observing relations between their everyday actions and greenhouse effect emission? This issue highlights the necessity of clearly defining target populations and attendant pedagogical goals in (re)design.

A final problem relates to the variety of theoretical frameworks within which experimental results are obtained, including experiential learning (e.g. Kolb 1984), discovery learning (e.g. de Jong & van Joostinga 1998), and self-regulated learning (e.g. Puustinen & Pulkkinen 2001; Zimmerman & Schunk 2001), together with more general theoretical frameworks such as constructivism. For redesign, it would be preferable to relate results and software within a unified theoretical approach. However relevant a set of recommendations may be, they do not in themselves constitute a framework for redesign.

**From evaluation to redesign**

Clearly, many of the existing educational software evaluation methods were not originally designed to help solve the redesign problem. It is, however, a common belief that evaluation results should somehow feed back into new designs, but what is feeding back? Given that our redesign problem is posed before user trials were to take place (it was decided to expend resources on user trials for the newly redesigned version of the software), we concentrate here on formative evaluation, including use of checklists, and expert evaluation.

For Maslowski and Visscher (1999), formative evaluation comprises four basic dimensions – problem coverage, scientific basis, usability, and effectiveness – each of which is associated with specific stages of design of educational computer software. In this process, designers are efficient as evaluators in the prototype stage because they can give quick feedback, but external evaluators are preferred in later stages for reasons of objectivity. One specific method for formative evaluation involves the use of checklists, or criteria, for diverse aspects of software, particularly interface ergonomics (Bastien & Scapin 1993; Scapin & Bastien 1997; see Ewing 2004, for more pedagogically oriented questions). Criteria are supposed to be independent from the experts who may have produced them – or indeed the software itself – and can be described as a set of rules for guiding design choices. One of the criticisms of checklists is the long duration of the analysis because of the large number of questions that must be answered. In response to this issue, Hüh (2001) has proposed a method that dynamically constructs an evaluation in terms of the depth, the choice, and the importance of the questions posed, thereby shortening and focusing the evaluation appropriately. Perhaps more importantly, some checklists can appear to be largely ad hoc, as they are not always based on a single coherent underlying theory or model of instructional design. To this extent, the set of replies to a checklist might provide little information concerning redesign.

Expert review is probably the evaluation approach that comes closest to addressing the redesign problem as it was posed in the VIRTU@LIS project. Such reviews can be carried out by experts in content-matter, pedagogy, ergonomics, or computer technology. The main aim is to make specific recommendations for
changes to the software. If we assume that educational software is designed for domain novices, the potential problem with this method is that experts do not always attempt to or indeed succeed in placing themselves in the position of a novice (Maslowski & Visscher 1999). In fact, they are often no longer able to ‘unpack’ their knowledge in order to suggest changes that take learners’ points of view into account.

As will be described below, our framework is in fact a general tool for carrying out expert reviews. In its early stages, it does not meet the problem levelled at other expert reviews, concerning the learners’ viewpoint. We propose that this is to be taken into account by asking learners themselves to use the framework and comparing their viewpoint with that of the pedagogical experts.

More generally, the issues of redesign and reuse of pedagogical software have been extensively discussed within the LO research community (e.g. Wiley 2000; Littlejohn & Buckingham Shum 2003). LOs can be defined as ‘any digital resource that can support learning’ (Wiley 2000, p. 7). Ideally, LOs are initially designed so that they can be flexibly recomposed into new courses, thus facilitating reuse and obviating the need for redesign and re-implementation. One important issue is the need to specify meta-data in LOs to facilitate decisions concerning their reuse, but this is quite far from the current situation. Although LOs can often reduce to ‘chunks of teaching content’ – with an attendant and limited knowledge transmission pedagogical approach – they can also include software (such as simulations), built on the basis of specific pedagogical approaches that make them difficult to recombine in other contexts.

Clearly, the GAS software was not designed for redesign; its content is fixed in an underlying scientific model, which is updated with values for gas emission in different countries, and that is also reified in specific screens. Depending on the target user population, it would be necessary to embed use of the software in a larger scale course to deal with prior knowledge and to draw out conclusions for actions to be taken, following exploration of the simulation.

**Synthesis**

Despite the existence of a large research literature on educational simulations, synthesising these results into a coherent whole, and applying them to concrete redesign remains a difficult problem. Expert evaluation techniques can be used to provide input to redesign, and yet have difficulties in envisioning future users.

Williams (2000, p. 4) has pointed out that underlying all current thinking on evaluation is the idea of ‘comparing what something is to what it ought to be, in order to facilitate judgement about the value of that thing’. From our point of view, what educational software is and what it ought to be are questions to which no simple answers are forthcoming. Determining what the software is requires a rational reconstruction of pedagogical (and ergonomic) foundations, whether these correspond to designers’ original intentions or not (to which access is not always possible). Determining what it ought to be requires clarification, together with empirical work (e.g. to determine target users’ characteristics). If we pose the redesign problem as one of identifying what contribution pedagogical experts can make prior to experimental user studies, then one possible contribution is to provide a framework that involves reconstructing designers’ possible intentions, and examining their internal coherence.

We therefore propose a change of stance, within which redesign recommendations are no longer seen as prescriptions but rather as objects for collective reflexion. The framework presented in the next section is intended to provide such a setting for unifying results, recommendations, and reflexions. Given that redesign is a collaborative enterprise, it requires mutual understanding. Our framework is thus deliberately simple enough to be understandable by researchers and developers who are not also pedagogical experts.

**GESTALT: a framework for redesign**

GESTALT (Goals, (E)SiTuations, Actions, Learners, Tools) is a framework for redesign of educational ICT. The name of the framework is a clin d’œil to Gestalt psychology, founded by Wertheimer, Köhler, and Koffka, who attempted to explain human perception by showing how the mind can perceive organised wholes by understanding relations between otherwise unconnected physical stimuli. In fact, we propose that pedagogical (re)design critique requires re-creating a priori precisely such an organised
whole, for educational software, embedded in specific situations.

The framework proposes that any educational software should be designed on the basis of a coherent set of relations between the tools that learners have at their disposal (e.g. text, graph, multiple choices), the actions that can be carried out with the tools (e.g. reading, data entry), the learners’ characteristics (e.g. prior knowledge), and the pedagogical goals that can be achieved as a result (e.g. understanding the processes underlying the greenhouse effect, becoming a more responsible citizen), in given situations (e.g. at home, at school, at work; see Fig 2). Clearly, the importance of considering how learning environments support specific activities, and what could be learned from doing so, has already been addressed in the educational design literature (e.g. Bell 1998). The specificity of GESTALT resides in the fact that it provides a unifying framework (or boundary object) for such considerations in the context of redesign.

**Foundations and elements**

GESTALT is inspired by activity theory (e.g. Leontjev 1981; Engeström & Cole 1997), both in the way its elements are theorised, and in the way they are seen to be interrelated. In this context, Barab and Plucker (2002) have contrasted analysis methods that consider interacting system components in isolation, with activity theory approaches that provide a means for accounting for the dynamics of teaching–learning situations. If activity is seen as a contextualised set of events, distributed across tools and contexts, all of which are embedded in a sociocultural history, then activity theory provides an important reflective tool for exploring the situated nature of cognition.

As is well known, Vygotsky’s (1934/1986) notion of tool-mediated action is one of the principal foundations of activity theory. Accordingly, tools are seen as both tangible (e.g. graphical interfaces) and cognitive–semiotic (the multiple types of semiotic representations shown on screens, and their significations, or concepts, in Saussure’s 1915/1972, terms). Actions do not exist independent of the tools; they are constituted by them. Furthermore, in terms of Leontjev’s (1981) three-level model of activity, actions are said to be ‘conscious’ to the extent that they are oriented to individual or collective goals. The specific goals to which actions are oriented can be seen – at least initially – as quite low-level. For example, if a set of actions with GAS involves inputting personal information concerning home heating appliances, food consumption, etc., then the associated low-level pedagogical goal might be simply that the user becomes aware of, and begins to reflect on, that information.

**Learners** are considered principally in terms of their prior knowledge and skills relating to the tool-medi-
ated actions that can be carried out with the software, in relation to pedagogical goals. The necessity to consider learners’ characteristics and their communities in the design of online learning environments has been well described in the literature (e.g. Schwen & Hara 2003).

The definition of the final element of GESTALT, situations, is somewhat complex. As Engeström and Cole (1997) point out, the notion of situation — underlying situated cognition theory — can be seen as a moment in time, a place, a life situation, a social situation, or a configuration of relations. We see the notion of situation as closely related to the level of activity, in Leontiev’s (1981) model. While goals underlie actions, and sets of coordinated actions are the concrete manifestations of activities, the latter are more precisely defined in terms of societal motives, or needs\(^1\) that stand behind them. For example, the main need standing behind activities such as hunting and agriculture is clearly that of hunger. The GAS software, presented above, relates to the activity of educating citizens, with respect to prevention of atmospheric pollution. The underlying societal need is clearly that of preserving our air or more generally, preserving health.

While situations are on one level defined by physical aspects (such as equipment available), human characteristics (knowledge, personal needs), and organisational constraints (e.g. school rules), we claim that the societal activities that are in play are also crucial to their definition. They are crucial, as the activities-needs define the meaning of the situation for the participants. For example, a doctor’s actions in treating patients can be seen as parts of two different activity systems: economic activity (making money) and medical activity (healing people). Which — or both — of these activities are involved for participants crucially alters the meaning of the situation for them, and most probably the actions that will be carried out. To return to the example of GAS, its users would be in a different activity-situation according to whether they see it to be a means for helping themselves to save money, to save the environment, or to help a European project to be successful.

Finally, knowledge (or teaching content) is not explicitly represented in the GESTALT diagram. In terms of the theoretical approach outlined above, content is seen as distributed across cognitive-semiotic tools (such as [hyper]texts, diagrams) that are reified in interfaces, the goal-oriented actions that learners perform with the latter, and the learners’ characteristics themselves. Such a close coupling between knowledge, tools, and actions is not incoherent with Boyle’s (2002) quite different theoretical base for educational multi-media design, which proposes that context consists of ‘the framing of content along with associated interactivity’ (p. 6).

Relations between elements

In this section, we discuss relations between GESTALT elements, relations between different GESTALT viewpoints of educational software, and the question of the dynamic nature of such relations.

Internal relations

Internal relations are those that exist between GESTALT elements within a single viewpoint, such as that of pedagogical experts (for multiple viewpoints, see below). Four specific types of relations are coherence, compatibility, completeness, and relevance (e.g. Bastien et al. 1998\(^2\)), depicted in Fig 2.

Coherence concerns the extent to which given interface tools are used in identical contexts of use, and are different for different contexts. For example, an interface that displays results of user actions (such as entering data) in the same way (e.g. same type of graph) for identical types of actions is more coherent than one that uses different graphs. Compatibility, on the other hand, concerns the extent to which the interface tools, together with the activities that are to be carried out with them, and the contents involved, are adapted to a given class of users. Coherence and compatibility are determinants of the learnability of the software, in terms of interface actions (a component of usability; Nielsen 1993) and in terms of relations between content and prior knowledge.

\(^1\)Confusingly, for certain research traditions, Leontiev also terms the need or motive behind an activity its ‘object’ (towards which the activity is oriented).

\(^2\)These authors describe general ergonomic criteria for evaluating websites. The concepts they introduce are, however, also relevant to pedagogical software.
Completeness concerns the extent to which the software supports all of the actions necessary for achieving a specific goal, and also the extent to which the sum of specific goals that can be achieved constitutes a meaningful whole. For example, does the GAS software support understanding of all of the major aspects of everyday life that relate to greenhouse emission, or are important aspects missing? Relevance is the relation between the set of (low-level) learning goals that can be achieved and societal activities that determine situations. For example, does understanding the relation between personal life and gas emissions really respond to a group’s most important needs (or might not preventing industrial pollution be what is required)? In a more narrow sense, we can also ask whether the knowledge elaboration supported by the software – for example, learning a restricted branch of geometry – fits into a specific national curriculum.

External relations
The objectives of GESTALT, based on collective exploration of redesign, are similar to the activity models described by Cole and Engeström (1993), and more specifically, GESTALT can be seen as what Star (1989) has called a boundary object.

Star (1989) devised this concept on the basis of studies of the actual practice of scientific research. In answering how ‘within what may sound like near chaos, scientists manage to produce robust findings’ (p. 45), she said ‘they create objects that are both plastic and coherent through a collective course of action’ (p. 45), that the aggregation of different viewpoints around deliberately vague boundary objects is the source of the robustness of science. In terms of Star’s taxonomy of boundary objects, GESTALT can be seen as an ‘ideal type or platonic object’, to the extent that it is ‘abstracted from all domains, and may be fairly vague’ (p. 49).

What we term an external relation in GESTALT can thus be obtained between (groups of) its elements, as defined from the different viewpoints of the multiple social actors or stakeholders who are concerned by it: citizens, teachers, ecologists, deciders, software designers, researchers, and so on. For example, it is possible to examine the relations between the pedagogical goals that a priori could be achieved, minimally, by users carrying out the sets of software-supported actions (the pedagogical experts’ viewpoint) with the goals seen from the software designer’s viewpoint (either from interviews, or from statements present in user manuals).

GESTALT was conceived to be understandable and useful as a boundary object within a research-development project, involving experts with very different viewpoints. At the present stage of this research, it has not yet been fully used in this way. In what follows, we shall therefore restrict discussion to the pedagogical experts’ viewpoint (i.e. the present authors) and its relations to the partly reconstructed information and communication technologies (ICT) designers’ viewpoint.

Dynamicity of relations
To complete the presentation of GESTALT, we mention simplifying assumptions that have been made with respect to the dynamic nature of complex systems. We would claim that real-life learning situations, involving ICT or not, are complex systems, in the strict sense of the term (e.g. Pavard 2002), and that the emergence of redesign proposals, from multiple viewpoints, is also a system of this kind. Three main characteristics of complex systems are difficulties in identifying discrete variables, the existence of bi-directional influences between variables, and – consequently – unpredictability beyond a certain time-window.

If we consider the relations between tool-mediated (inter)actions and learners’ evolving knowledge, then they are precisely circular in this way. If interactions can be seen as producers of new knowledge, then evolving knowledge influences the nature of the interactions themselves (Baker 2004). If we consider cognitive and social dimensions of human interactions (Perret-Clermont et al. 1991; Grossen et al. 1997), then it is clear that they are either inseparable (all cognition is social) or else at least mutually influencing (a cognitive change influences the perceived relation, a change in the social relation influences the cognitions that will be elaborated).

The process of instrumentalisation of technological artefacts has been described by Rabardel (1995). An artefact becomes an instrument for a user when it is associated with action schemas. What this means is that the artefact seen from the designer’s point of view...
will not be the same as the instrument that the user creates.

The above considerations mean that if situations of use of ICT are complex systems, then (re)designing them cannot be based on the kinds of procedural approaches that assume predictability. What can be done – we propose – is to make explicit the foundations of design by reconstruction, and to reflect collectively on the types of relations described.

**Guidelines for using GESTALT in redesign**

Given the above, it would not be meaningful to prescribe deterministic procedures for carrying out redesign, either with GESTALT or with other approaches. We therefore limit ourselves to a description of the main steps we went through in applying GESTALT to GAS. Recall that GESTALT involves largely *a priori* expert analysis, which can draw only on the additional information available at the time of that analysis (user or publicity documentation, published papers on user trials of the system, etc.). The process is one of rational reconstruction, on the basis of incomplete knowledge.

The first step is to analyse the *tools*, grouping together interfaces based on functionality. For example, interfaces requiring numerical data entry (whether personal data or scientific data) can be grouped together. There are practical constraints to carrying out such analyses. If it is thought that sufficient feedback for redesign can be obtained from zooming in to certain features, it would not be necessary to analyse an ICT programme exhaustively.

The second step requires extrapolating from tools to sequences of associated human *actions*. Here, a fruitful approach seems to be successively generalising from low-level interface actions, to cognitive, linguistic, or physical actions that may in turn be grouped into meaningful wholes. For example, a tool for numerical personal data entry can require the following low-level actions: read and follow screen instructions, obtain required personal values, input personal values, check values sum to 100. Each of these actions can be associated with others, such as ‘try to remember what kind of car you use’, or even communicative and physical actions such as ‘phone family member and ask for value’. The set of such actions could be grouped together as ‘determine and input personal values’.

The third step is to examine the degree of tools ↔ actions *coherence*. Do the tools implement sets of actions in comparable ways? Carrying out these first two steps, together with examining coherence, to some extent resembles performing a ‘cognitive walkthrough’ (Wharton *et al.* 1994), in which the evaluators go through each step in tasks supported by the software, and provide stories about why those steps are or are not good for new users.

While the above steps are focused on tangible artefacts and the actions they can support, the subsequent steps require a greater degree of extrapolation and information search. In extrapolating from tools-actions to *goals*, we adopted an initial minimalistic approach. For example, the actions ‘determine and input personal values’ can enable the user to become aware of what those values are. However, by grouping together such goals and progressively generalising them in a bottom-up way, it is possible to arrive at broader pedagogical goals – such as ‘reflecting on one’s lifestyle, and how it could be changed in order to preserve the environment’ – and even general putatively associated pedagogical theories (e.g. experiential learning). We propose that such a minimalistic approach provides a sobering basis for comparison with some of the more extensive claims that ICT designers and manufacturers can make. Where learning goals of the software are stated by designers, in accompanying documentation, these can be compared with the pedagogical experts’ view, by a process of examining *external coherence* across viewpoints. Finally, in this context, the *completeness* relation between mediated actions and goals can be reflected upon: would other actions need to be supported in order that the set of goals become a coherent whole?

The analysis of target *learners*, together with the actions ↔ learners *compatibility* relation, requires a similar degree of extrapolation. What type of prior knowledge and skills (with computers, in reading, in arithmetic, etc.) would be required in order for those actions (e.g. reading technical terms) to be carried out with understanding? What assumptions are made about users’ everyday lives (e.g. asking about a user’s car assumes this is relevant). Where suitable documentation is available, *external coherence* with respect to learners’ characteristics can be examined.
across the pedagogical experts’ reconstructed viewpoint, and designers’ statements.

Finally, it is possible to inquire as to the relevance of the pedagogical goals (purportedly) supported by the software in relation to the educational situation. Clearly, this requires empirical research on the requirements of situations (e.g. how aware are certain classes of citizens concerning environmental problems?). In the absence of this, existing studies can be consulted, together with official documents such as national curricula.

Table 1 General and Specific Tools Used in GAS.

<table>
<thead>
<tr>
<th>Specific tools</th>
<th>General types of tools</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Navigation</td>
</tr>
<tr>
<td>Text</td>
<td>x</td>
</tr>
<tr>
<td>Graph</td>
<td></td>
</tr>
<tr>
<td>Numerical value</td>
<td>x</td>
</tr>
<tr>
<td>Multiple choice</td>
<td>x</td>
</tr>
<tr>
<td>Qualitative information</td>
<td>x</td>
</tr>
<tr>
<td>Menu choice</td>
<td>x</td>
</tr>
<tr>
<td>Parametrisation</td>
<td>x</td>
</tr>
</tbody>
</table>

An example: GAS analysed with GESTALT

We started our analysis by going through the software screen by screen and determining the main type(s) of tools for each screen. Four general types of tools were found: navigation tools, information presentation tools (e.g. help), data entry tools, and result presentation tools. Furthermore, we carried out a more fine-grained analysis of the tools across specific interfaces, into seven categories: text, graphs, numerical values, multiple choices, qualitative information (e.g. choice of the appropriate value on a scale low–medium–high), menu choice (e.g. choice of an item from the menu bar), and parametrisation (e.g. choice of font size; see Table 1).

Next, we determined actions associated with the tools presented above. We distinguished them into low- versus high-level (groups of) actions, the former referring to concrete interface operations (i.e. click on the icon; read a text, a graph, or numerical values; and enter data) and the latter to the corresponding cognitive actions. Three main categories of such cognitive actions were found, each of them corresponding, globally, to one general type of tool: discovering information (e.g. discovering the aim of a GAS module, discovering information concerning a particular country or the whole world, discovering website addresses with supplementary information), corresponding mainly to information presentation tools; recalling personal (e.g. everyday) knowledge, corresponding essentially to data entry tools; and comparing results or examining the effects of choices, corresponding mainly to result presentation tools.

An evaluation of tools ← actions coherence revealed some incoherencies. First of all, even if a GAS screen corresponded, most of the time, to one main type of tool, some screens corresponded to different types of tools (e.g. data entry and information presentation; see Fig 1, left-hand side, for example). In addition, as shown in Table 1, three types of data entry tools were used in GAS. However, the number and the combination of tools were different for most data entry screens; for example, some data entry screens included up to four different data entry tools, such as two different types of numerical values (i.e. absolute values and proportions), qualitative data, and multiple choices. Opting for more uniformity – or coherence – would certainly help the user. Furthermore, even if the

---

3 A tool is characterised both by its function (i.e. the concrete interface operations or the cognitive actions that could be realised with the tool, for example, recall personal knowledge, enter personal data) and its semiotic register (e.g. text, graph).

4 The term operation refers to the lowest level of activity in Leontjev’s (1981) model, these being routine-like rather than conscious and oriented towards goals, as with actions.
data entry tools used in the GAS Meter and the GAS Explorer were the same (e.g. multiple choices, numerical values), the results were presented using different types of graphs. In the GAS Meter, only bar graphs were used, whereas in GAS Explorer, both bar graphs and radar graphs were used. In addition, related textual and graphical results were presented simultaneously in the different GAS output windows, but the links between the different representations (e.g. bar chart, radar graph, and text in GAS Explorer) were not made explicit. However, research on multi-representational learning environments (e.g. van Someren et al. 1999) has shown that coordinating information from multiple representations is a difficult process and should thus be supported. Finally, the scales used for the three greenhouse gases in the output windows were different (see Fig 1, right-hand side, for example). It would be preferable to use identical scales for the different gases in order to avoid misinterpretations; otherwise, users’ attention should be called to this lack of uniformity.

From the viewpoint of our pedagogical expert analysis, the main pedagogical goals that could be achieved as a result of carrying out the activities listed above and using the above-mentioned tools include becoming aware of one’s own gas emissions, understanding the connections between input (i.e. own behaviour) and output (i.e. gas emissions), reflecting on the effect that changes in personal behaviour may have on gas emissions, feeling concerned about reducing gas emissions in one’s own personal life, and understanding that while some factors are highly relevant, other factors are much less so and that some choices are thus better than others. The broader pedagogical goals associated with such low-level goals could be ‘educating citizens to reflect on how they could change their lives to preserve the environment’, within a general discovery learning or experiential learning approach.

As far as the potential users of GAS (i.e. learners) are concerned, our analysis (extrapolating from mediated, goal-oriented actions) led us to conclude that a minimum of scientific knowledge (concerning the different gases, combustion, etc.), corresponding approximately to secondary school level, as well as a minimum of specialised vocabulary (e.g. sustainability, governance) are obviously required.

Evaluating actions → learners compatibility requires that the learner population be defined: who are the learners, and what learning situations are they in? In the user manual for GAS (Guimaraes Pereira 2001, p. 7), it is stated that it is designed for ‘ordinary citizens, NGOs and other stakeholders who wish to investigate their contributions to a global issue and explore alternative pathways to reduce their burden’. From our perspective, there were two problems here. On the one hand, a category of target users such as ‘ordinary citizens’ seemed too broad to be credible, and on the other, it seemed plausible that stakeholders and persons working in NGOs would have specific professional constraints and needs. If educational software that is suitable for anyone in any circumstances and for any task could really exist, then the question of compatibility is no longer of any importance.

Finally, the complex and open knowledge of the environment on which GAS is based seems to constitute an excellent potential topic for collaborative learning situations (e.g. constructive debate). However, our analysis showed that in its present form, GAS is exclusively designed for individual, as opposed to collaborative, learning situations. In fact, no tools for supporting knowledge sharing, constructive argumentation, or coordination, for example, are concretely provided to the users (e.g. make a button that has to be clicked with ‘ok’ by all collaborators, in order to execute a task). This is important in terms of an analysis of societal activities that constitute situations, in relation to relevance. The need to preserve the environment can only be satisfied by dialogue and collaboration between social actors (citizens, stakeholders).

In sum, our analysis of GAS with GESTALT led us to make proposals for design relating principally to interface coherence and compatibility for users in specific situations. These proposals were considered by other members of the VIRTU@LIS project 5 given that they were perceived to be motivated by a coherent approach.

Conclusions

The work described here is largely programmatic, to the extent that limited empirical support has been described for the usefulness of GESTALT in educa-

5Our proposals were not necessarily acted upon directly by members of the project responsible for re-implementation, for various practical and resource-related reasons. However, given that they were proposed as objects for discussion, we did not expect such direct effects.
tional software redesign. Given that GESTALT is intended as an open-ended boundary object, it synthesises much of what is ‘common knowledge’ in educational psychology and ICT design, while at the same time unifying it into a single framework, for whose application certain guidelines can be provided. Such a relatively simple synthesis (in terms of its basic elements, but not in the underlying theories of learning) was proposed so that it could provide a mutually comprehensible means for coordinating multiple theories, models, and results, together with multiple viewpoints of stakeholders concerned by ICT redesign. Nevertheless, here, we could only describe analysis according to a single viewpoint – that of pedagogical experts, with some consideration of designers’ views, as manifested in supporting documentation. However, our description of the processes involved in a specific analysis, together with the redesign proposals to which it gave rise, can provide some guide for future development of GESTALT in redesign practice. In part, the present limited application of GESTALT is due to the fact that it was largely developed during the course of the VIRTU@LIS project, rather than before it. We therefore plan to use and develop GESTALT in subsequent projects of this kind.

In conclusion, we would like to re-emphasise that GESTALT is designed as a tool for mediating collective and practically oriented reflexion on the redesign of educational multimedia, and as in the case with any tool, we expect it to evolve as a result of its uses, an example of which has been described here.

Acknowledgements

We would like to thank Prof. Martin O’Connor, for having given us the opportunity to carry out this research, within the VIRTU@LIS project, and Jim Ewing, of Dundee University, for inspiring discussions on pedagogical issues, some of which contributed to the genesis of GESTALT.

This paper was prepared as part of VIRTU@LIS (Social learning on enVIRonmental issues with inTer-active information and commUnicAtion technoLo-gieS), a European Union-funded research programme (no. IST-2000-28121; http://neptune.c3ed.uvsq.fr/vir-tualist) and was presented at the 7th Biennial Conference of the International Society for Ecological Economics (ISEE), Sousse (Tunisia), 6–9 March 2002.

References


