A Laboratory of Ergonomic Analyses for Children Suffering from Cerebral Palsy

Monique Noirhomme-Fraiture [1], Clairette Charrière [2], and Jean Vanderdonckt [3]

[1] Institut d'Informatique, Facultés Universitaires Notre Dame de la Paix,
rue Grandgagnage, 21 – B-5000 Namur, Belgium
Phone : +32-(0)81/72.49.79 – Fax : +32-(0)81/72.49.79
monique.noirhomme@info.fundp.ac.be
[2] Service specialisé en infirmité cérébrale et reeducation fonctionnelle neurologique,
Département de Pédiatrie, Hôpital de Bicêtre
rue du Général Leclerc, 78 – F-94275 Kremlin-Bicêtre, France
[3] Université catholique de Louvain (UCL) - Institut d'Administration et de Gestion (IAG)
Place des Doyens, 1 – B-1348 Louvain-la-Neuve, Belgium
vanderdonckt@qant.ucl.ac.be

Abstract. Several computing platforms and software today enable the user to specify some adaptation decisions for the interaction techniques, styles, devices and presentation styles to accommodate the broadest possible range of users having varying behaviors, including users with special needs. However, these adaptation decisions are mostly achieved by hand on a trial and error basis. Moreover, these adaptation decisions are dynamically changing, depending on the type of user and other parameters. The goal of the ERGOLAB software described in this paper is to carry out ergonomic analyses for children suffering from cerebral palsy. These analyses are intended to provide results that would help to take these adaptation decisions in a more rigorous and systematic way. For this purpose, three interrelated aspects, namely perception, motor functions and intelligence, are involved to identify, to evaluate the factors that would enable a particular child to adapt to software. ERGOLAB consists of three stages divided into progressive levels where different exercises are proposed either in initiation mode or in test mode. Two experiments are described.

1 Introduction

Nearly all existing computing platforms and software today provide several facilities to tailor the user interface of the environment to the user's needs. These facilities often consist in adapting, refining, and customizing parameters governing interaction techniques (e.g., drag and drop, point and click, data entry), interaction styles (e.g., direct manipulation, questions/answers, form filling), interaction devices (e.g., mouse, track ball, touch pad), and presentation styles (e.g., screen colors, icon size, font size). The environment is expected therefore to accommodate the broadest range of human abilities, requirements, preferences, cognitive styles, habits and conventions. Universal design [13] entails the consideration of all potential users' abilities, skills, requirements, and preferences, and seeks to proactively accommodate the resulting diversity in the user population. It thus includes users with disabilities as an important and equitable portion of the user population.

J. Vanderdonckt et al. (eds.), *Tools for Working with Guidelines*
© Springer-Verlag London 2001

1.1 Goals and Aims of Adaptation

Adaptation decisions generally occur in the selection and the tailoring of alternative modalities, depending on the type of user and/or the type of disabilities (e.g., auditive feedback for visually impaired people, alternative interaction techniques for people with motor impairment, speech generator for blind persons). Since extensive research already exists in the selection and use of these modalities ([12,15] are among others), our purpose would rather focus on how to adapt these selected modalities to a fine-grained tune, depending on precise human abilities or disabilities.

First, these adaptation decisions are up to now mostly achieved by hand with the facilities provided by computing platforms and software. For instance, in the Windows'98 environment, accessibility facilities enable the user to:

1. Adjust the mouse pointer speed by specifying how fast or how slow it should be (Fig. 1a). This adaptation decision is entirely achieved **manually** by assuming that the user knows the required pointer speed and the corresponding value of the slider.
2. Tailor the double-click speed by specifying the time interval between the two clicks (Fig. 1b). This adaptation decision is entirely achieved **automatically** by the environment by allowing the user double-clicking on a set object and by adapting the speed accordingly.

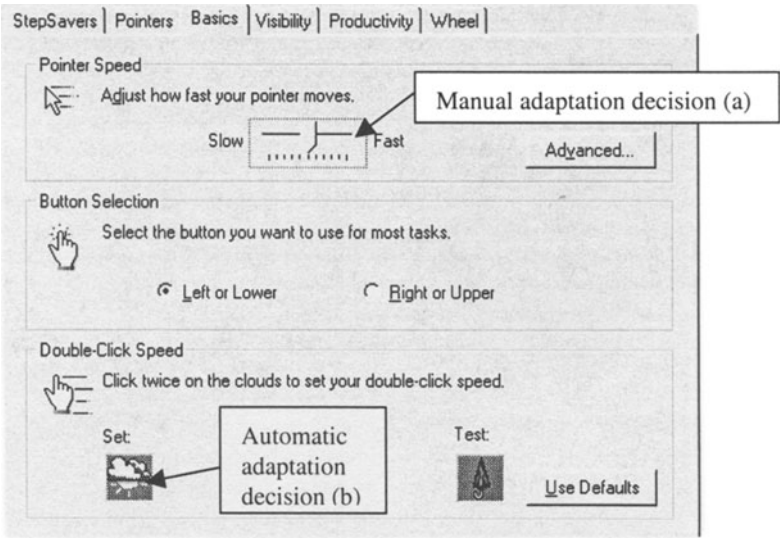


Fig. 1. Controlling the mouse pointer speed (a) and double-click speed (b) in Windows'98.

Second, these adaptation decisions are basically tuned on a trial and error basis so that the final adaptation decision is assumed to be the one wanted by the final user. On one hand, these adaptation decisions are sometimes hard to specify for people with disabilities. On the other hand, some people with disabilities are using devices whose adaptation decisions are not supported by the computing platform, but by sepa-

rate software, which is not so usable as the computing platform facilities are.

Third, these adaptation decisions are changing dynamically with time, user experience and usage of alternative modalities, devices so that the adaptation decisions do not stay valid for a long time. The standard deviation of these changes with respect to the mean is often large comparatively with the standard deviation of the parameters for an able-bodied user.

Fourth, the default values proposed by computing platform facilities are generally intended to accommodate the abilities of an able-bodied user, although they claim to support other users as well, but without any guidance for this purpose. At least for these reasons, investigating how these adaptation decisions can be taken to a fine-grained tune in a more rigorous and systematic way seems important to us.

1.2 State of the art

Several works have already been conducted in the domain of adaptation decisions:

McMillan *et al.* introduced a rule-based advice-giving system about how to configure a computing system for users who have special needs [7]. It employs a simple user model describing visual, cognitive, motor, and other abilities. Recommendations are made about appropriate input and output devices, including screens, keyboards, speech devices, and many others. The program was tested against professionals in this field and was shown to agree with them about as well as they agree with one another. Potential uses include advising those who configure computer systems.

Stephanidis *et al.* developed the USE-IT environment where these adaptation decisions are automatically derived by the tool itself [1,12]. These decisions are taken in a three-steps approach: (1) a reasoning about the user interface components which are subject to adaptation, (2) a selection of possible adaptations for each component, (3) a final decision on adaptations maximizing the user preferences. One important advantage of this environment relies on its ability to apply adaptation decisions on Abstract Interaction Objects (AIOs), which form an abstraction of identified user interface component [15], rather than directly on the concrete components themselves.

Keates *et al.* conducted experiments to identify the differences between adaptation decisions for able-bodied users and for motion-impaired persons [6]. From these results, they developed a methodology for the design of adapted user interfaces.

1.3 Our current work

For several years we have been developing software designed to assist persons with disabilities [3,4,8,9]. The software described in this paper, ERGOLAB, is the result of a joint effort involving the Computer Science Institute and the Psychology Department at "Facultés Universitaires Notre-Dame de la Paix" and the Cerebral Palsy Department of the Kremlin-Bicêtre Hospital in Paris [8]. This software may either serve to help a mentally handicapped person in performing daily tasks (e.g. keeping accounts, planning meals, making purchases) or take the form of didactic or even therapeutic games (e.g., knowing about one's body, puppet-show game). The last mentioned is intended for children suffering from cerebral palsy whereas the other programs have been designed for adults. The methodology employed to design the software consists

in:

- Defining the users and the purpose of the software.
- Setting up as exact as possible a schedule of conditions with instructors.
- Producing screens.
- Trying out the first version of the screens on a few users.
- Making correction to the screens and linking them together.
- Evaluating the end product, making particular use of questionnaires.

As the population concerned is very varied and as what may suit one user does not necessarily suit another, we solved the problem by introducing software parameters (e.g., sound speech rhythm, mouse speed, mode of data introduction, precision of target to reach). Nevertheless, there still remained the problem of regulating the parameters.

Moreover, despite the help provided by the therapists who know the users well, it is no easy task to work out these individual capabilities when put in front of a computer: what are they able to do? Can they operate a mouse? Can they understand how scrolling or a pull-down menu works? Etc. This gave rise to the idea of creating software to diagnose a person's capabilities in respect of a computer with a view to facilitating the creation of other programs and their parameters. This software will also provide figures to adapt the currently existing facilities to tailor the modalities of a particular computing platform or other piece of software.

The idea took shape when the Cerebral Palsy Department at the Kremlin-Bicêtre Hospital experienced several troubles using the different modalities provided by these computing platforms and their adaptation facilities. This department makes considerable use of computers in teaching handicapped children but comes up against a serious obstacle when trying to decide on the appropriate interface, the appropriate modalities, interaction devices and their parameters.

ERGOLAB, a software for carrying out ergonomic analyses and intended for children suffering from cerebral palsy, has been developed to address these issues. It helps, for a particular child with multiple deficiencies, to determine the objective, technological features that need to be introduced and adapted. In this way, it is expected adaptation decisions be taken in a more rigorous and systematic way than simply by hand. It is aimed at helping therapists to decide these adaptations according to experimental results provided by the ERGOLAB test bed.

1.4 Cerebral palsy and computer as a technical aid

The term cerebral palsy is employed here to cover the consequences of a cerebral lesion occurring prior to birth. The infant with cerebral palsy retains the after-effects of the inter-related deficiencies; the extent to which he will in fact be able to compensate for them will vary. The child with cerebral palsy learns to act differently, adopting the most effective strategies put to him, and perhaps off-setting particular deficiencies by means of individualized technical aids. Among these technical aids, the computer is first and foremost employed as a communication tool. It is highly valued in an educational context where it makes up for lack of knowledge of know-how, especially writing.

2 ERGOLAB: A Laboratory of Ergonomic Analyses

ERGOLAB allows the child's ability to grasp and handle the information presented on the screen to be studied, as well as its behavior with respect to the computer tool. It must be seen as part of the evaluative methodology developed by Prof. Tardieu in the Cerebral Palsy Department at Kremlin-Bicêtre Hospital, which in turn forms part of the much larger project of *Therapeutic Education*. The latter focuses on the child and combines a particular education with analytic forms of therapy, which may be global or functional and are prescribed. The "factorial" evaluation underlying *Therapeutic Education* identifies the causes of the observed disorders, their relative responsibility for the disturbed function and the possible forms of treatment. The faculties required to perform the tasks proposed by ERGOLAB involve three aspects that are closely interrelated, namely **perception**, **motor functions** and **intelligence**.

ERGOLAB is therefore most particularly useful to ergo therapists, whose work is to evaluate functional capabilities and handicap situations as well as to study ergonomic improvements and functional learning. This software consists in four features:

1. *Everything is reduced to essentials.* The software is intended to support evaluation of interaction techniques, not learning them. That is why any playing portion is intentionally excluded and the drawings have been made as simple as possible so as to allow the child to concentrate on the task in hand. Any kind of distraction is avoided. His might make certain screens seem slightly off-putting at first sight, but it is intended in order to avoid any interference in the evaluation. It is also with this end in view that the controls not directly related to the actual test are made accessible only to the instructor and do not appear on the child's screen.
2. *There is no progression.* The exercises are progressive in the sense that they are presented in three stages, each stage being divided into several levels and each level capable of comprising several exercises. The tests range from the simple click in response to a concrete or abstract stimulus to manipulating an object such as a pull-down menu having two dimensions or even a circular structure. The therapist has, however, the possibility of beginning where he wishes and of skipping certain exercises.
3. *The software is bimodal.* Each exercise exists in two modes: (1) an *initiation mode* when the instructor explains the test to the user and in the course of which the latter may receive training; (2) a *test mode* during which no explanation is provided and certain statistical data are recorded. A different color background is used to distinguish the two modes: gray for the training, white for the test.
4. *Data are recorded.* During each test, data are recorded for future usage. These data obviously depend on the test in question. The data relate to pressing down time, response time, number of errors, mean, standard deviation, number of clicks.

The rationale behind the exercises design is driven by two elements: the experience with children with cerebral palsy of the therapists of Kremlin Bycêtre hospital and the tasks to be performed with standard software. From the abilities necessary to perform tasks in standard software, therapists have derived simple exercises to test these abilities, separately and systematically [2].

3 Description of the Stages

This section summarizes the different exercises proposed. Exercises are structured in stages, which are further decomposed into several levels of difficulty [10]. For more details, we refer the reader to [2].

3.1 Stage 1: Pressing the Contactor

This stage measures the possibility of reacting to a stimulus by pressing down on a mouse contactor (usually, the main button). At level 1, the exercise tests the simple click, i.e. pressing down on the contactor as soon as an abstract (Fig. 2a)or a concrete visual stimulus appears on the screen (Fig. 2b).

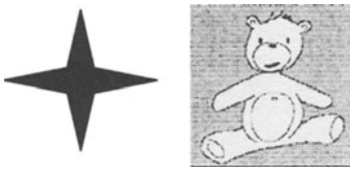


Fig. 2. Pressing the contactor in reaction to a visual abstract (a) or concrete (b) stimulus.

At level 2, the exercise consists in testing maintaining the pressure on the contactor, i.e. pressing down on the contactor until an abstract or a concrete object has reached its target (Fig. 3).

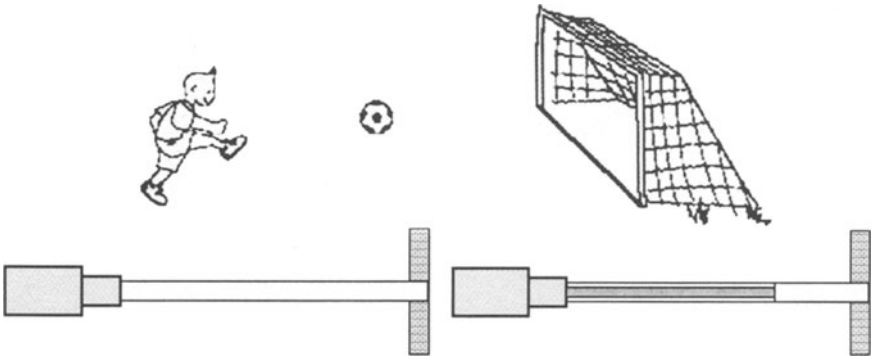


Fig. 3. Pressing and maintaining the pressure on the contactor.

At level 3, the exercise is testing the scrolling activity. A cross is presented in a square. A frame moves sequentially from one square to the next one, at a certain speed. The user has to select (to press the contactor) when the frame is around the square with the cross. The scrolling may be horizontal (Fig. 4), line/column or quadrangular. Scrolling speed is adjustable. This type of test is very important as certain children with cerebral palsy interact with the computer only by means of scrolling devices. The test enables, among other things, scrolling speed to be adapted to any software used in the future.

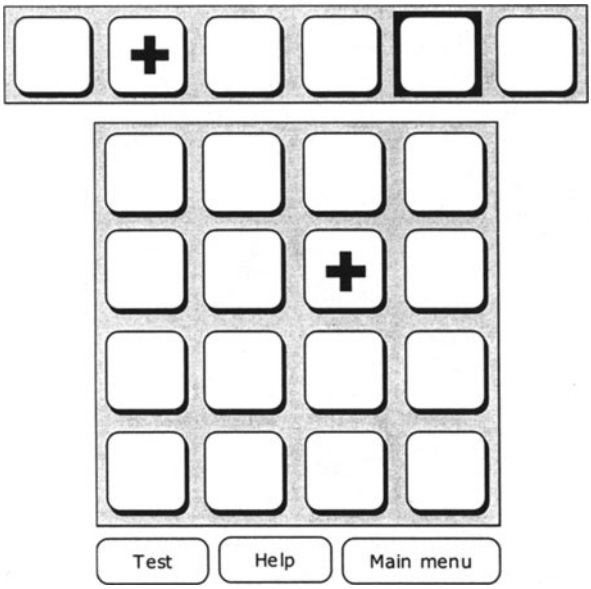


Fig. 4. User selection of a figure from a scrolling sequence

4.2 Stage 2: Screen Navigation

This stage measures the ability to move the cursor about in the screen space using an indirect method like mouse, trackball, head-mounted mouse, etc. At level 1, several exercises are proposed to evaluate the ability of navigation through moving and designating moving targets. The first exercise enables the user to designating a target using the current cursor. The target appears at different places on the screen (Fig. 5). The exact position or distance from the position of the cursor at the moment of the "click" to the center of the target is recorded. This allows the size of the targets to be adjusted in further test. This test also provides information about the size needed for the squared objects employed in other applications. The second exercise is similar to the first one but this time the size of the square is determined by the user's performance in the preceding exercise and the traced path pattern is recorded as a series of points reached by different intervals of time.

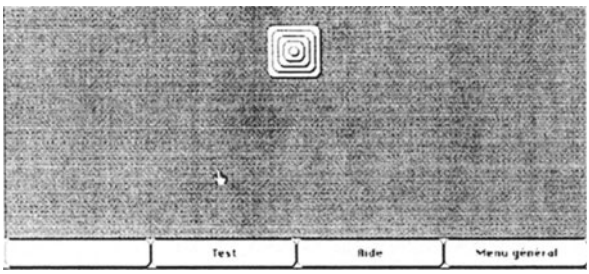


Fig. 5. Designating a target using the cursor.

At the end of the test, this traced path can be displayed (Fig. 6) and printed. There is no learning phase for this test. As observed [6], there is a cyclic combination of ballistic phases (consisting of rapid mouse movements) and homing phases (consisting in reaching the target precisely). Fig. 6 shows the importance of the homing phases, while ballistic phases seem to remain similar to those of able-bodied users.



Fig. 6. Record of the mouse path on the screen.

At level 2, several exercises are proposed to test the moving of an object so as to place it correctly at foreseen places. These exercises require good manual and visual coordination. The first exercise at this level is based of a section of a racing-car circuit. The screen displays a circuit from which one or more sections have been removed. A piece is provided at the base of the screen (Fig. 7). The task is to move the cursor and place it on the piece, to move the piece while moving the mouse and, by keeping it pressed down, put it where it belongs on the circuit. When the part has been correctly placed in position, the child releases the pressure on the mouse and confirms its selection by clicking on the appropriate screen button.

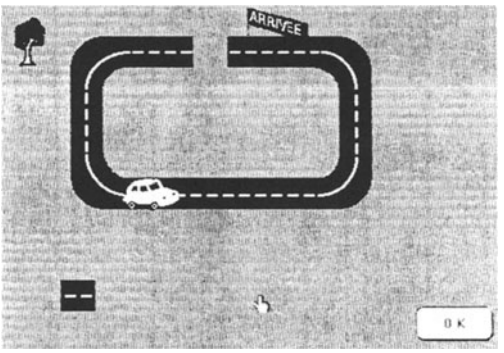


Fig. 7. Racing-car circuit.

Exercises two and three at this level were inspired from the Gesell board (Fig. 8) [14] where one or all the forms displayed have to be placed in one position or several. The moving about principle is the same as before but it is a matter of recognizing the geometric forms.



Fig. 8. Test inspired by the Gesell board.

In these two exercises the therapist can observe the plan of action and the strategy employed by the child to place each figure. For certain children, the multiple type of placing the forms is complicated, depending on the visual strategy they employ. The fourth exercise at this level was suggested by the Terman-Cesselin tests [14] and is also based on geometric figures, which have to be recognized and moved (Fig. 9). For each of the four exercises, the child is allowed a margin of accuracy in respect of the spot where the object must be placed. The length of the training, the test duration, the response time, the pressing down time and the number of times that the child releases the button on the contactor before finally placing the object are all measured and recorded. The screen captures allow the instructor to note the child's accuracy as well as the mistakes made.

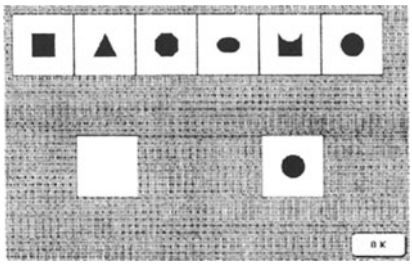


Fig. 9. Test suggested by the Terman-Cesselin tests.

The level 3 focuses on the activity of linking up points by pressing on the contactor while moving the cursor on the screen. A line appears from the starting points to the point reached (Fig. 10).

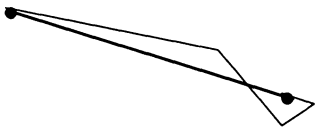


Fig. 10. Linking up points.

4.3 Stage 3: Controlling Hidden Information

In the exercises at this stage, part of the useful information is hidden, the aim being to display the information being sought by employing pull-down systems. At level 1, a linear structure is tested. To illustrate the notion of "preceding" and "following" (or "previous" and "next"), the screen displays a book, whose pages must be turned, first by clicking on the lower right-hand corner of the book, then by clicking on the arrows situated outside the book (Fig. 11).

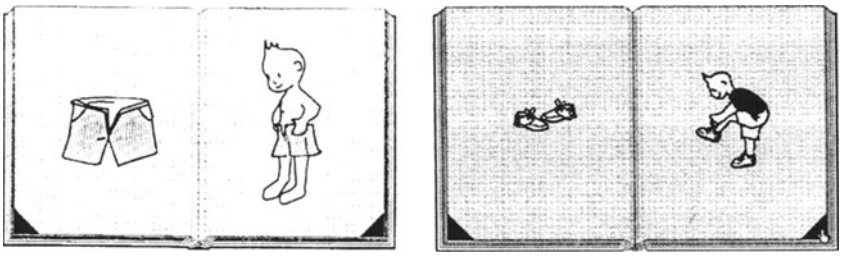


Fig. 11. Book featuring a sequential story.

At level 2, the task consists in finding among a set of square figures of different shades the figure that matches the proposed model item. The squares are displayed by means of pull-down bars and arrows. Three exercises are provided: a horizontal ordering of the figures of different shades, vertical for figures of different sizes or rectangular for figures ordered horizontally from lightest to darkest and vertically from smallest to largest (Fig. 12).

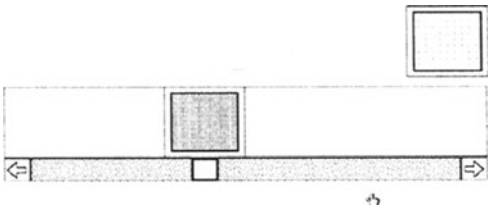


Fig. 12. Series of squares and user selection of a square in 1D.

A model is provided and the child must find the right square in the picture, which may be one- (Fig. 12) or two-dimensional (Fig. 13) by moving the only visible square using the pull-down arrows.

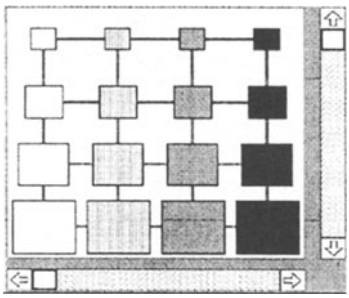


Fig. 13. User selection of a square in 2D.

The level 3 is similar to the level 2, except that information is conveyed in a circular structure. The test is to find a number in a circular structure part of which is hidden. A number is proposed on the top of the screen. The first exercise allows the wheel to be moved by clicking on the hands. In the second exercise, it is no longer possible to click on the hands; instead, the child must click on the arrows placed at the bottom of the screen (Fig. 14).

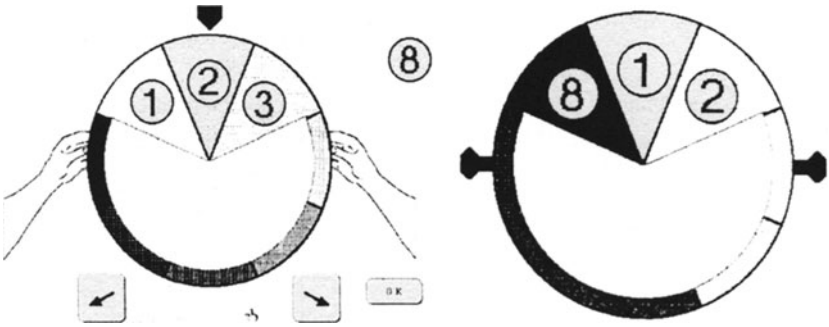


Fig. 14. Circular structure with arrows (a) and handles (b).

5 Experiments

The ERGOLAB software is regularly used at the Kremlin-Bicêtre Hospital. Two very typical cases are here described to depict how ERGOLAB can be useful to evaluate the usage of interaction techniques and to decide adaptation accordingly.

5.1 Dominique (Aged 11)

Dominique is severely handicapped. He is kept stable and protected from orthopedic deterioration by a corset-seat with head cap and leg extensions. He moves about in an electronic wheelchair to which this protective ensemble is attached. As he does not have sufficient hand control, he operates the control lever by means of a head-mounted mouse. Since he cannot speak, he communicates using speech synthesis by means of the head-mounted mouse.

He is using communication software in BLISS code, a kind of abridged pictographic writing. Its use requires memorizing sequences of dual entry tables. On several occasions Dominique has performed the first stage of ERGOLAB (pressing on the contactor). ERGOLAB has enables the best environment to be worked out for Dominique (*Mouvement Site Interface Positionnement*). This MSIP is linked to determining the part of the body that is best controlled and choosing a means of access accurately positioned for using it. This concept was originally in the Motor Evaluation software developed in the early generations of Apple, the value of which influencing our adopting it.

After comparing the reliability of various types of contactor, we decided on a remote control cable from the left temple, which could nevertheless be sued in a resting position, by leaning on the support at the nape of the neck. It should be noted that the results are very variable according to the environment, reflecting a very marked tendency to fatigue in respect of internal and external occurrences.

The test at level 3 (two-dimensional scrolling) has enables the speed of the automatic scrolling to be regulated.

5.2 Stéphane (Aged 10)

Stéphane was a premature dyspraxic baby, i.e. he experienced cerebral lesions, which prevent him coordinating several operations involved in spatial relations. On the other hand, his language development has been normal and psychometric assessment using the Zescher scale (WESC) shows a clear superiority of verbal intelligence quotient, which is normal, over visual-spatial performance quotient, which is at least 20 points lower. Dyspraxia disturbances are accompanied by visual problems (e.g., distinguishing a figure from its background, constancy of form). The computer provides many advantages for this type of handicap, as

- The visual field is restricted.
- The child can manipulate information in standard and comprehensible fashion.
- Action space and representation space are kept well apart.

However, one can imagine that the usability of the screen of a program is of vital importance in this case. At 8, Stéphane was reading simple texts but was unable to copy an oblique line and produce a triangle. He could transcribe only a few letters but could find them using a keyboard. Two years later, as a result of patient teaching, he can use the must of a simple word processor adapted to his abilities, record and find his files. He uses a trackball and sometimes even the ordinary mouse, but which he revolves very strongly without attempting to hold straight.

Stéphane has performed all the tasks provided by ERGOLAB. He has, of course, experienced the greatest difficulty with the navigation tasks (stage 2). In order to better foresee the path to be taken, we recommended the use of a trackball equipped with a braking system, which limits its force. The occupational therapist has noted a sudden loss of attention at a certain level of difficulty when the child disconnects. Through the stage 3 exercises, Stéphane has been able to understand how pull-down menus operate, although it was not our intention to teach him this at the outset. These scrolling menus and windows are based on the Macintosh style guide with Human Interface guidelines applied [5], without him having understood them before, but through short, interlocking exercises of progressive difficulty, he was better able to understand their working.

6 Evaluating the Software

To evaluate ERGOLAB, we decided to take a group of non-handicapped children, as this would allow us to have a large and more homogeneous group; actually, the population of handicapped children with cerebral palsy is very heterogeneous, which makes it difficult to reach sound statistical results. This evaluation enables us to detect inconsistencies or gaps in the proposed software. Of course the results of this study will not be entirely applicable to the case of children with cerebral palsy but we think that if some exercises appear to be not adapted and too difficult for non handicapped children, they will be also not suited to children with cerebral palsy.

Experimentation is far from finished but a study involving a score of children aged

between 2 years 8 months and 7 years has already provide a number of interesting results. Exercise 1 of level 2 (screen navigation), which consists in moving part of a circuit to the correct spot, appears more difficult than exercises 2 and 3 which, however, require geometric forms to be recognized. The reasons for failing to perform this exercise are of two kinds:

- To put back certain sections, it is necessary to cross the circuit; if there is no pedestrian crossing, many children do not dare to do so!
- For certain exercises, several sections are missing and it is not easy to see where the part provided should go: many children proceed by trial and error; others do not notice that the part is incorrectly placed. This exercise will, therefore, have to be rethought.

Level 3 of the same stage (linking up points) has a serious defect: the point indicating the spot where the pointer controlled by the mouse is situated is scarcely visible, especially during the training phase because of the difficulty in distinguishing it against the gray background. This emerged even more clearly in trials with handicapped children, the difficulty being accentuated when the child's movements were uncoordinated. This defect is unfortunately due to the development environment and has not yet been corrected. Other less important defects also came to the surface, such as the need for training phases and intermediary exercises for stage 3 (controlling hidden information). We should also note that the gray shades are not always clearly distinguished, this not necessarily depending on age.

ERGOLAB is now currently used in different institutions for disabled children. These children do not have necessarily cerebral palsy. For example, some of them are trisomic. Therapists told us that ERGOLAB can give good help to measure the different use parameters (reaction time, scrolling speed, etc) and to know more precisely what are the difficulties of a particular child. After that, they use commercial software like KENIX to adapt the computer parameters to the child abilities.

7 Conclusion

Although ERGOLAB is not complete, it provides a hierarchy of exercises structured into 3 levels (Fig. 15). We have been able to appreciate its usefulness not only with children with cerebral palsy, but also with children with other handicap or even with non-handicapped children. We think that the software is useful in the following instances:

- In choosing a technological environment and computer configuration for a multideficient child;
- In regulating the parameters of existing programs for a given child;
- In choosing software well adapted to the cognitive abilities of the child;
- In specifying new software, which would be targeted for specific users.

Various other tests must be incorporated into ERGOLAB: we are currently working on defining tests that will enable us to check a child's ability to use various types of menu. We would again draw attention to the fact that, when evaluating, we were able

to appreciate the pedagogical quality of certain tests: this prompts us, therefore, to design another software which would be more software for learning basic forms of manipulation that would serve as an introduction to standard software. ERGOLAB could also provide a useful aid for consolidating and formulating context- specific accessibility guidelines [11]. The development of this important problem is beyond the scope of this paper and will be addressed in the future

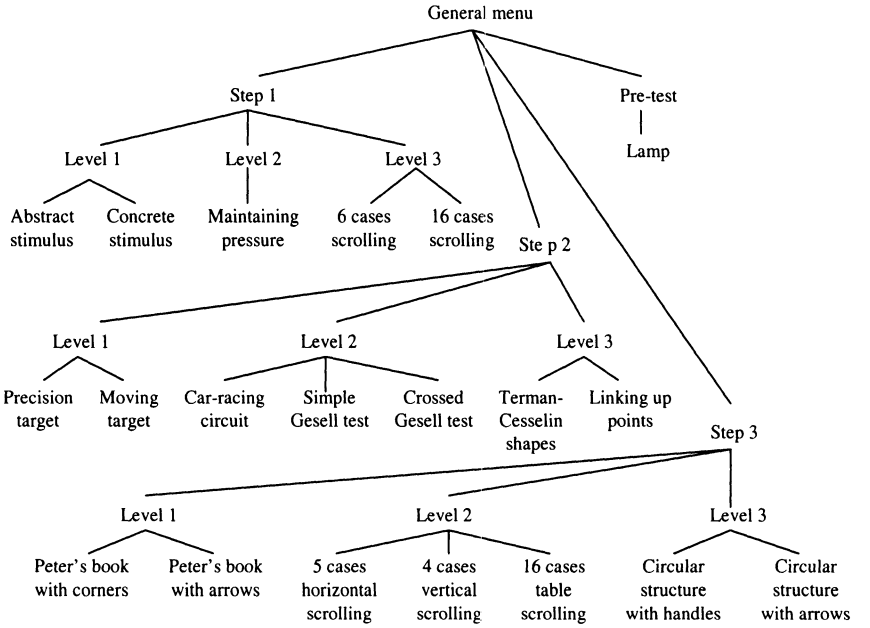


Fig. 15. Exercises hierarchy as proposed in ERGOLAB.

Acknowledgments

The authors would like to thank Claudy Bernard for developing the ERGOLAB software, Mrs. de Barbot, psychologist at Kremlin-Bicêtre Hospital for her advice, and the anonymous reviewers for their helpful comments on earlier version of this manuscript.

References

1. Akoumianakis, D., Stephanidis, C.: Supporting User-Adapted Interface Design: the USE-IT System. Interacting with computers 9, 2 (1997) 73–104
2. Bernard, Cl.: Evaluation d'ergonomie d'écrans pour logiciels destinés à des enfants Infirmes Moteurs Cérébraux. Master of Computer Science thesis. Institut d'Informatique, Facultés Universitaires Notre-Dame de la Paix, Namur (1992)
3. Brousmische, L, Gillet, X.: Logiciel de simulation d'un jeu de marionnettes pour enfants Infirmes Moteurs Cérébraux. Master of Computer Science thesis. Institut d'Informatique,

Facultés Universitaires Notre-Dame de la Paix, Namur (1990)
4. Deplechin, M., Strappazon, G., Noirhomme-Fraiture, M, Mercier, M.: Logiciel de gestion des comptes. In Proc. of 12th International Congress on Cybernetics (Namur, August 1989), IEA Press (1989)
5. DMI Macintosh Software. Centre Marc Sautelet, Villeneuve d'Ascq (1992)
6. Keates, S., Clarkson, P.J., Robinson, P.: Developing a methodology for the design of accessible interfaces. In Stephanidis, C., Waern, A. (eds.): Proc. of 4th ERCIM Workshop on "User Interfaces for all", Special theme "Towards an accessible Web" (Stockholm, 19-21 October 1998). ERCIM Press, Stockholm 1998. Accessible at http://www.ics.forth.gr/proj/at-hci/UI4ALL/UI4ALL-98/keates_ps.zip
7. McMillan, W.W., Zeiger, M., Wisniewski, L.: A Rule-Based System that Suggests Computer Adaptations for Users with Special Needs New Directions. In Proc. of 1st Annual ACM Conference on Assistive Technologies. ACM Press, New York (1994) 129–135
8. Noirhomme-Fraiture, M.: Evaluation de logiciels pour personnes handicapées mentales. In: Actes du Colloque « Les apprentissages chez les personnes handicapées mentales modérées et sévères » (Namur, 1 June 1990)
9. Noirhomme-Fraiture, M.: Evaluation of Software for People with Mental Disabilities. In Proc. of World Congress on Technology (Washington, 1-5 December 1991). Vol. 5 (1991) 810–821
10. Noirhomme-Fraiture, M., Charrière, Cl., Vanderdonckt, J., Bernard, Cl.: ERGOLAB: A Screen Usability Evaluation Tool for Children with Cerebral Palsy. In: Adjunct Proc. of ACM Conf. on Human Aspects in Computing Systems InterCHI'93 (Amsterdam, April 1993). ACM Press, New York (1993) 23–24. Accessible at http://www.qant.ucl.ac.be/membres/jv/publi/InterCHI93-ErgoLab.pdf
11. Noirhomme-Fraiture, M., Vanderdonckt, J.: Screen Usability Guidelines for Persons with Disabilities. In: Adjunct Proc. of ACM Conf. on Human Aspects in Computing Systems InterCHI'93 (Amsterdam, April 1993). ACM Press, New York (1993) 25–26. Accessible at http://www.qant.ucl.ac.be/membres/jv/publi/InterCHI93-Usability.pdf
12. Stephanidis, C., Akoumianakis, D., Savidis, A.: Design Representations And Development Support For User Interface Adaptation. In: Stephanidis, C. (eds.): Proc. of 1st ERCIM Workshop on "User Interfaces for all" (Heraklion, 30-31 October 1995). ERCIM Press, Heraklion 1995. Accessible at http://www.ics.forth.gr/proj/at-hci/UI4ALL/_UI4ALL-95/stephani_dis.ps.gz
13. Stephanidis, C., Salvendy, G., Akoumianakis, D., Bevan, N., Brewer, J., Emiliani, P.L., Galetsas, A., Haataja, S., Iakovidis, I., Jacko, J., Jenkins, P., Karshmer, A., Korn, P., Marcus, A., Murphy, H., Stary, C., Vanderheiden, G., Weber, G., Ziegler, J.: Toward an Information Society for All: An International R&D Agenda. Int. J. of Human-Computer- Interaction. 10-2 (1998) 107–134
14. Terman-Cesselin test. Gesell test. EAP, Ed. Scientifiques et Psychologiques, 6 bis, rue André Chénier, F-92130 Issy-les-Moulineaux, France (1992).
15. Vanderdonckt, J., Bodart, F.: Encapsulating Knowledge for Intelligent Interaction Objects Selection. In: Proc. of ACM Conf. on Human Aspects in Computing Systems InterCHI'93 (Amsterdam, 24-29 April 1993). ACM Press, New York (1993) 424–429. Accessible at http://www.qant.ucl.ac.be/membres/jv/publi/InterCHI 93-Encaps.pdf