Citation:
Katre D (2005). Juxtaposing Commentary of Task Performance (CTP) and Interface Play Script (IPS) for Discovering Interface Metaphors, In Proceedings of International Conference on Information Management (ICIM), Mumbai, India, Volume II, pp. 809-818.

Juxtaposing the Commentary of Task Performance & Interface Play Script: A Method for Discovering Interface Metaphors

By

Dinesh S. Katre

National Multimedia Resource Centre
Centre for Development of Advanced Computing (C-DAC)
Ministry of Communications and Information Technology, Government of India,
Agriculture College Campus, Near District Industries Centre,
Shivaji Nagar, Pune 411005, India. Tel/Fax: 91+020+5533250
Personal Homepage: www.hceye.org
dinesh@cdacindia.com

ABSTRACT:

It is observed that the user interface designer always faces two different states of interface metaphor namely 'unmanifested' and 'manifested'. The journey of the interface metaphor from 'unmanifested' to 'manifested' state is yet to be defined. Therefore, it is proposed to write the Commentary of Task Performance (CTP) and Interface Play Script (IPS) and then juxtapose them against each other for discovering the probable interface metaphors. Both CTP and IPS need to be written in the form of a typical drama script. This method was used for identifying the interface metaphors while designing JATAN: Virtual Museum Builder software. JATAN is a digital library solution specially designed for museums. Proposed method is also compared with the 'cognitive walkthrough', which is an interface evaluation technique and 'use cases'.

KEYWORDS:

User Centred Design, Interface Metaphor, Usability, Mental Model, Sensory Clues, Task, Script, Cognitive Walkthrough, Virtual Museum, Digital Library

1. INTRODUCTION:

C-DAC's National Multimedia Resource Centre has developed a digital library solution called as JATAN:Virtual Museum Builder. In this project, we were keen to capture the mental model of museum staff [Katre, 2004] through the interface design of JATAN. Also, we wanted to align the software procedures along with the regular workflow of museums. Naturally, there was no option but to search for suitable interface metaphors. User Centred Design (UCD) approach recommends techniques like Contextual Inquiry [Beyer, 1996], Field Study, and Ethnographic Study as general solutions to many usability problems [Dix and colleagues, 2004]. As per the standard practices of UCD approach [Moll-Carrillo, 1995] [Lundell, 1995], we studied the activities and the work environment of users.

All such techniques highlight the need of focusing on the user but no specific method for finding the interface metaphors [Vaantinen, 1994] is proposed so far. Lewis and Rieman [1994] in their paper on Task Centred User Interface Design have not mentioned about interface metaphors. Moll-Carrillo, Lundell and other UCD theorist seem to have assumed that the interface metaphor is already discovered and straight away one can proceed for user trials. Erickson [2000] and Hudson [2000] have suggested metaphoric options quite spontaneously without mentioning how they arrived at those. Most experiments pertaining to interface metaphor have started at the interface metaphor itself. It is unexplained how did the candidate interface metaphor come into existence and how did one notice it? Brenda Laurel [1993] in her book entitled ‘Computer as Theater’ also does not mention how the interface metaphors can be discovered.

2. STATES OF INTERFACE METAPHOR

User interface designer always faces two different states of interface metaphor namely ‘unmanifested’ and ‘manifested’.

▪ Unmanifested Metaphor State

In the beginning of software project, many times, user interface designer is clueless about which interface metaphor will be suitable. The list of software requirements is not indicative enough. (S)he is stuck up thinking about where to find the interface metaphors. There is a general assumption that the interface metaphor is always present in the context but the user interface designer may not have noticed it. This state of interface metaphor is called as ‘unmanifested metaphor state’.

▪ Manifested Metaphor State

In some cases, the candidate interface metaphor immediately clicks to the user interface designer. At times, (s)he foresees the probable metaphor even before starting to elicit software requirements. In some occasions, the customer brief explicitly suggests the desired interface metaphor. This state is called as ‘manifested metaphor state’. It means one does not require searching the interface metaphor. The transition of candidate interface metaphor from ‘unmanifested’ to ‘manifested’ state needs to be defined. Usually the ability of identifying the suitable interface metaphor is attributed as insight, talent and creativity. One tries to seek inspiration from various sources [Katre, 2002]. Therefore, in order to help the user interface design community, we propose a method through which the candidate interface metaphors manifest. In this case one does not require relying so much on things like talent and insight.

3. METHOD FOR DISCOVERING THE INTERFACE METAPHOR

It is proposed to write the Commentary of Task Performance (CTP) and Interface Play Script (IPS). **This approach has entirely different objectives and hence it should not be mixed with use cases [Schneider and colleagues, 1999] or cognitive walkthrough techniques [Polson and colleagues, 1992]. The comparative observations are presented in point 4.**

Overall Guidelines

- Both CTP and IPS have to be written in the manner of a typical drama script. The script is divided on the basis of ‘tasks’ similar to ‘scenes’ in a drama script.
- **The main focus of documentation should be on detailing the experiential factors (The sensory clues in terms of visual, spatial, verbal, auditory and tactile details familiar to the user.) of the task being performed.**
- The main purpose of this activity is to capture the mental model of users.
- In place of ‘character names’, one has to mention the participants involved in a task e.g. operator, director, manager, salesman, etc. The list of participants of a task, description of the situation, assumptions if any have to be documented before starting to write the CTP and IPS.
- The language of writing the commentary and the script should be simple and straightforward.
- Having finished the documentation of CTP and IPS, both should be juxtaposed and compared for identifying the unmanifested metaphors.

Commentary of Task Performance (CTP)

CTP focuses on conventional processes in user environment.
Writing the CTP involves following steps-

- Observe the selected tasks performed by the targeted user using conventional processes. These tasks should be the ones that are expected to be computerized. User interface designer has to repeatedly observe each task performance until the common pattern and flow of activities become evident.
- CTP should mention all non-trivial and trivial objects used for completing the task and the purpose for which they were used. The sensory clues in terms of visual, spatial, verbal, auditory and tactile details related to the task should be documented. Users need these clues for predicting the status of the task. These details define the mental model of users.
- The commentary of such task performances by the users should be documented in narrative form. It should include the details of situations, options, goals, actions, decisions, activities and the outcomes.
- It should also include the documentation of significant interruptions, dependencies and constraining factors faced while performing the task.
- Photographs or video recordings of the key steps of the specified task may be taken. (The author of the paper has tried using camera-phone and digital camera for this purpose.)

Interface Play Script (IPS)

Preparation of IPS involves following steps-

- User interface designer should now visualize how the user would perform the same task using the proposed software.
- The visualization should be documented in terms of inputs given by user and the response of system.
- The documentation may include general details of user interface. There is no need to stretch your imagination to mention component level details of user interface.
- Source of user input (background work) should also be mentioned.

4. COMPARISON WITH COGNITIVE WALKTHROUGH

Polson and colleagues [1992] to evaluate the user interface reintroduced cognitive walkthrough technique, which was originally used for evaluation of code in software engineering. In this approach, the evaluator steps through every action that user will perform for achieving a goal. In cognitive walkthrough, the objective is to check the actions, user goals, communication of actions and feedback. This method is to be used after the prototyping stage [Dix and colleagues, 2004]. It focuses on identifying the probable usability problems.

Table 1. shows the principal differences between the cognitive walkthrough and the method proposed in this section. The reason for directly comparing it with cognitive walkthrough is that both methods deal with user interface.

Table 1. Comparison of IPS and CTP with Cognitive Walkthrough

Cognitive Walkthrough	**IPS and CTP**
Action sequence is restricted within the software.	Focus on the details of context, situations, objects, human roles involved in the task along with the proposed software.
Technical description of actions, user interface and goals [Rieman and colleagues, 1995]	Main focus is on capturing the experiential aspects (look and feel) of a task.
The purpose is not to capture users' mental model	The main purpose is to captures users' mental model.
It is an evaluation method.	It is a visualization method.
Objective is to find usability problems.	Objective is to find the candidate interface metaphors.

Above comparison conclusively brings how the cognitive walkthrough and the method proposed in this paper are different. Whereas, the prime objective of use case modeling is to define the system functionality in object oriented software development [Schneider and colleagues, 1999]. The technical report of Usability Throughout SDLC: A Summary of Interact 2001 Workshop prepared by Jan Gulliksen and Colleagues concludes the debate on Use Case versus Task Analysis by mentioning that use case modeling does not address the usability requirements. It also mentions that greater amount of detailing is desired from the viewpoint of user interface designers. It corroborates with the need of the method proposed in this paper.

During the development of JATAN: Virtual Museum Builder the user interface designer faced the 'unmanifested metaphor state'. The proposed method was used for discovering the candidate interface metaphors. The format used for documentation of CTP and IPS is shown in table 2.

Table 2 Juxtaposing CTP and IPS

Task 1. Creation of a museum record

Participants:
Data Entry Operator, Curator, Curatorial Assistant, Director, Office Boy, Proposed Software

Assumptions:
The Data Entry Operator, Curator and Director are always looking through the web browsers and waiting for change in the status of a given record. They know the result of setting a particular flag.

	Commentary of Task Performance	Interface Play Script
1.	**Curator:** Prepares the draft records by writing them on separate sheets. He refers books, consults experts and finishes the draft version.	**Curator:** Prepares a hand written record of an artifact and gives it to the Data Entry Operator. *(Either he delivers it himself or asks the office boy to hand it over)*
2.		**Data Entry Operator:** On receiving the hand written record, he clicks on new record button in the Integration Module of VMB.
3.		**Software:** Provides a new form before for data entry.
4.		**Data Entry Operator:** Clicks the browse button for selection of relevant images.
5.		**Software:** A typical file browser window appears on the screen with thumbnail images.
6.	**Curator:** Provides a photograph or a transparency from an album maintained by museum. He uses a light box for viewing the transparencies.	**Data Entry Operator:** Selects the image by matching the accession no. of a record with the image file name. Confirms the digitized image by tallying it with the transparency. On finishing the data entry he sets the flag as ‘Finished’. If the data entry is incomplete, he sets the flag as ‘Unfinished’. An unfinished record is not forwarded to Curator.
7.		**Software:** On setting the flag as ‘Finished’ the software flashes a message that ‘The finished record is now forwarded to Curator for verification.’ (A ‘finished’ record is read only.)
8.		**Curator:** Checks the record, comments it if it has typographic errors and sets the flag as ‘Commented’. (A commented record is editable.)
9.		**Data Entry Operator:** Reads the comment. Opens the record. Corrects the typing errors and then sets the flag as ‘revised’.
10.		**Software:** Flashes a message that the revised record is forwarded to Curator for verification.
11.		**Curator:** Opens the record. Checks the revised record and sets the flag as ‘checked’. (A checked record is read only.)
12.	**Curator:**	**Software:**

	The handwritten record is forwarded to the Director of museum in an envelop / folder with a covering note requesting his approval. (Covering note is dated and signed).	Flashes a message that the checked record is now forwarded to Director for approval.
13.	**Office Boy:** Picks up the envelope/folder and delivers it to the Director. (Keeps it in the In Tray of Director)	
14.	**Director:** Writes / makes corrections in the draft records, signs, puts the date and then marks the folder / envelope back to Curator. Uses red/ green colored ink pen for writing. The envelope/folder is kept in the Out Tray.	**Director:** Finds a Checked record submitted for approval. He verifies the contents and in case of correction 'Comments' it.
15.	**Office Boy:** Picks up the envelope/folder and delivers it to the Curator. It is kept in the In Tray of Curator.	
16.	**Curator:** Opens the folder and finds Director's comments for corrections.	**Curator & Data Entry Operator:** Work towards improving the record as instructed by the Director and set the flag as 'Finalized'.
17.	**Curator:** Incorporates the corrections (rewrites it) and submits it again by placing his signature and date to Director for approval.	**Software:** Flashes a message that the Finalized record is now forwarded to Director for approval.
18.	**Director:** Approves it (places his signature and date) and returns it to Curator.	**Director:** Verifies the contents of finalized record and checks the history of comments so far. Sets the flag to 'Approved'.
19.	**Curatorial Assistant:** Enters it in the Main Accession Register.	The approved record is automatically added in the Main Accession Register.

It is possible to detail a particular step further in similar manner. For example, the sample commentary presented in Table 2. does not provide enough details of manual record preparation and data entry using the software. User interface designer can decide to write elaborate commentary of the sub-tasks.

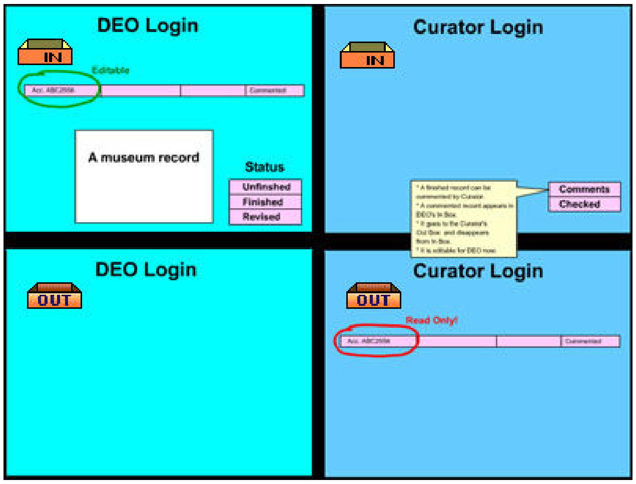
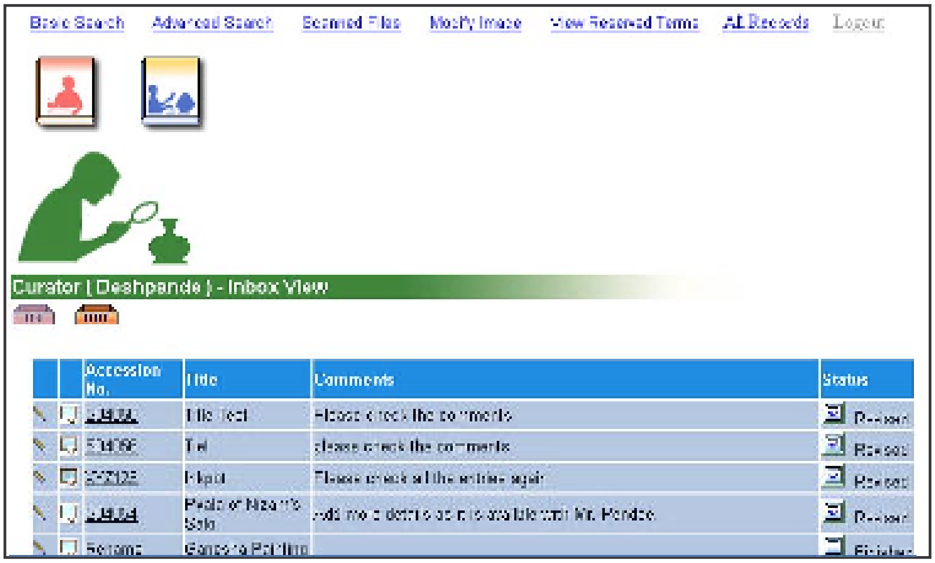

DEO Login
IN
Editable
Acc. ABC2556
Commented
A museum record
Status
Unfinshed
Finished
Revised
Curator Login
IN
* A finished record can be commented by Curator
* A commented record appears in DEO's In Box.
* It goes to the Curator's Out Box and disappears from In Box.
* It is editable for DEO now.
Comments
Checked
DEO Login
OUT
Curator Login
OUT
Read Only!
Acc. ABC2556
Commented

Figure 1. Traversal of a record through In/Out trays rendered using multimedia

Figure 1. shows how the traversal of a record through in/out trays of data entry operator, curator and director is rendered using multimedia. Three different consoles of the specified users were visualized in the same screen to indicate the traversal of a record and the change in its status.



		Accession No.	Title	Comments	Status
		[illegible]	Title Test	Please check the comments	Revised
		534098	Tel	please check the comments	Revised
		232125	Mapa	Please check all the entries again	Revised
		[illegible]	Peda of Nizam's Sala	add more details as it is available with Mr. Pandse	Revised
		Refrome	Ganesha Painting		Finished

Figure 2. The screenshot of JATAN: Virtual Museum Builder software

5. INTERFACE METAPHORS DISCOVERED

1. It was possible to identify the candidate interface metaphor like ‘In and Out Trays’ in the offices of Curator and Director (Steps 13. and 15.). It is used for representing the received and forwarded records in JATAN.

2. Many steps in CTP show that the task is ending up with a register entry (Step 19.). ‘Register’ is the most commonly used term in Indian museums. The museums have variety of registers like gallery, section, store, loan, conservancy, main accession, etc. ‘Register’ metaphor is used in JATAN for representing various modules like Data Entry, Administration, Subscription, Main Accession, etc.

3. We have introduced a concept of ‘Image Catalog’ for arranging the preview of digitized images (Step 6.).

4. Every time, the Curator and Director place their signatures with dates after recording their decisions (Steps 12. and 18). In JATAN, a record becomes ‘Read Only’ for the sender after (s)he comments and forwards it. The history of comments shows all transactions with dates.

5. ‘Closed and open envelopes’ are used with unique markings to indicate the status of a record (Steps 12. to 15.). Each record comes with a flag indicating its status such as finished, revised, checked, commented, approved, published, withdrawn, etc.

6. Different ink colors are used for presenting the comments of curator and director (Step 14.).

The advantage of using CTP and IPS is that it helps in conceiving the interface metaphor during the requirements elicitation stage. It helps in designing the software with full knowledge of the interface metaphor. Interface and software designers can foresee the design implications.

The author of this paper has applied this method in many other software projects. The task scenario related to JATAN: Virtual Museum Builder is one of the examples of this kind.

CONCLUSIONS

Comparison of CTP and IPS is helpful in discovering the candidate interface metaphors. This method captures the mental model of users associated with the tasks taken up for computerization. It mainly captures the experiential details (The sensory clues in terms of visual, spatial, verbal, auditory and tactile details familiar to the user.) of the task, which are most essential for interface design. CTP and IPS also provide the justification for selecting a particular interface metaphor. This method is applicable only for computerization type of software projects.

ACKNOWLEDGEMENTS

I am extremely grateful to Dr. Mukul K. Sinha, Managing Director, Expert Software Consultants, New Delhi for his valuable guidance and for giving confidence to me in my explorations. I also thank C-DAC and the team at National Multimedia Resource Centre for their unstinting support.

REFERENCES

1. Beyer H, Haltzblatt K (1996). Contextual Techniques, Interactions (6)1, pp. 32-42

2. Dix A, Finlay J, Gregory D, Abowd, beale R (2004). Human Computer Interaction, Published by Pearson Education

3. Erickson, T (2000). Working with Interface Metaphors, Readings in HCI: Towards the year 2000, Edited by Baecker R, Grudin J, Buxton W, Greenberg S, Published by Morgan Kaufmann, II Edition, pp. 147-151

4. Gulliksen J (2001). The technical report of Usability Throughout SDLC: A Summary of Interact 2001 Workshop

5. Hudson W (2000). Metaphor: A Double-Edged Sword, ACM Interactions, May + June 2000, pp. 11-15

6. Katre D (2002). Unconventional Inspirations for Creating Software Interface Metaphors, in Proceedings of International Conference on Media and Design (Mumbai, India, Sept. 2002), Volume I, 1-15.

7. Katre D (2004). Pragmatic and Usable Approach for Digital Library Initiatives in India, Proceedings of International Conference on Digital Libraries (ICDL), New Delhi, India.

8. Lewis C and Rieman J (1994). Task-Centered User Interface Design: A Practical Introduction
   Retrieved on 13 Oct. 2004 from
   http://hcibib.org/tcuid/

9. Laurel, B. (1993). Computers as Theatre, Addison-Wesley

10. Lundell (1995). Designing a "Front Panel" for Unix: The Evolution of a Metaphor, In the electronic proceedings of CHI 1995, ACM Press.

11. Moll-Carrillo H, Salomon G, Marsh M, Suri J, Spreenberg P (1995). Articulating a Metaphor Through User-Centered Design, In the electronic proceedings of CHI 1995, ACM Press

12. Polson P, Lewis C, Rieman J, Wharton C (1992). Cognitive Walkthroughs: A method for theory based evaluation of user interfaces, International Journal of Man-Machine Studies, 36:741-73

13. Schneider G, Winters J (1999). Applying Use Cases: A Practical Guide, Published by Addison-Wesley

14. Vaantinen K and Schmidt J (1994). User Interfaces for Hypermedia: How to Find Good Metaphors?, In the proceedings of Conference Companion CHI’94, ACM Press, pp. 263-264