

シミュレータ対応 UI テスタによるユーザビリティ評価

岡田英彦, 松田良一*, 旭敏之, 井関治

NEC 関西 C&C 研究所

* NEC パーソナルコミュニケーション事業部

ユーザインタフェース(UI)設計ツール「OST」と使いやすさ評価ツール「UI テスタ」の統合による, 設計段階でのユーザビリティテストが可能な計算機環境を提案する. OSTによりUI部を設計し, シミュレーション機能を利用してユーザテストを行う. このとき収集した対話履歴をUIテスタで分析し, 使いやすさ上の問題点を抽出することができる. さらに, この評価結果をフィードバックして設計仕様を改善するという設計と評価のサイクルが, 効果的に実施できるようになった. これらのツールを実際の新製品開発に適用した結果, 問題点を9個抽出し, そのうち5個に対して, 出荷前に設計を改善することに成功した.

Usability Evaluation with UI-tester for User Interface Simulator

Hidehiko Okada, Ryoichi Matsuda*, Toshiyuki Asahi, Osamu Iseki

Kansai C&C Research Laboratory, NEC Corporation

4-24, Shiromi 1-chome, Chuo-ku, Osaka 540, Japan

(e-mail: h-okada@obp.cl.nec.co.jp)

* Personal Communications Division, NEC Corporation

Mutual integration of user interface design tool and computer-aided usability testing (CAUT) tool is proposed. This environment enables product engineers to adopt effective iterative design strategy in order to achieve high usability. These tools are Operation Simulating Tool (OST) and UI-tester, respectively. Having three main functions of designing, simulation and program code generation, OST has been implemented for developing user interface of facsimile machines. UI-tester is a CAUT tool which aims at evaluating product usability by analyzing user-machine dialogue data. It can show several forms of diagrams for an evaluator to see what types of troubles users had in operating the machine. Iterative design, which means designing-simulation-evaluation cycle, had been practiced for a new facsimile product. Nine usability problems were extracted by analyzing common user error patterns with UI-tester.

1 はじめに

使いやすさ(ユーザビリティ)を指向した製品開発の代表的な方法として、ユーザインタフェース(User Interface:UI)の反復的設計が挙げられる[1-4]。ここで反復的設計とは、使いやすさ評価とその結果に基づく再設計のサイクルを繰り返し行うことである。

この反復的設計を効果的に行うためには、製品開発の上流段階、つまりUIの設計段階において使いやすさ評価を行い、抽出された問題点をなくすために早期改良を行わなければならない。このための評価手法として、ヒューリスティック評価法[5]、コグニティブウォークスルー法[6]などや、チェックリストを用いた方法[7]、さらにはシミュレータを利用する方法[1]などがある。ヒューリスティック評価法やチェックリスト法では、被験者を用いずに済むことや、比較的低コストで実施できるといった利点があるが、ユーザ不参加型の評価であるために、実際のユーザに影響を与えるような問題点を見つけられるとは限らない[8]。これに対し、シミュレータを利用する方法では、ユーザ参加型の評価を実施できるという利点があり、[1]で紹介されているように、製品開発において実施されるケースが増えている。しかし、これらのシミュレータ評価例には次のような問題点がある。

- ハイパーカードやVTRを利用した画面遷移のシミュレーションを行っているので、シミュレータの作成作業を実際のUIプログラミングとは別の作業として行わなければならない。
- プロトコル解析などによる評価が行われるので、手作業にかかる工数が大きく、かつ評価結果の客観性を高めるのが難しい。

本研究では、このような問題点を解決するため、UI設計/評価の統合的計算機支援環境を構築することを目的とし、UI設計ツール「OST¹」を評価対象とした「シミュレータ対応UIテスト」の開発を行った。さらに、これらのツールを実際の新製品開発に適用した。本稿では、これらのツールについて説明するとともに、製品開発への適用をケーススタディとして報告する。

¹Operation Simulating Tool.

2 UIテストの概念

2.1 CAUT

ユーザビリティ評価の客観性と自動化を追求するためには、その計算機支援環境が有効であると考え、CAUTの概念を提案した[9]。使いやすさ評価全体を完全に自動化することは容易ではないが、計算機による処理が可能な側面を抽出し、積極的に計算機を導入することで、次のような効果が期待される。

- テストとデータ解析に要する工数を大幅に削減できる。
- 専門家を必要としないテスト環境を実現できる。
- 被験者数を増加して共通傾向を抽出することにより、個人差の影響を低減できる。つまり、評価結果の品質を均質化かつ客観化できる。

2.2 UIテストを用いた評価

CAUTの有効性を検証するため、評価ツールの一つとして「UIテスト」を開発してきた[9-12]。UIテストは、従来手法では困難であったユーザ対話の流れの抽出を対話履歴のダイアグラム表示によって実現し、これをベースに使いやすさの評価を試みるものである。

UIテストを用いた使いやすさ評価の基本概念は、UI設計によって規定される対話(評価対象のUI構造を反映していると考えられる)と、ユーザが試行錯誤的に行った実際の対話(同様にユーザモデルを反映していると考えられる)を比較し、その差異を抽出することによって、ユーザの視点に立った使いにくさの問題点を見つけることである(図1)。ここで、UI設計によって規定される対話とは、マニュアルに説明されるような基準の操作方法に従って行わ

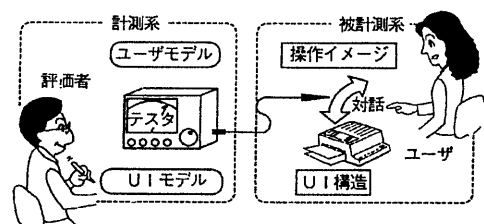


図1: UIテストの基本概念

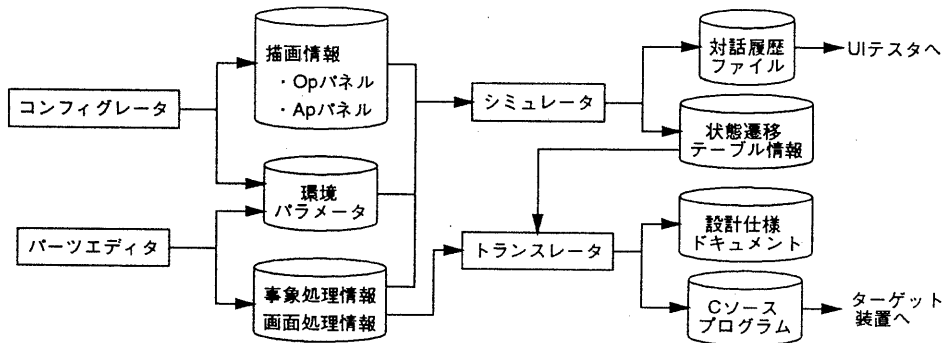


図 2: OST のシステム構成

れる対話のことであり、これを「基準対話」と呼んでいる。これに対し、ユーザが実際に試行錯誤的に行う対話を「ユーザ対話」と呼んでいる。ユーザモデルの記述方法としては GOMS [13] がよく知られているが、その目的がユーザの操作時間や心的負荷の予測であるのに対し、本研究ではユーザの行動を予測するのではなく、ユーザモデルが対話履歴に反映されていると仮定し、評価データとして用いる。

3 FAX 設計ツール OST

UI テスタが評価対象とする FAX 装置用 UI 設計 / シミュレーションツール OST は、オブジェクト指向型設計機能、シミュレーション機能、プログラム生成機能を有する。そのシステム構成を図 2 に示す。

(1) パーツエディタ

OST で用いられる部品は、基本部品と定義部品に大別される。パーツエディタでは、この基本部品や定義部品を用いて、FAX 装置の状態遷移構造を規定する。ここで、基本部品や定義部品はそれぞれ次の通り。

基本部品群 FAX 装置の UI 設計に用いられる事象処理や画面処理を規定するための基本的な関数や文の集合。これらの部品は約 120 個用意されており、OST のプログラムに組み込まれている。

定義部品群 基本部品のいくつかを組み合わせ、1 つの部品ファイルとして定義したもの。これにより、設計者が多くの箇所共通して用いるような基本部品の系列を、定義部品として登録しておくこと

ができる。

また、OST における FAX 装置の状態遷移構造は以下のようなファイル構成で規定される。

状態規定ファイル FAX 装置が取り得る状態 XXXX を規定する。ここで、XXXX は状態コード (16 進数 4 けた) に対応する。

初期処理規定ファイル 状態 XXXX へ遷移するときの初期処理を規定する。

画面処理規定ファイル 状態 XXXX における操作パネル (Op パネル) 上の画面表示構成を規定する。

事象処理規定ファイル 状態 XXXX において、FAX 装置のユーザが操作可能なキーを規定する。

状態遷移規定ファイル 状態 XXXX において、キーコード *keycode* で表されるキーを押下したときの状態遷移処理を規定する。

(2) コンフィグレータ

FAX 装置の操作パネルウィンドウ (図 3)、および操作パネル以外でのユーザ操作や FAX 装置のエンジン部分の動作をシミュレートするためのアプリケーションパネル (Ap パネル) ウィンドウを描画するための情報を規定する。

(3) シミュレータ

Op パネルを用いて、設計した UI のシミュレーションを行う。このとき、ユーザの対話履歴を収集することができる。対話履歴の例を付録に示す。現在、

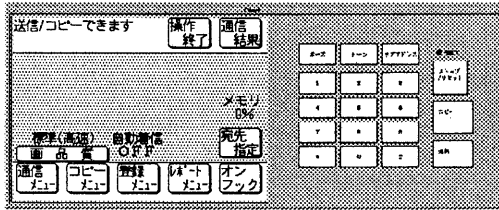


図 3: 操作パネルウィンドウ

FAXの送信あるいは書類のコピーにおける用紙の処理などの動作は、Apパネルによる操作で代行している。しかし、将来的には、このような動作まで完全にシミュレートできるような拡張を行うとともに、FAX装置や送信書類を表すアイコンなどを用いて実際の製品の操作状況を図的にシミュレート可能にすることを検討している。

(4) トランスレータ

設計情報を最終的に装置に組み込むために、テーブルデータをCプログラムに変換し、また、オペレーションマニュアルの生成を支援するために、操作制御仕様書に変換する。この変換は簡単な規則の集合に基づいて行われる。現在、操作制御仕様書ファイルの中味は設計データファイルとほとんど同じであるが、将来的には、仕様書ファイルをPostScriptで記述するようにし、仕様を図的に表現できるように拡張することを検討している。

4 シミュレータ対応 UI テスタ

UI テスタの評価対象を実機からOSTへと拡張し、シミュレーションベースでの評価が可能になった。UI テスタのシステム構成を図4に示す。

(1) 対話履歴

OSTのシミュレーション機能を利用した実験により収集されたユーザ対話履歴と、各タスクに関する基準対話履歴が保存される。このような対話履歴は、ユーザの操作コードと装置の状態コードの基本対が中心となっている²。

²対話履歴ファイルには、この他に操作時間間隔などのデータも含まれる。

(2) ユーザ誤対話ボタン

ユーザ対話履歴を基準対話履歴と先頭から順に比較し、ユーザの誤対話ボタンを抽出する。この誤対話ボタンは、ユーザ対話履歴のなかで、基準対話と手順が異なっている部分のことである。ユーザ対話履歴ごとに、対話履歴比較部で抽出された誤対話ボタンが保存される。

(3) 共通誤対話ボタン

被験者の個人差の影響を低減させるためには、複数の被験者の対話履歴のなかから、よく似た誤対話ボタンを見つける必要がある。そこで、誤対話ボタンを被験者間で比較し、互いに似ているものを共通誤対話ボタンとしてグループ化する。ただし、この場合ボタンの類似性は一意に決定することはできず、使いやすさの分析観点により複数の「類似性」があり得る。そこで、この分析観点を4つのカテゴリ(誤対話の発生、誤対話中の操作、誤対話からの復帰、対話の全体的な流れ)に分類した。また、同じカテゴリ内でも程度によって類似誤対話のとらえ方が変わり、そのような観点も問題点の分析に役立つと考えられる。これは、同一カテゴリ内のルール間でマッチング条件の組み合わせを調節することにより、ルールにレベルを設けることで対応した(このようなルール設定については、詳しくは[12]を参照されたい)。

(4) ダイアグラム表示

各データを用いて評価者にダイアグラムやグラフを提示し、問題点の分析を支援する。UI テスタでは、次の4種類のダイアグラム/グラフを提供している。

状態遷移ダイアグラム 装置の状態とユーザ操作をそれぞれノードとアークに対応させることで、ユーザの対話履歴を状態遷移図で表示する。このダイアグラムでは、ユーザ対話の量や複雑さがダイアグラムの大きさや複雑さに直接的に反映される。

対話構造ダイアグラム 基準対話を表すダイアグラム上にユーザ誤対話ボタンをマッピングする。このダイアグラムにより、特定区間のループ、手順の飛び越し/逆戻り、ある状態での無効操作の繰り返しといった、対話の流れを表現できる。

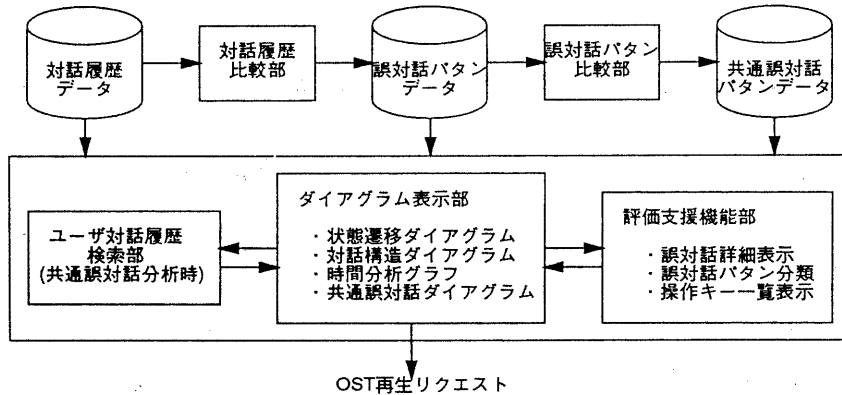


図 4: UI テスタのシステム構成

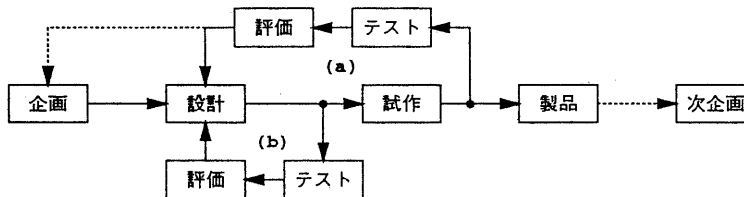


図 5: 反復的設計のサイクル

時間分析グラフ ユーザが個々の操作に要した時間をグラフ形式で表す。このグラフを用いて、次の操作を選択するまでに長い時間がかかっているような、誤操作ボタンに反映されない問題点を抽出できる。

共通誤対話ダイアグラム 基準対話を表すダイアグラム上に、共通誤対話ボタンをマッピングする。このダイアグラムにより、多数の被験者が共通して行っているような誤対話ボタンを、共通性の観点や類似度を様々に変化させながら抽出できる。

これらの表示の際には、ダイアグラム中の各時点における対話履歴収集時のシミュレータ画面を再現し、そのときの状況を把握するため、OST に再生用データを引き渡す。OST はこのデータを受け取り、シミュレーション時の Op パネルウィンドウを再生表示する。

(5) ユーザ対話履歴の検索

共通誤対話ダイアグラムを用いた誤対話ボタンの分析時には、表示されているボタンのなかから一つ

以上を選択し、そのボタンをもつユーザ対話履歴を検索することができる。検索されたユーザ対話履歴は対話構造ダイアグラムで表示される。これにより、評価者は共通誤対話ボタンに関する対話の前後関係を分析し、その誤対話の発生原因を推測できる。

5 OST と UI テスタによる反復的設計

従来、反復的設計のサイクルは図 5 の (a) のような大きなループに従って行われていたが、OST と UI テスタを用いることにより、小さいループ (b) に従って素早く行うことが可能になった。(b) のループの内容は次の通りである。

設計 OST を用いて、UI の設計 (または評価結果のフィードバックをもとにした再設計) を行う。

テスト 開発対象の装置に関する評価用のタスクを設定し、OST のシミュレーション機能を用いて、被験者にこれらのタスクを行わせ、ユーザ対話履歴を収集する。また、各タスクに関する基準対話履歴は、あらかじめ設定された基準の操作手順を、例えば評

評価者が実際に入力して用意しておく。

評価 UI テスタの各種機能 (ダイアグラム表示, 時間分析, 共通誤対話分析) を用いて, ユーザ対話履歴の分析を行い, UI 設計上の問題点を抽出する。

6 ケーススタディ

OST と UI テスタを, 実際の新製品開発に適用したので, その内容をケーススタディとして報告する。

6.1 テスト内容

OST 上で設計した FAX 装置の対話手順を評価するため, シミュレーション機能を利用して対話履歴収集テストを行った。このテストでは, 被験者にタスクの説明書を与え, 3 種類のタスク (時刻修正, 発信元登録, ワンタッチキー登録) を行わせて, それぞれに関する対話履歴を収集した。協力を得た被験者は 18 人であり, テストの 1 人当たりの所要時間は, 約 20 分程度であった。また, 被験者の操作環境を実際の製品の場合に近づけるため, タッチパネル付き端末を利用した。つまり, 被験者のタッチイベントによりシミュレーションが進行するようにした。

6.2 評価結果

本実験の結果, 使いやすさに関する問題点を 9 個抽出した。ここでは, そのうちの 4 例について内容を示す (問題点 1, 2 は共通共通誤対話パターンも示す)。

問題点 1: 漢字入力時の確定方法

「完了」キーが漢字入力の確定に割り当てられているが, 漢字入力以外の時は, 「完了」キーはカレント画面を終了する際に用いられるため, ユーザは「完了」キーによって入力漢字が確定されることがわからず, 次の文字の入力に移ろうとしたり, カーソルを移動させようとして「→」キーを用いたりしている。図 6 はこの問題点を表す共通誤対話パターンであり, () のなかの数字は, この誤対話パターンを含むユーザ対話履歴の数を表している。

問題点 2: かな漢字変換に失敗したとき

変換を試みた文字列が辞書に登録されていなかったとき, 変換が失敗するが, 被験者にはその理由がわからないため, 「変換」キーが受け付けられなかったと勘違いしてしまい, 何度も「変換」キーを押し

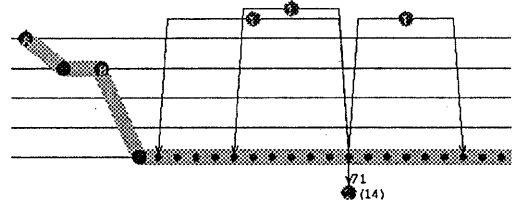


図 6: 問題点 1 を表す共通誤対話パターン

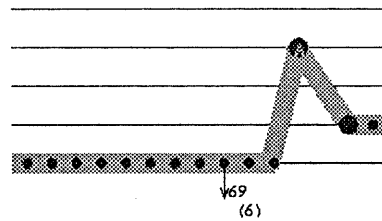


図 7: 問題点 2 を表す共通誤対話パターン

ている。図 7 はこの問題点を表す共通誤対話パターンである。

問題点 3: 電話番号のなかのポーズ (-)

電話番号を「03-945-3214」と入力するとき, 「-」も入力する必要があるのかどうかかわからない。また, 入力しようとしても, 「-」が「ポーズ」キーによって入力されることがわからず, 結局とばしてしまう。

問題点 4: 全角モードと半角モードの移行

半角カタカナ文字の入力モードに移行するには, 先に「カナ」キーでカタカナモードにし, 次に「全/半」キーで半角モードにしなければならないが, ひらがなモードのときに「全/半」キーを押して半角モードに移ろうとしている。

6.3 設計仕様の改善

抽出された問題点のうち 5 個に対し, それらを取り除いて使いやすさを向上させるための設計仕様改善を行った。例として, 上記問題点 1, 2 に対して行われた改善について示す。

問題点 1 に対して 文字入力の際に全角ひらがなを入力した場合のみ, 「変換」キーを表示するように



図 8: 設計改善点 1

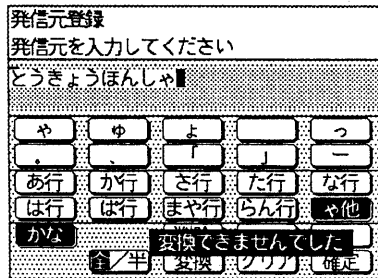


図 9: 設計改善点 2

し、「完了」キーの表示が「確定」キーに変わるようにした(図 8).

問題点 2 に対して かな漢字変換を試みて「変換」キーを押したとき、変換候補の漢字がひとつも登録されていなかった場合は、「交換できませんでした」というメッセージが表示されるようにした(図 9).

この他にも、同様のキーラベル変更やメッセージ追加などの改善を行った。

6.4 設計改善の有効性の検証

行った設計改善の有効性を検証するため、新しい設計データを用いてシミュレーションを行い、対話履歴を収集した。協力を得た被験者は 6 人で、前回のテストには参加していない。例として、問題点 1 に関する再テストの結果を示す。

問題点 1 に対する改善結果

文字列「東京本社」の入力において「変換」キー → 「確定」キーという正しい手順で操作せず、誤対話を行っている被験者が前回のテストでは 18 人中 14 人(約 78%)もいたが、今回は 6 人中 2 人(約

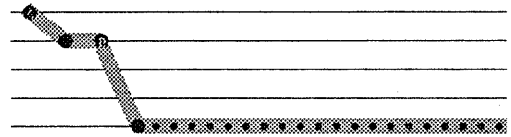


図 10: 再評価実験の結果(問題点 1 について)

33%)に減少した。ダイアグラム上では、前回の実験において抽出されていた共通誤対話パターン(図 6)が出現しなくなった(図 10)。マッチングルールとしきい値の設定 [12] は、図 6 と図 10 で共通であり、ルールは R_1^3 (同始点の誤対話)、しきい値は 70% である。また、誤操作を行った 2 人については、「変換」キー → 「は行」キーの順番で誤操作を行っていたことがわかり、変換中の文字列が次の文字の入力によって自動的に確定されることを期待していたものと推測されたが、確定操作が必要とわかったあとはすぐに「確定」キーを選択できており、前回のよう「完了」キーを選択するまでに様々な無効操作を繰り返すといったトラブルは起こらなかった。

7 まとめ

本稿では、シミュレーション機能を有する FAX 装置用 UI 設計ツール「OST」と、OST を対象とした使いやすさ評価ツール「シミュレータ対応 UI テスタ」について説明した。UI-CAD ツールとしての OST と、CAUT ツールとしての UI テスタによる設計 / 評価の統合的計算機支援環境の構築により、使いやすさを考慮にいたれた製品開発に有効な、UI の反復的設計が効果的に運用可能となった。また、これらのツールを実際の新製品開発に適用した結果をケーススタディとして報告し、これらのツールの有効性を示した。

今後は、これらのツールを用いた製品開発を進めていくとともに、CAUT ツールの対象拡張を目標にしている。近年、グラフィカルユーザインタフェース(Graphical User Interface: GUI)をもつ対話型ソフトウェアアプリケーション(Ap)が広く普及しており、そのユーザビリティ評価のニーズは非常に大きい。しかし、現状の UI テスタをこのような評価に直接適用することは難しく、拡張が必要である。現在、これらの課題を解決する方法を検討中であり、

今後は GUI-Ap を評価する CAUT ツール「GUI テスタ」の開発を中心に進めていく予定である。

謝辞

本研究の機会を提供して下さった関西 C&C 研究所真名垣所長に深く感謝致します。また、プログラム開発に携わって頂いた日本電気技術情報システム(株)大津氏、および快く被験者を引き受けて下さった方々に深く感謝致します。

参考文献

- [1] 稲葉, 今井: 「使いやすさ」の設計手法が実際に, 日経エレクトロニクス, No.529, pp.107-124, 1991.
- [2] 今井: ユーザインタフェースの開発に、「使いやすさの評価」が組み込まれる, 日経エレクトロニクス, No.602, pp.75-91, 1994.
- [3] B. Shneiderman: Designing the User Interface (second edition), Addison-Wesley, 1992. (邦訳) 東, 井関: ユーザーインタフェースの設計, 日経 BP 社, 1993.
- [4] J. Nielsen: Usability Engineering, Academic Press, 1993.
- [5] J. Nielsen, R. Molich: Heuristic Evaluation of User Interface, CHI'90 Conference Proceedings, pp.249-256, 1990.
- [6] C. Lewis, P. Polson, C. Wharton: Testing a Walkthrough Methodology for Theory-Based Design of Walk-up-and-Use Interfaces, CHI'90 Conference Proceedings, pp.235-242, 1990.
- [7] 加藤, 堀江, 小川, 木村: HI 設計チェックリストとそのユーザビリティ評価, 情報処理学会研究報告, Vol.93, No.80, pp.17-24, 1993.
- [8] R. Jeffries, H. Desurvire: Usability Testing vs. Heuristic Evaluation: Was there a contest?, SIGCHI Bulletin, Vol.24, No.4, pp.39-41, 1992.
- [9] 旭, 井関: 使いやすさ評価システム「UI テスタ」の提案, 情報処理学会研究報告, Vol.91, No.76, 1991.

- [10] 旭, 井関: 使いやすさの計測器「UI テスタ」- ユーザ対話履歴の分析 -, 情報処理学会第 44 回全国大会, 1991.
- [11] 旭, 岡田, 井関: 「使いやすさ」の計測: UI テスタ, 第 8 回ヒューマンインタフェースシンポジウム, 1992.
- [12] 岡田, 大津, 旭, 井関: UI テスタにおける共通誤対話の分析, 情報処理学会研究報告, Vol.93, No.80, pp.25-32, 1993.
- [13] S. Card, T. Moran, A. Newell: The Psychology of Human-Computer Interaction, Lawrence Erlbaum Associates, Hillsdale, NJ, 1983.

付録: 対話履歴ファイル例

```
<Title> "時刻修正"  
<MainState> 2  
  
<State> 0008, 0037, 0004  
<Event> A065 (RegistMenuKEY)  
<ScreenClear>  
<CharDisp> 0, 2, "登録メニュー"  
<CharDisp> 0, 26, "登録する項目を選んでください"  
  
— 中略 —  
  
<PanelDisp> 8401, 262,216, 310,216, 0, "完了"  
<LED_on> 0  
<LCD> 13  
<Time> 14:23:31 1993.06.28  
<Interval> 00:00:03  
<State> 0008, 0037, 0009  
  
— 中略 —  
  
<PanelIcon> 8502, 262,194, 0, OnHookKey,  
6,34  
<LED_on> 0  
<LCD> 13  
<Time> 14:24:24 1993.06.28  
<Interval> 00:00:03  
<State> 0008, 0037, 0004  
<End>
```