

UNIVERSITÉ DE NICE - SOPHIA ANTIPOLIS  
**ÉCOLE DOCTORALE STIC**  
SCIENCES ET TECHNOLOGIES DE L'INFORMATION  
ET DE LA COMMUNICATION

# THÈSE

pour obtenir le titre de

**Docteur en Sciences**

de l'Université de Nice - Sophia Antipolis

**Mention : INFORMATIQUE**

Présentée et soutenue par

**André KALAWA**

## **Migration des applications vers les tables interactives par recherche d'équivalences**

Thèse dirigée par Michel RIVEILL

préparée au sein du laboratoire I3S, Equipe RAINBOW

soutenue le 30 août 2013

**Jury :**

<i>Rapporteurs :</i>	Daniel HAGIMONT	- Professeur, INPT/ENSEEIH de Toulouse
	Jacky ESTUBLIER	- Professeur, LIG Grenoble
<i>Directeur :</i>	Michel RIVEILL	- Professeur, Université de Nice
<i>Co-Encadrante :</i>	Audrey RIVEILL	- Docteur, Université de Nice Sophia Antipolis
<i>Président :</i>	Mireille BLAY-FORNARINO	- Professeur, Université de Nice Sophia Antipolis



## Résumé

Dans le domaine du génie logiciel pour les Interactions Homme Machine (IHM), la migration des interfaces utilisateurs (UI) est un moyen pour réutiliser des applications sur des plateformes ayant des modalités d'interactions différentes des environnements de départ. Les approches existantes de migration des UI sont manuelles dans le cadre des approches spécifiques, elles sont automatiques dans le cadre des services d'adaptation des UI aux contextes d'usage, ou elles sont semi automatiques dans le cadre d'une migration flexible dirigée par un concepteur.

Dans cette thèse nous nous intéressons à la migration semi automatique des UI vers une cible comme une table interactive dans l'objectif de transformer des UI Desktop en UI qui favorisent la collaboration et l'utilisation des objets tangibles. Les tables interactives sont des plateformes qui disposent des instruments d'interactions permettant de décrire des UI tangibles et multi-utilisateurs. En considérant que le noyau fonctionnel (NF) des applications de départ peut être réutilisés sur les cibles sans changement, les UI des applications sont caractérisées par la dimension des dialogues entre les utilisateurs et le système, la dimension de la structure et du positionnement des éléments graphiques et la dimension du style des éléments visuels. La migration d'une UI dans ces conditions consiste à transformer ou à recréer les différentes dimensions d'une UI de départ pour la cible tout en considérant les critères de conception des UI pour les tables interactives.

Nous proposons dans cette thèse un modèle d'interactions abstraites pour établir les équivalences entre les dialogues et la structure des UI indépendamment des modalités d'interactions des plateformes source et cible. Les primitives d'interactions et la structure des composants graphiques permettent de décrire des opérateurs d'équivalences pour retrouver et classer les éléments graphiques équivalents en prenant en compte les guidelines des tables interactives. Nous proposons aussi des règles de substitution et de concrétisation pour accroître l'accessibilité des éléments graphiques et favoriser l'utilisation des objets tangibles.

**Mots clés :** migration des interfaces utilisateurs, équivalences des plateformes, modalités d'interactions, critères de conception, guidelines

## Abstract

In software engineering, in the field of human computer interaction (HCI), the migration of user interface (UI) is a way to reuse existing applications on platforms with different interactions modalities. The existing approaches for UI migration can be manual (for specific applications), they can be automatic (for services which adapt UI based on context aware), or they can be mix of the previous - semi automatic (providing a flexible migration process driven by the person in charge).

This thesis proposes a semi automatic process for migration of UI from a desktop to interactive table for the purpose of transforming the UI of desktop to support further collaboration and usage of tangible objects. The interactive tables are platforms with interactions instruments which allow the description of tangible and multi users UIs. Considering that the functional core (FC) of source applications can be reused on target platform without transformation, any UI can be characterized with three dimensions : the first dimension concerns the dialogues between the users and the system, the second dimension concerns the structure and the layout of graphical components, and the third dimension concerns the visual style of graphical elements. In this context, the problematic regarding the UI migration is how to transform or re inject these different dimensions of source UI into the target, while considering the UI design criteria for interactive tables.

This thesis proposes an abstract interactions model for establishing equivalences (independent of modalities of interactions) between the source and the dialogue and structure of the target. The primitives of interaction and the structure of graphical components are used to describe equivalence operators to find and to rank equivalent elements on interactive tables. Furthermore, this thesis proposes substitution and concretization rules to increase the accessibility of graphical elements and to facilitate the usage of tangible objects. The ranking process and the transformation rules are based on guidelines for UI migration to interactive tables which are interpreted form design criteria.

**Keywords :** user interface migration, equivalence of platform, design criteria, guidelines



## **Part I**

# **Introduction**



# Introduction

---

## Contexte

Le nombre grandissant de plateformes et surtout la très grandes variétés de dispositifs d'interactions dont ils disposent telles que les smartphones, les tablettes, les terminaux tactiles ou les tables interactives ont généralisé l'usage des applications ayant des modalités d'interactions [C93] beaucoup plus sophistiquées qu'une simple utilisation du clavier et de la souris. Les tables interactives par exemple offrent la possibilité de mettre en œuvre des applications réellement multi-utilisateurs mêlant interactions tactiles et objets tangibles <sup>1</sup> sur une surface de prêt de 1 mètre carré.

Le domaine du génie logiciel pour les Interactions Homme Machine (IHM) propose plusieurs approches pour la conception d'une application. En particulier la tendance actuelle est de concevoir les Interfaces Utilisateurs (UI) selon les modèles du Framework de Référence Cameleon (CRF) [CCB<sup>+</sup>02] qui identifie différents niveau d'interaction. De la plus abstraite ou indépendante des dispositifs d'interactions disponibles sur la plateforme vers un niveau concret. Ceci permettant la mise en œuvre de l'UI selon la modalité d'interaction souhaitée et disponible sur la plateforme cible. Si l'interface utilisateur d'une application est construite selon cette approche, alors la migration de cette dernière vers des terminaux ayant des modalités différentes est abordé partiellement par [PSO11]. Notre travail aborde la possibilité de faire migrer entre plateformes ayant des modalités d'interactions différentes des applications qui n'ont pas été conçues selon les modèles du framework de référence (CRF).

La migration des UI est une activité de génie logiciel qui implique la transformation des différents aspects qui constituent une UI existante tels que les interactions (ou le dialogue entre l'utilisateur et le système) qu'il faut nécessairement préserver pour que l'utilisateur puisse toujours accomplir les mêmes tâches, les structures (organisations et types de données des éléments graphiques), le positionnement et les styles des composants graphiques qui doivent être adaptés pour être conformes aux spécificités de la plateforme cible et toujours satisfaire les utilisateurs finaux dans les choix de configuration qui ont pu être fait sur la source.

Au delà des problèmes liés à des différences possibles entre les environnements d'exécution des plateformes source et cible [TKB78] qui nécessite le portage du code, les différences des modalités d'interactions entre la source et la cible impliquent nécessairement la prise en compte des critères usuels de conception des UI. La transformation de l'UI de départ doit être guidée non seulement par les critères ergonomiques de conception mais aussi par les dispositifs d'interactions disponibles sur la plateforme cible qu'il peut être intéressant d'utiliser.

Évidemment, en complément de la volonté de rendre disponible des dispositifs d'interactions nouveaux, peut-être plus intuitifs, les critères ergonomiques de conception constituent un ensemble de principes à respecter pendant la mise en œuvre. Ils permettent de garantir l'utilisabilité d'une UI. Par exemple, l'utilisation d'une application pour desktop sur une table interactive sans aucune adaptation

---

<sup>1</sup> Les interfaces utilisateurs tangibles (TUI) permettent à une application de pouvoir interagir avec des objets physiques directement manipulable par les utilisateurs [UI97]. Les objets tangibles sont ceux pouvant être manipulés par une interface utilisateur tangible

pose des problèmes d'utilisabilité, car la simulation du clavier et de la souris n'est pas le meilleur mode d'interaction en terme d'utilisabilité. Si l'application le permet, on peut même imaginer une utilisation multi-utilisateurs avec une nouvelle UI à déduire de l'UI d'origine.

## Enjeux de la migration

Dans le cadre de la migration des applications pour desktops vers les tables interactives par exemple, nous notons que les dialogues, la structure et le positionnement des éléments des UI doivent prendre en compte l'éventualité d'avoir plusieurs utilisateurs mais surtout la possibilité d'utiliser des objets tangibles. Travailler sur ces deux exemples va nous permettre d'avoir une réelle évolutivité des UI entre le terminal source et le terminal cible et permet de favoriser la collaboration dans un espace de travail co-localisé et multi-utilisateurs offert par la table interactive et donc de l'utiliser pleinement.

Nous faisons l'hypothèse que le cœur de l'application peut être migré sans modification et qu'il ne faut agir qu'au niveau de l'UI qu'il faut bien évidemment faire migrer. Cette migration, d'un terminal source vers un terminal cible donnée doit permettre de conserver dans la mesure du possible les dialogues, la structure et les positionnements relatifs des éléments de l'interface, le respect du style des éléments graphiques des UI de la source mais bien évidemment proposer une adaptation pour utiliser au mieux, sans perdre les utilisateurs, les nouveaux moyens d'interaction mis à la disposition des utilisateurs.

La transformation des dialogues d'une UI desktop vers une cible collaborative peut-elle garantir à chaque utilisateur des interactions qui favorisent la collaboration? De nombreux éléments sont à prendre en compte. En effet, chaque dialogue ou message d'erreur ne doit pas perturber les activités des autres utilisateurs; par exemple les ?feedbacks? ou les messages d'erreurs destinés à un utilisateur ne doivent pas bloquer un autre utilisateur si une réponse est attendue. Par ailleurs la transformation des dialogues peut-elle assurer que chaque dialogue reste cohérent avec l'intention et les actions des utilisateurs ? Par exemple la modification d'une valeur par deux utilisateurs distincts peut les perturber. Il faut en effet décider si le résultat est la somme des deux modifications, la moyenne, le min ou le max. Évidemment le contexte et la nature de la valeur peuvent guider sur le choix à effectuer.

La transformation des dialogues sur les UI et les fonctionnalités de l'application source peuvent provoquer de nombreux changements? En effet, la transformation des dialogues des UI desktops pour les tables interactives peuvent par exemple impliquer des modifications pour prendre en compte la présence de plusieurs utilisateurs ou d'objets tangibles.

La transformation de la structure et du positionnement des éléments d'une UI pour desktop vers une table interactive consiste à assurer l'accessibilité aux différents utilisateurs des éléments graphiques pertinents <sup>2</sup>. Par ailleurs la transformation des aspects structurels d'une UI doit elle aussi préserver au mieux l'utilisabilité de l'UI. En effet les solutions de transformations proposées doivent produire des UI conformes aux spécificités de la plateforme cible et satisfaisantes pour les utilisateurs finaux.

Pour terminer, la transformation du style a nécessairement un impact sur la collaboration ou sur l'utilisation des objets tangibles. Généralement, chaque application adopte une charte graphique qu'il faut soit respecter, soit rénover.

Les questions soulevées par les transformations des différents aspects des UI lors d'une migration sont usuellement traitées de différentes manières. En premier lieu, il est possible d'adopter une approche manuelle [WGM08]. Celle-ci permet d'avoir, au prix d'un travail minutieux, une UI conforme aux attentes des utilisateurs finaux car les transformations sont flexibles et donc la qualité de

---

<sup>2</sup>Un menu par exemple



L’UI produite dépend directement des compétences de celui qui effectue la migration manuelle. Mais cette approche est difficilement réutilisable, à moins de la consigner dans un document, que ce soit pour d’autres applications ou pour d’autres personnes car elle nécessite une bonne connaissance des critères de transformation des différents aspects et des technologies des UI des plateformes source et cible.

Il est aussi possible d’adopter une approche automatique basée ou non sur des modèles abstraits [Bes10] [PZ10] qui permet une transformation des différents aspects des UI. Bien que ces approches automatiques soient réutilisables, elles sont nécessairement moins flexibles car le concepteur qui ne peut pas intervenir pendant la transformation, adapte l’UI produite dans un second temps.

Entre ces deux approches, il est aussi possible d’adopter une approche semi automatique [Mar97] qui présente de nombreux inconvénients car non seulement elle induit un travail supplémentaire pour la personne en charge de la migration qui guide la transformation mais en permettant plus de flexibilité, il est assez difficile de capitaliser sur le travail effectué si la personne en charge de la migration change. Néanmoins, cette approche permet une transformation interactive des différents aspects de l’UI. L’intérêt d’une flexibilité dans l’approche de migration d’UI permet d’avoir des UI migrées proches des attentes des utilisateurs finaux.

## Contribution de la thèse

Cette thèse a pour premier objet d’étudier la migration des UI entre plateformes ayant des dispositifs d’interactions différents. Nous avons fait le choix de cibler en particulier la migration des UI depuis une station possédant clavier et souris vers une table interactive. Nous souhaitons que cette migration se fasse au moindre coût pour les concepteurs ou les développeurs tout en prenant en compte les spécificités de la cible et, bien évidemment, les critères ergonomiques usuels utilisés pour la conception des UI.

La solution que nous proposons repose sur un processus semi automatique de migration d’UI. Celui-ci comporte plusieurs étapes interactives pour que les concepteurs ou les développeurs puissent effectuer des choix conformes aux critères de conception qu’ils souhaitent privilégiés. Nous avons fait le choix de l’interactivité, basé sur des choix simples pour garantir simultanément la réutilisabilité, minimiser les coûts mais aussi pour accroître la flexibilité et garantir ainsi l’UI la plus pertinente.

Il est primordial de prendre en compte, lors de la migration des interfaces utilisateurs, les critères ergonomiques de conception. Nos travaux utilisent des concepts issus de plusieurs domaines de recherche.

Dans le domaine de l’utilisabilité, nous nous sommes intéressés aux travaux qui modélisent les critères ergonomiques de conception pour les traduire en règles opérationnelles et utilisables pendant la conception. L’objectif est de pouvoir intégrer ces règles dans la plateforme de migration dans le but de réduire la charge de travail des personnes en charge de la migration.

Dans le domaine de l’ingénierie des modèles, nous nous sommes intéressés aux travaux permettant de modéliser une plateforme dans le but d’effectuer la migration à un niveau abstrait : de concept à concept. L’objectif est de rendre notre travail réutilisable si la source et la cible évoluent. Cette approche nous permet d’abstraire non seulement les UI mais aussi les interactions. Nous proposons ainsi un modèle d’interactions basées sur les primitives d’interactions [KOR<sup>+</sup>11] pour décrire les actions atomiques qui constituent les dialogues entre l’utilisateur et le système.

## Plan du manuscrit

Notre manuscrit est structuré de la manière suivante :

Le chapitre 2 présente l'espace des problèmes liés à la transformations des différents aspects d'une UI. Il délimite le périmètre de la migration des UI d'un poste fixe vers une table interactive. Il fixe nos objectifs.

Le chapitre ?? présente une étude du modèle d'interactions des tables interactives dans le but d'identifier les spécificités et les critères ergonomiques de conception à intégrer pour la migration des UI vers cette cible.

Le chapitre ?? est un état de l'art des approches de migration des UI. Dans ce chapitre nous décrivons les critères nécessaires pour atteindre nos objectifs et nous évaluons les différentes approches à l'aide de ces critères. Ce chapitre se termine par une synthèse des différentes approches présentées.

Le chapitre ?? propose un modèle d'UI qui prend en compte deux aspects des UI : leurs interactions et leur structure. Les objectifs de ce modèle d'UI sont de décrire les UI à migrer indépendamment des plateformes mais aussi de décrire des opérateurs d'équivalences entre les dispositifs d'interactions des plateformes source et cible.

Le chapitre ?? résume les mécanismes de transformation de la structure et des interactions de l'UI source vers la cible. L'objectif de ce chapitre est de décrire la prise en compte effective des critères ergonomiques de conception sous forme de guidelines par les mécanismes de transformation.

Le chapitre ?? présente le prototype que nous avons réalisé. Il constitue une preuve de concept des mécanismes proposés. Plusieurs applications ont été migrées afin de valider les éléments des différents modèles, mettre en évidence le respect des critères ergonomiques et surtout mettre en évidence les bénéfices que l'on peut retirer d'une telle approche.

Le chapitre ?? est une conclusion. Elle rappelle les objectifs que nous souhaitons atteindre, met en évidence nos contributions et leurs apports. Elle donne aussi quelques éléments sur des travaux complémentaires qui pourraient être menés pour parfaire nos travaux.

## **Part II**

# **Domaine d'étude et état de l'art**



# Migration des applications

## Sommaire

<b>2.1</b>	<b>Motivations . . . . .</b>	<b>9</b>
<b>2.2</b>	<b>Problèmes liés à la migration des UI vers une table interactive . . . . .</b>	<b>10</b>
2.2.1	Cas d'une application desktop . . . . .	11
2.2.2	Problèmes liés à la migration du dialogue . . . . .	12
2.2.3	Problèmes liés à la migration de la structure et du positionnement . . . . .	14
2.2.4	Problèmes liés à la migration du style . . . . .	15
2.2.5	Espace des problèmes liés la migration vers les tables interactives . . . . .	16
<b>2.3</b>	<b>Périmètres de la migration des UI vers les tables interactives . . . . .</b>	<b>17</b>
2.3.1	Hypothèse de travail . . . . .	18
2.3.2	Pistes de migration . . . . .	18
<b>2.4</b>	<b>Objectifs de la migration des UI vers les tables interactives . . . . .</b>	<b>20</b>

Dans le but de décrire les différentes étapes et les problèmes liés à la migration des interfaces Hommes-Machines en général, et vers une table interactive en particulier, nous avons pris le parti d'utiliser un exemple "fil rouge". Les motivations du choix de l'étude de cette cible sont décrites dans la section 2.1, puis la section 2.2 décrit pas à pas l'application initialement conçue pour un desktop que nous souhaitons mettre en œuvre une table surface. La section 2.3 présente les objectifs de cette thèse.

## 2.1 Motivations

Les tables interactives comme les desktops et les smartphones sont utilisées dans plusieurs domaines d'activités[Kub11]. Une table interactive peut servir comme plateforme de jeux, pour la consultation de photos, de cartes, de dessins, pour noter les idées lors d'une session de brainstorming, etc. Les tables interactives permettent de mettre en œuvre un espace de travail collaboratif.

Néanmoins, on remarque que les tables interactives ne se sont pas démocratisées comme les tablettes ou les smartphones. Évidemment, le prix de vente en est une des raisons peut constituer un frein à l'achat. Il est par exemple d'environ 6600\$ pour la version 2.0 commercialisées par Microsoft. Même si nous observons une nette baisse du prix par rapport à la première version celui-ci constitue indiscutablement un frein à l'achat. Par ailleurs, la faible diffusion de cette plateforme fait que les applications pour les tables interactives sont loin d'être aussi nombreuses que pour les tablettes ou les smartphones. Selon des données provenant de Distimo<sup>1</sup>, en mars 2012 nous avons noté plus de 350.000 applications développées aux Etats-Unis pour des desktops, 150.000 applications pour les

<sup>1</sup>Distimo : [www.distimo.com](http://www.distimo.com)

smartphones et presque 100.000 spécifiquement pour les tablettes<sup>2</sup> (cf figure 2.1) et aucune pour les tables interactives.

Pour accroître le nombre d'applications disponibles et toucher un éventuel public, nous pensons que la migration des applications existantes peut être une piste sérieuse si la migration utilise effectivement les divers dispositifs d'interactions disponible sur la table. Dans cette thèse, nous étudions par conséquent les divers problèmes posés par le processus de migrations en espérant pouvoir en complément enrichir les applications d'une dimension collaborative et tangible.

Nous présentons donc à la section 2.2 les problèmes liés à la migration d'une application desktop vers une table interactive puis les objectifs de nos travaux. Les hypothèses et les différentes approches permettant de faire migrer des UI sont présentés dans la section 2.3. L'ensemble de ce chapitre nous permet de décrire le périmètre de nos travaux.

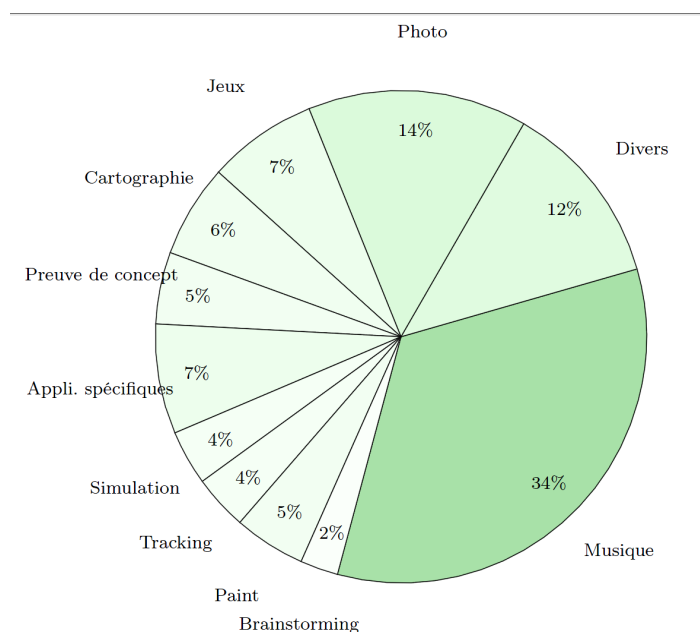


Figure 2.1: Applications disponibles sur les marchés de téléchargement par plateformes en mars 2012 aux États Unis, source Distimo

## 2.2 Problèmes liés à la migration des UI vers une table interactive

Cette section décrit le fonctionnement d'une application permettant l'élaboration de bande dessinée que nous souhaitons faire migrer vers une table interactive. Dans un second temps, nous présentons les problèmes liés à cette migration et les différents objectifs que nous souhaitons atteindre sont décrits à la fin de cette section.

<sup>2</sup>En considérant Windows Phone 7 Marketplace, Google Play, Nokia Ovi Store, Apple Mac App Iphone, Apple App Store - iPad, Apple Mac App Store, Amazon Appstore et BlackBerry App World

### 2.2.1 Cas d'une application desktop

Considérons l'exemple suivant : soit une application conçue pour un desktop qui permet l'élaboration des bandes dessinées (BD) telle que *Comics Book Application* (CBA). Cette application est conçue pour être utilisée par un dessinateur de BD qui est assis devant son écran en utilisant sa souris et son clavier. La fenêtre principale de l'application CBA décrite par la figure 2.2 nous permet d'identifier trois zones majeures que sont la zone des menus correspondant aux composants graphiques situés en haut de la fenêtre principale, l'espace de travail qui contient les cadres d'une page de bande dessinée et la zone de légende correspondant aux groupes, aux formulaires et listes à gauche de la fenêtre.

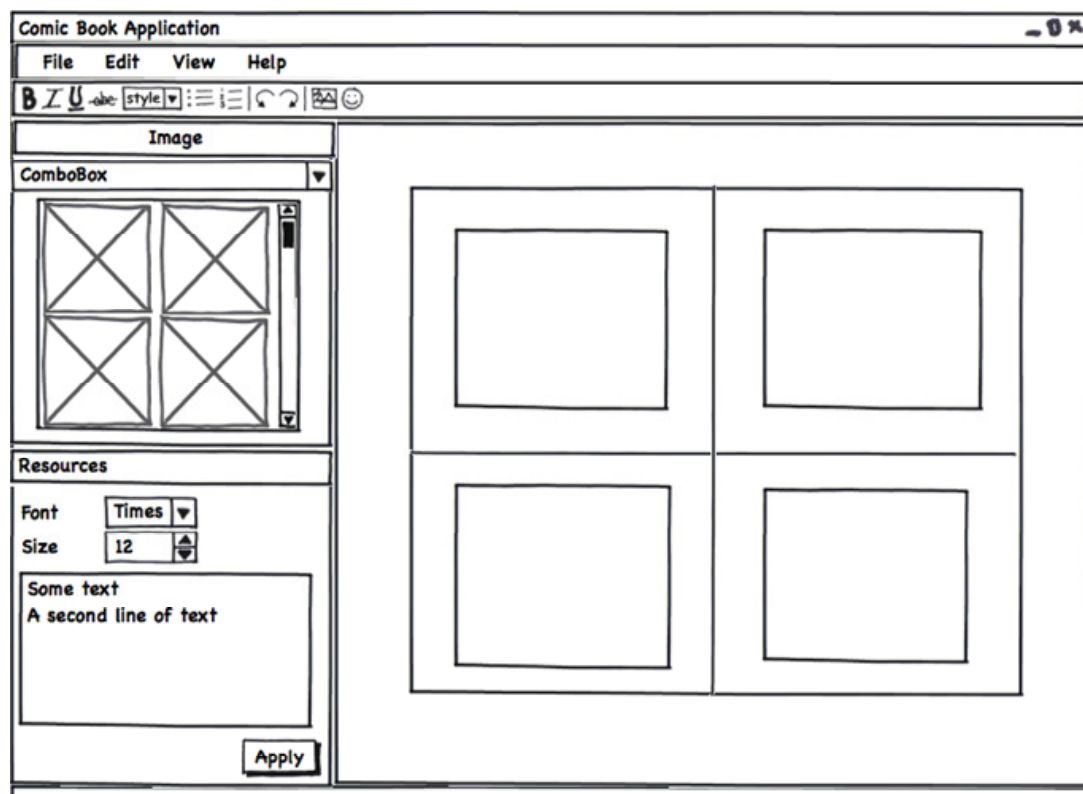


Figure 2.2: Fenêtre principale de l'application CBA

Nous considérons de manière globale que les applications à migrer doivent respecter une décomposition minimale permettant de séparer aisément l'UI et le Noyau Fonctionnel (NF). Si cette hypothèse est vérifiée, les différents aspects des UI des applications peuvent être décrites de manière générale comme suit :

- le **dialogue** entre l'utilisateur et le système permet d'organiser les interactions et les comportements d'une UI graphique,
- la **structure** et le **positionnement** des éléments graphiques qui décrivent l'organisation et les types de données d'une UI,
- le **style** pour décrire les tailles, les couleurs et les polices d'une UI graphique.

Nous considérons ces trois aspects comme autant de dimensions qui permettent de construire l'espace des problèmes lié à la migration d'une UI prévue pour un desktop pour que l'application s'exécute sur une table interactive. Pour chacune de ces dimensions nous étudions les problèmes soulevés par l'utilisation des objets tangibles/physiques comme moyen d'interactions. Dans le cadre des tables interactives, une **interaction tangible** consiste à utiliser des objets physiques (généralement en contact avec l'écran ou la surface d'affichage) comme des dispositifs de manipulation directes ou comme des moyens d'accès à des fonctionnalités. Pour une application "conception de bandes dessinées" (CBA) sur une table interactive, il est envisageable d'utiliser un stylo pour écrire dans les bulles de dialogue d'une bande dessinée. Dans ce cas le stylo est un **objet tangible** de manipulation directe. Par ailleurs, il est également possible que les concepteurs des bandes dessinées souhaitent utiliser un cube comme **objet tangible** dans le but sélectionner des images (représentant des personnages par exemple). Chaque face du cube sera associée à une catégorie de liste d'images ou d'objets graphiques utilisables dans une bande dessinée.

Nous étudions aussi les problèmes soulevés par la transformation d'une UI mono-utilisateur en UI multi-utilisateurs co-localisées. Dans le cadre des tables interactives, les applications peuvent naturellement être utilisées par plusieurs personnes ce qui est plus difficilement imaginable sur un smartphone. Il est alors nécessaire de re-concevoir l'UI de l'application qui est alors destinée à plusieurs personnes (UI multi-utilisateurs) mais qui partagent le même écran (utilisateur co-localisé). L'application "conception de bandes dessinées" que l'on souhaite migrer sur table interactive peut être utilisée simultanément par plusieurs créateurs de bandes dessinées qui ainsi collaborent pour concevoir ensemble une nouvelle BD.

Dans les sections 2.2.2, 2.2.3, et 2.2.4., nous illustrons ces problèmes pour les différents aspects de l'application CBA décrite par la figure 2.2 ci-dessus.

## 2.2.2 Problèmes liés à la migration du dialogue

Le dialogue est une dimension des UI de départ qui varie fortement dans le cadre de la migration vers les tables interactives. En effet le nombre d'utilisateurs et les modalités d'interactions changent entre la source et la cible. Dans ce cadre, la transformation du dialogue de départ soulève des questions par rapport à l'utilisation des objets tangibles, à l'équivalence des interactions et au nombre d'utilisateurs. Cette section expose ces questions.

### 2.2.2.1 Utilisation des objets tangibles comme moyen d'interactions

Dans le cadre de l'application CBA, les dessinateurs de BD peuvent utiliser différents objets (cube, disque, etc.) pour afficher un menu de manière contextuelle dans le but d'effectuer une tâche de changement de couleur ou de police d'un texte. De manière générale, il est indispensable d'identifier dans l'UI départ les composants graphiques ou les fonctionnalités que les utilisateurs utiliseront par des **interactions tangibles**.

Les objets physiques peuvent aussi servir d'instruments de manipulations directes ou pour entrer des données. L'exemple d'un stylo pour sélectionner des options ou pour écrire dans le cadre de l'application CBA est envisageable. Dans ce cas, la migration des dialogues impacte à la fois l'UI et le NF. En effet, les **interactions tangibles** peuvent aussi nécessiter l'ajout dans le NF des traitements inexistants dans l'application de départ et la transformation de l'UI. Par exemple l'écriture à main levée avec un stylo à main levée peut nécessiter l'ajout d'un module de reconnaissance d'écriture si la plateforme cible n'offre pas cette fonctionnalité.



L'utilisation des objets tangibles comme moyen d'interaction pour les UI de la cible soulève un sous-ensemble de la dimension des problèmes liés à l'équivalence des interactions entre la source et la cible.

#### 2.2.2.2 Équivalences des interactions entre la source et de la cible pendant la migration

Pour établir des équivalences entre les plateformes source et cible, il est indispensable d'étudier les instruments qui permettent aux utilisateurs de dialoguer avec les UI sur ces plateformes. En effet les différents instruments d'interactions offrent différentes modalités aux utilisateurs des applications. Dans le cas de la migration de l'application CBA du desktop vers une table interactive, il est indispensable de transposer par les interactions du clavier et de la souris sur les tables par des interactions tactiles ou par un objet tangible.

La transposition des dialogues de la source sur une table interactive impose des transformations des interactions initiaux présentes entre l'UI desktop et l'utilisateur et entre l'UI et le noyau fonctionnel. Dans le but d'assurer la cohérence des dialogues après la migration, il est alors indispensable de s'assurer que les nouveaux dialogues permettent toujours une utilisation de l'application. Pour ce faire, nous pensons que les équivalences d'interactions doivent :

- garantir que l'ensemble des fonctionnalités de l'application desktop restent accessible sur les tables interactives après la migration. En effet, une migration ne doit pas altérer les fonctionnalités de l'application initiale sauf si la personne en charge de la migration le souhaite.
- assurer que les dialogues modifiant les données d'une application prennent en compte les interactions de plusieurs utilisateurs.

#### 2.2.2.3 Transformation du dialogue de la source pour une plateforme multi-utilisateurs et co-localisée

Dans le cadre de la migration, ce problème consiste aussi à transformer une application mono-utilisateur en une application multi-utilisateurs pour les tables interactives au cas où le nombre d'utilisateurs est supérieur à un. La migration dans ce cas impacte à la fois l'UI et le NF.

En ce qui concerne les transformations au niveau de l'UI, comment éviter d'avoir un dialogue gênant pour les autres utilisateurs? Dans le cadre de l'application CBA par exemple les feedbacks, les alertes ou les messages d'erreurs provoqués par un utilisateur ne doivent pas être bloquants pour d'autres. Par ailleurs, comment décrire le couplage entre les utilisateurs et les dialogues? En effet, une tâche<sup>3</sup> ou une activité est exprimée par ensemble de dialogues entre l'utilisateur et le système pour une application desktop. Sur une table interactive, une tâche peut être effectuée par plusieurs utilisateurs de manière concurrente ou en collaboration. Dans ce cas, comment transformer une UI pour prendre en compte les tâches concurrentes ou les tâches effectuées en collaboration? Comment surtout garantir que le noyau fonctionnel gère de manière cohérente la multiplicité des interfaces et les différents utilisateurs.

En ce qui concerne les transformations au niveau du NF, la migration de l'application CBA par exemple implique que les tâches de sauvegardes de fichiers ou de modifications des données puissent être faites par les utilisateurs en préservant la cohérence des données de l'application. Comment transformer le NF des applications mono-utilisateur pour prendre les dialogues **multi-utilisateurs**?

---

<sup>3</sup>Nous considérons les tâches et les activités impliquant les utilisateurs dans cas

Généralement les spécifications conceptuelles des applications à migrer ne sont pas disponibles. Il est indispensable de retrouver la description des tâches d'une application à partir de son code source par exemple pour transformer les dialogues. Comment retrouver et représenter la description des dialogues d'une application afin de les transformer ? Dans notre cadre de migration, si l'extraction des tâches est difficile ou impossible à partir du code source de l'application à migrer, est-il possible de transformer les dialogues sans description des tâches ? Quels sont les limites d'une transformation des dialogues non basée sur les tâches ?

### **2.2.3 Problèmes liés à la migration de la structure et du positionnement**

Nous pensons que la migration des UI vers les tables interactives soulève des problèmes liés à la transformation de sa structure et du positionnement de ses éléments. En effet, la structure et le positionnement sont des caractéristiques des UI qui se déclinent de différentes manières suivant les plateformes et les utilisateurs. La transformation de la structure et du positionnement dans le cadre des tables interactives a pour objectif de favoriser l'accessibilité des éléments graphiques et de façon globale à l'utilisabilité des UI migrées.

La transformation de la structure et du positionnement des UI de départ soulève des questions sur les équivalences entre les éléments graphiques qui composent l'UI source et ceux proposés par la cible, le regroupement et le positionnement des éléments graphiques pour favoriser la collaboration.

#### **2.2.3.1 Équivalences entre les éléments graphiques**

Dans le cadre de l'application CBA par exemple, les menus ou les listes d'éléments peuvent être présentés sur les tables interactives dans le but d'avoir des interactions directes et intuitives. Les éléments graphiques proposés par les tables interactives permettent d'atteindre ce but. Par exemple, une liste contenant des items textuels peut être remplacée par une liste d'éléments d'images associées aux items sur la table interactive. L'ensemble des éléments de la liste textuelles doit être transposé sur la table interactive dans une liste équivalente.

De manière globale, les équivalences entre les éléments graphiques de la source et de la cible doivent préserver les données qu'ils contiennent. La multitude d'éléments graphiques et des plateformes implique de mettre en place une solution équivalente entre les éléments graphiques automatisable.

#### **2.2.3.2 Favoriser la collaboration par un regroupement et un positionnement adéquat des éléments graphiques**

La fenêtre principale de l'application CBA est conçue en respectant la métaphore du bureau qui permet de décrire les applications pour les ordinateurs personnels [App95]. Les menus sont placés en haut à des positions fixes, les outils ou les légendes sont toujours placés à gauche ou à droite et la zone de travail au centre. Cette structuration en zones permet aux utilisateurs des desktops de développer des réflexes quelque soit l'application dans une approche similaire.

Sur une table interactive, cette structuration n'est pas recommandée car l'accessibilité des composants graphiques dépend directement de la position de l'utilisateur autour de la table : le haut pour un utilisateur peut être le bas d'un autre. La table interactive ne possède pas d'axe bas-haut immédiat et il est donc nécessaire de repenser la disposition de différentes fenêtre. Par exemple, pour l'application CBA, les différentes zones de la structure de départ doivent pouvoir être conservées mais chaque zone n'est plus associée à un espace géographique spécifique de l'écran. Sur la droite de la figure 2.3, les

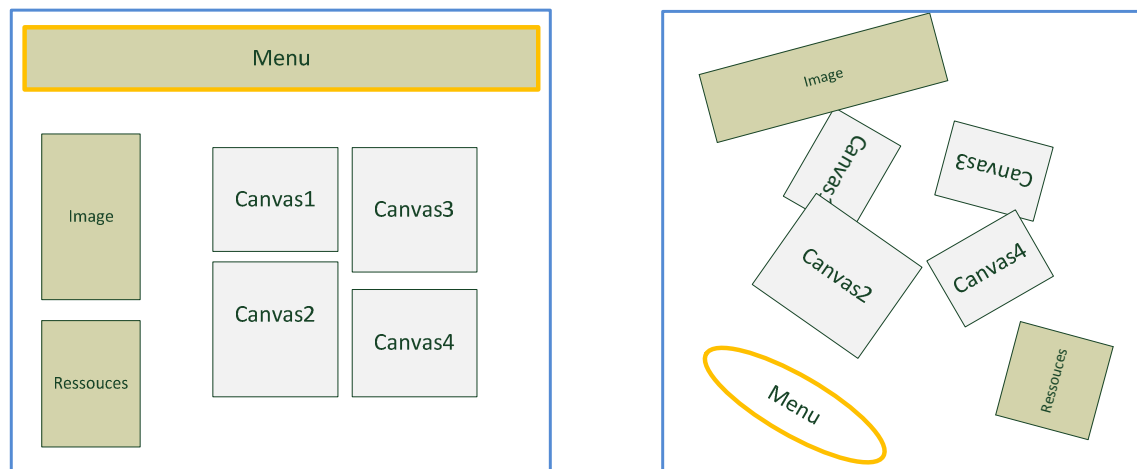


Figure 2.3: Migration de la structure d'une UI Desktop sur une table interactive

différentes zones de l'UI CBA de départ n'ont plus de position fixe sur une table interactive. On peut même imaginer dupliquer certaines zones pour qu'elle deviennent accessibles aux utilisateurs.

Les transformations de la structure et du positionnement de l'UI a pour objectif de favoriser l'**accessibilité**. Pour ce faire, il est important de déterminer les **groupes d'éléments graphiques** à transformer.

Dans le but de concevoir des UI utilisables, l'ingénierie des IHM préconise de prendre en compte des critères ergonomiques (ou des heuristiques) pendant la conception des UI [Sca86, NM90]. Nous considérons dans cette thèse que le respect des critères ergonomiques de conception pendant la migration permet de favoriser l'utilisabilité des UI obtenues. Dans ce cadre, il reste à définir comment on peut prendre en compte ces critères ergonomiques pendant la migration des UI. Il est donc important de les identifier et de les intégrer aux processus de migration pour pouvoir ultérieurement affiner les propositions d'UI et émettre des recommandations utilisables par la personne en charge de la migration.

### Remarque

La migration de la structure et du positionnement sont des transformations uniquement au niveau des UI des applications à migrer.

#### 2.2.4 Problèmes liés à la migration du style

Dans notre cadre, la migration du style consiste à modifier l'aspect visuel des différents éléments graphiques de l'UI d'arrivée. Nous avons identifié trois caractéristiques des éléments graphiques pour décrire la modification du style.

- La **taille des éléments graphiques** est importante sur une table interactive car les dimensions des composants graphiques peuvent favoriser ou non un travail commun. Selon ce que l'on souhaite : véritable travail en commun ou chacun doit être capable de visualiser immédiatement l'activité d'un autre utilisateur ou au contraire, un travail plus individuel ou l'activité de l'un ne doit pas perturber l'activité de l'autre ; il sera nécessaire d'ajuster la taille des composants

graphiques. En effet des composants graphiques trop larges peuvent gêner les autres utilisateurs et au contraire, des composants trop petit peuvent rendre difficile le travail commun.

- La **couleur et la police des textes** sont des propriétés subjectives car elles peuvent ne pas être acceptées par tout le monde. Cependant dans le cadre de la migration nous pensons que leur choix doivent être cohérents entre les éléments graphiques d'une UI. La cohérence du style consiste à s'assurer que les éléments graphiques de même type doivent avoir des couleurs ou des polices ?compatibles? par exemple (respect d'une charte graphique).

Les tables interactives ont une surface d'affichage plus grande qu'un desktop ou une tablette. La taille des éléments d'une UI doit nécessairement être adaptée en fonction de cette surface mais aussi en fonction du nombre d'utilisateurs prévus afin de faciliter la participation de tous.

#### 2.2.4.1 Cohérence globale du style des UI migrées

Il est important que les couleurs, les polices, les icônes, les images ou la taille des éléments graphiques soient définies suivant des recommandations pour assurer une cohérence globale. L'homogénéité de caractéristiques <sup>4</sup> pour chaque type d'éléments graphiques constitue la cohérence globale à un style. Il ne s'agit pas d'un problème spécifique aux tables interactives mais la taille de l'écran disponible pose d'une manière nouvelle le respect des chartes graphique et des recommandations de style.

#### 2.2.5 Espace des problèmes liés la migration vers les tables interactives

Les problèmes liés à la migration que nous avons étudiés à la section 2.1 peuvent être représentés à l'aide d'un espace à trois dimensions décrit par la figure 2.4.

La dimension **D1** correspond aux problèmes de migration des dialogues. Elle consiste à transformer l'UI et porter le NF des applications. Il est indispensable d'avoir des processus génériques pour prendre en compte plusieurs applications et réduire les coûts de la migration. Elle regroupe trois sous-dimensions.

- La dimension **D11** correspondant aux problèmes liés à la transformation des dialogues par ajout des **interactions tangibles** aux applications de départ.
- La dimension **D12** correspond aux problèmes d'**équivalence des interactions** des UI migrées.
- La dimension **D13** correspondant aux problèmes liés à la transformation des dialogues **mono-utilisateur** en **multi-utilisateurs co-localisés**.

Ensuite la dimension **D2** correspond aux problèmes de la transformation de la structure et du positionnement des éléments graphiques. Elle consiste à transformer l'UI de l'application source.

Dans ce cas, il est important d'avoir un processus qui prend en compte les critères ergonomiques de conception afin de garantir l'utilisabilité. La genericité du processus de transformation est importante. Elle regroupe deux sous-dimensions.

- La dimension **D21** correspondant aux problèmes liés à l'**équivalence structurelle** des éléments graphiques de la source et de la cible.

---

<sup>4</sup>Les couleurs, les polices, les icônes, les images ou la taille des éléments graphiques

- La dimension **D22** correspondant aux problèmes de **regroupement des éléments graphiques** des UI migrées.
- La dimension **D23** correspondant aux problèmes de **positionnement** des éléments graphiques des UI migrées.

Enfin la dimension **D3** correspond aux problèmes liés à la transformation du style de l'UI départ pour favoriser leur accessibilité. Dans le but de réduire la charge de travail des concepteurs et d'assurer l'homogénéité, il est important d'utiliser des styles génériques. Cependant la flexibilité de la migration du style permet aussi de personnaliser les UI cibles. Elle regroupe deux sous-ensembles de problèmes :

- La dimension **D31** correspondant aux problèmes liés à la **taille** des éléments graphiques sur un espace de travail partagé.
- La dimension **D32** correspondant aux problèmes liés à la **police des textes** des UI migrées.
- La dimension **D33** correspondant aux problèmes liés à la **couleurs des éléments graphiques** des UI migrées.

## 2.3 Périmètres de la migration des UI vers les tables interactives

Dans notre contexte, les applications à migrer sont des systèmes interactifs (SI) conçus en respectant un modèle d'architecture et identifiant un noyau fonctionnel (NF) et des interfaces utilisateurs (UI). Les modèles d'architecture sont des patrons de conception logiciel, ils préconisent des stratégies de répartition des services qui se traduisent par un ensemble de constituants logiciels. Il existe plusieurs modèles d'architecture qui permettent la séparation UI-NF.

- Le modèle d'architecture de ARCH [UIM92] par exemple, permet une séparation entre le NF, le Contrôleur de Dialogue (CD) et la Présentation (P). Elle permet aussi la migration des composants de l'UI (CD et P) sans modification des composants du NF.
- Le modèle d'architecture PAC-Amodeus[Nig94] est basé sur ARCH, il permet de définir plusieurs contrôleurs de dialogue pour une application grâce aux agents PAC[Nig94].
- Le modèle d'architecture MVC[KP88] permet la conception des SI réutilisables. Il divise les applications en trois types de composants : le modèle (M), la vue (V) et le contrôleur (C). Le modèle est une représentation du domaine d'une application, il peut contenir des données, des services, etc. et il fait partie du NF d'une application. La vue est la structure de l'UI d'une application, elle est constituée des éléments d'une bibliothèque graphique. Le contrôleur est une interface entre le modèle, la vue et les dispositifs d'interactions en entrée. Cette architecture permet une migration de la vue sans modification du modèle et du contrôleur.

Nous présentons à la section2.3.1 notre hypothèse de travail. La section2.3.2 présente les pistes de migration des UI et la section2.4 les objectifs de notre travail.

### 2.3.1 Hypothèse de travail

Les applications à migrer ont une décomposition fonctionnelle minimale qui comprend une interface utilisateur (UI) et un noyau fonctionnel (NF). Cette décomposition identifie clairement la partie qu'il faut migrer (l'UI) et la partie qui sera réutilisée sur la plateforme cible (le NF). Dans cette thèse, pour ne pas aborder simultanément tous les problèmes de migration et parce que ceux-ci ont été abondamment étudiés par ailleurs, nous considérons que nous pouvons réutiliser le NF de l'application source sur la plateforme cible sans modifier son code. Il offre donc les mêmes fonctionnalités. Il est évident que si ce NF devrait lui-même être migré, quel qu'en soit les raisons, cela doit être effectivement traité, éventuellement par les propositions décrites dans [TKB78] [BC10] du NF par exemple.

Les deux hypothèses simplificatrices que nous faisons sont réalistes. La plupart des applications interactives adoptent aujourd'hui une architecture proche du modèle MVC très largement répandu permettant d'identifier très clairement la partie UI et la partie NF de l'application. Par ailleurs, la table surface que nous avons utilisé - une des rares tables interactives disponibles - utilisent le même système d'exploitation que celui utilisé par la très grande majorité des desktops rendant inutile le portable du NF. Si celui-ci s'avérait nécessaire, il serait alors possible d'utiliser une solution à base de machine virtuelle.

Il est évident que notre approche conserve les dialogues existants entre le NF et l'UI et que nous n'avons que très peu abordé les questions relatives à l'évolution de ceux-ci lors de la migration.

### 2.3.2 Pistes de migration

A partir des hypothèses précédemment exposées, la migration des UI des applications existantes vers une nouvelle plateforme tout en conservant le NF peut se faire de plusieurs manières :

- Une **nouvelle conception de l'UI** pour la plateforme cible sans tenir compte de l'UI source.
- Une **déduction de l'UI** à partir des spécifications du NF de départ.
- Une **adaptation de l'UI** de départ par rapport à la plateforme cible.

#### 2.3.2.1 Nouvelle conception de l'UI

Cette approche consiste à concevoir une nouvelle UI pour la plateforme cible sans nécessairement tenir compte de l'UI de départ. Cette nouvelle conception permet de prendre en compte les spécificités de la plateforme d'arrivée et d'intégrer ses critères de conception des UI (chaque plateforme est livrée avec un guide du bon usage). Les approches de conception des UI telles que DIANE+ [Bar88] ou MUSE[LL09] et les approches de conception basées sur des modèles [YFS14] peuvent être utilisées pour concevoir la nouvelle UI. DIANE+ et MUSE permettent d'inclure dans la conception de l'UI des critères ergonomiques [Van97].

L'avantage d'une nouvelle conception est de construire une UI sur mesure, prenant en compte l'ensemble des nouveaux besoins liés à la plateforme cible. Cependant le temps pris pour la conception de cette nouvelle UI comprends nécessairement le temps pris pour la transformation des dialogues de l'UI source et les temps nécessaire à la conception de la structure, du positionnement et du style de la nouvelle UI. Ces différents temps constituent des coûts de mise en œuvre non négligeables. Cette piste de solution, ne capitalisant pas sur le travail déjà effectué, demandant des compétences métiers fortes et un temps non négligeable de mise en œuvre, ne nous permet pas de **réduire les coûts de mise en œuvre** des applications pour les tables interactives.

### 2.3.2.2 Dédution de l'UI à partir du NF

La deuxième piste de migration préconise de s'appuyer sur le NF de l'application de départ, particulièrement sur les liens entre ce NF et l'UI. Il est en effet possible de déduire une UI en se basant sur la représentation abstraite des liens entre le NF et l'UI. Il existe des travaux qui préconisent la génération des UI à partir des descriptions de services Web par exemple [KKM03]. Par ailleurs l'approche ALIAS [JOF11] propose des modèles pour une représentation abstraite du NF et des liens entre NF et UI qui peuvent être utilisés pour déduire les UI à partir des NF. La déduction des UI en se basant sur les NF implique la déduction à partir du NF des éléments suivants :

- les composants graphiques qui composent l'UI,
- la structure et le positionnement (layout) de ses composants graphiques,
- les dialogues et les enchaînement des fenêtres et des activités de l'UI.

Les travaux sur la génération d'une UI à partir du NF [TC02] montrent qu'en plus du NF, il est indispensable d'avoir une spécification des tâches sous forme d'un contrôleur de dialogue et d'adaptateur du NF (dans une architecture ARCH). Le NF ne permet usuellement pas de déduire la structure et les différents types de regroupements des composants graphiques (fenêtres, panels, etc.). Les UI obtenues par une déduction à partir du NF sont **utilisables** à condition d'avoir une description des tâches pour permettre l'organisation des dialogues et de sa structure.

### 2.3.2.3 Adaptation de l'UI par recherche d'équivalences

La troisième piste de migration est celle qui préconise de s'appuyer sur les spécifications de l'UI à migrer et de les adapter à la plateforme d'arrivée. Ces spécifications sont décrites par des modèles abstraits. Les modèles servent à décrire les différentes dimensions des UI tels que les dialogues décrits par l'UI, le placement ou le layout des éléments de l'UI mais aussi les styles de présentations des caractéristiques visuelles (tailles et couleurs des textes, etc.). Les différents modèles abstraits des UI à migrer peuvent être retrouvés par abstraction à partir du code source par exemple [BPS08, BV02, GPF10, TS99].

L'adaptation des UI à migrer pour les tables interactives peut se faire en transformant partiellement ou en totalité les trois dimensions identifiées.

1. L'adaptation des dialogues consiste à produire des dialogues équivalents à partir de l'UI de départ et en prenant en compte les caractéristiques de la cible : introduction d'interactions tangibles et présence potentielle de plusieurs utilisateurs.
2. L'adaptation de la structure consiste à produire pour la plateforme cible une structure équivalente de l'UI de départ en prenant en compte les problèmes liés au regroupement et au positionnement des composants graphiques sur les tables interactives pour lesquels l'axe haut-bas dépend de la position de l'utilisateur. Selon les cas, la structure et le positionnement de départ peuvent être ignorés ou réutilisés partiellement pendant la migration. Dans le cas où ils sont ignorés, il est indispensable de proposer des outils pour les réintroduire facilement.
3. En ce qui concerne le style, il est indispensable de le réintroduire pour la cible, en respectant la charte graphique et en favorisant l'accessibilité des éléments graphiques. En effet, la taille, la couleur ou la police des textes des tables interactives sont différentes de celles des desktops. Le style de la cible peut se baser sur un modèle de style lié aux tables interactives.

L'introduction manuelle du positionnement et du style permet d'avoir une approche de migration des UI plus de flexibilité mais avec un coût de mise en ?uvre non négligeable. Au contraire, une adaptation automatique du positionnement et du style présente un coût de migration faible mais au prix d'une moindre flexibilité et un faible respect des critères ergonomiques de la cible.

## 2.4 Objectifs de la migration des UI vers les tables interactives

Nous avons fait le choix dans cette thèse d'explorer la troisième piste de migration (cf. section 2.3) et d'appliquer nos travaux aux tables interactives qui sont des plateformes innovantes provoquant un changement assez important dans la conception des UI. En effet, il ne s'agit pas uniquement de trouver les composants similaires entre les toolkits de la plateforme source et de la plateforme cible mais d'introduire au sein de l'UI de nouveaux dispositifs d'interactions par l'utilisation d'interfaces tangibles et de prendre en compte la réelle possibilité qu'ont des utilisateurs de pouvoir enfin partager de manière intuitive, pour peu que l'UI soit bien construite, une même application. La migration d'une application existante sans avoir à reconcevoir l'UI est à même d'enrichir le catalogue d'application disponible sur les tables interactives par un abaissement important du coût de mise en ?uvre de ces applications.

Ces arguments nous ont amenés à nous poser deux questions :

- d'une part, nous devons adapter les différentes dimensions d'une UI fondamentalement conçue pour une personne en une UI collaborative et tangible,
- et d'autre part, il est nécessaire d'inclure dans le processus de migration la prise en compte de critères ergonomiques de conception [Sca86, NM90], liés à la plateforme cible pour réduire effectivement le coût de la migration tout en concevant une UI réellement utilisable.

La réponse à ces deux questions doit nous permettre d'atteindre les objectifs que nous nous sommes fixés dans le cadre de cette thèse. Pour ce faire, la deuxième partie du document présente les spécificités des tables interactives qui constituent notre domaine d'étude pour identifier les critères de conception. Nous considérons que ces critères sont les critères ergonomiques de conception qui doivent être affinés en recommandations lors de la migration des UI vers les tables interactives.



# Tables interactives et Migration des UI

---



# **Part III**

## **Contributions**



# Approches de migration des UI

---



# Modélisation des UI

---





# Mécanismes de migration des UI

---



# **Validations du processus**

---



## **Part IV**

# **Conclusion et Perspectives**



# Conclusions et perspectives

---





# Bibliography

- [App95] Apple Computer Inc. *Macintosh Human Interface Guidelines*. Addison-wesley Publishing, 1995.
- [Bar88] Marie F Barthet. Logiciels interactifs et ergonomie. pages 101–116, 1988.
- [BC10] Thiago Tonelli Bartolomei and Krzysztof Czarnecki. Swing to swt and back: patterns for api migration by wrapping, in. In *Proc. ICSM*, 2010.
- [Bes10] Guillaume Besacier. *Interactions post-WIMP et applications existantes sur une table interactive*. PhD thesis, UNIVERSITÉ PARIS-SUD 11, 2010.
- [BPS08] Renata Bandelloni, Fabio Paternò, and Carmen Santoro. Reverse engineering cross-modal user interfaces for ubiquitous environments. In Jan Gulliksen, Morton Borup Harning, Philippe Palanque, Gerrit C. van der Veer, and Janet Wesson, editors, *Engineering Interactive Systems*, pages 285–302, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
- [BV02] L. Bouillon and J. Vanderdonckt. Retargeting web pages to other computing platforms with vaquita. In *Ninth Working Conference on Reverse Engineering, 2002. Proceedings.*, pages 339–348, Nov 2002.
- [C93] Laurence " and Joëlle Coutaz. A design space for multimodal systems. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '93*, pages 172–178, New York, New York, USA, May 1993. ACM Press.
- [CCB<sup>+</sup>02] Gaëlle Calvary, Joëlle Coutaz, Laurent Bouillon, Murielle Florins, Quentin Limbourg, L. Marucci, Fabio Paternò, Carmen Santoro, N. Souchon, David Thevenin, and Jean Vanderdonckt. CAMELEON Project. Technical report, 2002.
- [GPF10] A. M. P. Grilo, A. C. R. Paiva, and J. P. Faria. Reverse engineering of gui models for testing. In *5th Iberian Conference on Information Systems and Technologies*, pages 1–6, June 2010.
- [JOF11] Cédric JOFFROY. *COMPOSITION D'APPLICATIONS ET DE LEURS INTERFACES HOMME-MACHINE DIRIGÉE PAR LA COMPOSITION FONCTIONNELLE*. PhD thesis, UNIVERSITÉ DE NICE“SOPHIA ANTIPOLIS”UFR Sciences, 2011.
- [KKM03] M. Kassoff, D. Kato, and W. Mohsin. Creating GUIs for web services. *IEEE Internet Computing*, 7(5):66–73, September 2003.
- [KOR<sup>+</sup>11] André Kalawa, Audrey Occello, Michel Riveill, Nice Sophia, and Antipolis Cnrs. *SUIT : a framework for Supporting UI Translation*. 2011.
- [KP88] Glenn E Krasner and Stephen T Pope. A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 System. Technical report, 1988.
- [Kub11] SÃbastien Kubicki. *Contribution à la prise en considération du contexte dans la conception de tables interactives sous l'angle de l'IHM , application à des contextes impliquant table interactive RFID et objets tangibles*. PhD thesis, UNIVERSITÉ DE VALENCIENNES, 2011.

- [LL09] Kee Yong Lim and John B. Long. *The Muse Method for Usability Engineering*. Cambridge University Press, New York, NY, USA, 2009.
- [Mar97] Panagiotis Markopoulos. A compositional model for the formal specification of user interface software. *Doctor*, (March), 1997.
- [Nig94] Laurence Nigay. *Conception et modélisation logicielles des systèmes interactifs : application aux interfaces multimodales*. PhD thesis, UNIVERSITÉ JOSEPH FOURIER - GRENOBLE 1, 1994.
- [NM90] Jakob Nielsen and Rolf Molich. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '90, pages 249–256, New York, NY, USA, 1990. ACM.
- [PSO11] Fabio Paternò, Carmen Santoro, and Rasmus Olsen. Migratory Interactive Applications for Ubiquitous Environments. pages 9–23, 2011.
- [PZ10] Fabio Paternò and Giuseppe Zichittella. Desktop-to-Mobile Web Adaptation through Customizable Two-Dimensional Semantic Redesign. pages 79–94, 2010.
- [Sca86] Dominique L. Scapin. Guide ergonomique de conception des interfaces homme-machine. Technical Report RT-0077, INRIA, October 1986.
- [TC02] David Thevenin and Joëlle Coutaz. Adaptation des IHM : Taxonomies et Archi. Logicielle. In *IHM 2002*, pages 207–210, 2002.
- [TKB78] Andrew S. Tanenbaum, Paul Klint, and A. P. Wim Böhm. Guidelines for software portability. *Softw., Pract. Exper.*, 8(6):681–698, 1978.
- [TS99] K. Tucker and R. E. K. Stirewalt. Model based user-interface reengineering. In *Sixth Working Conference on Reverse Engineering (Cat. No.PR00303)*, pages 56–63, Oct 1999.
- [UI97] Brygg Ullmer and Hiroshi Ishii. The metaDESK : Models and Prototypes for Tangible User Interfaces. In *UIST '97 Proceedings of the 10th annual ACM symposium on User interface software and technology*, pages 223 – 232, 1997.
- [UIM92] UIMS. A METAMODEL FOR THE RUNTIME ARCHITECTURE OF AN INTERACTIVE SYSTEM The UIMS Tool Developers Workshop \*. In *SIGCHI Bulletin*, number January, 1992.
- [Van97] Jean Vanderdonckt. Conception assistée de la présentation d’une interface homme-machine ergonomique pour une application de gestion hautement interactive. 1997.
- [WGM08] Xin Wang, Yaser Ghanam, and Frank Maurer. From desktop to tabletop: Migrating the user interface of agileplanner. In Peter Forbrig and Fabio Paternò, editors, *Engineering Interactive Systems*, pages 263–270, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
- [YFS14] Enes Yigitbas, Holger Fischer, and Stefan Sauer. Model-based user interface development for adaptive self-service systems. In Aaron Marcus, editor, *Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience*, pages 206–213, Cham, 2014. Springer International Publishing.