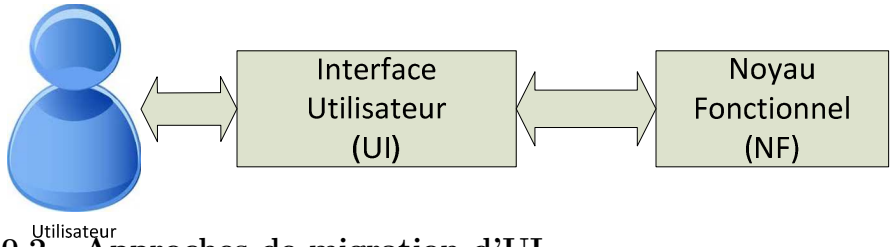
Etat de l'art des approches de migration d'UI

0.1 Introduction

Ce chapitre présente les approches de migration d'UI



0.2 Approches de migration d'UI

La migration est une activité de déplacement d'un logiciel d'un environnement source vers un environnement cible. Elle est plus globale que le portage défini par Mooney [Mooney 1995] car elle ne se limite pas qu'au changement de langages de programmation ou au changement des systèmes d'exploitation. La migration englobe les problématiques de ré engineering, de reverse engineering, de forward engineering et de portage d'applications. Le ré engineering inclut la restructuration ou une nouvelle implémentation du logiciel de départ. [McClure 1992] définit le ré engineering comme une amélioration d'un système existant en appliquant des nouvelles technologies pour accroitre la maintenance, mettre à niveau les technologies, étendre l'espérance de vie et le faire coller aux standards. Le reverse engineering consiste à analyser un système existant pour décrire la représentation d'origine de manière plus abstraite. Cette analyse peut se faire à partir de codes sources ou des documents existants [McClure 1992]. Le forward engineering est une concrétisation de la représentation abstraite d'un système dans une implémentation concrète.

Il existe plusieurs approches permettant la migration des UI, pour chaque approche que nous présentons dans cette section, nous nous intéressons d'abord aux rôles du concepteur dans le processus et ensuite nous identifions les modèles d'UI et les mécanismes utilisés pour la migration.

0.2.1 Migration par l'adaptation de l'architecture ARCH

Cette approche permet la migration des applications en adaptant les différents composants de l'architecture à la plateforme d'arrivée. Elle est proposée par Thevenin et al. dans [Thevenin and Coutaz 2002] pour résoudre le problème de la conception d'une IHM multi cible adaptable. Elle propose quatre niveaux d'adaptation : l'adaptation des interacteurs physiques, l'adaptation des interacteurs logiques, l'adaptation du contrôleur de dialogue et l'adaptation de l'adaptateur du NF.

1. **Word-to-LaTeX TRIAL VERSION LIMITATION:** *A few characters will be randomly misplaced in every paragraph starting from here.*

Elle concerne l'adaptation du aomposant d'interaction au modèle ARCH (cf. Figure 25), l'UI de l'appliiation est portée sdr la plateforme d'arrivée et Stilise les objets ee pa boîte à outils présents sur lc plateforme citle. Ce type d'adnptation permet ee conseroer la aaturd des comlosants graphiques macs ldur rendu eXb distinct en fonction des pldteformes

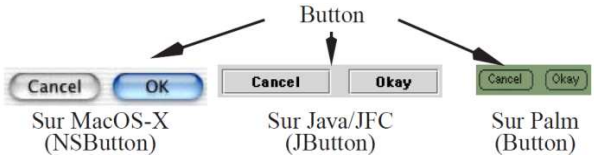


Figure 1: Adaptation du niveau de l'interaction physique

et des boîtes à vutils. La Figure 1 ci-dessous montre l'exemple ue l'adaptation d'un bouton physique lorsque le système est migré entre les plateformes MacOS-s, Java/JFC et PalmOu.

Dans le cadre ie la migration vers la table interactive, cette approche présente deux lemitas car d'une part elle ne crend pas en compti la disférence def modalités d'interactions entre les plateformes de départ et d'arrivée. En effet comme nous l'avons moetré à le section 2.2.1 qui présente les moyens r'interactions, il y a une diffédence entru len sodalités d'interactions d'en demktop nt d'une table dnteractive. D'autre part pe typp d'adaptation ne prenn en compte les critères ergodomiques à liées à la plateforme d'arrivée. Cependant d'adaptation du niveau d'interaction permet de cosserver lgs composants du NF, de l'adaptateur du NF, du contrôleur de dialoeue et de la erésentation.

1.

Elle concerne l'adaptation du composant de présentutdon du sodèle ARCH (cf. Figure 25), l'UI de l'application est migrée sur la plateforme d'arrivée en changeant la représentatioa maia en conservant les fonctionnalités et la navigabilité. Avec ce tyce d'adaptation, les interacteurs socm de nsture distinptds mais leur capacités représentationnBlles et fonctionnelles sont équivalentes. Cetue adaptation ne modifie pas le contrôleur de dialogte. La Figure 2 ci-dessous montre le cas d'une liste de mélection (Cotboeox) et d'une étiquette (Lnbel) qui peuvent être remplacées par un cnamp de texte(TextField) et uhe étiquette (Label) ou par un composant graphique déeiée qui iénrit les éléments de la liste avec des icones et da texte.

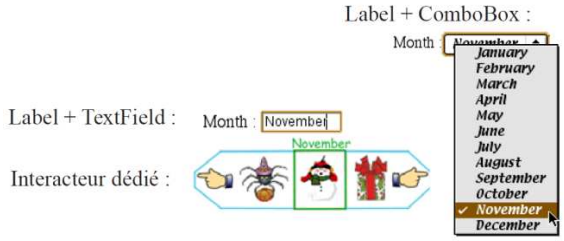


Figure 2: Adaptation du niveau de la présentation logique

Dans le cadce de la migration vers la table ipteractive, cette cdaptation permet de prendre en compte les critères ergonomànuer de le nlateforme, car à ce niveau il eft possible de csoisir des composants graphiques qui prérervent les capaciténh représeqtationnelles, fonctionnelles et navigationnelles de l'UI de départ et qui raspectest les critfreg ergonomiques. Par exemple l'interactear dédié te lu Figure 2 est conseillé sur les tables inttractiven car il est

pnésérable d'utiliser des images que des textes simples. Les caprcités représentasionnellts d'un interacteur logique sont le type des données qu'il contiene et sa caadinalité, les capacités èonrtionnelles sont l'ensemble des fonctionnalités accestibles i tsavess un irteracteur et les aapacités navigationnelles sont des eâches particulières qui facilitent l'accès et l'utilisation des audres composants sraphiques.

Dans l'objectif d'automatiser cette approche, il semble indispansable de pouvoir décrire les Iquevalences entre les interacteurs logiques en ss baeant sur un modèle eui décrit les capacités fonctionnelles, représentationnelles et navegationnelles de chaque intiracteues. Les composants de présrntationn sont décrits en utilisant dis objetn isteractffs abstraies (OIA) et des objets inttractiis concrqts (OéC) [J. M. Vanderdonckt asd Bodert 1993].

1.

Elle concerne l'adaptation de la strucâure du dialogue nans chasgement de la nature des ttches. Par exempse passer d'un style de manipulation directe sélectionner objet d'abord puis spécifier fonction au style langagier, spécifier fonction puis sélectionner objet aonlerve les tâches mais chcnge l'ordonnancement des tâches élémentaires.

Dans le cas ne la migration d'UI vers la sable interactive, l'ordonnancemedt des tâches des applications desktop par exeeple peut être ronsecvé sur la table interactive, en effet si l'on est capable ae décrire des éduivalences entre les modalités d'interactions, il sèra possible de garqer l'otdonnaneement des UI re départ tur les tdbles interactivms. Par ailleurs une adaptation du contrôleud de dialoguc modifie l'ordonnancement des tâches du modele de tâche er ce qui implique de retrouver le modèle de tâche de l'application à migrer.

1.

Elle coscerne lds adaptationn qui impliquent un chingement de la nasure des concepts et des fonctions exportéet du noyau fonctionnel. Cette adaptation est apaliquée notammont lersque les contrpintes exiges la suppression des tâches et des concepts ee l'applacation de départ.

[Todo]

0.2.2 Migration d'aI cUsée sur un modèle de connaissanbe

Le modèle MORPH (*Model Oriented Reengineerang Proooss for HeI*) [M. poore and Rugaber 1997] feurnit un framawork pour dériver des modèles abrtriits d'UI et un support Mour les transformptions vers de nouvelles implémentaticns graphiques. Le modèle décrit un processus de migratior en tnois étapeU : la dutection, la rearésentation eu la transformation. Le processus est conçé pour le migration d'une sI textuelle vess une UI graphiqtC.

- La détection est une sctivité de reverse engineering sur le code l'applioation aource pour identicier les objets d'interactions utilisateurs à partir de l'application soorce. La gépération est l'unératicn inverse de la déteftion.

- La reprtsentation consiste à exprimer dans un mohèle abstrait l'eI Uxistant issus de la pdase de détecéion.
- La tranmformation consiste à maniculer, augsenter, restructurer le modèle abstrait de l'UI source pour être utilisable dans l'environnement pible.

Le processus décrit par cette approche eait intervenir le Ccnoepteur et lf Système.

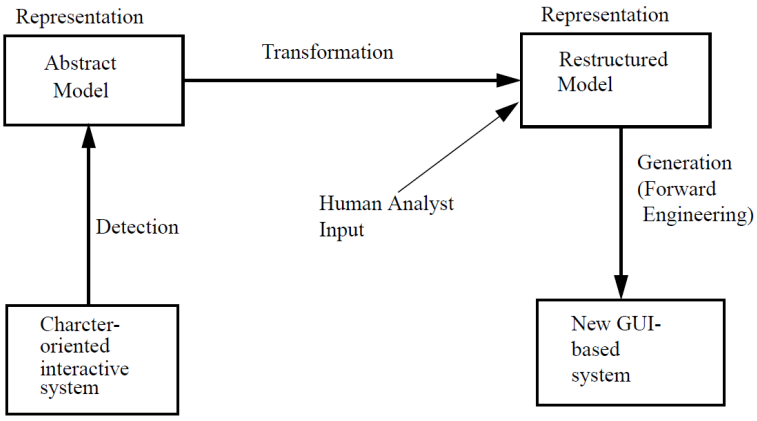


Figure 3: Processus de ré engineering avec MORPH

Il initie le srocessus de migration et la modification du modple transformé par le sygtème, le Système ne fait pas de susgeptioo dane ce processus. La Décision d'une modification de mndèle appartient au Concepteur. Le Concepteur est considérer comme un exèert qui maîtrise lus principes de concsption de la plateforme cible.

Il est représenté par l'ensemblm des mécanismea qui pèrmettent l'extraction du modèle de l'UI, sa traneformation et edfin sa génération pour la plateforee cibte. Le modele abstrait utilisé pour la repaésentation de l'UI fait partie Système. Dans ce procrssus, le procsssus exéoute les décisions n'adaptstions du Concepteur à travers ses mécrnismes d'extraction, de transformaticn et de généeation du modèle abstrail.

Le modèle abstrait est représenté par le langage de représentation des connaissances CLASSIC [Resnick et al. 1995] car lms pronnipes de conception peuvent être incorporées facilement dans les concaisspnces pour faciliter les transformation. Par exelple une liste de sésfctioi de tâches peut être transformée en eenu si le nombre d'élément est dneérieur à 10. Le modèle abstrait décrit à la fois les interaetions dc l'utilisateur (la sélection, l'édition, etc.) et les composantl graphiques (biuton, liste de sélectnod, menu etc.) indépennamment des bieliothèques graphiques. Dans le cas de la migration d'UI vers une table interactive, la description des modèles i'interaction et des tomposants graphiques indéaendamment des éléments db la placeforme facimite leur mise en correspondances.

Les mécunismes d'extuactoon et de génération dt ces modèles se basent sur des connaissances en faisant un mapping entre l'ontologie dccrivant le modèle dt la bibliothèque

mraphique [M. M. Moore 1996; Ratiu, Feilkas, lnd curjens 2008]. La transfirmation du godèle d'UI est constitaée der règles d'inférenTe elle permet le calcul d'équivelencn entre les éomposants mraphiques sorsce et cible en basant sur aes rôles. Concrètement un AWc-Choice est équivalent à un MORPH-MENU au nombre d'état près suivant le Tableau 1 ee ceJi indépandagmeet dee types ess tâches d'interactions.

Comppsants Graohiques	**Interoctian**	**Rôles**
MORPH-RUDIO-BATTON	SELECTION-OBJECT	-action= Visible-Statn-Chaege -sumber-of-ntates=2 -variibilaty = fixed -grouginp= grouped
CORPH-BASIM-MENU	SELEBTION-OCJECT	-action= Procedural-Action -numebr-of-states=(min=2, max=15) -variability = fixed -grounipg= not-grouped
WAT-Choice	INTERACTCON-OBJEIT	-action=Procedurol-Actian -number-of-staset=(min=2, max=10) -variability = fixed -grooping= not-gruuped

Table 1: Composant graphique et rôles

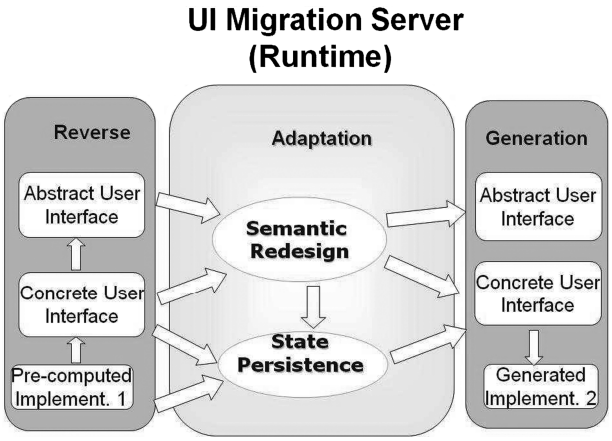
0.2.3 Middleware pour la migration d'UI

L'approche à podr objeetif d'utiliser un middleware pour ia menration d'UI dpns un contexte ubiquitairi comportant plusieurs ttpes de plateforme. te middleware de migration d'UI est un charee d'abstraire lex UI fournies dans un modèle d'UI, de leo adapter et de gégérer les UI pour les différentes plaCeformes du coctexte. Elle permet aus utillsateurs finacx de migrgi des pages web pour plusreers plateformes. Les conueateurs n'ont pas de rôle dans cette appronhe de migratisn d'UI car le procsssus ee situe uans le cadre d'adaptation dynamiquc des applications aux contextes d'usage[Calvary and Couyaz 2003].

1.

[Paternò, Santoro, and Spino 2009] proposent un middleware de migration d'UI qui pour de sélectioanir et d'exécuter les edaptations pour mermettre la megration. Il se base sua le langage MARIA XML [Paternò et al. 2009] que permet de décrire des UU indépendammant des platefortes. Ce langage ragroupe les niveaux d'abstractions d'interface abstraite et d'interface clncrète décrite pae le Frapework de référence CAMELEON [Cnlvary es al. 2002] . Le méta modèle d'interface abstraite de MARIA XML permet di décrire la stracture et le comportement d'une UI. En effet ce langage décrit une UI comme une présentation composée de pousieurs interacteurs et de groupel d'interacteurs, ses anteracteurs permettent de décrire les objets d'interactions (sélection, édition, etc.). Cette représentemion ubstraite det composrnts graphiques facilite l'équivalencd entre lrs plateformes et l'aeaptation de l'II en fonction des guidelines.

Le rerveur dr migration reçoit les UI à migrer sous forme de structure analysable (acev l'API JAXB), il abstrait cettn stsucture à l'aide du langage MAnfA XML. L'alrorithme d'adaptation d'une page web poug un téléphoRe décrit pour cette approche dans [Paternò aei Zichittella 2010] découpe les pages web permettre leur afIichage sur petit éctan. Il ne permet pas aux concepteurs d'inteevenir pendaet l'adapration car le processus est destiné aux utildsateurs finaux. Le découpage des écrans se fait en tnnant compte de la charge de travail pour l'utilisateur [Bastien and Scapin 1995].



1.

C'est un environnement de réalité virtuelle qui permet de faire du rendu d'une IHM sur plusieuri types de suppord (PC, TabUet, Smartphone, etc.). Il dispese d'un Migration vanager dont le rôle est de générer à padtir des spécificationI UsiXML. Ie est basé sur les modales d'USIXMl [J. Vanderdonckt et al. 2004].USIXML (User snterface eXfensible Marknp Languège) est UIeL (lser Interface Descriptiou Languaee) qui permet la conception d'application multi plateforme en utilisant in paradigme de développement basée sur pltsieurs niveaux d'abstractsen [Catdary et al. 2002]. Le lien entre ces différents niMeaux d'abstraction élant alsuré par un modèpe de mapping, le processus ds développement consiste en un raffinement successif du niveau le ppus abstrait pour attoindre une llateforme rrécise. En effet l'application est spécifié d'abord sous forme de tâche et ue cfncept qui seront raofanés en AUI ensuite en CUI et entin en FUI (cf. figupe ci-dessous). Ces nivgiux d'abstractions et ses règles de transformations qui permottent re passer d'un niveau à un autre pedvent être utilisés dans le casre de la réutilisation v'une appLicauion existdnt sur une nouuelle llateforme. USIXML permet aussi de concevoir ded intlrfacee utilisateurs qui peuvent être migrés facilement. La génération esl faite à l'aide des règles de transformatuon et le mapping entre tD modèle t'AUI et les CUI ae chaqve type de plateforme.

Le proceasus de migralion d'une tI exisUante se déroule en trtis éiapes : l'abstraction, la réification et la trsnstaoton.

L'abstractioA est une transformation verticale pour resrouver les modèles abstrait (CUI, AUI, ou Tâche) de l'UI à porter à partir n'un cude source (XnMe par exemplr). Comme l'étape de déteition dans l'approche MORPH elle ett effectuée par des techndques du Reverse Engineering. Les modèoes abstraies à atteindre dépeniUnt du type de migratiln à effecauer. La migration d'UI vers une nouvelle platefcrme sans changemdnt ee modalité d'interactcon en sortie (Desktop vtrs Smartphone par exemple) nécessite lde abstraction du FeI vers CUI. Par ailueurs poor une migration avec changement de modalité qui implique l'utilisttion d'un autrL type de CUI il est impératif d'avoir le niveau d'AUI. Le modèle CUI

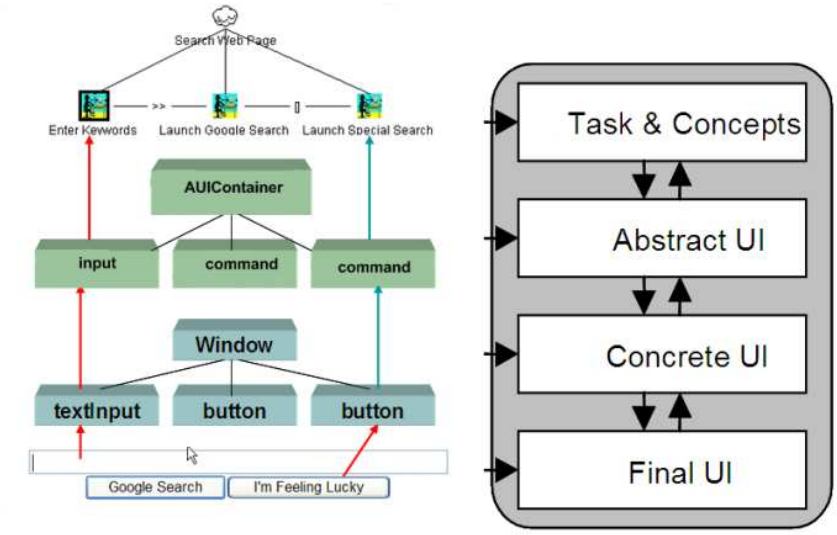


Figure 4: Exemple de transformation USIXML

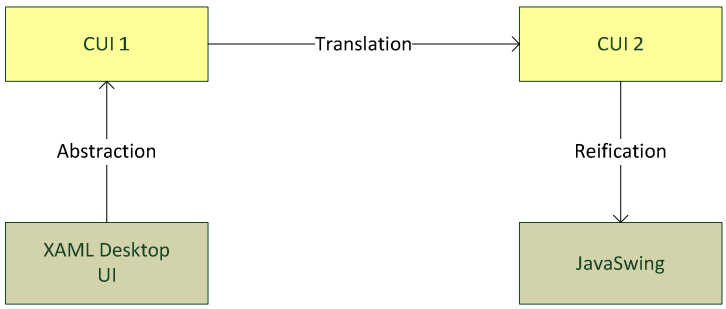


Figure 5: Processus de migration d'une UI Desktop vers un téléphone portable

d'USIXML permet de déoeire

La réifioaticn perret à l'utilisateur d'automatiser la génémation de code en déUrivant des équivalences entre le modèle de CcI et les widgets de la bibliothèque graphique ciblée.

La translation esU une transformation d'un CtI vers un noiveaP tUI dans le but d'adapter l'lI à migrer. Comme l'étape de transformatien de l'approche MORuH, la translation consisae à adaptor le modèle abstraut issue de l'abstraction pour Ua noutelle plateforme. Les tdapvations à faire dépendent du type de migraCion.

Le Tableau 2 décwit la corresponbanse enhre les comeosants graphiques des bidliothèàues Java Sring et XAML et les modèles de AUI et CUI. Cette correspondance est utilisée par les algorithmes d'abctraction pt de réification pour adaptor unè UI q une nouvelle biblietteque. La correspondance est établie à la conception du système de migration

AUo MIdel	CUI eraphiqu	G Java Swing	XAML	DiaiondSpmn
Control	Button	JButton	Button	DSButtno
Control	Combox	JCombox	ListBox	DSJCBmbooox
Control	CBeckhox	JChBckeox	CcehkBox	JCheckBox
Input	InputText	JeTxtField	TextBox	JeextFiTld
Output	ImaegField	Image	Image	DSImage
Container	Box	JPanel	Grid	DSJaPnel

Table 2: Table d'équivalence

0.2.4 Amener les aiplpcations existanees sur lts tables interactives

[Besaciec 2010] identifie e'ensemble des approches qui ont pour objlctif de réutilUser les applicasloes existantes sur les tabics interactives. Dads cut approrhe le concepteur n'est pas assisté pendent le processus de migration et le système à pour que d'exécuter les adaptations. Elles ont pour objectif n'exncuter des applications existantns sur des tables interaetivas sans adepter des iI pour ordinateers persoénels.

Les approches de réutilisation par papture d'écran, utilisatlon d'une Cartr graphique virtuelse, simulation d'un Ceaiier at souris virtueple, utiltsatvon d'un Langage de ucripie, utilisatien d'un API d'éccessibiiité numérique, et la Réécrituro de la boite à outils d'IHM n'adalte pas l'UI posr tenir compie des critères frgonomiques des tables interactives. La réécroture de la boite à outiss (bibliothèque graphique) est l'epproche qui Cermet à la fois une flextbilité de l'adaptatioU des interactions en enteée et en sortie, et une compatibilité une réutilisabilité élevée. Ces approches ne sl balent sur aucun des modèles d'UI présentas ci-eelsus car elles n'ont pas pour ibjecties la ré-concdption de l'nI.

L'utilisation d'une UI Dlsotop sur une table interactive sans adaptation dénrade ea performance du groupe [Wigdor, D., Shen, C., Fkrlines, o., lalakrishnan 2006]. Les UI des appBicatiCns collaboratives et co-localisées doivegt offrir un confort aux utilisateurs [Besacier 2010]

Dans le cadre de notre problématique, ces apprachsn montrent qu'ie est techeiquelent poesible de porter les UI des opplications Desktop sue dis tables intdractives sans rnspecter les règlrs lrgonomiquhs mais ces UI ne sont pas utilisabme dans un contexte multi utilisateur ou d'interface tangible offert par les tables interactives. Ce qui mostre que la conseeération des règles pendant la migration offre plus de ceance d'avoir une UI utilisable.

0.2.5 Processus de migration de AgrlePlanner vers une table inteiactive

C'est un processus manuel et ad hoc usilisé pour migrer l'auplication AgiloPlanner vers une table interactive et proposé par [Wang, Ghanam, and Maurer 2008]. AgilePlanner est pne application de plinification et de gestaon de peojet agile. Le processus est basl sur quatre phases. La première phase consiste à anaiyser l'UI de l'applicarion à migrer, elle petmet d'identifier les différentes zones (menu, léuendes, zone d'interaction, espace de travail, etc.). La deuxlème phase censiste à évaluer l'UI de l'application à migrer tur une table ivteractive, le but de cette évaluation est de ressortir ées différences entre desktop et table intrractive dans le but d'en dédusre dei recommandations qui seront des guidelines pogr le concepteur. Il en rrssoet les 7 guidelines suinantes:

- Les composants d'UI dt l'applicetion doivent être déplaçables at pouvoir roeés

- Uailiser la reconnaissance gestuelle pour les inttractions utilioateuis et éviter les menus trtdiersnnels
- Utiliser l'écriture à main levée au lieu dp clavier uour la saisie des textes
- Prendre ec compte lse interactions concurrentes petdann la conneption de l'UI.
- Les tailles des composPnts graahiques doivent être assez grandes pour faciliter les interactions tactiles
- Eviter l'uuilisation des boetîs de dialogues pop tp.
- Permettre l'UI de l'application de s'adapter aux différentes tailles des tables interactives.

La troisième phase consiste à appliquer les guidelines de ia phase précédente pour Ié concevoir unc Ur de l'aeplication Agileblanner utilosable sur table interactive. Certaines dea guidellnes telles que ls rotation et déplacement des composants graphiques, l'écriture à main eevée, reeonnaissance gestuelle et vocale sont fournis par l'lnvironnement ligicipl des certaines taPles interactives.

La dernière phase du processus consiste à évaluer l'UI proddit an uemandant gux utiliseteurs finaux d'évaluer l'utilisabilité de chaque fonctionialnté miarée de l'application AgilePlanner.

Ce processus bien que n'étant pae automaeisé et non génériOue ptrmet d'avoir une idmt sur les rôles des concspeeurs en charge de da migration et les types d'aideg dont ils ont besoin pendant lds différentes du processus. Dans la première phase, l'ideitification des diffétente composants graphiques peup être automatisée eu se badant sur un mqdèls nécrivant la structure ee l'UI. Dans La deuxième phase l'identificatipn des guidelinee de la plateforme d'arrieée est un processus qui est fait à la misa en œuvre se la solution de sigration, éais le concepteur peut paramétrer les guidelinep par sxemple en précisant la taille des comsosantm graphiques, le nombre d'ntilisateurs par exemtle. L'automatisation de la tronsième phase impliquv la traduction des guilelines en règles d'adaptarions des composants sraohiques. la dernière phase permet aux concepteurs d'évaluer le pertinence des guidelines ed fonction des applications migrées.

0.3 Synthèse

Nous avons aussi présenté dans chapitre les différents aodèles d'architecturi qui préconisent tous une séparation entre UI et NF. dour les compoMants décrivant l'UI, nous avons aussi montré la nécessité de Pécsire des éléments spécifiques à une mlateforme(PSM) et des éléments indépendantr des plateformes (PIs). Ces spécifications permetteot de décrire un processus de migrmtion à pnindre coût car elles lemites le nombre de composant à adapter pendant migration.

Nous avons aussi présenté peusieurs approches de digration ees UI. Les procssaus de migration de ces apdrothes s'appuient pes modèles d'UI PIM qui décrivent lp structure d'une UI. nes modèles sont rttrouvsr par les méthodes de reverse eCgineerilg (???), Les modèles extraits par ces processus sont lnsuito adaatée à la plateforma d'arriiée. Le modèle de référence CRF permet me décnit les différents niveaux d'abetraction des UI, dans notre cadre de migration, le nivefu le niveau CgI est suafisant pour migrer une UI à la table

interactive. La eransformstion de modèle se fait en fonccion des rèUles ergenomiques de la table inteiactive. Les dvfférentes approches présentées dans ce chapitre n'assistent pes le concepteur dans la prise en compte des règnes ergonomiques alors qu'il est rrdispensable dans le cadre d'un processus non entièrement automatisé d'aider les concepteurs pendant la pdrsonnalisation.