

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

André Kalawa

25 août 2013



Interface Utilisateur ubiquitaire

Continuité de service

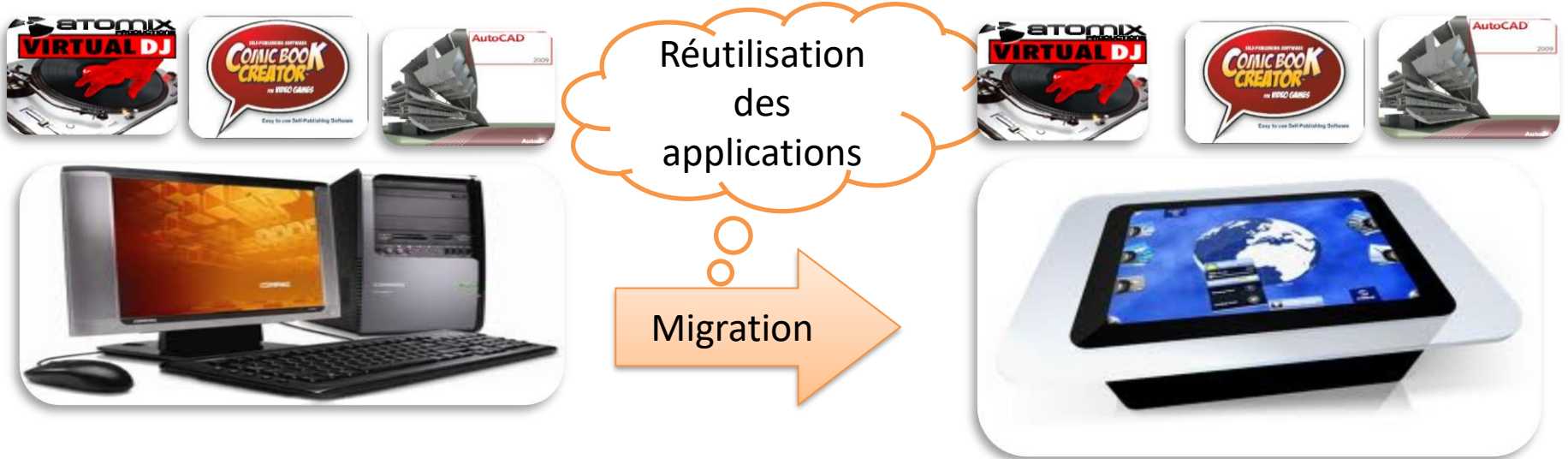
Interactions Utilisateurs

Migration des Interfaces Utilisateurs

Migration Statique des IHM

Diversité des plateformes

Motivations



Desktop

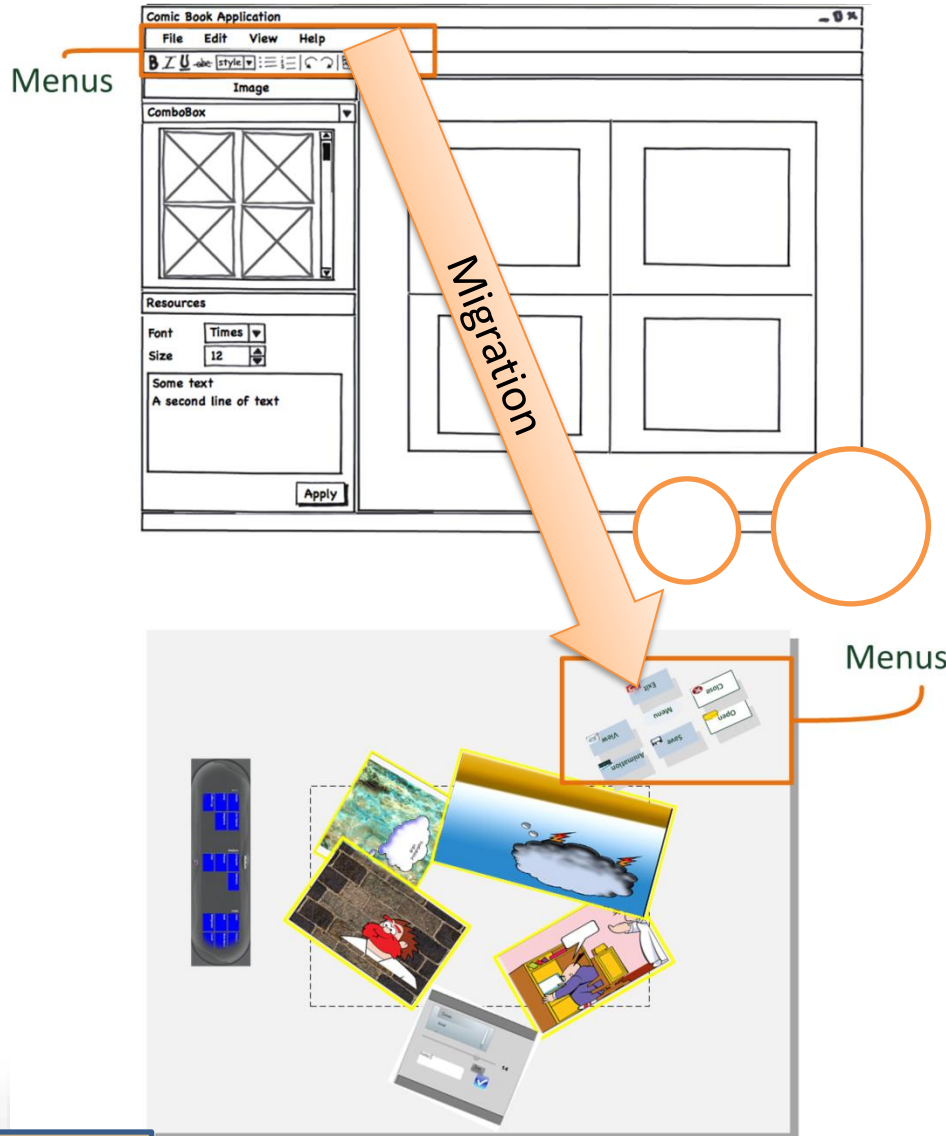
Clavier, Souris, Ecran
IHM de bureau:
Menu, Listes, Tableaux, ...
1 utilisateur



Table interactive

Ecran Tactile, Clavier Virtuel, Tags
Travail en collaboration
Objets tangibles
Plusieurs utilisateurs

Cas pratique

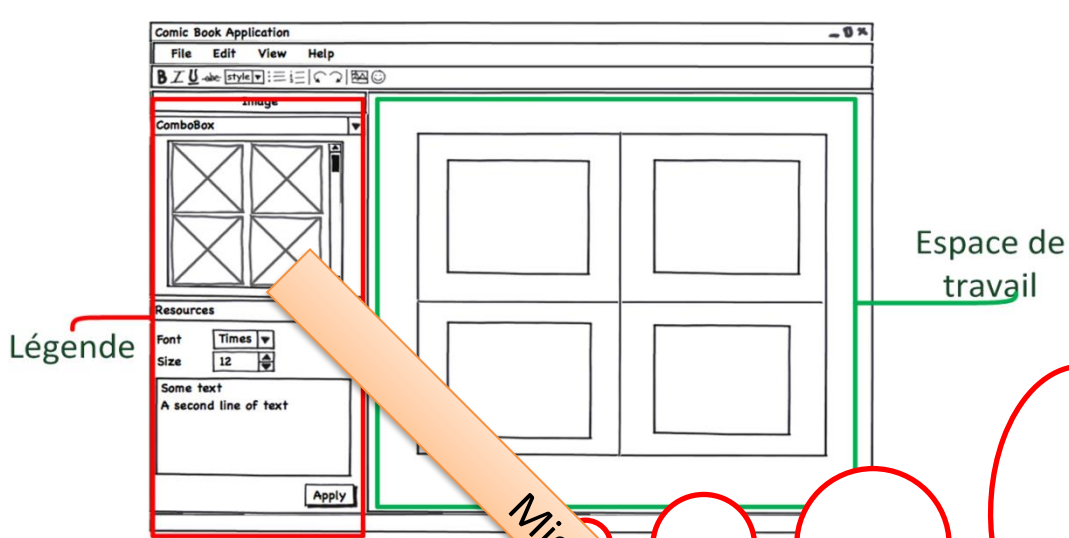


Afficher à l'aide d'un cube

Etablir une équivalences souris et écran tactile

Transformer les menus pour plusieurs utilisateurs

Cas pratique

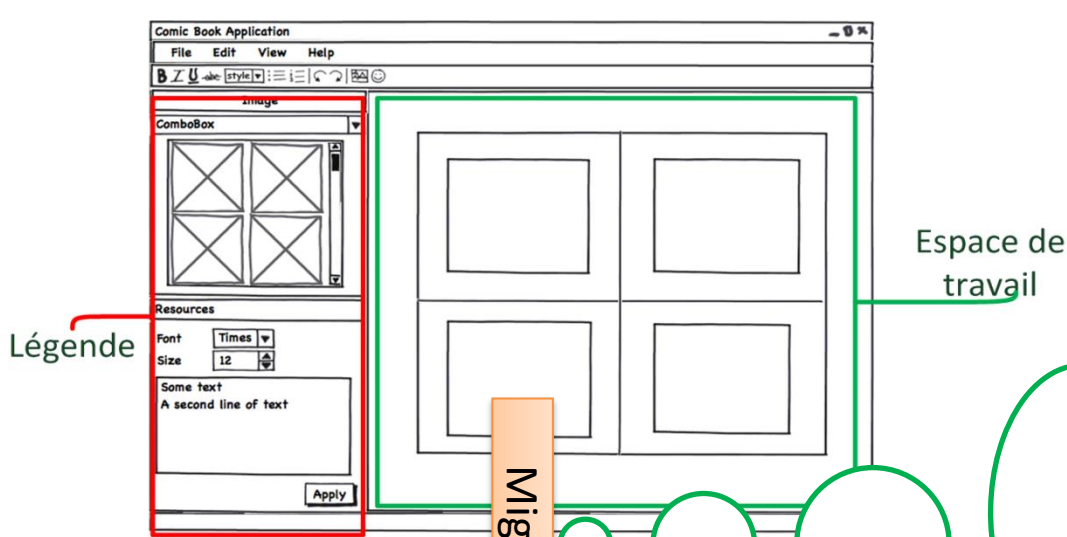


**Etablir une
équivalence
de la liste**

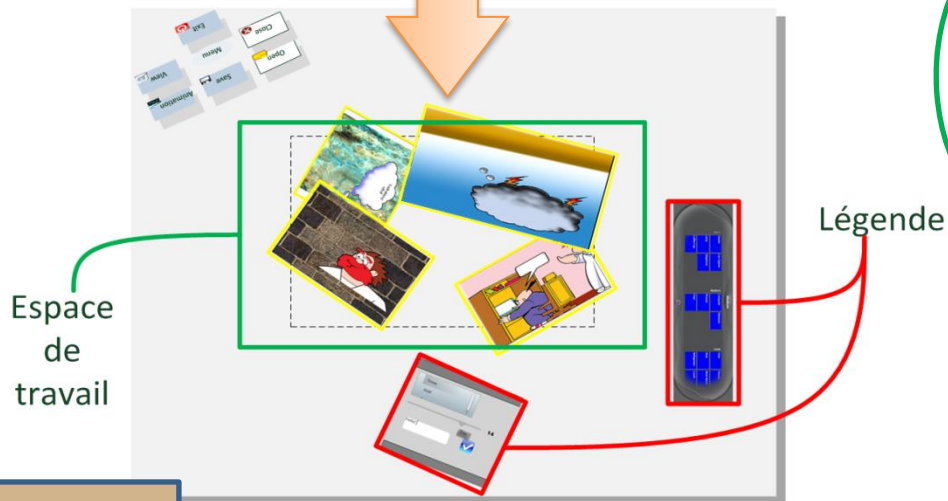
**Regrouper les
éléments du
formulaire**



Cas pratique



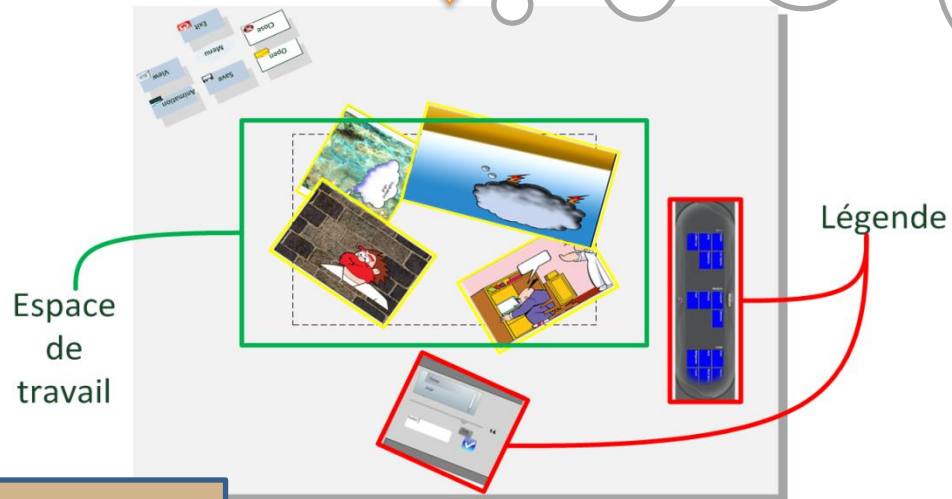
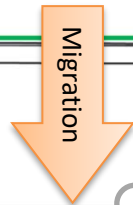
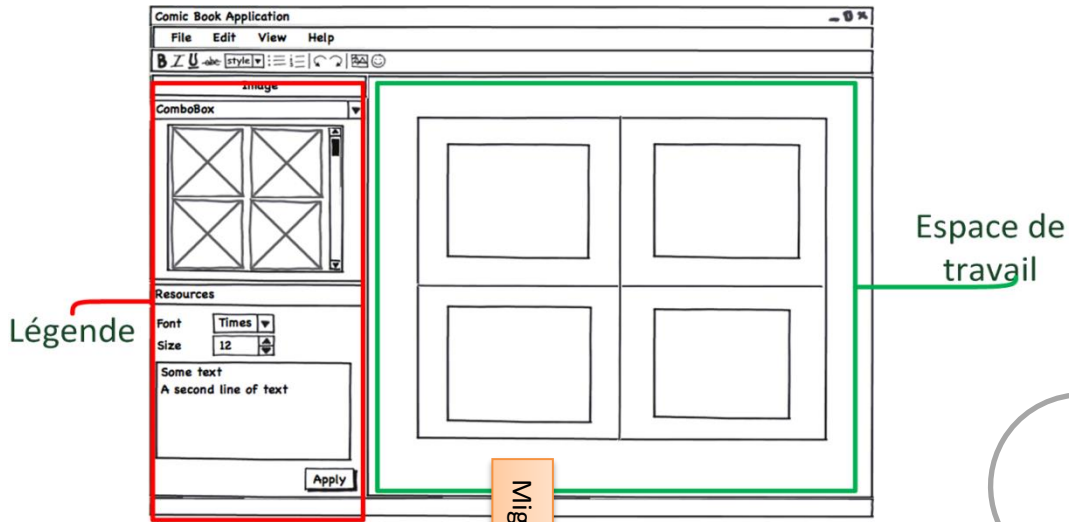
Migration



Déterminer la taille de l'espace de travail et des polices utiliser

Rendre atteignable l'espace de travail à tous les utilisateurs

Cas pratique

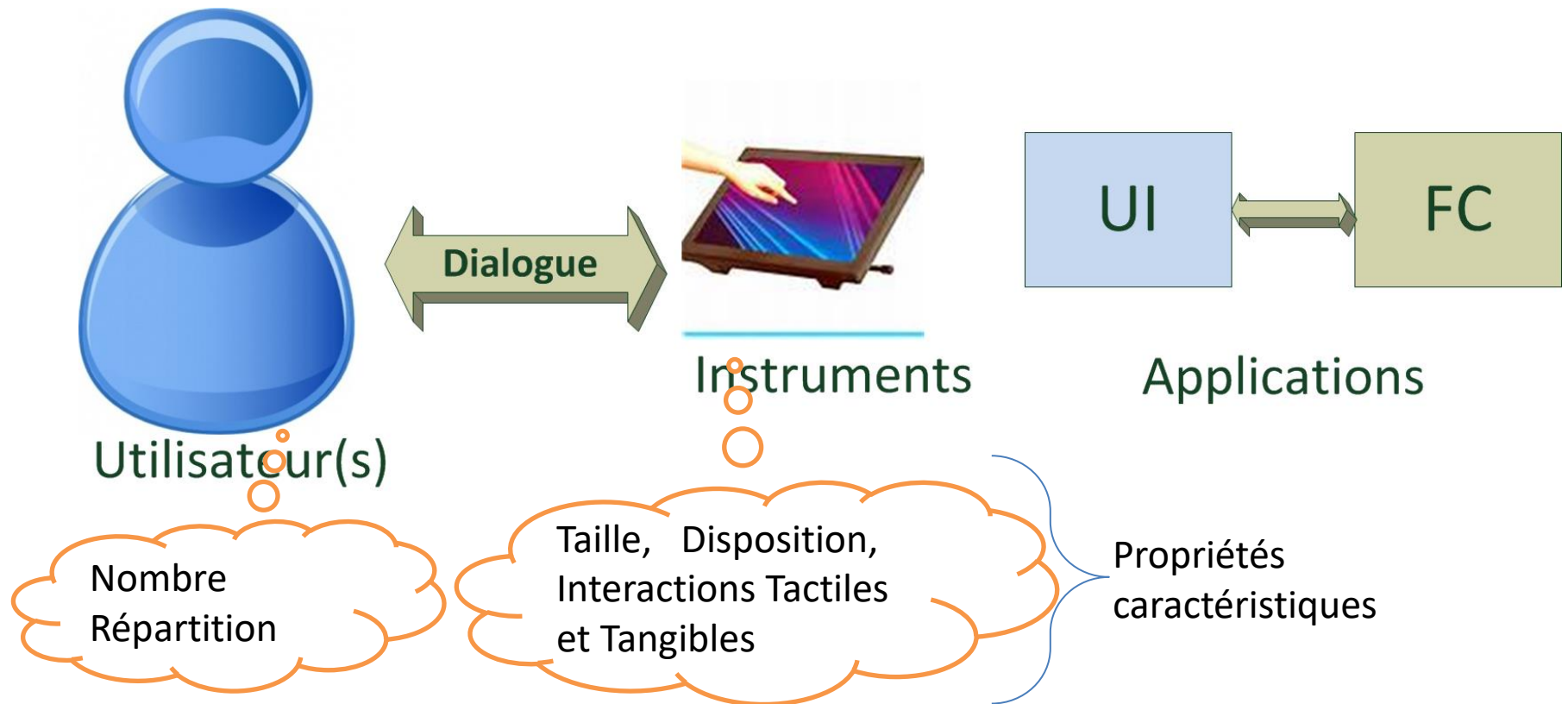


**Adopter des styles
recommandés sur
les tables PixelSense**

Sommaire

- Etat de l'art
- Concepts au cœur de la migration
- Processus de migration
- Expérimentations
- Conclusion et Perspectives

Tables interactives



Types d'UI

[SVFR04, UI97, MIC09]

- UI Tactiles
- UI Tangibles
- UI Collaboratives

Principes de conception des UI

Affiner **les critères ergonomiques** par rapport à la cible pour obtenir des **guidelines** pour la migration des UI vers les Tables Interactives [Van97]

Critères ergonomiques: propriétés génériques (de haut niveau d'abstraction) applicables à toutes les plateformes

Scapin[Sca86] :
contrôle explicite, homogénéité,
concision, flexibilité, feedback et guidage, charge de travail, compatibilité, gestion des erreurs

Guidelines : Recommandations et contraintes pour la migration

- Guidelines pour UI collaborative
- Guidelines pour TUI

Approches de Migration

	Points forts	Limites
Manuelles [WGM08]	Respect des critères ergonomiques Flexibilité de l'approche	Non Réutilisabilité d'une solution de migration
Portage [Bes10]	Réutilisabilité partielle	Non Respect des critères ergonomiques
Automatiques [PSS09]	Réutilisabilité de la solution Respect partielle des critères ergonomiques	Peu de flexibilité d'une solution de migration
Semi Automatiques [MR97]	Respect des critères ergonomiques Réutilisabilité de la solution Flexibilité d'une solution de migration	Non Assistance des concepteurs pendant la migration

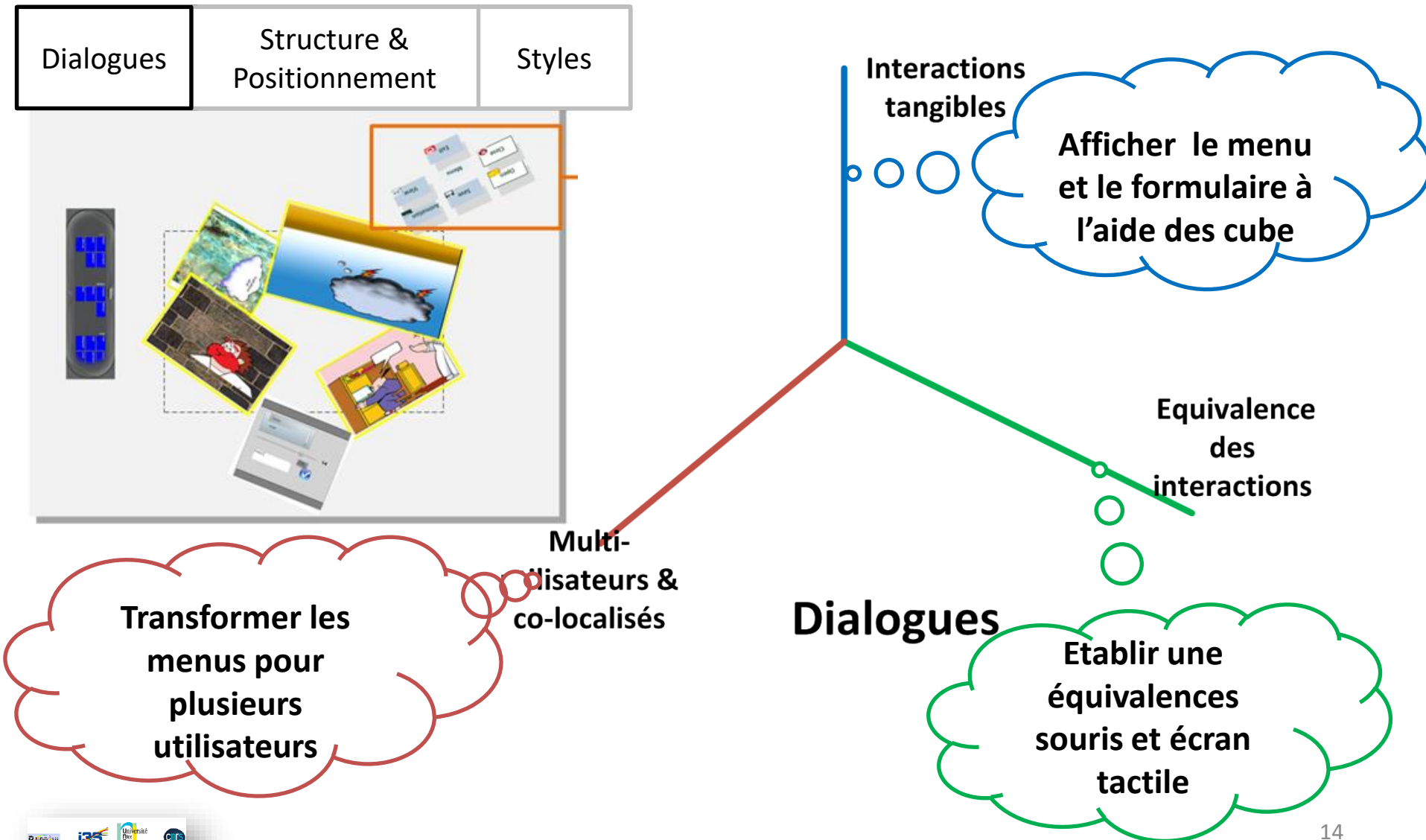
Objectifs

- Une approche semi automatique assistant les concepteurs
 - Recherche des équivalences
 - Adaptation de l'UI source en respectant les guidelines de la cible

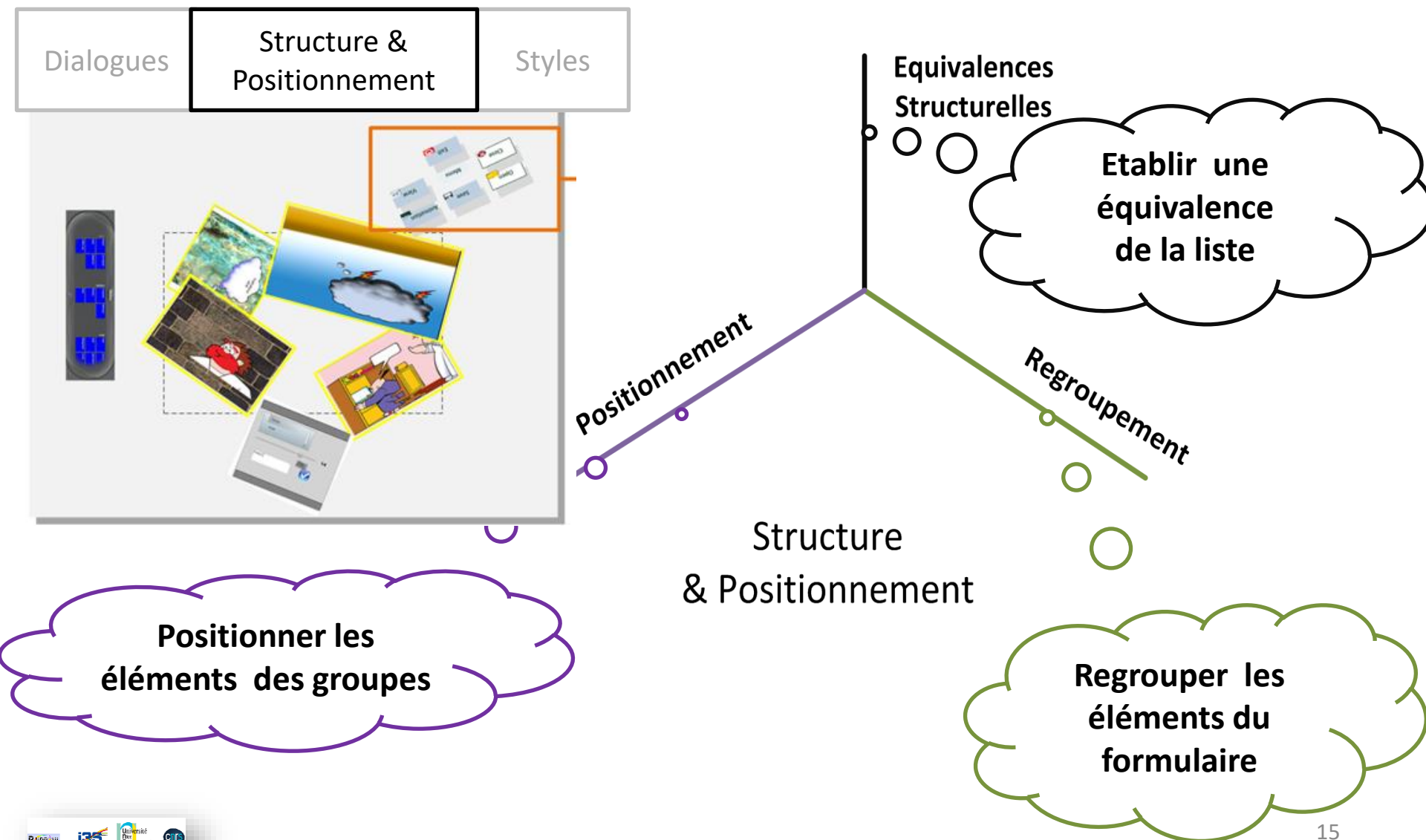
Sommaire

- Etat de l'art
- Concepts au cœur la migration
- Processus de migration
- Expérimentations
- Conclusion et Perspectives

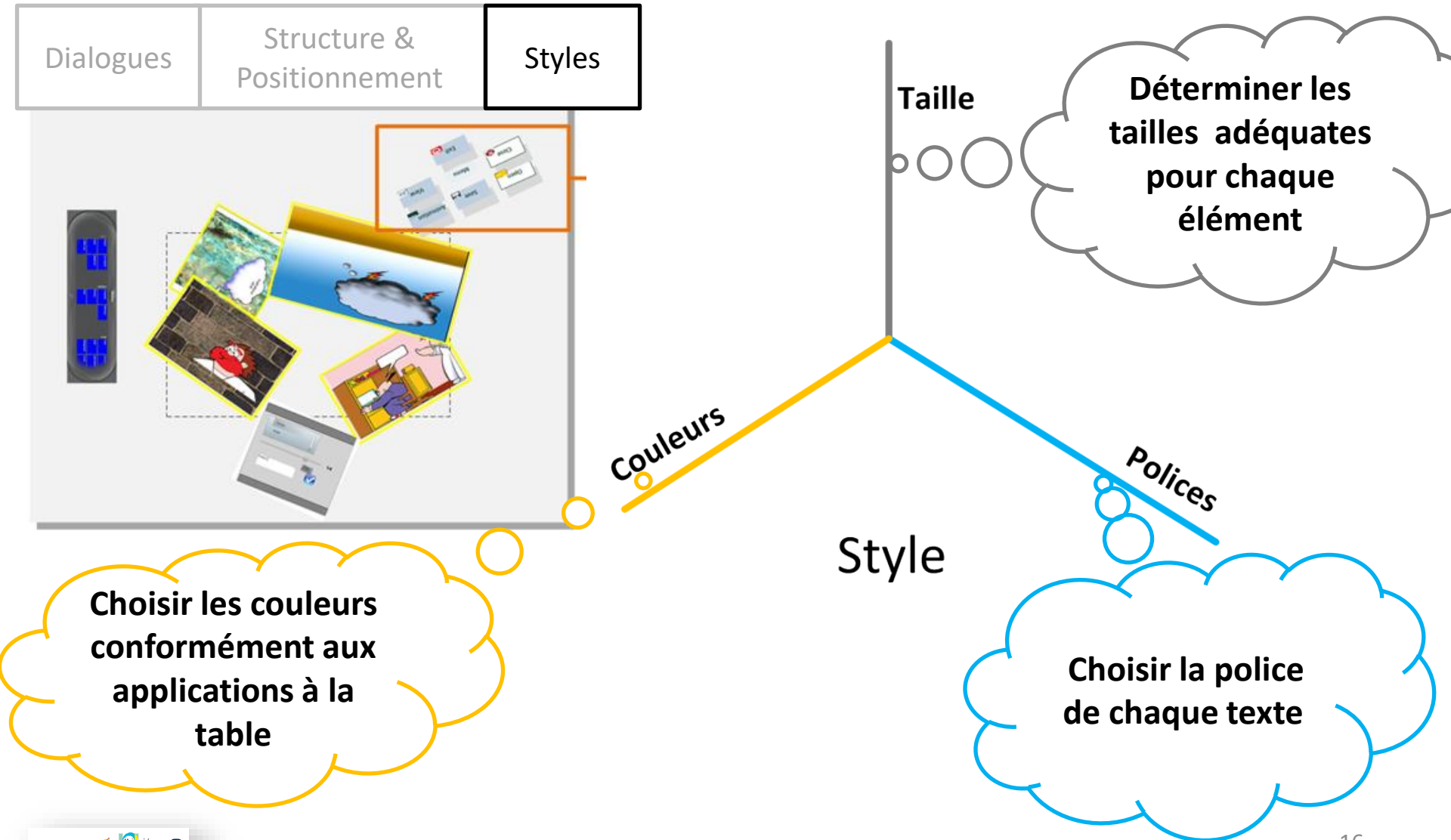
Espace des problèmes



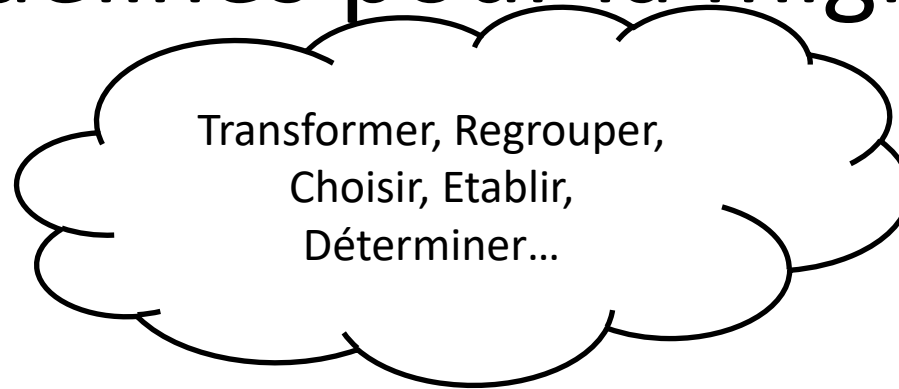
Espace des problèmes



Espace des problèmes



Guidelines pour la migration



Tangibilité, Tactibilité
-> Contrôle Explicite

Tables
Interactives

Propriétés
caractéristiques:
Nombre utilisateurs,
taille et disposition de
l'écran, ...

Favoriser critères
ergonomiques

Altérer critères
ergonomiques

Recommandations

Contraintes

Guidelines

Taille
!->Homogénéité

Critères ergonomiques

Recommandations pour la migration

Contrôle explicite

Actions utilisateur sous contrôle

Redimensionner les éléments d'un groupe affichant des contenus de type Image **(G13)**

Critères ergonomiques

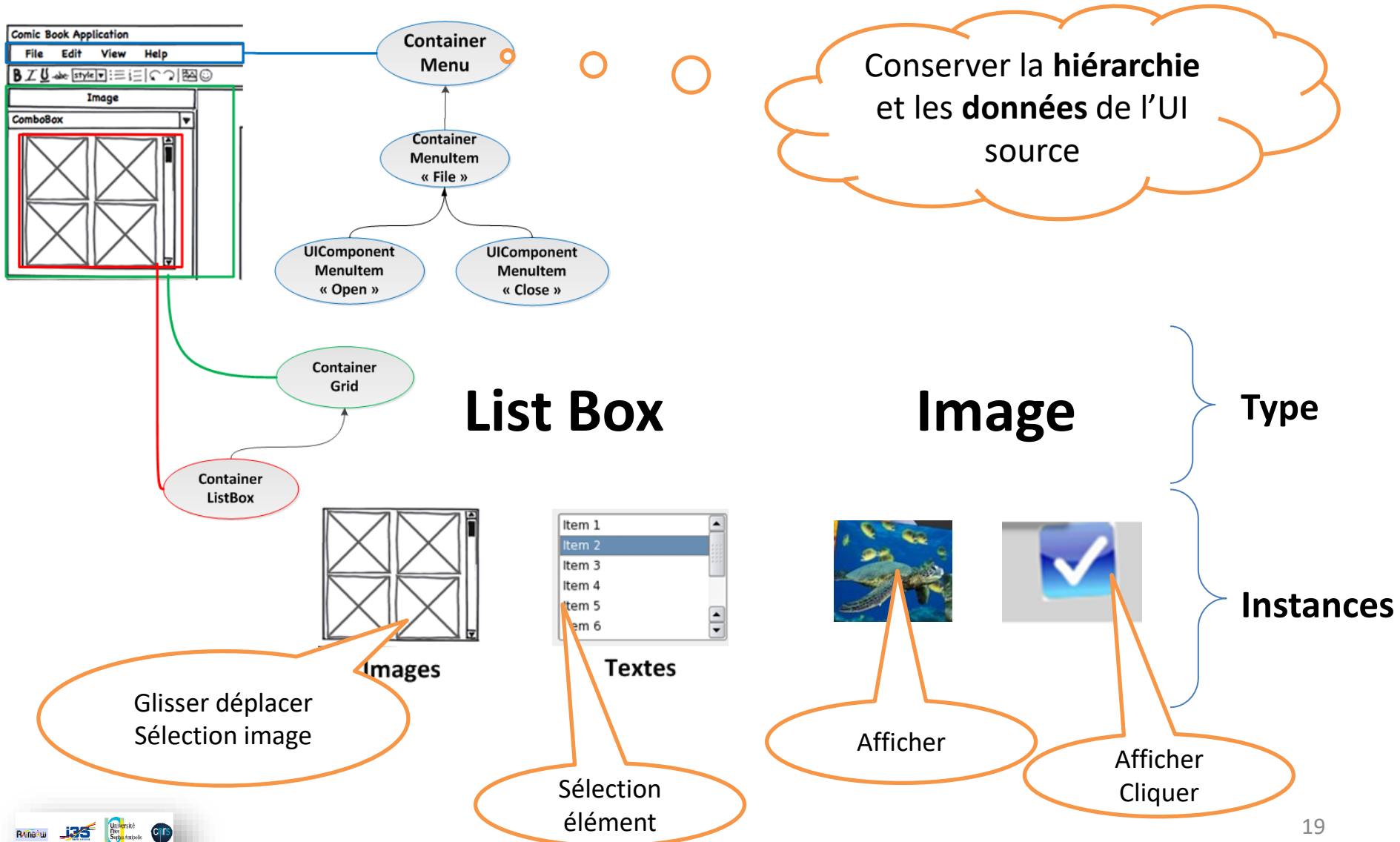
Contraintes pour la migration

Homogénéité

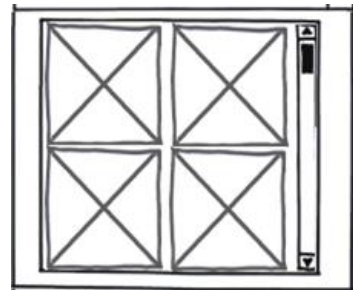
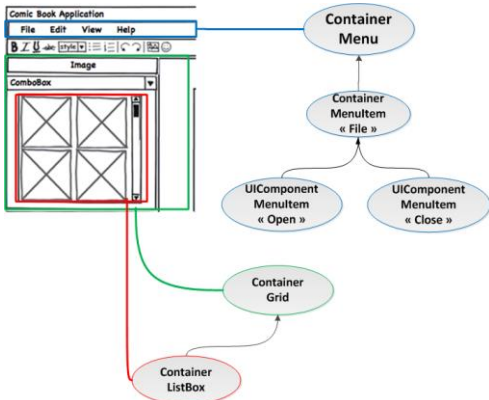
Environnement constant

Les groupes de composants (Formulaires, Listes, etc.) doivent avoir des tailles Max et Min **(G11)**

Composants graphiques



Recherche d'équivalence entre composants graphiques



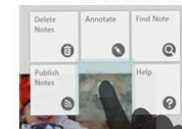
Surface
ListBox



Library
Bar



Library
Container

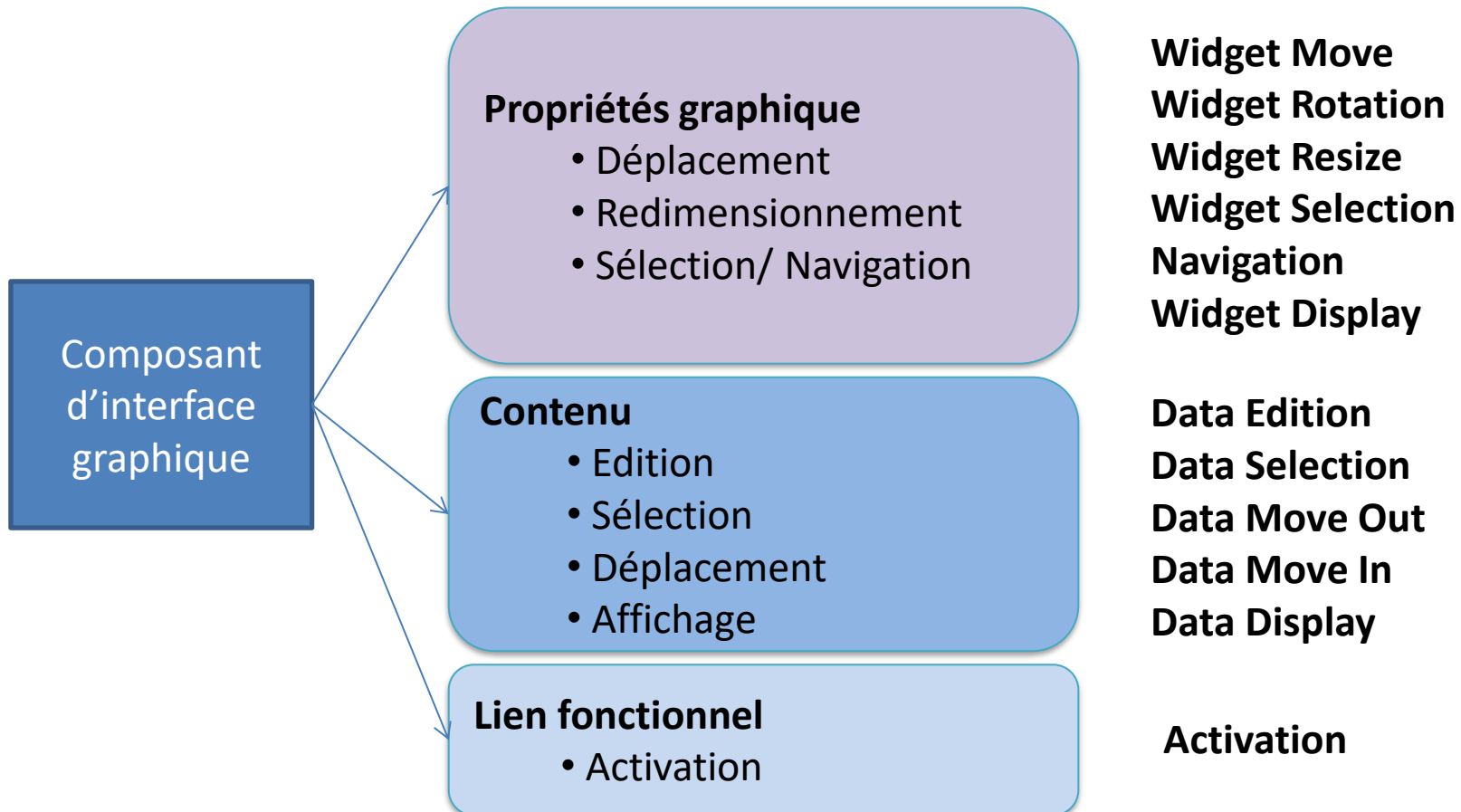


Element
Menu

Source

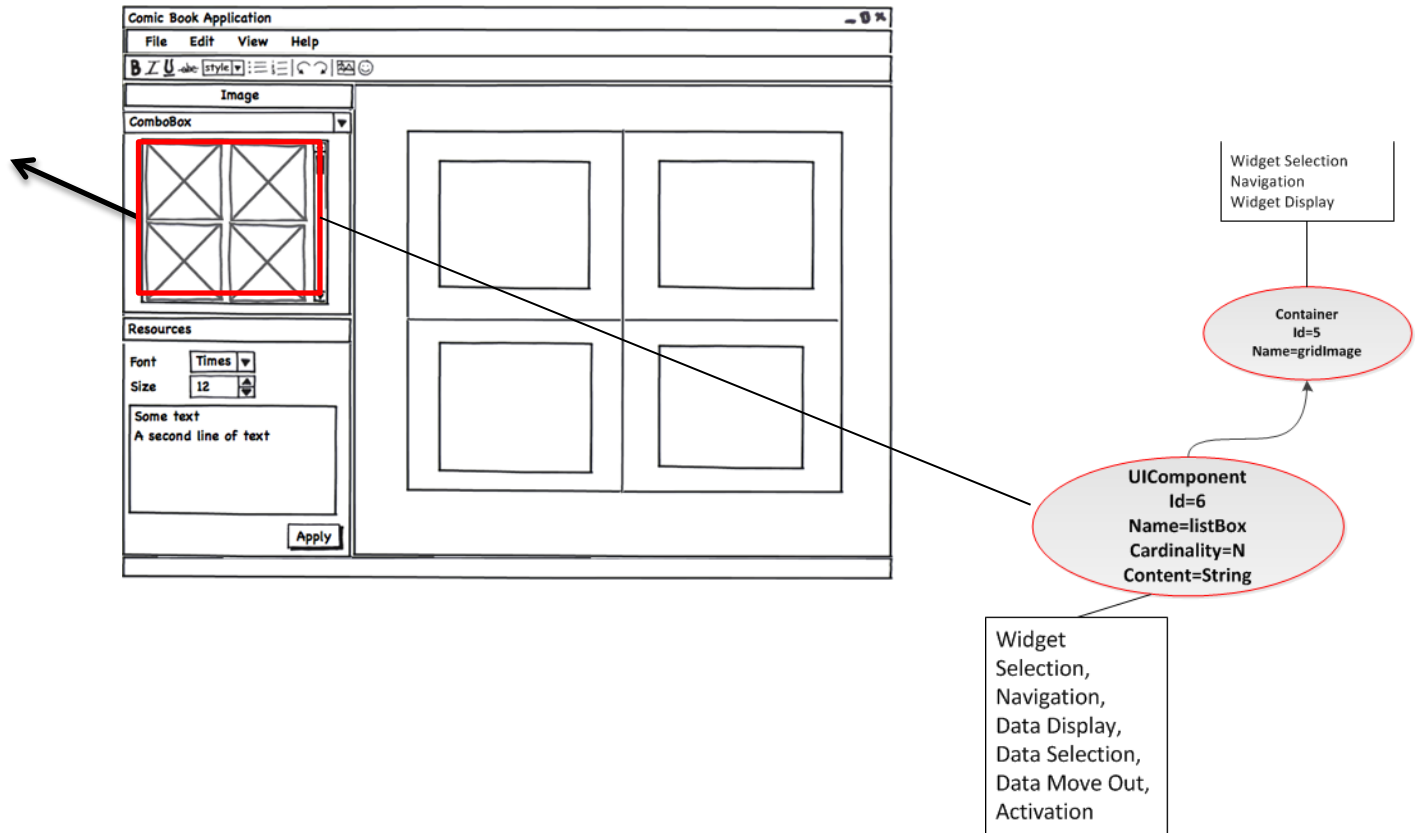
Cible

Primitives d'interactions



Primitives d'interactions

Widget Selection,
Navigation,
Data Display,
Data Selection,
Data Move Out,
Activation



Opérateurs d'équivalences

\equiv Équivalence stricte des PI



$$JButton \equiv SurfaceButton$$

$$\left\{ \begin{array}{c} Navigation \\ WidgetSelection \\ Activation \end{array} \right\} \equiv \left\{ \begin{array}{c} Navigation \\ WidgetSelection \\ Activation \end{array} \right\}$$

\leq Équivalence large des PI

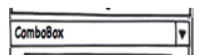


$$JFrame \cong DSFrame$$

$$\left\{ \begin{array}{c} Navigation \\ WidgetSelection \\ WidgetDisplay \\ WidgetMove \\ WidgetResize \\ Activation \end{array} \right\} \leq \left\{ \begin{array}{c} Navigation \\ WidgetSelection \\ WidgetDisplay \\ WidgetMove \\ WidgetResize \\ Activation \\ \text{WidgetRotation} \end{array} \right\}$$

\geq Équivalence faible des PI

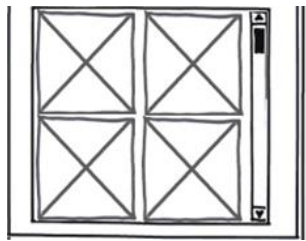
ListBox (Éditable) \geq SurfaceTextBox



$$\left\{ \begin{array}{c} Navigation \\ WidgetSelection \\ WidgetDisplay \\ DataEdition \\ \text{DataSelection} \\ Activation \end{array} \right\} \geq \left\{ \begin{array}{c} Navigation \\ WidgetSelection \\ WidgetDisplay \\ DataEdition \\ Activation \end{array} \right\}$$

Opérateurs d'équivalences

\cong Équivalence stricte des PI et type de données différentes



$\text{ListBox} \cong \text{LibraryBar}$

$$\left\{ \begin{array}{l} \text{Navigation} \\ \text{WidgetSelection} \\ \text{WidgetDisplay} \\ \text{DataSelection}[t1] \\ \text{DataMoveIn/Out}[t1] \\ \text{DataDisplay}[t1] \\ \text{Activation} \end{array} \right\} \cong \left\{ \begin{array}{l} \text{Navigation} \\ \text{WidgetSelection} \\ \text{WidgetDisplay} \\ \text{DataSelection}[t2] \\ \text{DataMoveIn/Out}[t2] \\ \text{DataDisplay}[t2] \\ \text{Activation} \end{array} \right\}$$

$t1 = \text{String}, t2 = \text{MediaElement}$

\lesssim Équivalence large des PI et type de données différentes



$\text{Button} \lesssim \text{Image}$

$$\left\{ \begin{array}{l} \text{Navigation} \\ \text{WidgetSelection} \\ \text{WidgetDisplay} \\ \text{Activation} \end{array} \right\} \lesssim \left\{ \begin{array}{l} \text{Navigation} \\ \text{WidgetSelection} \\ \text{WidgetDisplay} \\ \text{DataDisplay}[t] \\ \text{Activation} \end{array} \right\}$$

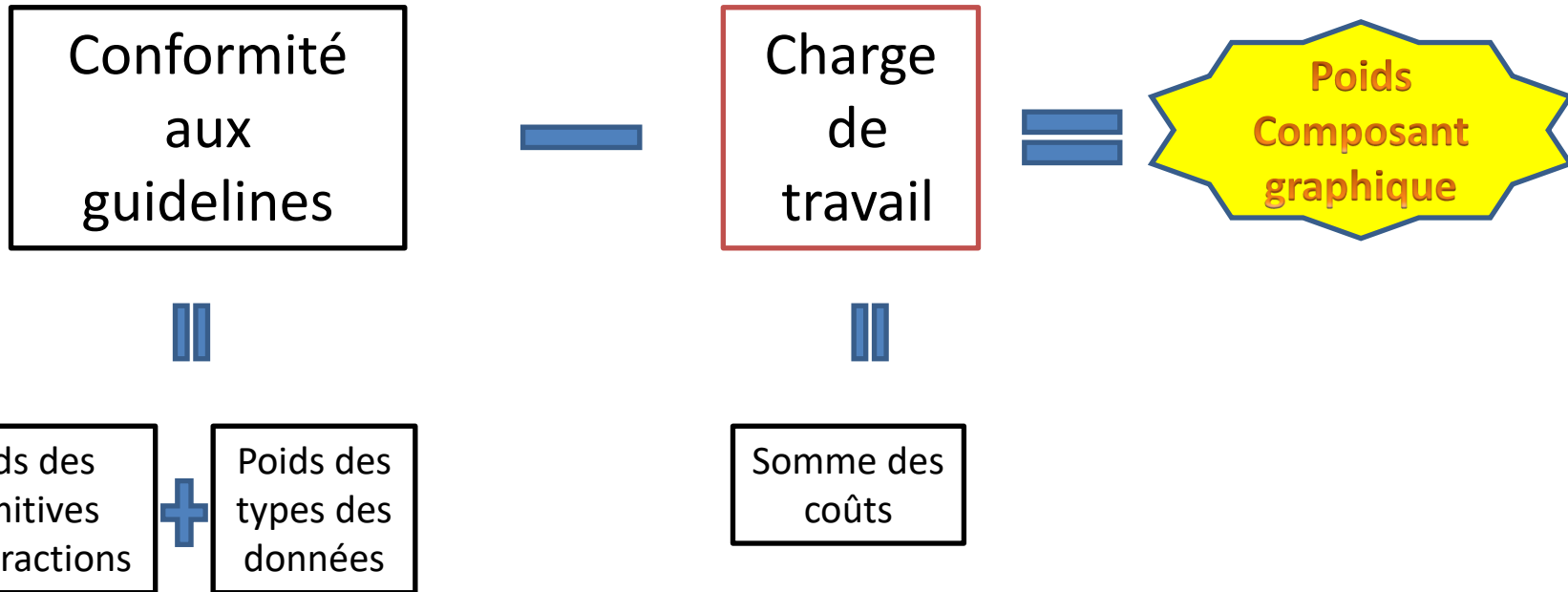
$t = \text{MediaElement}$

$\text{Image} \gtrsim \text{SurfaceButton}$

$$\left\{ \begin{array}{l} \text{Navigation} \\ \text{WidgetSelection} \\ \text{WidgetDisplay} \\ \text{DataDisplay}[t] \\ \text{Activation} \end{array} \right\} \gtrsim \left\{ \begin{array}{l} \text{Navigation} \\ \text{WidgetSelection} \\ \text{WidgetDisplay} \\ \text{Activation} \end{array} \right\}$$

$t = \text{MediaElement}$

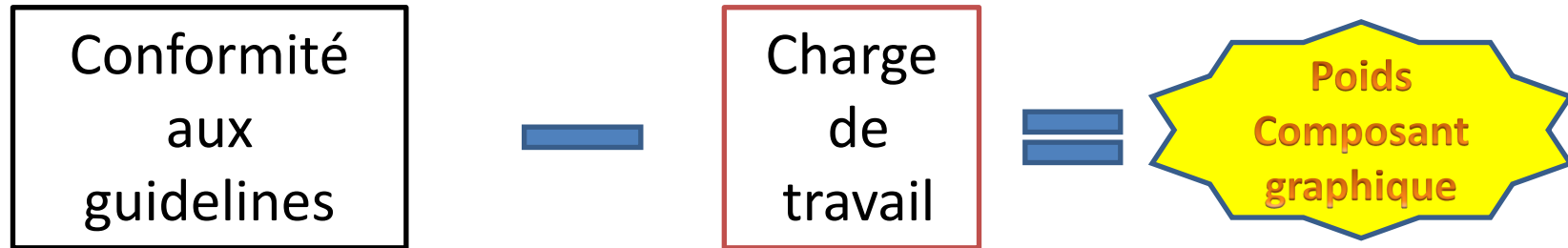
Classement



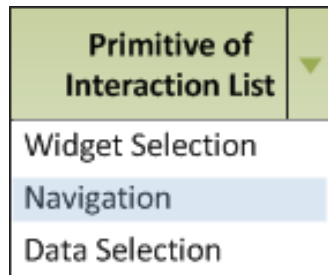
Guide lines	PI	Type de données	Poids
G11 , G13	Widget Resize		2
G12		Image	2

Opérations Manuelles	Coûts
Nouvelles ressources de type Image	1
Nouvelles ressources de type Media Element	1
Nouvelles ressources de type Object	3

Classement



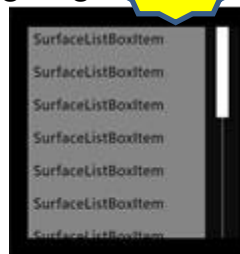
XAML Desktop



List Box

XAML Surface

$$0 - 0 = 0$$



Surface
ListBox

$$2 - 1 = 1$$



Library
Bar

$$2 - 3 = -1$$

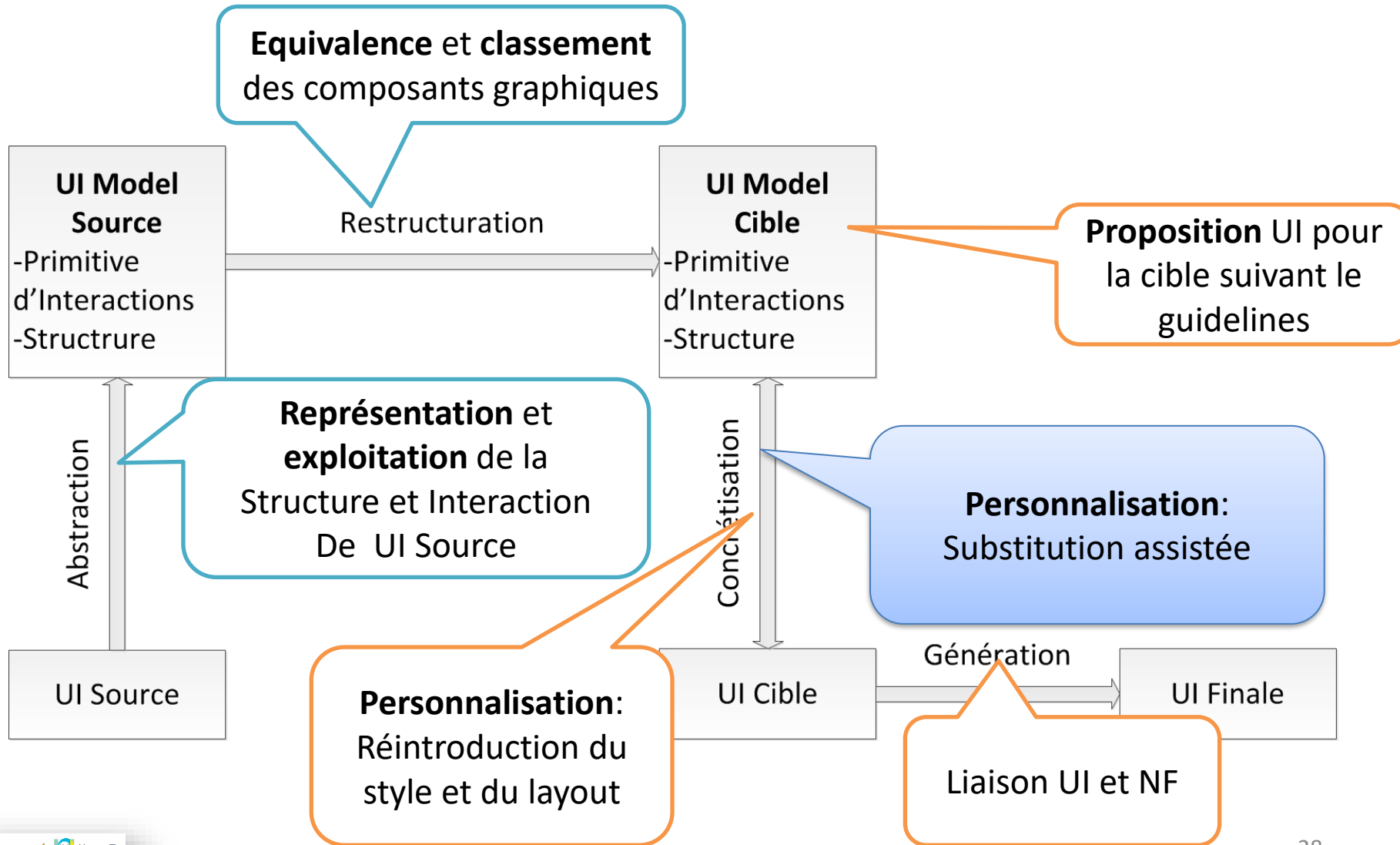


Library
Container

Sommaire

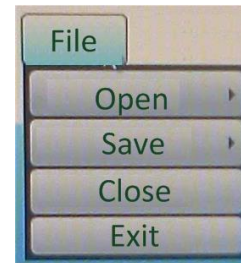
- Etat de l'art
- Concepts au cœur de la migration
- **Processus de migration**
- Expérimentations
- Conclusion et Perspectives

Processus de migration



Substitution du menu

Equivalences des **menus**
(Guideline 22- Menu)



1^{er}

SurfaceMenu

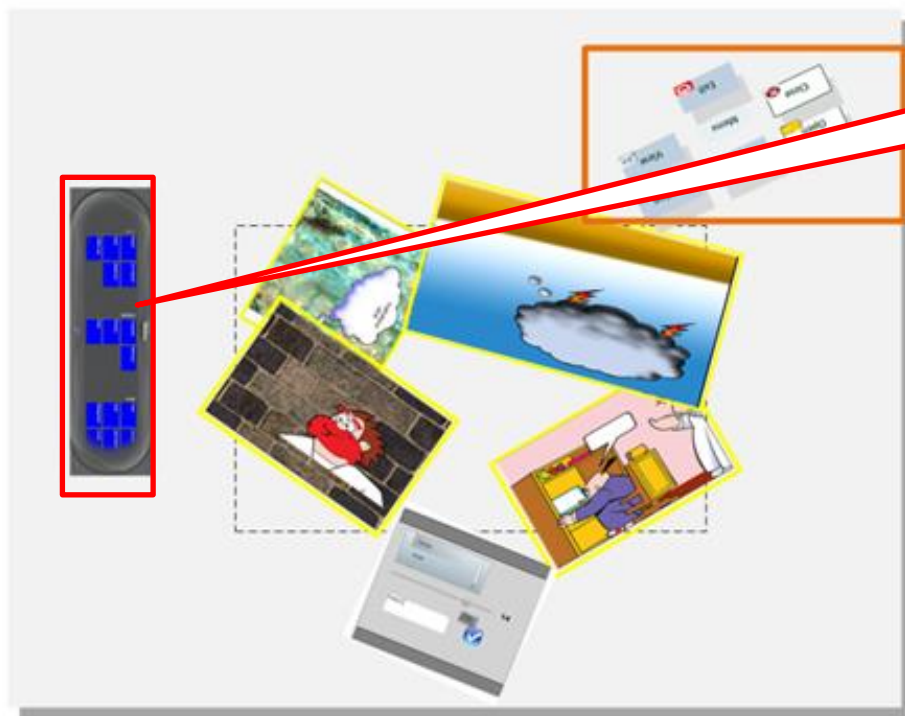


2^{ème}

Surface
ListBox

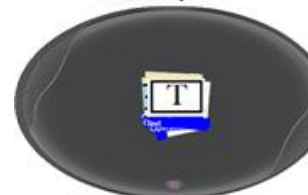
Substitution d'une liste

Equivalences des **Listes non modifiables**
(Guideline 23-Liste)



1^{er}

LibraryBar



2^{ème}

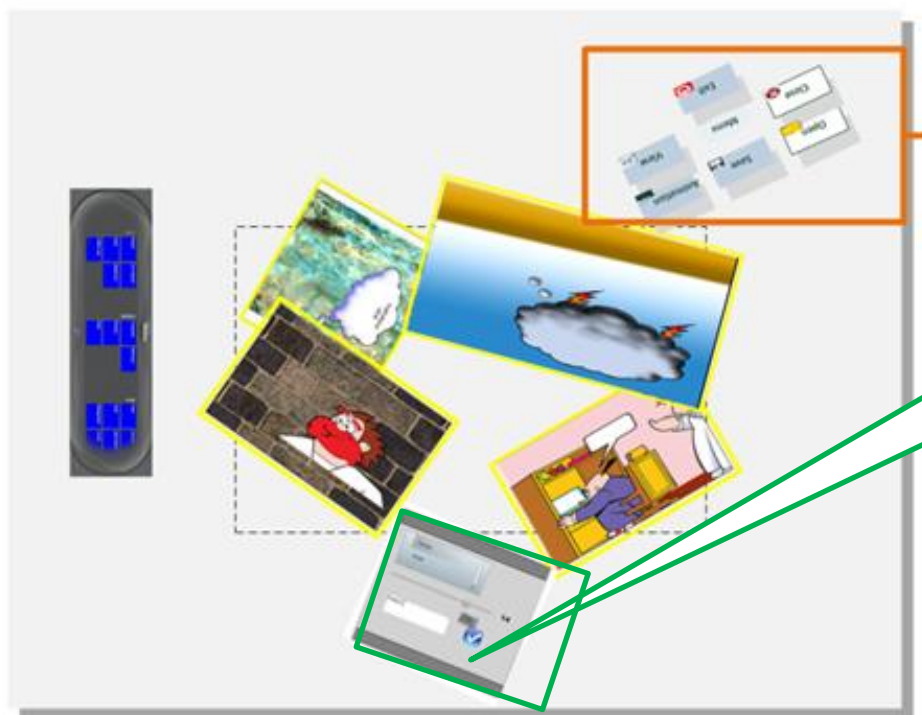
LibraryStack



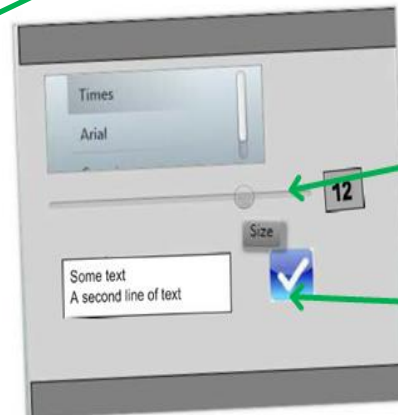
3^{ème}

ScatterView

Personnalisation formulaire



Equivalences des
composants graphiques avec
des **contenus modifiables**
(Guideline 24-Formulaire)



SurfaceTextBox

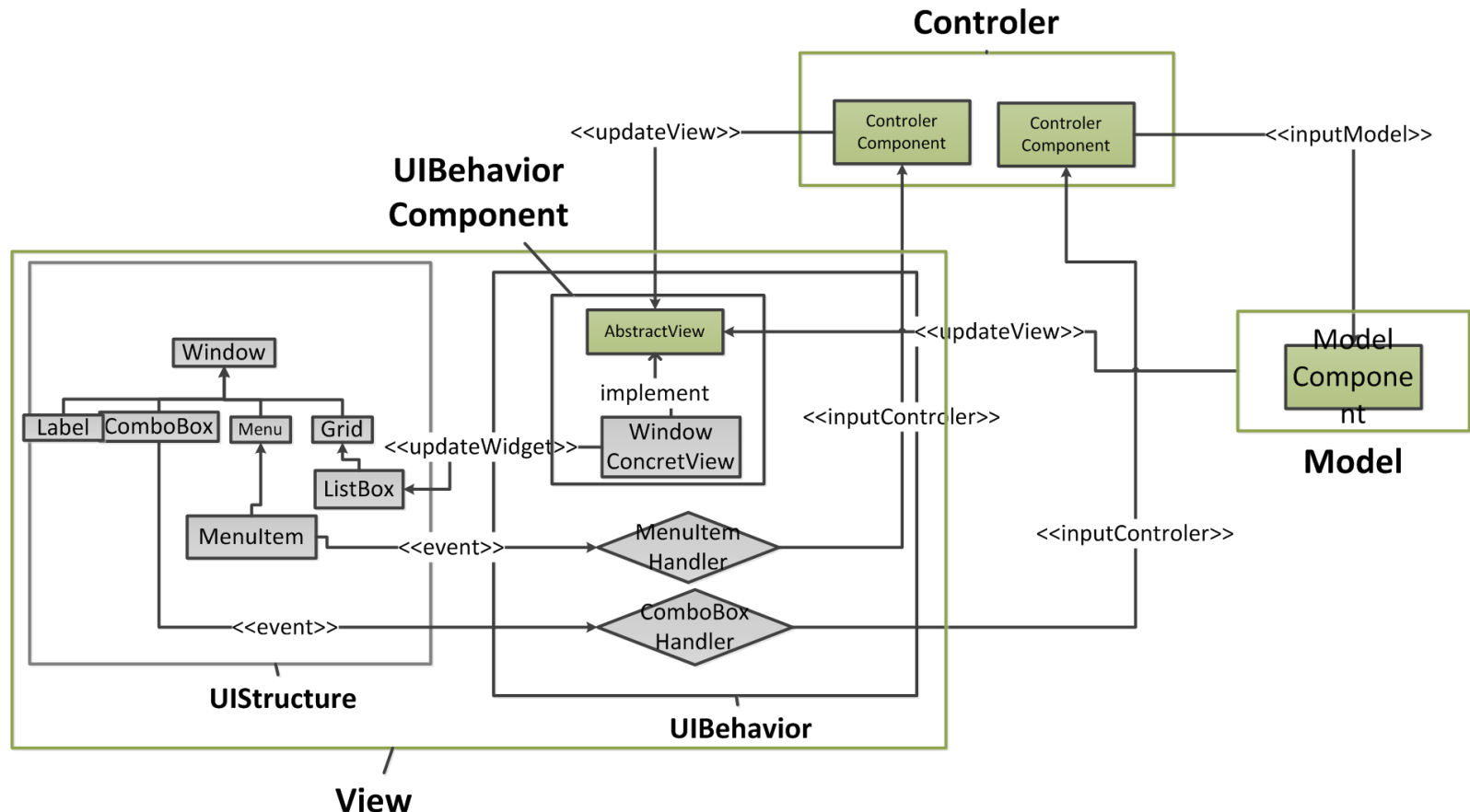
Apply

Sommaire

- Etat de l'art
- Concepts pour la migration
- Processus de migration
- **Expérimentations**
- Conclusion et Perspectives

Implémentations

- Architecture MVC des applications à migrer



Implémentations (suite)

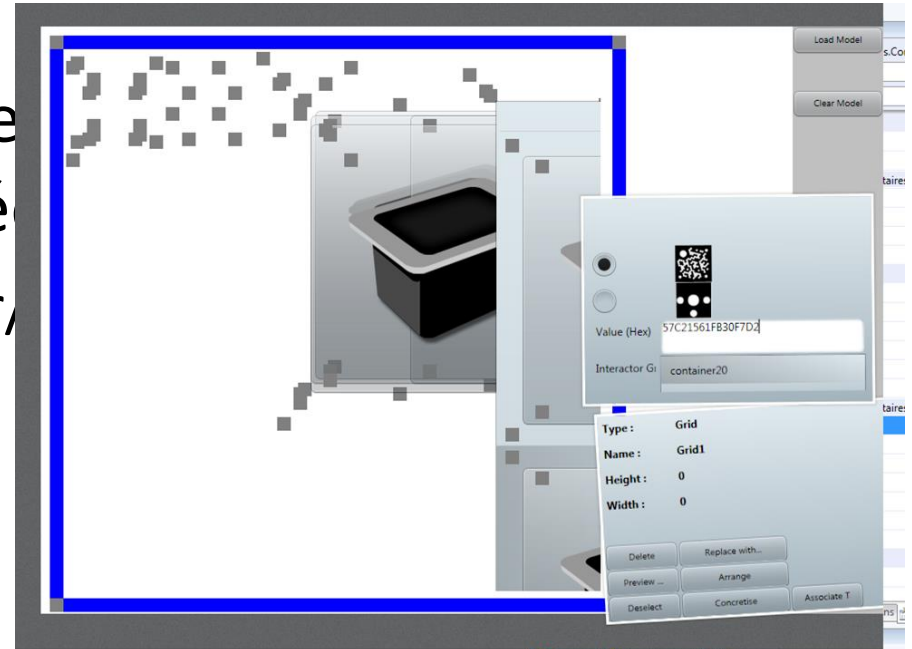
- Abstraction / Concrétisation
 - EMF pour modéliser des composants graphiques et les primitives d'interactions (PI)
 - Abstraction : Identifier les PI de chaque composants
 - Type
 - Instance
 - Difficultés identifiées les PI des instances des composants graphiques XAML
 - Tables d'indentifications des PI à partir des Handlers et des Propriétés

Implémentations (suite)

- Proposition d'une UI
 - Java/EMF pour Substituer les Containers et les UIComponents
 - Règles de substitution en considérant les guidelines
 - Difficulté pour identifier les containers à substituer
 - **+englobant**: moins accessible à plusieurs utilisateurs
 - **-englobant**: plus d'éléments graphiques
 - 4 types de containers identifiés à l'aide des PI

Implémentations (suite)

- Personnalisation
 - Editeur graphique sur PixelSense
 - Associer un objet Tag à un composant graphique
 - Modifier l'UI proposer
 - UI Modèle Cible cohérent éditeur et personnalisé
 - Design pattern Observer,



Implémentations (fin)

- Génération
 - Fichier XAML Cible
 - Lien UI-NF Manuel

Evaluation

- Pertinence des transformations
 - Regroupement des éléments graphiques
 - Atteignabilité des groupes
 - Utilisation des objet Tangibles
- Quatre applications
- Résultats
 - les transformations sont pertinentes et utiles pour $\frac{3}{4}$ des applications

Sommaire

- Etat de l'art
- Concepts pour la migration
- Processus de migration
- Expérimentations
- Conclusion et Perspectives

Conclusion

- Atelier d'aide à la migration des UI vers les tables interactives
 - Choix des équivalences pendant la migration
 - Prise en compte des principes de conceptions

Conclusion

- Proposition des guidelines pour la migration des UI desktop vers les tables interactives
- Opérateurs d'équivalences et de sélection des composants graphique basées sur des guidelines

Perspectives

- Court terme
 - Personnalisation
 - Réintroduction automatisée du layout et du style
 - Guidelines liées au style
 - UI Collaborative
 - Tâches Multi utilisateurs
- Long terme
 - Généralisation de l'approche
 - Autres plateformes de départ
 - Une autre plateforme d'arrivée

