

De la règle à l'usage : la maquette numérique de construction comme support de contrôle réglementaire automatisé - Accessibilité des handicapés

Alain ZARLI- alain.zarli@cstb.fr

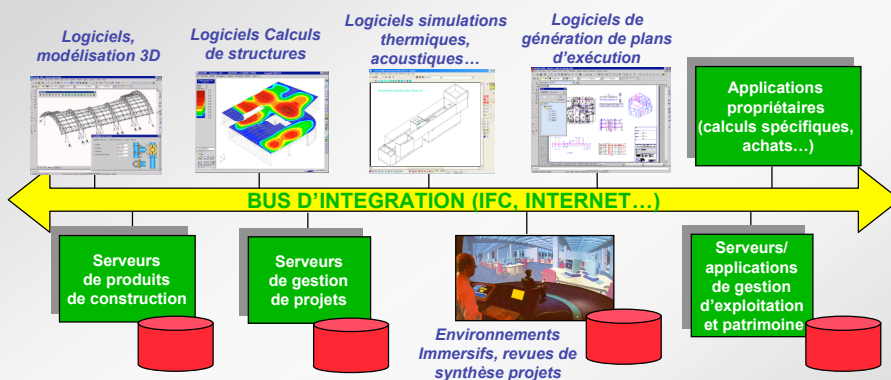
& Rémi VANKEISBELCK

Centre Scientifique et Technique du Bâtiment – CSTB

Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 1

La Maquette Numérique: vers une nouvelle chaîne d'applications « interopérables »



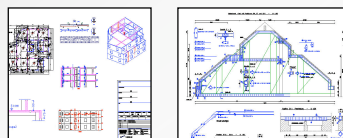
- De la production des produits de construction à l'exploitation des ouvrages, en passant par la conception, la simulation et le chantier, les logiciels commencent à communiquer autour d'un standard universel de maquette numérique : les IFC

Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 2

Contrôle réglementaire automatisé sur maquette numérique de bâtiment: contexte

- On dispose de plans d'architectes exportés depuis les outils de CAO
- On a accès à la réglementation (au moins les documents et normes relatifs à l'accessibilité)
- Ne peut-on pas automatiser tout ou partie du processus de vérification de la réglementation sur nos bâtiments ?

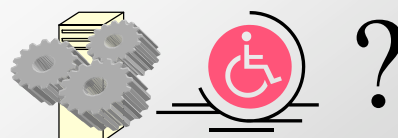


article L. 111-1

(Loi n° 86-13 du 6 janvier 1986, art. 1) «Ainsi qu'il est dit à l'article L. 421-1 du code de l'urbanisme et sous réserve des disp

article L. 111-3

de f Conformément à l'article L. 111-6 du code de l'urbanisme, les bâtiments, locaux et installations soumis aux dispositions des articles L. 111-1, L. 421-1 ou L. 510-1 dudit code ne peuvent, nonobstant toutes clauses contraires du cahier des charges, de concession, d'affermage ou de régie intéressée, être raccordés définitivement aux réseaux d'électricité, Le n de gaz ou de téléphone, si leur construction ou leur transformation n'a pas été, selon le cas, autorisée ou agréée en vertu des articles précités.



OUI !

Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 3

Plan de la présentation

- 1 - Réglementation pour l'accessibilité
 - 2 - La maquette numérique et les IFC pour le contrôle de l'accessibilité des bâtiments aux personnes handicapées
 - 3 - Algorithmes de vérification
 - 4 - Vue globale du processus
 - 5 - Exemple d'implémentation : le « *Code Checking Service* » de la plate-forme *ISTforCE*
 - 6 – Conclusions
- Une vision globale du processus de vérification, depuis l'élaboration des modèles de bâtiments, jusqu'à la génération de rapports de conformité, en passant par toutes les étapes intermédiaires

Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 4

1 - Réglementation pour l'accessibilité (1/5)

- **Source(s), format(s), contenu**

- Diffusé par les organismes gouvernementaux (AFNOR, etc.)
- Normes, législatif, articles, etc. (texte brut, schémas, images, etc.)
- Documents et outils annexes (tutoriels, CD-multimedia, etc.)
- Format papier ou « numérisé » (SGML, XML, etc.)
- Ex : « [...] les portes intérieures du bâtiment doivent être d'une largeur de passage d'au moins 90cm [...] »

- **Exploitation: constat**

- Inexploitable dans l'état !
- L'information est fragmentée dans divers endroits du corpus, avec parfois une redondance ou une dépendance
- Le format « texte » induit forcément une intervention humaine pour structurer les règles et programmer les algorithmes de vérification
- Travail de compilation des documents, de structuration, etc.

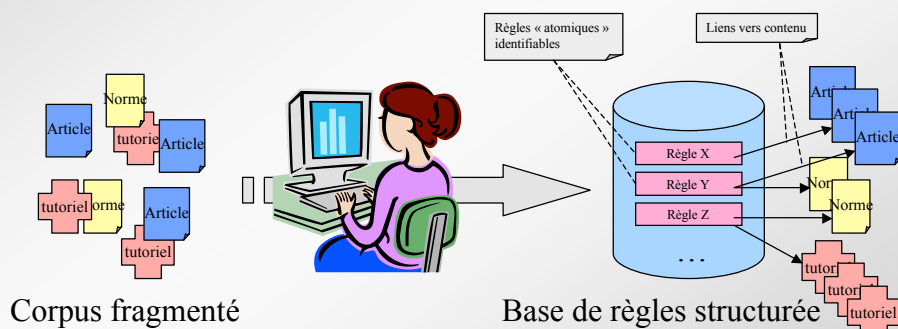
1 - Réglementation pour l'accessibilité (2/5)

- **Exploitation**

- Regroupement de l'information, définition de **règles** « atomiques » sans recouvrement
- Affectation d'**identifiants** uniques pour chaque règle
- Résumé de la règle dans un **texte bref** et explicite pour le lecteur
- **Liens** vers la **documentation officielle** attachée à chaque règle
- **Liens** vers les **tutoriels** et autres outils si on en dispose

1 - Réglementation pour l'accessibilité (3/5)

• Exploitation



Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 7

1 - Réglementation pour l'accessibilité (4/5)

• Attacher de l'intelligence aux règles : les algorithmes de vérification

- Le texte doit être transformé en algorithme
- Intervention humaine obligatoire (impossibilité pour une machine de comprendre le français !)
- Toutefois, il est possible, une fois les règles structurées, d'y associer de l'intelligence

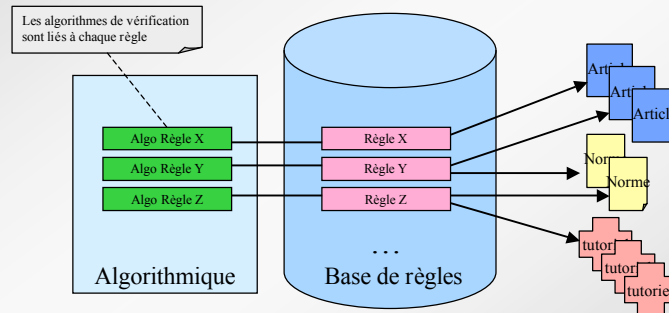
```
pour toutes les portes du bâtiment {
  si ((porte est interieure) et (largeur de porte < 90cm)) alors
  {
    renvoyer une anomalie
  }
}
```

Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 8

1 - Réglementation pour l'accessibilité (5/5)

- **Attacher de l'intelligence aux règles : les algorithmes de vérification**



- **Problèmes principaux**

- Dispersion de l'information (articles, normes, etc.)
 - Quantité de l'information, rapport signal/bruit minuscule...
- => *Travail de compilation, de structuration parfois lourd*

Workshop CRAI - Nancy - CSTB / Alain ZARLI

16 décembre 2004 - 9

2 - La maquette numérique et les IFC pour le contrôle de l'accessibilité (1/4)

- **Entités principales du schéma**

- 2 catégories majeures dans l'information :
 - Géométrie (généralement couverte complètement par le modèle et les outils CAO)
 - IfcSpace, IfcProduct, IfcOpening, IfcSlab, IfcShapeRepresentation, etc.
 - Sémantique (classification, méta-description, etc. : couverts en partie par le modèle, mais c'est en amont, au niveau des outils, qu'il devient plus difficile de renseigner l'info)
 - IfcClassification ? PropertySets ?
- ⇒ *Pas de vrai « standard » à ce niveau. On doit définir des conventions « propriétaires »*
- ⇒ Ex: comment savoir si tel espace est une cuisine ou une chambre à coucher ?

Workshop CRAI - Nancy - CSTB / Alain ZARLI

16 décembre 2004 - 10

2 – La maquette numérique et les IFC pour le contrôle de l'accessibilité (2/4)

• Extraction et « pre processing » de l'information : le Format Pivot

- Extraction des données : requiert une bonne connaissance des IFC
- Même si les concepts d'accès et de manipulation du modèle sont simples, la complexité interne du modèle IFC induit forcément un coût au niveau de l'accès et de la manipulation des instances
- Les standards sont aussi sujets au changement (changements de version des IFC)
- Utiliser les IFC directement dans l'implémentation des algorithmes de vérification n'est peut-être pas une si bonne idée...
- Dans le génie logiciel, le principe de séparation ne peut pas être mauvais...

=> Comment peut-on s'y prendre, en plus d'être indépendant des éditeurs de logiciels CAO, pour être indépendant des IFC ?

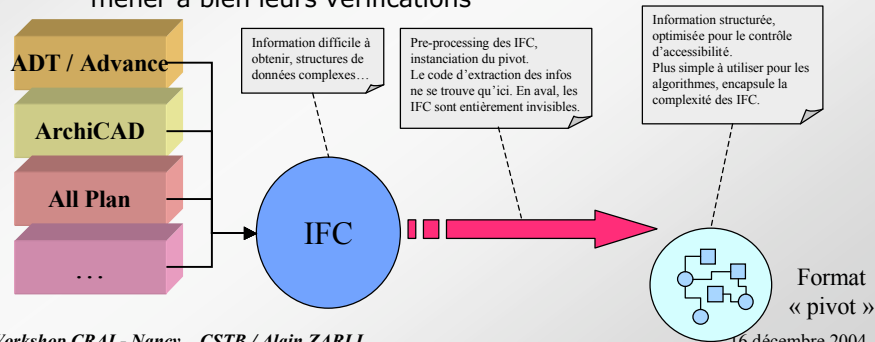
Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 11

2 – La maquette numérique et les IFC pour le contrôle de l'accessibilité (3/4)

• Extraction et « pre processing » de l'information : le Format Pivot

- Il faut passer par un *format pivot* !
- Ce pivot doit contenir l'information dont on a besoin pour l'accessibilité - on élimine tout ce qui est « inutile » du modèle IFC)
- Les algorithmes doivent se baser sur ce format pivot pour mener à bien leurs vérifications



Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 12

2 – La maquette numérique et les IFC pour le contrôle de l'accessibilité (4/4)

- **Extraction et « pre-processing » de l'information : le Format Pivot**

- Les avantages sont nombreux :
 - Centralisation de la manipulation des IFC dans des « extracteurs » propres à chaque version
 - Possibilité de supporter virtuellement tous les formats
 - Pre-processing des données « disparates » dans le modèle IFC, vue simplifiée du modèle, dédiée à notre problématique
 - Algorithmique simplifiée, on se concentre sur la problématique de vérification et non sur la manipulation des données

3 – Algorithmes de vérification (1/3)

- **Lien avec la réglementation**

- On va « attacher » les algorithmes à chaque règle :
 - Règle XXX :

[...] les portes intérieures du bâtiment doivent être d'une largeur de passage d'au moins 90cm [...]

- Algorithme :

```
pour toutes les portes du bâtiment {  
  si ((porte est interieure) et (largeur de porte < 90cm)) alors {  
    renvoyer une anomalie  
  }  
}
```

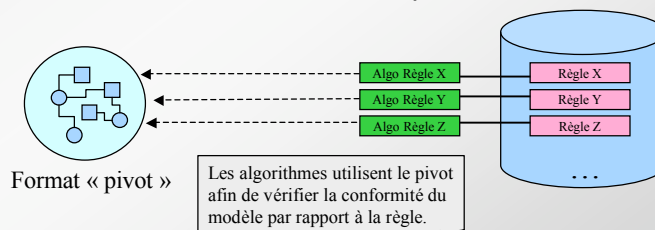
3 – Algorithmes de vérification (2/3)

- **Utilisation du pivot**

- Algorithme :

```
pour toutes les portes du bâtiment {
  si ((porte est interieure) et (largeur de porte <
    90cm)) alors {
    renvoyer une anomalie
  }
}
```

- Utilisation des objets provenant du pivot et accès à leurs « attributs » - via l'API du pivot



3 – Algorithmes de vérification (3/3)

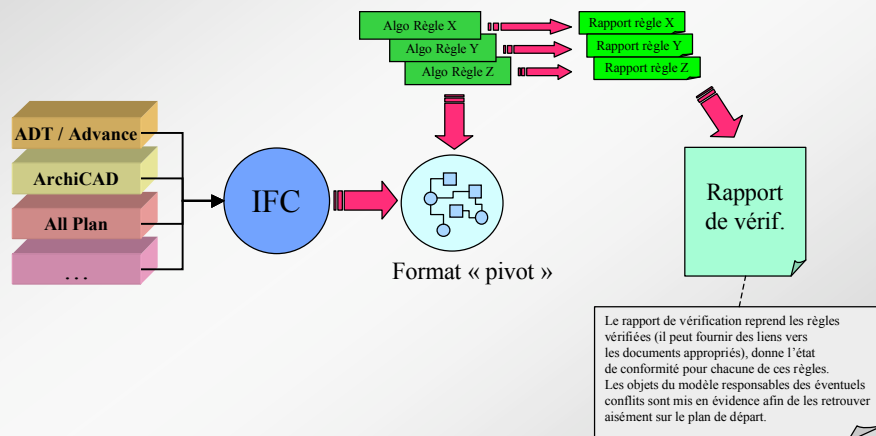
- **Exécution des algorithmes**

- Une fois les algorithmes liés aux règles, le jeu consiste à « exécuter » ces algorithmes sur base d'un format pivot donné
- L'exécution de chaque algorithme génère un rapport individuel, par exemple :
 - Vérifié OK
 - Vérifié partiellement (+raison)
 - Non vérifié (+objets responsables de la non-conformité)
 - Non vérifiable (+raison)
- La concaténation de ces rapports individuels constitue le rapport final de vérification pour le bâtiment

- **Problématiques majeures**

- La plupart des algorithmes sont très simples (e.g. portes)
- Toutefois, une attention particulière est à porter aux calculs de chemins dans les espaces
- Théories de graphes, diagrammes de Voronoï, etc.

4 – Vue globale du processus



Workshop CRAI - Nancy – CSTB / Alain ZARLI

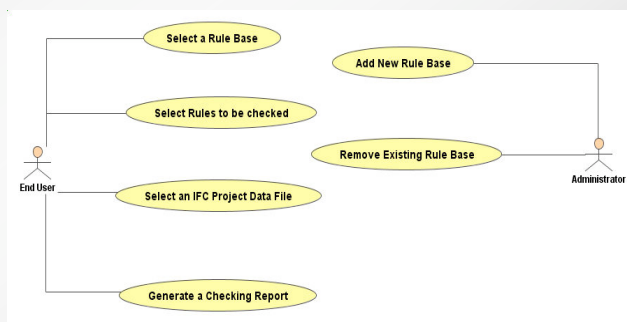
16 décembre 2004 - 17

5 – Le « Code Checking Service » de la plate-forme ISTforCE (1/6)

• Présentation du service

- Vérification de réglementations sur modèles IFC
- Exemple : accessibilité des handicapés aux Etablissements Recevant du Public
- Service offert au travers d'une plate-forme d'ingénierie collaborative
- Web-based : tout doit être accessible via un browser Web

ISTforCE
IST 11508
(5ème PCRD)



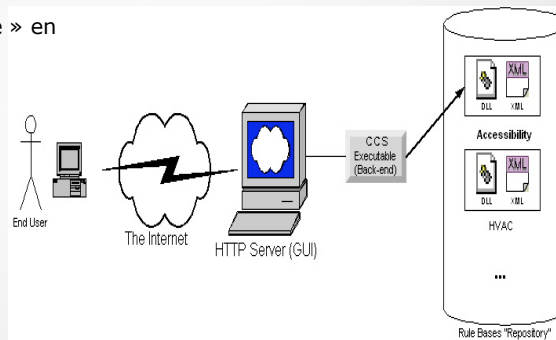
Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 18

5 – Le « Code Checking Service » de la plate-forme ISTforCE (2/6)

• Architecture

- Back-end :
 - Bases de règles « pluggables »
 - Règles XML, refs vers contenu HTML
 - Pivot, extraction et Algorithmes embarqués dans des DLL
 - Rapport XML
 - Le tout « packagé » en exécutable Win32
- Front-end
 - J2EE web app
 - XSL, JSP

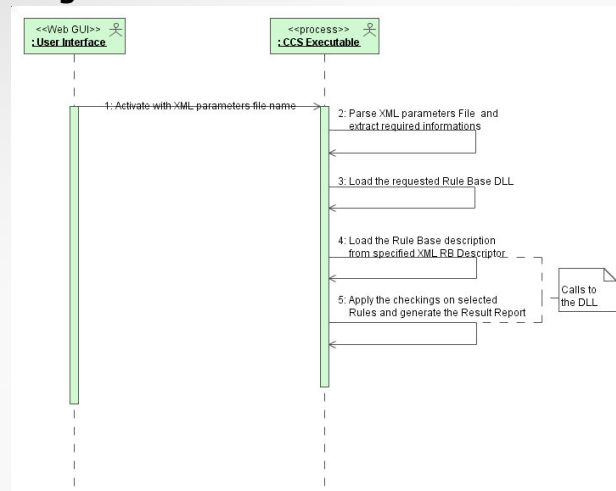


Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 19

5 – Le « Code Checking Service » de la plate-forme ISTforCE (3/6)

• Scénario global



Workshop CRAI - Nancy – CSTB / Alain ZARLI

16 décembre 2004 - 20

5 – Le « Code Checking Service » de la plate-forme ISTforCE (4/6)

• Résultats

- Le prototype prouve le concept
 - L'extraction des données IFC est réalisable
 - Il est possible d'automatiser en partie la vérification de la réglementation accessibilité sur des IFC
 - Les règles « basiques » (taille des portes, etc) ainsi que les circulations, plus complexes, (chemins, évitements d'objets, etc) ont été implémentées
- Toutefois, certains « problèmes » sont apparus, pour la plupart liés aux processus métier de la construction eux-mêmes !
 - Manque caractérisé d'informations dans les plans d'architectes
 - Mauvaise utilisation des outils récents
 - Pas ou peu d'approche « objet » dans les plans... les outils le permettent pourtant !
 - Problèmes liés aux classifications (comment identifier une cuisine de façon standardisée ?)
 - Bref, la couverture de la base de règles « accessibilité » est très faible, à cause du manque d'information dans les plans d'architectes

5 – Le « Code Checking Service » de la plate-forme ISTforCE (5/6)

• Résultats (suite)

Table 1. couverture de la base de règles (estimation)

	Rules count	Percent covered
Full checking	34	21
Partial checking	11	6
Not checkable	117	72
Total	162	100

- **Full Checking** : la règle peut être vérifiée entièrement, le modèle IFC, les outils CAO et les pratiques architecturales « usuelles » le permettent ;
- **Partial Checking** : la règle ne peut être vérifiée que partiellement, du à un manque d'informations dans les IFC, les exports des outils CAO ou les pratiques architecturales ;
- **Not checkable** : la règle ne peut pas être vérifiée ! Le modèle IFC ne contient pas l'info (très rare) ou ne donne pas de moyen « standard » d'y parvenir (classifications), les exports CAO sont trop « pauvres » en information, ou encore les pratiques des utilisateurs ne sont pas adaptées ;

5 – Le « Code Checking Service » de la plate-forme ISTforCE (6/6)

- **Conclusions**

- C'est faisable (finalement, ce n'est ni plus ni moins qu'une simulation particulière !)
- Mis à part l'extraction des données IFC qui nécessite une certaine connaissance du modèle, le projet est facilement maîtrisable par une équipe de programmeurs
- On peut imaginer vérifier tout autre type de « contrainte » sur le bâtiment de la même façon
- Un bémol tout de même en ce qui concerne le travail de préparation des données réglementaires
 - Travail manuel d'expert !
 - Nécessite la connaissance du contenu réglementaire
 - Beaucoup d'information à traiter
- *Aussi, le travail de préparation et d'enrichissement des modèles IFC est crucial : c'est le seul et unique moyen d'arriver à une couverture convenable !*

6 – Conclusions

- **Etapes principales**

- **Structuration** de la **réglementation**
- **Enrichissement des modèles** exportés par les outils CAO
- **Définition** d'un **format pivot** adapté, **extraction** des données depuis les **IFC**
- Implémentations d'**algorithmes** (qui vont manipuler des instances de pivot), liés avec les règles

- **Références**

- International Alliance for Interoperability, *IFC*
 - <http://www.iai-international.org>
- Projet *ISTforCE* - wp4000 : *Code Checking Service*
 - <http://www.istforce.com>