

Remerciements

un grand merci ...

A **Jean-claude Bignon** et **Gilles Halin** pour m'avoir suivi durant cette étude et m'avoir orientée dans le développement de ce sujet.

Aux enseignants de DEA pour leur investissement et leur capacité à communiquer leur savoirs comme leurs passions.

A l'équipe du CRAI en général qui a été d'un réel appui dans certains moments d'incertitudes, plus particulièrement **Sylvain Kubicki**.

Remercie **M. Pierre**, dépense trop de temps à corriger mon pauvre français.

sommaire

REMERCIEMENTS	1
SOMMAIRE	2
INTRODUCTION.....	4
PROBLÉMATIQUE	5
1. PRINCIPES FONDAMENTAUX DES LOGICIELS DE PLANIFICATION EXISTANTS	6
1.1. OPTIMISER PAR LE TEMPS.....	7
1.2. OPTIMISER PAR LES RESSOURCES	8
1.3. GÉRER LES CONTRAINTES	9
1.4. VISUALISER LE COÛT.....	10
1.5. GÉRER LES RISQUES	10
1.6. BILAN SUR LES LOGICIELS DE PLANIFICATION ACTUELS	11
2. MÉTHODE « »4D CAD» » ET LOGICIELS EXISTANTS	13
2.1. PRÉSENTATION.....	13
2.2. CONCEPTS FONDAMENTAUX DES MÉTHODES 4D.....	14
2.2.1. <i>Le composant « maquette 3D »</i>	14
2.2.2. <i>Le composant « planning »</i>	15
2.2.3. <i>Le composant « 4D CAD»</i>	17
2.3. FONCTIONNALITÉS DES LOGICIELS 4D EXISTANTS.....	19
2.3.1. <i>Indiquer l'évolution de la construction</i>	19

2.3.2.	<i>Analyser les composants du planning</i>	19
2.4.	BILAN.....	20
2.4.1.	<i>Avantage et limite des logiciels «4D CAD»</i>	20
2.4.2.	<i>Amélioration possible des logiciels «4D CAD»</i>	21
3.	HYPOTHÈSES ET PERSPECTIVES	22
3.1.	LES ÉLÉMENTS D'IFC CORRESPONDANT LE PLANNING.....	23
3.1.1.	<i>le « temps »</i>	23
3.1.2.	<i>les « ressources »</i>	24
3.1.3.	<i>Le « coût »</i>	25
3.1.4.	<i>Les « contraintes »</i>	25
3.1.5.	<i>bilan</i>	26
3.2.	LA DESCRIPTION DES ÉLÉMENTS DU PLANNING PAR IFC.....	26
3.2.1.	<i>Préparer un fichier d'IFC d'un objet simple</i>	27
3.2.2.	<i>Ajouter les informations de planification</i>	28
3.2.3.	<i>bilan</i>	31
3.3.	LA DESCRIPTION DES ÉLÉMENTS DE 4D PAR LES IFC.....	32
3.3.1.	<i>Préparer un fichier d'IFC de 3 objets</i>	32
3.3.2.	<i>Ajouter les informations temporelle</i>	33
3.3.3.	<i>bilan</i>	37
4.	CONCLUSION	39
	BIBLIOGRAPHIES	40
	ANNEXES	41
	LE FICHIER D'IFC D'UN MUR AVEC LES INFORMATIONS DE PLANIFICATION.....	41
	LE FICHIER D'IFC DE TROIS MURS AVEC LES INFORMATIONS DE PLANIFICATION.....	44

Introduction

Les récentes recherches dans le domaine de l'informatique offrent la possibilité de l'utilisation d'outils informatiques dans le domaine de l'Architecture, du Génie civil, et de la construction. Ces recherches s'intéressent normalement au standard de description architectural international IFC (Industry Foundation Classes) ainsi qu'à la visualisation du planning par l'intermédiaire d'une maquette virtuelle 3D.

La spécification IFC est fondée en 1997 par l'organisation IAI. Elle a pour objectif l'interopérabilité et la coordination des informations durant tout le cycle de la vie du bâtiment. Cette norme propose un format de données spécifique au domaine architectural pour transférer les documents entre différents types de logiciels.

La visualisation de la planification est un champ de recherche récent, on l'appelle aussi « 4D CAD ». La méthode consiste à ajouter le temps comme « 4e dimension » à la maquette 3D virtuelle que l'on a créé avec un outil de CAO. Tous les participants du projet peuvent suivre l'évolution de la construction. De plus, s'il y a un changement du planning (principal ou partiel), les acteurs peuvent le vérifier et le représenter graphiquement dans une fenêtre de visualisation 4D.

Nous nous intéresserons donc ceux logiciels de planning classique et de 4D, afin d'analyser leurs avantages et leurs inconvénients. Ensuite, nous chercherons la possibilité de décrire les éléments de planification et de « 4D CAD » grâce aux IFC.

Problématique

Actuellement, il y a de nombreux plannings dans les activités de construction. Pour le coordinateur du chantier qui contrôle la planification principale, l'organisation des « sous plannings » (ou plannings détaillés) est un problème courant. La mise à jour du planning, afin de suivre la réalité du chantier est complexe.

Les « sous plannings » produits par chaque intervenant doivent répondre aux contraintes du planning principal, et en même temps, ils doivent être coordonnés entre eux. Donc, un format commun pour échanger les informations de planification est vraiment utile et nécessaire. De plus, il existe de nombreux types de logiciels de planification.

La visualisation de la planification se développe afin de proposer de nouvelles techniques de visualisation des objets en construction. Des méthodes ont été développées et commercialisées comme les logiciels « 4D suite » et « Project 4D ». Nous proposons faire une analyse de ces logiciels pour distinguer leurs méthodes et leurs limites actuelles.

Les IFC sont une spécification internationale pour l'interopérabilité, mais actuellement aucun logiciel de planification classique ou 4D ne supporte le format IFC. Notre travail a pour objectif de voir si les IFC ont la capacité de décrire les éléments principaux de la planification et de la méthode 4D.

Nous allons analyser les logiciels de planification et les logiciels «4D CAD» existants pour distinguer leurs modes de fonctionnement et mettre en évidence leurs limites, puis nous allons manipuler la spécification IFC, particulièrement la description de la planification. Enfin, nous allons explorer la possibilité de transférer les éléments de planification avec le format IFC et de décrire les éléments de 4D par IFC.

1. Principes fondamentaux des logiciels de planification existants

Il existe de nombreux logiciels de planification commerciale sur le marché mais chacun a ses propres avantages et inconvénients, et surtout chacun vise un type d'utilisation spéciale. Actuellement, pour le chantier et la construction, il y a également un grand nombre d'outils informatiques. Nous avons choisi quelques logiciels de planification dans le cadre de notre étude. Il s'agit de « Microsoft Project » et « Hyper Archi ». Concernant ces logiciels, nous avons analysé d'abord l'environnement informatique et les conditions d'utilisation. Ensuite, nous choisirons ces « Microsoft Project » et « Hyper Archi » à analyser avec un projet réel de laboratoire de CRAI pour chercher et manipuler les éléments fondamentaux de planification.

« Microsoft Project » est un outil informatique de planification très connu, il est pratiquement utilisé dans tous les métiers et pas seulement dans le domaine de l'architecture. Sa méthode et la façon de traitement de données influencent beaucoup d'autres logiciels de planification, on peut le trouver de temps en temps dans autre logiciel, il a même influencé les habitudes des utilisateurs de logiciel de planification aussi.

« Hyper Archi » est un nouveau logiciel qui est connu spécifiquement pour la gestion de chantier architectural : il gère analysé tout le cycle d'une construction,

du dessin au suivi de chantier. la planification est une fonctionnalité importante de ce logiciel.

1.1. Optimiser par le temps

Le « temps » est le concept le plus important et crucial dans les méthodes de planification. Logiquement, l'objectif principal d'un logiciel de planification est « d'organiser le temps efficacement avec des relations optimisées ».

Dans « Microsoft Project » et « Hyper Archi », le « temps » est un élément fondamental et obligatoire. Le « temps » est une notion abstraite qui s'exprime à travers les notions de « date » et « durée » pour exprimer

La « date » peut être définie soit comme la « date prévue » soit comme la « date réelle » que l'administrateur de chantier modifie une fois la tâche exécutée. Dès lors, le planning est changé fréquemment lorsque les tâches ne peuvent absolument pas être réalisées en suivant la planifications.

La « durée » est la même notion que la « date », si le coordinateur définit la « date de début prévu » et la « date de fin prévu ». Le logiciel calculera la « durée prévue » directement et automatiquement. En revanche, si le coordinateur donne justement la « date début prévue » et la « durée prévue », le logiciel pourra calculer la « date fini prévu » aussi. Pour la « date réelle » et la « durée réelle », le même méthode s'appliquera.

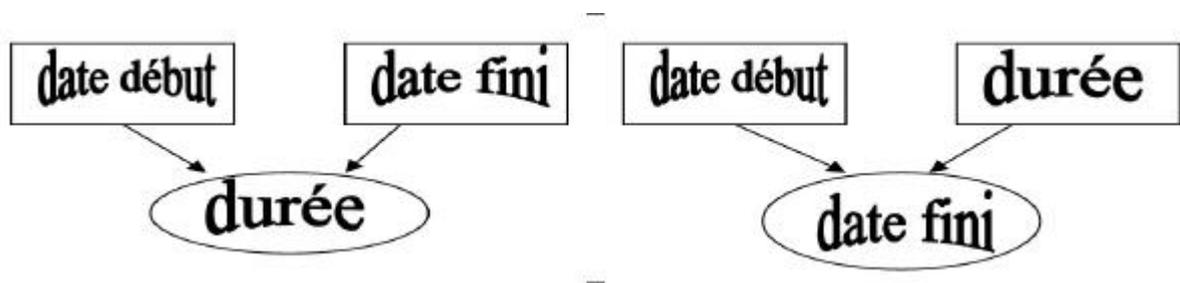


Schéma01 la relation entre les dates du « temps »

On conçoit facilement que le « temps » dans la planification est une donnée fluctuante et instable, donc la planification est influencée également. En effet, il s'agit d'un système dynamique et flexible.

1.2. Optimiser par les ressources

Les « ressources » sont les éléments auxquels le coordinateur de chantier s'intéresse le plus. Comme le « temps », cette notion est un élément crucial dans la planification. Il existe de nombreux types de « ressources », normalement les « ressources » humaines et les « ressources » en matériaux, qui sont les plus utilisées dans les logiciels.

Dans « Microsoft Project », on peut faire le planning sans utiliser la « ressource » pour autant. Pour les « ressources » humaines, le coordinateur utilise la notion de groupe qui fait un type de tâche particulier avec un prix unitaire. Concernant les « ressources » en matériaux, « MS Project » ne peut pas les définir précisément. Le principal problème est que nous ne pouvons distribuer qu'un type de « ressource » pour une tâche : soit la « ressource » humaine , soit la « ressource » en matériaux.

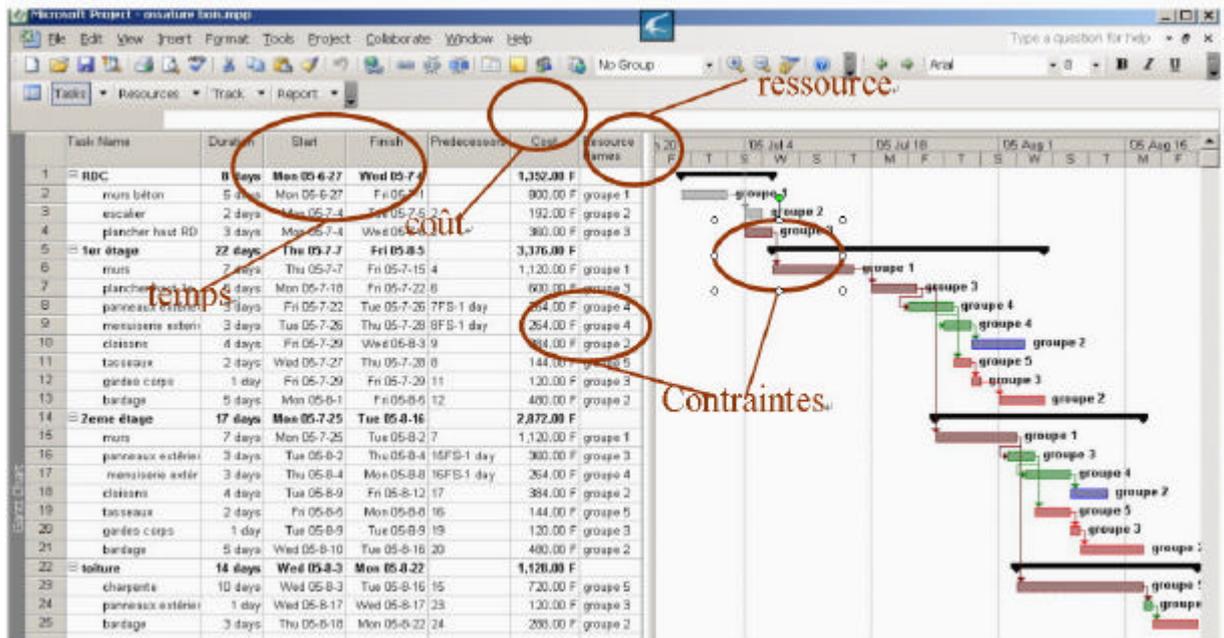


schéma02 les éléments principaux de planification

En revanche, pour «Hyper Archi », les «ressources » en matériaux sont un élément très important, on peut définir une « ressource » en matériaux en donnant le prix unitaire et la distribuer dans plusieurs tâches différentes. Par contre, une tâche peut aussi contenir plusieurs types de « ressources ».

1.3. Gérer les contraintes

Dans tous les logiciels de planification, les « contraintes » sont identiques. Il en existe 4 types : « début – début », « début – fin », « fin – début » et « fin – fin ». Elles permettent de relier les tâches temporellement.

Comme on l'a décrit au 1.1, le « temps » est le premier élément dans le logiciel de planification, et les « contraintes » justement gèrent les relations temporelles. Par ces « contraintes », on peut anticiper les problèmes d'organisation. Pour les projets de construction, il y a certaines séquences qui doivent être respectées, c'est un principe aussi pour le coordinateur. Par ailleurs, « Microsoft Project » et « Hyper Archi » donnent la possibilité de vérifier la

correction des séquences des tâches par les « contraintes » quand nous changeons le « temps » d'une tâche.

1.4. Visualiser le coût

Dans les logiciels de planification, le « coût » n'est pas un élément vraiment manipulable. L'aspect financier est le résultat de différentes contraintes manipulées dans le logiciel.

Pour « Microsoft Project », le « coût » est surtout basé sur le « temps », car supporte bien les « ressources » humaines, par contre il ne représente pas le « coût » en matériaux. Donc on ne peut pas manipuler la planification par le « coût » si le chef de chantier veut obtenir une planification qui est la plus économique.

Pour « Hyper Archi », le « coût » est calculé automatiquement par le « temps » et les « ressources ». Il contient aussi tous les éléments matériaux dont on a besoin. Le plus intéressant est que l'on pourrait analyser la planification par le « coût », pas seulement par le « temps ». On peut donc faire une planification qui n'est pas forcément la plus rapide, mais certainement la moins coûteuse. Cela est plus intéressant pour le maître d'ouvrage ou le chef de chantier.

1.5. Gérer les risques

Dans un chantier de bâtiment, le coordinateur doit gérer beaucoup d'activités. Il est normal que le planning contienne des erreurs, surtout vu le dynamisme de la planification. Ainsi, les logiciels de planification professionnels nous offrent la possibilité de prendre en compte la gestion de ce risque. Cependant l'efficacité d'un tel service dépend beaucoup du coordinateur.

« Microsoft Project » est un logiciel de la planification général, c'est-à-dire qu'il n'est pas destiné spécialement pour l'architecture. Ainsi nous pouvons gérer les risques surtout par le « temps », ou bien plus précisément, par le « contrainte du temps » pour respecter les séquences des tâches et assurer un changement d'une tâche sans affecter le projet final.

Ce logiciel propose aussi la gestion des risques pour les « ressources », afin d'éviter d'utiliser une même ressource pour plusieurs tâches en même temps. Cependant ce fonctionnement n'est pas très utile lorsqu'il ne supporte que des « ressources » simples.

« Hyper Archi » est un logiciel dédié aux métiers de l'architecture pour toutes les tâches de gestion de projet architectural, on doit forcément respecter la spécification officielle, et on peut définir la spécification particulière pour certaines tâches. En revanche, il supporte la notion « lot » contrairement à « Microsoft Project ». En raison du « lot », nous pouvons bien définir une série de tâches par une séquence unique avec les « ressources » unitaires nécessaires. Donc pendant le travail de planification ou d'actualisation de la planification, on peut anticiper les risques fréquents.

1.6. Bilan sur les logiciels de planification actuels

Dans cette première partie, nous avons analysé des logiciels de planification courants, leurs méthodes et leurs fonctionnalités.

Dans le cadre de ce travail, nous avons choisi deux logiciels de planification classiques: « Microsoft Project » utilisé en tant que logiciel de planification général et « Hyper Archi », logiciel de gestion de projet architectural.

Ces deux logiciels permettent de prévoir et de contrôler le rythme d'un projet grâce au « contrôle du temps ». Les différentes contraintes temporelles permettent d'optimiser les tâches des différentes entreprises intervenant sur le chantier. Il est possible d'anticiper les ressources humaines et matérielles par « le contrôle des ressources ». Enfin certains outils proposent des méthodes pour anticiper les risques liés au planning et à l'avancement des tâches sur le chantier.

L'inconvénient majeur est qu'il est difficile de visualiser la planification. Lorsqu'il y a de nombreuses tâches planifiées, il est difficile d'indiquer la partie individuelle du travail d'un ouvrier particulier à une certaine date. De plus, le planning prévisionnel ne correspond pas exactement à la réalité du chantier qui

est dynamique et change tout le temps. Il est par conséquent difficile de comparer le planning prévisionnel et l'avancement réel des tâches de façon claire et visuelle.

2. Méthode « »4D CAD» » et logiciels existants

2.1. Présentation

Actuellement, la recherche dans le domaine des méthodes « »4D CAD» » est très importante [1] [2]. En l'espace de 20 ans, la technique CAD en trois dimensions s'est développée rapidement. On peut utiliser le model 3D pour manager les informations de la construction (via) voir sa stabilité de l'image. Mais il y a un manque d'intégration de donnée, par exemple, il n'y a pas d'interaction entre le model 3D et le planning. Le coordinateur travaille actuellement avec des plans papier ou avec les dossiers électroniques 2D, il n'a pas la possibilité de bénéficier des nouvelles techniques de 3D.

Il y a une réelle besoin de relier la planification de la construction et le model 3D, qui décrit bien la complexité du model de bâtiment. Maintenant, le model 3D contient plusieurs spécialités comme l'électricité, le pipe-line, le climatisation, etc. Il y a beaucoup de solutions qui voudrait satisfaire ce besoin, parmi ces idées géniales, «4D CAD» est l'étoile qui scintille.

Depuis 10 ans, les chercheurs ont analysé un grand nombre de méthodes possibles, et ont obtenu des résultats théoriques innovants. Williams [5] a fait un film animé 4D représentant une série d'activités séquentielles pour mieux comprendre la construction. McKinney [6] et al présente un prototype 4D outil pour que les coordinateurs puissent faire la planification avec les models 3D

manuellement. Liston et al. [7] ont développé un «4D CAD» d'aider à la décision pour identifier la zone problématique rapidement. Staub-French et al. [8] ont montré que la simulation de «4D CAD» est meilleur que Gantt charts or CPM planning. Dawood et al. [9] ont travaillé finement sur le développement de la base de données de la simulation de 4D construction.

Grâce à ces théories et ces méthodes, quelques entreprises ont déjà élaboré des logiciels «4D CAD» qui ne sont pas encore tout à fait « mûrs ». Pour mieux comprendre ces méthodes et la façon de traitement que les logiciels existants ont utilisé, nous avons choisi d'analyser « Project 4D ».

« Project 4D » est élaboré par Common Point, qui est fondé par les chercheurs de l'université Stamford. Il supporte plusieurs formats de CAD 3D et de la planification, comme Autocad, Archicad, VRML et CSV etc. Il visualise la vue de 4D par OPENGL, plus efficace et plus puissant. Nous l'avons manipulé par le model 3D du laboratoire CRAI, avec le point de vue de conception.

2.2. Concepts fondamentaux des méthodes 4D

Comme les outils de planification normaux, les outils 4D manipulent les éléments principaux comme le « temps » ou les « ressources ». Pour mieux comprendre les concepts fondamentaux des méthodes 4D, nous pouvons diviser les éléments de «4D CAD» par rapport au logiciel de planification normal. On peut distinguer trois composants essentiels : le composant « maquette 3D », le composant « planning » et un composant spécifique « 4D ».

2.2.1. Le composant « maquette 3D »

La maquette 3D est un élément essentiel des outils de «4D CAD». «Project 4D » permet d'importer les maquettes 3D Autocad ou Archicad qui sont les formats plus courants de CAD. En effet, « Project 4D » est assez intelligent pour résoudre le problème de mise à jour de version de logiciel CAD. Actuellement, le format de fichier de CAD est très complété, même les différents versions d'AutoCAD ont différents formats. « Project 4D » utilise WRL, un format commun de VRML, pour importer tous les fichiers de maquette. En plus, il délivre les outils

informatiques pour transférer la donnée CAD au WRL. En tous cas, presque tous les logiciels CAD supportent le format WRL.

Par contre « Project 4D » ne supporte pas le format IFC, une spécification que la norme internationale ISO fonde justement pour le champ d'architecte, de la construction et de la gestion de chantier. Malgré tout l'IFC peut être facilement transféré au format WRL, mais IFC est un futur format commun pour l'intégration de tous les données architecturales.

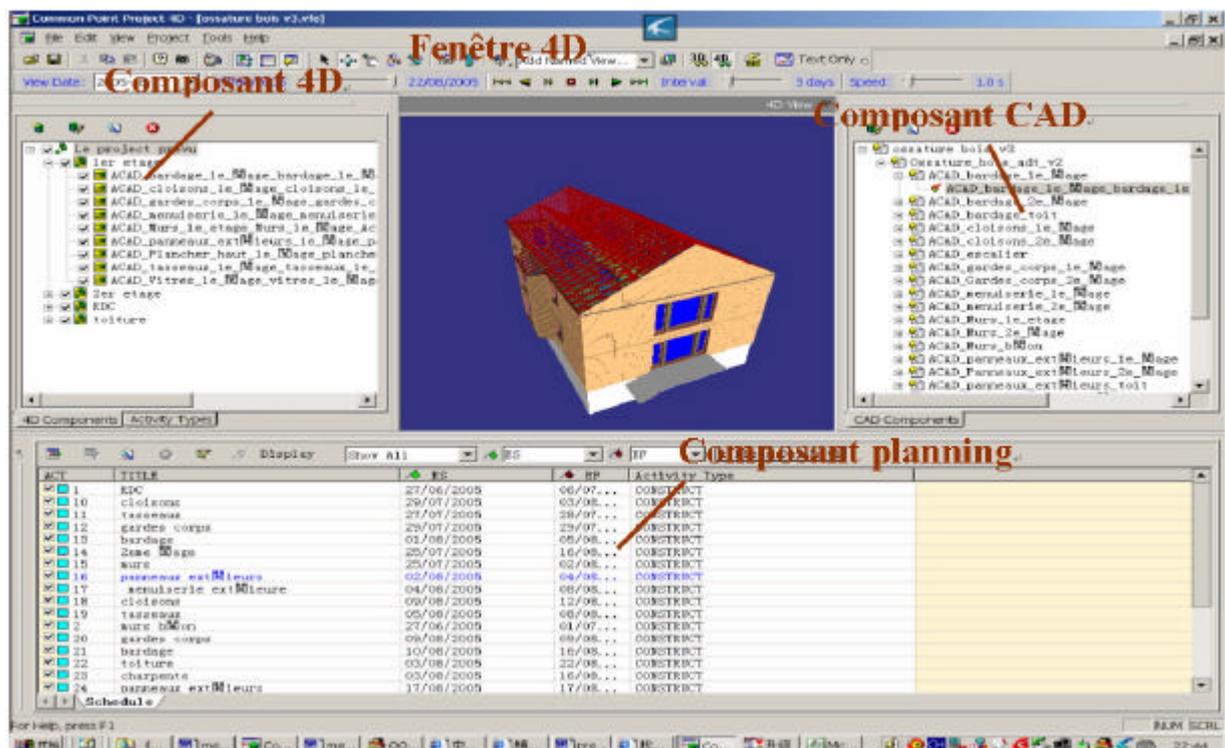
Pour « Project 4D », le composant maquette 3D se caractérise par des « couches » et pas clair à définir dans 4D. La notion de « couche » est une spécification fondamentale de CAD classique, pour distinguer et grouper les différents objets par leurs fonctionnements et leurs propriétés. Logiquement «4D CAD» hérite de cette notion qui est bien comprise et utilisée par l'utilisateur d'un outil de «4D CAD». Nous avons trouvé que ce découpage est aussi utile pour faciliter identifier et trouver un objet 3D parmi les nombreux objets du modèle.

Actuellement l'architecte s'habitue de plus en plus à la réalisation d'un modèle 3D pendant son travail. Le composant « maquette 3D » doit être rigoureusement conçu et dessiné dans le logiciel de 3D CAD afin de retrouver ses propriétés géométriques dans l'outil de 4D. Donc pour l'architecte ou le dessinateur, il faut faire un model efficace et complexe.

2.2.2. Le composant « planning »

Après avoir importé une maquette 3D, le besoin logique est d'importer les informations de planification. « Project 4D » que nous avons manipulé, supportent les formats CSV, TXT et MPX. Comme nous avons analysé dans le premier chapitre, ces formats sont produits par presque tous les logiciels planification existants.

Par contre, les formats CSV et TXT couramment utilisés dans les logiciels de traitement de données. En effet nous ne pouvons pas seulement utiliser les données des logiciels planification, mais il est nécessaire d'utiliser les autres données de n'importe quels logiciels comme EXCEL, WORD, même HTML ou Email, ceux qui supportent d'exporter le format CSV ou TXT.



Les Schéma 05 concepts fondamentaux de «4D CAD» (Project 4D)

fichiers de planning classiques contiennent de nombreuses informations, mais nous avons surtout besoin du « temps » qui est le 4^e paramètre. « Project 4D » connaît la date du début réel et la date de fin réel en même temps. Donc dans « Project 4D », nous pouvons donc regarder et comparer l'état du construction dans deux vues : prévu et réel. C'est une solution au problème de la dynamique du planning.

Dans « Project 4D », les « contraintes » et les « ressources » ne sont pas utilisés. Comme nous l'avons analysé, la « Gestion des contraintes » est un

fonctionnement crucial, spécialement pour « gérer les risques » de temps. Lorsqu'il n'y a pas de « contraintes » dans le «4D CAD», l'utilisateur de «4D CAD» ne peut pas changer le planning pour éviter le risque de conflits. S'il y a des problèmes de planning, que nous trouvons dans «Project 4D » grâce à la technique 4D, il nous faut revenir le logiciel planification et traiter les données planification, et puis réexporter le fichier à « Projet 4D », ce qui fait perdre beaucoup de temps.

Les « ressources » actuellement n'affectent pas le model 4D simple, donc ils ne sont pas très importants quand ils ne peuvent pas être reçu par «Project 4D ». Nous avons cependant discuté dans le premier chapitre qu'il est intéressant pour le maître d'ouvrage ou le chef de chantier d'avoir un planning plus économique. Pour l'instant, « Project 4D » n'a pas la possibilité de réaliser cette mission.

2.2.3. Le composant « 4D CAD »

Le composant «4D CAD» est l'élément principal de la méthode «4D CAD». Il présente l'évolution de la construction en relation avec le temps. Il est défini par l'utilisateur du logiciel 4D et se compose de deux choses : le composant CAD et le composant planification. Les trois types de composants sont liés entre eux, ce qui permet d'obtenir l'information de l'un à partir d'un autre. La relation est représenté par le schéma au-dessous.

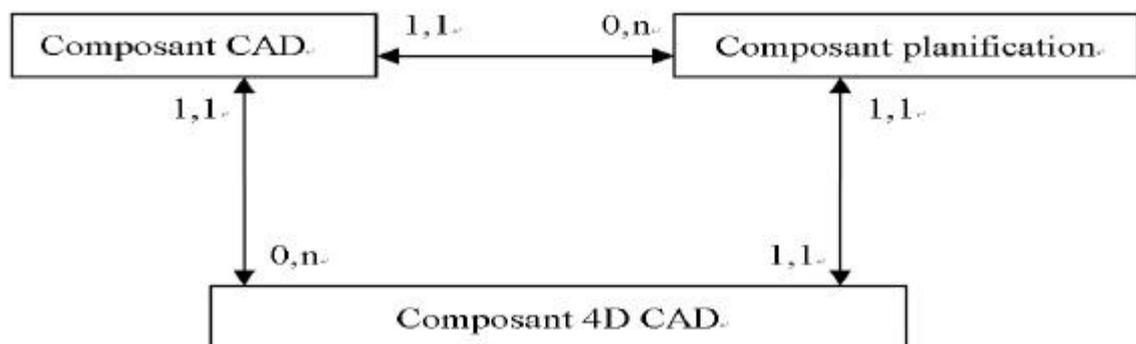


schéma 03 la relation entre les composants de «4D CAD»

A partir de ce schéma, nous pouvons trouver que le composant est obligatoirement relié avec un composant «4D CAD» et un composant de planification. Par contre, un composant planification peut être relié avec plusieurs composants CAD, ou bien il ne relie aucun composant CAD, mais il faut relier un composant «4D CAD». Pour le composant «4D CAD», la relation est plus complète s'il peut relier avec zéro ou plusieurs éléments CAD, mais seulement avec un élément planification.

Normalement, le nombre de composant «4D CAD» équivaut le nombre de composant CAD, et est moins nombreux que le nombre de composant planning parce qu'il y a beaucoup de tâches comme par exemple ceux qui ne travaillent pas sur le bâtiment directement, ou préparer le béton, le déblai etc. En revanche, il y a plein de tâches fonctionnent sur un même élément. par exemple, fabrication de charpente, transport de charpente, installation de charpente, la peinture de charpente, toutes ces tâches sont portées par la charpente, lorsqu'on a analysé la relation entre trois composants, un composant CAD peut seulement correspondre un seul composant 4D et un seul composant planification. Pour résoudre ce grand conflit, nous devons soit copier plusieurs éléments CAD soit annexer plusieurs tâches à un « lot ».

	Composant «4D CAD»	Composant CAD	Composant planification
Composant «4D CAD»	/	Zéro,un,plusieurs	un
Composant CAD	un	/	un
Composant planification	un	Zéro,un,plusieurs	/

Schéma 04 la réfection des composants de « 4D CAD »

2.3. Fonctionnalités des logiciels 4D existants

2.3.1. Indiquer l'évolution de la construction

La fonctionnalité la plus claire et la plus utile des logiciels «4D CAD» est d'indiquer l'évolution de la construction. Actuellement, les logiciels permettent de représenter l'évolution séquentiellement, par la ligne de temps que nous avons définie dans le logiciel de planification. Nous ne pouvons pas représenter l'évolution par le « coût » ou les « ressources » car « Project 4D » ne peut pas recevoir ces informations de planification. Par contre, pour l'évolution sur la ligne de temps, nous pouvons raffiner la finesse de semaine jusqu'à l'heure. En plus, nous pouvons inverser la ligne de temps aussi pour regarder l'évolution de de la construction.

Cette fonction semble être une démonstration inutile puisque tous les professionnels connaissent bien l'évolution du bâtiment par la planification ou par leur expérience. Il est par contre utile de montrer les tâches en interface entre les différents acteurs.

« Project 4D » a pour but de montrer le « temps prévu » et le « temps réel » pour une même construction, c'est-à-dire que nous pouvons regarder deux types d'évolutions pour comparer la différence entre la planification prévu et la planification réelle, et nous pouvons changer la planification actuelle et mettre en évidence les tâches à réaliser en priorité pour rattraper les objectifs de la planification initiale.

Pour le coordinateur de chantier, cette fonction peut aider à examiner le planning final. En cas de problème dans le planning, la représentation de l'évolution dans le temps la mettra en évidence.

2.3.2. Analyser les composants du planning

Lorsqu'il y a beaucoup de tâches dans un planning de chantier, vérifier et examiner le planning est un problème. De plus, le planning est un système

dynamique, qui change tous les jours à cause des actions des intervenants du chantier ou pour des raisons naturelles.

La méthode «4D CAD» est une solution pour ce problème. Les composants de planification liés avec les composants CAD permettent de mettre en évidence les problèmes du planning ou les erreurs de définition des composants 4D.

2.4. Bilan

La coordination des acteurs est un objectif dans la gestion de l'information du bâtiment (« Building Information Management »). Les méthodes de «4D CAD» permettent une forte liaison entre coordinateurs et équipes de construction. Dans un chantier, de nombreuses entreprises travaillent ensemble, se référant à un planning principal. Les outils de «4D CAD» les aident à mieux comprendre leurs relations de travail.

2.4.1. Avantage et limite des logiciels «4D CAD»

Les logiciels «4D CAD» constituent une bonne solution pour une meilleure coordination des acteurs et des entreprises du chantier. Pendant le travail, ils peuvent ainsi se référer à la planification. Pour les coordinateurs de chantier, ils permettent de mieux connaître l'évolution de la construction et ainsi d'éviter les risques potentiels. Enfin, les entreprises peuvent mieux comprendre leurs propres tâches et la relation avec les tâches des autres participants.

Mais, pour mieux utiliser les outils de «4D CAD», il faut construire la maquette 3D de façon rigoureuse et avec suffisamment de détails. Actuellement, les outils de «4D CAD» ne supportent que les formats «Autocad » et «Archicad », donc les fichiers d'autres outils de CAD ne peuvent fonctionner. De plus, les « contraintes » ne sont pas bien décrites dans les logiciels «4D CAD», s'il y a des changements dans la planification, il est nécessaire de modifier le logiciel de planning pour respecter ces contraintes. Dès lors, «4D CAD» n'est pas actuellement un logiciel planification complet.

2.4.2. Amélioration possible des logiciels «4D CAD»

Tout d'abord, les outils «4D CAD» devraient supporter plusieurs formats de modèles CAD et de planning. Actuellement, ils ne peuvent pas remplacer les logiciels de planification classiques. Il existe des solutions informatiques, comme les formats IFC, XML et WRL. Le format IFC semble intéressant et envisageable, car il est un format spécifique pour l'architecture.

La deuxième amélioration possible pour les outils «4D CAD», serait d'adjoindre les « contraintes » dans les composants de planification. Les « contraintes » et le « temps » sont symbioses, et dans la réalité du chantier, on doit modifier les « durées », « dates de début » ou « dates de fin » des tâches régulièrement.

3. Hypothèses et perspectives

Ces limites et les améliorations possibles des outils de «4D CAD» nous permettent d'envisager des hypothèses pour la suite de notre étude. Tout d'abord, le problème de l'interopérabilité pourrait être surmonté par l'utilisation d'un format de données commun pour la description des objets et du planning. La recherche actuelle dans le domaine de la spécification IFC semble prometteuse.

La spécification IFC [3] est fondée en 1997 par l'organisme IAI, visant l'objectif d'interopérabilité et de coordination du cycle de vie de bâtiment ou de construction. Quelques entreprises de logiciel et organismes voulaient en effet créer une spécification architecturale pour transférer les documents entre les différents logiciels de DAO. Depuis 2005, IFC change son nom en SMART BUILDING, et est géré par l'organisation standard internationale (ISO). Dans cet article, pour une question de facilité, on l'appellera encore IFC. Maintenant, l'objectif d'IFC est BIM (Building Information Model), c'est-à-dire qu'un modèle du bâtiment contient toutes les informations nécessaire avec la description correspondant la spécification OOD (Orienté Objet Direction), donc, ce genre de modèle peut remplacer les autres modèles, et peut mettre à jours avec les autres spécifications, surtout il peut avoir ses propre propriétés.

Actuellement, presque toutes les entreprises importantes utilisant le logiciel d'architecte ont intérêt sur IFC, beaucoup de logiciels architecturaux et construction supportent la spécification IFC déjà, comme Autocad Architecte Desktop, Archi CAD, Bentley etc. Nous pouvons prédire que IFC est un format unique dans la région architecte et construction.

3.1. Les éléments d'IFC correspondant le planning

IFC est un système énorme, complète et complexe. Il se compose de 110 « Defined Types » [4] 161 « Enumerations » [4] 42 « Select Types » [4] et 629 « Entités » [4] avec la relation hétérogène. Pour arriver à l'objectif « BIM », il définit presque tous les éléments qui doivent être utilisés dans toute la vie de cercle de construction. Pour les principes fondamentaux de planification que nous avons analysé dans le premier chapitre, IFC a aussi bien définit les entités .

3.1.1. le « temps »

le « temps » est un élément crucial de planification, lorsque nous avons analysé dans le premier chapitre, comme il est une notion abstraite, le « temps » se présente par des notions réelles : « data début » « data fini » et « durée ». IFC donne une série d'entités pour décrire ces notions.



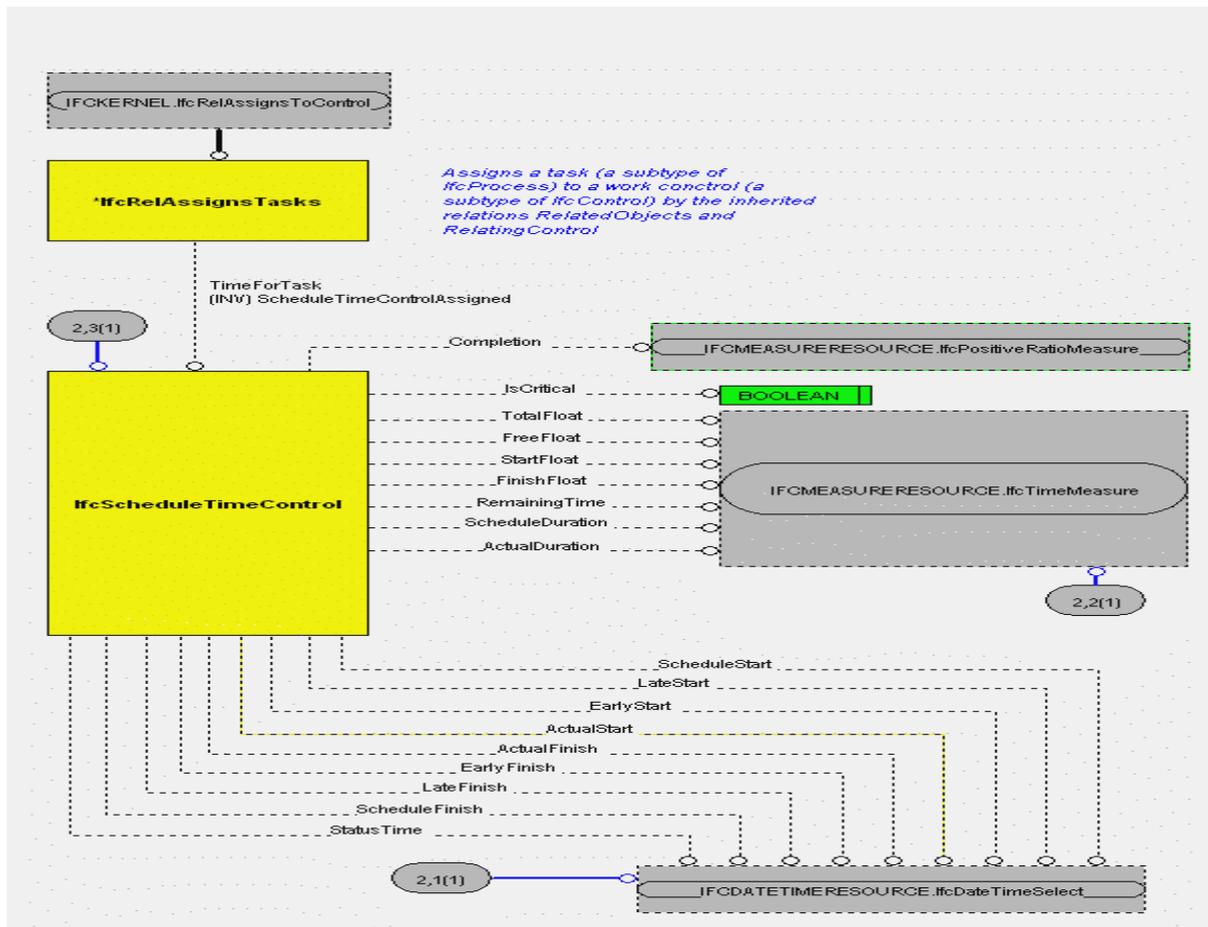


Schéma 06 la schéma d'IFC d'*IfcScheduleTimeControl*

Le problème est que la planification est un système dynamique et flexible, lorsque la date début et la date fini sont souvent modifiées, les logiciels planification courants ne peuvent pas résoudre facilement ce problème. IFC donne la solution raisonnable : la data contient la data prévue et la data réelle, ceux qui sont les propriétés d'un objet. En effet, Nous pouvons définir plusieurs genres dates en un seul objet.

3.1.2. les « ressources »

Comme nous l'avons analysé, les « ressources » sont déparés deux types généralement, la « ressource » en matériaux et la « ressource » humaine. Un grande inconvénient de « Microsoft Project » est qu'une tâche ne peut pas définir les « ressources » en matériaux raffinement et ne peut pas contenir les « ressources » humaines et les « ressources » en matériaux en même temps.

Dans la syntaxe de l'IFC, ces deux ressources sont définies par deux entités différentes : *IfcResource* et *IfcActor*. Pour *IfcResource*, nous pouvons définir n'importe quelle ressource en matériel comme nous voulons avec le prix unique. Par contre, l'*IfcActor* est une entité spéciale pour les gens qui travaillent dans le champ de l'architecture et de la construction. Surtout, un objet d'IFC peut avoir plusieurs « ressources » en même temps, bien que quelque ressource soit utilisé par un autre objet d'IFC.

3.1.3. Le « coût »

Le « coût » est un résultat dépendant du « temps » et des « ressources » dans le logiciel planification, mais il y a quelques « coûts » existant en réalité qui ne peuvent pas être comptés par le logiciel planification. Par exemple, le « coût » de communication, le « coût » de maintenance, etc.

IFC définit le « coût » par les notions : *IfcCostItem*, *IfcCostSchedule* et *IfcCostValue*. Lorsque le « coût » est une propriété d'un objet, il ne dépend pas du « temps » ou des « ressources ». Nous pouvons définir un coût particulier avec le prix unique par *IfcCostValue* et *IfcCostItem*, et puis distribuer ce coût au objet. Si on a besoin de compter le prix total, l'entité *IfcCostSchedule* peut bien nous aider.

3.1.4. Les « contraintes »

Les « contraintes » sont une fonctionnalité très importantes dans le logiciel planification pour gérer les risques et économiser le « temps ». Mais, cette fonctionnalité utile n'existe pas dans le «4D CAD». IFC donne une ENUMERATION *IfcSequenceEnum* qui justement correspond à 4 types de « contraintes ».

```
TYPE IfcSequenceEnum = ENUMERATION OF  
  
START_START,  
START_FINISH,  
FINISH_START,  
FINISH_FINISH,  
NOTDEFINED);  
  
END_TYPE;
```

schéma 07 la syntaxe d'IFC d'IfcSequenceEnum

3.1.5. bilan

IFC est une spécification internationale architecturale, nous l'analyserons un peu pour chercher les notions de la planification. On en conclura que les problèmes de logiciel planification que nous analyserons dans un premier chapitre seront bien résolus en IFC.

Un objet d'IFC est d'avoir un « temps » dynamique et flexible, pouvant relier avec plusieurs types de « ressources » en même temps, pouvant compter tous les « coût » possibles se rencontrant durant une construction. Dans tous les cas, pour décrire une planification complète et complexe, IFC a prévu toutes ces éventualités.

3.2. La description des éléments du planning par IFC

Le fait que les IFC contiennent des notions temporelles donne la possibilité d'utiliser ces notions pour décrire la planification. Nous allons faire un expérimentation d'un objet décrit par IFC avec les informations de planification.

3.2.1. Préparer un fichier d'IFC d'un objet simple

Actuellement, les logiciels CAD courants supportent d'exporter le fichier avec le format IFC avec la version 2X. Pour faciliter le problème que nous analyserons, nous choisirons ARCHICAD 9.0 et ADT 2006 pour créer un mur ayant un mètre de longueur et trois mètres de hauteurs. Nous n'avons pas défini le matériel dans le logiciel CAD.

Pour deux fichiers d'IFC réalisant le mur qui viennent de différents logiciels, il y a aucune différence. C'est une autre preuve qu'IFC est une spécification générale et incroyable.

Pour un simple mur d'un mètre, il y a 70 lignes de descriptions d'IFC.

D'abord, il définit toutes les informations nécessaires pour un objet physique comme l'unité, la direction des coordonnées, la façon de l'axe etc. De plus, il identifie ce mur par un nom unique de projet et un nom unique de bâtiment. Enfin, il utilise des données géographiques pour décrire le format de mur.

```

ISO-10303-21;
HEADER;
FILE_DESCRIPTION ( ' IFC 2x' , ' 2 ; 1 ' ) ;
.
.
.
DATA;
#1=IFCSIUNIT ( * , . TIMEUNIT , $ , . SECOND . ) ;
#2=IFCSIUNIT ( * , . MASSUNIT , $ , . GRAM . ) ;
#3=IFCSIUNIT ( * , . LENGTHUNIT , $ , . MILLI . , . METRE . ) ;
.
.
.
#6=IFCUNITASSIGNMENT ( ( #3 , #4 , #5 , #1 , #2 ) ) ;
.
.
.
#9=IFCDIRECTION ( ( 1 , 0 , 0 ) ) ;
#10=IFCAXIS2PLACEMENT3D ( #7 , #8 , #9 ) ;
.
.
.
#16=IFCOWNERHISTORY ( #14 , #15 , $ , . ADDED . , 0 , $ , $ ) ;
#17=IFCPROJECT ( ' 2JSXdPS_97$xeV8hzyweNb' ... $ ) ;
.
.
.
#23=IFCBUILDING ( ' DuEtP16mH498Tn2GnHrV35' , . . $ ) ;
.
.
.
#32=IFCLINE ( #29 , #31 ) ;
.
.
.
#70=IFCRELAGGREGATES ( ' 0vQH8wfmX7fJTExiR3' ... $ ) ;
.

```

La description de fichier-

La définition d'unité

La définition de coordonné

L'identité de projet et bâtiment

La définition graphique

Schéma 08 la description d'IFC d'un objet

Nous pouvons regarder ce mur par un logiciel de troisième partie pour vérifier la correction de fichier d'IFC. Nous choisissons « IFC Engine Basic » pour regarder la syntaxe d'IFC, et « IFC Engine Viewer » pour vérifier la géographie de mur. Lorsque tous les objets qu'on analyse représentent un modèle 3D, on peut tourner, déplacer la caméra, et changer l'échelle d'objectif de caméra avec ce mur dans « IFC Engine Viewer ».

3.2.2. Ajouter les informations de planification

Quand le fichier d'objet d'IFC a terminé son travail, nous avons besoin d'ajouter les descriptions supplémentaires pour décrire les informations de planification. Actuellement, nous n'avons pas la façon de définir les descriptions d'IFC de planification automatiquement par l'ordinateur, donc nous devons les écrire à la main.

?? Le « temps »

En réalité, dans la planification, la date début prévue, la date finie prévue, la date début réelle, la date finie réelle sont plus importantes et intéressantes. La durée prévue et la durée réelle dépendent des dates début et fins, donc le logiciel peut compter logiquement. De plus, on peut également obtenir le pourcentage d'exécution de tâche.

« ressources » en matériaux, donc ici, nous ne traitons pas de la description du « coût ».

?? Les « contraintes »

Les « contraintes » sont plus faciles que le « temps » et les « ressources » en IFC, nous pouvons définir une tâche, avoir une ou plusieurs contraintes avec une ou plusieurs autres tâches. Dans ce cas, pour un seul mur, nous n'avons pas d'autre objet pour relier avec ce mur, donc ici nous ne faisons pas les descriptions des « contraintes ». Nous essaieront d'analyser ces descriptions dans le prochain chapitre.

3.2.3. *bilan*

Dans ce chapitre, nous avons fait les descriptions d'IFC de planification d'un mur. Nous pouvons trouver qu'IFC est capable de décrire tous les aspects de planification. Pour le « temps », nous sommes parvenus à définir tous les types de date et durée, pour les « ressources », nous avons réussi à attribuer plusieurs ressources pour un seul objet (tâche). Donc, les deux problèmes majeurs de logiciels planifications normaux ont été résolus parfaitement. Dans le prochain chapitre, nous essaierons de décrire plusieurs objets dans un même fichier, et d'utiliser les « contraintes » pour faire une série de tâche séquence, c'est-à-dire «4D CAD».

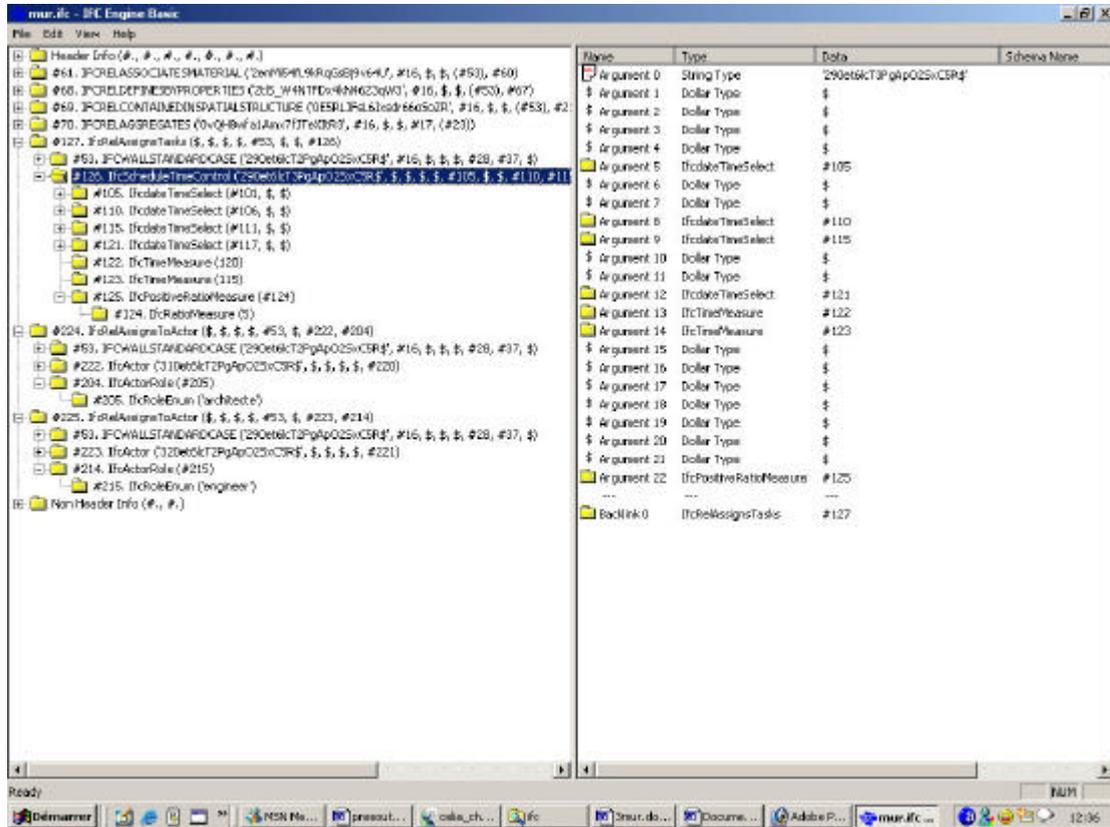


schéma 11 la vérification de syntaxe de fichier d'IFC d'un objet

3.3. La description des éléments de 4D par les IFC

Logiquement, nous pourrions décrire les éléments de 4D grâce aux IFC qui ont la capacité d'interpréter toutes les entités de planification. Dans ce chapitre, nous ferons un autre fichier d'IFC qui contiendra trois murs avec la séquence de constructions, de plus, chaque mur aura un même architecte, ingénieur et constructeur.

3.3.1. Préparer un fichier d'IFC de 3 objets



schéma 12 la description d'IFC de trois objets

Nous faisons un fichier d'IFC par Archicad 9.0 cette fois. Dans ce fichier, il y a trois murs qui s'appelle 'mur 001' 'mur 002' et 'mur 003'. Nous devons fabriquer 'mur001' d'abord, et puis 'mur 002', en fin 'mur 003'. Pour résoudre le problème principal que nous ajoutons les informations temporelle, nous ne pensons pas les « ressources » en matériaux.

Pour un fichier de trois murs, il y a 625 lignes de descriptions.

3.3.2. Ajouter les informations temporelle

Avec le fichier d'IFC de trois murs, il y a trop de descriptions lorsqu'ils ont une relation graphique hétérogène. Nous devons bien définir les informations raffinées de planification

Le « temps »

Pour plusieurs objets en IFC, la façon de définir l'identité est un gros problème. Nous devons bien distinguer les différents objets qui appartiennent à un même type. IFC donne un Defined Types, *IfcGloballyUniqueId*, pour identifier chaque objet. Cette identité est un numéro de 128 bits, fabriqué par Open Group. Il offre un ID unique universel.

En revanche, lorsqu'il y a plusieurs de « temps », nous devons définir une séquence pour exécuter des tâches. Nous imaginons que la séquence est que 'mur001' → ' mur002' → 'mur003', et chaque mur a une différente date réelle que la date prévu.

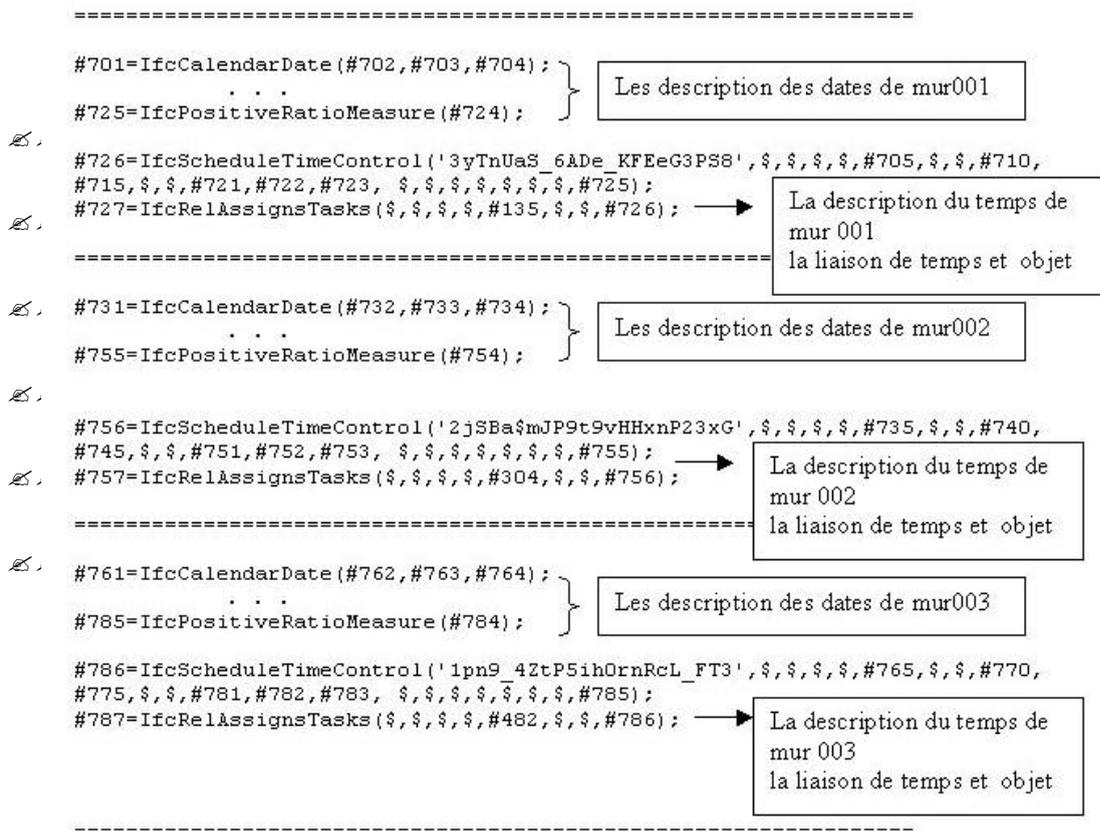


schéma 13 la description d'IFC du « temps » de trois objets

Les « ressources »

Comme le cas d'un mur, nous ne pensons pas les « ressources » en matériaux dans notre article, même Archicad a défini le 'Masonry Block' à ces murs automatiquement. Les « ressources » humaines nous intéressent plus particulièrement car ils sont facteurs de conflit et nous devons parvenir à mettre en accord ces personnes. Nous définissons un architecte et un ingénieur comme le premier exemple. Chacun occupe tous les trois murs aussi. En tant que nous ne comptons pas le « coût » dans ce cas, nous ne donnons pas le prix unitaire à ces deux acteurs.

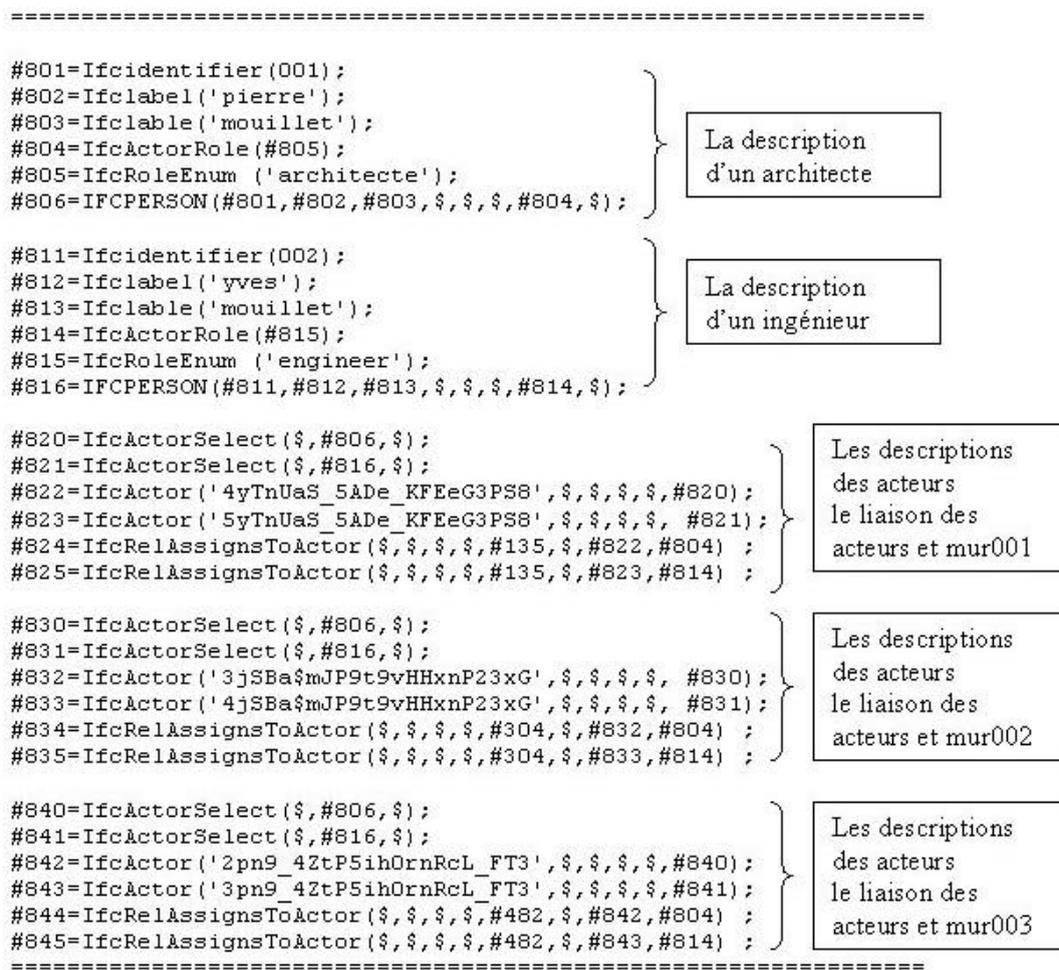


schéma 14 la description d'IFC des acteurs de trois objets

Les « contraintes »

Les « contraintes » que nous allons rencontrer et analysé par la suite sont les propriétés seulement du « temps », il y a également d'autres types de « contraintes » en IFC pour les « ressources » et les « prix ». Pour cet article, la séquence pour construire ces trois murs est plus importante, dès lors nous devons créer une contrainte pour insister sur cette séquence quand on change le « temps » des tâches. Insistons sur le fait que la date prévue et la date réelle ne sont pas exactement pareilles.

Pour faire une contrainte, il faut définir une séquence par l'entité d'IFC `IfcRelSequence`, ensuite, il faut définir un entité `IfcProcess` qui, lorsque la séquence ne peut pas être relié avec un objet, mais seulement avec un processus, donc logiquement, nous devons attribuer le processus que nous avons fait à un mur par `IfcRelAssignsToProcess`. Enfin, on définit une type de contrainte par `IfcSequenceEnum`. Quand tous les éléments nécessaires sont réunis, nous pouvons écrire la description de `IfcRelSequence`, on peut donner un nom à cette séquence, ensuite noter quel processus a la relation avec le processus que nous somme en train de décrire, enfin définir un type de contrainte entre ces processus. Nous pouvons noter deux processus, une est avant le processus présent, l'autre est après.

```

=====
#901=IfcRelSequence($,$,'contrainte_mur001',$,#902,#911,$,#904);
#902=IfcProcess($,$,'mur001',$,$);
#903=IfcRelAssignsToProcess($,$,'mur001',$,#135,$,#902,$);
#904=IfcSequenceEnum('FINISH_START');
} La contrainte « fini-
début » Entre mur001
et mur002

#910=IfcRelSequence($,$,'contrainte_mur002',$,#911,#921,$,#913);
#911=IfcProcess($,$,'mur002',$,$);
#912=IfcRelAssignsToProcess($,$,'mur002',$,#304,$,#911,$);
#913=IfcSequenceEnum('FINISH_START');
} La contrainte « fini-
début » Entre mur002 et
mur003

#920=IfcRelSequence($,$,'contrainte_mur003',$,#921,#911,$,#923);
#921=IfcProcess($,$,'mur003',$,$);
#922=IfcRelAssignsToProcess($,$,'mur003',$,#482,$,#921,$);
#923=IfcSequenceEnum('START_FINISH');
} La contrainte « début-
fini » Entre mur003 et
mur002
=====

```

schéma 15 la description d'IFC des « contraintes » de trois objets

Le « coût »

En ce cas, nous n'avons pas pensé le prix unitaire de l'acteur et la « ressource » en matériaux, donc nous ne pouvons pas avoir les descriptions de « coût » actuellement.

3.3.3. bilan

Se basant sur le résultat de l'expérimentation de 3.1 et 3.2, nous savons que pour un objet, nous pouvons ajouter les informations planification parfaitement dans le fichier 3D d'IFC. Dans ce chapitre, nous décrirons de multiples objets avec les informations temporelles. Il est claire que le fichier que nous créons est une descriptions complète de «4D CAD». Il a deux composantes principales que nous avons analysé dans le chapitre 2 de «4D CAD» et réussit à les combiner ensemble.

Contre un objet, il faut décrire les «contraintes » pour plusieurs objets. Du fait que les « contraintes » ne peuvent pas s'attacher avec un objet, parce que le « contrainte » n'est pas une propriété, il faut définir un processus pour relier l'objet et les « contraintes ».

Actuellement, nous ne pouvons que vérifier la syntaxe du fichier d'IFC de «4D CAD», lorsqu'il n'y a pas de façon de visualiser ce fichier avec le point de vue de 4D. Par ailleurs, les informations planifications que nous manipulons ne contiennent pas les « ressources » en matériaux et le « coût », mais celles-ci sont aussi importantes dans le cycle de la vie de la construction.

Dans le cadre de futures recherches, nous pourrions travailler sur la description d'IFC sur «4D CAD» qui est plus raffinée et efficace. De plus, il permet de trouver les descriptions des « ressources » en matériaux et de « coût ». L'objectif final est que nous pouvons créer un système d'IFC pour toutes les informations de la construction, on peut l'appeler ND CAD.

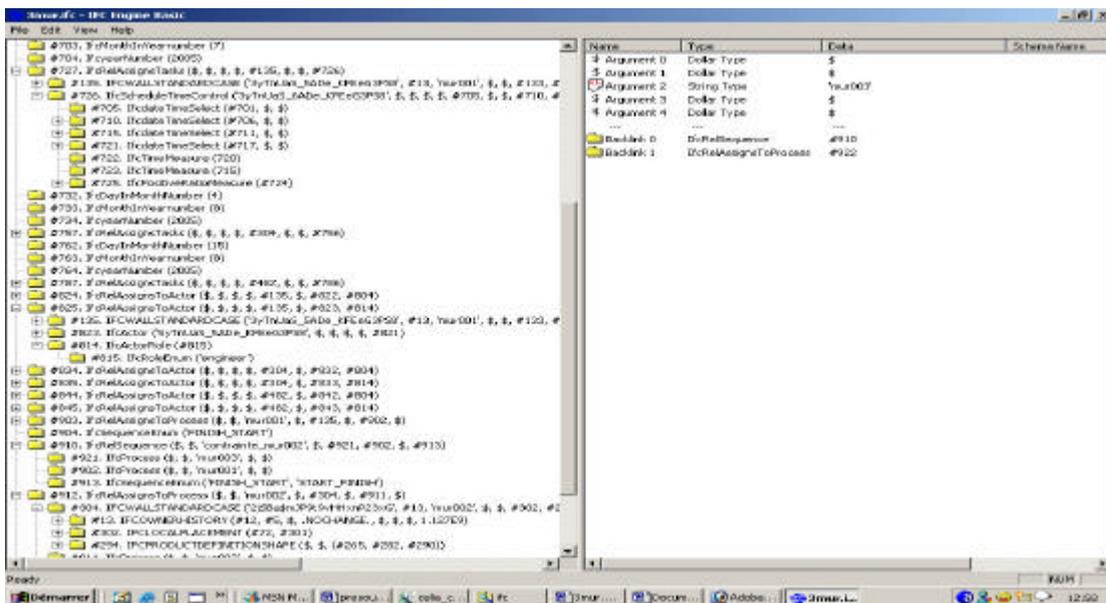


schéma 16 la vérification de syntaxe de fichier d'IFC de trois objets

4. Conclusion

Les analyses des différents types de logiciels de planification classiques ont mis en avant les principes fondamentaux et les différentes fonctionnalités liés à la planification. Pour les méthodes «4D CAD» existantes, nous avons décrit leurs composants nécessaires et les principales fonctionnalités.

Afin de décrire leurs éléments fondamentaux dans un format commun pour partager et échanger les informations plus facilement, nous proposons d'utiliser la spécification IFC.

La spécification IFC est destinée aux objets architecturaux, comme elle permet de décrire une maquette 3D efficacement. De plus, elle contient aussi les entités et les relations nécessaires afin d'interpréter la planification de manière la plus complète. Nous testons donc la possibilité de joindre la définition de planification à un 3D objet architectural qui est numérisé sous le format IFC. De plus, si nous faisons un test comprenant plusieurs objets avec les informations planifications, nous constatons que la spécification IFC est complètement capable de décrire un système de «4D CAD» standard.

Le problème actuel est que nous n'avons pas d'outils informatiques pour visualiser la 4D que l'on a fait par IFC, nous pouvons seulement vérifier la correction de syntaxe pour l'instant. L'autre problème est que l'on ne peut pas décrire toutes les informations de planification par un autre logiciel, car jusqu'à ici, tous les exemple ont été écrits à la main.

Néanmoins, IFC reste l'une des meilleures solutions pour «4D CAD». Pour atteindre l'objectif final, l'intégration des informations architecturale, il reste une proposition charmante et impressionnante également.

Bibliographies

- [1] K.Z. Chau., M. Anson., J.P. Zhang., *4D dynamic construction management and visualization software* , *Automation in Construction 14* (2005) 512-524
- [2] K.M. Liston., M. Fischer, J. Kunz, *4D annotator: a visual decision support tool for construction planners*, in : *Kelvin C.P. Wang(Ed), computing in Civil Engineering , Proceedings of International Computing Congress, Boston, October 18-21, ASCE,1998*
- [3] International Alliance for Interoperability Modeling Support Group, *IFC 2x Edition 2 Model Implementation Guide, version 1.7, march 18,2004*
- [4] Bo- Christer Bjork, *IFC-Compliant design information modeling and sharing*, <http://www.itcon.org/2003/1> , 04 2003
- [5] M. Williams, *Graphical simulation for project planning: 4D-planner*,*Proceedings of the Third Congress on Computing in Civil Engineering*, ASCE, Anaheim, CA, 1996, PP.404-409
- [6] K.Mc Kinney, J.Kim, M. Fischer, C. Howard, *Interactice 4D-CAD*, *Proceeding of the Third Congress in Computing in Civil Engineering*, ASCE, Anaheim, CA, 1996, pp. 383-396
- [7] K.M. Liston, M. Fischer, J. Kunz, *4D annotator: a visual decision support tool for construction planning*, in: *Kelvin C.P. Wang (Ed.), Computing in Civil Engineering*, *Proceedings of International Computing Congress*, Boston, October 18-21, ASCE, 1998, pp. 330-341
- [8] S.Staub-French, M. Fischer, M. Spradlin, *Into the fourth dimension*, *Civil Engineering* . 69(5) (1999) 44 – 47
- [9] N. Dawood, E. Sriprasert, Z. Mallasi, B. Hobbs, *Development of an integrated information resource base for 4D/VR construction processes simulation*, *Autom. Constre.* 12 (2) (2003) 123 - 131
- [10] J. Son, M.J. Skibniewski, *Multiheuristic approach for resource leveling problem in construction engineering : hybrid approach*, *Journal of Constraction Engineering and Management*, ASCE 125 (1)(1999) 23- 31

Le fichier d'IFC d'un mur avec les informations de planification

```

/* les infos général du fichier de l'IFC */
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('IFC 2x'),'2;1');
FILE_NAME('Drawing2.dwg','2005-07-20T23:54:14',(''),(''),'IFC-Utility 2x for ADT V. 2, 0, 4, 0 (www.inopso.com) - IFC Toolbox Version 2.x (00/11/07)', 'Autodesk Architectural Desktop','');
FILE_SCHEMA(('IFC2X_FINAL'));
ENDSEC;

DATA;
#1=IFCSIUNIT(*,TIMEUNIT,,$,SECOND.);
#2=IFCSIUNIT(*,MASSUNIT,,$,GRAM.);
#3=IFCSIUNIT(*,LENGTHUNIT,,$,MILLI,,$,METRE.);
#4=IFCSIUNIT(*,AREAUNIT,,$,SQUARE_METRE.);
#5=IFCSIUNIT(*,VOLUMEUNIT,,$,CUBIC_METRE.);
#6=IFCUNITASSIGNMENT((#3,#4,#5,#1,#2));
#7=IFCCARTESIANPOINT((0,0,0));
#8=IFCDIRECTION((0,0,1));
#9=IFCDIRECTION((1,0,0));
#10=IFCAXIS2PLACEMENT3D(#7,#8,#9);
#11=IFCGEOMETRICREPRESENTATIONCONTEXT('TestGeometricContext','TestGeometry',3,0,#10,$);
#12=IFCPERSON('','','',$,$,$,$);
#13=IFCORGANIZATION('','','',$,$);
#14=IFCPERSONANDORGANIZATION(#12,#13,$);
#15=IFCAPPLICATION(#13,'IFC-Utility 2x for ADT V. 2, 0, 4, 0 (www.inopso.com)', 'Autodesk Architectural Desktop','');
#16=IFCOWNERHISTORY(#14,#15,$,ADDED,0,$,$,1121896454);
#17=IFCPROJECT('2JSXdPS_97$xeV8hzyweNb',#16,'DefaultProject','Automatically generated project',$,$,$,#11,#6);

/* infos de projet, l'identifiant unique et stochastique */

#23=IFCBUILDING('OuEtPl6mH498Tn2GnHrV35',#16,'DefaultBuilding','Automatically generated building',$,#22,$,$,ELEMENT,,$,$);
/* infos de building, l'identifiant unique et stochastique */

#32=IFCLINE(#29,#31);
#29=IFCCARTESIANPOINT((0,0));
#31=IFCVECTOR(#30,31485.87994717408);
#30=IFCDIRECTION((1,0));
#33=IFCCARTESIANPOINT((0,0));
#34=IFCCARTESIANPOINT((31485.87994717408,0));
#35=IFCTRIMMEDCURVE(#32,(IFCPARAMETERVALUE(0),#33),(IFCPARAMETERVALUE(1),#34),.T,.$,CARTESIAN.);
#36=IFCSHAPEREPRESENTATION(#11,'Axis','Curve2D',(#35));
#38=IFCCARTESIANPOINT((0,120));
#39=IFCCARTESIANPOINT((0,-120));

```

```
#40=IFCCARTESIANPOINT((31485.87994717408,-120.));
#41=IFCCARTESIANPOINT((31485.87994717408,120.));
#42=IFCCARTESIANPOINT((0.,120.));
#43=IFCPOLYLINE((#38,#39,#40,#41,#42));
#44=IFCARBITRARYCLOSEDPROFILEDEF(.AREA.,$,#43);
#45=IFCCARTESIANPOINT((0.,0.,0.));
#46=IFCDIRECTION((0.,0.,1.));
#47=IFCDIRECTION((1.,0.,0.));
#48=IFCAXIS2PLACEMENT3D(#45,#46,#47);
#49=IFCDIRECTION((0.,0.,0.9999999999999999));
#50=IFCEXTRUDEDAREASOLID(#44,#48,#49,2750.);
#52=IFCSHAPEREPRESENTATION(#11,'Body','SweptSolid',(#50));
#28=IFCLOCALPLACEMENT(#22,#27);
#27=IFCAXIS2PLACEMENT3D(#24,#25,#26);
#24=IFCCARTESIANPOINT((6037.955741997197,35813.90330382715,0.));
#25=IFCDIRECTION((0.,0.,1.));
#26=IFCDIRECTION((1.,0.,0.));
#22=IFCLOCALPLACEMENT($,#21);
#21=IFCAXIS2PLACEMENT3D(#18,#19,#20);
#18=IFCCARTESIANPOINT((0.,0.,0.));
#19=IFCDIRECTION((0.,0.,1.));
#20=IFCDIRECTION((1.,0.,0.));
#54=IFCCARTESIANPOINT((0.,-120.,0.));
#55=IFCBOUNDINGBOX(#54,31485.87994717408,240.,2750.);
#56=IFCSHAPEREPRESENTATION(#11,'','BoundingBox',(#55));
#37=IFCPRODUCTDEFINITIONSHAPE($,$,(#36,#52,#56));
#53=IFCWALLSTANDARDCASE('290et6lCT2PgApO2SxC5R$',#16,$,$,$,#28,#37,$);
#57=IFCMATERIAL('Unnamed');
#58=IFCMATERIALLAYER(#57,240.,.F.);
#59=IFCMATERIALLAYERSET((#58),'Standard');
#60=IFCMATERIALLAYERSETUSAGE(#59,.AXIS2.,.POSITIVE.,-120.);
#61=IFCRELASSOCIATESMATERIAL('2enMi54fL9kRqGsBj9v64U',#16,$,$,(#53),#60);
#62=IFCPROPERTYSINGLEVALUE('Layername',$,IFCLABEL('A-Wall-G'),$);
#63=IFCPROPERTYSINGLEVALUE('Red',$,IFCINTEGER(255),$);
#64=IFCPROPERTYSINGLEVALUE('Green',$,IFCINTEGER(255),$);
#65=IFCPROPERTYSINGLEVALUE('Blue',$,IFCINTEGER(0),$);
#66=IFCCOMPLEXPROPERTY('Color',$,'Color',(#63,#64,#65));
#67=IFCPROPERTYSET('179319achDXwxRDjvq1pDU',#16,'PSet_Draughting',$,(#62,#66));
#68=IFCRELDEFINESBYPROPERTIES('2tIS_W4NTFDx4kN4623qW3',#16,$,$,(#53),#67);
#69=IFCRELCONTAINEDINSPATIALSTRUCTURE('OE5RLJFsL6lxsdr66oSoZR',#16,$,$,(#53),#23);
#70=IFCRELAGGREGATES('0vQH8wfa1Amx7fJTeXlTR3',#16,$,$,#17,(#23));
```

```
#101=IfcCalendarDate(#102,#103,#104);
#102=IfcDayInMonthNumber(25);
#103=IfcMonthInYearNumber(07);
#104=IfcyearNumber(2005);
#105=IfcdateTimeSelect(#101,$,$);
```

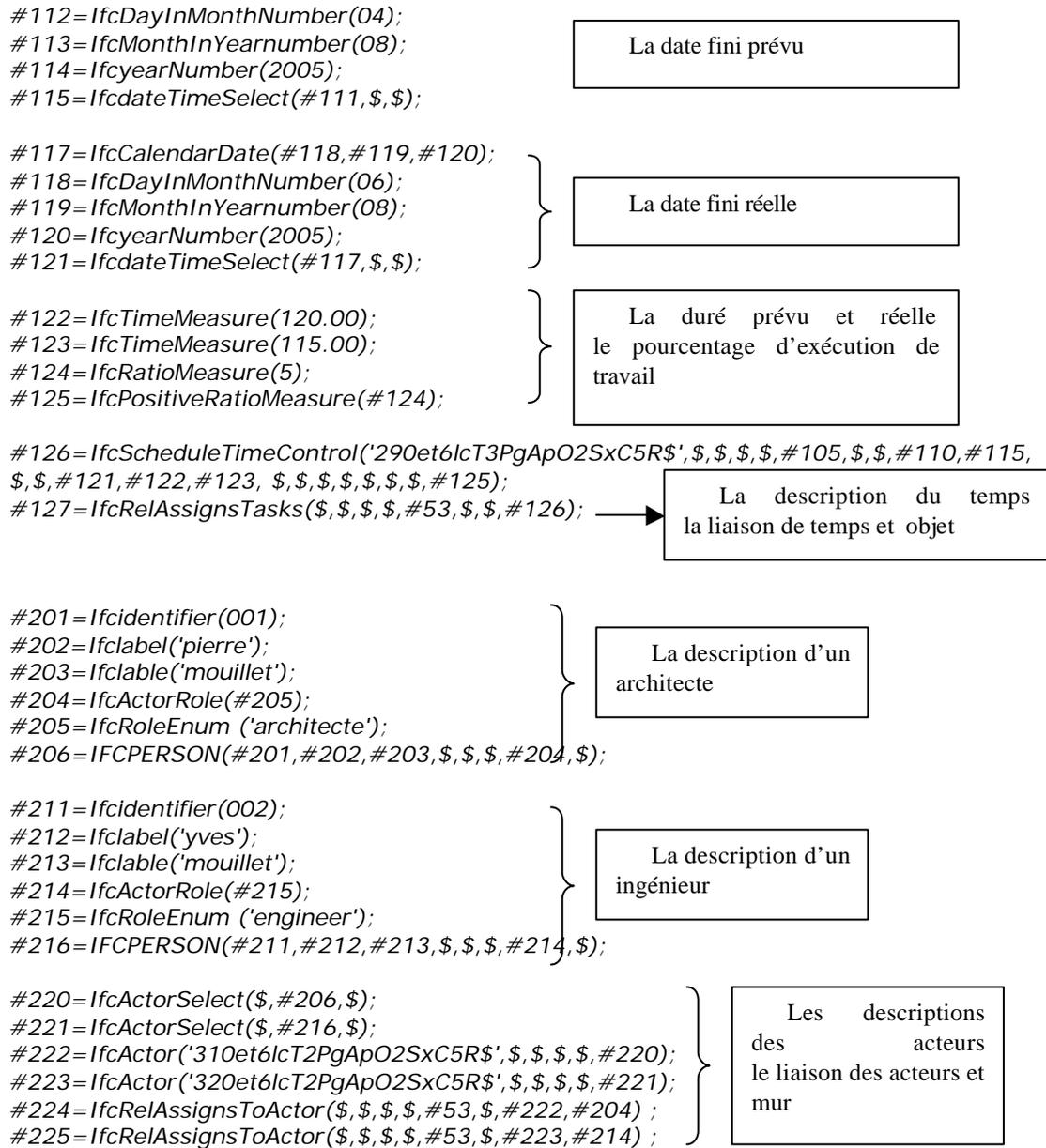
} La date début prévu

```
#106=IfcCalendarDate(#107,#108,#109);
#107=IfcDayInMonthNumber(26);
#108=IfcMonthInYearNumber(07);
#109=IfcyearNumber(2005);
#110=IfcdateTimeSelect(#106,$,$);
```

} La date début réelle

```
#111=IfcCalendarDate(#112,#113,#114);
```

}



ENDSEC;
END-ISO-10303-21;

Le fichier d'IFC de trois murs avec les informations de planification

```

ISO-10303-21;
HEADER;
FILE_DESCRIPTION (('IFC Engine Kernal version 1.07 beta release 0 generated IFC file.',
'Build Number of the Ifc 2x interface: unknown'), '2;1');
FILE_NAME ( ('Architect'), ('Building Designer Office'), '...', 'Windows System', 'The
authorising person. ');
FILE_SCHEMA (('IFC2X_FINAL'));
ENDSEC;
DATA;
#1 = IFCORGANIZATION('GS', 'Graphisoft', 'Graphisoft', $, $);
#5 = IFCAPPLICATION(#1, '9.0', 'ArchiCAD 9.0', 'ArchiCAD');
#6 = IFCPERSON($, 'Undefined', $, $, $, $, $);
#8 = IFCORGANIZATION($, 'OrganizationName', $, $, $);
#12 = IFCPERSONANDORGANIZATION(#6, #8, $);
#13 = IFCOWNERHISTORY(#12, #5, $, .NOCHANGE., $, $, $, 1127423488);
#14 = IFCSIUNIT(*, .LENGTHUNIT., .MILLI., .METRE.);
#15 = IFCSIUNIT(*, .AREAUNIT., $, .SQUARE_METRE.);
#16 = IFCSIUNIT(*, .VOLUMEUNIT., $, .CUBIC_METRE.);
#17 = IFCSIUNIT(*, .PLANEANGLEUNIT., $, .RADIAN.);
#18 = IFCMEASUREWITHUNIT(IFCPLANEANGLEMEASURE(0.017453293), #17);
#19 = IFCDIMENSIONALEXONENTS(0, 0, 0, 0, 0, 0, 0);
#20 = IFCCONVERSIONBASEDUNIT(#19, .PLANEANGLEUNIT., 'DEGREE', #18);
#21 = IFCSIUNIT(*, .SOLIDANGLEUNIT., $, .STERADIAN.);
#22 = IFCSIUNIT(*, .MASSUNIT., $, .GRAM.);
#23 = IFCSIUNIT(*, .TIMEUNIT., $, .SECOND.);
#24 = IFCSIUNIT(*, .THERMODYNAMICTEMPERATUREUNIT., $, .DEGREE_CELSIUS.);
#25 = IFCSIUNIT(*, .LUMINOUSINTENSITYUNIT., $, .LUMEN.);
#26 = IFCUNITASSIGNMENT((#14, #15, #16, #20, #21, #22, #23, #24, #25));
#28 = IFCDIRECTION((1., 0., 0.));
#30 = IFCDIRECTION((0., 1., 0.));
#32 = IFCDIRECTION((0., 0., 1.));
#35 = IFCCARTESIANPOINT((0., 0., 0.));
#37 = IFCAxis2Placement3D(#35, #32, #28);
#38 = IFCDIRECTION((6.1230318E-017, 1.));
#40 = IFCGEOMETRICREPRESENTATIONCONTEXT('Plan', 'Design', 3, 0.000010000000,
#37, #38);
#42 = IFCPROJECT('3B$VIRN3j8vg1ByMJfkqw5', #13, 'Default Project', $, $, $, $, (#40,
#120), #26);
#49 = IFCLOCALPLACEMENT($, #37);
#51 = IFCSITE('21fxv8gbP2ggHx6cxjHdiV', #13, 'Default Site', $, $, #49, $, $, .ELEMENT.,
(24, 28, 0), (54, 25, 0), $, $, $);
#60 = IFCLOCALPLACEMENT(#49, #37);
#62 = IFCBUILDING('19t51bSRL3_hgEahEJShIk', #13, 'Default Building', $, $, #60, $,
$, .ELEMENT., $, $, $);
#71 = IFCAxis2Placement3D(#35, #32, #28);
#72 = IFCLOCALPLACEMENT(#60, #71);
#74 = IFCBUILDINGSTOREY('2$0a7vaAXAeebVKYluxrVB', #13, '', $, $, #72, $,
$, .ELEMENT., 0.);
#83 = IFCMATERIAL('Masonry Block');
#85 = IFCMATERIALLAYER(#83, 190., $);
#87 = IFCMATERIALLAYERSET((#85), 'Masonry Block');
#89 = IFCMATERIALLAYERSETUSAGE(#87, .AXIS2., .POSITIVE., 0.);

```

```

#90 = IFCCARTESIANPOINT((0., 0.));
#92 = IFCCARTESIANPOINT((2000., 0.));
#94 = IFCPOLYLINE((#90, #92));
#96 = IFCSHAPEREPRESENTATION(#40, 'Axis', 'Curve2D', (#94));
#100 = IFCCARTESIANPOINT((190., 0.));
#102 = IFCCARTESIANPOINT((2000., 0.));
#104 = IFCCARTESIANPOINT((2000., 190.));
#106 = IFCCARTESIANPOINT((0., 190.));
#108 = IFCPOLYLINE((#100, #102, #104, #106, #100));
#110 = IFCCARTESIANPOINT((190., 0.));
#111 = IFCCARTESIANPOINT((2000., 0.));
#112 = IFCCARTESIANPOINT((2000., 190.));
#113 = IFCSHAPEREPRESENTATION(#40, 'Body', 'SweptSolid', (#112));
#117 = IFCBOUNDINGBOX(#35, 2000., 190., 3000.);
#118 = IFCDIRECTION((6.1230318E-017, 1.));
#120 = IFCGEOMETRICREPRESENTATIONCONTEXT('Plan', 'Sketch', 3, 0.000010000000,
#37, #118);
#123 = IFCSHAPEREPRESENTATION(#120, 'IAI', 'BoundingBox', (#117));
#127 = IFCPRODUCTDEFINITIONSHAPE($, $, (#96, #113, #123));
#130 = IFCCARTESIANPOINT((0., 2000., 0.));
#132 = IFCCARTESIANPOINT((0., 2000., 0.));
#133 = IFCCARTESIANPOINT((0., 2000., 0.));
#135 = IFCWALLSTANDARDCASE('3yTnUaS_5ADe_KFEeG3PS8', #13, 'mur001', $, $,
#133, #127, $);
#152 = IFCRELASSOCIATESMATERIAL('2Z_DC$tDgYBMd6zJU1H', #13, $, $, (#135),
#89);
#154 = IFCPROPERTYINGLEVALUE('LAYERNAME', $, IFCDESCRIPTIVEMEASURE('Exterior
Walls'), $);
#158 = IFCPROPERTYINGLEVALUE('INFO', $, IFCDESCRIPTIVEMEASURE('mur001', ''));
#163=IFCPROPERTYINGLEVALUE('REFMATNAME', $, IFCDESCRIPTIVEMEASURE('Surface-
Whitewash'), $);
#167=IFCPROPERTYINGLEVALUE('SIDEMATNAME',
#171=IFCPROPERTYINGLEVALUE('OPPMATNAME', $, IFCDESCRIPTIVEMEASURE('Surface-
Whitewash'), $);
#175=IFCPROPERTYINGLEVALUE('WALL CONTPEN', $, IFCDESCRIPTIVEMEASURE('Pen1'),
$);
#179=IFCPROPERTYINGLEVALUE('WALLCONTLTYPE', $, IFCDESCRIPTIVEMEASURE('Solid
Line'), $);
#183=IFCPROPERTYINGLEVALUE('WALLCONTPEN3D', $, IFCDESCRIPTIVEMEASURE('Pen2'
), $);
#187=IFCPROPERTYINGLEVALUE('WALL FILLPEN', $, IFCDESCRIPTIVEMEASURE('Pen10'),
$);
#191=IFCPROPERTYINGLEVALUE('WALLFILLBGPEN', $, IFCDESCRIPTIVEMEASURE('Pen91'
), $);
#195 = IFCPROPERTYINGLEVALUE('WALL USECOMPENS', $, IFCINTEGER(0), $);
#199 = IFCPROPERTYINGLEVALUE('WALL USECOMPBGPEN', $, IFCINTEGER(0), $);
#203 = IFCCOMPLEXPROPERTY('WALL', $, 'ArchiCAD', (#154, #158, #163, #167, #171,
#175, #179, #183, #187, #191, #195, #199));
#208 = IFCPROPERTYSET('3XF$4i$sb7a800A1uEfcoE', #13, 'Graphisoft AC90 WALL',
'Graphisoft AC90', (#203));
#213 = IFCRELDEFINESBYPROPERTIES('0WCIA0eQHCsh6g2maRvt0b', #13, 'ArchiCAD',
'ExtendedProperties', (#135), #208);
#215 = IFCPROPERTYINGLEVALUE('IsExternal', $, IFCBOOLEAN(.T.), $);
#219 = IFCPROPERTYSET('1sf4Wt86L1MxODxP$vrRtWq', #13, 'Pset_WallCommon', $,
(#215));
#224 = IFCRELDEFINESBYPROPERTIES('01yj3RRRPEaxwfEsVQSxoc', #13, $, $, (#135),
#219);
#226 = IFCPROPERTYINGLEVALUE('Layername', $, IFCLABEL('Exterior Walls'), $);

```

```

#230 = IFCPROPERTY SINGLEVALUE('Red', $, IFCINTEGER(0), $);
#234 = IFCPROPERTY SINGLEVALUE('Green', $, IFCINTEGER(0), $);
#238 = IFCPROPERTY SINGLEVALUE('Blue', $, IFCINTEGER(0), $);
#242 = IFC COMPLEXPROPERTY('Color', $, 'Color', (#230, #234, #238));
#247 = IFCPROPERTY SET('3$SL4g9Vv72PwCyz2Ksit3', #13, 'PSet_Draughting', $, (#226,
#242));
#252 = IFCRELDEFINESBYPROPERTIES('3SQfH3KLL5Qf6upN18IwMw', #13, $, $, (#135),
#247);
#254 = IFCMATERIAL LAYER(#83, 190., $);
#256 = IFCMATERIAL LAYER SET((#254), 'Masonry Block');
#258 = IFCMATERIAL LAYER SET USAGE(#256, .AXIS2., .POSITIVE., 0.);
#259 = IFCCARTESIANPOINT((0., 0.));
#261 = IFCCARTESIANPOINT((2000., 0.));
#263 = IFCPOLYLINE((#259, #261));
#265 = IFC SHAPEREPRESENTATION(#40, 'Axis', 'Curve2D', (#263));
#269 = IFCCARTESIANPOINT((-190., 0.));
#271 = IFCCARTESIANPOINT((2000., 0.));
#273 = IFCCARTESIANPOINT((1810., 190.));
#275 = IFCCARTESIANPOINT((0., 190.));
#277 = IFCPOLYLINE((#269, #271, #273, #275, #269));
#279 = IFCARBITRARYCLOSEDPROFILEDEF(.AREA., $, #277);
#280 = IFCAxis2PLACEMENT3D(#35, #32, #28);
#281 = IFCEXTRUDEDAREASOLID(#279, #280, #32, 3000.);
#282 = IFC SHAPEREPRESENTATION(#40, 'Body', 'SweptSolid', (#281));
#286 = IFCCARTESIANPOINT((-190., 0., 0.));
#288 = IFCBOUNDINGBOX(#286, 2190., 190., 3000.);
#290 = IFC SHAPEREPRESENTATION(#120, 'IAI', 'BoundingBox', (#288));
#294 = IFCPRODUCTDEFINITION SHAPE($, $, (#265, #282, #290));
#297 = IFCDIRECTION((0., -1., 0.));
#299 = IFCCARTESIANPOINT((0., 2000., 0.));
#301 = IFCAxis2PLACEMENT3D(#299, #32, #297);
#302 = IFCLocalPLACEMENT(#72, #301);
#304 = IFCWALLSTANDARD CASE('2jSBa$mJP9t9vHHxnP23xG', #13, 'mur002', $, $, #302,
#294, $);
#321 = IFCRELASSOCIATES MATERIAL('0tw6IOIVP8$vZBxxnDP$hq', #13, $, $, (#304),
#258);
#323 = IFCCARTESIANPOINT((0., 190.));
#325 = IFCCARTESIANPOINT((-190., 0.));
#327 = IFCPOLYLINE((#323, #325));
#329 = IFCCARTESIANPOINT((190., 0.));
#331 = IFCCARTESIANPOINT((0., 190.));
#333 = IFCPOLYLINE((#329, #331));
#335 = IFC CONNECTION CURVE GEOMETRY(#327, #333);
#336 = IFCRELCONNECT SPATHELEMENTS('0TjHqvxcXAtf$oLXX6NRTF', #13, $, $, #335,
#304, #135, (), (), .ATSTART., .ATSTART.);
#339 = IFCPROPERTY SINGLEVALUE('LAYERNAME', $, IFCDESCRIPTIVEMEASURE('Exterior
Walls'), $);
#343 = IFCPROPERTY SINGLEVALUE('INFO', $, IFCDESCRIPTIVEMEASURE('mur002'), '');
#347 = IFCPROPERTY SINGLEVALUE('REFMATNAME', $, IFCDESCRIPTIVEMEASURE('Surface-
Whitewash'), $);
#351 = IFCPROPERTY SINGLEVALUE('SIDEMATNAME', $, IFCDESCRIPTIVEMEASURE('Surface-
Whitewash'), $);
#355 = IFCPROPERTY SINGLEVALUE('OPPMATNAME', $, IFCDESCRIPTIVEMEASURE('Surface-
Whitewash'), $);
#359 = IFCPROPERTY SINGLEVALUE('WALL CONTPEN', $, IFCDESCRIPTIVEMEASURE('Pen1'),
$);
#363 = IFCPROPERTY SINGLEVALUE('WALLCONTLTYPE', $, IFCDESCRIPTIVEMEASURE('Solid
Line'), $);

```

```

#367=IFCPROPERTYSINGLEVALUE('WALLCONTPEN3D',$,IFCDESCRIPTIVEMEASURE('Pen2'
), $)
#371=IFCPROPERTYSINGLEVALUE('WALL FILLPEN', $, IFCDESCRIPTIVEMEASURE('Pen10'),
$);
#375=IFCPROPERTYSINGLEVALUE('WALLFILLBGPEN',$,IFCDESCRIPTIVEMEASURE('Pen91'
), $);
#379 = IFCPROPERTYSINGLEVALUE('WALL USECOMPENS', $, IFCINTEGER(0), $);
#383 = IFCPROPERTYSINGLEVALUE('WALL USECOMPBGPEN', $, IFCINTEGER(0), $);
#387 = IFCCOMPLEXPROPERTY('WALL', $, 'ArchiCAD', (#339, #343, #347, #351, #355,
#359, #363, #367, #371, #375, #379, #383));
#392 = IFCPROPERTYSET('3hGkfQ9r9C$exL5AhY76nS', #13, 'Graphisoft AC90 WALL',
'Graphisoft AC90', (#387));
#397 = IFCRELDEFINESBYPROPERTIES('2TXCCiSND7DfZrbQbbuAaR', #13, 'ArchiCAD',
'ExtendedProperties', (#304), #392);
#399 = IFCPROPERTYSINGLEVALUE('IsExternal', $, IFCBOOLEAN(.T.), $);
#403 = IFCPROPERTYSET('ObsfFTMWj7NxN1go9Pxx', #13, 'Pset_WallCommon', $,
(#399));
#408 = IFCRELDEFINESBYPROPERTIES('1RFzdfi0TEwRtzwvvsutD7', #13, $, $, (#304),
#403);
#410 = IFCPROPERTYSINGLEVALUE('Layername', $, IFCLABEL('Exterior Walls'), $);
#414 = IFCPROPERTYSINGLEVALUE('Red', $, IFCINTEGER(0), $);
#419 = IFCPROPERTYSINGLEVALUE('Green', $, IFCINTEGER(0), $);
#423 = IFCPROPERTYSINGLEVALUE('Blue', $, IFCINTEGER(0), $);
#427 = IFCCOMPLEXPROPERTY('Color', $, 'Color', (#414, #419, #423));
#432 = IFCPROPERTYSET('OTYdOXvsD8LgqOwfgyZvq_', #13, 'Pset_Draughting', $, (#410,
#427));
#437 = IFCRELDEFINESBYPROPERTIES('14vtkgh7bDsP9PTxEfdcdX', #13, $, $, (#304),
#432);
#439 = IFCMATERIALLAYER(#83, 190., $);
#441 = IFCMATERIALLAYERSET((#439), 'Masonry Block');
#443 = IFCMATERIALLAYERSETUSAGE(#441, .AXIS2., .POSITIVE., 0.);
#444 = IFCCARTESIANPOINT((0., 0.));
#446 = IFCCARTESIANPOINT((2000., 0.));
#448 = IFCPOLYLINE((#444, #446));
#450 = IFCSHAPEREPRESENTATION(#40, 'Axis', 'Curve2D', (#448));
#454 = IFCCARTESIANPOINT((0., 0.));
#456 = IFCCARTESIANPOINT((2000., 0.));
#458 = IFCCARTESIANPOINT((2000., 190.));
#460 = IFCCARTESIANPOINT((190., 190.));
#462 = IFCPOLYLINE((#454, #456, #458, #460, #454));
#464 = IFCARBITRARYCLOSEDPROFILEDEF(.AREA., $, #462);
#465 = IFCAXIS2PLACEMENT3D(#35, #32, #28);
#466 = IFCEXTRUDEDAREASOLID(#464, #465, #32, 3000.);
#467 = IFCSHAPEREPRESENTATION(#40, 'Body', 'SweptSolid', (#466));
#471 = IFCBOUNDINGBOX(#35, 2000., 190., 3000.);
#472 = IFCSHAPEREPRESENTATION(#120, 'IAI', 'BoundingBox', (#471));
#476 = IFCPRODUCTDEFINITIONSHAPE($, $, (#450, #467, #472));
#479 = IFCAXIS2PLACEMENT3D(#35, #32, #28);
#480 = IFCLOCALPLACEMENT(#72, #479);
#482 = IFCWALLSTANDARDCASE('1pn9_4ZtP5ih0rnRcL_FT3', #13, 'mur003', $, $, #480,
#476, $);
#499 = IFCRELASSOCIATESMATERIAL('1D_zAJ5RDDkActMGKvHbc7', #13, $, $, (#482),
#443);
#501 = IFCCARTESIANPOINT((190., 190.));
#503 = IFCCARTESIANPOINT((0., 0.));
#505 = IFCPOLYLINE((#501, #503));
#507 = IFCCARTESIANPOINT((1810., 190.));
#509 = IFCCARTESIANPOINT((2000., 0.));

```

```

#511 = IFCPOLYLINE((#507, #509));
#513 = IFCCONNECTIONCURVEGEOMETRY(#505, #511);
#514 = IFCRELCONNECTSPATHELEMENTS('0h4Ep6k_b2VR1_$pi5Qt5S', #13, $, $, #513,
#482, #304, (), (), .ATEND., .ATSTART.);
#517 = IFCPROPERTYSINGLEVALUE('LAYERNAME', $, IFCDESCRIPTIVEMEASURE('Exterior
Walls'), $);
#521 = IFCPROPERTYSINGLEVALUE('INFO', $, IFCDESCRIPTIVEMEASURE('mur003'), '');
#525=IFCPROPERTYSINGLEVALUE('REFMATNAME', $, IFCDESCRIPTIVEMEASURE('Surface-
Whitewash'), $);
#529=IFCPROPERTYSINGLEVALUE('SIDEMATNAME', $, IFCDESCRIPTIVEMEASURE('Surface-
Whitewash'), $);
#533=IFCPROPERTYSINGLEVALUE('OPPMATNAME', $, IFCDESCRIPTIVEMEASURE('Surface-
Whitewash'), $);
#537=IFCPROPERTYSINGLEVALUE('WALL CONTPEN', $, IFCDESCRIPTIVEMEASURE('Pen1'),
$);
#541=IFCPROPERTYSINGLEVALUE('WALLCONTLTYPE', $, IFCDESCRIPTIVEMEASURE('Solid
Line'), $);
#546=IFCPROPERTYSINGLEVALUE('WALLCONTPEN3D', $, IFCDESCRIPTIVEMEASURE('Pen2'
), $);
#550=IFCPROPERTYSINGLEVALUE('WALL FILLPEN', $, IFCDESCRIPTIVEMEASURE('Pen10'),
$);
#554=IFCPROPERTYSINGLEVALUE('WALLFILLBGPEN', $, IFCDESCRIPTIVEMEASURE('Pen91'
), $);
#558 = IFCPROPERTYSINGLEVALUE('WALL USECOMPENS', $, IFCINTEGER(0), $);
#562 = IFCPROPERTYSINGLEVALUE('WALL USECOMPBGPEN', $, IFCINTEGER(0), $);
#566 = IFCCOMPLEXPROPERTY('WALL', $, 'ArchiCAD', (#517, #521, #525, #529, #533,
#537, #541, #546, #550, #554, #558, #562));
#571 = IFCPROPERTYSET('2tEfocRof58RGr2EhIxgVd', #13, 'Graphisoft AC90 WALL',
'Graphisoft AC90', (#566));
#576 = IFCRELDEFINESBYPROPERTIES('0IDVVoTgb4_gEZUmQRzBWV', #13, 'ArchiCAD',
'ExtendedProperties', (#482), #571);
#578 = IFCPROPERTYSINGLEVALUE('IsExternal', $, IFCBOOLEAN(.T.), $);
#582 = IFCPROPERTYSET('0QxWN44cTB7BNg6D$zvqN6', #13, 'Pset_WallCommon', $,
(#578));
#587 = IFCRELDEFINESBYPROPERTIES('24x_qLXXfBzwa7KkLIBmQ', #13, $, $, (#482),
#582);
#589 = IFCPROPERTYSINGLEVALUE('Layername', $, IFCLABEL('Exterior Walls'), $);
#593 = IFCPROPERTYSINGLEVALUE('Red', $, IFCINTEGER(0), $);
#597 = IFCPROPERTYSINGLEVALUE('Green', $, IFCINTEGER(0), $);
#601 = IFCPROPERTYSINGLEVALUE('Blue', $, IFCINTEGER(0), $);
#605 = IFCCOMPLEXPROPERTY('Color', $, 'Color', (#593, #597, #601));
#610 = IFCPROPERTYSET('2iNHmiYGPE0gsFlrTDxZ6J', #13, 'PSet_Draughting', $, (#589,
#605));
#615 = IFCRELDEFINESBYPROPERTIES('1Ji4R5han4Xuz3fZeX9340', #13, $, $, (#482),
#610);
#617 = IFCRELCONTAINEDINSPATIALSTRUCTURE('38vW3mAs16wR7_SkQqCkkH', #13,
'BuildingStoreyContainer', 'BuildingStoreyContainer for Building Elements', (#135, #304,
#482), #74);
#619 = IFCRELAGGREGATES('0XjYIWuD98t879Z_tI01hk', #13, 'BuildingContainer',
'BuildingContainer for BuildigStories', #62, (#74));
#623 = IFCRELAGGREGATES('1g9jWTTkn3JfGvCb1U4fRU', #13, 'SiteContainer',
'SiteContainer For Buildings', #51, (#62));
#625 = IFCRELAGGREGATES('13r$PhFIL1PufS0IwNHJgx', #13, 'ProjectContainer',
'ProjectContainer for Sites', #42, (#51));

```



```

#747=IfcCalendarDate(#748,#749,#750);
#748=IfcDayInMonthNumber(13);
#749=IfcMonthInYearNumber(08);
#750=IfcYearNumber(2005);
#751=IfcDateTimeSelect(#747,$,$);

#752=IfcTimeMeasure(750.00);
#753=IfcTimeMeasure(745.00);
#754=IfcRatioMeasure(5);
#755=IfcPositiveRatioMeasure(#754);

#756=IfcScheduleTimeControl('2jSBa$mJP9t9vHHxnP23xG',$,$,$,$,#740,#745,
,$,#751,#752,#753,$,$,$,$,$,$,#755);
#757=IfcRelAssignsTasks($,$,$,$,#304,$,$,#756);

```

**M
U
R
O
O
2**

```

#761=IfcCalendarDate(#762,#763,#764);
#762=IfcDayInMonthNumber(15);
#763=IfcMonthInYearNumber(08);
#764=IfcYearNumber(2005);
#765=IfcDateTimeSelect(#761,$,$);

#766=IfcCalendarDate(#767,#768,#769);
#767=IfcDayInMonthNumber(13);
#768=IfcMonthInYearNumber(08);
#769=IfcYearNumber(2005);
#770=IfcDateTimeSelect(#766,$,$);

#771=IfcCalendarDate(#772,#773,#774);
#772=IfcDayInMonthNumber(30);
#773=IfcMonthInYearNumber(08);
#774=IfcYearNumber(2005);
#775=IfcDateTimeSelect(#771,$,$);

#777=IfcCalendarDate(#778,#779,#780);
#778=IfcDayInMonthNumber(25);
#779=IfcMonthInYearNumber(08);
#780=IfcYearNumber(2005);
#781=IfcDateTimeSelect(#777,$,$);

#782=IfcTimeMeasure(780.00);
#783=IfcTimeMeasure(778.00);
#784=IfcRatioMeasure(5);
#785=IfcPositiveRatioMeasure(#784);

#786=IfcScheduleTimeControl('1pn9_4ZtP5ihOrnRcL_FT3',$,$,$,$,#765,$,$,#770,#775,$
,$,#781,#782,#783,$,$,$,$,$,$,#785);
#787=IfcRelAssignsTasks($,$,$,$,#482,$,$,#786);

```

**M
U
R
O
O
3**

