



HAL
open science

Génération automatique de plateformes de simulation pour des systèmes organisés en flux tirés

Hind El Haouzi, Rémi Pannequin, André Thomas

► **To cite this version:**

Hind El Haouzi, Rémi Pannequin, André Thomas. Génération automatique de plateformes de simulation pour des systèmes organisés en flux tirés. 7e Congrès International de Génie Industriel, Jun 2007, Trois Rivières, Canada. pp.CDRom. hal-00168539

HAL Id: hal-00168539

<https://hal.science/hal-00168539>

Submitted on 29 Aug 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

GÉNÉRATION AUTOMATIQUE DE PLATEFORMES DE SIMULATION POUR DES SYSTÈMES ORGANISÉS EN FLUX TIRÉS

Hind El Haouzi, Rémi Pannequin, André Thomas

Centre de Recherche en Automatique de Nancy CNRS UMR 7039

Nancy Université,

1, rue Lyautey

54000 Nancy

{Hind.Elhaouzi, Remi.Pannequin, Andre.Thomas}@cran.uhp-nancy.fr

RÉSUMÉ : Innovations technologiques, demandes clients de plus en plus versatiles, cycles de vie produits de plus en plus courts, sont autant de facteurs qui ont poussé les entreprises à s'organiser de manière réactive, à réduire leurs coûts de production en standardisant leurs produits et leurs processus. Les méthodes comme Demand Flow Technology sont ainsi apparues. Cette méthodologie implique une gestion logistique en flux tirés et conduit à cette standardisation stricte des process dont la conception et le dimensionnement sont vérifiés mathématiquement afin d'assurer une adaptabilité au profil de la demande. Cependant, ces calculs ne permettent pas de prendre en compte le caractère dynamique de ces lignes: des files d'attente peuvent se créer à cause du déséquilibre de charge entre les différents postes d'une ligne et les problèmes liés à la synchronisation des flux matières entre une ligne d'assemblage et ses lignes d'approvisionnement. La simulation dynamique semble un cadre de résolution intéressant. Dans cet article, nous proposons un outil de génération de modèles de simulation pour résoudre ces problèmes, ainsi que les problèmes inhérents à la simulation tels que la réutilisation et la synchronisation.

MOTS-CLÉS : Plateforme de Simulation, DFT, Réutilisabilité, Synchronisation des flux matières,

1. Introduction

Poussées par de constantes innovations technologiques, et par la demande de produits dont les cycles de vie sont de plus en plus courts, les entreprises s'organisent en flux tirés. Dans ce contexte exigeant, les systèmes de production doivent être extrêmement agiles. Ils ne peuvent plus s'appuyer sur des plans statiques, mais doivent répondre de façon dynamique à la demande et aux spécificités des clients. Plusieurs travaux ont été effectués pour répondre à ces problèmes notamment des travaux basés sur des méthodes de philosophie Juste à Temps (JIT). Parmi celles-ci, la DFT (Demand Flow Technology) (Costanza, 1996) est particulièrement intéressante car elle conduit à la standardisation des process. Cette méthodologie s'appuie sur des calculs mathématiques pour concevoir des lignes d'assemblage répondant à la demande client. Cependant, ces calculs ne permettent pas de prendre en compte le caractère dynamique de la ligne : par exemple les files d'attente qui peuvent se créer à cause du déséquilibre de charge entre les différents postes d'une ligne, ainsi que les problèmes liés à la synchronisation des flux matières entre une ligne d'assemblage et ses lignes d'approvisionnement.

C'est pourquoi il devient donc nécessaire d'utiliser des outils comme la simulation dynamique (Kelton *et al.* 1998). En effet, la simulation permet des apports à plusieurs niveaux. D'abord, la phase d'analyse du modèle permet de trouver des propriétés comportementales ou quantitatives sous forme d'indices de performances. Ensuite, la phase de modélisation permet une représentation abstraite du système réel et donc conduit à améliorer sa connaissance. Enfin, la simulation permet d'étudier les variations des indices de performance et le comportement dynamique du système observé.

L'utilisation de la DFT conduit donc à des processus standardisés et par conséquent à des systèmes de production structurés de la même manière. Ainsi l'objectif de notre étude a été de

construire un outil d'assistance à la génération de plateformes de simulation de tels systèmes industriels. Le but étant d'utiliser les avantages de la simulation dynamique pour concevoir, évaluer ou valider les process établis selon cette méthode. Par ailleurs, cet outil doit permettre aux experts du domaine non initiés aux techniques de la modélisation de facilement modéliser leur système.

Ce papier est décomposé en deux parties : la première partie est la construction des composants paramétrables pouvant être réutilisés, alors que la deuxième s'intéresse à l'automatisation de la construction du modèle à partir de ces composants. Nos travaux antérieurs (El Haouzi *et al.* 2005) nous ont permis de proposer une méthodologie de construction des modèles de simulation à partir d'un modèle de connaissance et des composants réutilisables de manière générale.

Dans un premier temps, nous exposerons les principes de la DFT et les problèmes qui suscitent de l'intérêt dans notre approche. Après, nous exposerons notre proposition. Enfin, un cas industriel (Entreprise Trane) permettra d'illustrer la méthodologie proposée.

2. Principes de la DFT

2.1. Objectifs

La DFT conduit à une mise en place d'une organisation de production en flux tirés. Cette réorganisation a plusieurs objectifs :

- gagner des parts de marché et de réduire les marges. En effet, la mise en place de la DFT permet une réduction des cycles de fabrication, une augmentation de la flexibilité et une augmentation de la qualité.
- réduire la surface de production. La DFT entraîne une réorganisation de l'espace productif en lignes de production spécifiques, appelées *pipelines*. Ce qui conduit à considérer qu'au centre de l'usine existe un flux principal de production. Pour alimenter ce flux, divers sous ensembles sont nécessaires, ces sous ensembles sont fabriqués sur des lignes appelées les *feeders*.
- réduire les stocks, à la fois de matière première, d'en-cours et de produits finis.
- réduire les investissements en optimisant les mixes de production.

2.2. Dimensionnement

Le dimensionnement des lignes supportant le flux principal et les feeders se fait à l'origine de la mise en place de la DFT et va de pair avec le calibrage d'un certain nombre de caractéristiques. Parmi celles-ci, l'IPK (In Process Kanban) qui représente une zone de stockage associée à un signal de synchronisation, la SOE (Sequence Of Event) qui consiste à lister les tâches à réaliser en identifiant celles qui sont à valeur ajoutée, et celles qui n'ont pas de valeur ajoutée (le temps machine, le temps main d'œuvre s'il est réalisé ou non en temps masqué, les critères de qualité...).

2.3. Simulation dans le contexte de la DFT

Dans ce contexte plusieurs problèmes vont se poser lors de la construction des modèles de simulation.

- la **réutilisabilité** de par la standardisation des moyens de production et de manutention
- la **synchronisation** entre les flux feeders et la ligne principale exigée par l'organisation en pipeline, afin d'éviter toute rupture de composants et optimiser la capacité des zones de stockage,

- la **simulation distribuée** permettant de faciliter le développement et la maintenance des modèles, réduire leur complexité, et d'accélérer l'exécution des simulations. Cet aspect est lié aux problèmes de synchronisation des lignes et des feeders alimentant plusieurs lignes à la fois.

Nos travaux se focalisent sur ces trois aspects, qui constituent les concepts fondamentaux que nous devons mettre en oeuvre. Toutefois, la simulation distribuée n'est pas évoquée dans cet article ; elle fait l'objet de travaux en cours, utilisant le protocole HLA. La section suivante va permettre de les décrire plus en détails.

3. Concepts fondamentaux de notre approche

3.1. Réutilisabilité

Dans la communauté scientifique du domaine de la simulation, ce problème n'est pas récent. Plusieurs travaux ont été effectués dans ce champ de recherche, on notera les travaux de Sragent (1986) et de Wyatt (1990). Plus récemment Robinson *et al.* (2004) ont donné une définition à « Modèles de simulation réutilisable ». Il ont également montré les motivations pour le développement des modèles réutilisables : le coût, et la « crédibilité de la réutilisabilité », c'est à dire le champs d'application effectif des modèles. Plusieurs niveaux de la réutilisabilité ont été établis (figure 1).

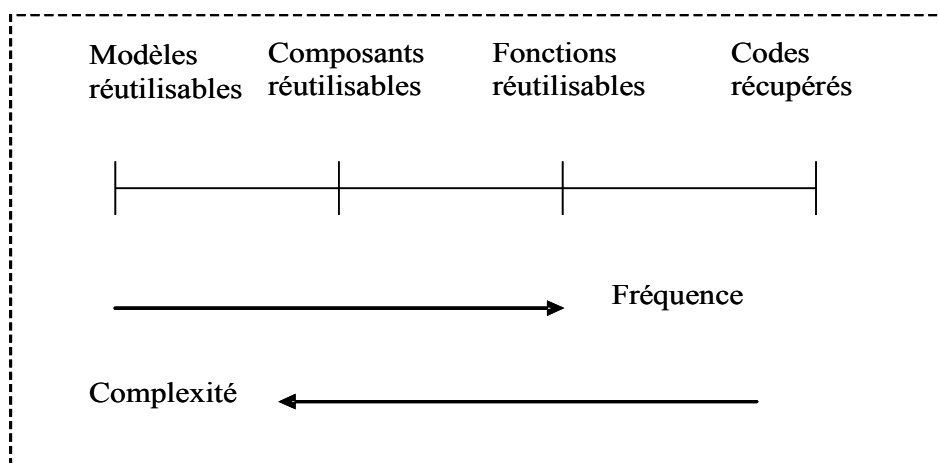


Figure 1: Échelle de réutilisabilité

Ces différents niveaux sont organisés selon deux axes : la fréquence et la complexité. On trouve que le « code récupéré » et les « fonctions » sont les plus fréquemment utilisés grâce au développement des langages de programmation et l'arrivée des langages orientés objets. De plus, l'apparition de différentes méthodologies (nous renvoyons le lecteur aux bibliographies citées pour plus de détails) comme CM (Conical Methodology) (Nance, 1994a, Nance, 1994b), IDEF (US Air Force, 1993), ASCI (Kellert *et al.*, 1998a, Kellert *et al.*, 1998b) ont permis de répondre à cette problématique au niveau de l'analyse et de la conception, et au niveau de l'implémentation en préconisant de développer des composants réutilisables. Nos premiers travaux se sont appuyés sur ces différentes méthodologies pour en établir une pour la construction de modèles de simulation dans le contexte du pilotage distribué et de la DFT, qui permet de proposer des composants réutilisables. Nous proposons aussi dans ce papier d'étudier le niveau supérieur qui consiste à construire nos modèles, non pas à partir des composants réutilisables, mais à partir d'un modèle réutilisable grâce à une automatisation de la création de ces modèles. Cela a pour objectif, de les uniformiser, d'automatiser les tâches

redondantes et surtout de permettre à des experts du domaine ne maîtrisant pas les techniques de modélisation, de modéliser leur ligne.

3.2. Synchronisation des flux feeders avec la ligne principale

Comme nous l'avons mentionné au paragraphe précédent, la DFT préconise l'utilisation d'une organisation selon un pipeline pour approvisionner la ligne au lieu d'une organisation standard. La figure suivante illustre ce principe.

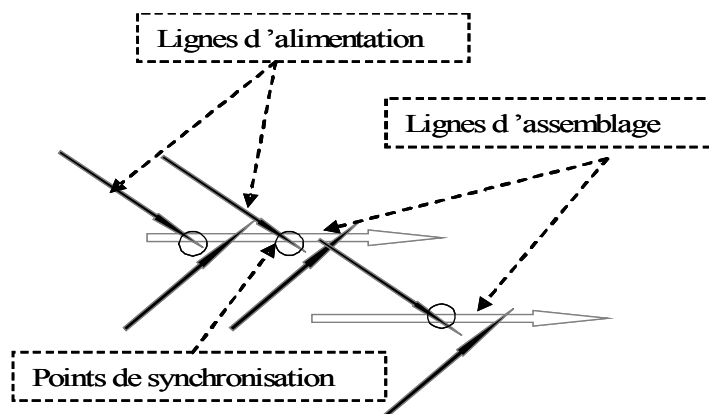


Figure 2: Organisation de la ligne selon la DFT

Cette organisation bien qu'elle permette une grande flexibilité et réactivité au niveau des flux matières, pose des problèmes de synchronisation qu'il faut résoudre au niveau des points de jonction. Aujourd'hui, la DFT recommande d'utiliser des emplacements Kanban (IPK) aux points de synchronisation pour pallier à des problèmes d'équilibrage de charge et éviter les ruptures, mais pour autant la DFT ne nous apporte pas d'éléments de réponse pour dimensionner ces emplacements. La simulation doit alors nous permettre de calculer la capacité nécessaire de ces emplacements.

3.3. Simulation distribuée pour les feeders non dédiés

Pour les feeders non dédiés (feeders approvisionnant plusieurs lignes à la fois), la simulation distribuée s'impose pour plusieurs raisons. D'une part, un feeder non dédié est physiquement éloigné de la ligne, il peut même être dans un autre site géographique et ainsi évoluer de manière indépendante de cette ligne principale. D'autre part, pour simuler toute une usine nous avons besoin de capacités de traitement importantes. Ainsi, distribuer la simulation induira d'utiliser plusieurs machines d'où un gain de capacité de traitement et donc de temps. Cependant, la simulation distribuée pose quelques problèmes tels que les problèmes d'interopérabilité et de gestion du temps.

4. Cadre de résolution

4.1. Structure du système de simulation

Pour répondre à tous ces problèmes nous avons mis en œuvre trois outils de développement relatifs aux trois sous systèmes : physique, informationnel et décisionnel (figure 3). En effet, l'approche méthodologique que nous avons utilisée préconise de subdiviser notre système selon ces trois sous systèmes. Cette vision systémique permet d'améliorer la modularité, la connaissance des flux et de leur nature.

Pour le sous-système physique, nous nous sommes intéressés aux logiciels de simulation utilisant une approche orientée objet pour plusieurs raisons. La première est de garder une cohérence entre toutes les étapes de la construction des modèles de simulation : le modèle de connaissance en UML de la phase d'analyse par exemple sera facilement exploité pour la construction du modèle d'action. Par ailleurs, cette approche permet d'identifier chaque objet du système réel par un objet du modèle (Anglania *et al.*, 2002). Plusieurs logiciels du commerce existent (Arena, Modelica, Witness, Promodel...). Notre choix s'est porté sur Arena, très utilisé dans le domaine industriel et académique (Al-Ahmari *et al.*, 1999), (De Swaan Arons *et al.*, 2000) (Perera *et al.*, 2000) (Kovács *et al.*, 1999). Le principe de ces logiciels est de générer un ou plusieurs flux d'entités discrètes, qui peuvent représenter des flux d'informations ou des flux physiques. Ces entités générées vont circuler dans un modèle composé de différents modules, lesquels auront une influence sur la vitesse du flux, sa nature ou son cheminement au sein du modèle. Ce type de logiciel a deux propriétés intéressantes: une interface programmable (API) pour communiquer avec d'autres logiciels et un module « Arena professionnel » qui offre la possibilité de créer sa propre bibliothèque regroupant des composants paramétrables. Il s'agit dans ce modèle de simuler les phénomènes physiques qui ont lieu dans l'atelier (transport, changement de forme, assemblage, etc...).

Le sous-système informationnel peut être mis en œuvre en utilisant tout type de logiciel capable de mémoriser de l'information structurée, comme un système de gestion de base de données relationnelle. Dans notre cas, nous avons utilisé MS Excel pour stocker les informations nécessaires au modèle Arena. Ce choix correspond au besoin d'un outil flexible et facile d'accès, satisfaisant aux contraintes techniques de notre projet. Le système d'information comporte les données techniques permettant de générer, de paramétrer et d'initialiser le modèle de simulation du système opérant (nb opérateurs, compétence sur les postes, produits à fabriquer...).

Le sous-système décisionnel est composé de deux parties : d'une part le système centralisé de pilotage de la production (gestion des ordres de fabrication) est mis en œuvre en utilisant une programmation en VBA. D'autre part, les décisions distribuées prises dans la réalité par les opérateurs sont modélisées grâce à une bibliothèque de modules. cela illustre la possibilité d'implanter le système de contrôle à plusieurs niveaux.

L'interface VBA servira aussi à la construction de la plateforme de simulation [Excel, Arena] automatiquement grâce à des macros dans Arena et Excel.

4.2. La réutilisabilité

4.2.1. Introduction

La réutilisabilité est assurée en plusieurs étapes : En premier lieu par le développement des composants génériques, ensuite, par l'automatisation de la création des modèles (modèle de simulation sur Arena et le modèle d'information sous Excel, sachant que le pilotage des lignes ne change pas). Plusieurs composants ont été développés : un composant générique pour le comportement de l'opérateur, un composant « IPK », un composant « gestion des ordres de fabrication », un composant « poste de travail » etc... Nous présentons ici le composant générique « poste de travail » qui nous servira à illustrer le mécanisme de synchronisation dans le paragraphe suivant. L'automatisation sera plus détaillée dans la section application industrielle.

4.2.2. Poste de travail

L'interface de l'objet « poste de travail » permet de renseigner les paramètres : le nom du poste, la compétence nécessaire pour travailler sur ce poste, le temps de travail qui est un attribut de l'unité passant sur ce poste, le nombre d'opérateurs maxi sur le poste. On ajoute à

ces paramètres des options comme par exemple : le poste est-il au début ou en fin de ligne, le poste est-il rattaché à un feeder ou pas ?

Au niveau implémentation, deux flux coexistent : le flux physique pour lequel les entités correspondent aux produits à fabriquer et le flux logique pour lequel les entités ne représentent rien, mais permettent d’effectuer divers traitements en fonction des variables d’environnement et d’événements issus du flux physique. Cette boucle représente les décisions distribuées prises localement sur le poste du fait de la DFT, de par la flexibilité des opérateurs et une gestion favorisant le pilotage par le produit. La figure suivante décrit le fonctionnement d’un poste de travail.

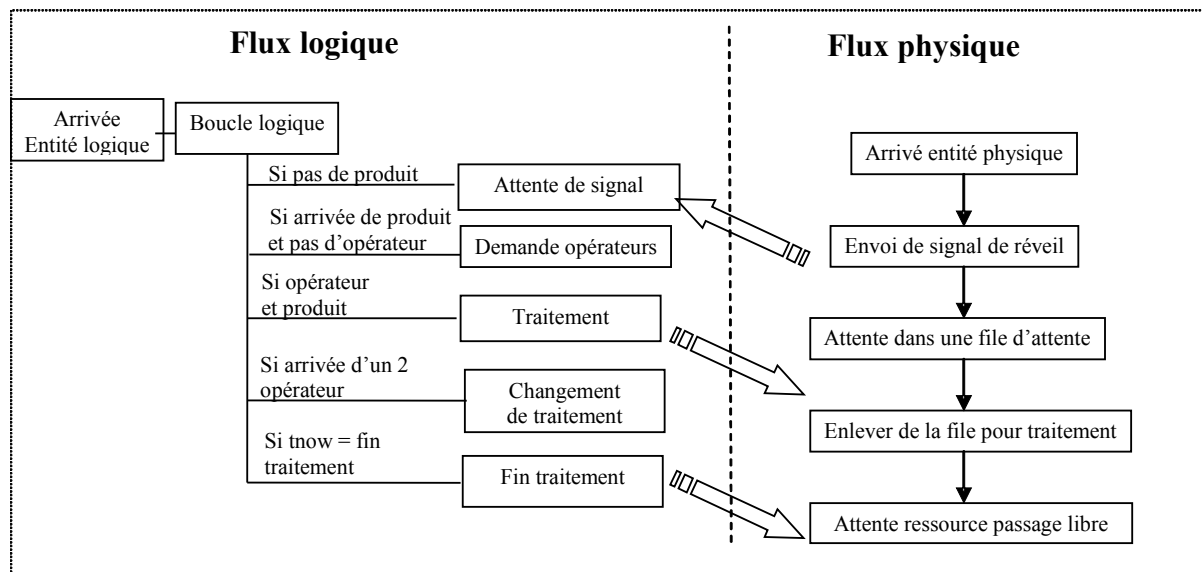


Figure 3: Comportement d’un poste de travail

4.3. Synchronisation d’une ligne et de ses feeders

La synchronisation entre une ligne et ses feeders se fait grâce à des signaux, pour éviter toute rupture, un emplacement IPK est prévu entre la ligne et le feeder, la capacité de cet emplacement est variable (de 1 à n).

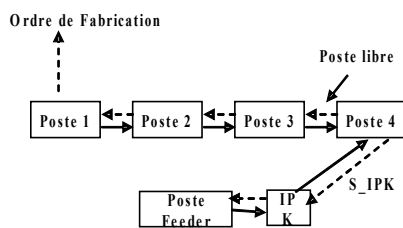


Figure 4: Synchronisation d'un feeder avec la ligne principale

Pour faciliter l'explication du fonctionnement de cette synchronisation nous nous basons sur la figure 4 : trois composants interagissent pour assurer la synchronisation : le poste de la ligne à alimenter (le Poste 4 qui est un poste de travail avec option feeder voir paragraphe 4.2), un emplacement IPK et le poste directement en amont de cet IPK (Poste Feeder). Nous commençons d’abord par décrire le comportement du poste de la ligne à alimenter.

4.3.1. Poste de la ligne à alimenter

Le flux logique suit le même cheminement qu'un poste normal (même traitement) par contre le flux physique suivra le cheminement suivant:

```
Si (Arrivée entité physique) alors
    Envoi (signal S_IPK)
    Attente (signal S_ArrC)
Si (Arrivée composant) alors
    Envoi (signal S_ArrC)
    Regroupement (composant, entité physique)
    Envoi (signal S_logique)
    Attente (traitement)
Fin si
```

4.3.2. Composant « IPK »

Pour l'IPK nous n'avons qu'un seul flux ; flux physique qui suit le cheminement suivant :

```
Si Arrivée (composant) alors
    Attente (signal S_IPK) (signal poste aval)
    Libérer (composant)
    PlaceIPK_libre= PlaceIPK_libre+1
Fin si
```

4.3.3. Poste en amont de l'IPK (feeder)

À ce poste, encore une fois, il n'y a pas de changement au niveau flux logique mais au niveau flux physique à la place d'attente ressource passage libre on attend que $PlaceIPK_libre > 0$. En effet, sur ce poste nous pouvons libérer plus d'une entité à la fois en fonction de la capacité de l'IPK.

Ce même mécanisme de synchronisation sera utilisé dans le cas de la simulation distribuée, avec comme différence la nature des signaux changés. En effet, la synchronisation s'effectuera avec des messages envoyés par les autres modèles, appelés sockets : ces sockets sont définis dans un format d'échanges neutre (XML).

La construction de la plate-forme de simulation de chaque ligne d'assemblage sera réalisée grâce aux composants génériques détaillés dans ce paragraphe et des fonctions d'automatisation que nous expliquerons par la suite.

4.4. Construction de la Plate-forme Simulation

La figure 5 résume la construction de la plateforme de simulation pour un système particulier à partir d'une plateforme générique du domaine des systèmes organisés en DFT. Nous distinguons trois niveaux d'abstraction. Le niveau conceptuel utilise la théorie des systèmes de Le Moigne (1984), et préconise d'étudier le système selon trois vues. Pour ces trois vues nous construisons des modèles de connaissance grâce à des diagrammes de classe UML pour identifier les objets du système et des diagrammes de séquence pour décrire les comportements de ces objets.

Le niveau opérationnel est consacré à l'implémentation de la plateforme générique à partir du modèle de connaissance développé au niveau conceptuel. Et enfin, au niveau applicatif, l'expert de domaine crée une instance de la plate-forme générique pour son système.

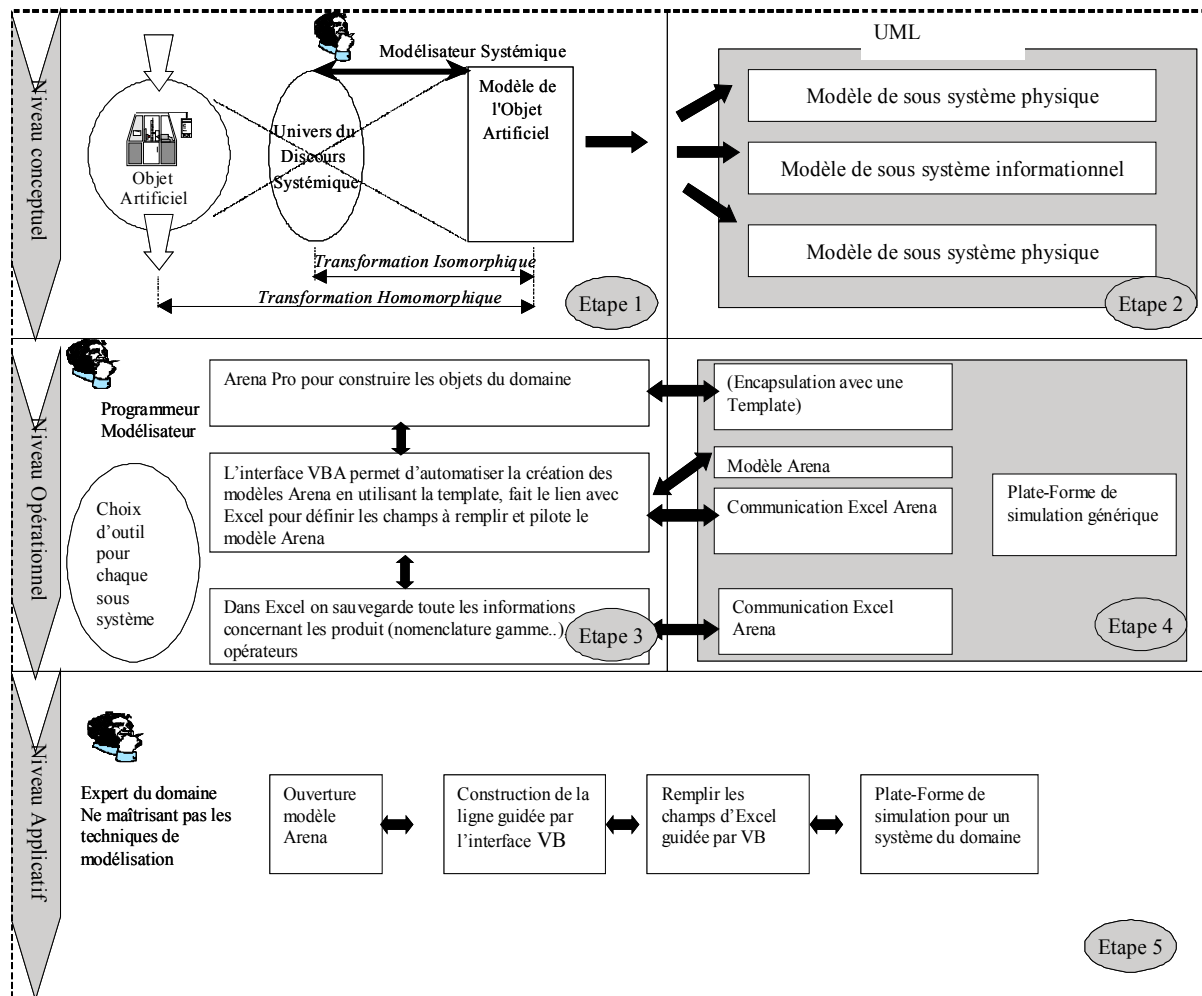


Figure 5: Génération d'une plateforme de simulation

5. Application industrielle : Société Trane

Trane est une société américaine dépendante du groupe American Standard. Elle propose une large gamme de produits de l'industrie du froid allant du simple climatiseur au système réfrigérant pour usine agro-alimentaire. Trane possède 27 sites de production dans le monde, toutes ses usines sont organisées selon la DFT. Ce qui, comme nous l'avons vu, justifie notre travail.

5.1. Application de la méthode : La création d'une ligne

Nous décrivons ici les différentes étapes à suivre pour construire un modèle de simulation d'une ligne et ses feeders dédiés.

Le menu principal utilisateur permet d'ajouter des compétences, de mettre à jour les champs sous Excel, de créer une nouvelle ligne, de calculer le Programme Directeur de Production (PDP) courant, d'ouvrir un fichier txt contenant un PDP, de visualiser la ligne, de modifier la productivité de la ligne, d'ajouter/recréer des opérateurs et enfin de démarrer la simulation.

La création d'une ligne s'effectue en 3 étapes, lors de la première étape on crée les compétences relatives à notre ligne dans le fichier Excel qui sera utilisé ultérieurement pour alimenter les listes déroulantes dans Arena.

Lors de la seconde étape on crée la ligne et les opérateurs sous ARENA. A cette étape, une fenêtre apparaît, demandant le nombre de postes de travail. Ensuite, le menu de création des

postes apparaît n fois. Il ne reste qu'à entrer le nom du poste, définir sa compétence grâce au menu déroulant, déterminer le nombre d'opérateurs maximum qui peuvent y travailler. Il faudra aussi déterminer si le poste est initial, final et combien de feeders il comporte. Les postes se créent successivement, il serait donc logique de les créer dans le même ordre que celui de la ligne physique (les liens entre les postes se font automatiquement).

S'il y a des postes à feeder, un menu apparaîtra qui demandera d'entrer le nombre de postes sur le feeder. Une fois que le feeder est créé, une nouvelle boîte de dialogue apparaîtra pour créer un IPK où il faudra donner le nom de l'IPK, sa taille (un nombre entier ≥ 0) et choisir son poste aval grâce au menu déroulant.

Une fois tous les postes créés, un menu apparaîtra pour créer les groupes d'opérateurs de la ligne. Un groupe d'opérateurs est un ensemble d'opérateurs qui ont les mêmes compétences. Il faudra définir le nombre d'opérateurs dans le groupe ainsi que son nom. Enfin, il faudra définir les caractéristiques du groupe c'est-à-dire ses compétences et l'image qui lui est associée.

Les compétences et l'image associées proviennent de la feuille Excel remplie à la première étape. A ce stade, la ligne physique est construite.

La troisième étape sera réalisée sous Excel, l'enchaînement de toutes ces étapes se fait automatiquement grâce à l'interface VBA. Cette étape consiste à créer une nomenclature des produits, remplir les feuilles de temps de travail pour chaque référence (les champs à remplir sont prédéfinis automatiquement à partir des noms donnés dans Arena pour éviter toute erreur de saisie).

Le plan de production (références + attributs des entités temps, type...) est créé automatiquement il suffit de rentrer les références à produire. Après ces trois étapes la plateforme de la simulation est créée, nous pouvons lancer la simulation.

6. Conclusion :

Dans cet article, nous avons proposé une méthodologie de génération de plates-formes expérimentales en général et en particulier une plateforme générique pour les systèmes de production organisés en DFT. Cette plate-forme a permis de créer en automatique 4 lignes d'assemblages avec leurs feeders (site TRANE en France), l'objectif était de pouvoir utiliser les avantages de la simulation comme support aux différentes méthodes de gestion en entreprise, avec la possibilité offerte aux experts du domaine ne maîtrisant pas les techniques de modélisation de modéliser leurs lignes à moindre coût, et dans un temps court. Aujourd'hui, nous poursuivons le déploiement de cette plate-forme sur l'ensemble des lignes du groupe Trane. Nos futurs travaux s'inscrivent dans le cadre des recherches de l'équipe systèmes contrôlés par le produit du laboratoire du CRAN. Dans ce projet nous considérons que le produit peut être porteur de ses propres informations et donc au lieu de subir des transformations, il peut en être l'acteur. C'est le principe même des produits intelligents issus des architectures tels que HMS (Holonc Manufacturing System) (Koestler, 1967). Dans le domaine de la DFT, cette notion de pilotage par le produit est très développée : c'est l'arrivée du produit qui représentera le signal de travail pour un opérateur et c'est la commande client qui déclenche la fabrication. Cependant, la synchronisation flux matières et flux d'informations n'est pas forcément assurée. Pour remédier à ce problème, les nouvelles technologies d'identification apparues telles le RFID (Lee, 2004) pourront être utiles. Le but de nos prochains travaux sera d'utiliser la simulation comme un outil d'intégration pour cette technologie.

1. Références bibliographiques

- Anglania A., Grieco A., Pacella M., Toliob T (2002) “Object-oriented modeling and simulation of flexible manufacturing systems: a rule-based procedure” *Simulation Modelling Practice and Theory* **10**, pp. 209–234.
- Al-Ahmari A.M.A and Ridgway K. (1999) “An integrated modeling method to support manufacturing system analysis and design” *Computers in Industry* **38**, pp. 225–238.
- De Swaan Arons H. and Boer C.A (2000) “Storage and retrieval of discrete-event simulation model” *Simulation Practice and Theory* **8**, pp. 555–576.
- Costanza J. (1996) *Just-In-Time manufacturing excellence* John Costanza Institute of Technology Inc.; 3rd edition. ISBN: B0006ENXLI.
- El Haouzi H., Thomas A. (2005) “A Methodological Approach to Build Simulation Models of Manufacturing Systems with Distributed Control”, *Proceedings of the International Conference on Industrial Engineering and Systems Management*, Marrakech, Morocco.
- Kelton W., Sadowski R. P. et Sadowski D. A. (1998) *Simulation with Arena*, McGraw-Hill.
- Koestler A. (1967) *The ghost in the machine*, Arkana Books.
- Kovács G.L., S. Kopácsi, J. Nacsá, G. Haidegger and P. Groumpos (1999) “Application of software reuse and object-oriented methodologies for modelling and control of manufacturing systems” *Computers in Industry* **39**.
- Kellert P. et Force C. (1998a), « Méthodologie de Modélisation Orientée Objets de Systèmes de Production – Application à une Chaîne d’ Etuvage de Bobines d’Allumage » *Journal Européen des Systèmes Automatisés*, **32**(1), pp. 107-131.
- Kellert P. et Ruch S. (1998b) « Méthodologie de Modélisation Orientée Objets de Systèmes de Production – Un Processus de Construction/Validation du Modèle Générique Orienté Objets d’un Système de Production ». *Journal Européen des Systèmes Automatisés*, **32**(1), pp. 51-105.
- Lee YM., Cheng F., Leung YT.(2004) “Exploring the impact of RFID on Supply Chain dynamics,” *Proceedings of the Winter Simulation Conference* Institute of Electrical Electronic Engineers, Piscataway, NJ
- Le moigne J-L. (1984) *La théorie du système général*, Presses Universitaires de France.
- Nance R. (1994a) “The Conical Methodology and the Evolution of Simulation Model Development”. *Annals of Operations Research*, **53**, pp. 1-45.
- Nance R. (1994b) “Conical Methodology: An Evolutionary Convergence of Systems and Software Engineering”. *Annals of operations research* **53**
- Perera T. and Liyanage K. (2000) « Methodology for rapid identification and collection of input data in the simulation manufacturing systems » *Simulation Practice and Theory* **7**, pp. 645–656.
- Robinson S, Richard E. Nance, Ray J.Paul, Michael Pidd, Simon J.E Taylor (2004) “Simulation model reuse: definitions, benefits and obstacles” *Simulation Modelling Practice and Theory* **12** pp 479-494.
- Sargent R.G (1986) “Issues in simulation model integration, reusability, and adaptability”, in *Proceedings of the Winter Simulation Conference*, Institute of Electrical Electronic Engineers, Piscataway, NJ, pp.512-516.
- US AIR FORCE (1993a), “Integrated Computer Aided Manufactured Definition Language (IDEF methods)”. Department of Commerce, National Institute of Standards and Technology, Computer Systems Laboratory. Rapport technique, Gaithersburg, USA
- Wyatt D.L (1990) « A framework for reusability using graph-based models », in *Proceeding of the 1990 Winter Simulation Conference*, Institute of Electrical and Electronic Engineers, Piscataway, NJ, pp. 472-476