

CONSERVATOIRE NATIONAL DES ARTS ET MÉTIERS

CENTRE REGIONAL ASSOCIÉ DE ROUEN

ANNEXES

présenté en vue de l'obtention du

DIPLÔME d'INGÉNIEUR C.N.A.M.

SPÉCIALITÉ : INFORMATIQUE

OPTION : SYSTÈMES d'INFORMATION (ISI)

par

El Houssaïne ZELLOU

**MISE EN PLACE D'UN PROGICIEL DE GESTION INTÉGRÉ DANS LE
CADRE D'UNE FUSION D'ENTREPRISE**

Soutenu le..... Mars 2011

Annexe I : Documentation SILOG.

Annexe II : Définition VAN.

Annexe III : Comité de pilotage N°1 du 02/06/2009.

Annexe IV : Comité de pilotage N°4 du 15/01/2010.

Annexe V : Documentation QAD.

Annexe VI : Diagramme général d'activité.

Annexe VII : Définition PMP.

Annexe VIII : Extrait du code Progress.

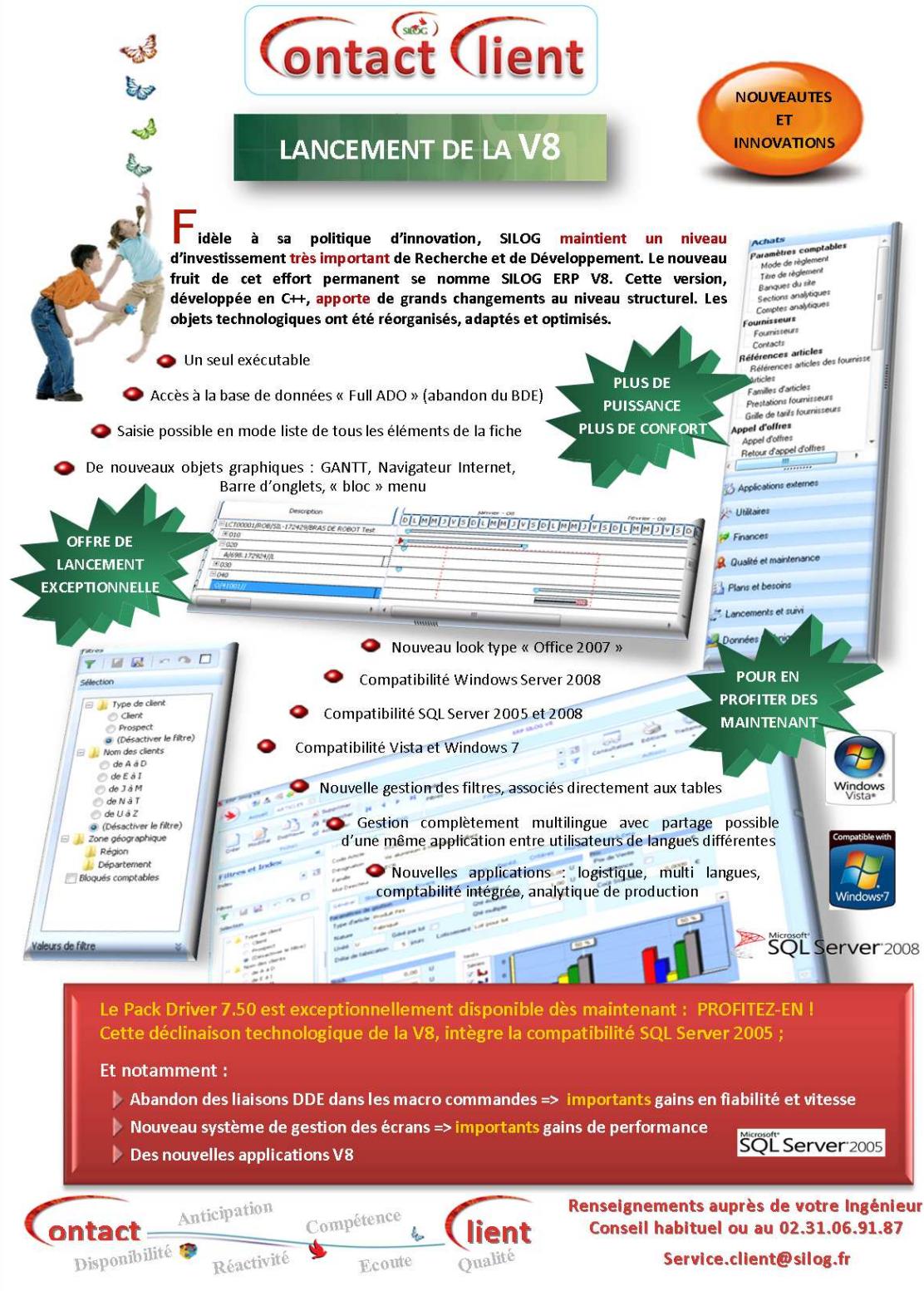
Annexe IX : Arborescence d'une nomenclature de machine.

Annexe X : Édition d'une nomenclature électrique.

Annexe I Documentation SILOG¹

CONTACT CLIENT : BULLETIN D'INFORMATION CLIENTS du GROUPE SILOG

Numéro 3 - Juin 2009



The image is a promotional graphic for the launch of SILOG ERP V8. It features a central title "LANCEMENT DE LA V8" above a green banner. To the left, a boy and girl are shown reaching for butterflies, symbolizing innovation. To the right, a large orange sphere contains the text "NOUVEAUTÉS ET INNOVATIONS". The main text area discusses the software's commitment to innovation and its structural changes. A green starburst highlights "PLUS DE PUISSANCE" and "PLUS DE CONFORT". Another green starburst on the left points to "OFFRE DE LANCEMENT EXCEPTIONNELLE". On the right, a screenshot of the software interface shows various menu items like Achats, Fournisseurs, et Applications externes. Below the interface, compatibility logos for Windows Vista, Windows 7, and Microsoft SQL Server 2008 are displayed. A red callout box at the bottom left promotes the Pack Driver 7.50 and its compatibility with Microsoft SQL Server 2005. The bottom of the page features the "contact client" logo and contact information.

LANCEMENT DE LA V8

NOUVEAUTÉS ET INNOVATIONS

Fidèle à sa politique d'innovation, SILOG maintient un niveau d'investissement très important de Recherche et de Développement. Le nouveau fruit de cet effort permanent se nomme SILOG ERP V8. Cette version, développée en C++, apporte de grands changements au niveau structurel. Les objets technologiques ont été réorganisés, adaptés et optimisés.

- Un seul exécutable
- Accès à la base de données « Full ADO » (abandon du BDE)
- Saisie possible en mode liste de tous les éléments de la fiche
- De nouveaux objets graphiques : GANTT, Navigateur Internet, Barre d'onglets, « bloc » menu

PLUS DE PUISSANCE

PLUS DE CONFORT

OFFRE DE LANCEMENT EXCEPTIONNELLE

Nouveau look type « Office 2007 »

Compatibilité Windows Server 2008

Compatibilité SQL Server 2005 et 2008

Compatibilité Vista et Windows 7

Nouvelle gestion des filtres, associés directement aux tables

Gestion complètement multilingue avec partage possible d'une même application entre utilisateurs de langues différentes

Nouvelles applications : logistique, multi langues, comptabilité intégrée, analytique de production

POUR EN PROFITER DES MAINTENANT

Le Pack Driver 7.50 est exceptionnellement disponible dès maintenant : PROFITEZ-EN !
Cette déclinaison technologique de la V8, intègre la compatibilité SQL Server 2005 ;

Et notamment :

- Abandon des liaisons DDE dans les macro commandes => importants gains en fiabilité et vitesse
- Nouveau système de gestion des écrans => importants gains de performance
- Des nouvelles applications V8

Renseignements auprès de votre Ingénieur Conseil habituel ou au 02.31.06.91.87
Service.client@silog.fr

Anticipation
Disponibilité Réactivité

Compétence
Ecoute Qualité

¹ <https://www.silog.fr>

Annexe II Définition VAN¹

La **valeur actuelle nette** (VAN) est utilisé pour déterminer si un investissement est rentable ou non. Un investissement rentable est un investissement qui accroît la valeur de l'actif économique, c'est à dire qui crée de la richesse. En d'autres termes, il faut que les recettes générées par l'investissement soit supérieur à son coût. Pour calculer la **valeur actuelle nette** (VAN), deux types de flux sont donc nécessaire. Le premier est le montant de l'investissement initial, qui inclut toutes les charges relatives à la réalisation du projet. C'est le coût de l'investissement. Ensuite, il faut évaluer les cash flows générés par ce même investissement pour toutes les périodes futures. Ce sont les recettes. Cette phase est très importante et souvent difficile. En effet, la conjoncture économique évoluant sans cesse, les cash flows peuvent varier de manière importante. Ces cash flows doivent être actualisé en date 0, permettant ainsi de connaître la valeur actuelle des flux futurs. Cette actualisation est indispensable car un euro aujourd'hui ne vaut pas un euro demain.

Ainsi, le calcul de la VAN est le suivant :

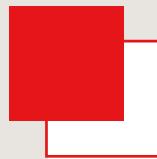
$$\text{VAN} = \text{CF}_1(1+\text{cmpc})^{-1} + \text{CF}_2(1+\text{cmpc})^{-2} + \dots + \text{CF}_n(1+\text{cmpc})^{-n} - \text{Investissement initial.}$$

CMPC étant le coût moyen pondéré du capital.

Le nombre de cash flows nécessaires aux calculs est lui déterminé par la durée de vie de l'investissement.

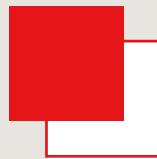
La valeur actuelle nette (VAN) permet à l'entreprise de choisir les projets les plus rentables. Il suffit pour cela de comparer la VAN entre deux projets. Ce critère est le plus utilisé dans le choix d'un investissement par les directeurs financiers des entreprises.

¹ Source : Actufinance.fr



Sopra
group

Annexe III
Comité de pilotage n° 1 du
02/06/2009



Déploiement MFG/PRO sur l'activité AUTOMATION

Comité de pilotage n°1 du 02/06/2009

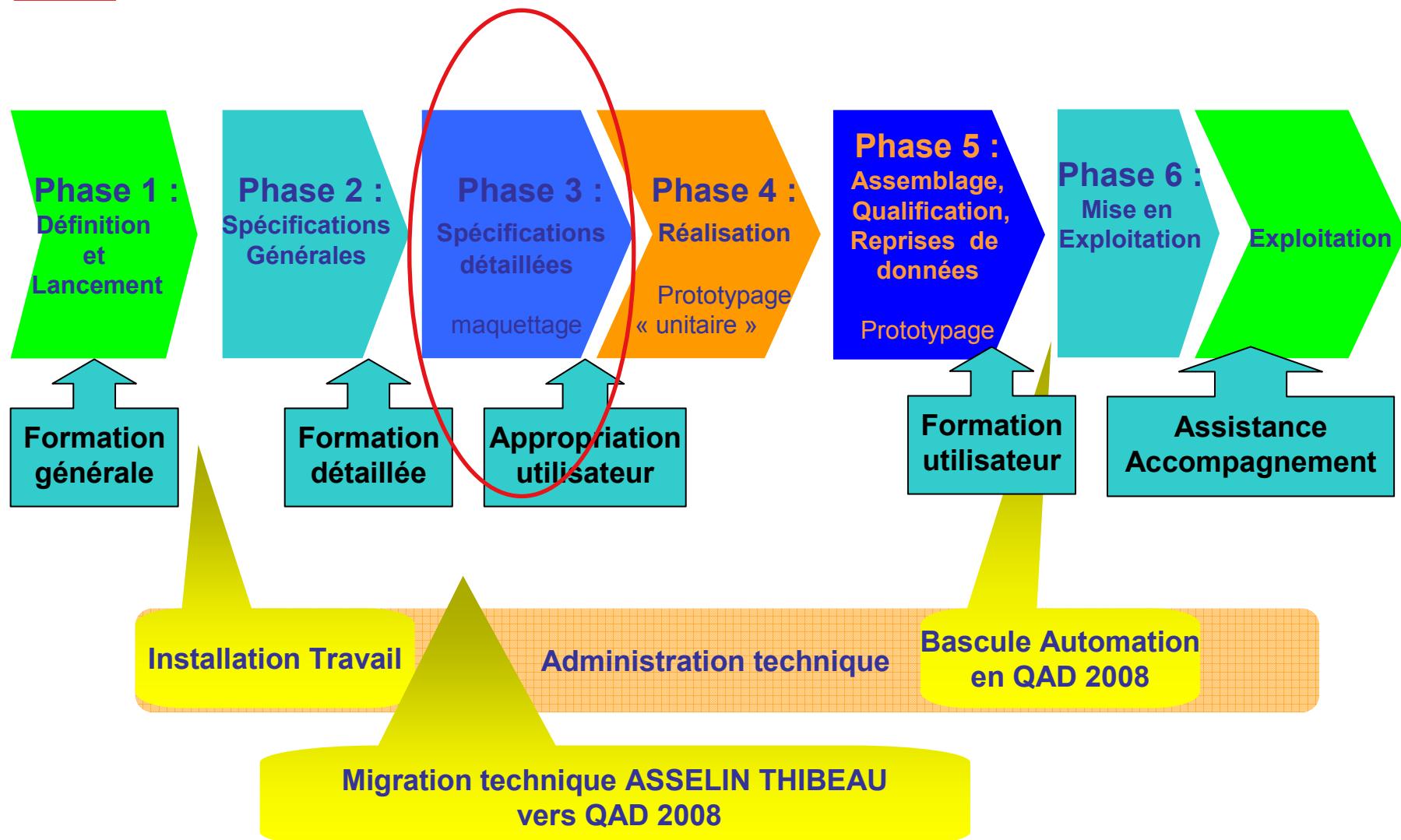
SOMMAIRE

- Rappel des objectifs
- Phases et planning général
- Déroulement du projet
 - Le partage des rôles SOPRA et ASSELIN THIBEAU
 - Le fonctionnement du projet
- Formations initiales
- Migration QAD 2008
- Spécifications Fonctionnelles Générales
- Spécifiques et interfaces identifiées
- Reprise de données
- Suivi budgétaire

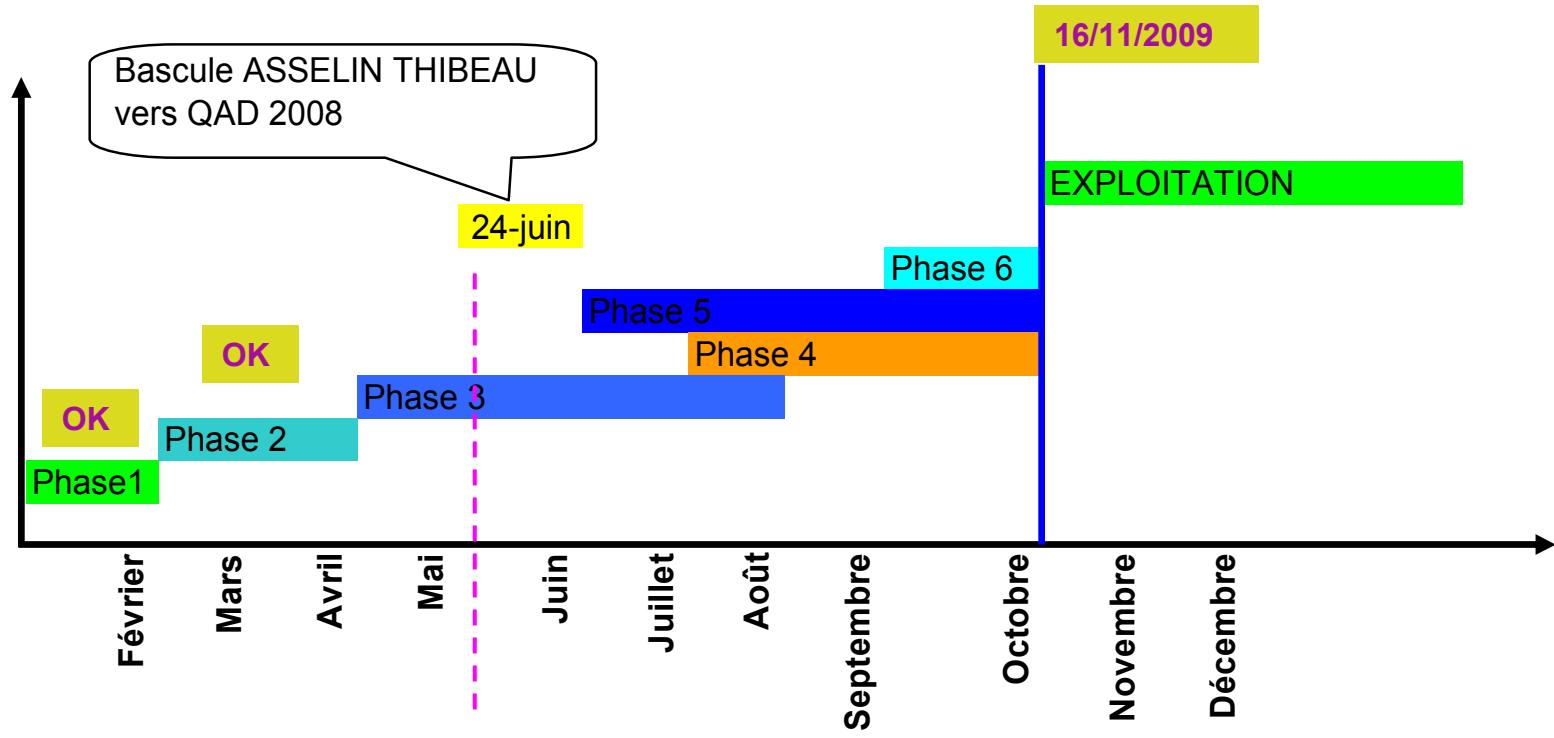
Rappel des objectifs

- Intégrer la gestion de production de automation dans l'ERP ASSELIN-THIBEAU (MFG Pro)
- Éviter les redondances de processus
 - Commandes
 - Ordonnancement/planification
 - Gestion comptable
- Adopter une codification commune aux 3 sites
- Adopter une tarification de vente unifiée
- Éviter les doublons de codification (problèmes de nomenclature)
- Prendre en compte les spécificités du métier « Industrie » et conserver le cœur de métier électrique

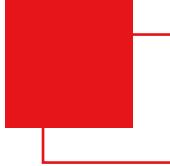
Les Phases du projet



Planning Général : situation au 02/06



Planning détaillé



Déroulement du Projet

■ Rôle de Sopra :

- Mise en place technique de QAD 2008
- Migration technique des spécifiques utilisés dans la précédente version
- Bascule technique vers QAD2008
- Formation à la nouvelle interface utilisateur
- Overview fonctionnelle de la version QAD 2008
- Formalisation des spécifications fonctionnelles générales
- Assistance aux spécifications détaillées de l'activité industrie
- Accompagnement fonctionnel à la demande
- Réalisation des personnalisations éventuelles hors paramétrage via l'interface utilisateur.
- Assistance au pilotage du projet
- Assistance au démarrage si nécessaire

Déroulement du Projet

■ Rôle de ASSELIN THIBEAU :

- Maîtrise d'œuvre du projet
- Installation et recette des spécifiques migrés
- Validation des spécifications générales
- Formations détaillées MFG/PRO sur base de la solution métier
ASSELIN THIBEAU
- Pilotage et réalisation des spécifications détaillées
- Paramétrage de l'interface utilisateur .NET
- Organisation et mise en œuvre du prototype avant démarrage
- Formation des utilisateurs finaux
- Reprise des données de SILOG vers QAD2008
- Pilotage de la bascule vers QAD2008

Déroulement du Projet

■ Le fonctionnement du projet

■ Le Groupe projet

- Disponibilité : correcte dans l'ensemble
- Participation : très bonne implication du groupe
- Evolution des méthodes de travail : en accord avec les processus en place A-T
- Appréciation générale du système : principalement interface caractère , interface graphique à partir du 24 juin

Les formations initiales

- L'overview à permis de survoler l'ensemble des fonctionnalités de MFG Pro.
- La formation s'est faite sur les fonctionnalités qui seront utilisées chez Automation :
 - Articles et nomenclatures
 - Clients et fournisseurs
 - Offres, commandes client, expéditions et facturation
 - Appels d'offres commandes fournisseur, réception et enregistrement des factures
 - OF
 - Gestion des stocks
 - Calcul des coûts standards

Migration QAD 2008

- Livraison des programmes effectuée le 06/05
- Prochaines étapes :
 - Installation des programmes par ASSELIN (d'ici fin Mai)
 - Recette interne des principaux spécifiques
 - Maquetage du projet AUTOMATION sur l'environnement QAD 2008 avec les programmes migrés installés
 - Bascule des sites ELBEUF et TOURCOING sur QAD 2008 le 24 Juin
 - Installation nouvelle interface utilisateur version 2.8 d'ici mi-juin (délai à confirmer)

Spécifications fonctionnelles générales

- Objectif : établir un état des lieux de l'existant, recenser tous les processus mis en jeu et identifier les adéquations et les écarts avec la solution ASSELIN THIBEAU
 - Recueil des besoins effectué début Mars (3 jours) avec les utilisateurs clés AUTOMATION
 - Document validé au mois de Mai
- Premières conclusions :
 - Forte adéquation sur les flux liés aux équipements électriques et pièces de recharge pour machines du groupe
 - Quelques écarts liés à la spécificité du métier électrique : repères topologiques, codes fabricant, description fabricant ...
 - Des impacts sur la gestion des données articles liées à l'harmonisation des codifications (fiches articles/sites)
 - La singularité de l'activité « industrie » tant du point de vue des flux dans le système d'information que de l'organisation humaine :
 - Compromis à trouver entre réactivité et quantité d'information nécessaire au bon fonctionnement du système d'information
 - Problématique du suivi des affaires en temps réel
 - Difficultés du passage d'une organisation fortement « papier » à une organisation davantage pilotée par le système d'information
 - Modélisations à caractériser des différentes activités liées à l'industrie (ventes de prestation, fournitures d'équipements fabriqués ou partiellement sous-traité, fournitures d'équipements totalement sous-traités ...)

Spécifiques et interfaces identifiés

- Spécifiques liés à l'harmonisation des codifications :
 - Création des articles par recopie
 - Mise à jour des données articles / sites
 - Maintenance des données générales articles
 - Rajout de champs pour limiter l'accès des utilisateurs à une seule transaction
- Spécifiques liés au métier électrique :
 - Edition de nomenclature
 - Edition du dossier de fabrication
 - Maintenance des nomenclatures (repère topologique)
- Cette liste sera affinée et finalisée au fur et à mesure des séances de maquetage.

Reprise de données statiques

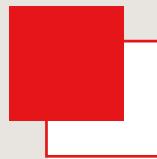
- Codification des articles électriques en cours de définition
- Extraction des données articles (données techniques et données achats) en juin-juillet (WB HZ)
- Mise en forme des données articles sous Excel selon codification définie
 - Attention : cela nécessite au préalable que la codification des articles soit finalisée (notamment, familles électriques à définir par L.VELLY)
- Extraction des nomenclatures après la phase de reprise articles (WB HZ)

Suivi budgétaire

- **Forfait de migration des spécifiques : 15 000 Euros**
 - Terminé
- **Assistance à la mise en œuvre du projet AUTOMATION**
 - Budget : 66 jours
 - Engagés :
 - Consultant technique : 4 jours (RAE : 9 jours)
 - Consultant fonctionnel : 2 jours (RAE : 12 jours)
 - Consultant expert : 14 jours (RAE : 25 jours)
 - Planifiés sur Juin :
 - Consultant expert : 4 jours
 - Consultant technique : 2 jours

Conclusion

- **Questions diverses**
- **Date de prochain comité pilotage : Sem 30 souhaitée (23 ou 24/07 à confirmer)**



Annexe IV Comité de pilotage n° 4 du 15/01/2010



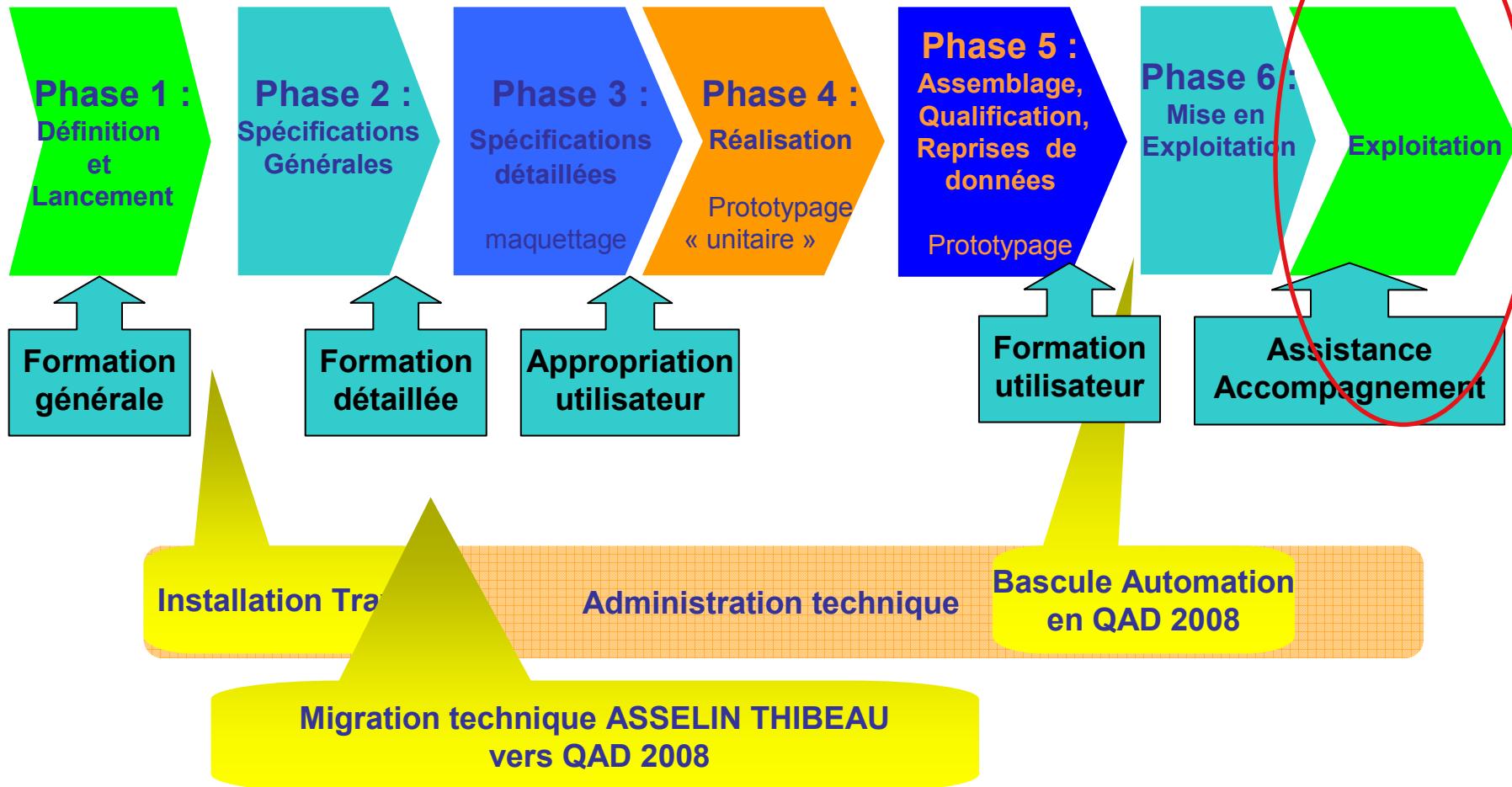
Déploiement MFG/PRO sur l'activité AUTOMATION

Comité de pilotage n°4 du 15/01/2010

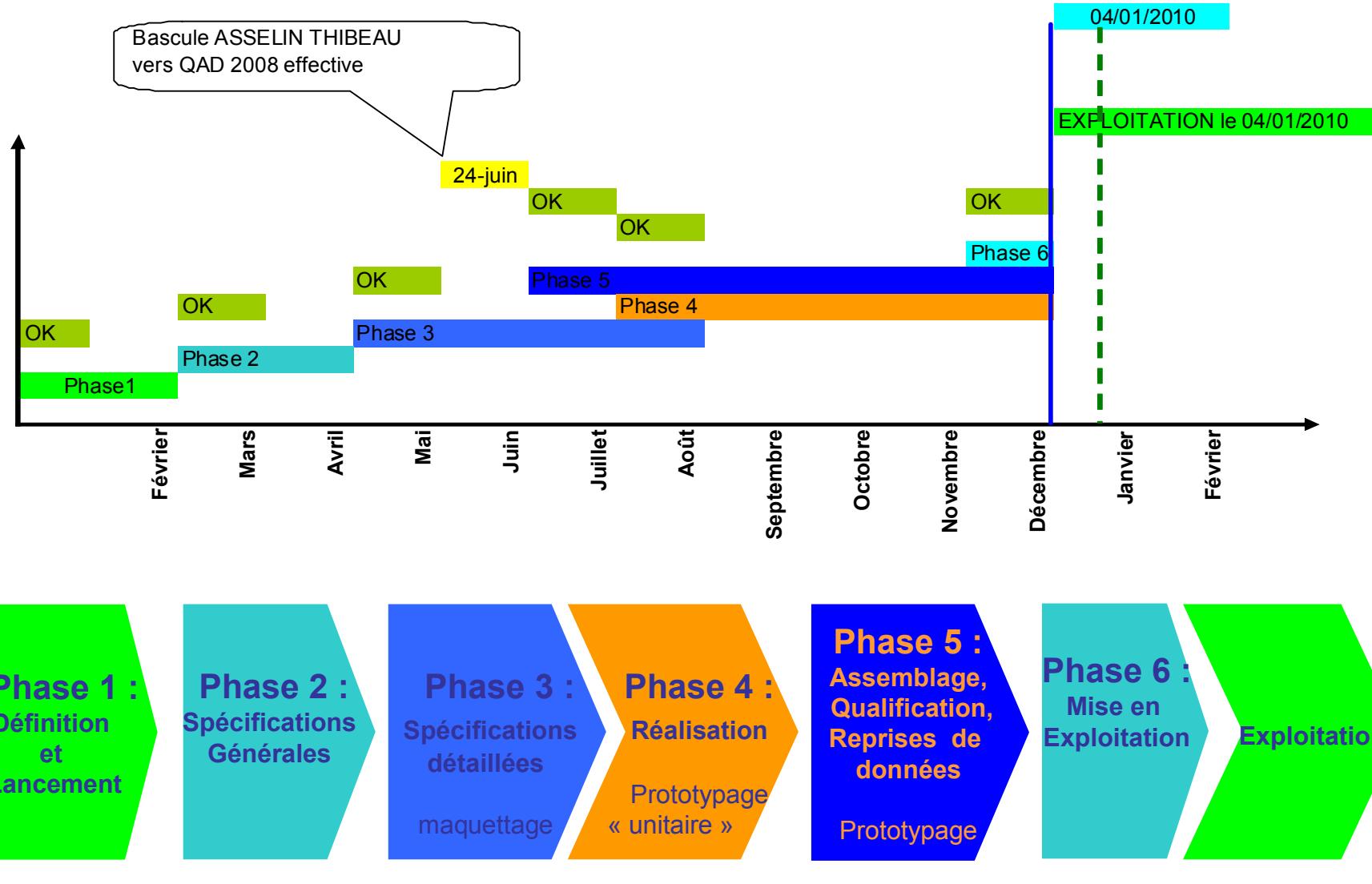
SOMMAIRE

- Phases et planning général
- Déroulement du projet
 - Le fonctionnement du projet
- Spécifiques et interfaces identifiées
- Questions diverses.
- Suivi budgétaire

Les Phases du projet



Planning Général : situation au 15/01/10



BILAN PROJET : Respect des délais

- **Mise en œuvre initialement prévue au 04/01/2010 et avancée au 11/11/2009**
- **Le démarrage a ensuite été recalé au 04/01/2010 suite aux constats suivants :**
 - Reprise de donnée plus conséquente que prévue
 - Utilisateurs non disponibles durant la phase de maquettage et pas suffisamment opérationnels à date
- **Démarrage effectivement réalisé au 04/01/2010**

BILAN PROJET : Reprises de données

- Etalées sur 3 mois (Octobre/Novembre/Décembre)
- Reprises automatiques :
 - 6500 articles (achetés et fabriqués) : données générales, prix par fournisseurs, liens articles - fabricants
 - 10 000 liens de nomenclatures
 - 400 Lignes d'ordres d'achats
 - 1800 lignes de Stocks
- Reprises manuelles :
 - 50 commandes industries
 - 50 OF
 - 100 Gammes
- Restent dans SILOG : uniquement en visualisation sans contrainte de durée
 - Contacts fournisseurs
 - Historiques de données

BILAN PROJET : Démarrage

- **Impression générale** : démarrage maîtrisé sans problèmes majeurs
- **Bilan par services :**
 - **BE** : Opérationnels sur création articles, nomenclatures et DA
 - **Commercial** : flux actuellement géré et maîtrisé par une seule personne (Vanessa)
 - **Approvisionnements** : Opérationnels grâce au support des utilisateurs ASSELIN
 - **Méthodes** : Pris en charge par ASSELIN
 - **Production / ordonnancement** : Opérationnels en collaboration avec le service ORDO ASSELIN
 - **Production / flux matières** : Totalement opérationnel
 - **Production / suivi d'atelier** : Sans problèmes majeurs même si quelques points restent à surveiller (entrée/sorties de stock, projet sur suivi opérateurs)
 - **Comptabilité** : interfaces opérationnelles
 - **Informatique** : Quelques soucis cette semaine à deux reprises, pour la partie .NET (interface utilisateur graphique)

BILAN PROJET : Déroulement

■ Adéquation fonctionnelle :

- Equipements électriques pour machines ASSELIN : utilisation des standards AT (Asselin-Thibeau)
- Activité industrie :
 - Compléments métiers nécessaires (12 J)
 - Utilisation des standards AT pour l'essentiel de l'activité

■ Solutions Métier Automation:

- Prise de commande (génération de la structure comptable cible pour l'activité industrie)
- Création des articles (référence fabricant)
- Gestion des OF (légères adaptation pour articles « Mémo » et opérations de gammes)
- Editions métiers (nomenclatures, liste à servir)
- Interface comptable
- Adaptation technique du SO pour intégrer les temps INGE Automation

BILAN PROJET : déroulement

■ Déroulement Projet :

■ Points positifs :

- Proximité d'utilisateurs confirmés
- Culture ERP existante
- Reprise de processus existant pour la partie industrie avec une bonne participation du pôle industrie
- Prestataire présent
- Esprit positif du personnel Automation
- Transfert de compétence sur le suivi opérateur malgré un timing serré

■ Points négatifs :

- Faible disponibilité de l'équipe projet
- Peu de supports et de procédures formalisés (prévu d'être pris en charge par les utilisateurs)

BILAN PROJET : en cours et reste à faire

- Restitution du suivi d'affaire
 - HZ/GM
- Reprise de nomenclatures spécifiques (Packages SIEMENS sous Excel)
 - H.Z. pour le 18/01
- Contrôle de reprise des stocks
 - HZ/WB (SELF) pour le 18/01
 - LOGISTIQUE : inventaire au fil de l'eau au fur et à mesure des chargements dans la tour
- Contrôle de bon fonctionnement des processus mis en place
 - HZ/Chefs de service : Contrôles ponctuels sur quelques projets cibles (par exemple, FIS FIBER)
 - Vanessa/HZ/Service commercial : Contrôles ponctuels sur une commande réseau IS (service) et une commande réseau IF (fourniture)
 - HZ/GM contrôle comptable et financier sur quelques projets ciblés
- Formations MFG/PRO aux ingénieurs de développement
 - HZ : Sem 5

BILAN après 15 jours de fonctionnement

■ Améliorations et optimisations possibles :

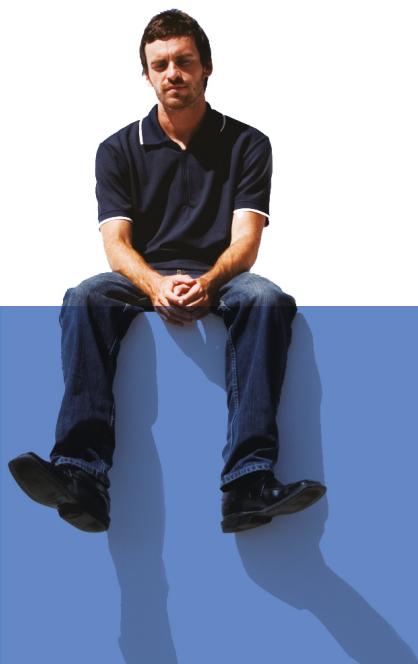
- Augmenter le nombre d'articles en nomenclature (pour diminuer le nombre de SELFS)
- Propositions des utilisateurs à recueillir

Bilan Projet : Respect des coûts

- **Forfait de migration des spécifiques : 15 000 Euros**
 - Terminé
- **Assistance à la mise en œuvre du projet AUTOMATION**
 - Budget : 66 jours
 - Engagés :
 - Consultant technique : 8 jours
 - Consultant fonctionnel : 2 jours
 - Formation DOT.NET : 2 jours
 - Consultant expert : 37 jours
 - Chef de projet : 7 jours
 - Développements : 10 jours



Our Passion. Your Advantage.



**QAD
Enterprise
Applications**

Built For You

Aperçu de QAD Enterprise Applications

Enterprise Financials	Financial Analytics		Management Reporting		Multi-GAAP		Budgeting		Governance, Risk & Compliance					
	Consolidations		Allocations		Financial Shared Services		Credit Management		Tax Management					
	General Ledger	Multi Currency		Accounts Receivable		Accounts Payable		Banking / Cash Management		Cost Management				
	Fixed Assets		Internationalization		Enhanced Controls		Vertex Sales and Use Tax Interface		Logistics Accounting					
Standard Financials	General Ledger	Multi Currency		Accounts Receivable		Accounts Payable		Cash Management		Cost Management				
	Fixed Assets	Enhanced Controls			Vertex Sales and Use Tax Interface			Logistics Accounting						
Customer Management	Sales Quotations		Sales Orders		Sales Analysis			Pricing						
	Customer Self Service			Distributed Order Management			Demand Management							
	Sales Force Automation		Marketing Automation		Trade Management			Configurator						
Manufacturing	Product Data Management			Manufacturing Planning			Manufacturing Execution							
	Lean	Just In Time Sequencing		Planner		Production Scheduler		PRO/PLUS						
	Execution Tools			Automated Data Collection			Compliance Quality							
Supply Chain	Purchasing			Supplier Performance			Quality Management							
	Warehousing		Release Management		Consignment Inventory		Supply Visualization							
	Supply Chain Planning			Transportation Management			Data Synchronization Interface							
Service & Support	Service & Support Management		Mobile Field Service		Field Service Scheduler		Project Realization Management							
Enterprise Asset Management	Projects Controls Management			Plant Maintenance			MRO Management							
Analytics	Reporting and Analysis				Business Intelligence									
Interoperability	QXtend			EDI eCommerce			Product Information Management							
	Document Management			Output Management			Integration Suite							

QAD Enterprise Applications

QAD Enterprise Applications est la dernière évolution de la suite logicielle QAD. QAD Enterprise Applications a été conçue pour répondre aux besoins des industriels dans un environnement économique où le contrôle des coûts, la maîtrise du changement et la profitabilité sont des facteurs de succès critiques.

Développée en collaboration étroite avec nos clients, QAD Enterprise Applications combine richesse fonctionnelle, facilité d'utilisation et flexibilité.

A travers cette dernière version, QAD propose aux industriels une plateforme robuste, capable de s'adapter à des environnements changeants pouvant être administrée par des équipes de support réduites.

QAD Enterprise Applications facilite l'exécution quotidienne des tâches et rend les utilisateurs plus opérationnels et efficaces.

QAD Enterprise Applications est disponible en deux éditions : Standard et Enterprise.

- **QAD Standard Edition** se compose d'un ensemble complet de fonctionnalités destinées à des sociétés mono ou multi sites. Dotée d'une gestion financière supportant les règles internationales, QAD Standard Edition permet une gestion intégrée des flux financiers, des chaînes d'approvisionnement, de la gestion de la demande, des transports et des différents modes de production.
- **QAD Enterprise Edition** s'adresse à des sociétés internationales qui doivent contrôler leur activité simultanément de façon locale, régionale et mondiale. Conçue pour supporter des environnements complexes multi-société, multi-site, multidevise, multilingue, QAD Enterprise Edition dispose de fonctions avancées en matière d'élaboration budgétaire, de consolidation et de reporting.

QAD Enterprise Financial vous permet d'agir en conformité avec les règles financières en vigueur telles que SOX, IFRS ou US GAAP. Cette édition vous permet de disposer d'une vision claire de la situation financière de votre entreprise et de procéder à des analyses sous différents axes, y compris dans le cadre d'environnements multinationaux.

QAD Standard Edition et QAD Enterprise Edition sont intégrées avec les suites de solutions suivantes :

- **QAD Customer Management** – Une gestion efficace et collaborative de l'ensemble des relations commerciales et de la demande.
- **QAD Manufacturing** – Un suivi détaillé des coûts et des flux de production, qu'il s'agisse de productions discrètes (Ordres de Fabrication), répétitives (Programmes) ou lean (Kanban).
- **QAD Supply Chain** – Une collaboration efficace avec les fournisseurs pour améliorer la réactivité de la chaîne d'approvisionnement, optimiser la gestion des flux physiques et des stocks.
- **QAD Service and Support** – Une gestion intégrée de toute l'activité du Service Après-Vente, qu'elle soit organisée autour de centres de support et de réparation locaux ou de techniciens itinérants.
- **QAD Enterprise Asset Management** – La planification et le suivi des activités de maintenance (préventive et curative) associées à la gestion de projets d'investissement.
- **QAD Analytics & Reporting** – La mesure de performance de l'entreprise au regard d'indicateurs clés personnalisés et d'un contrôle de cohérence et de pertinence des données.
- **QAD Interoperability** – Ou comment associer simplement les environnements QAD à d'autres applicatifs déployés dans l'entreprise pour créer un écosystème collaboratif.

QAD Enterprise Applications a été développé sur la base des principes suivants :

- **Facilité d'utilisation** – QAD a particulièrement travaillé sur l'interface utilisateur. Les cartographies de processus interactifs, les capacités de personnalisation, de workflow et d'automatisation rendent les utilisateurs plus performants et l'entreprise plus productive.
- **Une plateforme globale** – Regroupant fonctionnalités et services permettant un déploiement de QAD Enterprise Applications tout autour du monde en respectant les besoins globaux ainsi que les normes, les méthodes et les équipes locales. Les modèles de déploiement centralisés, décentralisés ou hybrides sont supportés.
- **Déploiement** – QAD Enterprise Applications est disponible en mode «On Demand», «On Premise» et «On Appliance».
- **Montée en charge** – QAD Enterprise Applications répond aux besoins d'entreprises de toutes tailles, de la PME à la multinationale, et supporte facilement des environnements de 5 à 5.000 utilisateurs.
- **Gouvernance, risque et conformité** – QAD Enterprise Applications permet aux industriels de piloter leur activité en conformité avec les normes industrielles ou financières.
- **Internationalisation** – Les capacités d'internationalisation de QAD Enterprise Applications assurent la conformité aux législations et exigences réglementaires locales, au sein d'une seule et même solution.

QAD Enterprise Applications a été conçue pour renforcer la performance de ses clients et leur apporter de la valeur ajoutée dans l'exercice de leur activité. Suite complète et intégrée, QAD Enterprise Applications fournit des solutions dans les domaines de la gestion financière, de la gestion des ventes, de la fabrication, de la gestion de la chaîne d'approvisionnement, de la gestion des transports, du service après-vente, de la gestion de la maintenance et de la Business Intelligence. Totalement intégrée, cette suite facilite les transferts d'informations entre les entités opérationnelles, accélère leur traitement et accroît la visibilité.

Les différentes composantes de QAD Enterprise Applications sont détaillées ci-dessous.

QAD Standard Financials

QAD Standard Financials est une suite financière complète qui aide les industriels à rationaliser leurs processus financiers, de la planification à l'enregistrement comptable en passant par le reporting et la gestion de la conformité. Complètement intégré à QAD Standard Edition, QAD Standard Financials a été mis en place avec succès dans plus de 90 pays dans le monde.

QAD Standard Financials comprend de base les fonctionnalités suivantes :

- Comptabilité Générale
- Comptabilité Clients
- Comptabilité fournisseurs
- Multi-devise
- Trésorerie
- Gestion des coûts

De plus, QAD Standard Financials peut s'enrichir des modules supplémentaires suivants :

- QAD Immobilisations
- QAD Comptabilité de la Logistique
- QAD Vertex Sales & Use Tax Interface (USA)
- QAD Contrôles avancés (signature électronique, audit des bases)

QAD Enterprise Financials

QAD Enterprise Financials est la nouvelle composante applicative de gestion financière dans QAD Enterprise Edition. QAD Enterprise Financials permet de gérer des entreprises locales, régionales ou mondiales et supporte les environnements multi-société, multi-site, multidevise, multilingue, multi-taxe, ainsi que la définition de structures de reporting et de consolidation complexes.

QAD Enterprise Financials permet d'assurer la conformité réglementaire, SOX/IFRS et US GAAP et fournit un reporting souple et fiable, permettant aux décideurs d'avoir des vues multiples de la position financière de leur société.

QAD Enterprise Financials est nativement intégrée aux modules Ventes, Distribution, Planification et Production afin de prendre en compte l'impact financier de toutes les activités de l'entreprise.

QAD Enterprise Financials comprend de base les fonctionnalités suivantes :

- Analyse financière
- Gestion du Reporting
- Multi-GAAP
- Création de budget
- Gouvernance, risque et conformité
- Consolidations
- Allocations
- Services Partagés
- Gestion du crédit client
- Gestion des taxes
- Multi-devise
- Comptabilité Générale
- Comptabilité Client
- Comptabilité Fournisseur
- Gestion de Trésorerie
- Gestion des Coûts

De plus, QAD Enterprise Financials peut s'enrichir des modules supplémentaires suivants :

- QAD Immobilisations
- QAD Contrôles avancés (signature électronique, audit des bases)
- QAD Vertex Sales and Use Tax Interface (USA)
- QAD Comptabilité de la Logistique

QAD Customer Management

Piloter les campagnes marketing et en analyser les résultats, connaître précisément son portefeuille d'opportunités et ses prospects, élaborer et réévaluer des prévisions de ventes de façon collaborative, gérer des ventes directes ou par un canal Internet, proposer des produits fortement configurés pour satisfaire les exigences de ses clients. Autant de besoins clés qui peuvent aider l'entreprise à développer ses marchés, fidéliser ses clients et garantir ses revenus dans un contexte fortement concurrentiel.

QAD Enterprise Applications comprend de base les fonctionnalités suivantes :

- QAD CRM
- Gestion des devis
- Gestion des commandes
- Analyse des ventes
- Gestion des tarifications

De plus, QAD Customer Management peut s'enrichir des modules supplémentaires suivants :

- QAD Customer Self Service
- QAD Distributed Order Management
- QAD Demand Management
- QAD Advanced Pricing Medical
- QAD Trade Management
- QAD Configurator

QAD Manufacturing

L'entreprise manufacturière doit se doter d'outils lui permettant d'être plus flexible pour s'adapter plus facilement et à moindre coût aux fluctuations de la demande, aux exigences de ses clients et aux contraintes macro-économiques.

C'est pourquoi, QAD a développé des solutions qui permettent de synchroniser la demande et les flux de production, d'adapter simplement ses modes de production en fonction des ressources ou des tailles de lots, de garantir une traçabilité totale, de se conformer aux meilleures pratiques ou à des normes exigées par vos marchés ou vos clients, de déployer des modèles de production plus fluides et enfin de contrôler précisément les coûts de production.

QAD Enterprise Applications comprend de base les fonctionnalités suivantes :

- Gestion de données techniques
- Planification de la production
- Suivi de la production
 - Discrète
 - Répétitive
 - Flow Manufacturing

De plus, QAD Manufacturing peut être enrichi avec les modules supplémentaires suivants :

- QAD Lean Manufacturing
- QAD Just-In-Time Sequencing
- QAD Production Scheduler
- QAD Pro/Plus
- QAD Planner
- Pilotage des flux par codes barres

QAD Supply Chain

Gérer des flux matières dans des environnements de plus en plus complexes et internationaux, multiplier les sources d'approvisionnement pour contrôler les coûts, héberger des stocks clients, améliorer la collaboration avec les fournisseurs pour fluidifier les échanges d'information, optimiser la gestion des entrepôts, identifier les meilleures conditions de transport sont autant de leviers qui permettent de réduire les cycles d'approvisionnement, les coûts logistiques et les volumes de stock.

QAD Enterprise Applications comprend de base les fonctionnalités suivantes :

- Achats
- Performance fournisseurs
- Gestion de la qualité

De plus, QAD Supply Chain peut s'enrichir avec les modules supplémentaires suivants :

- QAD Warehousing
- QAD Release Management
- QAD Consignment Inventory
- QAD Supply Visualization
- QAD Supply Chain Planning
- QAD Transportation Management System

QAD Service and Support

Un Service Après-Vente performant est une garantie de satisfaction Clients et une source de revenue de plus en plus prisée par les entreprises industrielles. Mais un S.A.V. fonctionne le plus souvent comme une entreprise dans l'entreprise, avec ses propres clients, ses propres ressources, parfois itinérantes, ses propres contraintes contractuelles, ses propres stocks.

QAD Service and Support a été conçu pour administrer l'ensemble des activités du Service Après-Vente tant pour les produits finis simples que pour les produits configurés.

QAD Service and Support couvre les besoins du S.A.V. en termes de suivi des contrats de service, des réclamations clients, de la gestion des centres de réparation et des techniciens itinérants.

QAD Service and Support améliore la satisfaction clients, accélère le flux de facturation des services et donc la trésorerie.

QAD Service and Support est composé des modules suivants :

- QAD Service and Support Management
- QAD Mobile Field Service
- QAD Field Service Scheduler
- QAD Project Realization Management

QAD Enterprise Asset Management

Une ressource de production critique qui tombe en panne et c'est tout le planning de production et de livraison qui est remis en cause. Garantir la continuité de la production en procédant à une maintenance préventive des équipements, suivre le coût des interventions sont donc des activités critiques. Au-delà des seules interventions curatives ou préventives, la gestion des achats de pièces détachées et le suivi de leur niveau de stock donnent aux services maintenance ou travaux neufs une visibilité complète sur leurs besoins et leurs ressources. Enfin, des capacités de gestion des projets d'investissement permettent d'élaborer et de suivre les budgets et les engagements de dépenses correspondant à la mise en œuvre des moyens de production.

QAD Enterprise Asset Management comprend les fonctionnalités suivantes :

- Gestion des moyens de production
- Gestion de la maintenance préventive et curative
- Gestion des achats hors production et des Stocks de pièces de rechange
- Gestion des projets d'investissement

QAD Analytics

Les données utilisées au quotidien sont-elles fiables ? Quelle est la situation par rapport aux indicateurs clés ? Les réponses à ses deux questions permettent de prendre les bonnes décisions rapidement pour ajuster les processus et piloter convenablement l'entreprise.

QAD Analytics permet aux entreprises de simplifier l'exploitation de leurs données, d'en mesurer la qualité et d'accroître la visibilité des indicateurs critiques pour faciliter la prise de décisions.

QAD Enterprise Applications comprend en standard les fonctions suivantes :

- Reporting
- Analyse

De plus, QAD Analytics peut être enrichi par le module supplémentaire suivant :

- QAD Business Intelligence

QAD Interoperability

Toutes les entreprises ont déployé un ensemble de solutions techniques pour satisfaire à des besoins fonctionnels locaux. Faire interagir tous ces systèmes représente souvent un véritable défi tant les développements d'interfaces « point à point » sont longs à mettre en œuvre et leur maintenance complexe. En s'appuyant sur les technologies de communication les plus récentes telles que XML, les Web Services et les ESB, QAD Interoperability fournit un ensemble d'outils qui permet une interopérabilité continue entre les applications tierces et QAD Enterprise Applications. Ces outils utilisent des règles événementielles qui simplifient le processus d'intégration sans codage.

Les solutions QAD Interoperability comprennent :

- QAD Product Information Management (PIM)
- QAD QXtend – Intégration Application-to-Application pour une intégration en temps réel
- QAD EDI eCommerce – Intégration Business-to-Business (B2B) pour l'échange des données entre les partenaires commerciaux
- Sterling Commerce Gentran Integration Suite

Interface Utilisateur

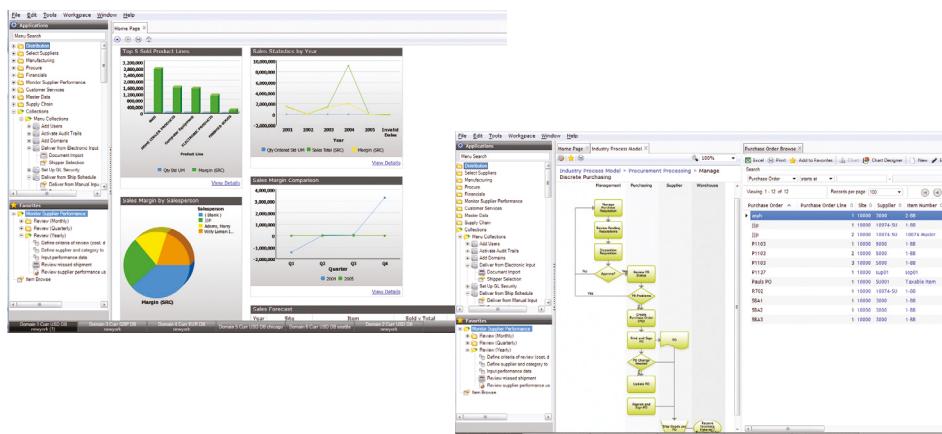
Pour être efficace, un utilisateur a besoin d'accéder aux données de façon simple et rapide. Mais dans un monde où Microsoft Windows™ est devenu la référence, l'interface utilisateur se doit aussi d'être souple et fortement personnalisable.

C'est la raison pour laquelle QAD a développé une interface utilisateur basée sur la technologie Microsoft .NET. Cette interface offre aux utilisateurs des fonctionnalités comme une gestion multifenêtre, des menus arborescents incluant des favoris et des recherches rapides, l'accès direct à des bases de données graphiques ou des sites web.

L'interface .NET, fortement personnalisable sans développement de code, facilite la navigation et l'accès à des ensembles de données, permettant ainsi d'accélérer la recherche d'informations et d'accroître l'efficacité des utilisateurs.

Les .NET Browsers s'intègrent entièrement aux applications de Microsoft Office et permettent d'exporter des données vers Microsoft Excel™ d'un simple click. La présentation sous forme de tableau ou de graphique facilite la compréhension et l'analyse des données sans nécessiter d'export préalable.

Le générateur d'état graphique intégré permet la réalisation de documents personnalisés sous différents formats (PDF, XLS) et leur traitement immédiat ou en mode différé ainsi que leur envoi par email.



Our Passion. Your Advantage.

QAD Enterprise Applications est une solution logicielle complète, dotée d'une interface utilisateur conviviale, qui répond aux besoins des entreprises du secteur industriel.

Pour de plus amples informations, nous invitons à accéder au site internet de QAD <http://www.qad.com>

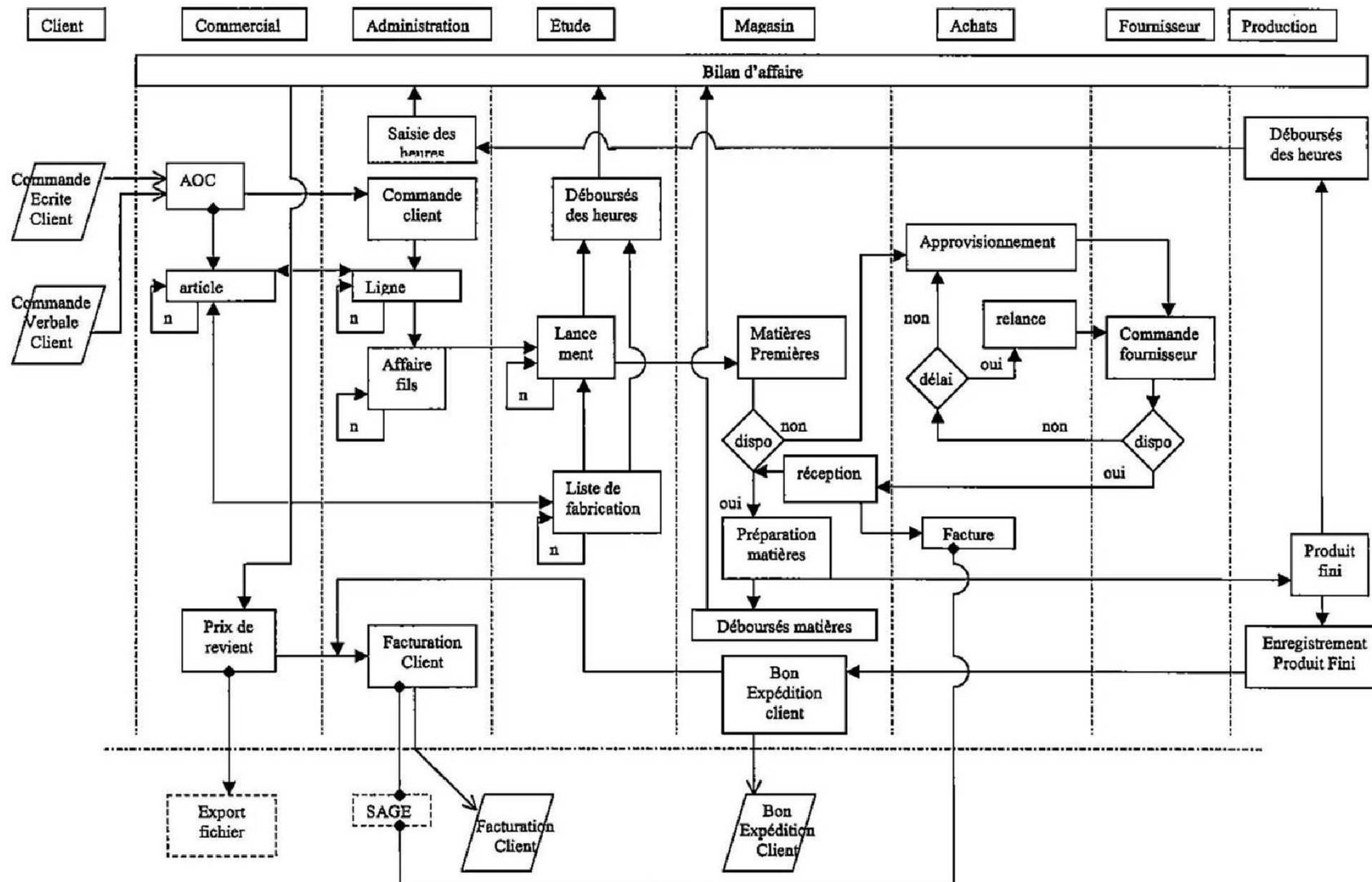
QAD Europe SAS
10 rue Auber
75009 Paris
France

Tel: +33 1 43 12 95 60

www.qad.com



Our Passion. Your Advantage.



Annexe VI : Diagramme général d'activité.

Annexe VII Définition PMP¹

Dans la valorisation des Stocks

Le prix moyen pondéré... est un prix de valorisation qui est calculé par référence aux quantités et à la valeur en stock et à la quantité et le prix unitaire d'un mouvement d'entrée.
La formule est la suivante : $((Q \text{ en stock avant} * \text{Pmp avant}) + (Q \text{ en entrée} * \text{PUHT net})) / (Q \text{ en stock avant} + Q \text{ en entrée})$.

$$\frac{(Nb \text{ de pièces en stock} * PMP) + (Nb \text{ pièces achetées} * DPA)}{Nb \text{ pièces en stock} + Nb \text{ pièces achetées}}$$

Dans la Gestion commerciale

Le prix moyen pondéré est un prix de valorisation qui est calculé par référence aux quantités et à la valeur en stock et à la quantité et le prix unitaire d'un mouvement d'entrée.
La formule est la suivante : $((Q \text{ en stock avant} * \text{Pmp avant}) + (Q \text{ en entrée} * \text{PUHT net})) / (Q \text{ en stock avant} + Q \text{ en entrée})$.

¹ <http://www.lomag-man.org/>

Annexe VIII

```
/*=====
/* ANNEXE VIII      :          EXTRAIT CODE PROGRESS      */
/*=====*/
/*=====
/* PROJET      : DEVELOPPEMENTS SPECIFIQUES ASSELIN-THIBEAU      */
/*=====*/
/* Nomprog      : s1ptmta.p          */
/* MFG/PRO      : QEA 2008.1 SE          */
/* Module       :          */
/* But          : MAINT données article          */
/* Prog. origine : ppptmta.p          */
/* Parametres   :          */
/* Appele par   : s1ptmt04.p          */
/* Run          : s1gpcmmt.p, s1ptmta1.p, s1ptmtc.p          */
/* Include      :          */
/*-----          */
/* N0 modif ! Auteur ! Date ! Commentaires          */
/*-----!-----!-----!*!
/* GA01 ! GA ! 14/05/03 ! Ajout saisie Qté ordre et libellé long */
/* ! ! ! SPC056          */
/* 4472 ! GA ! 08/04/04 ! FE017829 : Traçabilité données de planif */
/* ! ! ! SPC073          */
/* GL01 ! G.L ! 19/09/06 ! Migration eB2.1 SP4          */
/* FC01 ! FC ! 05/03/07 ! Evolution pour Projet THIBEAU */
/* LV01 ! LVA ! 10/04/09 ! Migration QEA 2008.1 SE          */
/* 9922 ! GA ! 02/09/09 ! FE028902: Adaptation Automation */
/* 9924 ! GL ! 02/09/09 ! FE028902: Adaptation Automation */
/* A4L1 ! GA ! 21/04/10 ! FE030224: SPC140 Gestion PVP          */
/*-----          */
/* ppptmta.p - ITEM MAINTENANCE          */
/* Copyright 1986-2008 QAD Inc., Carpinteria, CA, USA.          */
/* All rights reserved worldwide. This is an unpublished work. */
/*!*!
```

This program is called from a variety of places to manage maintenance of some or all item-related data.

The variable PPFORM is used to track:

- From where this program is being called
- Which subset(s) of item master data are valid for maintenance/display

PPFORM may contain any of these values:

- " " : Item Master Maintenance (ppptmt.p)
- "A" : Indicates Engineering Item Maintenance (ececmt.p or ppptmt04.p)
- "B" : Item Inventory Maintenance (ppptmt01.p)
- "C" : Item Planning Maintenance (ppptmt02.p)
- "D" : Item Cost Maintenance (ppptmt03.p)

The following types of activities are allowed or restricted based on ppform:

- 1) New items may be created only when ppform is " " or "A".
- 2) Items may be deleted only when ppform is " ".
- 3) Frame "a" (part number, UM, description) is always displayed. It is only maintainable for ppform = " " or "A". Appearance of other frames is enabled for ppform = " " and the appropriate value listed above.
- 4) PPPTMTA1.P is called to maintain Item Engineering Data.

Annexe VIII

- 5) PPPTMTC.P is called when ppform is " " or "C" to maintain Item Planning Data.
- 6) PPPTMTD.P is called when ppform is " " or "D" to maintain Item Costing Data.
- 7) When ppform is " " and svc_item_data (in 11.24) is set, FSPTMT1.P is called to maintain Item Service Data.

*/

```
/* REVISION: 6.0 LAST MODIFIED: 03/06/90 BY: emb *D001*      */
/* REVISION: 6.0 LAST MODIFIED: 04/23/90 BY: MLB *D021*      */
/* REVISION: 6.0 LAST MODIFIED: 05/10/90 BY: MLB *D024*      */
/* REVISION: 6.0 LAST MODIFIED: 05/17/90 BY: WUG *D002*      */
/* REVISION: 6.0 LAST MODIFIED: 07/02/90 BY: emb *B724*      */
/* REVISION: 6.0 LAST MODIFIED: 07/31/90 BY: WUG *D051*      */
/* REVISION: 6.0 LAST MODIFIED: 08/17/90 BY: MLB *D055*      */
/* REVISION: 6.0 LAST MODIFIED: 08/31/90 BY: WUG *D051*      */
/* REVISION: 6.0 LAST MODIFIED: 09/11/90 BY: WUG *D069*      */
/* REVISION: 6.0 LAST MODIFIED: 10/25/90 BY: MLB *D141*      */
/* REVISION: 6.0 LAST MODIFIED: 10/31/90 BY: emb *D158*      */
/* REVISION: 6.0 LAST MODIFIED: 11/06/90 BY: pml *D184*      */
/* REVISION: 6.0 LAST MODIFIED: 06/10/91 BY: emb *D682*      */
/* REVISION: 7.0 LAST MODIFIED: 08/28/91 BY: pma *F003*      */
/* REVISION: 7.0 LAST MODIFIED: 11/16/91 BY: pml *F048*      */
/* REVISION: 7.0 LAST MODIFIED: 11/28/91 BY: pml *F061*      */
/* REVISION: 6.0 LAST MODIFIED: 01/07/92 BY: WUG *D981*      */
/* REVISION: 7.0 LAST MODIFIED: 01/11/92 BY: RAM *F033*      */
/* REVISION: 7.0 LAST MODIFIED: 03/11/92 BY: WUG *F281*      */
/* REVISION: 7.0 LAST MODIFIED: 05/14/92 BY: tjs *F495*      */
/* REVISION: 7.0 LAST MODIFIED: 05/26/92 BY: pma *F532*      */
/* REVISION: 7.0 LAST MODIFIED: 05/30/92 BY: pma *F560*      */
/* REVISION: 7.0 LAST MODIFIED: 06/09/92 BY: afs *F601*      */
/* REVISION: 7.0 LAST MODIFIED: 06/22/92 BY: emb *F687*      */
/* REVISION: 7.0 LAST MODIFIED: 07/16/92 BY: emb *F777*      */
/* REVISION: 7.0 LAST MODIFIED: 07/23/92 BY: pma *F782*      */
/* REVISION: 7.3 LAST MODIFIED: 10/12/92 BY: tjs *G035*      */
/* REVISION: 7.0 LAST MODIFIED: 10/29/92 BY: pma *G249*      */
/* REVISION: 7.3 LAST MODIFIED: 02/13/93 BY: pma *G032*      */
/* REVISION: 7.3 LAST MODIFIED: 04/16/93 BY: pma *G961*      */
/* REVISION: 7.3 LAST MODIFIED: 04/30/93 BY: pma *GA44***rev only */
/* REVISION: 7.3 LAST MODIFIED: 06/15/93 BY: qzl *GC15***Rev Only */
/* REVISION: 7.3 LAST MODIFIED: 06/29/93 BY: afs *GC22*      */
/* REVISION: 7.3 LAST MODIFIED: 07/29/93 BY: emb *GD82*      */
/* REVISION: 7.3 LAST MODIFIED: 08/05/93 BY: pma *H055*      */
/* REVISION: 7.4 LAST MODIFIED: 08/25/93 BY: dpm *H075*      */
/* REVISION: 7.4 LAST MODIFIED: 02/15/94 BY: pxd *FL60*      */
/* REVISION: 7.2 LAST MODIFIED: 04/07/94 BY: pma *FN30*      */
/* REVISION: 7.2 LAST MODIFIED: 05/31/94 BY: pxd *FO35*      */
/* REVISION: 7.4 LAST MODIFIED: 08/11/94 BY: afs *FQ07*      */
/* REVISION: 7.4 LAST MODIFIED: 09/01/94 BY: bcm *H501*      */
/* REVISION: 7.3 LAST MODIFIED: 09/03/94 BY: bcm *GL93*      */
/* REVISION: 8.5 LAST MODIFIED: 01/05/95 BY: pma *J040*      */
/* REVISION: 8.5 LAST MODIFIED: 01/18/95 BY: taf *J041*      */
/* REVISION: 8.5 LAST MODIFIED: 02/01/95 BY: tjm *J042*      */
/* REVISION: 8.5 LAST MODIFIED: 02/02/95 BY: jlf *J042*      */
/* REVISION: 7.2 LAST MODIFIED: 02/19/95 BY: qzl *F0JL*      */
/* REVISION: 7.4 LAST MODIFIED: 02/23/95 BY: jzw *H0BM*      */
/* REVISION: 7.2 LAST MODIFIED: 03/03/95 BY: qzl *G0G9*      */
/* REVISION: 7.4 LAST MODIFIED: 03/22/95 BY: srk *F0NN*      */
/* REVISION: 8.5 LAST MODIFIED: 03/30/95 BY: dpm *J044*      */
/* REVISION: 7.4 LAST MODIFIED: 05/31/95 BY: qzl *F0SK*      */
```

Annexe VIII

```
/* REVISION: 7.4 LAST MODIFIED: 07/13/95 BY: jzs *G0S6* */
/* REVISION: 8.5 LAST MODIFIED: 07/27/95 BY: ktn *J05Z* */
/* REVISION: 7.4 LAST MODIFIED: 11/14/95 BY: str *G1B0* */
/* REVISION: 8.5 LAST MODIFIED: 01/09/96 BY: wep *J054* */
/* REVISION: 8.5 LAST MODIFIED: 03/19/96 BY: *J0CV* Robert Wachowicz */
/* REVISION: 8.5 LAST MODIFIED: 06/18/96 BY: *J0TY* Robert Wachowicz */
/* REVISION: 8.5 LAST MODIFIED: 07/03/96 BY: *J0XP* Sue Poland */
/* REVISION: 8.6 LAST MODIFIED: 10/10/96 BY: svs *K007* */
/* REVISION: 8.6 LAST MODIFIED: 10/18/96 BY: *K017* Jean Miller */
/* REVISION: 8.6 LAST MODIFIED: 10/11/96 BY: *K003* forrest mori */
/* REVISION: 8.6 LAST MODIFIED: 10/22/96 BY: *K004* Kurt De Wit */
/* REVISION: 8.6 LAST MODIFIED: 10/31/96 BY: *J17C* Sue Poland */
/* REVISION: 8.6 LAST MODIFIED: 11/27/96 BY: *J17Q* Murli Shastri */
/* REVISION: 8.6 LAST MODIFIED: 01/02/97 BY: *K040* Sue Poland */
/* REVISION: 8.6 LAST MODIFIED: 06/09/97 BY: *K0D8* Arul Victoria */
/* REVISION: 8.6 LAST MODIFIED: 06/24/97 BY: *K0DH* Arul Victoria */
/* REVISION: 8.6 LAST MODIFIED: 10/07/97 BY: *J22Q* Nirav Parikh */
/* REVISION: 7.4 LAST MODIFIED: 12/22/97 BY: *H1HN* Jean Miller */
/* REVISION: 8.6E LAST MODIFIED: 05/15/98 BY: *J2LM* Suresh Nayak */
/* REVISION: 8.6E LAST MODIFIED: 02/23/98 BY: *L007* A. Rahane */
/* REVISION: 8.6E LAST MODIFIED: 03/24/98 BY: *J2FG* Niranjan R. */
/* REVISION: 8.6E LAST MODIFIED: 10/04/98 BY: *J314* Alfred Tan */
/* REVISION: 9.0 LAST MODIFIED: 11/10/98 BY: *M01K* Yelena Karnovskaya */
/* REVISION: 9.0 LAST MODIFIED: 11/25/98 BY: *M002* Mayse Lai */
/* REVISION: 9.0 LAST MODIFIED: 12/02/98 BY: *J33T* Thomas Fernandes */
/* REVISION: 9.0 LAST MODIFIED: 03/13/99 BY: *M0BD* Alfred Tan */
/* REVISION: 9.0 LAST MODIFIED: 03/23/99 BY: *J3C4* Satish Chavan */
/* REVISION: 9.1 LAST MODIFIED: 06/07/99 BY: *N005* David Morris */
/* REVISION: 9.1 LAST MODIFIED: 06/17/99 BY: *N00J* Russ Witt */
/* Old ECO marker removed, but no ECO header exists *F0PN* */
/* Revision: 1.38 BY: Satish Chavan DATE: 03/30/00 ECO: *N03T* */
/* Revision: 1.39 BY: Annasaheb Rahane DATE: 05/08/00 ECO: *N0B0* */
/* Revision: 1.44 BY: Satish Chavan DATE: 05/16/00 ECO: *N0B9* */
/* REVISION: 9.1 LAST MODIFIED: 08/17/00 BY: *N0LJ* Mark Brown */
/* Revision: 1.46 BY: Anil Sudhakaran DATE: 05/24/01 ECO: *M0P1* */
/* Revision: 1.47 BY: Mark Christian DATE: 06/05/01 ECO: *M14Z* */
/* Revision: 1.48 BY: Russ Witt DATE: 09/21/01 ECO: *P01H* */
/* Revision: 1.49 BY: Rajiv Ramaiah DATE: 10/16/02 ECO: *N1X6* */
/* Revision: 1.50 BY: Veena Lad DATE: 02/04/03 ECO: *N26B* */
/* Revision: 1.51 BY: Narathip W. DATE: 04/16/03 ECO: *P0PW* */
/* Revision: 1.53 BY: Paul Donnelly (SB) DATE: 06/28/03 ECO: *Q00K* */
/* Revision: 1.56 BY: Jyoti Thatte DATE: 09/15/03 ECO: *P106* */
/* Revision: 1.57 BY: Rajiv Ramaiah DATE: 11/05/03 ECO: *N2M0* */
/* Revision: 1.58 BY: Karan Motwani DATE: 02/23/04 ECO: *N2PR* */
/* Revision: 1.59 BY: Shivanand H DATE: 05/25/04 ECO: *P238* */
/* Revision: 1.60 BY: Abhishek Jha DATE: 12/23/04 ECO: *P2ZV* */
/* Revision: 1.61 BY: Vandna Rohira DATE: 02/14/05 ECO: *P35T* */
/* Revision: 1.61.2.1 BY: Preeti Sattur DATE: 11/28/05 ECO: *P49K* */
/* Revision: 1.61.2.2 BY: Munira Savai DATE: 12/06/05 ECO: *Q0NB* */
/* Revision: 1.61.2.3 BY: Deepak Taneja DATE: 09/20/07 ECO: *P686* */
/* Revision: 1.61.2.6 BY: Kumar Desai DATE: 10/19/07 ECO: *Q1DC* */
/* Revision: 1.61.2.7 BY: Chi Liu DATE: 01/23/08 ECO: *P6GT* */
/* $Revision: 1.61.2.8 $ BY: Alex Joy DATE: 02/28/08 ECO: *P6M7* */
/*-Revision end-----*/

```

```
*****
```

```
/* All patch markers and commented out code have been removed from the source */
```

```
/* code below. For all future modifications to this file, any code which is */
```

```
/* no longer required should be deleted and no in-line patch markers should */
```

```
/* be added. The ECO marker should only be included in the Revision History. */
```

Annexe VIII

```
*****  
/*V8:ConvertMode=Maintenance */  
  
{mfdeclre.i}  
{gplabel.i} /* EXTERNAL LABEL INCLUDE */  
{cxcustom.i "PPPTMTA.P"}  
  
&SCOPED-DEFINE QXO-TABLE pt_mstr  
{qxodef.i &TABLE='&QXO-TABLE'}  
  
{pxmaint.i}  
{pxpgmmgr.i}  
{pxsevcon.i}  
  
/* THE DATABASE FIELD pt_net_wt BELOW WAS FORMERLY pt_weight */  
/* THE DATABASE FIELD pt_net_wt_um BELOW WAS FORMERLY pt_weight_um */  
  
/* NEW SHARED */  
define new shared variable promo_old like pt_promo.  
define new shared variable new_part like mfc_logical.  
define new shared variable esset like cs_set initial "".  
define new shared variable s_mtl like sct_mtl_tl.  
define new shared variable s_lbr like sct_lbr_tl.  
define new shared variable s_bdn like sct_bdn_tl.  
define new shared variable s_ovh like sct_ovh_tl.  
define new shared variable s_sub like sct_sub_tl.  
define new shared variable global_category like sc_category.  
define new shared variable site like si_site initial "".  
define new shared variable undo_all like mfc_logical no-undo.  
  
define new shared variable undo_del as logical.  
define new shared variable frtitle as character format "x(24)".  
define new shared variable inrecno as recid.  
define new shared variable sct1recno as recid.  
define new shared variable sct2recno as character.  
define new shared variable transtype as integer.  
  
define new shared frame a1.  
define new shared frame a2.  
define new shared frame b.  
define new shared frame b1.  
define new shared frame c.  
define new shared frame d.  
define new shared frame d0.  
define new shared frame d1.  
  
define new shared workfile sptwkfl  
fields element like spt_element column-label "Element"  
fields primary like mfc_logical column-label "Pri"  
fields prim2 like mfc_logical  
fields ao like spt_ao column-label "A/O"  
fields cst_tl like spt_cst_tl column-label "This Level"  
fields old_cst_tl like spt_cst_tl  
fields cst_ll like spt_cst_ll column-label "Lower Level"  
fields cst_tot like spt_cst_tl column-label "Total"  
fields cat_desc as character column-label "Category"  
fields seq like spt_pct_ll  
fields part like pt_part.
```

Annexe VIII

```
/* SHARED */
define shared variable ppform as character.

/* LOCAL */
define variable del-yn      like mfc_logical initial yes.
define variable old_site    like pt_site.
define variable msg-nbr     like msg_nbr.
define variable bom_code    like pt_bom_code.
define variable bomValue   like bom_batch.
define variable old_lot_ser like pt_lot_ser.
define variable source_part like pt_part no-undo.
define variable pt_bom_codesv like pt_bom_code.
define variable pt_pm_codesv like pt_pm_code.
define variable pt_networksv like pt_network.
define variable regen_add   like mfc_logical initial no.
define variable part_node   like anx_node.
define variable temp_um     like pt_um.
define variable cfg         like pt_cfg_type format "x(3)" no-undo.
define variable valid_mnemonic like mfc_logical no-undo.
define variable btb-type    like pt_btb_type format "x(8)" no-undo.
define variable btb-type-desc like glt_desc no-undo.
define variable atp-enforcement like pt_atp_enforcement
                                format "x(8)" label "ATP Enforce" no-undo.
define variable atp-enforce-desc like glt_desc no-undo.
define variable l_comm_code  like comd_comm_code no-undo.
define variable error_flag   like mfc_logical      no-undo.
define variable err_mess_no  like msg_nbr        no-undo.
define variable v_std_cost   like sct_cst_tot    no-undo.
define variable v_std_cost_set like sct_sim       no-undo.
define variable l_pt_article like pt_article     no-undo.
define variable l_pt_break_cat like pt_break_cat  no-undo.
define variable l_pt_buyer   like pt_buyer      no-undo.
define variable l_pt_desc1   like pt_desc1      no-undo.
define variable l_pt_desc2   like pt_desc2      no-undo.
define variable l_pt_group   like pt_group      no-undo.
define variable l_pt_part    like pt_part      no-undo.
define variable l_pt_part_type like pt_part_type no-undo.
define variable l_pt_prod_line like pt_prod_line no-undo.
define variable l_pt_site    like pt_site      no-undo.
define variable l_ana_build  like mfc_logical initial no no-undo.

define variable err-flag    as integer.
define variable ps-recno   as recid.
define variable msg         as character no-undo.
define variable apm-ex-prg  as character format "x(10)" no-undo.
define variable apm-ex-sub  as character format "x(24)" no-undo.
define variable cfglabel   as character format "x(24)" label "" no-undo.
define variable l_err_msg   as character no-undo.
define variable item_def_div as character format "x(8)" no-undo.
define variable prgrp1_desc as character format "x(26)" no-undo.
define variable prgrp2_desc as character format "x(26)" no-undo.

/*GA01*/ define      variable v-lib like mfc_logical.
/*GA01*/ define new shared variable v-cmmt as character format "x(76)" extent 3.
/*GA01*/ define new shared frame cmmt.
/*GA01*/ define      variable print_ok like mfc_logical.

/*A4L1*/ define variable famille as character format "x(24)" no-undo.
/*A4L1*/ define variable criterie as logical extent 10 no-undo.
/*A4L1*/ define variable i      as integer no-undo.
```

Annexe VIII

```
/*A4L1*/ define variable old_part as character no-undo.  
/*@MODULE APM BEGIN*/  
define new shared frame f_apmdata.  
  
/* Variable added to perform delete during CIM. Record is deleted  
* Only when the value of this variable is set to "X" */  
define variable batchdelete as character format "x(1)" no-undo.  
  
/* DEFINE NEW SHARED TEMPORARY TABLES */  
{ifttdiv.i "new" }  
{iftpgrp.i "new" }  
{iftpprgp.i "new" }  
{iftptdv.i "new" }  
{iftptpf.i "new" }  
  
{ppacdef.i &type="new shared"}  
  
define variable item_prig1 like tt-prgrp_prig      no-undo.  
define variable item_prig2 like tt-prgrp_prig      no-undo.  
  
{ppptapm1.i} /* DEFINE f_apmdata */  
/*@MODULE APM END*/  
  
/* NEW FORM DEFINITION TO INCLUDE THE batchdelete VARIABLE */  
form {ppptmtaa.i}  
with frame a side-labels width 80 attr-space.  
  
/* SET EXTERNAL LABELS */  
setFrameLabels(frame a:handle).  
  
/*V8!  
on  'entry':U of pt_desc1 in frame a  
  or 'entry':U of pt_desc2 in frame a  
do:  
  if last-event:event-type <> "MOUSE"  
  then  
    self:set-selection(1,25).  
end. /* ON 'ENTRY' OF pt_desc1... */  
*/  
  
/* ITEM MAINTENANCE FRAME INCLUDE FILE */  
form  
/*9924** {ppptmta2.i} **/  
/*9924*/ {s1ptmta2.i}  
with frame a1 title color normal (getFrameTitle("ITEM_DATA",15))  
  side-labels width 80 attr-space.  
  
/* SET EXTERNAL LABELS */  
setFrameLabels(frame a1:handle).  
  
form {ppptmta3.i}  
with frame b title color normal (getFrameTitle("ITEM_INVENTORY_DATA",28))  
  side-labels width 80 attr-space.  
  
/* SET EXTERNAL LABELS */  
setFrameLabels(frame b:handle).  
  
form {ppptmta4.i}  
with frame c title color normal (getFrameTitle("ITEM_PLANNING_DATA",26))
```

Annexe VIII

```
side-labels width 80 attr-space.

/* SET EXTERNAL LABELS */
setFrameLabels(frame c:handle).

form {ppptmta5.i}
with frame d1 title color normal (getFrameTitle("ITEM_PRICE_DATA",23))
    side-labels width 80 attr-space.

/* SET EXTERNAL LABELS */
setFrameLabels(frame d1:handle).

form
{ppptmta9.i}
with frame d ? down title color normal frtitle
    width 80 no-attr-space.

/* SET EXTERNAL LABELS */
setFrameLabels(frame d:handle).

{&PPPTMTA-P-TAG1}
form {ppptmta9.i}
with frame d0 1 down width 80 overlay row startrow no-labels
    title color normal (getFrameTitle("TOTALS",16)) no-attr-space.
{&PPPTMTA-P-TAG2}

form {ppptmt10.i}
with frame b1 title color normal (getFrameTitle("ITEM_SHIPPING_DATA",26))
    side-labels width 80 attr-space.

/* SET EXTERNAL LABELS */
setFrameLabels(frame b1:handle).

/*GA01*/ FORM
/*9922*/ pt_site colon 15
/*9922*/ pt_lot_ser colon 45
/*9922*/ pt_loc colon 58
/*GA01*/ pt_ord_qty colon 15
/*GA01*/ v-lib colon 58
/*GA01*/ with frame lib side-labels width 80 attr-space.

/*GA01*/ /* SET EXTERNAL LABELS */
/*GA01*/ setFrameLabels(frame lib:handle).

/*GA01*/ FORM
/*GA01*/ v-cmmt no-label
/*GA01*/ /*V8!view-as fill-in size 76 by 1 at 2 */
/*GA01*/ with frame cmt title color normal (getFrameTitle("S1-LONG-LABEL",40))
/*GA01*/ side-labels width 80 attr-space.

/*GA01*/ /* SET EXTERNAL LABELS */
/*GA01*/ setFrameLabels(frame cmt:handle).

/*A4L1*/ /* Begin ADD */
form
famille colon 15
critere[2] colon 15
critere[3] colon 15
critere[4] colon 15
critere[5] colon 15
```

Annexe VIII

with frame criteres side-labels width 80 attr-space.

```
/* SET EXTERNAL LABELS */
setFrameLabels(frame criteres:handle).

/*A4L1*/ /* End ADD */

/*@CTRL BEGIN*/
for first soc_ctrl
    fields( soc_domain soc_apm)
        where soc_domain = global_domain no-lock:
end. /* FOR FIRST soc_ctrl */
/*@CTRL END*/

startrow = 6.

/*@MODULE APM BEGIN*/
if soc_apm
then do:

    /*TrM DB NOT CONNECTED */
    {ifapmcon.i "6316" "pause. return"}
    if ppform = "" or
        ppform = "A"
    then do:
        apm-ex-sub = "if" . /* ASSIGN APM INTERFACE SUB-DIRECTORY */
        for first apm_ctrl fields (apm_domain apm_prig1)
            where apm_domain = global_domain no-lock:
        end. /* for first apm_ctrl fields(apm_prig1) no-lock: */
        if not available apm_ctrl
        then do:
            /* 6323 - TrM CONTROL FILE NOT FOUND. PLEASE CREATE USING TrM */
            {pxmsg.i &MSGNUM = 6323
             &ERRORLEVEL = {&APP-ERROR-NO-REENTER-RESULT}
             &PAUSEAFTER = true
            }
            return.
        end. /* if not available apm_ctrl */

    /* LOAD APM PRODUCT GROUPS */
    {gprunex.i
     &module = 'APM'
     & subdir = apm-ex-sub
     &program = 'ifapm015.p'
     &params = "(output error_flag,
                 output err_mess_no) "
    if error_flag
    then do:
        {pxmsg.i &MSGNUM = err_mess_no
         &ERRORLEVEL = {&APP-ERROR-NO-REENTER-RESULT}
         &PAUSEAFTER = true
        }
        return.
    end. /* IF error_flag */

    /* LOAD APM DIVISIONS */
    {gprunex.i
     &module = 'APM'
     & subdir = apm-ex-sub
     &program = 'ifapm004.p'
     &params = "(output error_flag,
```

Annexe VIII

```
        output err_mess_no)" }
if error_flag
then do:
{pxmsg.i &MSGNUM = err_mess_no
 &ERRORLEVEL = {&APP-ERROR-NO-REENTER-RESULT}
 &PAUSEAFTER = true
}
return.
end. /* IF error_flag */

if apm_prig1 = yes
then do:
/* LOAD APM PRICING GROUPS */
{gprunex.i
 &module = 'APM'
 &subdir = apm-ex-sub
 &program = 'ifapm016.p'
 &params = "(output error_flag,
           output err_mess_no)"
if error_flag
then do:
{pxmsg.i &MSGNUM = err_mess_no
 &ERRORLEVEL = {&APP-ERROR-NO-REENTER-RESULT}
 &PAUSEAFTER = true
}
return.
end. /* IF error_flag */
end. /* IF apm_prig1 = Yes */

/* LOAD APM ITEM PROFILE DETAILS */
{gprunex.i
 &module = 'APM'
 &subdir = apm-ex-sub
 &program = 'ifapm020.p'
 &params = "(output error_flag,
           output err_mess_no)"
if error_flag
then do:
{pxmsg.i &MSGNUM = err_mess_no
 &ERRORLEVEL = {&APP-ERROR-NO-REENTER-RESULT}
 &PAUSEAFTER = true
}
return.
end. /* IF error_flag */

end. /* IF PPFORM = "" OR PPFORM = "A" */

end. /* IF soc_apm */
/*@MODULE APM END*/
for each temp_inactive
  exclusive-lock:
    delete temp_inactive.
end. /* FOR EACH temp_inactive */

/*4472*/ {gpaud.i &uniq_num1 = 01   &uniq_num2 = 02
          &db_field = pt_mstr &db_field = pt_part}

/*DISPLAY*/
mainloop:
```

Annexe VIII

repeat:

```
/*4472*/ loopa: do:  
  
/* IF WE'VE ADDED A NEW PART & AUTOGEN IS ON, ADD IT TO ANX */  
for first pic_ctrl  
    fields( pic_domain pic_item_regen pic_qadc01)  
        where pic_domain = global_domain no-lock:  
    end. /* FOR FIRST pic_ctrl */  
  
if available pic_ctrl  
then do:  
  
    if pic_item_regen and regen_add  
    then do:  
        {gprun.i ""ppptgen.p"" "(input ""6"", input part_node)"}  
        regen_add = no.  
    end.  
  
    if substring(pic_qadc01,2,1) = "Y" and l_ana_build  
    then do:  
        for each anx_det  
            where anx_domain = global_domain  
            and anx_node = part_node  
            and anx_type = "6" exclusive-lock:  
  
/* CREATING TEMP TABLE FOR INACTIVE NODES */  
if not anx_active  
then do:  
    create temp_inactive.  
    assign  
        temp_inactive_node = part_node  
        temp_inactive_code = anx_code.  
    end. /* IF not anx_active */  
  
    delete anx_det.  
end. /* FOR EACH ANX_DET */  
  
{gprun.i ""ppptgen.p"" "(input ""6"", input part_node)"}  
l_ana_build = no.  
end.  
end. /* IF AVAIL PIC_CTRL */  
  
hide frame a1 no-pause.  
hide frame b no-pause.  
hide frame b1 no-pause.  
hide frame c no-pause.  
  
clear frame d no-pause.  
hide frame d1 no-pause.  
  
hide frame d0 no-pause.  
hide frame d no-pause.  
  
/*GA01*/ hide frame lib no-pause.  
/*A4L1*/ hide frame criteres no-pause.  
  
view frame a.  
if ppform = "" or ppform = "a"  
then
```

Annexe VIII

```
view frame a1.  
if ppform = "b" or ppform = ""  
then  
    view frame b.  
if ppform = "c"  
then  
    view frame c.  
  
/*GA01*/ if ppform = "a" then view frame lib.  
  
frtitle = getFrameTitle("GENERAL_LEDGER_COST_DATA",18).  
  
if ppform = "d"  
then do:  
    view frame d1.  
    view frame d.  
end. /* if ppform = "d" then do: */  
  
do transaction with frame a on endkey undo, leave mainloop:  
  
view frame a.  
if ppform = "" or ppform = "a"  
then  
    view frame a1.  
if ppform = "" or ppform = "b"  
then  
    view frame b.  
if ppform = "b"  
then  
    view frame b1.  
  
if ppform = "c"  
then  
    view frame c.  
if ppform = "d"  
then do:  
    view frame d1.  
  
frtitle = getFrameTitle("GENERAL_LEDGER_COST_DATA",18).  
view frame d.  
end. /* if ppform = "d" then do: */  
  
/*GA01*/ if ppform = "a" then view frame lib.  
  
/* INITIALIZE batchdelete VARIABLE */  
batchdelete = "".  
  
/* PROMPT-FOR batchdelete VARIABLE ONLY DURING CIM */  
prompt-for pt_mstr.pt_part  
    batchdelete no-label when (batchrun) space(0)  
    editing:  
  
/* SET GLOBAL ITEM VARIABLE */  
{pxrun.i &PROC = 'setSystemDefaultItem' &PROGRAM = 'ppitxr.p'  
    &PARAM = "(input pt_part:SCREEN-VALUE)"  
    &NOAPPEROR = true  
    &CATCHERROR = true  
}  
  
new_part = no.
```

Annexe VIII

```
/* FIND NEXT/PREVIOUS RECORD */
{mfnp.i pt_mstr pt_part " pt_mstr.pt_domain = global_domain and
    pt_part " pt_part pt_part pt_part}

if recno <> ?
then do:
    display pt_part pt_desc1 pt_desc2 pt_um
    with frame a.

    if ppform = "" or ppform = "a"
    then
        display
        pt_draw
        pt_rev
/*9924** pt_dsgn_grp */
    pt_dwg_loc
    pt_dwg_size
    pt_prod_line
    pt_group
    pt_added
    pt_part_type
    pt_status
    pt_break_cat
    pt_promo
/*9924*/ pt_model
    with frame a1.

/*GA01*/ if ppform = "a" then do:
/*GA01*/     print_ok = no.
/*GA01*/     for first cd_det no-lock
/*GA01*/     where cd_ref = pt_part
/*GL01*/     and cd_domain = global_domain
/*GA01*/     and cd_lang = global_user_lang
/*GA01*/     and cd_type = ""
/*GA01*/     and cd_seq = 0:
/*GA01*/         assign v-cmmt[1] = cd_cmmt[1]
/*GA01*/         v-cmmt[2] = cd_cmmt[2]
/*GA01*/         v-cmmt[3] = cd_cmmt[3]
/*GA01*/         print_ok = yes.
/*GA01*/     end.
/*9922*/     display pt_site pt_lot_ser pt_loc with frame lib.
/*GA01*/     display pt_ord_qty available cd_det @ v-lib with frame lib.
/*GA01*/     if print_ok then display v-cmmt with frame cmt.
/*GA01*/             else hide frame cmt no-pause.
/*GA01*/ end. /* if ppform = "a" */

if ppform = "" or ppform = "b"
then do:
    display
    pt_abc
    pt_lot_ser
    pt_site
    pt_sngl_lot
    pt_critical
    pt_loc
    pt_loc_type
    pt_auto_lot
    pt_rctpo_status
    pt_rctpo_active
```

Annexe VIII

```
pt_lot_grp
pt_rctwo_status
pt_rctwo_active
pt_article
pt_avg_int
pt_cyc_int
pt_shelflife
with frame b.
end /* do: */

{pxrun.i &PROC = 'processRead' &PROGRAM = 'ppccxr1.p'
 &PARAM = "(input pt_part,
 buffer comd_det,
 input {&NO_LOCK_FLAG},
 input {&NO_WAIT_FLAG})"
 &NOAPPERROR = true
 &CATCHERROR = true
}

if return-value = {&SUCCESS-RESULT}
then
    l_comm_code = comd_comm_code.
else
    l_comm_code = "".

if ppform = "b"
then
    display
    l_comm_code
    pt_fr_class
    pt_ship_wt pt_ship_wt_um
    pt_net_wt pt_net_wt_um
    pt_size pt_size_um
with frame b1.

if ppform = "c"
then do:

    {pxrun.i &PROC = 'getEmtDescription' &PROGRAM = 'soemxr.p'
     &PARAM = "(input pt_btb_type,
                 output btb-type,
                 output btb-type-desc)"
     &NOAPPERROR = true
     &CATCHERROR = true
    }

    {pxrun.i &PROC = 'getATPDescription' &PROGRAM = 'soatpxr.p'
     &PARAM = "(input pt_atp_enforcement,
                 output atm-enforcement,
                 output atm-enforce-desc)"
     &NOAPPERROR = true
     &CATCHERROR = true
    }

define buffer inmstr4 for in_mstr.

{pxrun.i &PROC = 'processRead' &PROGRAM = 'icinxr.p'
 &PARAM = "(input pt_part,
           input pt_site,
           buffer inmstr4,
```

Annexe VIII

```
input {&NO_LOCK_FLAG},  
input {&NO_WAIT_FLAG})"  
&NOAPPERROR = true  
&CATCHERROR = true  
}  
  
display inmstr4.in_mrp when (available inmstr4) @  
pt_mrp  
pt_cum_lead  
pt_ms  
pt_plan_ord  
pt_ord_pol  
pt_ord_qty  
pt_ord_per  
pt_sfty_stk  
pt_sfty_time  
pt_rop  
pt_rev  
pt_buyer  
pt_vend  
pt_po_site  
pt_pm_code  
pt_mfg_lead  
pt_pur_lead  
pt_insp_rqd  
pt_insp_lead  
pt_timefence  
pt_iss_pol  
pt_phantom  
pt_ord_min  
pt_ord_max  
pt_ord_mult  
pt_op_yield  
pt_yield_pct  
pt_run pt_setup  
btb-type  
pt_qad15  
pt_network  
pt_routing  
pt_bom_code  
pt_run_seq1  
pt_run_seq2  
atp-enforcement  
pt_atp_family  
with frame c.  
  
{pxrun.i &PROC = 'getConfigTypeDesc' &PROGRAM = 'ppitxr.p'  
&PARAM = "(input pt_cfg_type,  
output cfg,  
output cfglabel)"  
&NOAPPERROR = true  
&CATCHERROR = true  
}  
display cfg with frame c.  
  
{pxrun.i &PROC = 'processRead' &PROGRAM = 'bmbmxr.p'  
&PARAM = "(input (if pt_bom_code <>  
then  
    pt_bom_code  
else
```

Annexe VIII

```
        pt_part),
buffer bom_mstr,
input {&NO_LOCK_FLAG},
input {&NO_WAIT_FLAG})"
&NOAPPERROR = true
&CATCHERROR = true
}

{pxrun.i &PROC = 'getBomBatchValue' &PROGRAM = 'bmbmxr.p'
&PARAM = "(buffer bom_mstr,
    output bomValue)"
&NOAPPERROR = true
&CATCHERROR = true
}

display
    bomValue
    @ pt_batch with frame c.

end. /* if ppform = "c" then do: */
if ppform = "d"
then
display pt_price
    pt_taxc pt_taxable
with frame d1.

end. /* IF recno */
end. /* PROMPT-FOR...EDITING */

/* ADD/MOD/DELETE */
del-yn = no.

{pxrun.i &PROC = 'processRead' &PROGRAM = 'ppitxr.p'
&PARAM = "(input pt_part:SCREEN-VALUE,
    buffer pt_mstr,
    input {&LOCK_FLAG},
    input {&WAIT_FLAG})"
&NOAPPERROR = true
&CATCHERROR = true
}
if return-value <> {&SUCCESS-RESULT}
then do:
    /* ADD ONLY ON ENGINEERING MAINT */
    if ppform <> ""
/*9922**      and ppform <> "a"      **/
    then do:
        /* 16 - ITEM NUMBER DOES NOT EXIST */
        {pxmsg.i &MSGNUM = 16
        &ERRORLEVEL = {&APP-ERROR-RESULT}
        }
        undo mainloop.
    end. /* IF ppform <> "" AND ppform <> "a" then do: */

{pxrun.i &PROC = 'validateCreate' &PROGRAM = 'ppitxr.p'
&PARAM = "(input pt_part:SCREEN-VALUE)"
&NOAPPERROR = true
&CATCHERROR = true
}

if return-value <> {&SUCCESS-RESULT}
```

Annexe VIII

```
then do:  
    undo,retry.  
end. /* IF return-value <> {&SUCCESS-RESULT} */  
  
else if return-value = {&SUCCESS-RESULT}  
then do:  
  
/* 1 - ADDING NEW RECORD */  
{pxmsg.i &MSGNUM = 1  
    &ERRORLEVEL = {&INFORMATION-RESULT}  
}  
  
{pxrun.i &PROC = 'processCreate' &PROGRAM = 'ppitxr.p'  
    &PARAM = "(input pt_part:SCREEN-VALUE,  
        input ",  
        buffer pt_mstr)"  
    &NOAPPERROR = true  
    &CATCHERROR = true  
}  
  
if return-value <> {&SUCCESS-RESULT}  
then do:  
    undo,retry.  
end. /* IF return-value <> {&SUCCESS-RESULT} */  
  
new_part = yes.  
  
/*GA01*/ pt_ord_qty = 1.  
  
end. /* IF return-value = {&SUCCESS-RESULT} */  
  
end. /* IF return-value <> {&SUCCESS-RESULT} */  
  
/* STORE MODIFY DATE AND USERID FOR AN ITEM CREATED/MODIFIED */  
{pxrun.i &PROC = 'setModificationInfo' &PROGRAM = 'ppitxr.p'  
    &PARAM = "(buffer pt_mstr)"  
    &NOAPPERROR = true  
    &CATCHERROR = true  
}  
  
/*4472*/ {gpaud1.i &uniq_num1 = 01 &uniq_num2 = 02  
    &db_file = pt_mstr &db_field = pt_part}  
  
assign  
    pt_bom_codesv = pt_bom_code  
    pt_pm_codesv = pt_pm_code  
    pt_networksv = pt_network  
    promo_old = pt_promo  
    recno = recid(pt_mstr)  
    pt_recno = recid(pt_mstr).  
  
do on error undo mainloop, leave mainloop:  
  
/* DISPLAY */  
display pt_desc1 pt_desc2 pt_um  
with frame a.  
  
if ppform = "" or ppform = "a"  
then  
    display pt_prod_line pt_added pt_group pt_rev
```

Annexe VIII

```
        pt_part_type pt_draw pt_status
/*9924** pt_dsgn_grp */
        pt_dwg_loc pt_dwg_size
        pt_break_cat
        pt_promo
/*9924*/ pt_model
with frame a1.

/*GA01*/ if ppform = "a" then do:
/*GA01*/ print_ok = no.
/*GA01*/ for first cd_det no-lock
/*GA01*/ where cd_ref = pt_part
/*GL01*/ and cd_domain = global_domain
/*GA01*/ and cd_lang = global_user_lang
/*GA01*/ and cd_type = ""
/*GA01*/ and cd_seq = 0:
/*GA01*/ assign v-cmmt[1] = cd_cmmt[1]
/*GA01*/      v-cmmt[2] = cd_cmmt[2]
/*GA01*/      v-cmmt[3] = cd_cmmt[3]
/*GA01*/      print_ok = yes.
/*GA01*/ end.
/*9922*/ display pt_site pt_lot_ser pt_loc with frame lib.
/*GA01*/ display pt_ord_qty (available cd_det or new_part) @ v-lib
/*GA01*/ with frame lib.
/*GA01*/ if print_ok then display v-cmmt with frame cmt.
/*GA01*/ else hide frame cmt no-pause.
/*GA01*/ end. /* if ppform = "a" */

if ppform = ""
then
    display
    pt_abc
    pt_lot_ser
    pt_site
    pt_sngl_lot
    pt_critical
    pt_loc
    pt_loc_type
    pt_auto_lot
    pt_lot_grp
    pt_rctpo_status
    pt_rctpo_active
    pt_rctwo_status
    pt_rctwo_active
    pt_article
    pt_avg_int
    pt_cyc_int
    pt_shelflife
with frame b.

temp_um = input pt_um.

/* SET */
if ppform = "" or ppform = "a"
then do:
    {pxrun.i &PROC = 'p-assign-msg'
     &PARAM = "(output l_err_msg)"
     &NOAPPERROR = true
     &CATCHERROR = true
    }
```

Annexe VIII

```
if not regen_add
then
    assign
    l_pt_desc1 = pt_desc1
    l_pt_desc2 = pt_desc2.

/* ADDED THE FOLLOWING CODE FOR GUI TO SEPARATE THE EDITING */
/* FROM THE SET STATEMENT AS IT WAS TRAPPING THE 'CTRL-C' */
/* AND THROWING THE USER OUT INSTEAD OF COPYING */

if ppform = ""
then
    ststatus = stline[2].
else
    ststatus = stline[3].
status input ststatus.

set1:
do on error undo, retry:

set pt_um
validate( {gpcode.v pt_um }, l_err_msg)
text(pt_desc1 pt_desc2)
go-on ("F5" "CTRL-D")
with frame a.

/* DELETE */
if ppform <> ""
and ( lastkey      = keycode("F5")
      or lastkey      = keycode("CTRL-D")
      or input batchdelete = "x")
then do:
/* DELETE NOT ALLOWED */
{pxmsg.i &MSGNUM = 1021
 &ERRORLEVEL = "3"}
undo set1,retry set1.
end. /*IF ppform <> "AND..."*/

if ppform = ""
and (lastkey      = keycode("F5")
      or lastkey      = keycode("CTRL-D")
      or input batchdelete = "x")
then do:
    del-yn = yes.

/* 11 - PLEASE CONFIRM DELETE */
{pxmsg.i &MSGNUM = 11
 &ERRORLEVEL = {&INFORMATION-RESULT}
 &CONFIRM = del-yn
}

if not del-yn
then
    undo set1, retry set1.
end. /* IF ppform = "" AND ... */

end. /* SET1 */
```

Annexe VIII

```
if not regen_add
  and (l_pt_desc1 <> pt_desc1
    or l_pt_desc2 <> pt_desc2)
then
  assign part_node = pt_part
  l_ana_build = yes.

end. /* IF ppform = "" ... */

/* DELETE */
if
  del-yn = yes or
  input batchdelete = "x"
then do:

  {gprun.i 'ppptdel.p'
    "(input pt_recno,
     output undo_del,
     output msg)"
  }
  if undo_del
  then
    next mainloop.

run fireOutboundEvent (input rowid({&QXO-TABLE}),
                      input oid_{&QXO-TABLE},
                      input 'DELETE').

{pxrun.i &PROC      = 'processDelete' &PROGRAM = 'ppitxr.p'
 &PARAM      = "(buffer pt_mstr)"
 &NOAPPERROR = true
 &CATCHERROR = true
}
if return-value <> {&SUCCESS-RESULT}
then
  next mainloop.

clear frame a.
del-yn = no.
next mainloop.
end. /* IF del-yn = yes */

assign qxo_write = true.

status input.

if pt_um <> temp_um
then do:
  {pxrun.i &PROC = 'validateUMInTransHist' &PROGRAM = 'ppitxr.p'
   &PARAM = "(input pt_part)"
   &NOAPPERROR = true
   &CATCHERROR = true
  }
end. /* IF pt_um <> temp_um */

/*@MODULE APM BEGIN*/
if ppform <> "b" and
  ppform <> "c" and
  ppform <> "d"
then
```

Annexe VIII

```
if soc_apm and
  (promo_old <> "" or pt_promo <> "")
then do:
  /* Future logic will go here to determine subdirectory*/
  apm-ex-prg = "ifprod.p".
  apm-ex-sub = "if".

  for first si_mstr
    fields( si_domain si_site si_cur_set si_gl_set)
      where si_domain = global_domain and si_site = pt_site
    no-lock:
  end. /* FOR FIRST si_mstr */

/* RUN INTERNAL PROCEDURE TO OBTAIN COST SET AND COST */
{pxrun.i &PROC = 'get_std_cost'
 &PARAM = "(input pt_part,
            input si_site,
            output v_std_cost,
            output v_std_cost_set)"
&NOAPPERROR = true
&CATCHERROR = true
}

/* UPDATE GENERIC ITEM RECORD IN APM */
{gprunex.i
  &module = 'APM'
  &subdir = apm-ex-sub
  &program = 'ifapm054.p'
  &params = "(input pt_part,
              input pt_desc1,
              input pt_desc2,
              input pt_net_wt,
              input pt_net_wt_um,
              input pt_price,
              input pt_promo,
              input pt_site,
              input pt_taxc,
              input pt_um,
              input v_std_cost,
              input v_std_cost_set,
              input pt_pm_code,
              output error_flag,
              output err_mess_no)"}

if error_flag
then do:
  /* ERROR RETURNED BY IFAPM054.P */
  {pxmsg.i &MSGNUM = err_mess_no
   &ERRORLEVEL = {&APP-ERROR-RESULT}
  }
  undo, return.
end. /* IF error_flag */

end. /* IF soc_apm and (promo_old <> "" and ... ) */
/*@MODULE APM END*/

if new_part
then do:
  undo_all = yes.
/*FC01** {gprun.i """ppptmta1.p"""} **/
```

Annexe VIII

```
/*FC01*/ {gprun.i ""s1ptmta1.p""}

if not regen_add
  and (l_pt_break_cat <> pt_break_cat
    or l_pt_group <> pt_group
    or l_pt_part_type <> pt_part_type
    or l_pt_prod_line <> pt_prod_line)
then
  assign part_node = pt_part
  l_ana_build = yes.

if keyfunction(lastkey) = "end-error"
  or undo_all
then undo mainloop, retry mainloop.
  regen_add = yes. /*we have a record so we can */
  part_node = pt_part. /*add this part (node) to anx_det*/
end. /* IF new_part */

if not regen_add
then
  assign
    l_pt_article = pt_article
    l_pt_break_cat = pt_break_cat
    l_pt_buyer = pt_buyer
    l_pt_desc1 = pt_desc1
    l_pt_desc2 = pt_desc2
    l_pt_group = pt_group
    l_pt_part = pt_part
    l_pt_part_type = pt_part_type
    l_pt_prod_line = pt_prod_line
    l_pt_site = pt_site.

end. /* DO ON ERROR */

end. /* DO TRANSACTION WITH frame a */

if (ppform = "" or ppform = "a") and not new_part
then do
  on endkey undo,
/*4472** next mainloop: **/
/*4472*/ leave loopa:

  undo_all = yes.
/*FC01** {gprun.i ""ppptmta1.p""} */
/*FC01*/ {gprun.i ""s1ptmta1.p""}

if not regen_add
  and (l_pt_break_cat <> pt_break_cat
    or l_pt_group <> pt_group
    or l_pt_part_type <> pt_part_type
    or l_pt_prod_line <> pt_prod_line)
then
  assign part_node = pt_part
  l_ana_build = yes.

  if keyfunction (lastkey) = "end-error"
    or undo_all
  then
/*4472**      next mainloop. **/
```

Annexe VIII

```
/*4472*/ leave loopa.

end. /* IF (ppform = "" or ppform = "a") AND ... */

if (ppform = "" or ppform = "b")
then do
  on endkey undo,
/*4472**    next mainloop:    */
/*4472*/  leave loopa:

{gprun.i """ppptmtb.p"""}  
  
if not regen_add
  and (l_pt_article <> pt_article
  or l_pt_site <> pt_site)
then
  assign part_node = pt_part
  l_ana_build = yes.  
  
if keyfunction (lastkey) = "end-error"
then do:
  if qxo_write then
    run fireOutboundEvent (input rowid({&QXO-TABLE}),
                           input oid_{&QXO-TABLE},
                           input 'WRITE').
/*4472**      next mainloop.    */
/*4472*/  leave loopa.
  end.  
  
end. /* IF ppform = ...*/
/*END LOOP B */  
  
/*4472** /*GA01*/ if ppform = "a" then do on error undo, next mainloop  */
/*4472** /*GA01*/      on endkey undo, next mainloop transaction: **/  
  
/*4472*/ if ppform = "a" then do on error undo, leave loopa
/*4472*/      on endkey undo, leave loopa transaction:
/*GA01*/  set pt_ord_qty v-lib with frame lib.
/*GA01*/  if v-lib then do:
/*GA01*/    {gprun.i """s1gpcmmmt.p"""" "(input pt_part)"}
/*GA01*/    view frame cmt.
/*GA01*/  end.
/*GA01*/ end. /* if ppform = "a" */  
  
/*A4L1*/ /* Begin ADD - Saisie des critères */
pause 0 before-hide.
hide frame lib no-pause.
hide frame v-cmmt no-pause.
if ppform = "a" then do transaction:
  do i = 1 to 9:
    criterie[i + 1] = (substring(pt_chr02,i,1) = "1").
  end.
  display pt_chr01 @ famille
    criterie[2]
    criterie[3]
    criterie[4]
    criterie[5] with frame criteries.  
  
do on error undo, leave loopa
```

Annexe VIII

on endkey undo, leave loopa:

```
old_part = global_part.  
global_part = pt_pm_code.  
  
set famille criteres[2] criteres[3] criteres[4] criteres[5]  
with frame criteres editing:  
    if frame-field = "famille" then do:  
        {mfnp05.i usrw_wkfl usrw_index1  
            "usrw_domain = global_domain and  
            usrw_key1 = 'S1-COEFFICIENT-PV' and  
            entry(2,usrw_key2,'|') = pt_pm_code"  
            entry(1,usrw_key2,'|') "input famille"}  
        if recono <> ? then  
            display entry(1,usrw_key2,"|") @ famille  
            with frame criteres.  
        end.  
    else do:  
        readkey.  
        apply lastkey.  
    end.  
end.  
  
global_part = old_part.  
  
for first usrw_wkfl no-lock  
where usrw_domain = global_domain  
and usrw_key1 = "S1-COEFFICIENT-PV"  
and usrw_key2 = famille + "|" + pt_pm_code: end.  
if not available usrw_wkfl then do:  
    /* Famille inexiste */  
    {pxmsg.i &MSGNUM=9822 &ERRORLEVEL=3 &MSGARG1=pt_pm_code}  
    next-prompt famille with frame criteres.  
    undo, retry.  
end.  
  
end. /* do on error... */  
  
pt_chr01 = famille.  
  
do i = 1 to 9:  
    substring(pt_chr02,i,1) = (if criteres[i + 1] = yes then "1"  
                                else "0").  
end.  
  
end. /* if ppform = "a" */  
/*A4L1*/ /* End ADD */  
  
hide frame a1 no-pause.  
hide frame b no-pause.  
hide frame b1 no-pause.  
if ppform = "" or ppform = "c"  
then do  
    on endkey undo,  
/*4472**      next mainloop:    **/  
/*4472*/      leave loopa:  
  
/*FC01**      {gprun.i ""ppptmtc.p""} **/  
/*FC01*/      {gprun.i ""s1ptmtc.p""}  
if not regen_add
```

Annexe VIII

```
and l_pt_buyer  <>> pt_buyer
then
assign part_node = pt_part
l_ana_build = yes.

{mgqqapp.i "pt_app_owner"}

if keyfunction(lastkey) = "end-error"
then do:
    if qxo_write then
        run fireOutboundEvent (input rowid({&QXO-TABLE}),
                               input oid_{&QXO-TABLE},
                               input 'WRITE').
/*4472**      next mainloop.    */
/*4472*/      leave loopa.
end.

end. /* REPEAT: */
/* END LOOP C */

hide frame c no-pause.

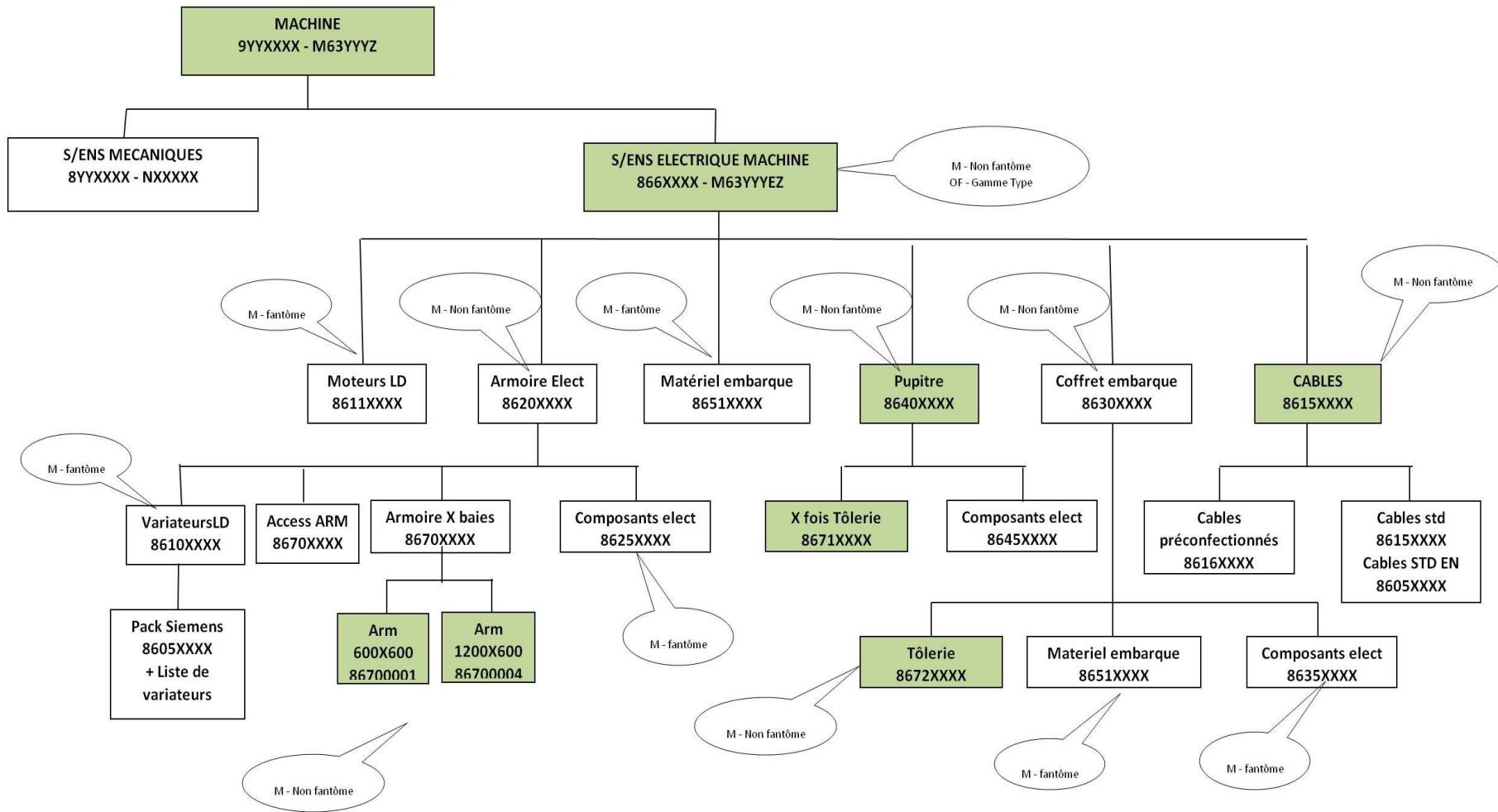
/* COSTING */
if ppform = "" or ppform = "d"
then do
    on endkey undo,
/*4472**      next mainloop:    */
/*4472*/      leave loopa:

    {gprun.i """ppptmtd.p"""}
    if keyfunction(lastkey) = "end-error"
    then do:
        if qxo_write then
            run fireOutboundEvent (input rowid({&QXO-TABLE}),
                                   input oid_{&QXO-TABLE},
                                   input 'WRITE').
/*4472**      next mainloop.  */
/*4472*/      leave loopa.
    end.
end. /* IF ppform = "" OR ppform = "d" */

if ppform = " "
then do:
    /* SEE IF USER WOULD LIKE TO SEE SERVICE DATA HERE */
    for first svc_ctrl
        fields( svc_domain svc_itm_data)
        where svc_domain = global_domain no-lock:
    end. /* FOR FIRST svc_ctrl */

if available svc_ctrl and svc_itm_data
then do
    on endkey undo,
/*4472**      next mainloop:  */
/*4472*/      leave loopa:
    hide frame d0.
    hide frame d1.
    {fsptfrm.i "new"}
    {gprun.i """fsptmt1.p"" "(input pt_part)"}
    end. /* IF AVAILABLE svc_ctrl AND ... */
end. /* IF ppform = " " */
```

Annexe VIII



Annexe IX : Arborescence d'une nomenclature de machine

	Ordre de Fabrication	Client		Page 1 / 4
	Composé 86000012	PUP. UNIVERSAL-TOL. ET CAB. PLATINE (IND B ET		Quantité 1.00
Chargé d'Affaire Stephane VASSEUR	Schéma Electrique ELC23456		Création 06/12/10	Impression 06/12/10
Ligne	Repère Topologique	Référence	Quantité	Référence Fabricant
01		86710055 ENVELOPPE PUP.UNIVERSALTC 20/10	1.00	
001		86710100 TOLERIE PUP. UNIVERSAL -TC 20/ 10	1.00	
002		26290053 CHARNIÈRE INVISIBLE 120° + GOJONS SERTIS REF: 85-7-1171	4.00	8571171 PINET
003		2110343 CIRCLIPS EXTERIEURDIAMETRE 6	0.04	
004		26290151 VERROU CHROMECORPS + ECROU)	2.00	8171001 PINET
005		26290128 ROTOR DOUBLE BARRE 3CHROME+ VIS + RONDELLES	2.00	8172107 PINET
006		26290028 CAME H=26MM	2.00	CAME26 SAME
0061		26290057 CLE DOUBLE PANNETON 3MM	1.00	100405 EMKA
007		26290081 JOINT A CLIPSER H=10 EPAIS 1 A 3.5 REF:1011-05	4.70	ARMH10X35 SAME
008		26290015 BORDURE DE PROTECTION EPAIS. 1 A 4.5 REF. EMK 1010-01	2.00	BORP4.5 SAME
009		26350386 PASSE FILS 60 X 25 NOIR REF:MPFO 60 N (PAR 100 PIÈCES)	2.00	FIL60X25 LEMO
0091		26350012 BOUCHON Ø 7,2 X 15 X 6.5NOIR	0.04	054070569903 SKIFFY
010		2100903 VIS H M10-40CLASSE 8.8-DIN 931	0.04	
011		2100887 RONDELLE PLATE M10 TRAITEMENT ARD SERIE MOYENNE NFE 27611	0.08	
012		2100916	0.04	

 Chargé d'Affaire Stephane VASSEUR	Ordre de Fabrication	Client			Page 2 / 4
	Composé 86000012	Schéma Electrique ELC23456	PUP. UNIVERSAL-TOL. ET CAB. PLATINE (IND B ET)		Quantité 1.00
Ligne	Repère Topologique	Référence	Quantité	Référence Fabricant	Fabricant
		ECROU H M10 DIN 934			
050		4620279 PALETTE 1200x800 IPSREF. NEFAB EXPACK TYPE P	0.00		
02		86450039 CABLAGE PUP. UNIVERSAL(VERSION ETHERNET)	1.00		
065	X3C - X3H	26130171 PRISE DE COURANT 2P+T 10/16A	2.00	15306	SCHNEIDER
066	Q3A	26190118 DISJONCTEUR C60N BI 2A COURBE C UL	1.00	24443	SCHNEIDER
067	Q3A	26190018 BLOC VIGI C60 BIPOLAIRE30MA 25A	1.00	26509	SCHNEIDER
070	M3A - VENTIL.	26300224 VENTILATEUR 230V 50/60HZ148X148-NOIR	1.00	8300171	RITTAL
071	M3A - VENTIL.	26300130 GRILLE DE SORTIE 148X148- NOIR	1.00	8300172	RITTAL
072	M3A - VENTIL.	2110344 GOUPILLE BETA SERIEN°805 DIAMETRE FIL3	0.01		L'ETOILE
073	M3A - VENTIL.	26350033 CABLE HO5VVF 3G0.75 AVECVERT/JAUNE	3.50	C3G0.75	SAME
105		26350446 TRESSE DE MASSE 16° LONG200 Ø8	3.00	TRESSE16X200-8	SAME
108	S1B - ARRET D'URGENCE	26430079 EMBASE + BLOC CONTACT O+O	1.00	ZB4BZ104	SCHNEIDER
109	S1B	26430152 TETE COUP DE POING ROUGE O40 - POUSSER/TOURNER A VERROUILLAGE BRUSQUE	1.00	ZB4BS844	SCHNEIDER
110	S1B	26430056 CONTACT O	1.00	ZBE102	SCHNEIDER
111	S1B	26430095 ETIQUETTE ARRET D'URGENCE Ø90 REF: ZBY8101	1.00	ZBY8101	SCHNEIDER

 Chargé d'Affaire Stephane VASSEUR	Orde de Fabrication	Client			Page 3 / 4
	Composé 86000012	Schéma Electrique ELC23456	PUP. UNIVERSAL-TOL. ET CAB. PLATINE (IND B ET)		Quantité 1.00
				Création 06/12/10	Impression 06/12/10
Ligne	Repère Topologique	Référence	Quantité	Référence Fabricant	Fabricant
112	S1B	26310083 CABLE CNOMO NO5VV5F 7G1AVEC VERT/JAUNE	3.00	C7G1	SAME
131	SUPPORT JOG	26350439 SUPPORT BOITIER PILOT 20(LIVRE PAR 2)	0.50	29600100	PHOENIX MECANO
135	XZO - PRISE JOG	26340182 PRISE FEMELLE 10P+T	1.00	09200102812	HARTING
136	XZO - PRISE JOG	26340123 EMBASE ENCASTREE + COUVERCLE TYPE: HAN 10A	1.00	09200100321	HARTING
137	XZO - PRISE JOG	26310057 CABLE CNOMO NO5VV5F 12G1AVEC VERT/JAUNE	1.00	C12G1	SAME
138	BORNIER	26370080 BORNES VERT/JAUNE 6 ² -REF: 282907	1.00	282907	WAGO
139	BORNIER	26370108 PLAQUE D' EXTREMITE GRISE 6 ² - REF: 282325	1.00	282325	WAGO
140	BORNIER	26370078 BORNES VERT/JAUNE 2,5 ² - REF: 280907	5.00	280907	WAGO
141	BORNIER	26370070 BORNES DE PASSAGE GRISE2,5 ² - REF: 280901	20.00	280901	WAGO
143	BORNIER	26340104 CONTACT DE PONTAGE HORIZONTAL - REF: 280402	4.00	280402	WAGO
144	BORNIER	26360065 REPÈRE VIEGE 5MM WMB - REF: 793-501 (5 CARTES DE 10 X 10 ETIQUETTES)	0.05	793501	WAGO
145	BORNIER	26370083 BUTEE D'ARRET - REF: 249116	2.00	249116	WAGO
149	MASSE PC	26350059 CABLE HO7VK2.5 VERT/JAUNE 700V	4.00	HO7VK2.5	SAME
150	W3A - ALIM. PC	26340136 FICHE DROITE PX 0686 M 32 P+T; A VISSE	1.00	551748	ORBITEC
151	W3A - ALIM. PC	26340133	1.00	551749	ORBITEC

	Ordre de Fabrication	Client			Page 4 / 4
	Composé 86000012	PUP. UNIVERSAL-TOL. ET CAB. PLATINE (IND B ET			Quantité 1.00
Chargé d'Affaire Stephane VASSEUR		Schéma Electrique ELC23456		Création 06/12/10	Impression 06/12/10
Ligne	Repère Topologique	Référence	Quantité	Référence Fabricant	Fabricant
		FICHE COUDEE FEMELLE 2 P+T CEE 22 HO			
152	W3A - ALIM. PC	26350034	3.00	F3X1NO	SAME
		CABLE HO5VVF 3X1 ² NOIR HOMOLOGUE VDE (GAINÉ PVC)			
155	U3A + W3B - ONDULEUR	26250192	0.00	APCBR500I	SAME
		ONDULEUR 500VA 300W APC BR500I AVEC 2 CORDONS AL IM. CEE22			
156	W3E - ONDULEUR	26250161	0.00	SERIEONDULEUR	IP SYSTEMES
		CABLE SERIE POURONDULEUR BR500I L=2,10MRJ10 - DB 9F (CABLE 80°C/300V/26AWG)			
157	W3D - ONDULEUR	26250058	1.00	808010	ABIX
		CORDON MALE 2+T COUDE / FEMELLE CEE22 / 1,80M REF: 808010			