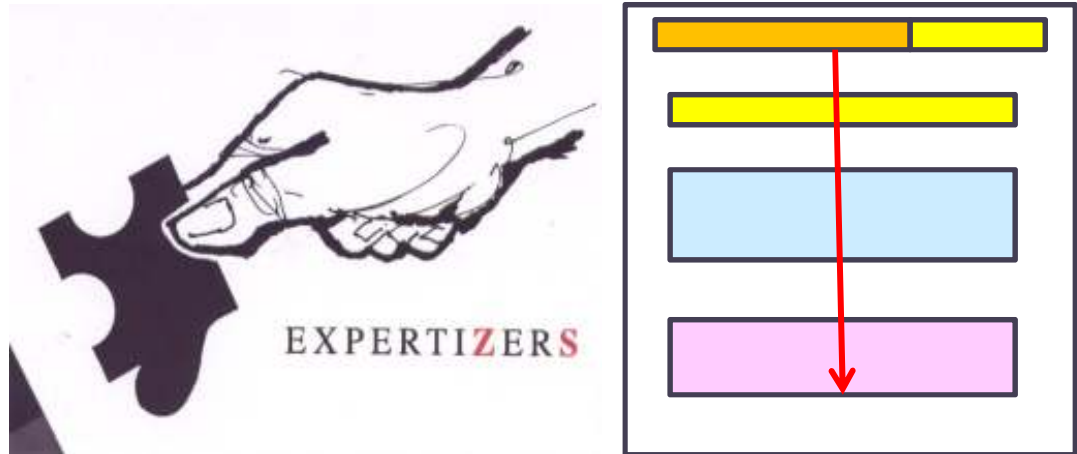


SYNTHESE DES REPARTITIONS D'EXPERTIZERS



Synthèse des deux modes de répartition qui participent à la puissance du logiciel

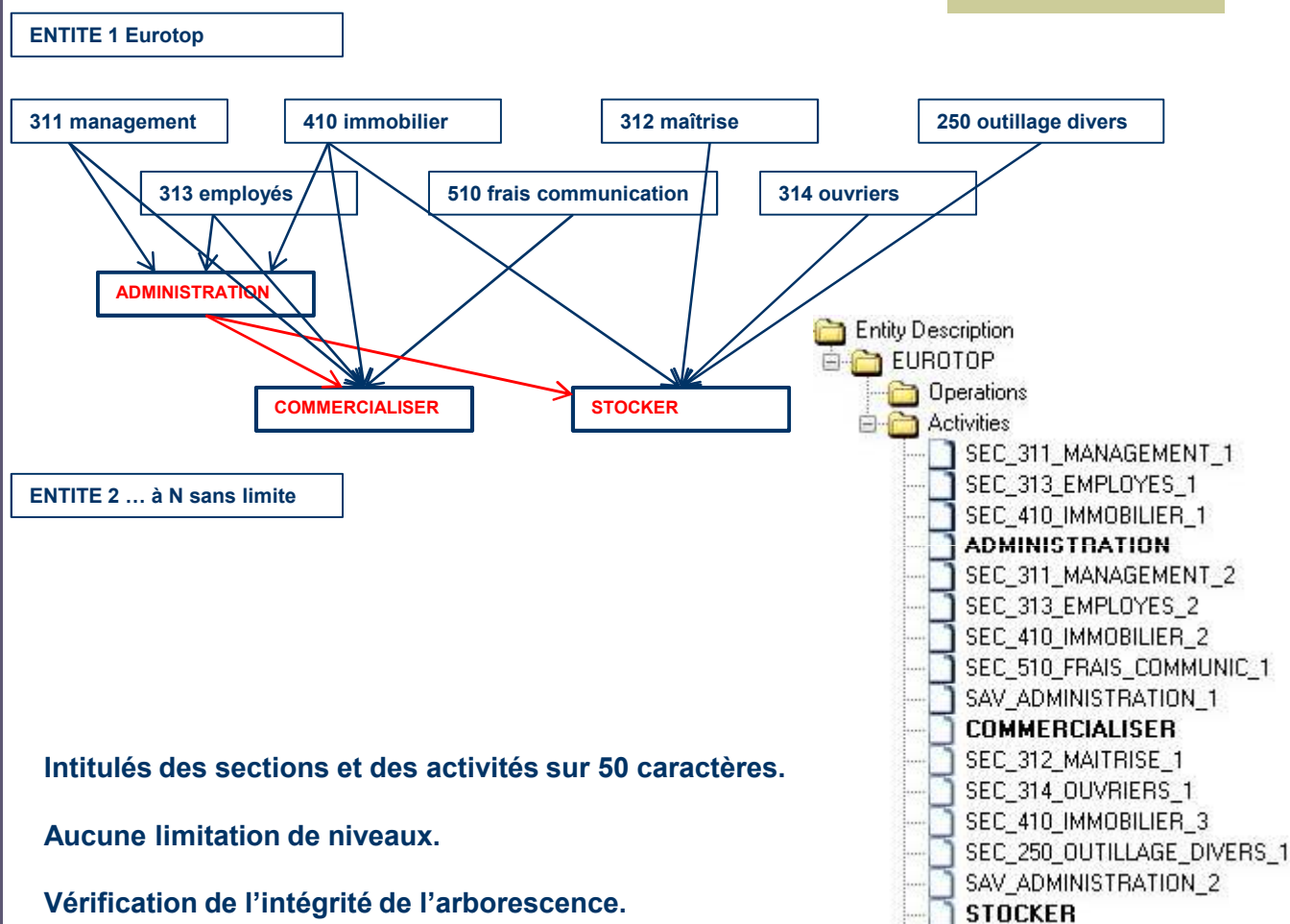
- 1. Répartition du coût de sections comptables sur des activités et des activités sur d'autres activités**
- 2. Répartition d'un coût d'activité sur des lignes d'évènements au prorata de leur consommation d'unités d'œuvre de l'activité**

Le point 1 est illustré en détail dans le document automatic_generation.pdf et le modèle eurotop_1

Le point 2 est illustré en détail dans le modèle eurotop_2, mais ici deux exercices vous aident à intégrer définitivement le fonctionnement interne de ce mode de répartition

Expertizers Copyright 2009-2010

1. REPARTITION DU COÛT DE SECTIONS COMPTABLES SUR DES ACTIVITES & REPARTITION DU COÛT D'UNE ACTIVITE SUR D'AUTRES ACTIVITES



Intitulés des sections et des activités sur 50 caractères.

Aucune limitation de niveaux.

Vérification de l'intégrité de l'arborescence.

Autant d'arborescences que d'entités incluses dans un même modèle.

Arborescences et coefficients de répartition sont saisis manuellement ou **générés automatiquement par l'import de données externes.**

Les liens entre les nœuds sont gérés automatiquement sous forme de pseudo activités de type SEC_section et SAV_activité en incluant leur coefficient de répartition.

Traçabilité et analyse assurées, sur l'origine des coûts d'un nœud ou de tout niveau de l'arborescence.

Autres documents de référence pour présentation et formation:

[Resolution_eurotop_1.pdf](#)

[Automatic_generation_1.pdf](#) (Génération automatique d'une modèle de répartition)

2. REPARTITION DES COÛTS D'ACTIVITES SUR DES LIGNES D'EVENEMENTS AU PRORATA DE LEUR CONSOMMATION D'UNITES D'ŒUVRE

Ce mode de répartition est traité dans le modèle Eurotop_2. Le modèle s'appuie sur un fichier d'évènements composé de factures. Les factures traversent les règles du modèle et aboutissent au **calcul des unités d'œuvres consommées** par chaque ligne de facture dans chaque activité.

L'objet du cas Eurotop_2 est aussi de **charger un fichier externe du coût des activités** et de les répartir sur chaque ligne de facture / activité au prorata de leur consommation d'unités d'œuvre.

Comme ce processus s'exécute très rapidement, il est important de présenter les mécanismes internes de son fonctionnement. Deux exercices de simulation vous sont proposés à cet effet dans les pages suivantes.

Nous attirons votre attention sur le fait que l'algorithme de répartition est d'utilisation très générique, comme vous pourrez le constater en consultant aussi le document [repartition_simple_de_couts.pdf](#)

Autre documents de référence pour présentation et formation:

Resolution_eurotop_2.pdf

Repartition_simple_de_couts.pdf (rapprochement de deux fichiers externes sur un autre critère que l'activité; par exemple la répartition du coût global d'un produit sur des factures au prorata des ses quantités vendues)

Cout_comme_uo.pdf (particularité d'un coût d'activité réparti qui devient une unité d'œuvre pour la répartition du coût d'une autre activité.

EXERCICE 1 DE SIMULATION

Comment répartir le coût de l'activité ORDER sur les événements au prorata du chiffre d'affaires

Variables descriptives du fichier d'évènement:

Variables d'initialisation en colonnes, traitées de gauche à droite comme dans un tableau:

Entity	period	Product_cod	Event_code	Quantity_init	Country	turnover		
PARIS		PROD_1	1	100	GB	1000		
PARIS		PROD_2	2	200	FR	2000		
PARIS		PROD_3	3	300	GB	3000		
PARIS		PROD_4	4	400	FR	4000		

Accès: Lot_batch / Charger + traiter les événements

Variables d'initialisation en lignes (User Internal variables section):

Variable	Rule

Description des processus et leurs règles:

Entity	operation	Obj	Nb obj.	Line rule	Condition	rule
PARIS						

Cet exercice ne comporte pas de description de processus

Définition des activités et leurs règles:

Entity	Activity	Memo rize	Display in results	Condition	Activity_value rule	Activity_cost rule	Activity_text
PARIS	ORDER	No	Yes		Activity_value = turnover	(cf NB2)	

Matrice (résultats / lot par activité) après traitement itératif (boucle de lecture des lignes d'évènements)

Entity	Period	Product_code	Event codé	Activity	Activity_value	Activity_cost	Activity_text
PARIS		PROD_1	1	ORDER	1000	5000	
PARIS		PROD_2	2	ORDER	2000	10000	
PARIS		PROD_3	3	ORDER	3000	15000	
PARIS		PROD_4	4	ORDER	4000	20000	

Chargement du fichier des coûts. Accès Lot_batch / Charger un fichier des coûts

Entity	Period	Activity	Activity_cost
PARIS	(cf NB1)	ORDER	50000

Matrice Σ des activity_value par entity / period / activity
Accès visé: résultats / totaux par période activité

Entity	Period	Activity	Activity_value	Activity_cost
PARIS		ORDER	10000	

répartition du coût sur les lignes d'évènement:

5 * activity_value de chaque ligne

Accès visé = Résultats / lot par activité

Matrice en visé en fin de chargement

Entity	Period	Activity	Activity_value (1)	Activity_cost (2)	Unit_cost (2/1)
PARIS		ORDER	10000	50000	(50000 / 10000) = 5

NB1: la colonne PERIOD est facultative. Si les coûts sont donnés sans période ils sont répartis sur l'intégralité du fichier des événements, période présente ou non.

NB2: Les coûts calculés par une règle sur une activité sont conservés quand l'activité est absente du fichier des coûts.



EXERCICE 2 DE SIMULATION

Comment répartir le coût de l'activité ORDER sur les événements à raison de 60% du coût sur le pays GB et 40% du coût sur le pays FR

Variables descriptives du fichier d'évènement:

Variables d'initialisation en colonnes, traitées de gauche à droite comme dans un tableau:

Entity	period	Product_cod	Event_code	Quantity_init	Country	turnover	Sum_turnover_by_cty (rule)	
PARIS		PROD_1	1	100	GB	1000	4000	
PARIS		PROD_2	2	200	FR	2000	6000	
PARIS		PROD_3	3	300	GB	3000	4000	
PARIS		PROD_4	4	400	FR	4000	6000	

Accès: Lot_batch / Charger + traiter les évènements

Cf note sur la variable sum page suivante_

Variables d'initialisation en lignes (User Internal variables section) :

Variable	Rule
List_1	GB=60/FR=40
Country_ratio	Variable qui récupère de list_1 la valeur correspondante au pays

Cf note sur la variable decode page suivante

Description des processus et leurs règles:

Entity	operation	Object	Nb obj.	Line rule	Condition	rule
PARIS						

Cet exercice ne comporte pas de description de processus

Définition des activités et leurs règles :

Entity	Activity	Memo rize	Display in results	Condition	Activity_value rule	Activity_cost rule	Activity_text
PARIS	ORDER	No	Yes	Sum_turnover_by_cty <> 0	Activity_value = country_ratio * (turnover / sum_turnover_by_cty)		

Matrice (résultats / lot par activité) après traitement itératif (boucle de lecture des lignes d'évènements)

Entity	Period	Product_code	Event codé	Activity	Activity_value	Activity_cost	Activity_text
PARIS		PROD_1	1	ORDER	$60 * (1000 / 4000) = 15$	7500	
PARIS		PROD_2	2	ORDER	$40 * (2000 / 6000) = 13.33$	6665	
PARIS		PROD_3	3	ORDER	$60 * (3000 / 4000) = 45$	22500	
PARIS		PROD_4	4	ORDER	$40 * (4000 / 6000) = 26.67$	13335	

Matrice Σ des activity_value par entity / period / activity
Accès visu: résultats / totaux par période activité

Entity	Period	Activity	Activity_value	Activity_cost
PARIS		ORDER	100	

Chargement du fichier des coûts. Accès lot_batch / Charger un fichier des coûts

Entity	Period	Activity	Activity_cost
PARIS		ORDER	50000

répartition du coût sur les lignes d'évènement:

$500 * \text{activity_value}$ de chaque ligne

Accès visu = Résultats / lot par activité

Matrice en visu en fin de chargement

Entity	Period	Activity	Activity_value (1)	Activity_cost (2)	Unit_cost (2/1)
PARIS		ORDER	100	50000	$(50000 / 100) = 500$



NOTES EXPLICATIVES SUR LES VARIABLES DE TYPE SUM_ & DECODE

Variables de type SUM_ : Ce type de variable ouvre une colonne dans la description du fichier d'évènements. Elle contient une règle de sommation générée automatiquement à sa création. Au démarrage du traitement du fichier des évènements, la règle est tout de suite exécutée dont les résultats sont reportés sur chaque ligne du fichier des évènements.

Chemin de création: Utilitaires / Créer une variable sum_

Variables de type DECODE : Ce type de variable peut être ouverte soit en colonne (comme la variable sum_) soit en ligne (User Internal variable section). Elle sert à récupérer sous son nom une valeur déduite de la valeur d'une autre variable. La création d'une variable decode génère en fait deux variables: une appelée list_n qui contient les valeurs associées, l'autre qui récupère (par règle automatique) la valeur déduite.

Chemin de création: Utilitaires / Créer une variable decode

Chemin de mise à jour de la liste: Click sur la variable list_n + *click droit souris 'action modifier'*

Sequence	country values	to decode in numeric value
1 FR	40	40
2 GB	60	60
3?	0	0
4		

= User Internal variable section

Avertissement

Ce document vous est présenté par DB.SOFTART.SERVICES. (« DB »).

Ce document et son contenu sont le produit de recherches et de développements importants conduits par DB.

Ils vous sont présentés dans le seul but de vous permettre d'évaluer une partie des capacités du logiciel Expertizers.

Vous devez donc traiter le contenu de cette présentation, ainsi que toute autre information ou document en relation avec les solutions présentées au cours de cette conférence (les « Informations ») comme étant confidentiels et la propriété exclusive de DB.

En recevant ainsi les Informations, vous vous engagez à n'utiliser les informations que dans le seul but d'évaluer pour votre propre compte les capacités du logiciel Expertizers, à maintenir leur confidentialité et à ne transmettre à quiconque, totalité ou partie des Informations, sans autorisation écrite et préalable de DB.

- DB.SOFTART.SERVICES
Didier RICHE
Conseiller et chargé d'affaires
+33 (0)6.63.92.20.60