

Analyse, conception et implantation d'un
logiciel de CRM pour la société AdaCore

Audran LE BARON

Tuteur scientifique : Laurent PAUTET

Directeur de stage : Franco GASPERONI

15 février 2004

Table des matières

1	Présentation d'AdaCore et du projet de CRM	3
1.1	AdaCore	3
1.2	L'Offre commerciale d'AdaCore	4
1.3	Le Projet de CRM	5
2	Analyse du projet de CRM	6
2.1	Les Techniques de modélisation	6
2.2	Le Travail d'enquête	7
2.3	Intérêts de la modélisation	7
2.4	Résultats de la modélisation	8
3	Design : construction d'une maquette	9
3.1	L'Interface utilisateur	9
3.2	Le Choix d'une interface Web	9
3.3	Réalisation d'une maquette	10
3.4	Conception du schéma de la base de données	10
4	Implantation	11
4.1	Base de données PostgreSQL	11
4.2	L'Utilisation de la technologie AWS	11
5	Passation du projet	12
A	Sales Workflow Diagrams	14
B	ER Diagram	26

Le présent rapport fait état des travaux menés durant mon stage — du 15 juillet 2003 au 30 janvier 2004 — dans la société ACT Europe. L'objet de mon stage était l'analyse, la conception et l'implantation d'un outil interne de gestion des clients pour les sociétés ACT Europe (Paris) et Ada Core Technologies (New-York).

Ce projet, mené depuis ses débuts et jusqu'à son implantation, a été l'occasion de mettre notamment en pratique les enseignements de la brique BDL (Bases de Développement Logiciel) dans un environnement réel. La réalisation du projet en Ada m'a permis par ailleurs de consolider et d'approfondir les bases de ce langage acquises à l'ENST dans le cadre de la brique SIP (Système, Interface, Programmation).

Après une courte présentation de l'entreprise, nous reprendrons chacune des grandes étapes constitutives du projet que j'ai été amené à réaliser.

1 Présentation d'AdaCore et du projet de CRM

1.1 AdaCore

Le nom *AdaCore* désigne deux sociétés « jumelles » : l'une basée aux États-Unis à New-York (*Ada Core Technologies, Inc.*) et l'autre située en France à Paris (*ACT Europe*).

Les offres de ces deux sociétés sont strictement identiques : leur produit-phare est *GNAT Pro*, un environnement de développement Ada¹ sous licence GPL², qu'elles vendent dans le cadre de contrats de souscriptions à un service de support.

Les deux sociétés développent ensemble les mêmes produits et se répartissent les clients de manière géographique : le continent américain, l'Australie et le Japon sont parmi les régions réservées à la société de New-York tandis que l'Europe et l'Asie (excepté le Japon) constituent le territoire de la société

¹Langage de programmation spécialement conçu pour le développement de logiciels à longue durée de vie, aisément maintenables et hautement fiables.

²*General Public License* : Licence de logiciel libre, détails disponibles à l'adresse : <http://www.gnu.org/copyleft/gpl.html>.

parisienne.

Dans la suite de ce rapport, le terme *AdaCore* fera donc référence indifféremment à la fois à *ACT Europe* et à *Ada Core Technologies*.

1.2 L'Offre commerciale d'AdaCore

Tous les produits développés par AdaCore sont *open source*, sous licence GPL ou approchant (certains composants sont notamment sous GMGPL³ ou LGPL⁴). Parmi ces produits se trouve le produit-phare, qui a donné naissance à la société : le compilateur *GNAT*, issu d'un projet universitaire à NYU et basé sur *GCC*⁵.

GNAT se situe au centre de l'offre commerciale d'AdaCore. Un grand nombre d'outils gravitent autour : parmi eux, l'IDE⁶ *GPS*⁷ et le *debugger* graphique *GVD*⁸, les bibliothèques *GNAT*, etc. Tous ces composants sont disponibles en accès libre et public⁹. Cette version libre est largement utilisée dans les universités ainsi que par les particuliers.

Une version payante, *GNAT Pro*, est proposée aux professionnels. Le produit *GNAT Pro* est vendu avec une souscription d'un an ou plus au service de support d'AdaCore. Ce service prévoit notamment la réponse aux questions des équipes de développement dans des temps très courts (souvent inférieurs à 15 minutes pour les requêtes simples) et directement par des professionnels spécialistes du langage Ada, ou encore la mise à disposition de *wavefronts*¹⁰ en cas de bug, etc. De plus, et contrairement à la version libre, le client est garanti d'avoir de nouvelles versions — qualifiées qui plus est — de la gamme de produit tous les six mois (chaque version corrigeant tous les bugs ayant été soumis par les différents clients depuis la version précédente).

Ce *business model* a déjà fait ses preuves. L'une des grandes raisons de ce succès est la qualité du service rendu aux clients, qui fait notamment que plus de 80 % des contrats sont renouvelés d'une année sur l'autre. Une autre

³GMGPL : *GNAT Modified GPL*.

⁴LGPL : *Lesser-GPL*.

⁵*GNU Compiler Collection* (<http://gcc.gnu.org/>).

⁶IDE : *Integrated Development Environment*.

⁷GPS : *GNAT Programming System*.

⁸GVD : *GNU Visual Debugger*.

⁹Voir le site *libre* d'AdaCore : <http://libre.act-europe.fr/>.

¹⁰Versions intermédiaires de la chaîne d'outil *GNAT* avant une prochaine *release*.

est également la présence dans la clientèle de grands groupes pérennes tels que BNP Paribas, Boeing, Canal Plus, Dassault, EADS, EdF, Eurocontrol ou Thales. Cette position confortable n'empêche pas pour autant les dirigeants d'AdaCore de sans cesse remettre en cause leur offre, leur système de tarification, leur image, afin de suivre au plus près l'évolution de leur marché et les besoins de leurs clients.

1.3 Le Projet de CRM

AdaCore est une entreprise récente (fondée en 1994). Elle est en croissance permanente, tant du point de vue du nombre de ces employés (une quarantaine environ, dont une quinzaine à Paris) que du point de vue du nombre de ses clients (quelques centaines à ce jour). La société a ainsi atteint une taille critique au-delà de laquelle il serait difficile de gérer l'ensemble des clients sans l'aide d'un logiciel de gestion de clients. Jusqu'à ce jour, la gestion administrative des clients (envois de devis, de factures, ouverture et fermeture de comptes, gestion des contacts, envois des produits, etc.) était une gestion purement manuelle, sous la responsabilité de Zépur Blot en Europe et de Karen Mason aux États-Unis.

Le besoin d'un outil informatique d'aide à la gestion des clients s'est fait sentir dès l'été 2001. Ce type d'outil est connu sous l'acronyme CRM (*Customer Relationship Management*). Il existe déjà sur le marché plusieurs solutions de CRM. Elles se présentent généralement sous la forme d'un grand nombre de composants logiciels nécessitant une longue période de configuration, d'intégration et de mise en place par des professionnels. Ces solutions sont de plus adaptées à des grandes entreprises dont l'échelle a peu de rapport avec l'échelle d'AdaCore. Une autre raison pour ne pas avoir choisi de solution commerciale existante est qu'aucune ne prévoyait une intégration réelle et complète de la communication électronique par email. De plus, le parc informatique d'AdaCore étant relativement hétérogène (majoritairement des PC sous GNU/Linux, certains sous Windows ainsi que quelques *PowerPC* sous Mac OS X), la condition de compatibilité avec l'ensemble de ces systèmes était primordiale et suffisait à éliminer la grande majorité des solutions, dont les clients ne fonctionnaient que sous Windows. Enfin, le système de fonctionnement d'AdaCore est spécifique en de nombreux points et n'aurait vraisemblablement pas permis l'intégration parfaite et sans compromis d'un outil de CRM existant.

C'est dans cette situation que je suis arrivé en juillet 2003 dans les

bureaux d'AdaCore à Paris : la décision avait été prise de développer en interne un logiciel de CRM. Franco Gasperoni est la personne à l'origine de ce projet. Une étude de l'offre existante — brièvement évoquée plus haut — avait déjà été menée et un premier cahier des charges rédigé. Mon but était donc de réaliser, en collaboration étroite avec Franco, l'analyse poussée du projet et la conception d'un logiciel de CRM adapté à AdaCore, puis d'en réaliser une maquette et enfin d'en commencer l'implantation.

2 Analyse du projet de CRM

Pour bien comprendre les objectifs du projet et les besoins auxquels devra répondre l'outil de CRM, il est indispensable de comprendre dans les détails ce en quoi consiste le travail du personnel lié à la vente, à savoir principalement Zépur Blot et Franco Gasperoni en Europe d'une part, Karen Mason et Bernard Banner aux États-Unis d'autre part.

L'idée est donc d'avoir une vue à la fois globale et détaillée de l'activité liée à la vente chez AdaCore. Il s'agit ensuite de mettre en lumière l'ensemble des tâches à caractère *administratif* (par opposition aux activités *créatives*) liées à la vente. C'est en effet sur cet aspect administratif du travail de la relation client qu'un outil informatique peut apporter une réponse convaincante.

2.1 Les Techniques de modélisation

Il existe des principes de modélisation des processus, notamment en UML¹¹, qui peuvent aider à obtenir cette vue globale et détaillée de certaines activités. Un travail préalable à mon stage a donc été, sous les conseils de Franco, d'approfondir mes connaissances sur ce sujet. Deux livres ont été particulièrement utiles :

- *UML Distilled : A Brief Guide to the Standard Object Modeling Language*¹² est une très bonne introduction à l'UML. Les auteurs insistent sur les apports de la modélisation dans les projets de développement logiciel, mais également mettent en garde le lecteur sur les

¹¹UML : *Unified Modeling Language*.

¹²Martin Fowler et Kendall Scott, 2nd edition, Octobre 1999, ISBN 0-201-65783-X.

excès de modélisation ou de conformation à un modèle (en l'occurrence, le langage UML), qui peuvent à leur tour devenir un frein au développement d'un projet. Plus qu'une simple introduction à l'UML, ce livre est aussi une introduction à la gestion de projet logiciel.

- *Workflow Modeling : Tools for Process Improvement and Application Development*¹³. Certains chapitres traitent exclusivement de la modélisation d'un processus de travail et ont été utiles pour la modélisation des activités de vente chez AdaCore.

2.2 Le Travail d'enquête

Les premières semaines du stage ont donc été consacrées à la découverte des métiers de vente chez AdaCore et à leur décortication. J'ai pour cela réalisé de multiples entrevues avec les principales personnes concernées, en particulier Zépur Blot et Karen Mason (par téléphone pour cette dernière, travaillant dans les bureaux de New-York) — que je tiens à remercier pour leur coopération et la bonne humeur avec laquelle elles ont accepté de répondre à mes (très) nombreuses questions. Ce travail, nécessaire à la suite du projet, a également été pour moi l'occasion de m'introduire auprès des employés d'AdaCore et de me familiariser avec ce nouvel entourage.

À ces entrevues, s'ajoute la consultation d'un échantillon de dossiers client : Zépur tient à jour un dossier pour chaque client dans lequel sont conservés au format papier tous les échanges avec un client (emails, lettres, fax, comptes-rendus de communications téléphoniques, devis, ordres d'achat, factures, etc.). Ces documents, ainsi que mon abonnement aux *mailing-lists* « sales », ont été précieux pour avoir une idée encore plus concrète de la relation avec le client.

2.3 Intérêts de la modélisation

Au fur et à mesure de la découverte des processus de vente, la modélisation de ceux-ci s'étoffe et se complète de plus en plus. Outre le fait de permettre d'avoir une idée *visuelle* des processus, la modélisation a ici deux intérêts annexes, loin d'être négligeables :

- La communication : ces schémas sont en effet un outil de communication avec les autres employés. Il a rendu en effet très facile, deux

¹³Alec Sharp et Patrick McDermott, 1st edition, Février 2001, ISBN 1580530214.

semaines après mon arrivée, la présentation au personnel européen d'AdaCore des processus de vente. Ce support visuel leur a permis de me formuler aisément leurs commentaires, de mettre le doigt sur les inexactitudes ou les oublis.

- La motivation : cette modélisation permet de *voir avancer* son propre travail de manière concrète, tangible. Cela constitue — en cela-même — un encouragement, une motivation à continuer le travail en cours.

En dehors de ces avantages annexes, le fait de travailler sur cette modélisation oblige à s'imprégner totalement de la chaîne de processus en jeu dans les activités de vente. Encore une fois, il est primordial de comprendre parfaitement cet enchaînement avant de pouvoir prétendre apporter une solution au problème posé. L'étape de modélisation est en cela très fructueuse. Elle permettra de garder en tête, tout le long de la phase de *design* du logiciel, les différentes phases qui constituent la prise en main d'un prospect, la négociation d'un contrat, le suivi d'un client, etc., afin que tous les cas d'utilisation (*use cases*) du futur outil soient bien prévus dans la maquette. En cela, cette modélisation constitue d'une certaine manière un cahier des charges du logiciel de CRM.

2.4 Résultats de la modélisation

Le résultat du travail de modélisation — présenté en annexe **A** — permet de faire ressortir les différents outils existants (par exemple, pour la rédaction de devis, la génération de factures, la création d'un compte client, etc.) qu'il sera important de prendre en compte lors de la future phase de transition et d'intégration de l'outil de CRM. Il permet également de mettre en lumière l'*absence* d'outils dans certains domaines (par exemple pour la gestion des factures) : c'est dans de tels cas qu'un outil informatique de traitement de l'information peut s'avérer très pertinent.

Enfin, il est possible de faire ressortir du schéma l'ensemble des documents (lettres, fax, emails, contrats, factures, etc.) qui interviennent dans les processus de vente. Comme écrit précédemment, tous ces documents sont à ce jour stockés sous format papier et classés dans des dossiers par client. Cette solution passerait manifestement difficilement à l'échelle et n'est donc pas satisfaisante à long terme. Ce sera donc l'un des rôles les plus importants de l'outil de CRM que de permettre l'archivage de tous ces documents sous forme numérique et de les classer selon le client ou prospect concerné.

3 Design : construction d'une maquette

3.1 L'Interface utilisateur

L'outil de CRM est destiné à devenir l'outil de travail quotidien des personnes touchant aux activités de vente de l'entreprise et un outil occasionnel du personnel technique souhaitant obtenir des informations de manière ponctuelle sur un client.

Par conséquent, l'interface utilisateur du logiciel sera l'une des clés de la réussite du projet : une mauvaise interface condamnera d'emblée l'outil à un avenir incertain, tandis qu'une interface bien pensée sera un atout essentiel pour l'adoption de l'outil par les employés. Il est des projets dont la difficulté réside dans des aspects techniques, algorithmiques ou autres : la phase de conception se concentrera alors sur le recherche de solutions techniques, la mise au point d'algorithmes, ou autres. Ici, le point délicat est l'interface utilisateur : où faire apparaître l'information ? Sous quelle forme ? Comment en permettre l'édition ? Comment établir des passerelles entre différents types d'information ? Autant de questions qui sont incontournables lors de la conception du logiciel.

3.2 Le Choix d'une interface Web

La technologie à utiliser pour l'interface utilisateur des clients était fixée depuis longtemps : ce serait une interface Web. Les raisons à cela sont multiples, la plus importante est la contrainte de compatibilité : les clients doivent pouvoir fonctionner indifféremment sous Linux, Mac ou Windows pour ne citer qu'eux. Avec une interface Web, le client est un navigateur : même si le respect des standards (HTML, CSS, DOM, etc.) varie d'un navigateur à un autre, la contrainte de compatibilité est quasiment acquise d'office avec ce choix d'interface. Un autre intérêt est justement le fait qu'aucune installation d'un logiciel client n'est nécessaire pour accéder aux données du CRM : n'importe quel ordinateur équipé d'une connexion Internet et d'un navigateur peut faire l'affaire.

Pendant cette longue phase de *design* (3 mois entiers, de septembre à novembre), il est important de ne pas trop s'isoler et ainsi perdre de vue les utilisateurs finaux : régulièrement, des réunions de mi-parcours doivent être organisées, afin de s'assurer que les attentes de chacun sont bien respectées

et que le produit final ne diverge pas du produit demandé. Un passage par les bureaux de New-York a même permis une présentation « en direct » de la maquette en cours de finalisation.

3.3 Réalisation d'une maquette

Ce travail de *design* nécessite beaucoup d'imagination : il faut garder son esprit ouvert à toute solution, toute idée nouvelle qui puisse contribuer à une meilleure ergonomie de l'interface est bonne à considérer. Pour cela, il faut non seulement être soi-même inventif, mais aussi et surtout, être à l'écoute de toute suggestion venue d'une tierce personne : certaines idées, même — surtout ? — évoquées au détour d'une conversation informelle, peuvent receler la clé d'un problème d'interfaçage.

La grande difficulté dans la conception d'une telle interface est la prise en compte de tous les cas particuliers qui peuvent survenir lors de négociations avec un client. Le point de départ pour la construction de la maquette est le travail précédent de modélisation des processus de vente : chaque circuit dans les diagrammes réalisés doit pouvoir être parcouru aisément et intuitivement à travers l'interface. Mais une fois tous ces circuits, ces *use cases*, pris en compte, restent les nombreuses exceptions, cas particuliers, que le logiciel doit être en mesure de supporter : l'outil de CRM, même dans ces cas-là, ne doit pas être un obstacle, mais une aide. C'est la raison pour laquelle l'accent a été mis sur la *modularité* de l'outil, donc de l'interface qui l'accompagne. En même temps que cette modularité, il est souhaitable de pouvoir lier le maximum d'éléments entre eux (par exemple, être en mesure de lier des factures à l'ordre d'achat correspondant et aux contrats qu'elles concernent). Ces deux objectifs contradictoires — modularité/souplesse *vs.* « connectivité »/rigidité — doivent se résoudre en un compromis adéquat, qu'il n'est pas toujours évident de trouver.

3.4 Conception du schéma de la base de données

Une fois cette maquette (un site HTML entièrement statique) suffisamment complète, souple et validée par les plus intéressés, les structures des données commencent déjà à apparaître d'elles-mêmes : la manière dont sont structurées les pages et la manière dont les différents écrans sont liés entre eux laissent en effet apparaître en filigrane ce que sera la base de données

sous-jacente au logiciel.

Étant donnée la complexité et la quantité des données à stocker et à structurer dans le cadre de l'outil de CRM, l'appui sur un système de gestion de base de données était une décision prise depuis longtemps.

L'annexe B présente le schéma de base de données relationnelle tel qu'il est à l'état actuel. Le long travail d'élaboration de l'interface web réalisé au préalable a rendu la conception de ce schéma extrêmement rapide : la structure des données découle littéralement du travail effectué auparavant.

4 Implantation

À ce stade, il s'agit de créer la base de données et de commencer l'implantation du serveur web de CRM.

4.1 Base de données PostgreSQL

Le choix de PostgreSQL¹⁴ comme SGBD¹⁵ était le résultat de deux contraintes : celle d'avoir un *véritable* SGBD, respectant notamment les propriétés ACID (*Atomicity, Consistency, Isolation, Durability*) et celle d'utiliser, dans la mesure du possible, des logiciels *open source*.

Le travail de traduction du schéma de la base de données en instructions de création SQL est un travail systématique qui s'est fait en parallèle avec la construction du schéma entité-associations évoqué plus haut.

4.2 L'Utilisation de la technologie AWS

Pour la réalisation du serveur web, le choix s'est porté sur la technologie *Ada Web Server* (AWS) développée par Pascal Obry et Dmitriy Anisimkov, tous deux d'AdaCore. Cette librairie Ada a déjà été utilisée avec succès dans le cadre de *GNAT Tracker*, site technique mis en place pour les clients d'AdaCore.

¹⁴Voir <http://www.postgresql.org/>.

¹⁵SGBD : Système de Gestion de Base de Données.

L'architecture globale a été en partie inspirée de l'article de J.-P. Rosen : *Experiences in developing a typical web/database application*¹⁶.

Pour la génération de pages HTML à partir de modèles, le package `Ada Templates_Parser` a été utilisé. Un bon usage des modèles et du package `Ada` associé permet de bien séparer l'affichage de la partie traitement des données, tandis que l'utilisation de *stylesheets* CSS permet en outre de séparer la mise en forme du contenu brut.

Une API relativement simple et générale a été mise en place pour le remplissage des modèles HTML directement à partir des données de la base et a permis de parvenir à couvrir très rapidement une grande partie de l'interface utilisateur.

5 Passation du projet

Les dernières semaines de mon stage ont été essentiellement consacrées à la passation du projet à d'autres développeurs. Un ingénieur de New-York, Albert Lee, s'est vu confier la responsabilité du module de gestion et traitement des emails, tandis que Jérôme Roussel, élève en deuxième année de l'ENST en stage chez AdaCore sous la direction d'Emmanuel Briot, se charge de compléter entièrement l'interface web.

Cette phase a été riche en réunions et en travail d'équipe. Un accent a également été mis sur la rédaction de toute la documentation nécessaire à la reprise du projet par d'autres, même si l'essentiel de la documentation avait été réalisée en parallèle du travail d'implantation.

La confrontation de mon travail à d'autres développeurs a été très enrichissante. Elle a fait naître des idées à la fois pour le logiciel de CRM en cours de construction, mais aussi pour d'autres projets tels que la suite d'outils utilisée en interne par AdaCore pour gérer la partie technique. En effet, l'utilisation faite des bases de données dans le CRM en a séduit certains, qui pensent pouvoir avantageusement tirer parti de cette technologie dans d'autres contextes.

La prochaine étape délicate du projet sera l'importation — depuis des sources diverses — des données client existantes vers la base de données. Puis, une période d'essai et de transition délicate à gérer sera certainement

¹⁶*Annual International Conference on Ada*, 2003, ISBN 1-58113-476-2.

décisive dans l'utilisation à terme de l'outil de CRM.

Conclusion

Contrairement à ce que la construction de ce rapport peut le laisser entendre, le développement de ce projet n'est pas tout-à-fait linéaire : même si les étapes semblent se succéder de manière linéaire, les retours à des étapes antérieures sont fréquents et nécessaires à la suite du projet. Par exemple, le travail sur le schéma de la base de données a pu faire apparaître des manques dans l'interface prévue par la maquette. Ce phénomène, loin d'être inconnu, a confirmé les enseignements que j'ai pu recevoir, notamment dans le cadre de la brique BDL.

Si j'ai pu aboutir à ce résultat au bout des sept mois de stage, c'est avant tout grâce à tous ceux qui croyaient en l'utilité de ce projet et qui m'ont aidé à le porter jusqu'à ce stade. Cependant, ce projet ne sera véritablement mené à bien que le jour où il sera devenu l'outil quotidien et apprécié des employés d'AdaCore.

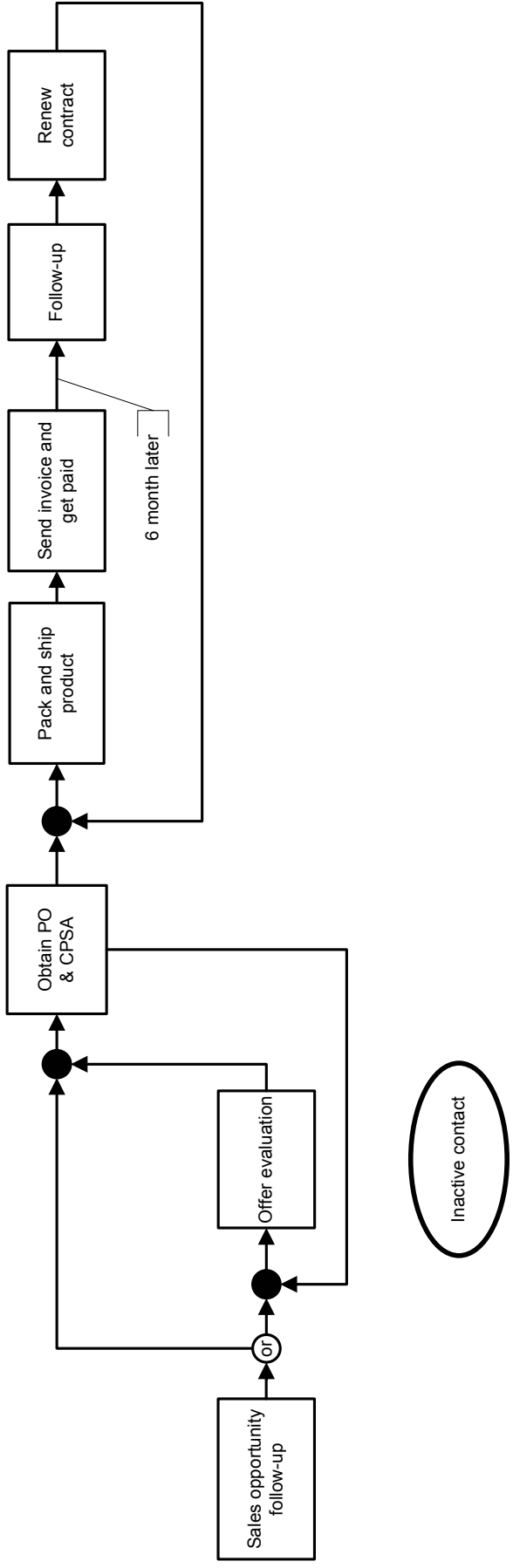
A Sales Workflow Diagrams

Les pages suivantes représentent les diagrammes modélisant le processus de vente tel qu'il existait à mon arrivée chez AdaCore. Le premier schéma représente une vue globale de l'activité de vente, tandis que les pages suivantes détaillent chacune des étapes introduites dans le premier schéma.

Ces schémas ont été réalisés sous Microsoft Office Visio 2003 Professionnel.

Global sales process

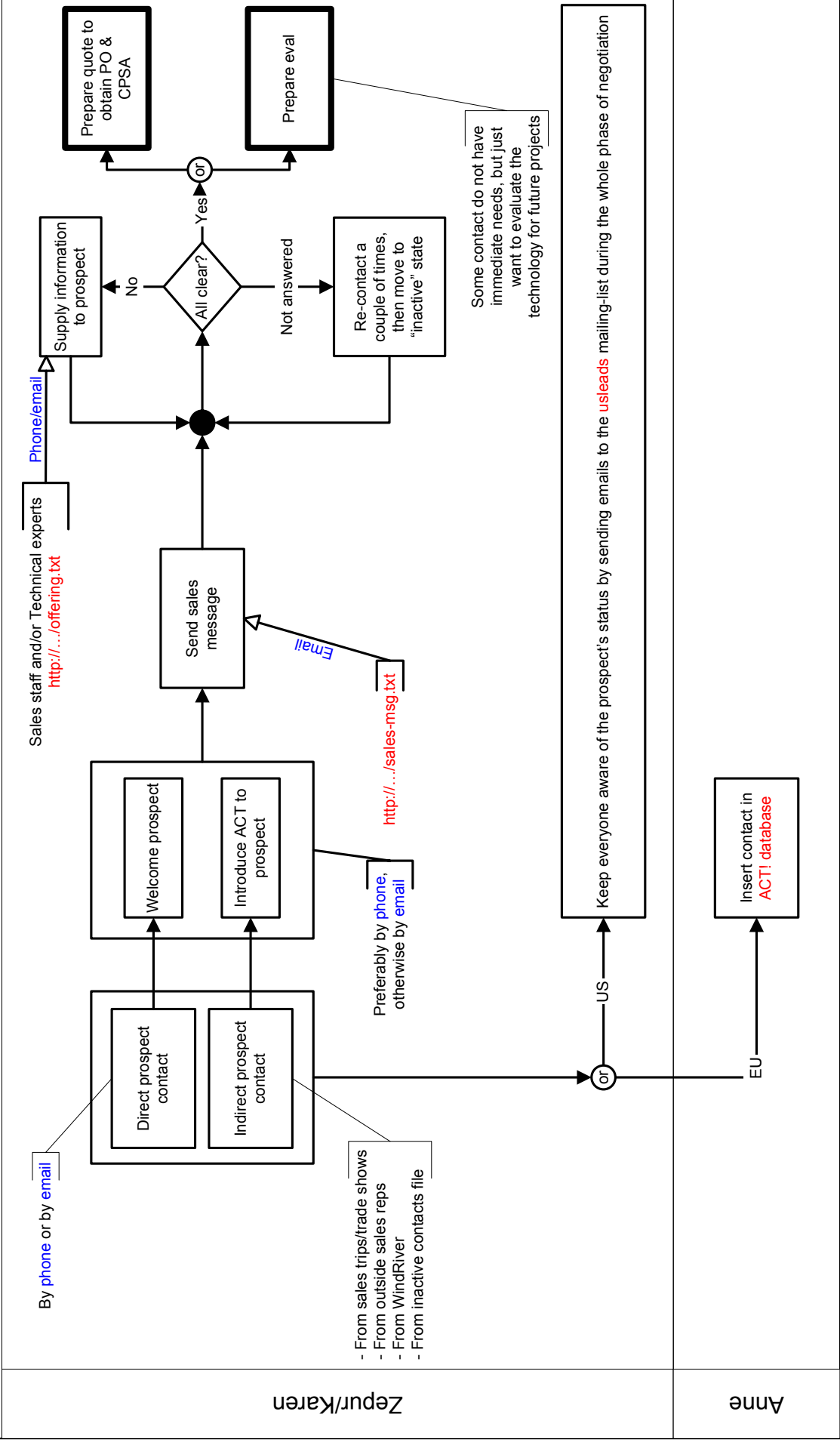
Sales personnel



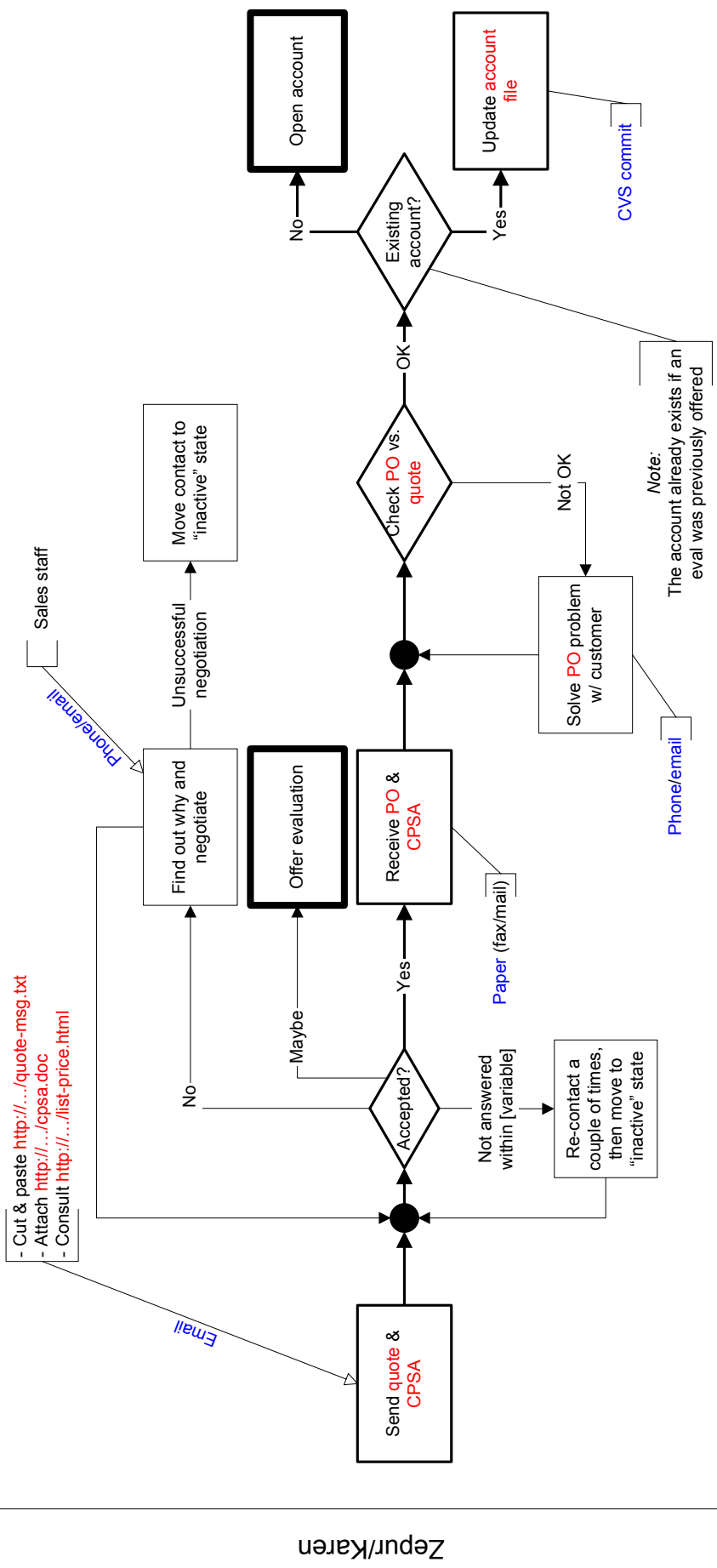
Notes:
 - During all sub-processes, emails, phone call reports, fax and mail are archived in a per-customer paper file, hereafter called "customer file (paper)"
 - In many of the above steps, a prospect can be moved to the "inactive" state (meaning, no current contract and no recent contact)

Legend:
 Documents involved are in red
 Tools that are used are in blue

Sales opportunity follow-up sub-process



Obtain PO & CPSA sub-process



Zepur/karen

Offer evaluation sub-process

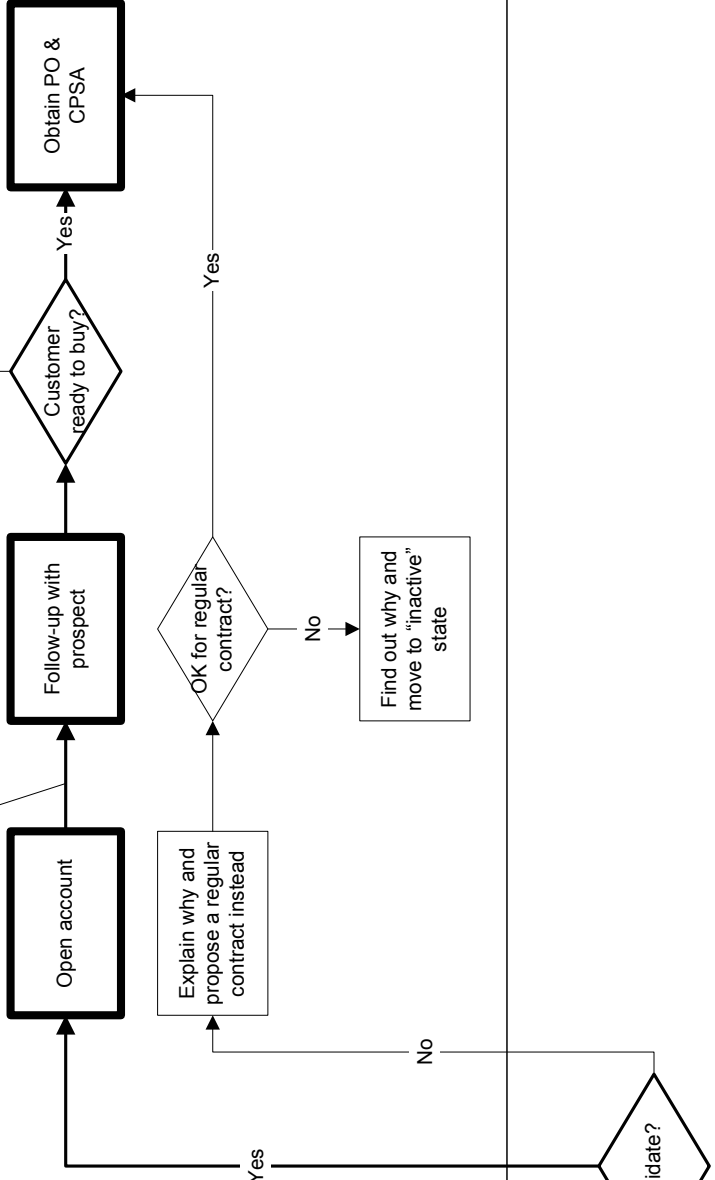
- Attach [http://.../cpsa.doc](#) (converted to pdf)
- Attach [http://.../test-plan.doc](#) (converted to pdf)
- Attach (if applicable) [http://.../quote-msg.txt](#)

Email

Send:
 - eval-CPSA
 - Test plan
 - quote (if applicable)

Note:
 The prospect might have already been sent a quote

6 weeks later



All docs in paper

Analyze Test Plan & eCPSA

Validate?

Explain why and propose a regular contract instead

OK for regular contract?

Follow-up with prospect

Open account

Obtain PO & CPSA

Find out why and:
 - extend evaluation period
 or
 - move to "inactive" state

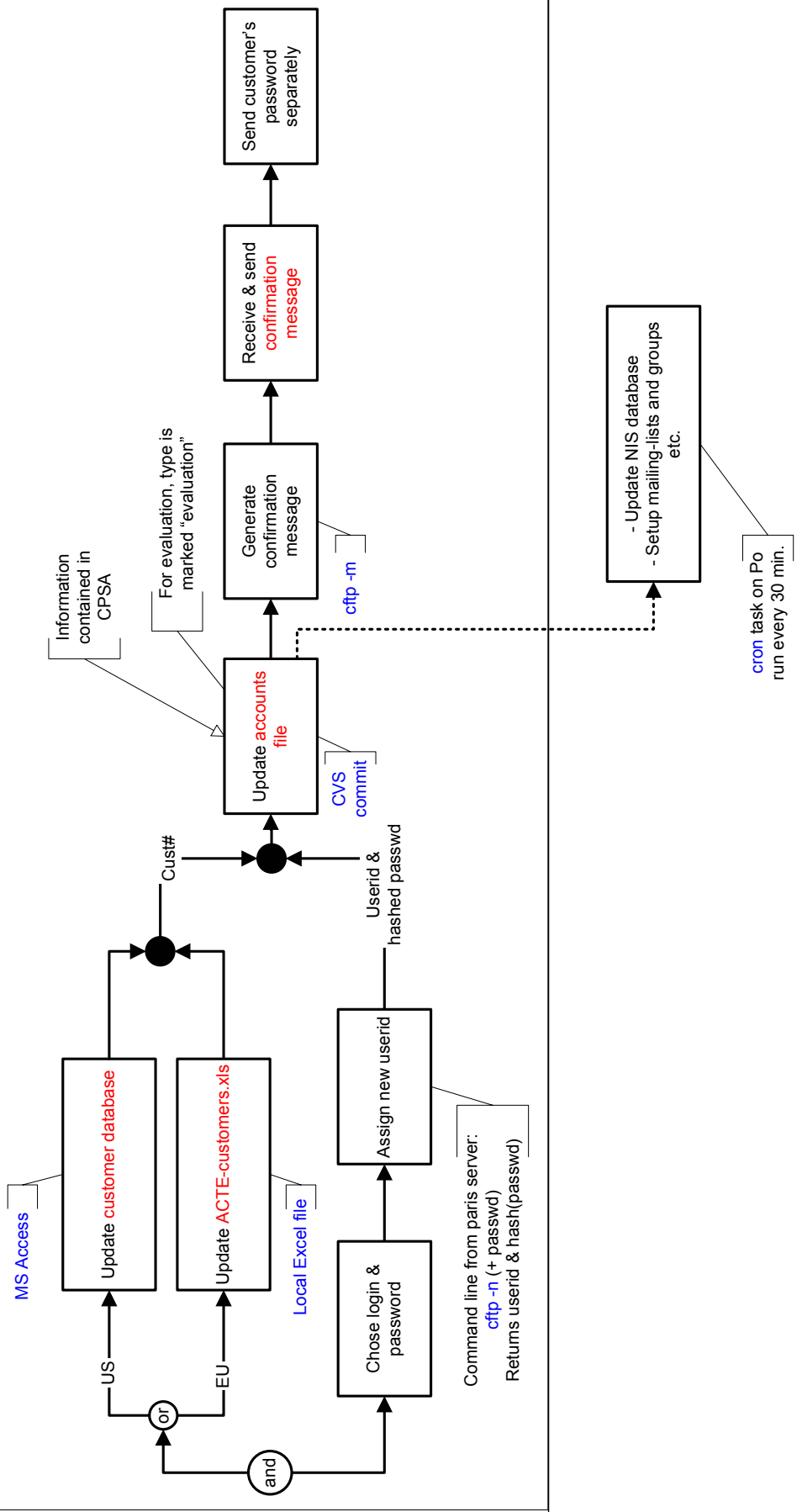
Find out why and move to "inactive" state

Zepur/Karen

Cyrille/Bernard

Open Customer Account sub-process

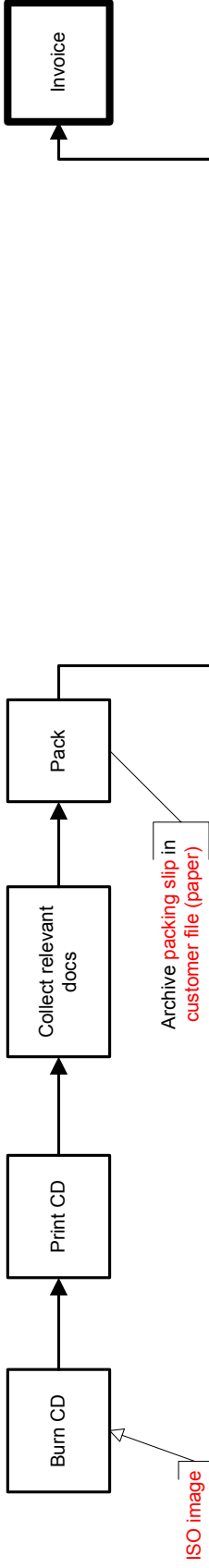
Zepur/Karen



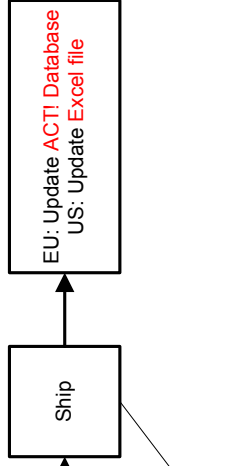
Note:
 This process is being transformed and simplified: "Chose login", "Assign userid", "Update account file", "Generate confirmation message" are to be regrouped into one step via the web interface of GNAT Tracker.

Pack & Ship sub-process

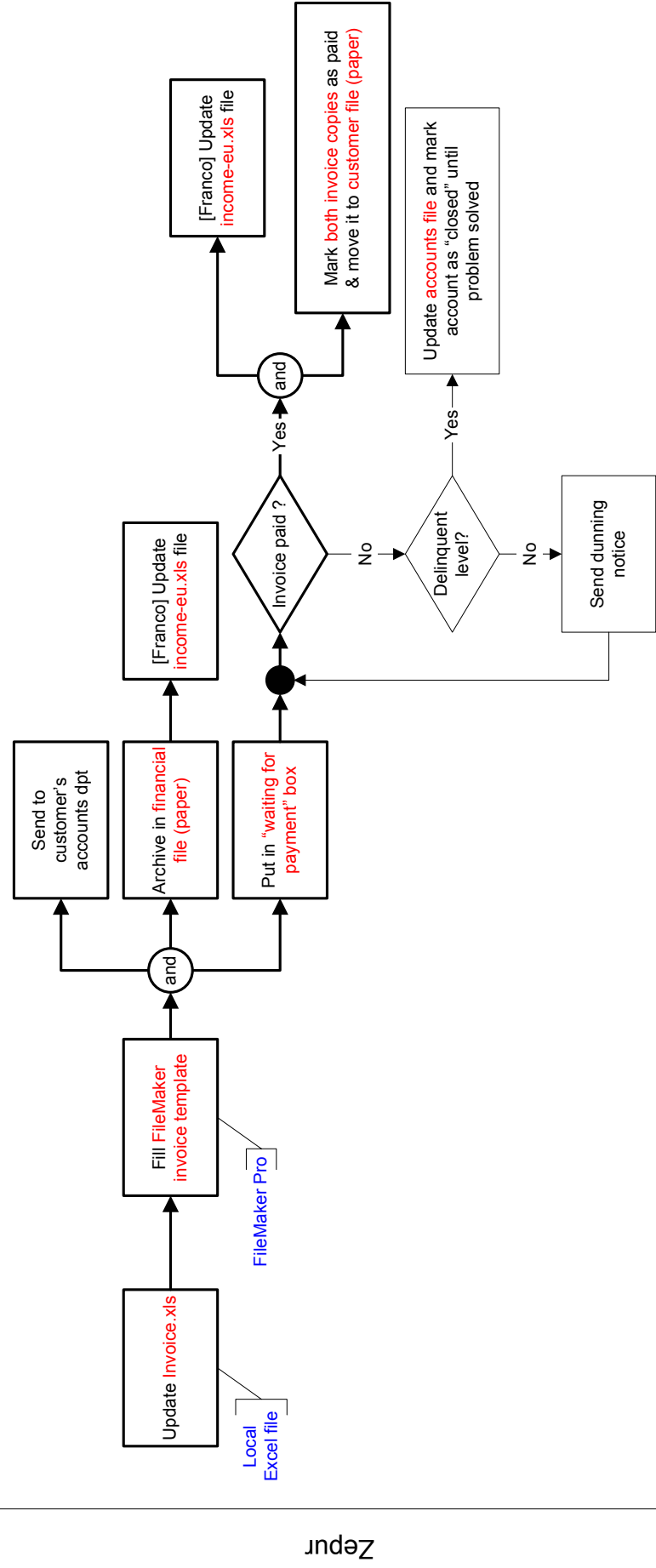
Sales Dpt
(Zepur/Anne/Karen)



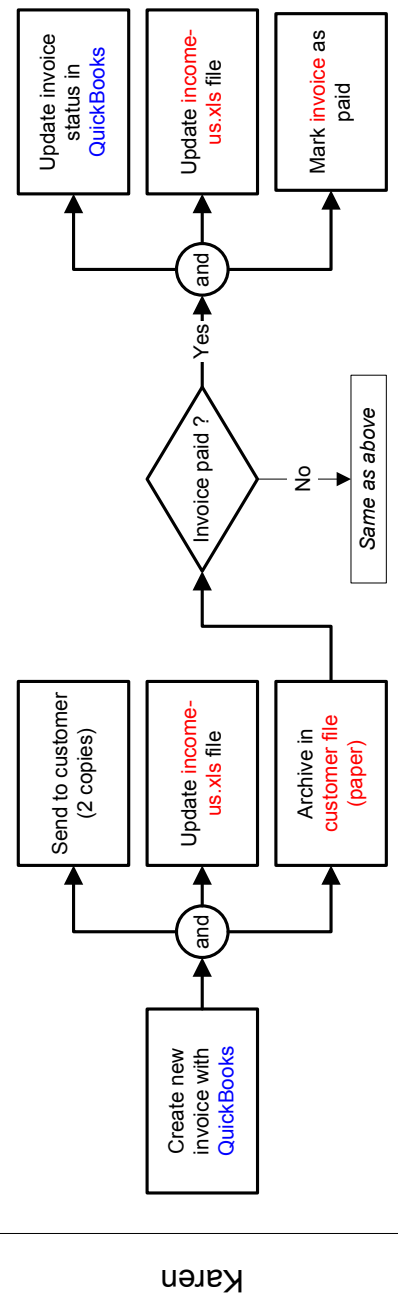
Ship Dpt
(Zepur/Anne/Jennifer)



Invoice sub-process



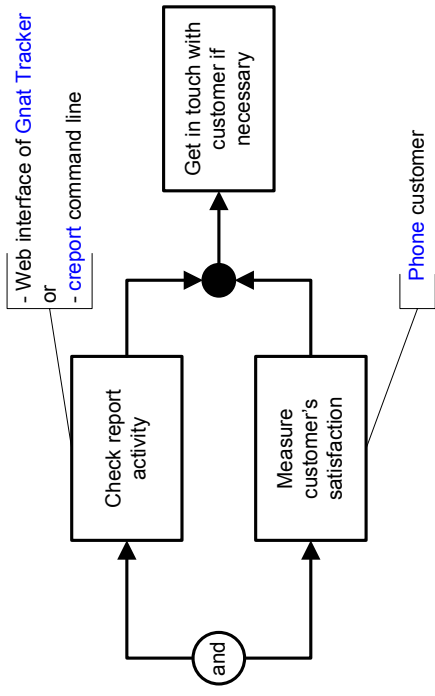
Zepur



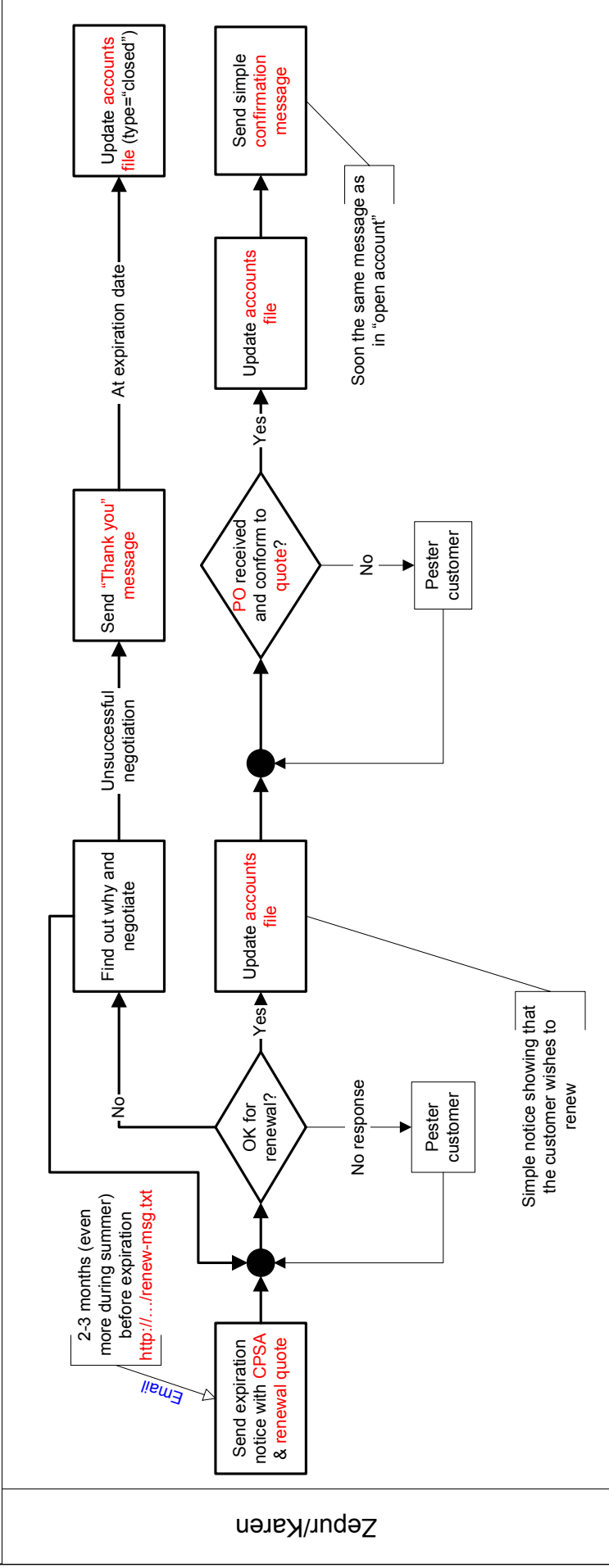
Karen

Follow-up sub-process

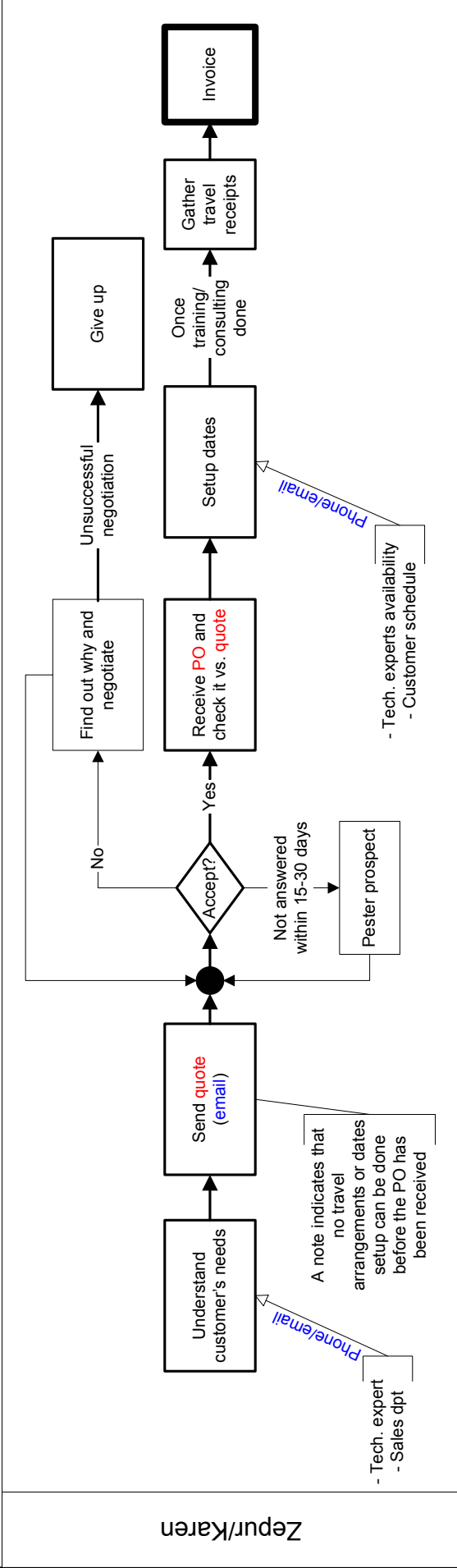
Zepur/Karen



Renew sub-process

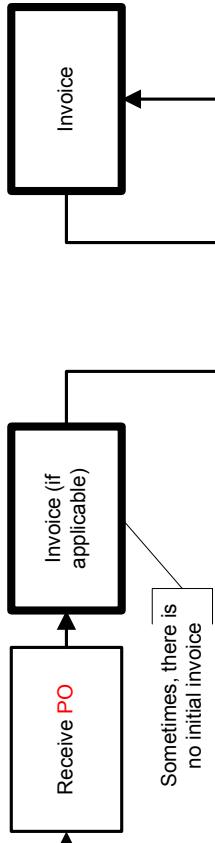


Customer requests training/consulting

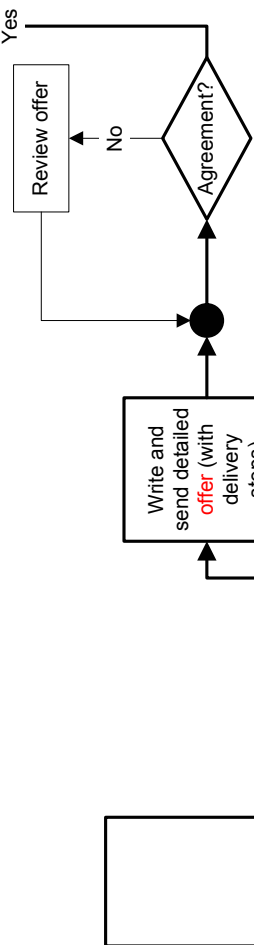


Customer requests special development/enhancement

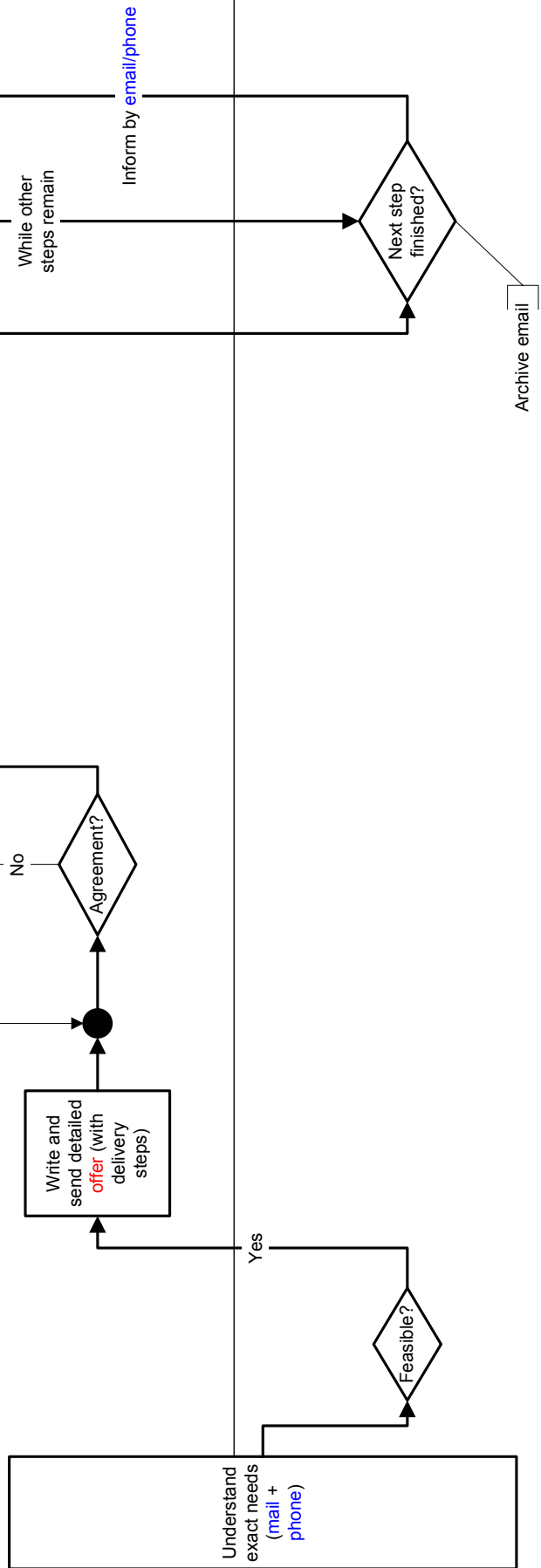
Zepur



Franco/Bernard/Robert



Technical personnel



B ER Diagram

Le schéma de la page suivante est celui de la base de données relationnelle adoptée. Le schéma a été réalisé sous Microsoft Office Visio 2003 Professionnel. Afin d'éviter une trop forte surcharge du diagramme, toutes les relations entre les tables ne sont pas indiquées explicitement par un lien. Une légende explique brièvement les codes adoptés.

