

Groupe de travail Réseau
Request for Comments : 4112
RFC mise à jour : 3106
 Catégorie : Sur la voie de la normalisation

D. Eastlake 3rd, Motorola Laboratories

juin 2005

Traduction Claude Brière de L'Isle

Spécification du langage de modélisation du commerce électronique (ECML) version 2

Statut de ce mémoire

Le présent document spécifie un protocole de l'Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Normes officielles des protocoles de l'Internet" (STD 1) pour connaître l'état de la normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Notice de copyright

Copyright (C) The Internet Society (2005).

Résumé

Le commerce électronique exige fréquemment un échange substantiel d'informations afin de mener à bien un achat ou une autre transaction, en particulier la première fois que les parties communiquent. Un ensemble standard de noms de champs d'informations en relation avec le paiement, organisés hiérarchiquement dans la syntaxe de XML est défini afin que cette tâche puisse être plus facilement automatisée. C'est la seconde version d'un langage de modélisation du commerce électronique (ECML, *Electronic Commerce Modeling Language*) et elle est destinée à satisfaire aux exigences de la RFC 3505.

Table des Matières

1. Introduction.....	1
1.2 Histoire et relations aux autres normes.....	2
2. Définitions de champs, DTD, et schémas.....	2
2.1 Liste et description des champs.....	2
2.2 Syntaxe XML exemplaire.....	7
3. Notes d'utilisation pour ECML v2.....	18
3.1 Présentation des champs.....	18
3.2 Méthodes et flux de réglage des champs.....	19
4. Considérations sur la sécurité et la confidentialité.....	19
5. Considérations relatives à l'IANA.....	20
5.1 Gabarit de schéma ECML v2.....	20
5.2 Gabarit d'URN ECML v2.....	20
5.3 Registres IANA.....	20
6. Remerciements.....	20
Appendice A. Changements de la v1.1 à la v2.....	21
Références normatives.....	21
Références pour information.....	21
Adresse de l'auteur.....	22
Déclaration complète de droits de reproduction.....	22

1. Introduction

Nombreux sont ceux qui font des affaires sur l'Internet en utilisant des champs et formulaires ad hoc. Les formats et la structure des données peuvent varier considérablement d'un acteur à l'autre. Lorsque les formulaires sont remplis à la main, certains utilisateurs sont perturbés par leur diversité, et le processus de remplissage manuel de ces formulaires peut être fastidieux et sujet à erreurs.

Des outils logiciels, y compris des portefeuilles électroniques, peuvent aider dans cette situation. De tels outils peuvent aider à mener à bien des transactions en ligne en mémorisant les informations de facturation, d'expédition, de paiement, de préférences, et ainsi de suite, et en utilisant ces informations pour compléter automatiquement les ensembles de données requis par les interactions. Par exemple, un logiciel qui remplit les formulaires a été construit avec succès dans les

navigateurs, comme serveurs mandataires, comme applications d'aide aux navigateurs, et comme applications autonomes, comme modules d'extension de navigateurs, et comme applications fondées sur un serveur. Mais la prolifération de logiciels de transaction plus automatisés a été entravée par le manque de normes.

Le langage de modélisation du commerce électronique (ECML, *Electronic Commerce Modeling Language*) fournit un ensemble de structures de données hiérarchisées orientées paiement qui vont permettre des logiciels automatisés, y compris des portefeuilles électroniques de multiples fournisseurs, pour fournir et demander les données nécessaires d'une façon plus uniforme.

La version 2.0 étend les versions 1.0 [RFC2706] et 1.1 [RFC3106] de ECML comme décrit à l'Appendice du présent document. Ces améliorations incluent la prise en charge de mécanismes de paiement et d'informations de transaction supplémentaires et utilisent la syntaxe du langage de balisage extensible (XML, *extensible markup language*).

ECML est conçu pour fournir un simple support utile dans divers contextes. Les utilisations probables de ECML v2 sont l'entrée des informations de paiement du consommateur et les transactions d'affaires. Pour l'instant, les premières vont probablement se faire avec des formulaires en langage de balisage hypertexte (HTML, *HyperText Markup Language*). Les secondes vont très probablement utiliser des documents XML.

1.2 Histoire et relations aux autres normes

Les champs ECML étaient à l'origine dérivés du schéma de base de données P3P du W3C [P3P.BASE] par l'Alliance ECML comme décrit dans les [RFC2706], [RFC3106]. Le développement technique et le contrôle des changements à ECML a alors été transféré à l'IETF. Dans la version 2, ECML est étendu par les champs d'une note P3P du W3C relative au commerce électronique [P3P.ECOM], par [ISO8583], et d'autres sources. Son principal formulaire exemplaire est maintenant en syntaxe XML.

2. Définitions de champs, DTD, et schémas

ECML v2 est la définition et la désignation d'un ensemble hiérarchique structuré de champs et la fourniture d'une syntaxe XML facultative pour leur transmission. Ces champs peuvent être codés avec d'autres syntaxes. Sans considération du codage utilisé, les champs peuvent être transmis via divers protocoles.

Le paragraphe 2.1 donne la liste et la description des champs, le paragraphe 2.2.1 donne une déclaration de type de données (DTD, *Data Type Declaration*) XML à utiliser avec les champs, et le paragraphe 2.2.2 donne un schéma XML.

Pour se conformer au présent document, les noms de champs doivent être donnés et hiérarchiquement structurés aussi près que possible en pratique de la structure et des désignations figurant ci-dessous étant donnés la syntaxe et le protocole de transaction utilisés. (Note : ceci n'impose aucune restriction sur l'étiquetage des champs visibles par l'homme, seulement sur leur nom ou sur les noms et la structure tels qu'utilisés dans la communication sur le réseau.)

2.1 Liste et description des champs

La liste des champs figure ci-dessous, avec la taille minimum d'entrée des données permise. Les mises en œuvre peuvent accepter de plus grandes tailles de données, si cela a un sens, et, pour certaines applications, elles auront besoin de permettre des tailles de données supérieures.

Noter que ces champs sont organisés hiérarchiquement comme indiqué dans ce tableau par le caractère de soulignement ("_") incorporé. Les mécanismes appropriés de transmission des données peuvent utiliser cela pour demander et envoyer des agrégats, comme *Ecom_Payment_Card_ExpDate* (pour englober tout en ensemble de composants de date d'expiration de carte) ou *Ecom_ShipTo* (pour englober les composants d'adresse d'expédition que veut fournir un consommateur). L'étiquetage, l'arrangement et le dés-arrangement des composants de tels agrégats dépend du protocole de transfert de données utilisé. La syntaxe suggérée est XML comme spécifié au paragraphe 2.2.

2.1.1 Liste des champs

Le tableau ci-dessous est la liste des champs ECML v2.

La colonne Nom donne la chaîne de nom structurée de chaque champ comme expliqué plus haut. La colonne Minimum est la taille minimum de données qui DOIT être permise pour chaque entrée de données. Ce N'EST PAS la taille minimum du

contenu valide du champ, et le logiciel du commerçant devrait, dans tous les cas, être prêt à recevoir une valeur plus grande ou plus courte. Les commerçants qui traitent des zones où, par exemple, le nom d'état/province ou de numéro de téléphone est plus long que le minimum indiqué ci-dessous doivent évidemment permettre une entrée de données plus longue. Dans certains cas, cependant, il y a une taille maximum qui a du sens et lorsque c'est le cas, c'est généralement documenté dans une Note pour le champ.

Les champs suivants sont normalement utilisés pour communiquer du consommateur au commerçant :

Champ	Nom	Minimum	Notes
titre du destinataire	Ecom_ShipTo_Postal_Name_Prefix	4	(1)
prénom du destinataire	Ecom_ShipTo_Postal_Name_First	15	(54)
second prénom du destinataire	Ecom_ShipTo_Postal_Name_Middle	15	(2)
nom de famille du destinataire	Ecom_ShipTo_Postal_Name_Last	15	(54)
suffixe du nom du destinataire	Ecom_ShipTo_Postal_Name_Suffix	4	(3)
nom de compagnie destinataire	Ecom_ShipTo_Postal_Company	20	
rue de destination ligne 1	Ecom_ShipTo_Postal_Street_Line1	20	(4)
rue de destination ligne 2	Ecom_ShipTo_Postal_Street_Line2	20	(4)
rue de destination ligne 3	Ecom_ShipTo_Postal_Street_Line3	20	(4)
ville de destination	Ecom_ShipTo_Postal_City	22	
état/province de destination	Ecom_ShipTo_Postal_StateProv	2	(5)
code postal de destination	Ecom_ShipTo_Postal_PostalCode	14	(6)
pays de destination	Ecom_ShipTo_Postal_CountryCode	2	(7)
téléphone de destination	Ecom_ShipTo_Telecom_Phone_Number	10	(8)
mèl de destination	Ecom_ShipTo_Online_Email	40	(9)
titre de facturation	Ecom_BillTo_Postal_Name_Prefix	4	(1)
prénom de facturation	Ecom_BillTo_Postal_Name_First	15	(54)
second prénom de facturation	Ecom_BillTo_Postal_Name_Middle	15	(2)
nom de famille de facturation	Ecom_BillTo_Postal_Name_Last	15	(54)
suffixe du nom de facturation	Ecom_BillTo_Postal_Name_Suffix	4	(3)
nom de compagnie de facturation	Ecom_BillTo_Postal_Company	20	
rue de facturation ligne 1	Ecom_BillTo_Postal_Street_Line1	20	(4)
rue de facturation ligne 2	Ecom_BillTo_Postal_Street_Line2	20	(4)
rue de facturation ligne 3	Ecom_BillTo_Postal_Street_Line3	20	(4)
ville de facturation	Ecom_BillTo_Postal_City	22	
état/province de facturation	Ecom_BillTo_Postal_StateProv	2	(5)
code postal de facturation	Ecom_BillTo_Postal_PostalCode	14	(6)
pays de facturation	Ecom_BillTo_Postal_CountryCode	2	(7)
téléphone de facturation	Ecom_BillTo_Telecom_Phone_Number	10	(8)
mèl de facturation	Ecom_BillTo_Online_Email	40	(9)
récepissé à			(32)
titre du récepissé	Ecom_ReceiptTo_Postal_Name_Prefix	4	(1)
prénom du récepissé	Ecom_ReceiptTo_Postal_Name_First	15	(54)
second prénom du récepissé	Ecom_ReceiptTo_Postal_Name_Middle	15	(2)
nom de famille du récepissé	Ecom_ReceiptTo_Postal_Name_Last	15	(54)
suffixe du récepissé	Ecom_ReceiptTo_Postal_Name_Suffix	4	(3)
nom de compagnie de récepissé	Ecom_ReceiptTo_Postal_Company	20	
rue de récepissé ligne 1	Ecom_ReceiptTo_Postal_Street_Line1	20	(4)
rue de récepissé ligne 2	Ecom_ReceiptTo_Postal_Street_Line2	20	(4)
rue de récepissé ligne 3	Ecom_ReceiptTo_Postal_Street_Line3	20	(4)
ville de récepissé	Ecom_ReceiptTo_Postal_City	22	
état/province de récepissé	Ecom_ReceiptTo_Postal_StateProv	2	(5)
code postal de récepissé	Ecom_ReceiptTo_Postal_PostalCode	14	(6)
pays de récepissé	Ecom_ReceiptTo_Postal_CountryCode	2	(7)
téléphone de récepissé	Ecom_ReceiptTo_Telecom_Phone_Number	10	(8)
mèl de récepissé	Ecom_ReceiptTo_Online_Email	40	(9)
nom sur la carte	Ecom_Payment_Card_Name	30	(10)
type de carte	Ecom_Payment_Card_Type	4	(11)
numéro de carte	Ecom_Payment_Card_Number	19	(12)
valeur de vérification de carte	Ecom_Payment_Card_Verification	4	(13)
numéro du producteur de carte	Ecom_Payment_Card_IssueNumber	2	(53)
jour d'expiration de la carte	Ecom_Payment_Card_ExpDate_Day	2	(14)
mois d'expiration de la carte	Ecom_Payment_Card_ExpDate_Month	2	(15)
année d'expiration de la carte	Ecom_Payment_Card_ExpDate_Year	4	(16)

jour de fin de validité de carte	Ecom_Payment_Card_ValidFrom_Day	2	(14)
mois de fin de validité de carte	Ecom_Payment_Card_ValidFrom_Month	2	(15)
année de fin de validité de carte	Ecom_Payment_Card_ValidFrom_Year	4	(16)
protocoles de carte	Ecom_Payment_Card_Protocol	20	(17)
nom de carte de fidélité	Ecom_Loyalty_Card_Name	30	(10)
type de carte de fidélité	Ecom_Loyalty_Card_Type	20	(52)
numéro de carte de fidélité	Ecom_Loyalty_Card_Number	40	(34)
vérification de carte de fidélité	Ecom_Loyalty_Card_Verification	4	(13)
jour d'expiration de carte de fidélité	Ecom_Loyalty_Card_ExpDate_Day	2	(14)
mois d'expiration de carte de fidélité	Ecom_Loyalty_Card_ExpDate_Month	2	(15)
année d'expiration de carte de fidélité	Ecom_Loyalty_Card_ExpDate_Year	2	(16)
jour de fin de validité de carte de fidélité	Ecom_Loyalty_Card_ValidFrom_Day	2	(14)
mois de fin de validité de carte de fidélité	Ecom_Loyalty_Card_ValidFrom_Month	2	(15)
année de fin de validité de carte de fidélité	Ecom_Loyalty_Card_ValidFrom_Year	4	(16)
Identifiant d'ordre du consommateur	Ecom_ConsumerOrderID	20	(18)
Identifiant d'utilisateur	Ecom_User_ID	40	(19)
mot de passe d'utilisateur	Ecom_User_Password	20	(19)
certificat d'utilisateur	Ecom_User_Certificate_URL	128	(55)
pays de données d'utilisateur	Ecom_UserData_Country	2	(7)
langage des données d'utilisateur	Ecom_UserData_Language	30	(33)
genre des données d'utilisateur	Ecom_UserData_Gender	1	(36)
jour de naissance de l'utilisateur	Ecom_UserData_BirthDate_Day	2	(14)
mois de naissance de l'utilisateur	Ecom_UserData_BirthDate_Month	2	(15)
année de naissance de l'utilisateur	Ecom_UserData_BirthDate_Year	4	(16)
données de préférences de l'utilisateur	Ecom_UserData_Preferences	60	(34)
version du schéma	Ecom_SchemaVersion	30	(20)
identifiant de portefeuille	Ecom_WalletID	40	(21)
URL du portefeuille	Ecom_Wallet_Location	128	(35)
identifiant de l'appareil du consommateur	Ecom_Device_ID	20	(37)
type de l'appareil du consommateur	Ecom_Device_Type	20	(38)
fanion de fin de transaction	Ecom_TransactionComplete	-	(22)

Les champs suivants sont normalement utilisés pour communiquer du commerçant au consommateur :

Champ	Nom	Minimum	Notes
domaine de rattachement du commerçant	Ecom_Merchant	128	(23)
domaine de rattachement du processeur	Ecom_Processor	128	(24)
identifiant de transaction	Ecom_Transaction_ID	128	(25)
URL d'enquête de transaction	Ecom_Transaction_Inquiry	500	(26)
montant de la transaction	Ecom_Transaction_Amount	128	(27)
monnaie de la transaction	Ecom_Transaction_CurrencyCode	3	(28)
date de transaction	Ecom_Transaction_Date	80	(29)
type de transaction	Ecom_Transaction_Type	24	(30)
signature de la transaction	Ecom_Transaction_Signature	160	(31)
fanion de fin de transaction	Ecom_TransactionComplete	-	(22)

Les champs suivants sont utilisés pour communiquer entre le commerçant et un processeur agissant pour le commerçant (un tel processeur est généralement appelé un acquéreur et est souvent une banque):

Champ	Nom	Minimum	Notes
identifiant du commerçant	Ecom_Merchant_ID	8	
terminal du commerçant	Ecom_Merchant_Terminal_ID	8	(39)
données du terminal du commerçant	Ecom_Merchant_Terminal_Data	128	
code de traitement de transaction	Ecom_Transaction_ProcessingCode	6	(40)
référence de transaction	Ecom_Transaction_Reference_ID	12	
acquéreur de la transaction	Ecom_Transaction_Acquire_ID	13	(41)
identifiant de transmission de transaction	Ecom_Transaction_Forward_ID	13	(42)
trace de transaction	Ecom_Transaction_Trace_Audit	6	(43)
date effective de transaction	Ecom_Transaction_Effective_Date	4	(44)
CID de transaction	Ecom_Transaction_CID	8	
POS de transaction	Ecom_Transaction_POSCode	12	(45)
utilisation privée de transaction	Ecom_Transaction_PrivateUseData	166	
réponse de transaction	Ecom_Transaction_ResponseData	27	

code d'approbation de transaction	Ecom_Transaction_ApprovalCode	12	(46)
code de restitution de transaction	Ecom_Transaction_RetrievalCode	128	
action de réponse de transaction	Ecom_Transaction_ActionCode	13	(47)
raison de la transaction	Ecom_Transaction_ReasonCode	4	
AAV de la transaction	Ecom_Transaction_AAV	3	
date de règlement de la transaction	Ecom_Transaction_Settle_Date	4	(48)
date de capture de la transaction	Ecom_Transaction_Capture_Date	4	(49)
données de trace 1 de la transaction	Ecom_Transaction_Track1	39	(50)
données de trace 2 de la transaction	Ecom_Transaction_Track2	39	(51)

Notes :

- (1) Par exemple : M., Mme, Mlle, Dr. Ce champ est souvent omis.
- (2) Peut aussi être utilisé pour une initiale
- (3) Par exemple: Ph.D., Jr. (Junior), 3rd, Esq. (Esquire). Ce champ est souvent omis.
- (4) Les lignes d'adresse doivent être remplies dans l'ordre, ligne1, puis ligne2, et enfin ligne3. Donc, par exemple, il y a une erreur si la ligne1 est nulle si la ligne2 ou la ligne3 ne l'est pas.
- (5) 2 caractères est le minimum pour les USA et le Canada ; les autres pays peuvent exiger des champs plus longs. Pour les USA, utiliser l'abréviation d'état postal US à deux caractères.
- (6) les longueurs minimum de champ pour le code postal vont varier selon le marché international servi. Utiliser le code postal à 5 caractères ou 5 + 4 ZIP pour les USA et le code postal à six caractères pour le Canada. La taille donnée, 14, est estimée être le maximum requis partout dans le monde.
- (7) Utiliser les codes de pays à deux lettres de la norme [ISO3166].
- (8) 10 chiffres est le minimum pour les numéros au sein du plan de numérotage de l'Amérique du Nord (<http://www.nanpa.com> : Il inclut les USA, le Canada et un certain nombre de petits états des Caraïbes et du Pacifique, mais pas Cuba). D'autres pays peuvent exiger des champs plus longs. Les numéros de téléphone sont compliqués par des codes d'accès internationaux différents, une ponctuation variable des codes de ville/cité au sein des pays, etc. Bien qu'il soit souhaitable que les numéros de téléphone soient au format standard international [E.164], il peut être nécessaire d'utiliser une heuristique ou un examen humain sur la base du numéro de téléphone et des adresses données pour voir comment appeler un utilisateur, car les gens peuvent entrer des numéros de format local sans les codes de zone/accès. Il est recommandé qu'un "x" soit placé avant les numéros d'extension et que le "x" et le numéro d'extension apparaissent après toutes les autres parties du numéro.
- (9) Par exemple : jsmith@exemple.com
- (10) Le nom du titulaire de la carte comme il apparaît sur la carte.
- (11) Insensible à la casse. Utiliser jusqu'aux quatre premières lettres du nom d'association (voir aussi la Note 102) :
 AMER : American Express
 BANK : Bankcard (Australie)
 DC : DC (Japon)
 DINE : Diners Club
 DISC : Discover
 JCB : JCB
 MAST : Mastercard
 NIKO : Nikos (Japon)
 SAIS : Saison (Japon)
 UC : UC (Japon)
 UCAR : UCard (Taïwan)
 VISA : Visa
- (12) Inclut le chiffre de contrôle à la fin mais pas d'espaces ni de traits d'union [ISO7812]. Le minimum donné, 19, est le plus long numéro permis par la norme ISO.
- (13) Numéro additionnel de vérification du titulaire de la carte imprimé sur la carte (mais non embossé ni enregistré sur la bande magnétique) comme les valeurs CIV d'American Express, CVC2 de MasterCard, et CVV2 de Visa.

- (14) Le jour du mois. Valeurs de 1 à 31. Un zéro en tête est ignoré, donc par exemple, 07 est valide pour le septième jour du mois.
- (15) Le mois de l'année. Janvier - 1, Février - 2, Mars - 3, etc. ; Valeurs de 1 à 12. Un zéro en tête est ignoré, donc par exemple, 07 est valide pour Juillet.
- (16) La valeur dans la cellule portefeuille est toujours de quatre chiffres ; par exemple, 1999, 2000, 2001.
- (17) Liste séparée par une espace des protocoles disponibles en connexion avec la carte spécifiée. La liste initiale des jetons insensibles à la casse est : none set setcert iotp echeck simcard phoneid.
"Set" indique que la carte est utilisable avec le protocole SET (c'est-à-dire, elle est dans un portefeuille SET) mais qu'elle n'a pas de certificat SET [SET]. "Setcert" indique que la carte est utilisable avec SET et un certificat set [SET]. "iotp" indique que le protocole IOTP [RFC2801] est pris en charge par le consommateur. "echeck" indique que le protocole eCheck [eCheck] est pris en charge chez le consommateur. "simcard" indique la capacité d'utiliser l'instrument de transaction installé dans le téléphone cellulaire de l'abonné pour l'identification. "phoneid" indique l'utilisation pour la transaction d'un numéro de téléphone facturable. "None" indique que le remplissage automatique fonctionne mais qu'il n'y a pas d'autre information.
- (18) Chaîne d'identifiant d'ordre univoque générée par le logiciel du consommateur.
- (19) Le champs Identifiant d'utilisateur et Mot de passe peuvent être utilisés si l'utilisateur a un compte pré établi auprès du commerçant auquel l'accès est authentifié par de telles valeurs. Pour cette utilisation, on s'attend à ce qu'une application exige exactement un identifiant d'utilisateur, et qu'un champ Mot de passe soit présent.
- (20) URI [RFC3986] qui indique la version de cet ensemble de champs. Égal à "urn:ietf:params:ecml:v2.0" pour cette version. Voir la Section 5. (Voir aussi la Note 101.)
- (21) Chaîne pour identifier la source et la version du logiciel de remplissage de formulaire qui agit au nom d'un utilisateur. Devrait contenir un nom de société et/ou un nom et version de produit ; par exemple, "Wallets Inc., SuperFill, v42.7". (Voir aussi la Note 101.)
- (22) Fanion pour indiquer que cette page/agrégat de la Toile est la dernière pour cette transaction. (Voir aussi Note 101.)
- (23) Nom de domaine du commerçant [RFC1034], comme www.commerçant.exemple. (Voir aussi Note 101.)
- (24) Nom de domaine [RFC1034] du processeur de passerelle de transaction qui accepte le paiement au nom du commerçant, comme www.processeur.exemple. (Voir aussi Note 101.)
- (25) Chaîne d'identification de transaction dont le format est spécifique du processeur.
- (26) URL [RFC3986] qui peut être invoqué pour enquêter sur la transaction. (Voir aussi Note 100.)
- (27) Montant de la transaction en format de devise ISO [ISO4217]. Ce sont deux nombres entiers avec un point entre eux mais sans autre marque de devise (comme un signe "\$" (dollar)).
- (28) Code de devise ISO de trois lettres [ISO4217]. Par exemple, dollars US est USD.
- (29) Date ISO de transaction.
- (30) Type de la transaction, si il est connu. Actuellement une des deux valeurs suivantes : débit, crédit
- (31) Signature numérique, codée en base64 [RFC2045]. (Voir aussi Note 101.)
- (32) Les champs ReceiptTo sont utilisés quand l'entité BillTo, la localisation, ou l'adresse et l'entité ReceiptTo, la localisation, ou l'adresse sont différentes. Par exemple, quand on utilise certaines formes de cartes d'achat d'entreprise ou de cartes d'achat d'agent, le détenteur de carte individuel va être dans les champs ReceiptTo, et l'entreprise ou autre titulaire sera dans les champs BillTo.
- (33) Étiquette de langage de l'IETF, comme défini dans la [RFC3066].
- (34) Préférences de l'utilisateur, comme spécifiées par le commerçant. (Voir aussi Note 102.)
- (35) Localisateur universel de ressource [RFC3986] pour accéder au logiciel de "portefeuille du consommateur. (Voir aussi

Note 100)

- (36) Une seule lettre majuscule: M=masculin, F=féminin, U=Inconnu [ISO5218].
- (37) Numéro immuable d'identification ou de série d'appareil. (Voir aussi Note 102.)
- (38) Nom de marque d'appareil compréhensible par l'utilisateur. (Voir aussi Note 102)
- (39) Champ [ISO8583] "identification du terminal d'acceptation de carte".
- (40) Champ [ISO8583] "code de traitement".
- (41) Champ [ISO8583] "code d'identification de l'institution acquéreuse".
- (42) Champ [ISO8583] "code d'identification de l'institution transmetteuse".
- (43) Champ [ISO8583] "examen de la trace système".
- (44) Champ [ISO8583] "date effective".
- (45) Champ [ISO8583] "code de date du point de vente".
- (46) Champ [ISO8583] "code d'approbation".
- (47) Champ [ISO8583] "code d'action".
- (48) Champ [ISO8583] "date d'établissement".
- (49) Champ [ISO8583] "date de capture".
- (50) Champ [ISO8583] "données de trace 1".
- (51) [Champ [ISO8583] "données de trace 2".
- (52) nom de marque de carte de fidélité reconnaissable par l'utilisateur. Les valeurs de ce champ ne sont pas contrôlées, et il n'y a pas de registre IANA ou autre pour elles. (Voir aussi Note 102.)
- (53) Numéro de producteur de carte requis par les acquéreurs du Royaume-Uni "Switch and Solo".
- (54) Les noms de champs "first_name" et "last_name" ont été conservés pour la compatibilité avec les versions antérieures de ECML. Cependant, "last_name" devrait être compris comme se référant au nom de famille ou aux noms hérités, tandis que "first_name" est le premier prénom ou nom non hérité et "middle_name" est le prénom suivant ou nom non hérité suivant, si il y en a.
- (55) Le localisateur universel de ressource [RFC3986] pour accéder au certificat X.509v3 de l'utilisateur, codé en DER binaire. (Voir aussi Note 100.)

Meta Notes (référéncées par d'autres notes)

- (100) ECML, une convention de base de dénomination et de structuration de champs qui n'impose aucune exigence particulière à ces URL. Il est prévu que la plupart des applications qui utilisent ECML imposeront de telles limitations et effectueront des vérifications pour s'assurer que les URL fournis se conforment à ces limitations avant de tenter de les invoquer.
- (101) C'est un champ qui, lorsque présenté dans une page de la Toile, est généralement caché.
- (102) Chaîne de caractères ASCII [ASCII] sans espaces en tête ou en queue.

2.2 Syntaxe XML exemplaire

Les paragraphes qui suivent donnent une DTD XML et un schéma XML qui exprime les champs ECML avec la structure hiérarchique et les dénominations ECML v2. En cas de conflit entre cette DTD et le schéma, le schéma devrait prévaloir.

Noter que la dénomination et la structure ECML v2 peuvent être utilisées dans des syntaxes non XML.

La syntaxe XML ECML v2 est délibérément libérale parce qu'on suppose que les applications spécifiques qui utilisent ECML imposeront leur propres contraintes supplémentaires.

Pour l'internationalisation de ECML, on utilisera les dispositions générales de codage de caractère de XML [XML] (qui rendent obligatoire la prise en charge de UTF-8 et UTF-16 et permettent la prise en charge d'autres jeux de caractères) et l'attribut `xml:lang`, qui peut être utilisé pour des informations spécifiques de langage.

2.2.1 DTD XML ECML v2

Ce qui suit est une DTD XML pour ECML v2.

```
<!--Langage de modélisation du commerce électronique v2 -->
```

```
<!ELEMENT Ecom ( #PCDATA | ShipTo | BillTo | ReceiptTo | Payment | Loyalty | User | Merchant | Transaction | TransactionComplete )* >
```

```
<!ATTLIST Ecom
```

```
  id ID #IMPLIED
  ConsumerOrderID CDATA #IMPLIED
  Merchant CDATA #IMPLIED
  Mode (Query|Assert) #IMPLIED
  Processor CDATA #IMPLIED
  SchemaVersion (urn:ietf:params:ecml:v2.0) #IMPLIED
  WalletID CDATA #IMPLIED
  WalletLocation CDATA #IMPLIED >
```

```
<!ELEMENT ShipTo ( #PCDATA | Postal | Telecom | Online )* >
```

```
<!ATTLIST ShipTo
```

```
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED >
```

```
<!ELEMENT BillTo ( #PCDATA | Postal | Telecom | Online )* >
```

```
<!ATTLIST BillTo
```

```
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED >
```

```
<!ELEMENT ReceiptTo ( #PCDATA | Postal | Telecom | Online )* >
```

```
<!ATTLIST ReceiptTo
```

```
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED >
```

```
<!ELEMENT Postal ( #PCDATA | Name | Company | Street | City | StateProv )* >
```

```
<!ATTLIST Postal
```

```
  id ID #IMPLIED
  PostalCode NMTOKEN #IMPLIED
  Mode (Query|Assert) #IMPLIED
  CountryCode NMTOKEN #IMPLIED >
```

```
<!ELEMENT Name EMPTY >
```

```
<!ATTLIST Name
```

```
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED
  Prefix NMTOKEN #IMPLIED
  First NMTOKEN #IMPLIED
  Middle NMTOKEN #IMPLIED
  Last NMTOKEN #IMPLIED
  Suffix NMTOKEN #IMPLIED >
```

```
<!ELEMENT Street EMPTY >
```

```
<!ATTLIST Street
```



```

id    ID    #IMPLIED
Mode  (Query|Assert) #IMPLIED
Line1 CDATA  #REQUIRED
Line2 CDATA  #IMPLIED
Line3 CDATA  #IMPLIED >

<!ELEMENT Company (#PCDATA) >
<!ATTLIST Company
  Mode  (Query|Assert) #IMPLIED >

<!ELEMENT City (#PCDATA) >
<!ATTLIST City
  Mode  (Query|Assert) #IMPLIED >

<!ELEMENT StateProv (#PCDATA) >
<!ATTLIST StateProv
  Mode  (Query|Assert) #IMPLIED >

<!ELEMENT Telecom ( #PCDATA | Phone )* >
<!ATTLIST Telecom
  Mode  (Query|Assert) #IMPLIED >

<!ELEMENT Phone EMPTY >
<!ATTLIST Phone
  id    ID    #IMPLIED
  Mode  (Query|Assert) #IMPLIED
  Number CDATA  #REQUIRED >

<!ELEMENT Online ( #PCDATA | Email )* >
<!ATTLIST Online
  Mode  (Query|Assert) #IMPLIED >

<!ELEMENT Email EMPTY >
<!ATTLIST Email
  id    ID    #IMPLIED
  Mode  (Query|Assert) #IMPLIED
  Address CDATA  #REQUIRED >

<!ELEMENT Payment (Card) >
<!ATTLIST Payment

  Mode  (Query|Assert) #IMPLIED >

<!ELEMENT Card (ExpDate, ValidDate?) >
<!ATTLIST Card
  id    ID    #IMPLIED
  Mode  (Query|Assert) #IMPLIED
  Name  CDATA  #IMPLIED
  Type  NMTOKEN #IMPLIED
  Number NMTOKEN #REQUIRED
  Protocols NMTOKENS #IMPLIED
  Verification NMTOKEN #IMPLIED
  Issuer NMTOKEN #IMPLIED >

<!ELEMENT Loyalty (ExpDate?, ValidDate?) >
<!ATTLIST Loyalty
  id    ID    #IMPLIED
  Mode  (Query|Assert) #IMPLIED
  Name  CDATA  #IMPLIED
  Type  NMTOKEN #IMPLIED
  Number NMTOKEN #REQUIRED
  Verification NMTOKEN #IMPLIED >

```

```

<!ELEMENT ExpDate EMPTY >
<!ATTLIST ExpDate
  id      ID      #IMPLIED
  Mode    (Query|Assert) #IMPLIED
  Day     NMTOKEN #IMPLIED
  Month   NMTOKEN #REQUIRED
  Year     NMTOKEN #REQUIRED >

<!ELEMENT ValidDate EMPTY >
<!ATTLIST ValidDate
  id      ID      #IMPLIED
  Mode    (Query|Assert) #IMPLIED
  Day     NMTOKEN #IMPLIED
  Month   NMTOKEN #IMPLIED
  Year     NMTOKEN #REQUIRED >

<!ELEMENT User ( #PCDATA | UserID | Password )* >
<!ATTLIST User
  id      ID      #IMPLIED
  Mode    (Query|Assert) #IMPLIED
  CertificateURL CDATA #IMPLIED
  DataCountry NMTOKEN #IMPLIED
  DataLanguage CDATA #IMPLIED >

<!ELEMENT UserID (#PCDATA) >
<!ATTLIST UserID
  Mode    (Query|Assert) #IMPLIED >

<!ELEMENT Password (#PCDATA) >
<!ATTLIST Password
  Mode    (Query|Assert) #IMPLIED >

<!ELEMENT Merchant (Terminal) >
<!ATTLIST Merchant
  Mode    (Query|Assert) #IMPLIED
  id      ID      #IMPLIED >

<!ELEMENT Terminal EMPTY >
<!ATTLIST Terminal
  Id      ID      #IMPLIED
  Mode    (Query|Assert) #IMPLIED
  Data    CDATA #IMPLIED >

<!ELEMENT Transaction ( #PCDATA | Id | Code | Date | Data | Inquiry | Signature )* >
<!ATTLIST Transaction
  Amount  CDATA #IMPLIED
  Currency NMTOKEN #IMPLIED
  Mode    (Query|Assert) #IMPLIED
  Type    NMTOKEN #IMPLIED >

<!ELEMENT Id EMPTY >
<!ATTLIST Id
  Id      ID      #IMPLIED
  Mode    (Query|Assert) #IMPLIED
  CID     NMTOKEN #IMPLIED
  Reference NMTOKEN #IMPLIED
  Acquire NMTOKEN #IMPLIED
  Forward NMTOKEN #IMPLIED >

<!ELEMENT Code EMPTY >
<!ATTLIST Code
  Mode    (Query|Assert) #IMPLIED

```

```

Processing CDATA #IMPLIED
Approval NMTOKEN #IMPLIED
Retrieval NMTOKEN #IMPLIED
Action NMTOKEN #IMPLIED
Reason NMTOKEN #IMPLIED
POS NMTOKEN #IMPLIED >

```

```
<!ELEMENT Date (Effective?, Settle?, Capture?) >
```

```
<!ATTLIST Date
  Mode (Query|Assert) #IMPLIED
  id ID #IMPLIED >
```

```
<!ELEMENT Effective EMPTY >
```

```
<!ATTLIST Effective
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED
  Day NMTOKEN #REQUIRED
  Month NMTOKEN #REQUIRED
  Year NMTOKEN #REQUIRED >
```

```
<!ELEMENT Settle EMPTY >
```

```
<!ATTLIST Settle
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED
  Day NMTOKEN #REQUIRED
  Month NMTOKEN #REQUIRED
  Year NMTOKEN #REQUIRED >
```

```
<!ELEMENT Capture EMPTY >
```

```
<!ATTLIST Capture
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED
  Day NMTOKEN #REQUIRED
  Month NMTOKEN #REQUIRED
  Year NMTOKEN #REQUIRED >
```

```
<!ELEMENT Data (#PCDATA | Trace | PrivateUse | Response | AAV | Track1 | Track2)* >
```

```
<!ATTLIST Data
  Mode (Query|Assert) #IMPLIED >
```

```
<!ELEMENT Trace (#PCDATA) >
```

```
<!ATTLIST Trade
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED >
```

```
<!ELEMENT PrivateUse (#PCDATA) >
```

```
<!ATTLIST PrivateUse
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED >
```

```
<!ELEMENT Response (#PCDATA) >
```

```
<!ATTLIST Response
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED >
```

```
<!ELEMENT AAV (#PCDATA) >
```

```
<!ATTLIST AAV
  id ID #IMPLIED
  Mode (Query|Assert) #IMPLIED >
```

```
<!ELEMENT Track1 (#PCDATA) >
```

```
<!ATTLIST Track1
  id ID #IMPLIED
```

```

    Mode    (Query|Assert) #IMPLIED >

<!ELEMENT Track2 (#PCDATA) >
<!ATTLIST Track2
    id      ID      #IMPLIED
    Mode    (Query|Assert) #IMPLIED >

<!ELEMENT Inquiry (#PCDATA) >
<!ATTLIST Inquiry
    id      ID      #IMPLIED
    Mode    (Query|Assert) #IMPLIED >

<!ELEMENT Signature (#PCDATA) >
<!ATTLIST Signature
    id      ID      #IMPLIED
    Mode    (Query|Assert) #IMPLIED >

<!ELEMENT TransactionComplete EMPTY >

```

2.2.2 Schéma XML ECML v2

Voici un schéma XML pour ECML v2.

```

<?xml version="1.0" encoding="utf-8"?>
<!-- Electronic Commerce Modeling Language v2 -->

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified">

    <xs:attribute name="Mode">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="Query"/>
                <xs:enumeration value="Assert"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>

    <xs:attribute name="id" type="xs:ID"/>
    <xs:complexType name="EcomSimpleText">
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute ref="Mode" use="optional"/>
                <xs:attribute ref="id" use="optional"/>
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>

    <xs:element name="Ecom">
        <xs:complexType mixed="true">
            <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element ref="ShipTo"/>
                <xs:element ref="BillTo"/>
                <xs:element ref="ReceiptTo"/>
                <xs:element ref="Payment"/>
                <xs:element ref="Loyalty"/>
                <xs:element ref="User"/>
                <xs:element ref="Merchant"/>
                <xs:element ref="Transaction"/>
                <xs:element ref="TransactionComplete"/>
            </xs:choice>
            <xs:attribute ref="Mode" use="optional"/>
        </xs:complexType>

```

```

<xs:attribute ref="id" use="optional"/>
<xs:attribute name="ConsumerOrderID" use="optional"/>
<xs:attribute name="Merchant" use="optional"/>
<xs:attribute name="Processor" use="optional"/>
<xs:attribute name="SchemaVersion" type="xs:string"
  fixed="urn:ietf:params:ecml:v2.0"/>
<xs:attribute name="WalletID" use="optional"/>
<xs:attribute name="WalletLocation" type="xs:anyURI"
  use="optional"/>
</xs:complexType>
</xs:element>
<xs:element name="ShipTo">
  <xs:complexType mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="Postal"/>
      <xs:element ref="Telecom"/>
      <xs:element ref="Online"/>
    </xs:choice>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="BillTo">
  <xs:complexType mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="Postal"/>
      <xs:element ref="Telecom"/>
      <xs:element ref="Online"/>
    </xs:choice>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="ReceiptTo">
  <xs:complexType mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="Postal"/>
      <xs:element ref="Telecom"/>
      <xs:element ref="Online"/>
    </xs:choice>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Postal">
  <xs:complexType mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="Name"/>
      <xs:element ref="Company"/>
      <xs:element ref="Street"/>
      <xs:element ref="City"/>
      <xs:element ref="StateProv"/>
    </xs:choice>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="PostalCode" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="CountryCode" type="xs:NMTOKEN"
      use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Telecom">
  <xs:complexType mixed="true">

```

```

<xs:sequence maxOccurs="unbounded">
  <xs:element name="Phone">
    <xs:complexType>
      <xs:attribute ref="Mode" use="optional"/>
      <xs:attribute ref="id" use="optional"/>
      <xs:attribute name="Number"/>
    </xs:complexType>
  </xs:element>
</xs:sequence>
<xs:attribute ref="Mode" use="optional"/>
</xs:complexType>
</xs:element>
<xs:element name="Online">
  <xs:complexType mixed="true">
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="Email">
        <xs:complexType>
          <xs:attribute ref="Mode" use="optional"/>
          <xs:attribute ref="id" use="optional"/>
          <xs:attribute name="Address"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute ref="Mode" use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Payment">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Card">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="ExpDate"/>
            <xs:element ref="ValidDate" minOccurs="0"/>
          </xs:sequence>
          <xs:attribute ref="Mode" use="optional"/>
          <xs:attribute ref="id" use="optional"/>
          <xs:attribute name="Name" use="optional"/>
          <xs:attribute name="Type" type="xs:NMTOKEN"
            use="optional"/>
          <xs:attribute name="Number" type="xs:decimal"/>
          <xs:attribute name="Protocols" type="xs:NMTOKENS"
            use="optional"/>
          <xs:attribute name="Verification"
            type="xs:NMTOKEN" use="optional"/>
          <xs:attribute name="Issuer" type="xs:NMTOKEN"
            use="optional"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute ref="Mode" use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Loyalty">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ExpDate"/>
      <xs:element ref="ValidDate" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="Name" use="optional"/>
    <xs:attribute name="Type" type="xs:NMTOKEN"
  </xs:complexType>

```

```

    use="optional"/>
    <xs:attribute name="Number" type="xs:NMTOKEN"/>
    <xs:attribute name="Verification" type="xs:NMTOKEN"
      use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="ExpDate">
  <xs:complexType>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="Day" type="xs:positiveInteger"/>
    <xs:attribute name="Month" type="xs:positiveInteger"/>
    <xs:attribute name="Year" type="xs:positiveInteger"/>
  </xs:complexType>
</xs:element>
<xs:element name="ValidDate">
  <xs:complexType>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="Day" type="xs:positiveInteger"/>
    <xs:attribute name="Month" type="xs:positiveInteger"/>
    <xs:attribute name="Year" type="xs:positiveInteger"/>
  </xs:complexType>
</xs:element>
<xs:element name="User">
  <xs:complexType mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="UserID"/>
      <xs:element ref="Password"/>
    </xs:choice>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="CertificateURL" type="xs:anyURI"
      use="optional"/>
    <xs:attribute name="DataCountry" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="DataLanguage" type="xs:language"
      use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Transaction">
  <xs:complexType mixed="true">
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="Id"/>
      <xs:element ref="Code"/>
      <xs:element ref="Date"/>
      <xs:element ref="Data"/>
      <xs:element ref="Inquiry"/>
      <xs:element ref="Signature"/>
    </xs:choice>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute name="Currency" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="Type" type="xs:NMTOKEN"
      use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Date">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="Effective" minOccurs="0"/>
      <xs:element ref="Settle" minOccurs="0"/>
      <xs:element ref="Capture" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

</xs:sequence>
<xs:attribute ref="Mode" use="optional"/>
<xs:attribute ref="id" use="optional"/>
</xs:complexType>
</xs:element>
<xs:element name="Data">
<xs:complexType mixed="true">
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:element ref="Trace"/>
<xs:element ref="PrivateUse"/>
<xs:element ref="Response"/>
<xs:element ref="AAV"/>
<xs:element ref="Track1"/>
<xs:element ref="Track2"/>
</xs:choice>
<xs:attribute ref="Mode" use="optional"/>
</xs:complexType>
</xs:element>
<xs:element name="Merchant">
<xs:complexType>
<xs:sequence>
<xs:element name="Terminal">
<xs:complexType>
<xs:attribute ref="Mode" use="optional"/>
<xs:attribute ref="id" use="optional"/>
<xs:attribute name="Data" use="optional"/>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute ref="Mode" use="optional"/>
<xs:attribute ref="id" use="optional"/>
</xs:complexType>
</xs:element>
<xs:element name="AAV" type="EcomSimpleText"/>
<xs:element name="Capture">
<xs:complexType>
<xs:attribute ref="Mode" use="optional"/>
<xs:attribute ref="id" use="optional"/>
<xs:attribute name="Day" type="xs:NMTOKEN"/>
<xs:attribute name="Month" type="xs:NMTOKEN"/>
<xs:attribute name="Year" type="xs:NMTOKEN"/>
</xs:complexType>
</xs:element>
<xs:element name="City" type="EcomSimpleText"/>
<xs:element name="Code">
<xs:complexType>
<xs:attribute ref="Mode" use="optional"/>
<xs:attribute name="Processing" use="optional"/>
<xs:attribute name="Approval" type="xs:NMTOKEN"
use="optional"/>
<xs:attribute name="Retrieval" type="xs:NMTOKEN"
use="optional"/>
<xs:attribute name="Action" type="xs:NMTOKEN"
use="optional"/>
<xs:attribute name="Reason" type="xs:NMTOKEN"
use="optional"/>
<xs:attribute name="POS" type="xs:NMTOKEN"
use="optional"/>
</xs:complexType>
</xs:element>
<xs:element name="Company" type="EcomSimpleText"/>
<xs:element name="Effective">
<xs:complexType>

```



```

    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="Day" type="xs:NMTOKEN"/>
    <xs:attribute name="Month" type="xs:NMTOKEN"/>
    <xs:attribute name="Year" type="xs:NMTOKEN"/>
  </xs:complexType>
</xs:element>
<xs:element name="Id">
  <xs:complexType>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="CID" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="Reference" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="Acquire" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="Forward" type="xs:NMTOKEN"
      use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Inquiry">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="xs:anyURI">
        <xs:attribute ref="Mode" use="optional"/>
        <xs:attribute ref="id" use="optional"/>
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element name="Name">
  <xs:complexType>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="Prefix" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="First" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="Middle" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="Last" type="xs:NMTOKEN"
      use="optional"/>
    <xs:attribute name="Suffix" type="xs:NMTOKEN"
      use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Password" type="EcomSimpleText"/>
<xs:element name="PrivateUse" type="EcomSimpleText"/>
<xs:element name="Response" type="EcomSimpleText"/>
<xs:element name="Settle">
  <xs:complexType>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="Day" type="xs:NMTOKEN"/>
    <xs:attribute name="Month" type="xs:NMTOKEN"/>
    <xs:attribute name="Year" type="xs:NMTOKEN"/>
  </xs:complexType>
</xs:element>
<xs:element name="Signature">
  <xs:complexType>
    <xs:simpleContent>

```

```

    <xs:extension base="xs:string">
      <xs:attribute ref="Mode" use="optional"/>
      <xs:attribute ref="id" use="optional"/>
    </xs:extension>
  </xs:simpleContent>

</xs:complexType>
</xs:element>
<xs:element name="StateProv" type="EcomSimpleText"/>
<xs:element name="Street">
  <xs:complexType>
    <xs:attribute ref="Mode" use="optional"/>
    <xs:attribute ref="id" use="optional"/>
    <xs:attribute name="Line1"/>
    <xs:attribute name="Line2" use="optional"/>
    <xs:attribute name="Line3" use="optional"/>
  </xs:complexType>
</xs:element>
<xs:element name="Trace" type="EcomSimpleText"/>
<xs:element name="Track1" type="EcomSimpleText"/>
<xs:element name="Track2" type="EcomSimpleText"/>
<xs:element name="TransactionComplete">
  <xs:complexType/>
</xs:element>
<xs:element name="UserID" type="EcomSimpleText"/>

</xs:schema>

```

3. Notes d'utilisation pour ECML v2

Cette section fournit un guide d'utilisation général pour ECML v2.

3.1 Présentation des champs

ECML v2 désigne simplement les champs et spécifie leur contenu et leur organisation hiérarchique. Il n'impose pas de contrainte sur l'ordre ou la complétude de communication ou d'interrogation pour ces champs.

Certains peuvent souhaiter fournir ou demander plus d'informations, et certains moins en omettant des champs. Certains peuvent demander les informations qu'ils veulent dans une interaction ou page de la Toile, et d'autres peuvent demander des parties des informations à des moments différents dans plusieurs interactions ou différentes pages de la Toile. Par exemple, il est courant de demander l'information "expédier à" plus tôt afin que les coûts d'expédition puissent être calculés avant les informations de méthode de paiement. Certains peuvent demander que toutes les informations nécessaires soient fournies tandis que d'autres peuvent rendre la plupart des informations facultatives. D'autres variations sont probables.

Chaque élément peut être marqué comme interrogation ou assertion en incluant, quand la syntaxe XML est utilisée, l'attribut Mode facultatif avec, respectivement, la valeur "Query" ou "Assert". L'attribut Mode affecte tous les éléments descendants jusqu'à être outrepassé par un élément de niveau inférieur avec un attribut Mode. Donc il est aisé d'indiquer que tous les éléments d'une structure ECML v2 sont présents comme interrogations ou assertions.

Les éléments d'interrogation peuvent avoir un contenu de données. Un tel contenu DEVRAIT être interprété comme une valeur par défaut à retourner si aucune meilleure valeur n'est connue. Il n'y a aucun moyen avec la version 2.0 de ECML d'indiquer quels champs d'interrogation une partie considère comme de réponse obligatoire. De ce point de vue, tous les champs d'interrogations sont facultatifs à remplir. Cependant, on peut mettre une erreur ou re-présenter une demande pour information si des champs exigés ne sont pas remplis, tout comme on peut le faire si un champ est rempli d'une manière considérée comme erronée.

3.2 Méthodes et flux de réglage des champs

Diverses méthodes de communication sont possibles entre les parties par lesquelles chacun peut indiquer quels champs il veut que remplisse l'autre. La méthode probablement la plus aisée pour les logiciels actuellement déployés est par des

champs en forme [HTML]. D'autres possibilités incluent d'utiliser un échange [XML], la transaction "IOTP Authenticate" [RFC2801], ou des protocoles propriétaires.

Afin que le logiciel de navigation puisse dire quelle version il traite, il est EXIGÉ que le champ `Ecom_SchemaVersion` soit inclus dans toute transaction quand ECML est utilisé sur la Toile. `Ecom_SchemaVersion` DEVRAIT apparaître sur chaque page de la Toile qui a des champs Ecom. C'est généralement un champ caché dans les formulaires HTML.

Une action de l'utilisateur ou l'apparition du champ `Ecom_SchemaVersion` sont des exemples de déclencheurs qui peuvent être utilisés pour initier une facilité capable de fournir des informations en réponse à une interrogation fondée sur ECML ou d'utiliser des informations provenant d'assertions ECML. Comme certains logiciels de la Toile peuvent requérir une activation par l'utilisateur, il est RECOMMANDÉ qu'il y ait au moins un champ Ecom visible par l'utilisateur sur toute page de la Toile avec des champs Ecom présents quand ECML est utilisé via la Toile.

Dans certaines circonstances, les communications peuvent être très lentes, de sorte qu'il n'est pas toujours clair pour une fonction de traitement automatisé si elle a fini de recevoir des champs ECML sur une page de la Toile ou autre. Pour cette raison, il est RECOMMANDÉ que le champ `Ecom_SchemaVersion` soit le dernier champ Ecom d'une page de la Toile.

Le transfert ou les demandes d'informations peuvent s'étendre sur plusieurs interactions ou pages de la Toile. Sans autres dispositions, une facilité pourrait soit exiger un redémarrage sur chaque page, soit éventuellement violer ou paraître violer la confidentialité en continuant à fournir des données personnelles au delà de la fin d'une transaction avec un commerce particulier. Pour cette raison, le champ `Ecom_TransactionComplete`, qui est normalement caché quand il fait partie d'un formulaire HTML, est fourni. Il est RECOMMANDÉ qu'il apparaisse sur la dernière interaction ou page de la Toile impliquée dans une transaction, juste avant un champ `Ecom_SchemaVersion`, de façon à ce que la logique d'interaction automatisée reçoive une indication sur quand arrêter si elle choisit de vérifier qu'il y a ce champ.

4. Considérations sur la sécurité et la confidentialité

Les informations demandées par nombre de ces champs sont sensibles. Elles devraient être protégées contre les modifications non autorisées et rester confidentielles si elles sont mémorisées dans un site ou transmises sur un canal où elles pourraient autrement être observées. De plus, l'authenticité des informations sera un problème dans de nombreux systèmes.

Les mécanismes pour une telle protection et authentification ne sont pas spécifiés ici mais peuvent, selon les circonstances, inclure des protocoles de sécurité d'objet (comme XMLDSIG [RFC3275], le chiffrement XML [XMLENC], ou CMS [RFC3852]), ou la sécurité du canal (comme TLS [RFC2246] ou IPsec [RFC2411]). Les systèmes dans lesquels un ou des champs ECML sont mémorisés et transmis ensuite vont probablement trouver la sécurité d'objet des plus appropriées.

Quand des informations sont demandées à un usager, son contrôle sur la divulgation de ces informations est nécessaire pour protéger la vie privée de l'usager.

Le logiciel qui est installé sur des terminaux partagés ou publics devrait être configurable de façon à ce que la mémorisation de toutes informations sensibles ou d'identité individuelle soit complètement désactivée. C'est vital pour protéger la vie privée des libraires, étudiants, et consommateurs qui utilisent des terminaux publics, et des enfants qui peuvent, par exemple, utiliser une forme de terminal public sans réaliser que leurs informations sont conservées.

Quand des informations sensibles ou d'identification individuelle sont conservées, l'opérateur ou l'usager devrait avoir une option pour protéger les informations ; par exemple, avec un mot de passe sans lequel les informations seront indisponibles, même à quelqu'un qui a accès aux fichiers dans lesquels elles sont conservées.

Tout mécanisme de remplissage multi pages/écrans ou autre multi champs agrégés ou de fourniture de données DEVRAIT vérifier le champ `Ecom_TransactionComplete` et cesser le remplissage automatique quand il se rencontre jusqu'à ce que le remplissage soit bien autorisé.

On devrait se souvenir que les valeurs par défaut, cachées, et autres, transférées à un tiers peuvent être modifiées de façon malveillante avant d'être retournées.

5. Considérations relatives à l'IANA

Les paragraphes qui suivent traitent de :

1. l'enregistrement du schéma XML de ECML v2 contenu dans ce document,
2. une version d'URN pour les versions d'ECML,
3. les enregistrements subsidiaires de versions particulières d'ECML et l'enregistrement spécifique de la version 2.0,
4. trois registres IANA supplémentaires pour les éléments qui apparaissent dans trois champs de ECML v2.

5.1 Gabarit de schéma ECML v2

Le schéma ECML v2 donné au paragraphe 2.2.2 est enregistré comme suit :

URI : urn:ietf:params:xml:schema:ECMLv2

Contact du registrant : <iesg@ietf.org>

XML : schéma XML du paragraphe 2.2.2.

5.2 Gabarit d'URN ECML v2

Comme spécifié par le gabarit ci dessus provenant de la [RFC3553], urn:ietf:params:ecml est enregistré de façon permanente avec des sous enregistrements via publication de RFC.

Nom du registre : urn:ietf:params:ecml

Spécification : RFC 4112

Répertoire : RFC 4112

Valeur d'indice : les valeurs subordonnées à urn:ietf:params:ecml sont enregistrées par publication de RFC. Conformément à la [RFC3553], quand une telle valeur est enregistrée, elle ne peut jamais changer.

5.2.1 Sous enregistrement de v2.0

La valeur subordonnée "v2.0" est enregistrée de façon permanente afin que l'URN urn:ietf:params:ecml:v2.0 soit utilisé pour indiquer un ou des champs ECML qui se conforment à la présente spécification. Bien qu'on ne prévoit pas que des valeurs subordonnées plus profondes de cet URN auront besoin d'être enregistrées, si nécessaire, elles seront enregistrées par approbation de l'IESG.

5.3 Registres IANA

Trois champs décrits au paragraphe 2.1.2 exigent l'établissement de registres IANA comme décrit ci-dessous :

Ecom_Payment_Card_Type : Registre de désignations de types de cartes en alphanumérique ASCII [ASCII] insensibles à la casse de un à quatre caractères sans espaces blanches. Voir le paragraphe 2.1.2, Note 11, les 12 désignations initiales. Les désignations sont ajoutées sur la base de l'approbation d'expert. Il sera normalement demandé aux demandeurs d'enregistrement d'avoir déjà un numéro d'identification de producteur ISO (IIN, *Issuer Identification Number*) ou ensemble de IIN.

Ecom_Payment_Card_Protocol : Ce champ contient une liste séparée par des espaces des protocoles désignés par des jetons alphanumériques ASCII [ASCII] insensibles à la casse tirés de ce registre ou contient le jeton "none". Voir au paragraphe 2.1.2, note 17, les sept jetons initiaux enregistrés (incluant "none") de d'autres informations. Les jetons sont ajoutés au registre sur la base de l'approbation d'expert.

Ecom_Transaction_Type : Valeur alphabétique ASCII [ASCII] insensible à la casse qui indique le type de transaction. Voir le paragraphe 2.1.2, note 30, les deux valeurs initiales enregistrées. Les valeurs sont ajoutées sur la base de l'approbation d'expert.

6. Remerciements

Voici la liste, par ordre alphabétique de ceux qui ont contribué au texte ci-dessus : Ray Bellis, Steve Bellovin, Scott Hollenbeck, Russ Housley, Jon Parsons, Lauri Piikivi, David Shepherd, et James J. Peter.

Appendice A. Changements de la v1.1 à la v2

Réécriture substantielle du texte pour changer l'expression des champs de forme HTML en syntaxe XML.
Ajout des champs de commerçant -> processeur.

Ajout des champs Ecom_Wallet_Location et Ecom_User_Certificate_URL.
Ajout de l'attribut "Mode".
Ajout des champs ECom_Payment_Card_IssueNumber, Loyalty Card, Device ID, Valid From, et User Data.
Ajout d'un schéma XML.
Corrections mineures relatives aux numéros de téléphone.
Ajout de la section "Considérations relatives à l'IANA".
Mise à jour des références des RFC obsolètes.

Références normatives

- [ASCII] American National Standards Institute, "USA Standard Code for Information Interchange" X3.4, New York, 1968.
- [E.164] Recommandation UIT-T E.164/I.331, "Plan de numérotage des télécommunications publiques internationales". 05/1997.
- [ISO3166] Norme ISO 3166-1, "Codes pour la représentation des noms de pays et leurs subdivisions -- Partie 1 : codes de pays", 1997.
- [ISO4217] Norme ISO 4217, "Codes pour la représentation des monnaies et devises", 2001.
- [ISO5218] Norme ISO 5218, "Échanges d'informations -- représentation des sexes humains", 1977.
- [ISO7812] Norme ISO 7812-1, "Carte d'identification - identification des producteurs - Partie 1 : système de numérotation", 2000.
- [ISO8583] Norme ISO 8583-1, "Messages générés par les cartes de transactions financières - spécifications des messages d'échange - Partie 1 : messages, éléments et valeurs des codes", 2001.
- [RFC2045] N. Freed et N. Borenstein, "[Extensions de messagerie Internet](#) multi-objets (MIME) Partie 1 : Format des corps de message Internet", novembre 1996. (*D. S., MàJ par 2184, 2231, 5335.*)
- [RFC3066] H. Alvestrand, "Étiquettes pour l'identification des langues", BCP 47, janvier 2001. (*Obsolète, voir la RFC4646.*)
- [RFC3986] T. Berners-Lee, R. Fielding et L. Masinter, "[Identifiant de ressource uniforme](#) (URI) : Syntaxe générique", STD 66, janvier 2005.
- [XML] F. Yergeau, T. Bray, J. Paoli, C. Sperberg-McQueen, et E. Maler, "[Langage de balisage extensible](#) (XML) 1.0 (3ème édition)", février 2004, < <http://www.w3.org/TR/REC-xml> >.

Références pour information

- [eCheck] < <http://www.echeck.org> >
- [HTML] D. Raggett, "HTML 3.2 Reference Specification", janvier 1997, < <http://www.w3.org/TR/REC-html32.html> >.
- [P3P.BASE] Cranor, L., Langheinrich, M., Marchiori, M., Presler-Marshall, M., et J. Reagle, "The Platform for Privacy Preferences 1.0 (P3P1.0) Specification", décembre 2000, < <http://www.w3.org/TR/WD-P3P/> >.
- [P3P.ECOM] Coco, J., Klien, S., Schutzer, D., Yen, S., et A. Slater, "Using P3P for E-Commerce", novembre 1999, < <http://www.w3.org/TR/P3P-for-ecommerce> >.
- [RFC1034] P. Mockapetris, "Noms de domaines - [Concepts et facilités](#)", STD 13, novembre 1987. (*MàJ par RFC1101, 1183, 1348, 1876, 1982, 2065, 2181, 2308, 2535, 4033, 4034, 4035, 4343, 4035, 4592, 5936, 8020, 8482*)
- [RFC2246] T. Dierks et C. Allen, "[Protocole TLS version 1.0](#)", janvier 1999. (*P.S. ; MàJ par RFC7919*)
- [RFC2411] R. Thayer, N. Doraswamy, R. Glenn, "[Feuille de route pour la sécurité](#) sur IP", novembre 1998. (*Obs., voir RFC6071*)

- [RFC2706] D. Eastlake 3rd, T. Goldstein, "ECML v1 : Noms de champ pour le commerce électronique", octobre 1999. (*Obsolète, voir RFC3106*) (*Information*)
- [RFC2801] D. Burdett, "Protocole Internet du commerce ouvert - IOTP version 1.0", avril 2000. (*Information*)
- [RFC3106] D. Eastlake 3 et T. Goldstein, "ECML v1.1 : spécification des champs pour le commerce électronique", avril 2001. (*Info.*)
- [RFC3275] D. Eastlake 3rd, J. Reagle, D. Solo, "Syntaxe et traitement de [signature en langage de balisage extensible](#) (XML)", mars 2002. (*D.S.*)
- [RFC3553] M. Mealling et autres, "[Sous-espace de noms d'URN de l'IETF](#) pour les paramètres de protocole enregistrés", juin 2003. ([BCP0073](#))
- [RFC3852] R. Housley, "Syntaxe de message cryptographique (CMS)", juillet 2004. (*Obsolète, voir la RFC5652*)
- [SET] "Secure Electronic Transaction", < http://www.setco.org/set_specifications.html >.
- [XMLENC] D. Eastlake 3rd, et J. Reagle, "XML Encryption Syntax and Processing", décembre 2002, <<http://www.w3.org/TR/xmlenc-core/>>.

Adresse de l'auteur

Donald E. Eastlake 3rd
Motorola Laboratories
155 Beaver Street
Milford, MA 01757
USA

téléphone : 1-508-786-7554 (bureau)
1-508-634-2066 (domicile)
mél : Donald.Eastlake@motorola.com

Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (2005).

Le présent document est soumis aux droits, licences et restrictions contenus dans le BCP 78, et à www.rfc-editor.org, et sauf pour ce qui est mentionné ci-après, les auteurs conservent tous leurs droits.

Le présent document et les informations contenues sont fournies sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations ci encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

Propriété intellectuelle

L'IETF ne prend pas position sur la validité et la portée de tout droit de propriété intellectuelle ou autres droits qui pourraient être revendiqués au titre de la mise en œuvre ou l'utilisation de la technologie décrite dans le présent document ou sur la mesure dans laquelle toute licence sur de tels droits pourrait être ou n'être pas disponible ; pas plus qu'elle ne prétend avoir accompli aucun effort pour identifier de tels droits. Les informations sur les procédures de l'ISOC au sujet des droits dans les documents de l'ISOC figurent dans les BCP 78 et BCP 79.

Des copies des dépôts d'IPR faites au secrétariat de l'IETF et toutes assurances de disponibilité de licences, ou le résultat de tentatives faites pour obtenir une licence ou permission générale d'utilisation de tels droits de propriété par ceux qui mettent en œuvre ou utilisent la présente spécification peuvent être obtenues sur répertoire en ligne des IPR de l'IETF à <http://www.ietf.org/ipr>.

L'IETF invite toute partie intéressée à porter son attention sur tous copyrights, licences ou applications de licence, ou autres

droits de propriété qui pourraient couvrir les technologies qui peuvent être nécessaires pour mettre en œuvre la présente norme. Prière d'adresser les informations à l'IETF à ietf-ipr@ietf.org.

Remerciement

Le financement de la fonction d'édition des RFC est actuellement fourni par la Internet Society.