

Membre	Clause / Sous-clause / Annexe	Paragraphe / Figure / Table / Note	Type de commentaire	Commentaire (justification de changement)	Changement proposé par l'Etat Membre
FR	Partie 4, Section 3.17.4.1	Page 2522	te	OOXML ne supporte que deux bases de temps et de dates (la base 1900 et la base 1904). Aucune de ces deux bases ne représente correctement le calendrier grégorien (ISO 8601) accepté dans tout le monde. De plus, cette restriction de seulement deux bases de date est arbitraire et liée uniquement aux logiciels d'un seul vendeur. Il y a d'autres prérequis pour les bases de dates, tels que le support de dates historiques antérieures à 1900. La plupart des bases de données SQL, qui échangent fréquemment des données avec des tableurs, supportent un plus grand interval de dates.	Permettre un interval de bases de dates plus grand, ou permettre explicitement des valeurs série négatives pour exprimer les dates d'avant 1900.
FR	Partie 4, Section 3.17.4.1	Page 2522	te	OOXML ne peut pas représenter certaines dates importantes et génère de faux calculs. Les calculs mandatés pour la date 1900 dans la base 1900 est inacceptable. Un standard ISO ne devrait pas définir un calcul de date incorrect pour le calendrier grégorien. Agir ainsi ne fera que mener à de la confusion, une interopérabilité pauvre ainsi qu'une perpétuation des erreurs.	Si cette fonctionnalité est nécessaire pour les anciens documents Excel, alors introduire un tag spécifique au vendeur tel que "doWrongDateCalculationsLikeExcel" ou similaire. C'est l'approche qui est utilisé partout dans OOXML pour les anciennes fonctionnalités Word.
FR	Partie 4, Section 3.17.4.1	-	te	OOXML spécifie que pour l'année 1900, "pour les dates entre le premier Janvier et le 28 février, WEEKDAY (jour de la semaine) doit retourner une valeur égale au jour précédent le jour correct", et doit aussi assigner une "valeur spéciale" pour la date non existante du 29 Février 1900. C'est une erreur. Un bogue logiciel doit être corrigé, et non exporté en tant que standard ISO dans les programmes des compétiteurs.	Faire en sorte que le calendrier soit configurable, avec comme défaut le calendrier grégorien, permettant la spécification d'alternatives en procurant un code qui calcule l'année, le mois, le jour de la semaine et le jour du mois.
FR	Partie 3, Section 3.16.9, Partie 4, Section 3.17.4.1	Page 2522, lignes 15 et 16	te	Dans le système de date basé sur 1900, la limite inférieure est le 1er Janvier 1900, qui a une valeur série de 1. Il existe des gens encore en vie nés avant 1900. Les études historiques considèrent souvent des dates avant 1900.	Enlever cette limitation et permettre un support des dates plus large. Permettre des valeurs négatives dans la valeur.
FR	Partie 4, Section 3.17.7.341	-	te	Telle qu'elle est écrite, cette fonction définit un calcul faux pour le jour de la semaine de certaines dates de l'année 1900. Un standard ISO ne devrait pas recommander des valeurs fausses pour le calendrier grégorien qui est bien établi. Agir ainsi ne fera que mener à de la confusion, une interopérabilité pauvre ainsi qu'une perpétuation des erreurs.	Supprimer le texte qui définit le comportement qui introduit de faux calculs de date.
FR	Partie 4, Section 3.17.4.1 3.17.6.7	-	te	Avoir deux systèmes de date différents avec des bases de date différentes côte à côte dans le même standard de format de document n'a aucun sens. A la place, il est approprié de fixer une seule base de date. Les applications qui utilisent une base de date différente peuvent convertir depuis leur représentation préférée vers celle du standard, et vice-versa.	Supprimer toutes les références au "1904 base system" ("format avec base année 1904").
FR	Partie 4, Section 2.8.2.16	-	te	L'utilisation des masques de bit n'est pas la bonne façon de faire en XML. Cet élément de la spécification utilise un ensemble de masques de bit pour spécifier quelles pages de code une police donnée supporte. L'usage de masques de bit plutôt qu'un type dérivé d'un Schéma XML rends ce type de données impossible à utiliser avec les outils standards XML tels que XSLT qui n'ont pas des fonctions au niveau du bit. Un des avantages de XML est de ne plus à avoir à encoder des données de cette façon.	Réécrire cette partie pour exprimer la possibilité en utilisant des constructions XML plutôt que des masques de bits.

FR	Partie 4, Section 7.4.2.4	-	te	La présence de caractères non-XML, échappés ou non échappés dans un document OOXML, est contraire à l'interopérabilité d'XML et des outils basés sur XML. L'Internationalization Activity du W3C confirme cette interprétation en disant "Codes de contrôle doivent être remplacés par des marquages appropriés. A partir du moment où XML fournit une façon standard d'encoder des données structurées, représenter des codes de contrôle autre qu'avec des marquages défierait les avantages d'utiliser XML. Utiliser les codes de contrôle en HTML et XHTML n'est jamais approprié, puisque ces langages de marquage sont pour représenter du texte, pas des données."	Enlever le type bstr d'OOXML.
FR	Partie 1, introduction (page xii) and l'intièreté du document	-	te	Il n'est pas acceptable pour un standard international d'être conçu autour de la compatibilité avec les produits d'un certain éditeur. Cela est particulièrement inapproprié quand, dans le cas présent, la compatibilité avec les standards internationaux existants est négligée en faveur d'un but unique qui est d'avoir une compatibilité maximale avec les formats de fichiers d'un seul éditeur, et où le standard proposé ne fournit pas des opportunités égales pour être compatibles pour les concurrents actuels et futurs. A moins que cette défaillance de OOXML/DIS29500 soit résolue, l'acceptation de cette spécification en tant que standard national ou international serait une violation des lois françaises et internationales.	Modifier l'objectif d'"être entièrement compatible avec les investissements existants dans les documents Microsoft Office" afin d'atteindre le même haut niveau de compatibilité non seulement avec les formats de Microsoft, mais aussi avec la norme internationale ISO/IEC 26300 (OpenDocument). Réviser l'entièreté de la proposition de standard et la modifier en correspondance avec ce nouvel objectif.
FR	Partie 1, Section 15.2.14	-	te	Trou de sécurité. OOXML permet l'inclusion de données binaires arbitraires (blobs) d'une manière qui pourrait être abusée par des auteurs de documents un peu malicieux. Par exemple: Partie 1, Section 15.2.14 recommande que les paramètres d'impression soient enregistrés dans le format DEVMODE utilisé par les pilotes d'imprimante Windows. Cependant, si quelqu'un parvenait à changer ces données binaires DEVMODE, ces données seraient chargées dans le pilote de l'imprimante la prochaine fois que l'utilisateur imprimerait. Etant donné que le pilote de l'imprimante s'exécute à un plus haut niveau de privilèges que l'utilisateur, cela pourrait permettre à un hacker de prendre contrôle de la machine de l'utilisateur en construisant un document spécifique.	La procédure actuelle pourrait être une bonne approche pour garder l'interopérabilité avec les vieux outils, mais un standard ISO doit fournir une spécification claire pour des implémentations futures qui ne perpétuent pas un trou de sécurité.
FR	Partie 4, 3.17.4.2, 3.17.6.7	-	te	A l'ère d'Internet, il n'est pas approprié de représenter une donnée temporelle uniquement à l'aide d'une valeur numérique sans spécifier le fuseau horaire.	Spécifier que les dates et heures, lorsqu'elles sont enregistrées dans des fichiers OOXML, doivent être exprimées en terme de date et heure UTC. Ajouter un mécanisme pour sauvegarder dans l'information du fichier le fuseau horaire qui doit être utilisé lors de la représentation du temps dans un format lisible par un être humain.
FR	Partie 4, Section 2.15.3.26	-	te	Ceci est l'élément "footnoteLayoutLikeWW8", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.31	-	te	Ceci est l'élément "lineWrapLikeWord6", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.

FR	Partie 4, Section 2.15.3.32	-	te	Ceci est l'élément "mwSmallCaps", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.41	-	te	Ceci est l'élément "shapeLayoutLikeWW8", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.51	-	te	Ceci est l'élément "suppressTopSpacingWP", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.54	-	te	Ceci est l'élément "uiCompat97To2003" qui est défini par : "Supprimer les fonctionnalités d'interface qui ne sont pas compatible avec Word97-2003". Quelle est l'utilité de cela si j'utilise OOXML dans OpenOffice ou WordPerfect Office ? Si je veux supprimer une fonctionnalité d'interface non compatible avec OpenOffice 1.5 ? Ou WordPerfect 8 ? Ou n'importe quelle autre application? Où est la possibilité pour les autres applications de spécifier leur préférences ?	Définir cette fonctionnalité de manière neutre par rapport à l'application. Si c'est vraiment spécifique à Word, alors supprimer de OOXML et l'exprimer en tant qu'extension définie par l'application.
FR	Partie 4, Section 2.15.3.54	-	te	Ceci est l'élément "truncateFontHeightsLikeWP6", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.6	-	te	Ceci est l'élément "autoSpaceLikeWord95", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.63	-	te	Ceci est l'élément "useWord2002TableStyleRules", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.64	-	te	Ceci est l'élément "useWord97LineBreakRules", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.65	-	te	Ceci est l'élément "wpJustification", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 2.15.3.66	-	te	Ceci est l'élément "wpSpaceWidth", qui est défini par l'imitation du comportement d'une ancienne application. Le standard ne contient pas suffisamment de détails pour répliquer ce comportement.	Définir ce comportement.
FR	Partie 4, Section 6.4.3.1	-	te	Les valeurs permises de cette énumération (EMF, WMF, etc...) sont des formats spécifiques à Windows. Aucune permission ne semble avoir été possible pour une utilisation par d'autres systèmes d'exploitation. Par exemple, sous Linux les images sont typiquement copiées dans le presse-papier dans un format	Plusieurs options sont possibles ici, mais le souhait est de permettre l'interopérabilité à travers les différentes plateformes.

				ouvert standard comme PNG.	
FR	Partie 4, Section 7.1	-	te	C'est la spécification d'Office Open Math Markup Language, un vocabulaire XML spécialisé dans la description de la représentation d'équations mathématiques. Cela résout le même problème que MathML, un standard W3C établi depuis longue date et ayant une vie en cours au W3C. Etant donné que la fonction d'édition des équations mathématiques a été réécrite entièrement dans Word 2007, il n'y pas d'argument pour supporter le fait qu'un nouveau langage soit introduit juste pour des raisons de compatibilité avec des documents existants.	Il est recommandé que cette section soit enlevée d'OOXML et que les déposants d'OOXML travaillent en relation avec les activités MathML du W3C où MathML 3.0 est en train d'être développé, afin de produire un standard unique pour les équations qui peuvent être utilisées et référencées dans une version future d'OOXML.
FR	Partie 4, Section 7.4.2.4	-	te	Ceci définit un nouveau type XML de chaîne de caractères qui permet une inclusion via un mécanisme d'échappement de caractères Unicode qui sont autrement inacceptables dans les documents XML. Néanmoins, tout mécanisme d'échappement doit aussi spécifier un mécanisme pour "échapper à l'échappement". Comment doit-on alors représenter l'exemple littéral donné en 7.4.2.4 dans un bstr ?	Compléter la définition du mécanisme d'échappement.
FR	Partie 4, Section 7.1	-	te	OOXML n'est pas compatible avec le standard industriel pour afficher les équations mathématiques — MathML — utilisé par la communauté scientifique et, plus important encore, les publications comme Science, Nature, etc. Aucune interopérabilité avec MathML ne se trouve dans les spécifications d'OOXML.	Ajouter cette pertinente interopérabilité avec MathML.
FR	Partie 4, section 6.1.2.19	p. 4653 "equationxml"	te	Cette section décrit l'attribut "equationxml" des éléments de type "shape" comme suit : "utilisé pour réhydrater une équation en utilisant la syntaxe de Office Open XML Math". Cependant, le "format du contenu de cet attribut est défini par l'application", ce qui les rends impossibles à échanger entre applications. Si nous devons avoir un nouveau langage XML pour les mathématiques, et ignorer le format MathML existant, faisons au moins en sorte qu'il le soit selon une syntaxe XML propre (et non pas coincé dans un attribut), et sans extensions dépendantes de l'application.	Définir les équations d'une manière interopérable.
FR	Tout le document	-	te	La spécification MS-OOXML contient des éléments brevetés non conformes aux directives ISO/IEC (partie 1, section 2.14) . L'utilisation de brevets est compatible avec une p'un angle de 270 derocédure ISO, mais uniquement si les directives ISO/IEC (Partie 1, section 2.14) sont respectées. Ce n'est pas le cas ici: certaines fonctionnalités brevetées sont en-dehors de la spécification OOXML, certaines fonctionnalités optionnelles non couvertes par la spécification OOXML sont aussi brevetées, et enfin, l'énoncé sur les brevets dans la spécification OOXML est tellement vague qu'elle n'offre aucune sécurité légale aux entreprises qui implémenteraient la spécification OOXML.	Se conformer aux directives ISO/IEC Directives (partie 1, section 2.14) pour le problème des brevets. Le possesseur des brevets devrait clarifier ce qu'il entend par "partie détaillée" en faisant référence aux endroits de la spécification qui sont couvertes par le "Covenant not to sue".
FR	Partie 4, Section 2.3.2.8	page 178	te	Il est désirable d'avoir une interopérabilité renforcée entre ODF et OOXML. Toutefois, l'attribut "vert" dans OOXML ne permet au texte d'être tourné uniquement que de 270 degrés, alors que son équivalent ODF permet des angles de 90 ou 270 degrés.	Inclure la possibilité de tourner le texte de 90 degrés en plus des 270 degrés existants.

FR	Partie 4, Section 2.3.2.1	page 160	te	Il est désirable d'avoir une interopérabilité renforcée entre ODF et OOXML. Toutefois, dans OOXML le texte ne supporte que deux épaisseurs de police, gras et normal, alors que ODF permet l'étendue plus complète provenant de XSL-FO.	Inclure le support pour les épaisseurs additionnelles 100, 200, 300, 400, 500,600. 700, 800 et 900.
FR	Partie 4, Section 2.4.46	page 421	te	Il est désirable d'avoir une interopérabilité renforcée entre ODF et OOXML. Toutefois, OOXML ne permet pas de spécifier un en-tête avec plusieurs lignes qui se répète sur toutes les pages, alors que ODF le permet.	Inclure dans cette section la possibilité de spécifier que les premières N lignes de la table peuvent être sélectionnées comme en-tête.
FR	Partie 4, Section 7.4.2.5	-	te	Cela n'a pas de sens de spécifier les chaînes de caractères comme des chaînes de caractères se terminant par null comme dans le langage C pour ensuite réencoder celles-ci en base 64. Cela revient à éviter XML et causera une mauvaise interopérabilité de la balise avec des outils basés sur XML.	ECMA devrait repenser l'entière représentation des données du Presse-Papier. Il semblerait probable qu'elle ne fasse que reproduire directement les données internes arbitraires d'une seule application. Cette clause devrait être réécrite afin d'exprimer cette caractéristique d'une manière neutre en fonction de l'application ou de la plateforme.
FR	Partie 4, Section 3.17.4.1	-	te	OOXML spécifie qu'en 1900, "pour les dates entre le 1er janvier et le 28 février, WEEKDAY doit retourner une valeur pour le jour précédant le jour correct" et également assigner une valeur série pour le 29 février de l'année 1900 (qui n'existe pas). Ceci est une erreur. Les bogues dans les logiciels devraient être corrigés, et non exportés vers les programmes concurrents par le biais d'une standardisation ISO.	Il faut essayer de rendre le calendrier configurable, en utilisant le calendrier grégorien par défaut, mais en permettant de spécifier des alternatives en fournissant du code pour le calcul de l'année, du mois, du jour de la semaine et du jour de la semaine.
FR	Partie 4, Section 2.15.2.32	-	te	Cette fonctionnalité a été définie dans un esprit qui ignore complètement l'existence d'alternatives à Internet Explorer en ce qui concerne les navigateurs. Qu'en est-il de Safari, d'Opera, de Firefox? Aucun de ces derniers ne peut être défini comme navigateur cible. Cette section requiert que "tous les paramètres non-compatibles avec le navigateur cible doivent être désactivés". Mais alors que se passe-t-il si je souhaite que mon application produise une sortie compatible avec les standards? Pour résumer, oui à PNG, non à VML, oui à MathML et SVG? Je ne peux spécifier ça nulle part.	L'E.C.M.A. devrait repenser la clause optimizeForBrowser dans son intégralité, car il semble vraiment que celle-ci colle parfaitement aux choix arbitraires effectués par une application d'un vendeur unique. Cette clause devait être réécrite pour exprimer clairement l'esprit de cette fonctionnalité dans un esprit indépendant de toute plate-forme ou application.
FR	Partie 4, Section 2.15.1.28	-	te	Cela dit que la protection du document "doit être mise en vigueur". "Doit" indique clairement que ce comportement est requis. Mais quelques lignes plus tard, on lit que la protection du document "peut être ignorée".	Il serait important de clarifier cette évidente contradiction.
FR	Partie 4, Section 2.15.1.29	-	te	Cet élément permet de classer le document en tant qu'un de ces trois types: "letter", "email" ou "general". Bien que la description spécifie que cette fonctionnalité peut être utilisée par des applications hôtes pour faciliter la personnalisation des interfaces utilisateur et/ou des comportements de formatage automatique basés sur le 'type' d'un document WordprocessingML donné, la taxonomie fournie est si pauvre qu'elle en est quasiment inutile.	Il faut donc soit fournir une taxonomie raisonnable pour les documents, ou alors assouplir le type du document en xsd:string pour permettre aux autres applications de définir leur propres classifications.
FR	Partie 4, Section 2.2.1	page 28, ligne 1	te	La phrase 'or auto to allow a consumer to automatically determine the background color as appropriate.' (qui peut être traduit comme 'ou auto pour permettre à l'utilisateur de déterminer automatiquement la couleur de fond appropriée') ne définit pas le comportement approprié de l'utilisateur, alors même que la définition du type simple correspondant, que l'on trouve à la page 1737 de la partie 4, spécifie clairement que 'Cette valeur doit être utilisée	Il faut donc définir clairement et correctement les caractéristiques de cette valeur auto pour l'attribut color de la couleur de fond.

				pour spécifier une valeur de couleur déterminée automatiquement, dont le sens doit être interprété en se basant sur le contexte de l'élément XML parent'.	
FR	Partie 4, Section 5.1.12.37	-	te	Il est dit que la longueur est "d'exactement 10 caractères". Ceci n'est pas cohérent avec le fragment de schéma donné qui définit cette longueur comme étant de 10 octets.	Il faut clarifier cette définition. En particulier il est bon de noter que xsd:hexBinary mesure la longueur en octets et pas en nombre caractères.
FR	Partie 4, Section 3.2.29	-	te	Un algorithme de hachage est fourni pour sécuriser les mots de passe, très probablement basé sur un algorithme préalablement utilisé sous Excel. Cet algorithme, héritage du passé, est connu pour son manque de robustesse et pour avoir été cassé. On pourrait argumenter qu'aucun algorithme de hachage ne peut être réellement efficace dans le cadre d'OOXML, dans la mesure où il suffirait à un utilisateur de décompresser le document et d'éditer le fichier à la main pour enlever le hachage ou alors l'initialiser à une valeur connue. Toutefois, certains types d'applications tels que l'édition en ligne via Google Docs ou des applications similaires peuvent sécuriser l'accès au document par d'autres moyens. L'accès en édition à un document ne signifie pas nécessairement pouvoir avoir accès physiquement au fichier XML sous-jacent à ce même document. On a donc réellement besoin d'un algorithme de hachage sûr et interopérable, tel que SHA-256 pour protéger les documents.	Utiliser un algorithme de hachage répondant aux standards tel que ISO/IEC 10118-3:2004, comme algorithme de protection par défaut. D'autres standards spécifiques à certains pourraient également être envisagés, comme par exemple FIPS-180 de NIST, mais en plus des standards globaux. Les algorithmes de hachage plus anciens pourraient parfaitement être supportés par le mécanisme d'extension décrit.
FR	Page 1754	ligne 1	te	Spécification du langage: OOXML ne s'appuie pas sur les normes ISO 639 et 639-2, et utilise à la place son propre système ("Microsoft Locale ID page").	Utiliser les normes ISO 639 et 639-2.
FR	Partie 4, Section 2.18.51	-	te	L'utilisation des 255 codes de langues énumérés, en plus des codes ISO 639-1, n'apporte aucune information complémentaire, et ne fait qu'alourdir le travail des applications qui traiteraient un document OOXML.	Abandonner l'utilisation du ST_LangCode, qui est redondant.
FR	Partie 4, Section 2.18.52	-	te	Avoir deux systèmes de représentation des langues, l'un basé sur des normes ISO et l'autre à base de code hexadécimaux arbitraires pour quelques langues n'a pas de sens. Les applications qui utilisent ces valeurs arbitraires pour représentes quelques langues peuvent faire la conversion du système standard de représentation des langues vers leur représentation interne, et vice versa.	Supprimer toute référence à ST_Langcode.
FR	Partie 4, Section 7.4.2.5	-	te	Cet élément définit les valeurs à utiliser sur plateforme Windows ou Macintosh, mais pas sous Linux ou tout autre système d'exploitation.	Il y a plusieurs possibilités, le but étant de garantir une interopérabilité entre plateformes.
FR	Partie 4, Section 7.4.2.5	-	te	Même sur une seule plateforme, les informations sont insuffisantes pour obtenir l'interopérabilité. Par exemple, quelles sont les valeurs admises et leur signification pour une "valeur au format natif du presse-papier Windows"?	A spécifier pour permettre l'interopérabilité.
FR	Partie 4, Section 2.15.1.28	-	te	L'algorithme donné n'a pas été examiné par un comité de spécialistes. Pourquoi est-ce que OOXML n'utilise pas un standard cryptographique reconnu, comme SHA-256 ou Whirlpool? Comme cet algorithme n'a pas été examiné, on ne peut pas savoir s'il est sûr. On peut rétorquer que la plupart des applications bureautiques n'ont pas besoin d'un cryptage fort,	Utiliser un algorithme de cryptage reconnu comme SHA-256, Whirlpool, ou autre.

				mais la spécification d'une norme doit avoir une portée générale et prendre en compte toutes les situations. Il n'est pas clairement défini comment cette fonctionnalité pourrait accepter d'autres systèmes de cryptage.	
FR	Partie 4, 3.17.4.3	-	te	La "représentation combinée des dates et heures" est inutilisable en France et dans tout pays qui utilisent des heures d'été/d'hiver. Les jours de transition durent 23 ou 25 heures plutôt que les 24 habituelles.	Ce problème peut être évité facilement en utilisant systématiquement UTC pour les "combinaisons date/heure".
FR	Partie 4, Section 2.15.1.28	ligne 13	te	Cet algorithme ne précise pas l'encodage du mot de passe saisi. Probablement de l'Unicode, mais quel est le codage? UTF-16BE? UTF-16LE? UTF-16 avec BOM (Byte Ordering Mark, indicateur d'ordre des octets)? Les algorithmes décrits utilisent des manipulations au niveau des octets, ce qui dépend de l'architecture de la machine ("big endian" ou "little endian"). Il est donc nécessaire que le choix de l'ordre des octets soit défini explicitement.	Rendre explicite le choix de l'ordre des octets, à la fois pour le mot de passe saisi et pour les traitements, de façon à permettre l'interopérabilité entre plateformes. Il faut garder à l'esprit que la signature résultant de l'algorithme (le "hash") peut être calculée sur une machine d'une autre architecture que celle sur laquelle le mot de passe a été entré.
FR	Partie 4, Section 2.15.1.86	-	te	Cet élément utilise un masque de bits pour spécifier un filtre d'affichage des styles. L'utilisation des masques de bits plutôt que d'un ensemble de booléens rend cette information impossible à traiter avec les outils standards XML comme XSLT, qui ne disposent pas d'opérations bit à bit.	Réécrire cette sous-clause avec une construction XML plutôt qu'un masque de bits.
FR	Partie 4, Section 2.15.1.87	-	te	Cet élément utilise un masque de bits pour spécifier un filtre d'affichage des styles. L'utilisation des masques de bits plutôt que d'un ensemble de booléens rend cette information impossible à traiter avec les outils standards XML comme XSLT, qui ne disposent pas d'opérations bit à bit.	Réécrire cette sous-clause avec une construction XML plutôt qu'un masque de bits.
FR	Partie 4, Section 2.16.4.3	page 1501, ligne 0	te	La définition de BATHTEXT fait référence au 'format Thai donné', ce qui n'a pas de sens dans le contexte de cette définition. Quel 'format Thai donné'?	Clarifier la définition de 'BATHTEXT'.
FR	Partie 4, Section 2.16.5.33	-	te	Ce champ récupère simplement l'image contenue dans le document donné. Ne faut-il rien faire d'autre avec cette image? Par exemple, doit-elle être affichée?	Définir ce qui doit être fait avec l'image une fois qu'elle est récupérée.
FR	Partie 4, Section 2.16.5.33	-	te	Ceci ne définit pas comment une image est nommée. Est-ce par une URL? Par un chemin sur le système de fichier local? Par l'un ou l'autre? L'exemple utilise un chemin de fichier DOS, une construction qui n'est pas portable.	Définir comment sont nommées les images.
FR	Partie 4, Section 2.16.5.34	-	te	Ceci ne définit pas comment un document est nommé. Est-ce par une URL? Par un chemin sur le système de fichier local? Par l'un ou l'autre? L'exemple utilise un chemin de fichier DOS, une construction qui n'est pas portable.	Définir comment sont nommés les documents.
FR	Partie 4, Section 2.16.5.34	-	te	Cette sous-clause définit un champ INCLUDETEXT qui "Insère tout ou partie des textes et graphiques contenus dans le document nommé". Toutefois, aucune mention n'est faite sur les formats autorisés pour les textes récupérés.	Au moins quelques formats interopérables devraient être mentionnés.
FR	Partie 4, Section 2.16.5.34	-	te	L'indicateur \t applique une transformation XSLT nommée à un fichier XML d'entrée et insère le résultat de la transformation. Cependant, aucune référence correcte n'est donnée pour XSLT, donc on ne sait pas quelle version est autorisée ici.	Fournir une référence correcte, externe et normative pour les versions de XSLT et Xpath autorisées ici.

FR	Partie 4, Section 2.16.5.5	page 1512, lines 11-12	te	Selon le texte, le champ AUTONUM est obsolète. Une nouvelle norme ne devrait pas contenir de parties obsolètes.	Retirer toutes les références à AUTONUM dans le texte d'OOXML.
FR	Partie 4, Section 2.16.5.77	-	te	L'exemple illustrant la section USERINITIALS montre en fait l'usage de USERNAME.	Corriger l'exemple.
FR	Partie 4, Section 2.18.106	-	te	Il est dit que la longueur est de "exactement 1 caractère". C'est incompatible avec le précédent langage et l'extrait de schéma donné qui la définit comme faisant un octet ou 2 caractères.	Clarifier la définition. En particulier, il faut remarquer que xsd:hexBinary mesure la longueur en octets, pas en caractères.
FR	Partie 4, Section 2.18.4	-	te	Aucun mécanisme d'extension de la liste des bordures décoratives n'est fourni. Comme les bordures définies sont fortement orientées à l'attention des usagers des pays de l'Ouest, il serait bon de fournir aux applications un moyen d'enrichir ces styles avec des graphismes adaptés à différentes cultures.	Fournir un mécanisme d'extension interopérable afin que l'auteur d'un document puisse définir ses propres bordures.
FR	Partie 4, Section 2.18.106	-	te	Il est dit que la longueur est de "exactement 3 caractères". C'est incompatible avec l'exemple donné qui la définit comme faisant 6 caractères.	Clarifier la définition. En particulier, il faut remarquer que xsd:hexBinary mesure la longueur en octets, pas en caractères.
FR	Partie 4, Section 2.18.52	-	te	Ce type est défini comme contenant "un code de langue de 2 chiffres hexadécimaux". Plus loin, il est dit : "Le contenu de ce type simple doit avoir une longueur d'exactement 2 caractères". Or deux caractères hexa peuvent représenter un nombre jusque 255, alors que les valeurs énumérées dans cette clause sont bien au-delà.	Réconcilier le description du type avec les valeurs énumérées.
FR	Partie 4, Section 2.18.57	-	te	La description de ce type indique qu'il contient quatre chiffres hexadécimaux, quatre octets hexadécimaux et exactement quatre caractères. Ces définitions ne sont pas compatibles. Un octet hexadécimal contient deux chiffres hexadécimaux.	Clarifier la définition. En particulier, il faut remarquer que xsd:hexBinary mesure la longueur en octets, pas en caractères.
FR	Partie 4, Section 2.18.66	-	te	Rien dans cette section n'est défini de manière normative à part quelques énumérations de valeurs. Aucune signification normative n'est donnée à ces valeurs. Un exemple à titre d'information est insuffisant sauf pour les cas les plus triviaux. Par exemple, où est défini le "Système de Numération Légal Coréen" ?	Donner des définitions explicites de ces styles, ou alors des références extérieures et normatives correctes.
FR	Partie 4, Section 2.18.66	"chicago"	te	Le format est défini en référence au "Manuel du Style Chicago", mais aucune édition ou référence de page n'est fournie.	Inclure soit la définition complète dans la norme, ou fournir une référence extérieure correcte.
FR	Partie 4, Section 2.18.66	"decimalEnclosedFullstop"	te	L'exemple donné ne montre pas de caractères inclus, et donc est contradictoire avec le texte normatif.	Réconcilier le texte et l'exemple.
FR	Partie 4, Section 2.18.66	"lowerLetter", etc.	te	Plusieurs systèmes de numération sont définis comme utilisant des lettres de l'alphabet, mais rien ne précise comment la numération se poursuit quand les lettres de l'alphabet sont épuisées.	Clarifier le texte pour couvrir explicitement ce cas.
FR	Partie 4, Section 2.18.66	"numberInDash", etc.	te	Le format requiert l'utilisation de "tirets" pour entourer le nombre, mais aucune indication quant au tiret Unicode est attendu : en-dash, em-dash, hyphen-minus, figure-dash, quotation-dash, etc.	Spécifier le type de tiret attendu.
FR	Partie 4, Section 2.18.72	-	te	Il est dit que la longueur est de "exactement 10 caractères". C'est incompatible avec l'exemple qui la définit comme faisant 20 caractères.	Clarifier la définition. En particulier, il faut remarquer que xsd:hexBinary mesure la longueur en octets, pas en caractères.
FR	Partie 4, Section 2.18.85	-	te	Les motifs de remplissage ne sont pas définis. Les illustrations données sont insuffisantes. Une application a besoin de savoir dans ces illustrations quels sont les comportements définis et ceux qui ne le sont pas. Par	Fournir les définitions normatives complètes de ces éléments graphiques.

				exemple, est-ce que le motif exact de "dithering" utilisé dans l'illustration est obligatoire?	
FR	Partie 4, Section 2.18.86	-	te	La description de ce type indique qu'il contient deux chiffres hexadécimaux, deux octets hexadécimaux et exactement deux caractères. Ces définitions ne sont pas compatibles. Un octet hexadécimal contient deux chiffres hexadécimaux.	Clarifier la définition. En particulier, il faut remarquer que xsd:hexBinary mesure la longueur en octets, pas en caractères.
FR	Partie 4, Section 2.2.1	page 29, ligne 0	te	Le mot "bordure" ("border") est employé plusieurs fois, alors qu'il n'a pas de signification dans ce contexte (ce passage est supposé décrire l'élément 'background' élément, et aucune "border" n'a été définie).	Préciser à quelle bordure le texte fait référence (si toutefois une notion bordure doit être introduite ici) ou bien réécrire le texte pour lui donner du sens.
FR	Partie 4, Section 2.2.1 background (Document Background)	page 27, ligne 8 et 21	te	Utilisation contradictoire de accent3 et accent5 – le texte dit une chose, l'exemple en dit une autre.	Retirer la contradiction
FR	Partie 4, Section 2.3.1.8	-	te	Cet élément utilise un masque de bits pour spécifier diverses propriétés de formatage conditionnel des paragraphes. L'utilisation des masques de bits plutôt que d'un ensemble de booléens rend cette information impossible à traiter avec les outils standards XML comme XSLT, qui ne disposent pas d'opérations bit à bit.	Réécrire cette sous-clause avec une construction XML plutôt qu'un masque de bits.
FR	Partie 4, Section 2.3.3.19	-	te	Ce passage dit que "Les propriétés de disposition de cet objet incorporé sont spécifiées en utilisant la syntaxe VML". Cependant, dans la Partie 1, Section 8.2.6 il est dit: "VML doit être considéré comme un format obsolète inclus dans Office Open XML uniquement pour des raisons de compatibilité ascendante, et les nouvelles applications qui ont besoin d'un format de fichier pour les dessins sont fortement encouragées à utiliser DrawingML". Il paraît évident qu'un nouveau document comprenant un objet OLE ne doit pas utiliser VML. Sinon, tous les utilisateurs de OOXML auront besoin du support de VML, même si aucun ancien document n'est utilisé.	Définir des propriétés de présentation d'objets embarqués utilisant DrawingML plutôt que VML.
FR	Partie 4, Section 2.4.51	-	te	Cet élément utilise un masque de bits pour spécifier diverses propriétés de formatage des tableaux. L'utilisation des masques de bits plutôt que d'un ensemble de booléens rend cette information impossible à traiter avec les outils standards XML comme XSLT, qui ne disposent pas d'opérations bit à bit.	Réécrire cette sous-clause avec une construction XML plutôt qu'un masque de bits.
FR	Partie 4, Section 2.4.52	-	te	Cet élément utilise un masque de bits pour spécifier diverses exceptions de formatage des tableaux. L'utilisation des masques de bits plutôt que d'un ensemble de booléens rend cette information impossible à traiter avec les outils standards XML comme XSLT, qui ne disposent pas d'opérations bit à bit.	Réécrire cette sous-clause avec une construction XML plutôt qu'un masque de bits.
FR	Partie 4, Section 2.4.7	-	te	Cet élément utilise un masque de bits pour spécifier diverses propriétés de formatage des cellules de tableaux. L'utilisation des masques de bits plutôt que d'un ensemble de booléens rend cette information impossible à traiter avec les outils standards XML comme XSLT, qui ne disposent pas d'opérations bit à bit.	Réécrire cette sous-clause avec une construction XML plutôt qu'un masque de bits.
FR	Partie 4, Section 2.4.8	-	te	Cet élément utilise un masque de bits pour spécifier diverses propriétés de formatage des lignes de tableaux. L'utilisation des masques de bits plutôt que d'un ensemble de booléens rend cette information impossible à traiter avec les outils standards XML comme XSLT, qui ne disposent pas d'opérations bit à bit.	Réécrire cette sous-clause avec une construction XML plutôt qu'un masque de bits.

				bit.	
FR	Partie 4, Section 3.18.86	-	te	Il est dit que la longueur est de "exactement 4 caractères". C'est incompatible avec l'extrait de schéma qui la définit comme faisant 4 octets ou 8 caractères.	Clarifier la définition. En particulier, il faut remarquer que xsd:hexBinary mesure la longueur en octets, pas en caractères.
FR	Partie 4, Section 3.2.29	p. 1917-1922	te	Aucune description normative de l'algorithme de hachage des mots de passe n'est fourni, donc on ne peut supposer que cette fonctionnalité sera interopérable. Dans une section d'information, un code source en C de 5 pages est fourni "à titre d'exemple", et il semble que l'algorithme implique des manipulations de bits dépendantes de la machine (little endian/big endian).	Fournir une définition normative et indépendante de la plateforme de l'algorithme de hashage. Le code source indépendant de la plateforme peut être fourni à titre d'exemple, mais ce texte normatif devrait être en anglais, pas dans un langage de programmation.
FR	Partie 4, Section 3.2.29	p. 1916	te	Il semblerait que si un mot de passe est entré dans une langue comme l'Arménien ou l'Ethiopien, tous les caractères le composant seront remplacés par 0x3F, ce qui rend la protection inutile. Ce n'est pas acceptable.	Arranger l'algorithme pour que le hash puisse être calculé sur n'importe quel mot de passe Unicode.
FR	Partie 4, Section 3.2.29	page 1916	te	Cet algorithme ne précise pas l'encodage du mot de passe saisi. Probablement de l'Unicode, mais quel est le codage? UTF-16BE? UTF-16LE? UTF-16 avec BOM (Byte Ordering Mark, indicateur d'ordre des octets)? Les algorithmes décrits utilisent des manipulations au niveau des octets, ce qui dépend de l'architecture de la machine ("big endian" ou "little endian"). Il est donc nécessaire que le choix de l'ordre des octets soit défini explicitement.	Rendre explicite le choix de l'ordre des octets, à la fois pour le mot de passe saisi et pour les traitements, de façon à permettre l'interopérabilité entre plateformes. Il faut garder à l'esprit que la signature résultant de l'algorithme (le "hash") peut être calculée sur une machine d'une autre architecture que celle sur laquelle le mot de passe a été entré.
FR	Partie 4, Section 3.2.29	page 1916	te	La conversion du mot de passe saisi vers une chaîne d'octets simples est ambiguë. Le mot de passe saisi peut assurément contenir des caractères de plus d'un jeu de caractères, disons par exemple un mélange Coréen/Chinois. Doit-on faire le traitement via plusieurs pages de code DBCS? Ou juste une et remplacer les caractères sans équivalent par 0x3F? Si une seule page DBCS est utilisée, comment est-elle déterminée?	Clarifier ce traitement, en particulier pour les mots de passe utilisant des caractères provenant de plus d'un jeu de caractères.
FR	Partie 4, Section 3.3.1.61	-	te	L'attribut pageSize accepte un ensemble de valeurs énumérées qui n'inclut pas toutes les valeurs de taille de page permises par ISO 216, ANSI Y14.1 et autres normes similaires.	Plutôt que d'essayer de maintenir une liste des tailles de papier, une approche plus souple serait de simplement enregistrer les dimensions du papier sélectionné.
FR	Partie 4, Section 3.3.1.69	-	te	Aucune description normative de l'algorithme de hachage des mots de passe n'est fourni, donc on ne peut supposer que cette fonctionnalité sera interopérable. Dans une section d'information, un code source en C de 5 pages est fourni "à titre d'exemple", et il semble que l'algorithme implique des manipulations de bits dépendantes de la machine (little endian/big endian).	Fournir une définition normative et indépendante de la plateforme de l'algorithme de hashage. Le code source indépendant de la plateforme peut être fourni à titre d'exemple, mais ce texte normatif devrait être en anglais, pas dans un langage de programmation.
FR	Partie 4, Section 3.3.1.69	-	te	L'attribut securityDescriptor "définit les comptes des utilisateurs qui ont le droit d'éditer cette section sans fournir un mot de passe pour accéder à cette section". C'est une chaîne de caractères. Mais aucune information n'est donnée sur le type de comptes utilisateurs auxquels il est fait référence, ni quel est le délimiteur. S'agit-il de comptes locaux sur la machine, séparés par des virgules? Ou des attributs DN de LDAP séparés par des point-virgules? Il n'y aura pas d'interopérabilité si ceci n'est pas défini.	Définir complètement cet attribut.
FR	Partie 4, Section 5.1.12.28	-	te	Il est dit que la longueur est de "exactement 3 caractères". C'est incompatible avec l'extrait de schéma qui la définit comme faisant 3 octets ou 6 caractères.	Clarifier la définition. En particulier, il faut remarquer que xsd:hexBinary mesure la longueur en octets, pas en caractères.

FR	Partie 4, Section 5.1.12.37	-	te	Il est dit que la valeur Panose est utilisée "pour que les applications générant des documents au format Office Open XML Standard puisse déterminer le type de police le plus proche si nécessaire". Cependant aucune métrique de distance entre polices, ou d'heuristique de correspondance des polices n'est décrit.	Décrire la procédure de correspondance de polices attendue.
FR	Partie 4, Section 5.1.12.37	-	te	Pourquoi y a-t-il plusieurs définitions différentes de la valeur Panose, à la fois dans Word Processing ML et dans Drawing ML?	Puisqu'il s'agit de la même chose, elle devrait être définie dans un schéma partagé.
FR	Partie 4, Section 5.1.3.4	-	te	Ceci décrit l'attachement d'une vidéo QuickTime à un objet de présentation. Aucune description du format QuickTime n'est fournie. Sans précision de la version et des codecs supportés, il n'y aura pas d'interopérabilité.	Fournir une référence externe pour la/les version(s) de format QuickTime attendues ici ainsi qu'un codec interopérable.
FR	Partie 4, Section 6	-	te	OOXML spécifie ici un langage à balises appelé Vector Markup Language (VML) qui, en plus de DrawingML, définit un vocabulaire pour décrire les objets graphiques. La section 6.1 précise: "Le format DrawingML est un format nouveau et plus riche créé dans le but de remplacer à terme toute utilisation de VML dans les formats Office Open XML. VML doit être considéré comme obsolète, et inclus dans Office Open XML uniquement pour des raisons de compatibilité ascendante, des les nouvelles applications ayant besoin d'un format de fichier pour les dessins sont fortement encouragées à utiliser DrawingML". Pour les utilisateurs de OOXML, le besoin du support de VML en plus de DrawingML a un coût d'implémentation important (la spécification VML fait plus de 600 pages), ce qui désavantagerait tous les éditeurs sauf Microsoft, et aurait un impact sur l'interopérabilité.	Retirer VML de OOXML. Les éditeurs ayant accès à la documentation de l'ancien format binaire, comme Microsoft, sont libres de convertir le VML inclus vers le format "nouveau et enrichi" DrawingML, en même temps qu'ils convertissent les documents en OOXML.
FR	Partie 4, Section 6.2.19	p. 4655, "gfxdata"	te	Décrit un attribut "gfxdata" pour l'élément "shape", qui "contient du contenu DrawingML" qui est "encodé en base-64". Cependant, le "contenu de ce paquetage est défini par l'application", donc même si il "doit utiliser les Parties définies par ce standard chaque fois que c'est possible", il n'y a pas suffisamment d'information pour qu'une implémentation indépendante lise ces données ou affiche ce "contenu DrawingML" incorporé. Si nous devons avoir un nouveau langage à balises de graphiques en XML, et ignorer le langage SVG existant, alors utilisons au moins ce langage à balises dans sa forme canonique, en tant que XML valide (pas empaqueté dans un attribut), et sans l'étendre de manière dépendante de l'application.	Définir ceci d'une manière interopérable.