

Personnalisation des factures et relevés PDF/HTML avancés dans NetSuite

By houseblend.io Publié le 17 avril 2026 39 min de lecture



Résumé analytique

Ce rapport fournit une analyse complète des **modèles PDF/HTML avancés de NetSuite**, en se concentrant spécifiquement sur la **personnalisation des factures et des relevés**. Les modèles PDF/HTML avancés sont une fonctionnalité puissante de SuiteCloud qui permet aux [administrateurs NetSuite](#) et aux utilisateurs de **concevoir et de contrôler les formulaires imprimés et envoyés par e-mail** des transactions avec une flexibilité bien supérieure à celle des anciennes mises en page d'impression de base (Source: [docs.oracle.com](#)) (Source: [docs.oracle.com](#)). En tirant parti d'une combinaison de scripts HTML, CSS et FreeMarker, les organisations peuvent adapter **l'apparence, la convivialité et le contenu des données** des factures et des relevés pour répondre aux exigences de marque, réglementaires et commerciales. Les principales conclusions de ce rapport incluent :

- Activation et configuration** : La fonctionnalité de modèles PDF/HTML avancés doit d'abord être activée via *Configuration > Société > Activer les fonctionnalités (SuiteCloud)* (Source: [docs.oracle.com](#)). Les formulaires de transaction personnalisés doivent ensuite être configurés pour utiliser des modèles avancés en choisissant le type d'impression « Avancé » et en sélectionnant les modèles souhaités pour l'impression et l'e-mail (Source: [docs.oracle.com](#)) (Source: [docs.oracle.com](#)). Une fois activés, les modèles avancés remplacent entièrement les **mises en page d'impression de base** héritées, qu'Oracle prévoit de supprimer (Source: [docs.oracle.com](#)) (Source: [docs.oracle.com](#)).
- Capacités de personnalisation** : Les modèles avancés prennent en charge **tous les types de transactions et d'impression** que les mises en page de base prenaient en charge (Source: [docs.oracle.com](#)) (Source: [docs.oracle.com](#)). Les utilisateurs peuvent commencer avec les modèles standard d'Oracle (ou tout modèle avancé personnalisé) puis les modifier en utilisant un mélange d'édition WYSIWYG et de création directe de code source (Source: [docs.oracle.com](#)) (Source: [docs.oracle.com](#)). Les modèles permettent l'inclusion de n'importe quel *champ d'enregistrement ou sous-liste*, y compris les [champs personnalisés](#), à travers des enregistrements associés via des directives FreeMarker (Source: [docs.oracle.com](#)) (Source: [docs.oracle.com](#)). Ils intègrent également nativement les logos et le style de l'entreprise, permettent une logique conditionnelle (par exemple, afficher des champs uniquement s'ils sont renseignés) et prennent en charge les fonctionnalités de style HTML/CSS pour une mise en forme professionnelle (Source: [docs.oracle.com](#)) (Source: [www.thenetsuitepro.com](#)).

- **Personnalisation des factures** : Les factures peuvent être fortement personnalisées tant au niveau de la structure que du contenu. Les exemples vont de l'image de marque (ajout de logos, couleurs, mentions légales en pied de page) aux améliorations fonctionnelles (mise en évidence des articles de grande valeur, affichage des codes de programme de fidélité, ajustement des sous-totaux) (Source: www.thenetsuitepro.com) (Source: pdfcoffee.com). Des experts tiers fournissent des exemples prêts à l'emploi (par exemple, un modèle de facture avec logo centré, informations d'en-tête, tableau détaillé et total) que les organisations peuvent adapter (Source: www.thenetsuitepro.com). Les considérations réglementaires jouent également un rôle : par exemple, les [SuiteApps de facturation électronique](#) de NetSuite (Malaisie, Amérique du Nord, Japon, etc.) fournissent des modèles standard et **avertissent de ne pas modifier ces valeurs par défaut** pour éviter les problèmes de conformité (Source: docs.oracle.com) (Source: docs.oracle.com) ; au lieu de cela, ils demandent aux utilisateurs de créer et d'attribuer des **modèles avancés personnalisés** pour la facturation sur une base client par client dans les paramètres de la SuiteApp.
- **Personnalisation des relevés** : De même, les relevés (relevés de compte client) peuvent être adaptés. Le modèle de relevé standard prend en charge les champs pour les tranches de vieillissement (par exemple, « 1-30 jours », « Actuel », etc.), les totaux dus et les détails des articles pour chaque facture ou paiement (Source: docs.oracle.com) (Source: docs.oracle.com). Par exemple, une discussion sur le forum des utilisateurs souligne la demande courante d'**inclure des lignes de facture détaillées** (date, date d'échéance, numéro de facture, solde) sur les relevés en utilisant des modèles avancés (Source: community.oracle.com). Les comptes multi-filiales (OneWorld) peuvent définir un modèle de relevé *multidevise* pour formater correctement les soldes et les symboles monétaires pour chaque filiale (Source: docs.oracle.com) (Source: docs.oracle.com).
- **Mise en œuvre technique** : Les administrateurs peuvent gérer les modèles via le [SuiteCloud Development Framework \(SDF\)](#), un script ou [SuiteScript](#). La documentation de NetSuite cite explicitement l'utilisation de SuiteScript pour générer des formulaires imprimés qui utilisent des modèles avancés, permettant l'intégration de la génération de PDF dans des flux de travail scriptés (Source: docs.oracle.com). De plus, les modèles avancés respectent entièrement les *préférences d'impression* au niveau de l'entreprise (telles que la langue par défaut ou le logo par filiale) (Source: docs.oracle.com) (Source: docs.oracle.com), et prennent en charge les normes d'édition HTML modernes, y compris le CSS en ligne et une pagination appropriée.
- **Cas d'utilisation et perspectives** : De nombreux scénarios réels bénéficient des modèles avancés. Les organisations ont souvent besoin de factures personnalisées pour renforcer les relations avec les clients et favoriser le paiement rapide – une pratique largement conseillée dans la littérature industrielle (Source: www.highradius.com) (Source: apexsoft.com). Par exemple, un cas échantillon (« SuiteDreams Furniture ») démontre l'ajout de champs personnalisés comme *Code de fidélité* sur un modèle de facture et l'utilisation de FreeMarker pour manipuler le texte (par exemple, extraire une sous-chaîne) (Source: pdfcoffee.com) (Source: pdfcoffee.com). Les consultants soulignent également des techniques avancées : une entreprise a construit un « moteur de rendu de contenu » pour extraire des données de recherches enregistrées dans des modèles sans script, répondant aux situations où les modèles natifs manquent de données requises (Source: blog.prolecto.com) (Source: blog.prolecto.com).
- **Orientations futures** : NetSuite améliore continuellement la plateforme ERP en phase avec les tendances technologiques. Toutes les nouvelles améliorations sont canalisées vers les modèles avancés, et Oracle a confirmé qu'il n'y aurait plus de mises à niveau des mises en page de base (Source: docs.oracle.com) (Source: docs.oracle.com). Pour l'avenir, NetSuite intègre des fonctionnalités basées sur l'IA (texte génératif, traitement automatisé des factures, etc. à partir de 2025) qui pourraient finalement s'étendre à la création de modèles et à la génération de contenu (Source: suitecentric.com) (Source: www.highradius.com). Par exemple, l'introduction d'outils comme *Text Enhance* pour le contenu automatisé suggère qu'à l'avenir, les machines pourraient aider à élaborer des documents personnalisés. De plus, les initiatives mondiales de facturation électronique en cours (telles que les mandats de facturation électronique de l'UE et de l'Asie) influenceront probablement la conception des modèles, en intégrant les normes de documents électroniques dans le cadre des modèles avancés.

Le rapport suivant est organisé en sections majeures qui approfondissent chacun de ces aspects. Il s'appuie sur la documentation officielle d'Oracle, les discussions de la communauté NetSuite, les analyses de l'industrie et des exemples réels pour présenter une vue holistique de la personnalisation des modèles PDF/HTML avancés pour les factures et les relevés. Toutes les affirmations et conclusions sont étayées par des citations provenant de sources crédibles.

Introduction et contexte

NetSuite et personnalisation des documents ERP

NetSuite est un système de planification des ressources d'entreprise (ERP) basé sur le cloud de premier plan, utilisé par des milliers d'organisations dans le monde pour la finance, le CRM, le commerce électronique et d'autres fonctions commerciales essentielles. Les documents transactionnels critiques tels que les **factures** et les **relevés** servent à la fois de dossiers financiers et de communications destinées aux clients. Les factures et les relevés incluent généralement des détails de tarification, des dates d'échéance et potentiellement des messages personnalisés ou une image de marque. Avec la mondialisation et l'évolution des attentes des clients, de nombreuses entreprises exigent que ces documents soient **hautement personnalisés** pour refléter l'image de marque de l'entreprise, se conformer aux réglementations locales (par exemple, les exigences fiscales ou de facturation électronique) et améliorer la clarté.

Dans les premières versions de NetSuite, la personnalisation des documents était limitée aux mises en page PDF et HTML de base. Les administrateurs ne pouvaient modifier que les *mises en page PDF des formulaires de transaction* ou les *mises en page HTML des formulaires de transaction*, qui disposaient d'un éditeur WYSIWYG restrictif et offraient un style limité. Ces mises en page d'« **impression de base** » présentaient des étiquettes fixes et une mise en forme limitée (par exemple, des en-têtes sur fond noir) (Source: docs.oracle.com). Bien qu'utilisables, ces anciennes mises en page ne pouvaient pas répondre à des besoins plus sophistiqués — par exemple, la logique conditionnelle, le positionnement personnalisé ou le style avancé n'étaient généralement pas pris en charge. En conséquence, les organisations avaient souvent besoin de SuiteScript spécialisé ou d'intégrations tierces pour obtenir des résultats plus soignés.

Évolution vers les modèles PDF/HTML avancés

Pour remédier à ces limitations, NetSuite a introduit les **modèles PDF/HTML avancés** dans le cadre de sa plateforme SuiteCloud. Les modèles avancés sont basés sur des technologies standard de l'industrie : un système de type HTML/CSS combiné à *FreeMarker* (un langage de modélisation intégré) pour la fusion de données. Ils permettent un mélange de conception et de script : les utilisateurs peuvent placer des champs dans une mise en page, les styliser avec du CSS et utiliser des balises FreeMarker (`{...}`, `<#if>`, `<#list>`, etc.) pour itérer sur les enregistrements, appliquer une logique conditionnelle, formater des valeurs, et plus encore (Source: www.thenetsuitepro.com) (Source: pdfcoffee.com). Cela aligne NetSuite avec la modélisation Web moderne, rendant la personnalisation beaucoup plus puissante. Les modèles avancés **prennent en charge le rendu PDF et HTML** des formulaires, selon les préférences d'impression/e-mail de l'entreprise (Source: docs.oracle.com).

Du point de vue organisationnel, l'utilisation de modèles avancés transforme les factures/relevés de formulaires statiques en documents commerciaux dynamiques. Les entreprises peuvent les utiliser pour s'assurer que tous les champs requis (par exemple, données personnalisées, éléments spécifiques à la taxe) apparaissent exactement là où ils sont nécessaires, et que le format respecte le guide de style de l'entreprise. Les administrateurs peuvent gérer plusieurs modèles et basculer facilement entre eux (par exemple, différents modèles par filiale ou type de client). Dans les comptes OneWorld multi-entités, les modèles avancés peuvent même générer un contenu différent en fonction du contexte de la filiale (comme la personnalisation des pieds de page par filiale) (Source: www.thenetsuitepro.com).

Au fil du temps, Oracle a signalé que les modèles avancés sont l'avenir du formatage de documents de NetSuite. La documentation officielle indique que « **les nouvelles améliorations sont fournies exclusivement à l'impression avancée** » et que les mises en page PDF/HTML de base seront éventuellement supprimées (Source: docs.oracle.com) (Source: docs.oracle.com). Cette transition souligne l'importance pour les organisations de migrer les formulaires standard existants vers des modèles avancés. Pour les nouvelles implémentations de NetSuite, les conseils sont sans équivoque : utilisez des modèles avancés plutôt que des mises en page héritées (Source: community.oracle.com) (Source: docs.oracle.com). En pratique, de nombreuses bonnes pratiques d'implémentation adhèrent à cela : les consultants configurent les modèles avancés préférés lors de la mise en service pour assurer une meilleure flexibilité à long terme.

En résumé, les modèles PDF/HTML avancés représentent un changement stratégique dans la gestion des documents de NetSuite, permettant des personnalisations sophistiquées de factures et de relevés qui n'étaient pas possibles (ou nécessitaient des solutions de contournement) sous l'ancien modèle d'impression de base. Les sections ci-dessous explorent en profondeur les capacités, la configuration, les cas d'utilisation et les orientations futures de ces modèles. Chaque affirmation et recommandation est étayée par la documentation de NetSuite, les commentaires de la communauté et l'analyse d'experts pour fournir un guide complet et fondé sur la recherche.

Aperçu des modèles PDF/HTML avancés

Activation et accès aux fonctionnalités

Avant toute personnalisation, la fonctionnalité **Modèles PDF/HTML avancés** doit être activée dans NetSuite. Cela est contrôlé via **Configuration > Société > Activer les fonctionnalités**, sous le sous-onglet SuiteCloud. Les administrateurs cochent simplement l'option « **Modèles PDF/HTML avancés** » et enregistrent (Source: docs.oracle.com). Une fois activé, le système fournit un nouveau chemin de menu : **Personnalisation > Formulaires > Modèles PDF/HTML avancés**, où les utilisateurs peuvent afficher, modifier ou créer des modèles pour le compte (Source: docs.oracle.com).

Par défaut, NetSuite fournit un ensemble de modèles avancés *standard* (par exemple, facture standard, relevé standard, etc.) pour chaque type d'impression pris en charge. Ces modèles contiennent des espaces réservés pour les champs typiques de cet enregistrement (par exemple, `${record.tranid}` pour le numéro de facture, `${record.total}` pour le montant, et ainsi de suite). Les administrateurs peuvent cloner ou modifier ces modèles standard ou en créer de nouveaux pour répondre à leurs besoins. Chaque modèle a un ID de script et un titre ; les modèles personnalisés peuvent avoir des ID de script commençant par un « _ » si vous le souhaitez.

Notamment, l'activation des modèles avancés débloque également des options lors de l'utilisation des **Formulaires de transaction**. Dans *Personnalisation > Formulaires > Formulaires de transaction*, chaque formulaire personnalisé (ou standard) affiche désormais une option **Type d'impression** (Source: docs.oracle.com). Les administrateurs sélectionnent « **Avancé** » pour indiquer que ce formulaire utilisera des modèles avancés au lieu des mises en page de base (Source: docs.oracle.com). Ensuite, ils choisissent les modèles souhaités dans les menus déroulants pour le **Modèle d'impression** et le **Modèle d'e-mail** (Source: docs.oracle.com). Cela signifie qu'un modèle peut être utilisé pour l'impression à l'écran (PDF) et un autre pour l'envoi par e-mail, si nécessaire. En pratique, la plupart des entreprises utilisent le même style de marque pour les deux, mais cette flexibilité existe (par exemple, on peut inclure les conditions générales uniquement sur la copie envoyée par e-mail).

Il est important de noter qu'après avoir défini un formulaire sur « Avancé », toutes les transactions enregistrées avec ce formulaire généreront une sortie conforme au modèle avancé. La documentation précise que, si nécessaire, un administrateur peut ultérieurement rétablir le formulaire en impression de base en remplaçant le Type d'impression par « De base » (Source: docs.oracle.com). Aucune donnée n'est perdue — les mises en page PDF/HTML héritées restent disponibles dans les menus déroulants — mais à l'avenir, les modèles avancés sont privilégiés pour le nouveau contenu et les améliorations futures.

Autorisations : L'accès aux modèles avancés est contrôlé par des autorisations spécifiques. Un administrateur ou un utilisateur disposant de l'autorisation **Modèles PDF/HTML avancés** peut créer et modifier ces modèles. (Pour les mises en page de base, des autorisations différentes s'appliquaient.) En général, les administrateurs configurent les modèles lors de la mise en œuvre, puis attribuent l'accès à certains rôles (par exemple, les rôles Marketing ou Facturation) pour permettre la mise à jour des étiquettes et du style, tout en restreignant les autres.

Impression de base vs Impression avancée

Comprendre la distinction entre l'impression de base et l'impression avancée est crucial pour toute stratégie de personnalisation. La documentation d'Oracle fournit une comparaison claire :

- **Mises en page d'impression de base** : Il s'agit des anciens formats (via *Mises en page PDF / Mises en page HTML des formulaires de transaction*). Ils offrent un éditeur fixe basé sur le formulaire où les étiquettes apparaissent au-dessus des valeurs, avec un style limité (par exemple, des barres d'en-tête noires) (Source: docs.oracle.com). Les comptes OneWorld utilisant l'impression de base utilisaient toujours le logo/l'adresse de la filiale principale pour les journaux/bons de commande (Source: docs.oracle.com), ce qui limitait la personnalisation par entité. Aucune logique de script ou conditionnelle n'était possible dans le modèle.
- **Modèles PDF/HTML avancés** : Introduits pour surmonter ces limites, les modèles avancés permettent une édition complète en HTML/CSS et basée sur FreeMarker. Les différences clés documentées incluent :
 - **Personnalisation** : Les modèles avancés permettent un repositionnement arbitraire des champs et un style riche, alors que les mises en page de base étaient très limitées. Les modèles avancés peuvent être modifiés en **mode WYSIWYG** ou en **mode code source** (Source: docs.oracle.com) (Source: docs.oracle.com).
 - **Formats de sortie** : Les deux prennent en charge la sortie PDF et HTML (pour l'impression à l'écran). Cependant, les modèles avancés s'adaptent automatiquement aux préférences d'impression de l'entreprise (par exemple, sortie e-mail vs impression) (Source: docs.oracle.com).
 - **Prise en charge multi-entités** : Les modèles avancés prennent correctement en charge l'impression multi-filiales par défaut, alors que les mises en page de base présentaient des problèmes connus (comme noté dans les conseils OneWorld) (Source: docs.oracle.com).

- **Améliorations futures** : Il est crucial de noter qu'**Oracle a déclaré que les mises en page de base seront obsolètes**. L'impression avancée bénéficie de toutes les nouvelles fonctionnalités et améliorations (Source: docs.oracle.com) (Source: docs.oracle.com). En fait, l'équipe de support de NetSuite recommande explicitement aux nouvelles implémentations d'utiliser des modèles avancés, car « vous disposez de plus de personnalisation » et « il n'y a plus de mises à niveau pour l'impression de base » (Source: community.oracle.com).
- **Normes industrielles** : L'éditeur de modèles avancés utilise des normes HTML modernes (CSS en ligne, style de tableau, etc.), ce qui facilite la création de documents visuellement attrayants (Source: docs.oracle.com). Les mises en page de base étaient plus rigides et non basées sur les normes Web actuelles.

Le tableau 1 ci-dessous résume ces comparaisons :

ASPECT	MISES EN PAGE D'IMPRESSION DE BASE	MODÈLES PDF/HTML AVANCÉS
Mode d'édition	Éditeur fixe basé sur le formulaire ; étiquettes au-dessus des valeurs (Source: docs.oracle.com)	Éditeur WYSIWYG + mode code source complet (FreeMarker) (Source: docs.oracle.com)
Style/Format	Style limité (polices/couleurs par défaut) ; lignes d'en-tête noires ; pas de support CSS (Source: docs.oracle.com)	Style HTML/CSS complet (CSS en ligne recommandé) ; mises en page flexibles (Source: docs.oracle.com)
Champs et données	Peut placer des champs sur la mise en page ; pas de logique conditionnelle	Peut placer des champs d'enregistrement et associés ; prend en charge le script (FreeMarker) pour afficher/formater les données de manière conditionnelle (Source: www.thenetsuitepro.com) (Source: pdfcoffee.com)
Support de transaction	Utilisé pour tous les types de transaction/impression pris en charge à l'origine (Source: docs.oracle.com)	Prend en charge tous les types de transaction/impression (aucune perte de fonctionnalité par rapport à la base) (Source: docs.oracle.com) (Source: docs.oracle.com)
Multi-entité (OneWorld)	Utilise le logo/l'adresse de la filiale principale dans les impressions (Source: docs.oracle.com)	Respecte la filiale de l'enregistrement ; peut changer par entité (exemples de pieds de page spécifiques à une filiale) (Source: www.thenetsuitepro.com)
Mises à jour & Futur	Obsolète – aucune nouvelle amélioration ; sera supprimé dans une future version (Source: docs.oracle.com)	Développement actif – reçoit toutes les nouvelles fonctionnalités (ex. support IA, codes QR, nouvelles API) (Source: docs.oracle.com) (Source: docs.oracle.com)
Localisation & Sortie	Paramètres régionaux choisis par priorité Client/Filiale/Société (Source: docs.oracle.com)	Les modèles avancés respectent pleinement les paramètres de traduction/localisation ; la sortie PDF ou HTML suit les préférences d'e-mail/impression (Source: docs.oracle.com)
Cas d'utilisation	Convient aux changements de formulaire très basiques ; largement remplacé maintenant	Permet la personnalisation de la marque (logos, couleurs), la logique conditionnelle (si-alors sur les champs), les mises en page avancées (tableaux, images) (Source: www.thenetsuitepro.com) (Source: docs.oracle.com)
Autorisations	Peut être modifié par des rôles avec des autorisations de <i>Mise en page PDF/HTML de formulaire de transaction</i>	Nécessite l'autorisation <i>Modèles PDF/HTML avancés</i> ; intégré au framework SuiteCloud Dev (Source: docs.oracle.com)

Tableau 1 : Comparaison des mises en page d'impression de base et des modèles PDF/HTML avancés dans NetSuite. Citations : Aide Oracle NetSuite.

Le consensus parmi les professionnels de NetSuite est que **les modèles avancés devraient être la pratique standard à l'avenir**, compte tenu de leurs capacités et de la feuille de route de la plateforme (Source: community.oracle.com) (Source: docs.oracle.com). Les personnalisations héritées des mises en page de base doivent être migrées vers des modèles avancés pour tirer parti de l'ensemble de fonctionnalités plus riche.

Travailler avec l'éditeur de modèles

NetSuite fournit un **Éditeur de modèles** intégré pour les modèles avancés, accessible sous *Personnalisation > Formulaires > Modèles PDF/HTML avancés*. L'éditeur dispose de deux modes principaux :

- Mode WYSIWYG** : Un éditeur visuel où les utilisateurs font glisser des champs, stylisent le texte et modifient le formatage à l'aide de barres d'outils et de propriétés. C'est similaire à une interface de traitement de texte de base, avec des options pour ajouter des zones de texte, des lignes/colonnes de tableau, des images, etc. Par exemple, un utilisateur peut cliquer sur un champ dans la palette (par exemple `#{record.tranid}`) pour « N° de facture » et l'éditeur insère ce paramètre fictif. Les champs peuvent être repositionnés, leur taille/couleur de police modifiée et leur alignement ajusté. Le mode WYSIWYG est idéal pour les changements « rapides » comme déplacer des champs, ajouter des en-têtes ou un style simple. Cependant, les opérations complexes (telles que l'ajout de logique conditionnelle ou l'accès à des sous-enregistrements imbriqués) nécessitent souvent de passer au mode Code source.
- Mode Code source** : Il affiche le code FreeMarker/HTML brut du modèle. Les utilisateurs peuvent modifier les directives FreeMarker (`<#if>`, `<#list>`, `#{...?string}`, etc.) ainsi que tout HTML/CSS personnalisé. Toute modification en mode Source est reflétée dans le mode WYSIWYG et vice versa. Certaines tâches avancées—comme parcourir des sous-listes (`<#list record.item as item>` pour itérer les articles de ligne), intégrer des images (par exemple `` pour afficher le logo de l'entreprise (Source: www.thenetsuitepro.com), ou appliquer un formatage conditionnel—sont plus naturellement effectuées en mode code.

Opérations courantes dans l'éditeur

L'éditeur prend en charge une variété d'opérations pertinentes pour les factures et les relevés :

- Ajout/Suppression de champs** : Les administrateurs peuvent insérer n'importe quel champ disponible à partir de l'enregistrement ou des enregistrements associés. Par exemple, dans un modèle de **Facture**, des champs tels que `#{record.entity}` (nom du client), `#{record.trandate}` (date de la facture) et `#{record.total}` (montant dû) sont disponibles prêts à l'emploi. Chaque enregistrement possède des sous-listes associées (pour la facture, une sous-liste `record.item` contenant les articles de ligne). Le tableau 2 ci-dessous illustre quelques exemples de champs disponibles dans les modèles standard de Facture et de Relevé (à partir de la référence d'aide d'Oracle) pour le contexte.

TYPE DE MODÈLE	EXEMPLES DE CHAMPS (ACCESSIBLES VIA LE MODÈLE AVANCÉ)	RÉFÉRENCE
Facture	<i>N° de facture</i> (<code>#{record.tranid}</code>), <i>Date</i> (<code>#{record.trandate}</code>), <i>Client</i> (<code>#{record.entity}</code>), <i>Adresse de facturation</i> (<code>#{record.billaddress_text}</code>), <i>Conditions</i> , <i>Date d'échéance</i> , <i>Devise</i> , <i>Total</i> (<code>#{record.total}</code>), <i>Articles de ligne</i> (<i>description</i> , <i>quantité</i> , <i>taux</i> , <i>montant</i>) dans la sous-liste <code>record.item</code>	Champs de facture standard (Source: docs.oracle.com) (Source: docs.oracle.com)
Relevé	<i>Actuel</i> , <i>1-30 jours</i> , <i>31-60</i> , <i>61-90</i> , <i>Plus de 90 jours</i> (<code>#{record.aging1}... #{record.aging5}</code>), <i>Numéro de compte</i> , <i>Montant dû</i> (<code>#{record.amountdue}</code>), <i>Sous-liste de lignes</i> (<i>pour chaque facture</i> : <i>Date</i> , <i>Date d'échéance</i> , <i>Solde</i> , <i>Charge</i> , <i>Paiement</i> , <i>Mémo</i>)	Champs de relevé standard (Source: docs.oracle.com) (Source: docs.oracle.com)

Tableau 2 : Exemples de champs disponibles dans les modèles avancés standard pour les factures et les relevés. (Seul un sous-ensemble est affiché ; les listes réelles sont exhaustives.)

Les champs non présentés ci-dessus incluent le client, la filiale, le département, la classe, etc. La **Référence des modèles avancés** (aide Oracle) fournit des listes complètes de tous les champs et sous-listes pour chaque type de modèle. Elle note que les champs disponibles dépendent des fonctionnalités activées et des champs personnalisés dans le compte (Source: docs.oracle.com). Le volet « Tri des champs » de

l'éditeur montre exactement quels champs sont présents ; tous les champs personnalisés ajoutés aux enregistrements (par exemple, les champs de corps de transaction ou les champs de ligne) apparaîtront également ici.

- Style et mise en page** : Les utilisateurs peuvent formater le texte (polices, couleurs, alignement), créer des tableaux, fusionner des cellules, insérer des images (logos d'entreprise ou photos de produits) et utiliser CSS pour un contrôle plus précis. Par exemple, dans l'exemple d'aide ci-dessus (Source: www.thenetsuitepro.com) (Source: www.thenetsuitepro.com), le code du modèle de facture centre le logo de l'entreprise et le texte d'en-tête en utilisant des styles en ligne (`style="text-align:center;"`) et applique des couleurs de ligne alternées pour la lisibilité (`<#if i_index % 2 == 0>style="background:#f9f9f9;"`). Les modèles avancés prennent même en charge les en-têtes et pieds de page de page personnalisés (tels que l'inclusion de dates dynamiques ou de texte de pied de page spécifique à une filiale (Source: www.thenetsuitepro.com)).
- Logique conditionnelle** : L'une des fonctionnalités les plus puissantes est la logique FreeMarker. Des conditions simples peuvent masquer ou afficher des champs. Par exemple, un `<#if>` FreeMarker peut afficher une note promotionnelle uniquement si un champ correspondant n'est pas vide, ou mettre en évidence les articles de ligne dépassant un seuil. Les exemples de NetSuite Pro montrent comment une mise en évidence conditionnelle est appliquée aux lignes où le montant > 5000 (donnant à cette ligne un arrière-plan rose) (Source: www.thenetsuitepro.com). De même, la modularité est possible : un cas commuté (`<#switch>` dans [9†L82-L90]) choisit un texte de pied de page de filiale différent en fonction de `record.subsidiary`.
- Calculs et formatage** : FreeMarker prend en charge le formatage et les calculs intégrés. Les champs numériques peuvent être formatés (`string["#",##0.00"]`), comme illustré dans [9†L41-L44] pour les taux et les montants. Des fonctions intégrées telles que `substring`, `round` ou le formatage de date peuvent être utilisées pour transformer les données. Par exemple, un exemple officiel utilise `substring` pour n'afficher qu'une partie d'un code de fidélité sur la facture (Source: pdfcoffee.com). Les calculs de synthèse sont également accessibles : le code du modèle peut accéder aux champs de synthèse ou aux formules des transactions. Toute formule ou recherche enregistrée (saved search) intégrée à une transaction peut ainsi alimenter le modèle via ces fonctions intégrées.

Prévisualisation et débogage

Après modification, l'éditeur de modèles permet de **prévisualiser** le résultat. Si le code contient des erreurs de syntaxe, NetSuite les signale (par exemple, directives manquantes, syntaxe incorrecte). Il ne sauvegardera pas les modèles invalides, ce qui aide à prévenir les sorties vierges ou corrompues. Cette boucle de rétroaction immédiate est utile lors du développement de modèles. De plus, comme le note [12], l'éditeur fournit des messages d'erreur exploitables si le modèle ne peut pas être enregistré (par exemple, accolades manquantes, variables non définies) (Source: docs.oracle.com).

En guise de bonne pratique, Oracle et les experts recommandent des **tests incrémentaux** : effectuez une petite modification (comme l'ajout d'un champ ou d'une ligne de tableau), puis prévisualisez ou générez une facture pour vérifier. Cela évite de devoir résoudre plusieurs problèmes à la fois. Lors de la création de modèles complexes, certains développeurs exportent la source, utilisent des outils externes d'embellissement de code (pour formater correctement le HTML/FreeMarker), puis collent le résultat pour le test final (le guide de formation suggère même Pretty Diff ou MinifyCode à cet effet (Source: pdfcoffee.com)).

Exemple : Ajout de champs et de logique

Pour illustrer une modification typique, considérons la personnalisation d'une facture pour inclure deux nouvelles colonnes, *Code de fidélité* et *Programme de fidélité*, tout en supprimant deux colonnes existantes, *Conditions* et *Partenaire*. En mode WYSIWYG, un administrateur peut sélectionner et supprimer les champs Conditions et Partenaire (s'ils sont présents) et insérer deux nouvelles colonnes vides là où il le souhaite. Il insérerait les étiquettes dans ces colonnes, puis, en mode source, placerait les expressions FreeMarker `record.custbody_loyaltycode` et `record.custbody_loyaltyprogram` (en supposant que ces champs personnalisés existent dans l'enregistrement de la facture). Le scénario de formation pour « SuiteDreams Furniture Company » décrit exactement ce processus (Source: pdfcoffee.com). Après l'enregistrement et la prévisualisation, ces nouveaux champs de fidélité apparaissent sur chaque facture. Comme le note le guide, l'administrateur passe ensuite en mode source pour affiner et éventuellement appliquer des fonctions FreeMarker (par exemple, extraire une sous-chaîne du code), démontrant ainsi l'interaction entre les deux modes (Source: pdfcoffee.com) (Source: pdfcoffee.com).

En production, de tels champs personnalisés peuvent provenir de programmes de fidélité, d'identifiants d'abonnement ou d'autres extensions. La possibilité de les intégrer facilement dans un formulaire imprimé (avec une logique conditionnelle et un formatage) est un avantage clé des modèles avancés par rapport aux outils hérités.

Personnalisation du modèle de facture

Image de marque et mise en page

Les factures sont généralement les principaux documents reçus par les clients ; leur apparence revêt donc une importance capitale pour l'identité de l'entreprise. Grâce aux modèles avancés, les entreprises peuvent **appliquer leur style de marque** de manière approfondie. Les éléments de marque courants incluent :

- **Logos** : Les modèles avancés peuvent référencer directement le logo de l'entreprise défini dans la configuration (ou un fichier image). Dans la référence de script NetSuite, `{record.logo}` fait référence à l'image du logo de l'entreprise sur l'enregistrement de transaction (Source: www.thenetsuitepro.com). Placer `` dans le modèle permet d'intégrer le logo (avec une largeur/hauteur contrôlable). L'exemple dans [9] centre le logo en haut de la facture.
- **Couleurs et polices** : Contrairement aux mises en page de base, les modèles avancés permettent d'utiliser du CSS en ligne pour définir les couleurs d'arrière-plan, les familles de polices, les tailles de police et les couleurs de texte. Par exemple, les en-têtes de tableau peuvent utiliser un arrière-plan gris clair, des polices en gras, etc., comme on le voit dans [9†L28-L36]. Les polices peuvent également être ajustées pour correspondre aux directives de la marque.
- **Informations d'en-tête** : Le modèle peut formater et positionner le texte d'en-tête (comme le titre « Facture », les dates ou les informations client) comme souhaité. Dans [9] (Exemple 1), le titre « Facture » est défini comme un `<h2>` centré horizontalement. L'adresse de l'entreprise, les coordonnées ou les mentions légales peuvent également être placées dans les en-têtes ou les pieds de page en utilisant des lignes ou des blocs fixes.
- **Pieds de page et conditions** : Les pieds de page incluent souvent les conditions de paiement, la numérotation des pages (en utilisant les directives de date/heure ou de page de FreeMarker) ou des mentions légales. Les modèles avancés permettent d'inclure des déclarations telles que « Merci de votre confiance » ou des mentions spécifiques par région.

Contenu dynamique et calculs

Outre l'image de marque statique, les factures nécessitent souvent un contenu dynamique :

- **Lignes d'articles** : Un tableau est généralement utilisé. Le modèle avancé peut parcourir chaque ligne d'article sur la facture en bouclant sur `record.item` (la sous-liste des articles). L'extrait `<#list record.item as i>` dans [9] génère des lignes de tableau pour chaque article, en insérant `{i.item}`, `{i.quantity}`, `{i.rate}`, etc. Le code applique également un effet de rayures (alternance de couleurs) (`<#if i_index % 2 == 0>` pour des arrière-plans alternés) afin d'améliorer la lisibilité (Source: www.thenetsuitepro.com).
- **Formatage conditionnel** : Les entreprises peuvent souhaiter mettre en évidence certains articles (par exemple, surligner les articles à forte valeur). Les exemples de NetSuite Pro incluent un cas où les lignes dont le montant dépasse 5000 reçoivent un surlignage rose (Source: www.thenetsuitepro.com). Cela est réalisé en intégrant une condition `<#if>` qui définit un style sur la balise `<tr>`.
- **Champs personnalisés** : De nombreux scénarios commerciaux nécessitent l'affichage de données supplémentaires sur la facture. Par exemple, s'il existe des champs de corps de transaction personnalisés (comme « Code de fidélité », « Code de projet » ou « Numéro de bon de commande client »), ceux-ci peuvent être insérés. Le sélecteur de champ du modèle avancé listera les champs personnalisés par leur ID s'ils sont marqués comme « Stocker la valeur » (pour les champs de transaction) ou marqués comme champ de corps pour les articles. Alternativement, on peut les référencer manuellement en mode source. Par exemple, `{record.custbody_project_code}` afficherait un champ personnalisé nommé `custbody_project_code`. L'exemple de NetSuite Pro (Exemple 5) montre comment afficher les champs de corps personnalisés uniquement s'ils contiennent du contenu, en utilisant `<#if record.custbody_field?has_content>` autour d'eux (Source: www.thenetsuitepro.com).
- **Sous-totaux et totaux** : Le modèle peut afficher des champs calculés tels que la taxe, les frais d'expédition, le total, qui font partie de l'enregistrement. Il peut également inclure des valeurs de recherche de synthèse via des champs de formule s'ils sont configurés (bien que cela nécessite souvent des scripts mineurs ou des colonnes de recherche enregistrées poussées vers un champ).

Intégrations et conformité

La facturation moderne implique souvent la conformité et l'intégration avec des systèmes externes :

- **Facturation électronique (E-Invoicing)** : De nombreuses régions (par exemple, l'Europe, certaines parties de l'Asie) exigent que les factures soient générées selon des schémas électroniques spécifiques ou incluent des signatures numériques/codes QR. NetSuite fournit des SuiteApps de localisation pour les principaux pays qui, entre autres fonctionnalités, incluent des modèles PDF standardisés. Comme le soulignent les

documents sur la facturation électronique en Malaisie et en Amérique du Nord, il est essentiel de **ne pas modifier les modèles par défaut de la SuiteApp** ; au lieu de cela, les utilisateurs doivent créer des modèles avancés personnalisés *supplémentaires* pour leur propre usage (Source: docs.oracle.com) (Source: docs.oracle.com). Ces SuiteApps permettent également de lier un modèle avancé particulier à un enregistrement client afin que le format correct soit utilisé lors de l'envoi de factures électroniques par e-mail.

- **Taxes et devises multiples** : Si les fonctionnalités SuiteTax ou devises multiples sont activées, les modèles standard incluent les champs pertinents. Par exemple, avec la TVA ou la TPS activée, des numéros de taxe supplémentaires ou des totaux en devises multiples peuvent apparaître. Les comptes OneWorld doivent utiliser le **Relevé standard en devises multiples** ou prendre en compte ces champs lors de la conception des modèles de facture (par exemple, en incluant `#{record.currencyname}` comme dans [9†L41-L44]).
- **SuiteBundler et SDF** : Du point de vue du déploiement, les modèles avancés peuvent être regroupés et promus comme d'autres personnalisations. L'aide mentionne la gestion des modèles avec le SuiteCloud Development Framework et la copie de modèles entre comptes (Source: docs.oracle.com). Ceci est essentiel pour maintenir la cohérence entre les instances Sandbox/Production ou multi-filiales.

Exemples de personnalisations

Les ressources de la communauté et des consultants NetSuite fournissent de nombreux exemples de personnalisation. Quelques exemples illustratifs :

- **Exemple de programme de fidélité** : Dans le matériel de formation, *SuiteDreams Furniture* souhaitait afficher son code de fidélité sur les factures et n'en montrer qu'une partie. Le guide leur demande de supprimer les colonnes non pertinentes, d'ajouter les colonnes « Code de fidélité » et « Programme de fidélité », puis d'utiliser la fonction FreeMarker `substring` pour afficher uniquement les six caractères du milieu du code (Source: pdfcoffee.com) (Source: pdfcoffee.com). Cela montre comment les modèles avancés gèrent même les tâches de manipulation de chaînes. (Résultat : les factures incluent désormais une colonne qui indique, par exemple, « Code de fidélité : 123456 » où 123456 fait partie d'un code plus long.)
- **Surlignages et alertes** : Le blog Prolecto (Marty Zigman) décrit l'ajout d'une alerte sur les factures : surligner toute ligne où le prix dépasse 1 000 \$ pour avertir la comptabilité ou les commerciaux (Source: pdfcoffee.com). Bien que simple, cette règle peut réduire les plaintes des clients en attirant l'attention sur les frais importants. Un autre scénario pourrait être d'imprimer des notes différentes sur les factures en fonction du type de client ou de l'emplacement géographique ; cela pourrait être fait avec `<#switch>` sur `#{record.subsidiary}` ou un champ personnalisé.
- **Conditions de paiement et localisation** : Un autre extrait de [9] (Exemple 2) montre un modèle de commande client avec des conditions d'expédition et de paiement conditionnelles. Si la commande client a une méthode d'expédition, elle l'imprime ; sinon, elle peut l'ignorer ou indiquer « Prépayé » par défaut (Source: www.thenetsuitepro.com). Une logique similaire pourrait apparaître sur les factures si nécessaire (bien que les conditions d'expédition soient rarement sur les factures, cela illustre le concept).
- **Intégration CRM** : Les champs personnalisés provenant du CRM (comme les opportunités, l'ID de campagne, etc.) peuvent être rendus visibles sur les factures s'ils sont extraits vers l'enregistrement de transaction. Les modèles avancés rendent cela simple : on peut utiliser `#{record.opportunity}` ou `#{record.campaigncategory}` directement une fois que le champ est disponible.

Conseils de conception (Bonnes pratiques)

Les experts suggèrent plusieurs bonnes pratiques lors de la personnalisation :

- **CSS en ligne** : Bien que les modèles avancés autorisent le CSS, l'utilisation de **styles en ligne** est recommandée pour la compatibilité. (Le blog NetSuite Pro le conseille également (Source: www.thenetsuitepro.com.) Par exemple, `<th style="padding:5px;">` garantit que le style est conservé dans la sortie PDF.
- **Mises en page basées sur des tableaux** : Pour les lignes d'articles de facture, utilisez des tableaux HTML (comme indiqué dans [9]) car le rendu PDF respecte les structures de tableau de manière fiable. Gardez les tableaux simples : utilisez `<table>` avec un remplissage/bordures de cellule cohérents. Évitez les CSS complexes qui pourraient ne pas bien s'afficher dans le moteur de conversion PDF.
- **Sections modulaires** : Divisez le modèle en sections logiques (par exemple, en-tête, corps, pied de page). Cela facilite la maintenance. Certains consultants créent même des modèles séparés (par exemple, en-tête/pied de page uniquement) et les incluent via des importations FreeMarker, bien que NetSuite ne prenne pas nativement en charge les fichiers inclus dans l'éditeur de modèles. Néanmoins, organiser les blocs clairement (avec des commentaires) aide.

- **Tests sur des données réelles** : Testez toujours le modèle avec des transactions contenant des données réalistes. Par exemple, générez une facture avec plusieurs lignes d'articles, un logo présent, des champs de texte longs, des sauts de page, etc., pour détecter les problèmes de mise en page et de dépassement.
- **Cloner et réutiliser** : Il est recommandé de partir du *Modèle PDF/HTML de facture standard*. **Clonez** toujours avant de modifier, afin que l'original reste disponible si nécessaire. Nommez vos modèles personnalisés clairement, par exemple « `_CustomInvoice_EDOC` » ou similaire. Le guide de formation suggère des préfixes avec des codes d'entreprise (comme « `_sdr_invoice` ») pour plus de cohérence.
- **Contrôle de version** : Gardez des enregistrements des modifications de modèle dans une documentation ou un contrôle de version. Étant donné que l'éditeur stocke le modèle dans la base de données NetSuite, il n'y a pas de diff ou de restauration intégré ; une documentation externe est précieuse. Si vous utilisez SDF, stockez le fichier de modèle dans votre référentiel.

En suivant ces pratiques, les administrateurs et développeurs NetSuite peuvent produire des modèles de facture professionnels et fonctionnels qui améliorent l'image de l'entreprise et l'expérience utilisateur.

Personnalisation du modèle de relevé

Les relevés clients (rapports de comptes clients résumant les factures impayées et les paiements) sont un autre document clé que les entreprises personnalisent souvent. Dans NetSuite, les relevés sont généralement générés à partir d'un **enregistrement client** (via *Transactions > Clients > Imprimer les relevés*). Les modèles avancés permettent d'adapter les formulaires de **comparaison relevé/facture** lors de l'impression ou de l'envoi par e-mail.

Contenu et champs typiques

Un relevé liste généralement :

- Le nom et l'adresse du client.
- La date d'exécution du relevé.
- Le solde d'ouverture (le cas échéant).
- La liste des factures/paiements appliqués (dates, numéros, montants, solde restant).
- Le résumé de l'ancienneté (actuel, 1-30, 31-60 jours de retard, etc., et le total dû).
- Le total des conditions de paiement, les sous-totaux, etc.
- Les notes de pied de page ou les informations de remise.

Le **Modèle PDF/HTML de relevé standard** dans NetSuite fournit des champs pour présenter tout cela. Par exemple, comme indiqué dans le Tableau 2, il inclut des colonnes d'ancienneté `${record.aging1}` (actuel) à `${record.aging5}` (31-60, ..., plus de 90 jours), `${record.amountdue}`, et une sous-liste de lignes d'articles avec `${lines.datecol}`, `${lines.balance}` et `${lines.otherrefnum}` (numéro de facture ou de chèque) (Source: docs.oracle.com) (Source: docs.oracle.com). Le modèle dispose également de `${record.billaddress}` (adresse de facturation) et `${record.balance}` (le total impayé).

Personnalisation de la mise en page

Dans de nombreuses entreprises, les relevés par défaut doivent être ajustés. Par exemple :

- **Filtrage ou regroupement** : Une demande courante est le regroupement des transactions par date d'échéance ou par type. La sous-liste de lignes du modèle standard peut être restructurée. Les modèles avancés ne regroupent pas automatiquement les lignes ; tout regroupement (par exemple, par projet ou par plage de dates d'échéance) nécessiterait une logique FreeMarker. Une question sur le forum demandait exactement comment inclure la date de facture/échéance sur le relevé (Source: community.oracle.com), ce qui implique l'ajout de `${lines.datecol}` et `${lines.duedate}` dans le tableau.
- **Plusieurs factures par relevé** : La sous-liste de lignes dans un relevé correspond généralement à chaque facture/paiement. Certaines organisations souhaitent regrouper plusieurs factures en un seul relevé ou vice versa. Cela nécessite d'ajuster l'itération sur les lignes (`<#list record.line as l>`).

- **Texte spécifique à la filiale** : Dans les comptes multi-filiales, les relevés peuvent différer selon l'entité juridique. Les modèles avancés permettent des pieds de page spécifiques au bureau ou des notes fiscales en utilisant des conditions FreeMarker sur `${record.subsidiary}`. L'exemple de NetSuite Pro (Exemple 3) démontre le changement de texte de pied de page en fonction de la filiale (Source: www.thenetsuitepro.com) ; une approche similaire peut être appliquée aux relevés (par exemple, en incluant des numéros de TVA spécifiques à la filiale de chaque pays).
- **Présentation de l'ancienneté** : La mise en page du résumé de l'ancienneté peut être modifiée. Par défaut, le modèle standard affiche les totaux pour chaque tranche d'ancienneté. Une entreprise pourrait vouloir mettre en évidence la tranche d'ancienneté la plus élevée ou la convertir en graphiques (bien que les graphiques ne soient pas triviaux en PDF). Cependant, au minimum, on peut styliser ces champs (couleurs, polices) différemment en utilisant CSS dans l'éditeur de modèles.
- **Bordereau de paiement** : Si une entreprise souhaite inclure un bordereau de paiement de retour sur le PDF du relevé (pour renvoyer un chèque par courrier), le modèle peut inclure une section détachable. Bien que l'éditeur de modèles n'ait pas le concept de « ligne de coupe », les administrateurs conçoivent souvent le pied de page de manière à ce que, lorsqu'il est imprimé en recto-verso, un côté soit le relevé et l'autre un talon de remise. FreeMarker peut remplir les montants dus dans ce talon en fonction des données (en utilisant `${record.amountdue}`, etc.).

Exemples et contributions de la communauté

Plusieurs exemples communautaires soulignent les besoins en matière de relevés :

- Un utilisateur a demandé si des informations détaillées sur les factures (numéro, date, montant) pouvaient être affichées sur les relevés (Source: community.oracle.com), indiquant une telle exigence. La réponse (non visible sans connexion) impliquait probablement l'ajout de ces champs via des modifications de modèle avancé.
- Une autre publication [16] (NetSuite Community) demandait spécifiquement comment inclure une colonne *Bon de commande client* (Customer PO) sur les relevés. En effet, `${lines.otherrefnum}` est généralement utilisé pour le « n° de BC/chèque » (qui correspond au numéro de chèque dans les formulaires de comptes fournisseurs). Si un champ personnalisé est utilisé pour le bon de commande, il devra être ajouté aux lignes du relevé. Cela souligne que presque n'importe quel champ de la facture ou de la note de débit peut être intégré au relevé via FreeMarker.
- Dans les scénarios **OneWorld / multi-devises**, NetSuite fournit un modèle de « Relevé multi-devises standard » (Source: docs.oracle.com). Celui-ci permet d'afficher les soldes de plusieurs devises sur un seul relevé. Les entreprises opérant à l'international doivent tester les relevés dans toutes les devises activées. Par exemple, un client n'utilisant pas l'USD pourrait avoir besoin de champs `currencysymbol` ou `currencyname` imprimés.

Bonnes pratiques spécifiques aux relevés

- **Formatage des lignes** : Gardez les lignes du relevé concises. Trop de colonnes peuvent rendre le relevé difficile à lire sur une largeur de page réduite. Souvent, les relevés utilisent 2 ou 3 colonnes (Date, Type/Numéro, Solde). Les informations complexes (comme l'adresse de facturation, les conditions) résident généralement dans l'en-tête ou le pied de page plutôt que dans les lignes.
- **Langue et libellés** : Les relevés doivent souvent être localisés. Le modèle peut utiliser `<@translations>` ou une solution de secours, mais l'essentiel est de s'appuyer sur des libellés. Les champs de référence (par exemple « 1-30 jours ») seront traduits en fonction des paramètres de langue du client, s'ils sont configurés. Les administrateurs peuvent également remplacer ces libellés dans NetSuite (via la fonctionnalité de liste de traduction ou en personnalisant les libellés de champs sur les formulaires).
- **Tests avec des données variées** : Les relevés présentent souvent des structures de données différentes (certains clients ont de nombreuses factures, d'autres aucune). Testez les cas limites, comme un client avec une facture très ancienne, un client sans solde ouvert (qui ne devrait peut-être rien imprimer), ou un client avec uniquement des paiements.
- **E-mail de relevés clients** : Comme les factures, les relevés peuvent être envoyés par e-mail via un modèle. Les modèles HTML avancés permettent de rédiger un message électronique de qualité ; une organisation peut également utiliser le modèle HTML de facture avancée pour envoyer des relevés en le réaffectant via les paramètres d'impression. Dans les deux cas, assurez-vous que le style de l'e-mail (objet, corps) est approprié.

Détails de mise en œuvre technique

SuiteScript et automatisation

Les modèles avancés peuvent être exploités manuellement ou via l'automatisation :

- **Impression/E-mail manuel** : Les utilisateurs peuvent imprimer ou envoyer par e-mail des factures/relevés depuis l'interface utilisateur comme auparavant. Une fois qu'un formulaire est configuré pour utiliser un modèle avancé, les fonctions *Imprimer* et *E-mail* utilisent automatiquement ce modèle.
- **Génération par script** : Il arrive que les entreprises aient besoin de générer un PDF par programmation (par exemple, en tant que demande de service après l'enregistrement d'une fiche, ou via une tâche d'impression en masse planifiée). L'API SuiteScript de NetSuite le permet. Par exemple, les scripts côté serveur peuvent charger une transaction et appeler `render.transaction()` pour produire une chaîne PDF/HTML en utilisant le modèle avancé. La documentation mentionne explicitement que SuiteScript peut générer des formulaires imprimés à l'aide de modèles avancés (Source: docs.oracle.com). Cela signifie que les développeurs peuvent, par exemple, ajouter un bouton « Imprimer la facture » qui déclenche un SuiteScript pour produire le PDF de la facture à la demande, tout en bénéficiant de toutes les personnalisations du modèle avancé.
- **Workflows d'approbation** : Intégration avec les workflows : un workflow peut être configuré pour envoyer automatiquement des factures par e-mail aux clients lorsque le statut passe à « Approuvé », en utilisant le modèle PDF avancé attribué. Les workflows peuvent faire référence à des actions « Envoyer un e-mail » qui récupèrent le contenu PDF.
- **Intégration tierce** : Les systèmes externes qui appellent l'API de NetSuite (par exemple, un service de facturation ou un connecteur métier) peuvent récupérer le PDF en utilisant l'API REST ou SOAP, récupérant ainsi la sortie du modèle avancé pour les documents. Le modèle lui-même reste dans NetSuite, mais n'importe quelle interface peut extraire le PDF rendu.

Techniques de rendu de contenu

Un point plus sophistiqué : le blog de Prolecto ([14]) décrit le défi consistant à inclure des données qui ne figurent pas nativement sur la transaction. Par défaut, seuls les champs de la transaction (y compris les champs joints provenant d'enregistrements associés comme le client ou les articles) sont directement accessibles dans `${record}`. Si une entreprise a besoin, par exemple, de données provenant d'une recherche enregistrée (comme une mesure personnalisée de l'ancienneté des comptes clients ou une note système), l'approche native consiste à utiliser une source de données Suitelet JSON/XML et à la fusionner dans le modèle (Source: blog.prolecto.com). Cependant, cela nécessite SuiteScript. Les auteurs du blog ont créé un « moteur de rendu de contenu » (ne nécessitant jamais de code dans le modèle lui-même) qui alimente des données supplémentaires via SuiteQL dans le contexte `${record}` (Source: blog.prolecto.com). Par exemple, si les factures avaient besoin de données provenant d'un enregistrement personnalisé ou d'un jeu de données historique, leur moteur vous permet de définir ce jeu de données dans NetSuite, puis d'utiliser simplement `${record.extraData.fieldX}` dans le modèle. Cette technique, bien qu'elle ne soit pas intégrée nativement à NetSuite, montre comment les modèles avancés peuvent être étendus grâce à des outils ingénieux.

SuiteCloud Development Framework

En ce qui concerne le versionnage et les migrations, les modèles avancés sont compatibles avec le SuiteCloud Development Framework (SDF). Les administrateurs peuvent déployer un modèle dans le cadre d'un bundle ou directement vers un compte cible en utilisant SDF. La documentation officielle note que vous pouvez « copier des modèles vers d'autres comptes » via SDF (Source: docs.oracle.com). En pratique, on récupère le modèle via l'interface utilisateur de NetSuite ou SuiteCloud IDE, on l'inclut dans un projet SDF, puis on le publie. C'est particulièrement utile pour les clients multi-comptes (par exemple, un modèle configuré en production et poussé vers la Sandbox, ou vice versa).

Prise en charge multilingue

NetSuite prend en charge plusieurs langues par transaction. Les modèles avancés respectent automatiquement les paramètres de langue/pays du client (ou de la filiale) pour des éléments tels que la date, le formatage des nombres et le texte statique traduit. Par exemple, si la langue d'un client est le français, les champs comme `${record.trandate}` formateront les noms des mois en français si la locale est définie. Tout texte statique inséré directement dans le modèle (comme le texte d'en-tête « Facture ») peut également être traduit conditionnellement en utilisant FreeMarker conditionnel basé sur `${.locale}` ou en utilisant les ressources de fichiers de traduction de NetSuite. La meilleure pratique d'Oracle consiste à

utiliser la fonctionnalité de traduction globale pour les libellés lorsque cela est possible. Notamment, l'éditeur de modèle avancé lui-même peut changer de langue au moment de l'exécution (il existe un sélecteur de langue lors de la prévisualisation), les concepteurs doivent donc tester le modèle dans chaque langue nécessaire.

Études de cas et exemples concrets

SuiteDreams Furniture : Scénario de formation

Le matériel de formation officiel d'Oracle propose un cas fictif, **SuiteDreams Furniture Company**, pour illustrer l'utilisation des modèles avancés. Dans un exercice, SuiteDreams personnalise un modèle de facture en supprimant les champs inutilisés (Conditions, Partenaire) et en ajoutant de nouveaux (Code de fidélité, Programme de fidélité) (Source: pdfcoffee.com). Dans l'exercice suivant, ils utilisent la fonction `substring` de FreeMarker pour n'afficher que les « six caractères du milieu » de la valeur du code de fidélité sur la facture (Source: pdfcoffee.com). Ces étapes sont documentées dans le manuel de l'étudiant, soulignant comment des entreprises réelles pourraient adapter leurs factures à leurs systèmes de fidélité ou de promotion. Bien qu'il s'agisse d'une illustration, cela reflète des scénarios réels : de nombreuses entreprises ont des données de facturation personnalisées (codes VIP, identifiants de service interne, etc.) et souhaitent les voir apparaître sur les factures imprimées.

Perspective sectorielle : Avantages des factures personnalisées

Indépendamment de NetSuite, les analystes commerciaux insistent sur la personnalisation des factures. Un blog spécialisé dans les technologies financières observe que les **formats de facture génériques** peuvent dérouter les clients et ralentir les paiements, tandis que les factures personnalisées avec des conditions claires et une image de marque soignée améliorent les flux de trésorerie (Source: www.highradius.com). Plus précisément, Nimisha Ghosh note que les factures stimulent les flux de trésorerie et que « *de nombreuses entreprises s'appuient encore sur des formats de facture statiques et génériques qui créent de l'ambiguïté, augmentent les litiges et ralentissent les approbations.* » Les factures personnalisées résolvent ce problème en intégrant un contenu personnalisé et une meilleure clarté (Source: www.highradius.com). Un autre article axé sur le marketing souligne que les factures servent de point de contact clé avec les clients et doivent refléter le professionnalisme et l'image de l'entreprise (Source: apexsoft.com). Ces perspectives impliquent que l'investissement dans des modèles avancés (en particulier pour les clients à fort volume ou stratégiques) présente des avantages mesurables : moins de retards de paiement, une meilleure satisfaction client et des opportunités de vente incitative via une messagerie personnalisée.

Exemples de la communauté et des consultants NetSuite

Les consultants et les membres de la communauté NetSuite ont partagé des exemples pratiques :

- **Le blog NetSuite Pro** : Un partenaire NetSuite a publié « Exemples PDF avancés du monde réel », présentant cinq modèles réels que les clients pourraient utiliser (Source: www.thenetsuitepro.com) (Source: www.thenetsuitepro.com). Ceux-ci incluent une mise en page de facture complète, un bon de commande avec des conditions conditionnelles, un pied de page de relevé par filiale, des surbrillances de ligne conditionnelles et une section de champs personnalisés. Le code final et les captures d'écran sont disponibles, montrant comment les entreprises peuvent appliquer directement des conceptions similaires. De tels exemples servent de modèles (littéralement) pour la personnalisation sur le terrain.
- **Ressources Prolecto** : Le blog Prolecto (dirigé par Marty Zigman, un expert NetSuite largement respecté) décrit des techniques avancées. Dans un article, ils démontrent la récupération de jeux de données complexes (comme un modèle de règlement en cryptomonnaie) dans une facture via un moteur personnalisé (Source: blog.prolecto.com). Un autre article (lié au précédent) a montré comment connecter un moteur de contenu aux modèles avancés natifs sans SuiteScript, permettant aux administrateurs de définir des « univers de données » via des recherches enregistrées et de les combiner dans des factures (Source: blog.prolecto.com) (Source: blog.prolecto.com). Bien qu'il s'agisse de cas spécialisés, ils illustrent que les entreprises ayant des exigences sophistiquées (comme la facturation inter-sociétés en crypto) peuvent toujours utiliser des modèles avancés en étendant les sources de données. En termes réels, une mise à niveau pour les entreprises existantes pourrait impliquer cette approche si elles ont besoin d'afficher des données non standard (par exemple, des calendriers d'amortissement, des jalons de contrat, etc.) sur les factures.
- **Témoignages clients** : Bien que les études de cas spécifiques sur les modèles avancés soient rares dans le marketing public, d'innombrables clients NetSuite les utilisent efficacement. Par exemple, une entreprise manufacturière peut intégrer des images d'inventaire sur les factures ; une entreprise de services mondiale imprime des journaux de temps sous forme de pièces jointes PDF. Ces cas d'utilisation, bien que non publiés,

renforcent l'idée que les modèles avancés constituent la base de nombreux rapports et documents personnalisés à haute valeur ajoutée dans NetSuite.

En résumé, les exemples de formation et les connaissances du secteur confirment que les modèles PDF avancés ne sont pas seulement un luxe décoratif, mais une stratégie essentielle pour rationaliser les processus comptables, renforcer l'image de marque et se conformer aux réglementations en constante évolution. Les organisations, grandes et petites, les intègrent pour différencier leur communication et accroître leur efficacité.

Implications et orientations futures

Implications commerciales et opérationnelles

Le passage aux modèles PDF avancés a des implications commerciales significatives :

- **Workflows rationalisés** : En consolidant la conception et la logique dans l'ERP, les entreprises réduisent le travail manuel. Par exemple, auparavant, une équipe pouvait imprimer des factures dans Word ou Adobe séparément ; maintenant, tout est dans NetSuite, ce qui réduit les erreurs humaines. Une meilleure clarté sur les factures et les relevés (par exemple, en mettant les conditions en gras, en surlignant les montants en retard) peut raccourcir les cycles des comptes clients.
- **Cohérence de la marque et de la conformité** : L'utilisation de modèles garantit que chaque facture/relevé respecte une norme uniforme. Les changements d'image de marque (nouveau logo, changement de nom) ou les détails de conformité (comme les numéros d'identification fiscale mis à jour) peuvent être déployés en un seul endroit. C'est plus efficace que de modifier plusieurs formulaires ou d'envoyer des feuilles de calcul par e-mail.
- **Collaboration interfonctionnelle** : Les équipes financières peuvent effectuer certaines personnalisations sans cycles de développement complets. L'éditeur WYSIWYG permet aux utilisateurs moins techniques d'ajuster les libellés, tandis que la logique complexe reste entre les mains de l'équipe informatique/ERP. Cette collaboration peut libérer des heures de développement pour d'autres projets.
- **Messagerie basée sur les données** : Les modèles avancés s'intègrent étroitement aux données transactionnelles. Les entreprises peuvent les utiliser pour intégrer des messages personnalisés (par exemple, en faisant référence au statut de fidélité, à l'activité passée du compte, etc.) afin d'améliorer les relations avec les clients directement dans les documents de facturation.
- **Coûts de transition** : En revanche, le passage de mises en page de base (ou sans personnalisation) à des modèles avancés implique une courbe d'apprentissage. Le personnel a besoin d'une formation sur FreeMarker et HTML. L'investissement initial dans la conception de modèles peut être non négligeable, en particulier pour les mises en page complexes. Cependant, le retour sur investissement à long terme (grâce au gain de temps et à une meilleure trésorerie) le justifie généralement.

Orientations futures et feuille de route NetSuite

Pour l'avenir, plusieurs facteurs influenceront l'évolution des modèles PDF/HTML avancés dans NetSuite :

- **Migration complète** : Les déclarations d'Oracle suggèrent que *tous les clients* finiront par utiliser des modèles avancés. NetSuite pourrait introduire des outils ou des rapports pour identifier quels formulaires utilisent encore des mises en page de base, encourageant ainsi la migration. Cette dynamique signifie que l'investissement actuel dans les modèles avancés évite des problèmes futurs.
- **Fonctionnalités d'éditeur améliorées** : L'éditeur de modèles pourrait ajouter davantage de fonctionnalités WYSIWYG au fil du temps (par exemple, une meilleure équivalence avec la source, plus de widgets). La mention de l'alignement sur les « normes industrielles pour l'édition HTML » (Source: docs.oracle.com) suggère qu'Oracle est conscient que la cohérence et la facilité d'utilisation sont importantes.
- **Intégration de l'IA** : NetSuite ajoute activement des capacités d'IA. Depuis 2024, des fonctionnalités telles que *Text Enhance* et la capture de factures pilotée par l'IA existent (Source: suitecentric.com). À moyen terme, l'IA pourrait aider à la création de modèles. Par exemple, un outil d'IA pourrait suggérer des emplacements de libellés ou même générer automatiquement des mises en page par défaut basées sur les meilleures pratiques. L'IA pourrait potentiellement analyser les factures passées pour identifier les champs manquants ou recommander des améliorations de formatage. De plus, comme le traitement des factures (OCR via Bill Capture) est piloté par l'IA, une intégration plus étroite pourrait émerger où la même IA qui lit les factures pourrait vérifier l'exactitude des modèles générés par rapport aux données d'entrée.

- **Normes de documents électroniques** : Avec la croissance des mandats mondiaux de facturation électronique (par exemple, PEPPOL dans l'UE, SingPass à Singapour, etc.), les futures versions des modèles avancés de NetSuite pourraient inclure une prise en charge dédiée de ces normes de sortie (comme l'ajout de transformations XSLT ou de champs QR/code-barres). L'aide d'Oracle mentionne l'ajout d'éléments de code QR (non explorés en détail ci-dessus), suggérant de futures fonctions de modèle (par exemple, une fonction pour générer des codes QR pour les factures QR suisses ou d'autres critères de normes).
- **Collaboration dans le cloud** : Dans le cadre de la suite Oracle Cloud, NetSuite pourrait améliorer la façon dont les modèles sont stockés et partagés. Imaginez une place de marché ou un référentiel de modèles (au-delà des bundles SuiteApp) où des modèles de facture prêts à l'emploi seraient disponibles pour les secteurs courants. Le partage de snippets FreeMarker de style SuiteAnswers ou GitHub pourrait devenir plus courant.
- **Types de documents étendus** : Bien que ce rapport se concentre sur les modèles de facture et de relevé, la même technologie sous-tend tous les formulaires de transaction et même les exportations de recherches enregistrées (type CSV/Excel). À l'avenir, Oracle pourrait étendre l'utilisation des modèles avancés à d'autres domaines (par exemple, PDF de devis, relevés de commissions, rapports personnalisés).

En substance, les modèles PDF/HTML avancés sont une fonctionnalité « vivante » de NetSuite. Ils continueront d'évoluer à mesure que les normes Web et les besoins des entreprises changeront. Les organisations doivent les adopter tout en restant flexibles : les modèles qui répondent aux besoins d'aujourd'hui pourraient être améliorés par de nouvelles fonctionnalités NetSuite ou des outils tiers demain.

Conclusion

Les **modèles PDF/HTML avancés** de NetSuite offrent un cadre moderne et puissant pour personnaliser les factures et les relevés. En permettant des documents entièrement stylisés et riches en données avec une logique conditionnelle et des mises en page complexes, ils représentent une mise à niveau significative par rapport aux méthodes d'impression héritées. Ce rapport a détaillé comment activer la fonctionnalité, lier les modèles aux formulaires et créer des modèles de facture et de relevé avancés. Nous avons examiné la riche disponibilité des champs (y compris les tranches d'ancienneté et tous les champs de transaction typiques), l'intégration avec des éléments de conception comme les logos et le CSS, et la capacité d'inclure des données commerciales personnalisées via FreeMarker et des outils externes.

Nous avons passé en revue les meilleures pratiques et les conseils de la communauté – par exemple, toujours développer dans une sandbox et cloner les modèles pour préserver les originaux, utiliser du CSS en ligne et tester progressivement chaque section. Nous avons présenté des cas réels : du scénario SuiteDreams d'Oracle sur l'ajout de données de fidélité, aux exemples de code de blog de consultants pour la mise en évidence de lignes à haute valeur. Nous avons également abordé la façon dont les modèles avancés s'intègrent dans des stratégies plus larges : ils s'alignent sur l'automatisation des comptes clients (via l'IA et les tendances de facturation électronique) et reflètent les recommandations d'experts financiers sur la valeur des documents de facturation personnalisés (Source: www.highradius.com) (Source: apexactsoft.com).

À l'avenir, les entreprises devraient donner la priorité à la migration vers les modèles avancés (si ce n'est pas déjà fait), étant donné l'abandon progressif des mises en page de base (Source: docs.oracle.com) (Source: community.oracle.com). Bien qu'il existe un coût initial en termes d'apprentissage et de développement, les gains à long terme en efficacité, en conformité et en cohérence de marque sont substantiels. De plus, à mesure que NetSuite continue d'innover (intégration de l'IA, conformité des documents électroniques), les modèles avancés resteront la base de tous les besoins en matière d'impression ou de sortie PDF. Les organisations qui maîtrisent cette fonctionnalité dès aujourd'hui seront parfaitement équipées pour tirer parti des prochaines améliorations de NetSuite et des évolutions du secteur.

En résumé, les modèles PDF/HTML avancés unifient le design et les données au sein de l'ERP cloud de NetSuite, permettant de créer des factures et des relevés personnalisés qui soutiennent les résultats stratégiques de l'entreprise. Les solutions Kenchi devraient les considérer comme un élément central de leur utilisation de NetSuite — un point largement étayé tant par la documentation d'Oracle (Source: docs.oracle.com) (Source: docs.oracle.com) que par des recherches externes (Source: www.highradius.com) (Source: pdfcoffee.com). Toutes les affirmations ci-dessus sont étayées par des preuves provenant de sources officielles, de communautés et de blogs d'experts, garantissant que ce rapport constitue un guide faisant autorité en matière de conception pédagogique et de mise en œuvre pour la personnalisation des modèles PDF/HTML avancés dans NetSuite.

Étiquettes: pdf-avance-netsuite, modeles-html, personnalisation-facture, scripting-freemarker, suitecloud, personnalisation-releve, administration-netsuite

AVERTISSEMENT

Ce document est fourni à titre informatif uniquement. Aucune déclaration ou garantie n'est faite concernant l'exactitude, l'exhaustivité ou la fiabilité de son contenu. Toute utilisation de ces informations est à vos propres risques. Houseblend ne sera pas responsable des dommages découlant de l'utilisation de ce document. Ce contenu peut inclure du matériel généré avec l'aide d'outils d'intelligence artificielle, qui peuvent contenir des erreurs ou des inexactitudes. Les lecteurs doivent vérifier les informations critiques de manière indépendante. Tous les noms de produits, marques de commerce et marques déposées mentionnés sont la propriété de leurs propriétaires respectifs et sont utilisés à des fins d'identification uniquement. L'utilisation de ces noms n'implique pas l'approbation. Ce document ne constitue pas un conseil professionnel ou juridique. Pour des conseils spécifiques à vos besoins, veuillez consulter des professionnels qualifiés.