

Gestion des changements de conception dans les installations SBM d'entreprise

Table des matières

page

Résumé	1
SBM : conçu pour faciliter l'adaptation et l'intégration	1
Développement d'applications de processus SBM	2
Responsabilités et personnel requis pour tirer parti de SBM	4
Coordination des efforts de développement itératifs et de long terme	8
Messages à retenir	14

SBM est un produit de gestion des tâches basée sur les processus qui permet de piloter, suivre et surveiller le travail accompli dans l'entreprise.

Résumé

Ce document explique comment organiser votre processus de maintenance et de développement OpenText™ Solutions Business Manager (SBM) pour faciliter la maintenance à long terme des applications de processus en fonction de l'évolution de vos besoins métiers.

SBM : conçu pour faciliter l'adaptation et l'intégration

SBM est un produit de gestion des tâches basée sur les processus qui permet de piloter, suivre et surveiller le travail accompli dans l'entreprise. Bien qu'il soit envisageable de confiner son utilisation à certains services de l'entreprise, nos clients les plus performants ont exploité la possibilité de l'étendre à l'ensemble de leur organisation pour optimiser la structure et la visibilité d'un plus large éventail de processus métiers. À titre d'exemple, nombre de services informatiques ont recours à SBM pour gérer leur service d'assistance. Les utilisateurs de l'entreprise passent par le logiciel OpenText™ Request Center pour soumettre leurs tickets de support, et SBM gère l'acheminement de chaque ticket à travers les processus de traitement du service approprié. Les demandes concernant l'intégration des nouveaux employés, les avantages sociaux et les congés sont acheminées vers les Ressources humaines ; le service informatique reçoit les demandes relatives à la messagerie électronique ; et les problèmes d'infrastructure et les rapports sur les dépenses sont soumis au service financier. Les fonctionnalités de création de rapports et d'audit intégrées à SBM permettent aux responsables concernés de superviser ces processus afin d'appréhender les grandes tendances, mais aussi de forer jusqu'au niveau de détail voulu pour analyser les caractéristiques d'une transaction ou d'un ticket donnés.

Nous avons axé la conception de SBM sur un ensemble de principes de base répondant aux besoins d'organisations clientes de plus en plus dynamiques.

Le changement est une donnée fondamentale

Le changement est inhérent à l'activité de l'entreprise. Face à l'évolution incessante des conditions de marché, des objectifs stratégiques et des exigences métiers, les entreprises gagnantes sont celles qui s'adaptent en permanence à ces changements. L'automatisation de ces processus métiers doit répondre à un double impératif d'accompagnement du changement et d'amélioration continue fondée sur la découverte de nouvelles optimisations.

SBM offre une adaptation rapide à l'émergence et à l'évolution des exigences utilisateur. Les modifications de processus s'effectuent au sein d'interfaces optimisées pour le glisser-déplacer, conçues pour faciliter l'accès de non-développeurs connaissant intimement les processus métiers à gérer.

Notre objectif est de donner aux collaborateurs qui interagissent au quotidien avec les processus gérés la possibilité de les adapter et de les ajuster au fur et à mesure de l'émergence de nouvelles exigences ou de la découverte d'inefficacités dans la prise en charge des exigences existantes. En simplifiant la modification des processus, SBM permet d'adapter ces derniers au renouvellement et à la modification des exigences métiers plutôt que d'imposer des processus rigides aux équipes métiers.

Les systèmes participent eux aussi aux processus

Le paysage informatique fait intervenir une grande variété d'outils et de systèmes. Aussi, le logiciel d'automatisation des processus doit-il non seulement coexister, mais aussi collaborer avec les systèmes actuels. Les autres systèmes d'enregistrement participent aux processus au même titre que les utilisateurs. Le concepteur de processus est à même d'identifier les points de communication entre les systèmes afin de les intégrer au plus haut niveau à la conception des processus. SBM est conçu pour permettre de telles intégrations. Par exemple, vous pouvez tirer parti de la prise en charge des services Web REST ou SOAP (Simple Object Access Protocol), des workflows d'orchestration, des modules AppScript et même de notre API C++ pour assurer la liaison entre SBM et les autres systèmes constituant l'environnement informatique.

Après l'acquisition d'une solution basée sur la plate-forme SBM d'OpenText, votre premier objectif peut être d'adapter les applications de processus existantes aux spécificités de votre environnement métier.

Développement d'applications de processus SBM

Dans l'idéal, le développement d'applications de processus doit s'effectuer de manière itérative, en étroite collaboration avec un propriétaire métier ayant une bonne connaissance du processus à automatiser. Le propriétaire métier connaît les acteurs du processus et le rôle qui leur est affecté, mais aussi les données à recueillir et les rapports à fournir aux superviseurs du processus. Il est également au fait des objectifs de l'effort de développement, et peut guider celui-ci de manière à assurer la réalisation des buts poursuivis en matière de productivité, de revenus, de qualité, de réduction des risques et de visibilité.

Après l'acquisition d'une solution basée sur la plate-forme SBM d'OpenText™, votre premier objectif peut être d'adapter les applications de processus existantes aux spécificités de votre environnement métier. Pour cela, il suffit souvent d'apporter des modifications simples au modèle de données (c'est-à-dire aux informations recueillies au cours du processus), ainsi qu'au diagramme de workflow, en fonction des différences relevées par rapport aux états et aux flux du processus.

OpenText vous recommande de désigner un propriétaire de processus, chargé de recueillir les retours d'expérience des participants, au processus SBM et de veiller à ce que ce processus évolue en tenant compte de ces commentaires.

Développement de nouvelles applications de processus

À la différence des déploiements entièrement personnalisés, SBM vous permet de commencer par un processus simple destiné à guider les participants dans leurs tâches, puis d'améliorer celui-ci par itérations successives au fil du temps.

Au cours du développement, le concepteur de processus définit visuellement les éléments suivants :

1. Les données à recueillir, présenter ou manipuler durant le processus.
2. Les workflows du processus, qui reflètent la progression des tâches au sein de l'organisation.
3. Les rôles et responsabilités des participants au processus.
4. Les interfaces utilisateur qui seront présentées aux participants au processus.
5. Les rapports destinés à offrir la visibilité requise aux personnes responsables du monitoring de l'état de santé du processus.

Une fois que l'application de processus a atteint un niveau de maturité suffisant et validé par le propriétaire métier, vous pouvez la déployer auprès des utilisateurs.

Affinement post-déploiement des applications de processus existantes

Avec SBM, le déploiement initial n'est pas une fin en soi, mais le point de départ d'un cycle d'amélioration continue. OpenText vous recommande de désigner un propriétaire de processus, chargé de recueillir les retours d'expérience des participants, au processus SBM et de veiller à ce que ce processus évolue en tenant compte de ces commentaires.

Certains clients ont recours à un forum, dont les membres rencontrent périodiquement les représentants des participants au processus pour assurer la collecte et la prise en compte des commentaires. Durant cette phase, vous pouvez vous appuyer sur les commentaires pour améliorer les formulaires personnalisés, automatiser certains comportements de l'interface utilisateur au moyen d'actions de formulaire, accéder à des données externes via la grille REST et fournir des rapports intégrés pour l'accès aux données associées. L'émergence de nouveaux besoins de création de rapports vous permet d'enrichir progressivement les rapports mis à la disposition des consommateurs des données concernées. Ces différents types de modifications ne requièrent pas de compétences techniques approfondies ni de connaissances poussées en programmation. Les seuls prérequis sont la maîtrise de l'outil SBM Composer et la connaissance du processus à modéliser.

Par exemple, des commentaires vous signalent que les utilisateurs sont obligés de faire défiler un formulaire de transition jusqu'en bas pour accéder à un champ obligatoire. Même si le déploiement SBM répond aux exigences fonctionnelles liées à la gestion du processus, l'inefficacité ainsi détectée peut et doit être corrigée sans délai. Vous pouvez corriger ce défaut en quelques minutes, en faisant glisser le champ concerné vers le haut du formulaire à l'aide d'OpenText™ SBM Composer.

Responsabilités et personnel requis pour tirer parti de SBM

Pour exploiter pleinement les avantages de SBM, votre installation doit être dotée d'un personnel adéquat, affecté à bon escient. La compétence clé pour accélérer les améliorations itératives d'un processus est une bonne connaissance du processus métier à automatiser. Des cycles de changement trop longs réduiraient en effet la capacité du processus à évoluer au rythme de l'activité. En revanche, pour les intégrations et les aspects plus techniques de la création et de la maintenance des systèmes SBM, il convient de définir expressément les exigences en amont du développement, dont la mise en oeuvre peut alors s'effectuer selon un cycle plus long et sans proximité immédiate avec les aspects métiers. La présente section décrit les diverses responsabilités intervenant dans la mise en oeuvre et la gestion d'un système SBM.*

Pour exploiter pleinement les avantages de SBM, votre installation doit être dotée d'un personnel adéquat, affecté à bon escient.

Installation et configuration système

L'exécution des tâches d'installation, de mise à niveau et de configuration système liées à SMB incombe en principe aux administrateurs système. Ces responsabilités sont souvent assumées par les membres de l'équipe informatique chargés de la maintenance de l'infrastructure physique et logicielle de l'entreprise.

Administration de SBM

À la différence des administrateurs système, l'administrateur SBM gère le fonctionnement des processus SBM au quotidien. Un administrateur SBM assume la gestion des utilisateurs et groupes d'utilisateurs SBM, des privilèges, des notifications, des projets et des dérogations aux workflows administratifs à l'aide de l'outil SBM Application Administrator. Bien que les utilisateurs aient la possibilité de configurer des rapports adaptés à leurs besoins spécifiques, les administrateurs définissent généralement une série de rapports répondant aux besoins des participants aux processus et des observateurs. Un administrateur doit avoir une bonne connaissance des processus métiers tels qu'ils sont mis en oeuvre dans SBM, ainsi que des besoins des utilisateurs SBM.

Maintenance des conceptions de processus SBM

Étroitement corrélées à l'administration SBM, ces responsabilités supplémentaires consistent à apporter des modifications incrémentielles à la conception de vos processus via SBM Composer. À travers sa participation au forum de la solution, un administrateur ou un concepteur SBM a la possibilité de collaborer avec les utilisateurs et le propriétaire d'un processus, de détecter toute évolution des exigences et d'implémenter rapidement les modifications les plus pertinentes pour les participants à ce processus. La désignation d'un intervenant proche du processus et capable d'effectuer rapidement des modifications de conception itératives permet de valoriser les fonctionnalités SBM de « création axée sur le changement ».

**Selon la taille et la complexité du système SBM, un même rôle peut être tenu par plusieurs personnes et une même personne peut assumer plusieurs rôles, ou encore cumuler un rôle SBM avec des responsabilités hors SBM.*

Chaque application déployée sur la plate-forme SBM est dotée d'une table principale définissant les informations recueillies dans un élément tout au long de son cycle de vie au sein du processus. L'émergence de nouvelles exigences amène généralement le concepteur du processus à modifier le modèle de données en ajoutant des champs à cette table principale pour recueillir des informations supplémentaires.

Les sections qui suivent présentent les différents types de tâches de conception effectuées dans SBM Composer, ainsi que les niveaux de compétences correspondants requis.

CONCEPTION DE WORKFLOWS

Le workflow définit le parcours d'un élément impliqué dans un processus au sein d'une organisation. Au niveau le plus général, un workflow consiste en un ensemble de jalons (états) ainsi que de transitions sanctionnant la progression de cet élément d'un état à un autre. Chaque état actif relève d'un propriétaire qui assume la responsabilité de l'élément à un moment donné. Un utilisateur peut activer la transition des éléments d'un état à un autre de façon à en transmettre la responsabilité à un autre utilisateur. La définition de noeuds de décision permet de recourir à des règles métiers pour déterminer le mode d'acheminement d'un élément en fonction des informations recueillies. Les diverses propriétés affectées au workflow déterminent l'accès aux transitions et la visibilité des données en fonction du rôle de l'utilisateur qui consulte un élément. Un concepteur de processus peut modifier un workflow de différentes manières : en ajoutant ou en supprimant des états ou des transitions ; en modifiant les règles métiers régissant le comportement des décisions ; ou en ajustant la visibilité des données au niveau des workflows, des états ou des transitions. Ces types de modifications, qui se produisent couramment à mesure que les processus évoluent dans le temps, ne requièrent qu'une connaissance élémentaire des processus métiers et de l'outil SBM Composer.

CONCEPTION DES DONNÉES

Chaque application déployée sur la plate-forme SBM est dotée d'une *table principale* définissant les informations recueillies dans un élément tout au long de son cycle de vie au sein du processus. L'émergence de nouvelles exigences amène généralement le concepteur du processus à modifier le modèle de données en ajoutant des champs à cette table principale pour recueillir des informations supplémentaires. Par exemple, si un concepteur de processus a besoin d'ajouter un champ *Urgence* au modèle de données, il doit faire glisser un champ Sélection (Sélection) de la palette Table jusqu'à la table principale, puis nommer ce champ « Urgence » et ajouter les valeurs de sélection « Faible », « Moyenne », « Élevée » et « Critique » aux propriétés du champ. Pour insérer ce nouveau champ dans des formulaires personnalisés, il suffit alors de spécifier un champ associé et une position relative via la fonction *Add to Forms...* (Ajouter aux formulaires) de l'onglet Forms (Formulaires) du champ. À l'instar des modifications du workflow, les changements de conception des données sont relativement simples et ne requièrent pas de compétences techniques ni de connaissances particulières en programmation.

CONCEPTION VISUELLE

SBM génère automatiquement des « formulaires rapides » pour les états et les transitions pour lesquels vous n'avez pas défini de formulaire personnalisé. L'utilisation de formulaires rapides permet d'exercer un contrôle limité sur la disposition des champs dans le formulaire au moyen de propriétés de champ telles que **Span entire row on forms** (Utiliser une ligne entière dans les formulaires) ou **Privilege section** (Section Privilèges). Vous pouvez contrôler l'ordre d'apparition des champs en modifiant l'ordre des privilèges de champ dans les propriétés de workflow, d'état ou de transition, et celui des boutons de transition en modifiant les propriétés des transitions au niveau des états.

Positionnement de la solution

Gestion des changements de conception dans les installations SBM d'entreprise

Si les formulaires rapides sont utiles pour prototyper un système ou gérer des états ou transitions rarement utilisés, les formulaires personnalisés permettent quant à eux de présenter les données de façon beaucoup plus souple, de définir le comportement actif au sein d'un formulaire et d'effectuer des intégrations par simple inclusion de rapports, de cadres HTML et de services Web REST. Les formulaires personnalisés vous permettent d'optimiser les formulaires de consultation ou de saisie de données en fonction des besoins de l'utilisateur final.

La définition de formulaires personnalisés est on ne peut plus simple. L'éditeur visuel de formulaires OpenText™ SBM Composer vous permet d'ajouter et de déplacer des champs par simple glisser-déplacer. Vous pouvez élaborer votre formulaire personnalisé en partant du formulaire rapide, ou concevoir celui-ci à partir de zéro en ajoutant des champs, des conteneurs, des boutons d'action, des images, des liens, des rapports, des grilles de données REST, etc. L'utilisation de formulaires personnalisés vous permet de contrôler parfaitement la position des boutons de transition sur le formulaire, de même que tous les aspects liés aux actions et à l'affichage des barres d'outils et des en-têtes.

La phase suivante du processus de conception visuelle consiste à ajouter des comportements aux formulaires. Par exemple, pour activer la sélection automatique de l'onglet de détails en fonction du niveau de gravité, vous pouvez introduire une action de formulaire structurée comme suit :

Quand

ce formulaire est chargé

Ou que la valeur du champ [Gravité](#) est modifiée

Si

la valeur du champ [Gravité](#) est *Urgente*

Alors

activer l'onglet [Détails](#)

Sinon

activer l'onglet [Standard](#)

Nous mettons à votre disposition une palette d'actions riche et diversifiée pour la création de formulaires dynamiques. Celle-ci couvre la plupart des événements, conditions et actions dont vous pouvez avoir besoin dans un formulaire. Si la condition dont vous avez besoin n'est pas proposée, vous pouvez faire référence à des règles métiers. Si cela ne suffit pas, vous avez aussi la possibilité d'écrire un extrait de code JavaScript pour mettre en oeuvre des conditions plus complexes. De même, si l'action que vous souhaitez exécuter n'est pas disponible, vous pouvez écrire un extrait de code JavaScript déterminant l'exécution de cette action lorsqu'un événement se produit et que la condition correspondante est satisfaite.

La définition de formulaires personnalisés est on ne peut plus simple. L'éditeur visuel de formulaires SBM Composer vous permet d'ajouter et de déplacer des champs par simple glisser-déplacer.

Après un premier déploiement SBM limité à un système départemental, nos clients les plus performants s'enhardissent à créer de nouvelles applications couvrant un large éventail de processus au sein de leur organisation.

L'essentiel du travail relatif aux formulaires personnalisés est à la portée d'un concepteur de processus ne possédant pas de compétences spécifiques ni de connaissances en programmation. Vous pouvez toujours faire appel à un concepteur visuel pour optimiser l'expérience graphique proposée aux utilisateurs, mais cela s'avère rarement nécessaire.

Création de conceptions de processus SBM

La création d'applications de processus s'apparente aux activités de maintenance décrites ci-dessus, à ceci près qu'au lieu de raisonner en termes de modifications incrémentielles d'un processus existant, le concepteur et le propriétaire du processus collaborent pour imaginer et concevoir l'application de processus dans son ensemble. Après un premier déploiement SBM limité à un système départemental, nos clients les plus performants s'enhardissent à créer de nouvelles applications couvrant un large éventail de processus au sein de leur organisation. Ils commencent par découvrir un processus manuel susceptible de bénéficier des capacités SBM d'automatisation, de contrôle, d'audit et de création de rapports. En collaboration avec le propriétaire de ce processus, ils en définissent alors les étapes et les transitions, les données à recueillir, ainsi que les rôles et les participants. Ils font ensuite appel à SBM pour créer rapidement des applications de processus permettant de résoudre ce problème métier. L'essentiel du travail s'effectue à l'aide des éditeurs d'application de l'outil SBM Composer, qui leur propose des métaphores familières et des interfaces simples d'utilisation pour la définition du processus. Les concepteurs de processus peuvent ainsi concevoir le modèle de données dans l'éditeur de table, définir le workflow à l'aide de l'éditeur de workflow et assigner les rôles de façon à disposer rapidement d'une version automatisée du processus qui soit opérationnelle dans un environnement de développement. Il ne leur reste alors qu'à améliorer la conception de manière itérative, jusqu'à disposer d'un produit minimal viable pouvant être déployé auprès des utilisateurs.

La responsabilité de la création et de la maintenance de la conception de processus peut être confiée à un analyste métier possédant une bonne connaissance de l'outil SBM Composer. Un concepteur de processus n'a pas nécessairement reçu une formation d'ingénieur logiciel ou de développeur, mais il doit posséder les compétences requises pour exploiter des outils de productivité d'entreprise tels que Microsoft Excel et PowerPoint.

Conception et mise en oeuvre d'intégrations

À la différence de la création et de l'affinement des processus, la mise en place d'intégrations à des systèmes externes représente souvent une tâche laborieuse impliquant une planification détaillée, une mise en oeuvre complexe et des tests approfondis. Le degré de complexité dépend du type d'intégration et de la nature des produits tiers concernés. Dans les cas les plus élémentaires, l'intégration peut se résumer à l'inclusion d'une grille REST relativement simple dans un formulaire personnalisé, ce qui requiert uniquement la connaissance d'une interface de services Web REST tierce et une bonne maîtrise des fonctions de création de formulaires personnalisés de SBM Composer. Lorsque les systèmes externes prennent en charge les services Web SOAP à un niveau de granularité correspondant à la logique métier, la mise en oeuvre de l'intégration peut impliquer une tâche plus difficile de développement d'orchestrations SBM. Enfin, dans les cas les plus complexes, un travail de programmation peut s'avérer nécessaire pour accéder aux données du système externe.

En dépit des efforts d'OpenText pour simplifier autant que possible la création d'intégrations à des systèmes externes, il s'agit souvent de la tâche la plus complexe à réaliser pour les clients SBM. Si le système à intégrer propose de riches interfaces de services Web SOAP et REST de niveau métier, ces intégrations peuvent s'effectuer sans programmation, par simple ajout d'orchestrations SBM et d'intégrations REST dans les formulaires. Cependant, même dans ce cas, la complexité du travail à accomplir peut dépasser les compétences normales d'un analyste métier. Bien que les compétences exigées d'un développeur d'intégration dépendent des tâches à effectuer, elles s'apparentent plutôt à celles d'un ingénieur logiciel.

Dans la mesure où les intégrations représentent un effort plus conséquent, de plus long terme et impliquant une planification plus détaillée, ce travail de développement n'a pas besoin d'être aussi étroitement associé à l'activité courante que les changements incrémentiels de processus décrits plus haut. Des fonctions SBM telles que les orchestrations indépendantes de l'application permettent d'isoler ces efforts de développement et d'attendre qu'ils soient achevés pour les distribuer au développeur de processus pour consommation.

Les seules compétences requises sont la compréhension du processus métier et la proximité avec celui-ci, ainsi qu'une bonne maîtrise des outils Composer et Application Administrator de SBM.

Coordination des efforts de développement itératifs et de long terme

Comme nous l'avons vu, le développement avec SBM englobe deux types de tâches nettement différenciés. Le plus gros du travail consiste en des tâches d'administration et de conception visuelle n'exigeant pas d'expertise technique approfondie ni de connaissances en programmation. Les seules compétences requises sont la compréhension du processus métier et la proximité avec celui-ci, ainsi qu'une bonne maîtrise des outils Composer et Application Administrator de SBM. En revanche, la gestion des orchestrations et des intégrations suppose un tout autre niveau de compétence technique. Il s'agit en effet d'un travail plus complexe et, en règle générale, plus suivi et plus délibéré que l'effort de maintenance itérative. Dès lors, comment s'y prendre pour coordonner les changements rapides, itératifs et adaptatifs avec les processus SBM impliquant un travail de plus long terme ? La présente section envisage différentes réponses possibles à cette problématique en décrivant les fonctions SBM correspondant à chaque approche.

Si la stabilisation des interfaces utilisées pour le travail de long terme est susceptible de simplifier l'intégration des développements résultants, la priorité doit être donnée à la fluidité des processus SBM et à leur évolution dynamique suivant les besoins des utilisateurs et de l'activité.

Changements itératifs

En règle générale, les changements itératifs ne requièrent pas de coordination détaillée avec le travail de développement de plus long terme, dès lors que ce dernier a été clairement défini et correctement encapsulé comme expliqué plus avant. Par exemple, si les orchestrations en cours de développement s'appuient sur certains aspects d'une application actuelle, tels que la présence de champs, d'états et de transitions déterminés, il importe que ces derniers ne soient pas modifiés lors de l'exécution des changements itératifs. Cependant, si la stabilisation des interfaces utilisées pour le travail de long terme est susceptible de simplifier l'intégration des développements résultants, la priorité doit être donnée à la fluidité des processus SBM et à leur évolution dynamique en fonction des besoins des utilisateurs et de l'activité. Si ces efforts de long terme ont été correctement encapsulés, leur introduction dans le processus restera simple et rapide.

Pour une description détaillée du processus type des changements itératifs, consultez le document « Mise en production de SBM dans les entreprises ».

Efforts de développement de long terme

Au moment d'entreprendre un travail de développement de long terme dans SBM, votre objectif central doit être de structurer les efforts de telle sorte qu'ils s'articulent avec les applications de processus associées via des interfaces simples et clairement définies. Il importe d'encapsuler la complexité plutôt que de l'exposer. Ces efforts de développement peuvent se poursuivre aussi bien à l'intérieur qu'à l'extérieur de SBM, et dépassent généralement les compétences ou le degré d'aisance d'un concepteur de processus. Les développeurs qui participent à ce travail doivent disposer d'une instance de développement d'un système SBM qui soit indépendante du système SBM principal utilisé pour le développement itératif, la préproduction et la production.

DÉVELOPPEMENT RÉALISÉ À L'EXTÉRIEUR DE SBM

Ce type de développement comprend, par exemple, la création d'un service REST permettant d'exposer des données issues d'un logiciel sectoriel tiers sous une forme compatible avec leur affichage dans une grille de données REST insérée dans un formulaire personnalisé SBM. Selon les cas, cette tâche peut impliquer l'écriture d'un module de code assurant l'interface avec une API du système externe, ainsi que l'hébergement du service. Elle peut également nécessiter la création et l'hébergement de services Web SOAP destinés à exposer les fonctionnalités d'un système externe, avec une granularité métier compatible avec leur utilisation dans une orchestration SBM ou en tant qu'action de service Web associée à une transition. Un système SBM de développement est indispensable pour assurer la compatibilité des interfaces en cours de développement avec les exigences SBM. Dans les exemples qui précèdent, cela suppose d'utiliser SBM pour tester les services Web REST et SOAP en cours de développement. Par nature, ce type de travail de développement est encapsulé dans la mesure où il présente à SBM une interface de haut niveau clairement définie.

DÉVELOPPEMENT RÉALISÉ À L'INTÉRIEUR DE SBM

Cette catégorie de développement implique l'utilisation des diverses technologies d'intégration plus techniques offertes par SBM. Un exemple en est le développement d'orchestrations dotées de mécanismes appropriés de traitement des erreurs et de compensation. Le développement de scripts complexes ou de logiciels appelant l'API C de SBM relève également de cette catégorie. Ces efforts s'effectuent souvent en conjonction avec des développements réalisés à l'extérieur de SBM. C'est notamment le cas lorsqu'un système externe n'expose qu'une API de très bas niveau. Il convient alors d'encapsuler cette API dans des services Web de niveau métier directement consommables par les orchestrations SBM.

TECHNIQUES D'ENCAPSULATION DES DÉVELOPPEMENTS SBM

Les actions associées aux transitions dans le workflow des applications fournissent l'essentiel des points d'interface permettant d'interagir avec les efforts de développement encapsulés. Par exemple, les workflows synchrones ou asynchrones, les modules AppScript et les services Web sont tous déclenchés à partir d'actions associées à une transition. L'autre grand point d'intégration réside dans les intégrations juste-à-temps « à même l'écran » réalisées sur les formulaires présentés aux utilisateurs. Les sections qui suivent décrivent les différents types d'intégrations, et présentent les pratiques de développement qui garantissent l'isolation et l'encapsulation requises pour faciliter l'intégration subséquente au système SBM principal.

Considérations d'ordre général sur les noms internes : nombre des techniques d'intégration envisagées ci-après font appel aux services Web, aux modules AppScript ou à l'API C SBM pour interagir avec les workflows des applications SBM. Les libellés de certains champs étant susceptibles d'évoluer, tous les services Web qui interagissent avec des champs doivent utiliser les noms enregistrés dans la base de données de champs ; ces noms ne peuvent pas être modifiés après le déploiement initial. Toute référence à un workflow, à un état ou à une transition doit utiliser le nom interne complet de cet élément, dans la mesure où sa propriété de nom peut se révéler ambiguë, ou peut être redéfinie lorsque l'application est modifiée à la suite d'une évolution des besoins.

Orchestrations : quatre types d'orchestrations sont envisagés dans cette section. Lorsque vous développez une orchestration dans le cadre d'un développement de long terme, veillez si possible à isoler cet effort dans une application de processus distincte afin de faciliter son intégration ultérieure. Les différents types d'orchestrations sont présentés ci-après.

- **Workflows d'orchestration synchrones :** ces workflows d'orchestration sont appelés depuis une application SBM par une action de transition. Les données des éléments sont assignées aux entrées du workflow d'orchestration et mises à jour par les sorties de ce même workflow. Une orchestration synchrone doit se trouver dans la même application de processus que l'application qui l'appelle. C'est pourquoi il convient, une fois l'orchestration développée, de fusionner celle-ci dans l'application de processus présente sur le système principal via la fonction Compare/Merge (Comparer/Fusionner) de SBM Composer. La spécification du workflow d'une orchestration synchrone doit définir les entrées et les sorties de cette orchestration, et tous ces champs doivent rester inchangés tout au long de l'effort de développement.

On distingue quatre types d'orchestrations SBM :

- les workflows d'orchestration synchrones ;
- les workflows d'orchestration asynchrones dépendants de l'application ;
- les workflows d'orchestration asynchrones indépendants de l'application ;
- les workflows d'orchestration asynchrones déclenchés depuis un système externe.

Les workflows d'orchestration asynchrones dépendants de l'application correspondent aux orchestrations les plus couramment déclenchées à partir d'une application SBM.

- **Workflows d'orchestration asynchrones dépendants de l'application** : il s'agit des orchestrations les plus couramment déclenchées à partir d'une application SBM. L'exécution d'une orchestration de ce type est déterminée par la génération d'un événement via une action de transition. Les données de l'événement correspondent à celles qui se trouvent dans la table principale de l'application considérée. Les orchestrations de ce type sont susceptibles d'appeler des services Web pouvant relever d'un système externe aussi bien que de l'application SBM. Ici encore, vous devez prendre soin de ne pas modifier l'interface entre cette orchestration et l'application pendant toute la durée du développement de l'orchestration. Veillez à cocher la case « Lock definition » (Verrouiller la définition) du lien d'orchestration *Event without Reply* (Événement sans réponse) de l'application afin d'empêcher toute modification de l'interface lorsque des champs de la table principale sont renommés. Ce mécanisme est disponible dans les versions 2009 R4.02 et ultérieures de SBM.

Les orchestrations de ce type ne doivent pas nécessairement faire partie de la même application de processus que l'application appelante. Pour créer une orchestration compatible dans une application de processus séparée, exportez le lien d'orchestration nommé *Event without Reply* (Événement sans réponse) de l'application d'origine, puis ajoutez une orchestration dans une application vide distincte. Sous l'élément Application Links (Liens d'application) de l'explorateur d'applications, cliquez avec le bouton droit de la souris et sélectionnez l'option de menu *Add New Event Definition...* (Ajouter une nouvelle définition d'événement...). Dans la boîte de dialogue qui s'affiche, sélectionnez l'option *Create from event definition file* (Création à partir d'un fichier de définition d'événement), puis accédez à la définition d'événement que vous avez exportée. Vous pouvez alors procéder au développement. Ici encore, vous devez disposer d'un système SBM pour générer l'événement nécessaire aux tests.

- **Workflows d'orchestration asynchrones indépendants de l'application** : ces workflows d'orchestration asynchrones servent à encapsuler des systèmes externes et peuvent être réutilisés avec plusieurs applications SBM. Indépendante de l'application, l'interface événementielle de ces workflows est définie par le développeur de l'orchestration afin de fournir toutes les instructions requises au système externe. La création de tels workflows revient à définir une interface réutilisable commune vers un système externe, chaque workflow représentant une commande distincte. Ces workflows sont appelés depuis une application SBM par une action associée à une transition. Ils se distinguent des workflows d'orchestration asynchrones dépendants de l'application par la possibilité d'assigner les données des éléments de l'application à l'interface événementielle, tandis que l'interface événementielle dépendante de l'application est définie automatiquement par les données des éléments de l'application.

Pour créer un workflow d'orchestration asynchrone indépendant de l'application, créez une application de processus, puis ajoutez-lui une orchestration en cliquant dessus avec le bouton droit de la souris dans l'explorateur d'applications et en sélectionnant l'option de menu *Add new->Orchestration* (Ajouter->Orchestration). Sous l'élément Application Links (Liens d'application) de l'explorateur d'applications, cliquez avec le bouton droit de la souris et sélectionnez l'option de menu *Add New Event Definition...* (Ajouter une nouvelle définition d'événement...). Dans la boîte de dialogue qui s'affiche, sélectionnez l'option *Create new custom event definition* (Créer une définition d'événement personnalisée), puis définissez les types d'objets et les types d'événements. Ces valeurs permettent au gestionnaire d'événements de déterminer les orchestrations à exécuter lors de la génération d'un événement. Définissez maintenant les données personnalisées à transmettre avec l'événement. Ces valeurs sont les entrées qui seront soumises aux workflows d'orchestration exécutés via cet événement.

- Pour générer un événement défini par ce lien d'application, commencez par l'exporter en cliquant sur le bouton *Export event definition...* (Exporter une définition d'événement...), puis enregistrez-le dans un fichier .mtd. Dans l'application d'origine, cliquez ensuite sur l'élément **External Events** (Éléments externes) sous **Orchestration Links** (Liens d'orchestration) dans l'explorateur d'applications, puis sélectionnez l'option de menu *Add New External Event...* (Ajouter un nouvel événement externe...), accédez au fichier d'événement et importez-le dans l'application. Enfin, pour utiliser cet événement, créez une action associée à une transition, sélectionnez le workflow d'orchestration, puis, à l'étape 2 de la première page de l'assistant, cliquez sur l'événement local et remplacez-le par un événement externe. Terminez l'assistant en sélectionnant l'événement externe que vous venez d'importer, puis cliquez sur l'assignation des données pour assigner les données d'élément présentes dans l'application aux entrées de cet événement. Ce mécanisme n'est disponible qu'à partir de la version SBM 10.1.
- **Workflows d'orchestration asynchrones déclenchés depuis un système externe** : il s'agit des workflows d'orchestration asynchrones qui sont déclenchés à partir d'un événement généré depuis un système externe. L'interface événementielle de ces workflows d'orchestration est définie en fonction des exigences propres au système externe. Ces workflows d'orchestration font couramment appel à des services Web SBM pour manipuler des éléments d'une application déterminée.
Pour créer une orchestration déclenchable depuis un système externe, suivez la procédure décrite ci-dessus afin de créer une orchestration et de spécifier une nouvelle définition d'événement personnalisée. Au lieu d'exporter la définition d'événement, cliquez sur le bouton *Export external event WSDL* (Exporter le fichier WSDL d'un événement externe) afin de créer un fichier WSDL spécifiant un service Web conçu pour générer un événement. Des explications plus détaillées sont disponibles dans la documentation.

Autres techniques de développement avancées : parmi les autres types d'efforts de développements avancés, il convient de citer le développement de modules AppScript ou d'API C. S'il est vrai que de nombreuses installations SBM font appel à ces interfaces, efforcez-vous dans la mesure du possible d'atteindre vos objectifs à l'aide des actions de formulaire, des services Web et des orchestrations, qui facilitent la gestion du système résultant.

Dans ces différents types d'efforts de développement conduits de manière séparée du système SBM principal, la préoccupation clé réside dans l'intelligibilité et la stabilité des points d'interface. La fonction de nom interne introduite dans SBM 10.1 allège sensiblement cette contrainte. Auparavant, toute référence à un workflow, à un état ou à une transition du modèle de workflow SBM reposait sur son libellé. Dans la mesure où les libellés des états et des transitions apparaissent dans les formulaires, les interventions de maintenance courante et l'évolution des workflows pouvaient entraîner leur modification et, par conséquent, la rupture des intégrations. Désormais, les modules AppScript, les API de service Web et l'API C peuvent faire référence à ces éléments par leur nom interne, qui ne change plus après la publication initiale de l'application de processus.

Fusion des changements de long terme dans les systèmes de développement, de test et de production

Une fois que le développement d'un projet de long terme est achevé, il vous reste à réintégrer cet effort dans le système SBM principal.

Une fois que le développement d'un projet de long terme est achevé, il vous reste à réintégrer cet effort dans le système SBM principal.

Tout développement réalisé sur un système séparé peut être fusionné dans une application de processus déployée sur le système SBM principal via la fonction de comparaison et de fusion de Composer.

ORCHESTRATIONS

Pour les orchestrations qui ont été développées dans des applications de processus séparées, cette étape s'avère relativement simple. Dans le cas d'une orchestration déclenchée par un système externe, il vous suffit de déployer l'application de processus contenant l'orchestration pour que cette application s'exécute lorsque le système déclenche l'événement. Pour les autres types d'orchestrations asynchrones, il vous faut uniquement modifier l'application appelante pour générer l'événement qui appelle l'orchestration. Si l'orchestration est définie via un événement externe, vous devrez assigner les données issues de l'application appelante aux entrées de l'orchestration. Si elle a été définie d'après le modèle de données de l'application (comme c'est le cas pour les orchestrations dépendantes de l'application), contentez-vous de générer l'événement via une action associée à une transition.

Dans le cas des orchestrations synchrones et de celles que vous avez conservées au sein de la même application de processus que l'application associée, vous devrez utiliser l'outil de comparaison et de fusion de Composer pour copier l'orchestration, ainsi que le lien d'orchestration correspondant, depuis le projet d'application de processus en cours de développement vers l'application de processus déployée dans le système SBM principal, puis appeler cette orchestration depuis les transitions au niveau desquelles le travail doit s'effectuer.

FUSION DES MODULES APPSCRIPT

Vous avez la possibilité de copier des modules AppScript d'une application de processus à une autre sous la forme de fichiers. Dès lors que la stabilité de l'interface entre le module AppScript et l'application avec laquelle il interagit a été préservée (par exemple, via l'utilisation de noms internes), ce module restera fonctionnel après avoir été copié sur le système principal.

FUSION DES AUTRES CHANGEMENTS

Enfin, tout développement réalisé sur un système séparé peut être fusionné dans une application de processus déployée sur le système SBM principal via la fonction de comparaison et de fusion de Composer. Cette fonction vous permet de réintégrer toutes les modifications apportées à une version divergente d'une application de processus dans l'application de processus du système principal. Elle s'applique à pratiquement tous les types d'éléments de conception, notamment aux orchestrations, formulaires, workflows, modules AppScript, règles, rôles, champs et tables auxiliaires. Dès lors qu'un élément de conception comporte des modifications, la fonction de comparaison vous présente un rapport indiquant les différences, qui ressortent visuellement dans l'environnement familier de SBM Composer.

Bien que cette technique vous permette de rapatrier n'importe quel type d'effort de développement conduit séparément dans le système principal, vous avez plutôt intérêt à encapsuler les projets de long terme dans des éléments de conception distincts, que vous pourrez ainsi transférer comme une seule entité dans le système principal sans avoir à trier les divergences affectant les propriétés des objets existants.

Messages à retenir

Tirez parti des avantages de SBM en adaptant les processus à l'évolution de l'activité au lieu d'imposer des processus rigides aux équipes métiers.

SBM permet de réaliser de manière intuitive la plupart des changements requis pour adapter et faire évoluer progressivement les processus. SBM est conçu pour permettre au personnel non technique orienté métier de modifier des processus de manière itérative. Préservez le lien de proximité entre ces collaborateurs et le processus.

Désignez un propriétaire de processus, qui participera périodiquement à un forum réunissant des représentants des participants au processus (les émetteurs de demandes de changement), des agents d'exécution et des observateurs du processus. Les réunions de ce forum ont pour buts de rendre compte du mode d'utilisation de la solution et de découvrir des opportunités d'amélioration. Améliorez le processus de manière itérative en fonction des commentaires formulés par cette équipe.

Certains efforts, dont les intégrations, donnent de meilleurs résultats lorsque les développeurs interviennent dans le cadre d'un projet de plus long terme. Encapsulez ces efforts de façon à permettre un développement indépendant et à faciliter l'incorporation des livrables dans le système SBM principal.

Évitez de regrouper des petits projets porteurs d'avantages immédiats pour votre organisation avec des efforts de développement de plus grande ampleur. Une telle pratique réduirait en effet la proposition de valeur de « création axée sur le changement » de SMB.

Pour en savoir plus, rendez-vous sur :

www.opentext.com

Communiquez avec nous

