

Adobe® LiveCycle® Enterprise Suite 2 (ES2)

目次

- 2 LiveCycle ES2 のアーキテクチャの概要
- 3 エンゲージメントアプリケーションの設計
- 5 デプロイ
- 5 アプリケーションの実行
- 5 LiveCycle サービスコンテナ
- 7 Invocation Layer
- 8 LiveCycle Foundation
- 8 LiveCycle Work Manager
- 9 レジストリとリポジトリ
- 9 エンタープライズアプリケーションとの統合
- 10 アプリケーションの管理
- 10 まとめ

企業や組織は、バックエンドリソースを効率的に使用しながら、企業・組織の境界を超えて顧客とより効果的に情報をやり取りできるようにする必要性に直面しています。IT 部門は、分散された機能を適切に組織できるような投資判断を求められています。そうした機能は、的確で状況に即した、リッチで魅力的なインターフェイスを通じて、拡張性の高いアプリケーションとして提供される必要があります。

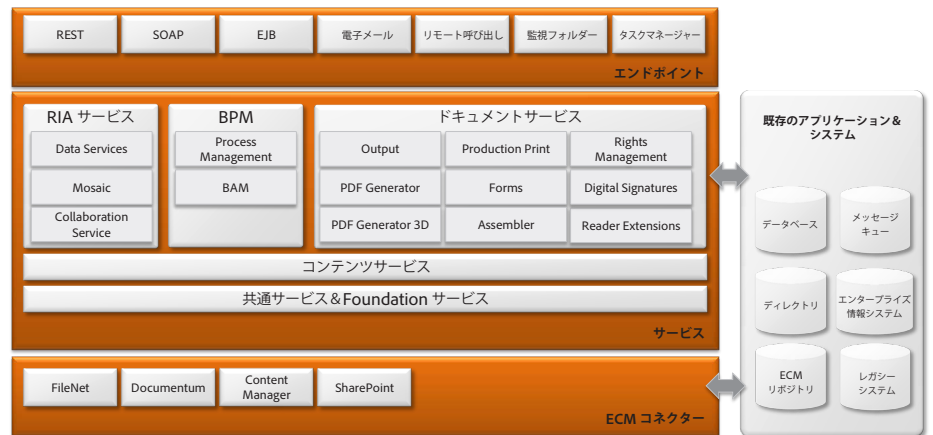
開発コストを抑え、最大限の投資収益率を実現しながら、急速に変化するビジネス環境の中で顧客を効果的に開拓し引き付けるには、サービス指向のアプリケーションを提供する必要があります。これにより、緩やかに結合されたサービスとして公開され、望まれるソリューションとして容易に組み立てることのできる、バックエンドプロセスの提供が可能になります。各アプリケーションは、ユーザーの満足度を高め、プロセスの利用率や完成度を向上させ、トランザクションサイズを増加させるような、高機能で直観的なインターフェイスを介して顧客に提供される必要があります。

Adobe LiveCycle ES2 は、ソフトウェアをサービスとして公開することで、こうした課題に対応します。これによって組織は、世界中のインターネット対応 PC の 98% 以上にインストールされている、クロスプラットフォームの Adobe Reader® および Adobe Flash® の技術を活用し、より安全で高機能なアプリケーションを構築し、エンゲージメントアプリケーション、すなわち優れた顧客体験により、顧客との絆を深めることのできるアプリケーションを、迅速に提供することができます。組織は、Flex、PDF、XML、HTML、Ajax などの標準ベースのインターフェイステクノロジーを適切に組み合わせて、顧客との対話を効率化し、柔軟性を強化し、カスタム実装やベンダー依存を回避できます。

このホワイトペーパーでは、Adobe LiveCycle ES2 のアーキテクチャを紹介し、拡張性の高いアプリケーションを迅速に開発できるようにする、LiveCycle ES2 のサービス指向の手法について簡単に説明します。組織は、Adobe LiveCycle ES2 のサービス指向アーキテクチャ (SOA) と、Adobe Flash、PDF、および Adobe AIR™ を活用することにより、顧客、パートナー、請負業者など、様々な顧客との対話を、以前は不可能だったレベルまで満たすことができます。このホワイトペーパーでは、直観的なユーザー体験、ビジネスプロセスの自動化、コミュニケーション管理など、LiveCycle ES2 で提供されるソリューションの種類についても説明します。

LiveCycle ES2 のアーキテクチャの概要

LiveCycle ES2 のアーキテクチャは、クライアント、ユーザーインターフェイステクノロジー、開発ツール、およびインフラ基盤を網羅しています。以降の各セクションでは、顧客エンゲージメントアプリケーションの設計、実行、統合、および管理という観点から、これらの要素について説明します。



The LiveCycle ES2 のアーキテクチャ

LiveCycle ES2 の設計および開発ツールは、アプリケーションモデル内で機能して、共有のアセットや依存関係を扱い、開発プロセスを効率化し、複数のユーザーが同じアプリケーションを同時に操作できるようにします。管理者は、デザイン時の改訂管理、セキュリティ、および監査を行うコンポーネントを使用して、アプリケーション開発全体を確実に管理できます。

LiveCycle ES2 のソリューションモジュールが提供する再利用可能なサービスを簡単に組み合わせて、顧客エンゲージメントアプリケーションを構築できます。これらのモジュールは、機能豊富なインターネットアプリケーションの作成、ビジネスプロセスの自動化、およびコミュニケーションの強化に幅広く対応しています。

LiveCycle Content Services ES2 は、他の LiveCycle モジュールと連携して使用できる、包括的なライブラリサービスを提供します。このモジュールを使用すると、コンテンツを管理してビジネスプロセスに組み込み、コンテンツのレビューを効率化し、組織の保管ポリシーに従ってコンテンツをアーカイブし、重要なビジネスデータを迅速に識別できるようにコンテンツを分類することができます。

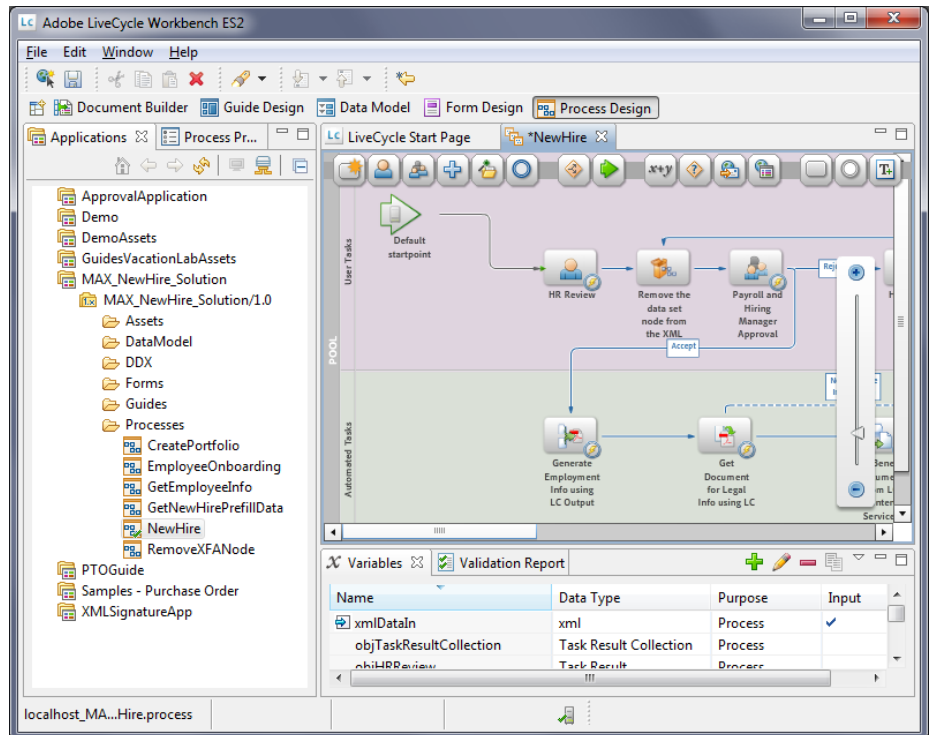
LiveCycle ES2 Connectors for ECM では、LiveCycle アプリケーションを、EMC® Documentum®、IBM® FileNet、IBM Content Manager、Microsoft® Office Sharepoint® Server など、業界屈指のエンタープライズコンテンツ管理システムに統合できます。

Adobe LiveCycle Foundation は、顧客エンゲージメントアプリケーションのデプロイ、実行、および管理を可能にする、統合環境を提供します。Foundation に不可欠なコンポーネントであるサービスコンテナは、すべてのソリューションモジュールと関連するサービスに、共通のランタイム環境を提供します。LiveCycle Foundation のサービスで提供される機能により、Web サービス、LDAP、JDBC、FTP などの一般的な IT インフラにソリューションを統合できます。

LiveCycle ES2 には、構成、管理、デプロイ、監視などの様々なタスクを簡略化する、一元化された各種ツールが備わっています。1つの Web インターフェイスから、各モジュールをそれぞれ管理できます。

エンゲージメントアプリケーションの設計

LiveCycle Workbench™ ES2 および Adobe® Flash Builder™ は、チームメンバーに適切な機能を公開する、統合されたフォーム、プロセス、およびリッチインターネットアプリケーション（RIA）の設計ツールを提供します。例えば、フォームデザイナー（設計者）とビジネスアナリスト（業務分析の担当者）は、LiveCycle Workbench ES2 で共同作業できます。また、開発者は、LiveCycle ES2 ソフトウェア開発キット（SDK）を使用して、データ統合、スクリプト作成、およびモジュール機能の拡張を行うことによって、アプリケーションの機能を拡張できます。ビジネスアナリストは概念プロセスを定義でき、フォームデザイナーは開発者が作成したフォームフラグメントとスクリプトコンポーネントを再利用できます。フォームテンプレート、サービス定義、プロセス定義、ポリシー、DDX ファイル、XML スキーマ、イメージなどの再利用可能なコンテンツコンポーネントと、複数のフォームで再利用できるヘッダーやフッターなどのフォームフラグメントは、LiveCycle ES2 で格納され、バージョン管理され、保護されます。こうしたアプローチにより、アプリケーション開発に伴うメンテナンス作業や更新作業が大幅に簡略化されます。LiveCycle ES2 のリポジトリはエンタープライズコンテンツ管理（ECM）システムによってシームレスに拡張でき、アセットはアクセス制御リストを使用して保護できます。



LiveCycle Workbench ES2 のプロセスデザインビューでは、スイムレーンなどの概念を使用して、ビジネスプロセスを簡単にモデリングできます。

LiveCycle Workbench ES2 では、フォーム、ガイド、プロセス、DDX（Document Description XML）、およびデータモデルを作成するタスクが統合されています。リポジトリと統合されていることによってアーティファクト（中間成果物）の管理が容易になり、改訂管理、セキュリティ、監査などの重要な管理作業がサポートされます。ビジネスアナリストは、スイムレーンやプールなど、標準の BPMN（Business Process Modeling Notation）構造体を使用して、プロセスを作成できます。フォームデザインの観点では、LiveCycle Designer ES2 の機能を埋め込んで XML フォームテンプレートのグラフィックレイアウトを作成し、それを LiveCycle Forms ES2 で PDF、SWF、または HTML として表示できます。フォームをバックエンドのデータソースや設定場所に簡単に組み込んで、データ収集などの一般的なタスクを自動化するアプリケーションを構築できます。既存の XForms を使用するユーザーは、XForms の読み込み機能、XFA ベースのテンプレートへの変換機能、HTML、PDF、および SWF 形式での表示機能を使用して、既存の投資を活用できます。開発者は、データモデリングによって高度なアプリケーションを作成することで、開発時間を短縮し、データ統合を簡素化することができます。デザイナーは、Guide Builder を使用して、Flash ベースの直観的なデータ収集インターフェイスを作成できます。ドキュメント作成者は、Document Builder を使用して、PDF ドキュメントを動的に操作する DDX コマンドファイルを視覚的に構築できます。Eclipse をベースとする LiveCycle Workbench ES2 では、トレーニング、コスト、およびリスクを最小限に抑え、Eclipse プラグイン環境をサポートすることができます。

組織は、LiveCycle ES2 を使用して、RIA やガイドをデプロイできます。これらは、Web アプリケーションやデスクトップアプリケーション向けの、高度な対話機能を備えたインターフェイスであり、Adobe Flash Player を使用してブラウザで実行したり、Adobe AIR を使用してデスクトップ上で実行したりできます。Flash Builder 4.0 および Flex フレームワークを使用して構築されたアプリケーションは、より多くのユーザーのニーズを満たし、ユーザーの生産性と満足度を高め、収益の増加をもたらします。これらの成果を得られるのは、LiveCycle ES2 を Flash と簡単に統合して、LiveCycle ES2 サービスが公開する Flex のリモート呼び出しを活用できるためです。LiveCycle Service Discovery Plug-in for Flash Builder 4.0 を使用すると、リモートエンドポイントから LiveCycle サービスを呼び出す Flex アプリケーションおよび AIR アプリケーションを簡単に構築できます。こうして構築されたアプリケーションでは、LiveCycle を使用して、データの処理、ドキュメントの作成、およびプロセスの開始を実行できます。

Flex フレームワークでは、100 を超えるインターフェイスコンポーネントが提供されます。Flex のユーザーインターフェイスは、業界標準の MXML で記述されています。Flex サポートは LiveCycle ES2 に完全に統合されているため、開発者はプラットフォームサービスを活用して、Flex コンポーネントを管理およびデプロイできます。フォームデザイナーも、ガイドをデータ収集インターフェイスとして利用して、ユーザーが理にかなう順序に従ってタスクを順次実行できるように設計できます。ガイドは、データ収集プロセスを案内する、ウィザード形式のアプリケーションです。入力データに応じてガイドを動的に変更および調整できるため、適切なデータだけが正確に収集されます。また、一般的なデータモデルを活用する、直観的な WYSIWYG 形式の Guide Builder を使用すると、高度なガイド付き操作を構築して、記述するコードの量を減らし、データ統合処理を簡略化できます。



LiveCycle ES2 では、一般的なフォームおよびデータモデルを使用して適切なインターフェイスを作成できるため、柔軟性とエンゲージメントが強化されます。

Web サイトの閲覧には要求-応答型 (Request/Response) のコミュニケーションモデルで十分ですが、多くのアプリケーションでは、最適化された高パフォーマンスのデータ転送や、他の対話モードが求められます。例えば、メッセージの発行-購読 (Publish/Subscribe)、サーバーからクライアントへのデータやアラートのプッシュ機能などが必要です。LiveCycle Data Services ES2 ソリューションモジュールでは、AIR、Flex、または Ajax ベースのアプリケーションインターフェイスとバックエンドシステム間での、大量のデータ交換が確実に実行される配信と同期が可能です。

開発者は、新しく導入されたアドビのアプリケーションモデリングテクノロジーを活用して、高度なアプリケーションを作成し、開発期間の短縮とデータ統合の簡略化を実現できます。LiveCycle Data Services ES2 は、標準の J2EE Web アプリケーションとしてデプロイされます。このため、Java™ 開発者は、アプリケーションが JMS トピックおよびキューに発行およびサブスクライブできるようにすると共に、既存の EJB コンポーネントと統合できるようにすることで、既存のビジネスロジックを活用できます。強力なデータ管理 API およびコネクタを活用するこのモジュールは、複数のクライアントで共有される共同作業用コンテンツなど、複数のソースを使用するクライアントとサーバー間のデータ通信を容易にします。

アドビのアプリケーションモデリングテクノロジーは、データエンティティとデータリレーションシップを定義するメカニズムに対応しているという点で、他のデータモデリング言語と似ています。ただし、最近のデータ処理を中心としたアプリケーションの構築には、この機能は適していません。アドビのアプリケーションモデリングテクノロジーでは、サービスとデータの振る舞い（ビヘイビア）のサポートを追加することによって、この問題を解決しています。モデルは、データを表すのではなく、アプリケーションを表します。このため、アプリケーションの様々なコンポーネントが、データおよびビジネスロジックを共有できます。

デプロイ

テスト、ステージング、および運用サーバーへのデプロイに関連する LiveCycle ES2 のアプリケーションコミショニングタスクは、LCA (LiveCycle Archive) ファイルを使用することで簡略化されます。このファイルは、アプリケーション関連のリソースを 1 つのアーカイブに自動的にパッケージ化して、ユーザー間やシステム間で転送できるようにするものです。管理者は、LiveCycle の管理ツールを使用して、アプリケーションを運用サーバーにステージングまたはデプロイできます。アプリケーションをデプロイするサーバー上に開発インフラを用意する必要はありません。

アプリケーションの実行

LiveCycle ES2 のアーキテクチャでは、分散した機能を効率的に組織して使用できるようにする、SOA の概念を採用しています。このホワイトペーパーで説明しているように、ビジネスアナリスト、アプリケーション設計者、フォームデザイナー、および開発者は、LiveCycle ES2 で提供されるフレームワークを利用して、ターゲットソリューションに簡単に組み立てることが可能であり、組織のアジリティ（俊敏性）を大幅に向上させて変化しつづけるビジネスニーズを満たす、バックエンドプロセスをサービスとして公開できます。

LiveCycle ES2 では、明確に定義されたインターフェイスを持つ個別のサービスとしてビジネス機能を公開するため、必要に応じて他のアプリケーションで再利用できます。例えば、LiveCycle Rights Management ES2 では、アクセス制御機能を必要とするすべてのアプリケーションに、堅牢でドキュメントレベルの同じアクセス制御機能を提供します。公開されるサービスは複数のオーナーシップ領域にわたって機能することがよくあるため、組織は、セキュリティ、管理、および関連する各機能を網羅した、技術とビジネスの両方の文脈でサービスを説明するポリシーを宣言する必要があります。LiveCycle ES2 のサービスは、それらのサービスの使用を規定する、XML 記述でデプロイされます。

LiveCycle サービスコンテナ

LiveCycle サービスコンテナは、SOA の概念に基づく拡張可能な単一の統合ランタイム環境であり、この環境内で、LiveCycle のすべての標準サービスおよびカスタムサービスが実行されます。サービスコンテナ機能には、サービスを同期プロセスおよび非同期プロセスに編成する機能、監視機能、監査機能、協調的セキュリティ機能などが含まれます。開発者や管理者にとっては、シンプルできめ細かい方法でサービスを開発、デプロイ、保護、および管理できる点もメリットになります。非同期的なクライアント要求が認証されると、その要求がジョブマネージャーに渡され、ジョブマネージャーでサービスが呼び出されます。

次に、サービスコンテナのいくつかの主要な特性について説明します。

- **コンポーネントモデル**：サービスコンテナで採用されている拡張可能なコンポーネントモデルでは、緩やかに結合されたサービスが相互に作用して、行き届いた対応により、絆を深める顧客体験を提供できます。LiveCycle Foundation で提供されるサービスによって、LDAP を介したディレクトリサーバー、Web サービス、JDBC、FTP、ファイルシステムなどの一般的なエンタープライズインフラと LiveCycle ES2 プラットフォームを統合できます。LiveCycle ES2 は、Adobe Reader の機能を拡張するための LiveCycle Reader Extensions ES2 や、顧客エンゲージメントアプリケーションを LiveCycle サービスやバックエンドシステムと統合するための LiveCycle Data Services ES2 など、標準のソリューションサービスも網羅しています。

- **オーケストレーション機能**：LiveCycle ES2 のランタイム環境の重要な要素は、プロセスを編成して複雑なタスクにまとめる機能のセットです。組織は、複数のプロセスを編成して新しいサービスを構築し、長期間有効なビジネスプロセスの一部としてエンドユーザーが LiveCycle ES2 サービスとやり取りできるようにし、サービスをサードパーティの製品やサービスと統合することができます。

既存のドキュメントサービスを編成することで、ソリューションを容易に組み立て、特定の要件に合わせて調整できます。例えば、PostScript® ファイルをディスクから取得し、そのファイルを PDF ファイルに変換してから、ディスクの別の場所に保存し直す必要があるとします。開発者は、LiveCycle Workbench ES2 を使用して既存の 3 つのサービスを ConvertPS という 1 つの統合的サービスを構築することで、この要件を容易に満たすことができます。このサービスでは、1 つの呼び出し処理を使用し、LiveCycle ES2 で使用可能な呼び出し方法を最大限に活用します。



ConvertPS として編成されたサービス

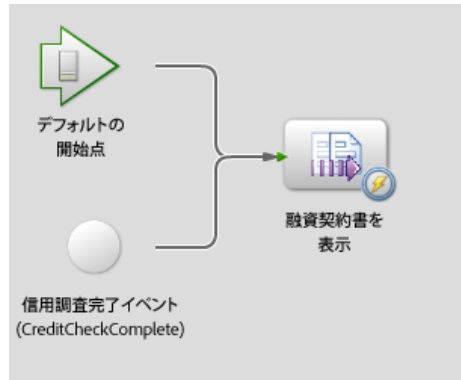
- **イベントフレームワーク**：ソリューション開発者は、LiveCycle ES2 で提供されるフレームワークを介してカスタムイベントの呼び出しと受け取りを行うことで、実行されているプロセス間の通信を容易にすることができます。イベントを使用して、関連するプロセスで依存処理が終了したときに、プロセスの開始、終了、状態変更などを行うことができます。イベントに関する情報は、イベントリスナーで使用できる XML 変数に格納されます。イベントをフィルター処理して、トリガーされるイベントフィルターを判断できます。LiveCycle ES2 は、非同期イベント、例外イベント、および時間指定イベントに対応しています。

イベントフレームワークを利用するプロセスの典型的なものとして、信用調査が完了した後で非同期イベント CreditCheckComplete がスローされる、ローンアプリケーションがあります。このプロセスは、顧客がローン申請書を提出したときに開始されます。



このローンアプリケーションプロセスでは、イベントフレームワークを使用して、信用調査の完了を非同期的に伝えます。

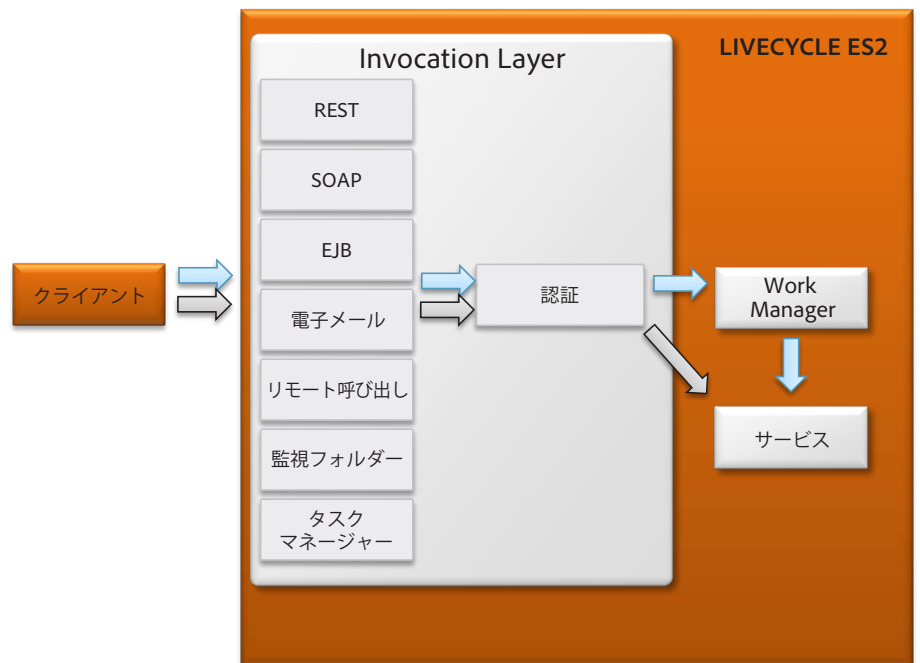
CreditCheckComplete イベントトリガーは、その後続く ApproveApplicant プロセスの開始点として機能します。



信用調査の完了後、他のプロセスが呼び出されて、他のビジネスアクティビティが自動化されることがあります。

Invocation Layer (プロセスの呼び出しを行うレイヤー)

LiveCycle ES2 Invocation Layer では、様々なメカニズムを使用して、ユーザーや外部アプリケーションがプロセスを開始できます。Invocation Layer は呼び出し要求を受け取り、セキュリティ、監査、ログ、およびトランザクション管理の各機能を実行して、最終的にターゲットサービスを呼び出します。 .NET、PHP、電子メールなどのクライアントが、サービスを開始できます。この統合された Invocation フレームワークによって、複雑さが解消し、相互運用性が強化されます。



青色の矢印は、クライアントが LiveCycle ES2 サービスへの非同期要求を行った場合のアクティビティフローを表します。灰色の矢印は同期通信を表します。

LiveCycle ES2 では、様々な統合方法でサービスを呼び出すことができます。

- **REST** : LiveCycle ES2 で作成されたプロセスは、REST (Representational State Transfer) 要求で呼び出すことができます。REST 要求は HTML ページから送信されます。つまり、LiveCycle ES2 サービスには、REST 要求を使用して Web ページから直接アクセスできます。例えば、任意の Web ブラウザーを使用して LiveCycle ES2 プロセスを呼び出し、HTTP POST 要求で送られたデータで表示される PDF ドキュメントを読み込むことができます。
- **SOAP** : Web サービスを介したサービス呼び出しは、Web Services Basic Profile 1.1 に準拠しています。また、Base64、MIME、DIME、HTTP、MTOM、および Swa Ref の添付を使用する SOAP (Simple Object Access Protocol)、WS Security 1.0 標準、SOAP エンドポイントおよび WSDL の動的生成をサポートしています。Web サービス呼び出しでは、短期間有効な同期サービス要求と長期間有効な非同期サービス呼び出しの両方が可能です。
- **EJB** : Java によるサービス呼び出しでは、RMI (Remote Method Invocation) を利用し、厳密に型指定された API を使用します。Java ベースの呼び出し方法は、ドキュメントスループット用に高度に最適化されており、サイズの大きいドキュメントを高いパフォーマンスでサービスに送信することができます。
- **電子メール** : 指定した電子メールアカウントが電子メールを受信したときに、サービスを呼び出すことができます。この呼び出し方法では、基本認証と SSL の両方を使用して、IMAP と POP3 をサポートします。デザイナーは、電子メール添付書類のパターンを定義し、電子メールのヘッダー、本文、および添付書類のデータをマッピングできます。例えば、特定のアカウントに電子メールで PDF ファイルが送信された場合に、ポリシー適用の処理を呼び出し、その PDF ファイルを送信元の電子メールアドレスに送り返すことができます。
- **監視フォルダー** : LiveCycle ES2 で監視するように構成されているネットワークフォルダーにファイルを書き込むことによって、サービスを呼び出すことができます。このようなフォルダーは、構成可能なファイル名パターン (.pdf など) を使用してスキャンされ、目的の処理が実行されます。処理された出力ファイルは、外部アプリケーションやユーザーが使用できるように、指定したフォルダーに格納することができます。
- **リモート呼び出し** : Flex クライアントは、Flex RemoteObject タグを使用して LiveCycle ES2 サービスを呼び出すことができます。この方法では、LiveCycle ES2 のブリッジに ActionScript が提供され、同期呼び出しと非同期呼び出しがサポートされ、動的なエンドポイント作成が行われるため、開発リソースを大幅に節約できます。
- **タスクマネージャー** : この統合方法は、LiveCycle Process Management ES2 でのみ利用できます。LiveCycle Workspace ES2 でサービスをタスクとして公開すると、適切なアクセス権を持つユーザーが、Workspace Start Processes リストからそのタスクを手動で呼び出すことができます。例えば、出張から戻った社員が、経費の提出プロセスを開始できます。

上記の呼び出し方法のいずれかを使用して非同期要求が受け取られると、メッセージレシーバーはその要求を認証し、正規化された呼び出し要求を作成します。レシーバーはサービスと処理の名前を調べ、この情報をルーターに渡します。

LiveCycle Foundation

LiveCycle Foundation は、クライアントから送られた要求のルーティングと管理など、すべてのアプリケーションに共通の機能を提供します。非同期要求のサービスと処理の詳細が取得されると、ルーターが呼び出しマネージャーを呼び出します。次に、レシーバーを介してジョブ ID がクライアントに返されます。

LiveCycle Work Manager

LiveCycle ES2 のサービスとして実行される処理には、短期間有効なもの (short-lived) と長期間有効なもの (long-lived) があります。短期間有効な処理は、呼び出し元と同じスレッドで同期的に完了します。大部分のプログラミング言語の場合と同様に、こうした処理は応答を待ってから続行されます。

しかし、エンタープライズプロセスとして実行されるタスクの多くは、同期的に完了することを期待できません。長期間有効なこれらの処理は、複数のシステムにわたって実行されたり、組織の枠を超えて実行されることがよくあります。例えば、複数の自動化されたタスクや手作業で実行されるタスクを統合した大規模なソリューションの一部として、ローン申請書を完成させて送信する必要がある場合などです。このような処理は、応答を待つ間にも継続されている必要があります。長期間有効な処理では基になる作業が非同期的に実行されるため、処理の完了を待つ間にも、リソースを使用できます。短期間有効な処理の場合とは異なり、LiveCycle Work Manager は、長期間有効な処理について、呼び出したあとは完了するものとは見なしません。システムが同じサービスの別の処理を要求したり、ユーザーがフォームを送信したりなど、外部からのトリガーによって処理が完了する必要があります。LiveCycle Work Manager では、業界標準の JSR-237 Work Manager for Application Servers を使用して、状態情報を受け取ります。LiveCycle ES2 によって、追跡や復元など、長期間有効な処理のためにすぐに使用できるフレームワークが提供されます。

レジストリとリポジトリ

LiveCycle ES2 のリポジトリは、アセットのアクセシビリティと管理が大幅に強化される、アプリケーション開発者向けの高度な機能が提供されます。このリポジトリは、単一または複数のクラスタ化された J2EE システムに対してシームレスに機能するため、拡張可能でフェイルオーバー機能を備えた開発を容易に実行できます。デプロイメントアセットのバージョン管理は、これらのアクティビティを実行時に構築および管理するのに役立ちます。中央のリポジトリに格納されたコンポーネントとサービスは、対応するレジストリで記録されるため、コンポーネントとサービスの参照、検索、開始、および停止や、サービスのデプロイとデプロイ解除が容易になります。

Adobe LiveCycle Connectors for ECM は、LiveCycle ES2 を、EMC Documentum、IBM FileNet、IBM Content Manager、Microsoft SharePoint などの ECM システムと簡単に統合できます。

サービスが呼び出されると、レジストリとリポジトリを使用して、その要求を完了させるのに必要なランタイムアセットにアクセスします。クライアントはその非同期要求の状態をポーリングし、ジョブが完了とマーキングされると、サービスマネージャーを呼び出して結果を取得します。

エンタープライズアプリケーションとの統合

LiveCycle Foundation は、他のプロセスをサーバーおよびクライアントとして統合するための、幅広い方法が提供されます。

LiveCycle ES2 は、前述の呼び出し方法に加えて、他のプロセスのサービスを要求し、結果をアプリケーションに返すための、いくつかの方法をサポートしています。これらの方法を使用して、既存のリソースを柔軟に活用できます。

- **Web サービス**：サービスは、既存の Web サービスのプロキシとして実装することもできます。component.xml ファイルのプロキシ定義には、プロキシサービスの一部として公開する処理とパラメーターを定義するのに必要な情報が含まれています。
- **カスタムコンポーネント**：開発者は、1 つ以上の処理が含まれている、ステートレスな POJO (Plain Old Java Object) を作成して、独自のサービスを追加できます。Java 開発者の場合は、最小限の作業を行うだけで、既存のコードを再利用することがよくあります。component.xml ファイルに記述される POJO と、ライブラリなどの依存関係は、1 つの JAR ファイルにまとめられます。その後で、他のデプロイの場合と同様のツールや手順を使用して、ドキュメントサービスが LiveCycle サービスコンテナにデプロイされます。
- **Connectors for ECM**：コネクタは、チェックイン/チェックアウト、ロック、コンテンツメタデータへのアクセスなど、コンテンツリポジトリサービスを提供します。コネクタサービスコンポーネントは、ドキュメントやその他のコンテンツオブジェクトを LiveCycle プロセスを介して受信および出力し、ECM システムをカスタマーアプリケーションに接続します。
- **JMS**：LiveCycle ES2 は、分散環境内の Java ベースアプリケーションとの通信用に、JMS API をサポートしています。
- **サービスプロバイダーインターフェイス**：LiveCycle ES2 には、標準の外部認証方法 (LDAP (Lightweight Directory Access Protocol) サーバー、Microsoft Active Directory、カスタムの認証方法および承認方法など) のための、サービスプロバイダーインターフェイス (SPI) が含まれています。
- **Foundation サービス**：LiveCycle ES2 は、一般的な IT インフラと統合するための主要な機能を提供する、すぐに使える多数のコンポーネントおよびライブラリをサポートしています。これには、LDAP、認証、ECM、JMS、SOAP、電子メール、およびその他のバックエンドシステム、アプリケーション、リソースを介したユーザーディレクトリへの統合が含まれます。

- **LiveCycle Data Services ES2:**このソリューションモジュールは、Flex または Ajax ベースのアプリケーションインターフェイスとバックエンドシステムとの間で高ボリュームのデータ交換および同期化を行うための、リアルタイムメッセージング、確実なメッセージ配信、質の高いサービス、データ管理サービスを提供します。

アプリケーションの管理

LiveCycle Administration Console はシステム管理者が使用する Web ベースのツールです。システム管理者はこのツールを使用して、ポート番号とログオプションの調整、LiveCycle ES2 アプリケーションのデプロイと構成の管理など、様々なタスクを実行します。また、LiveCycle Administration Console を使用して、ソリューションコンポーネントのサービスエンドポイントを作成および管理したり、アプリケーションのデプロイを実行したりすることもできます。このツールでは、プラットフォーム全体ですべてのサービスにわたって使用される共通の認証管理サービスも提供されるため、ユーザーおよびグループの定義と構成、LDAP との統合などのタスクも容易に実行できます。管理者は、LiveCycle Administration Console を介して、JMX 互換のサービス監視にもアクセスできます。

さらに管理者は、アプリケーションのプロセス定義や、フォーム、イメージ、XML ファイルなどの関連リソースが含まれている LCA ファイルの書き出し、読み込み、および管理を実行することもできます。LCA を使用することで、システム間でのアプリケーションの書き出しと転送が容易になります。また、システムを停止することなく、LCA を LiveCycle ES2 に読み込むことができます。スクリプトを使用するデプロイもサポートするこのメカニズムを利用すると、開発、テスト、およびデプロイの各段階にソリューションをステージングする必要がある組織の作業負荷が軽減されるだけでなく、パートナーや顧客との間でアプリケーションを簡単に共有および配布できます。

システム管理者は、Business Activity Monitoring Workbench を使用して、イベントや文脈依存型のデータソースを構成し、複数の並行データストリームにアクセスすることで、ビジネスの相互接続性の監視を簡単にセットアップできます。また、BAM Workbench では、ビジネスルールとダッシュボードオブジェクトに加え、ビューやキューブも作成できます。管理者は、単純なものから複雑なものまで、様々な条件を定義して、アラートをトリガーできます。

まとめ

LiveCycle ES2 は、データ収集、ビジネスプロセス管理、ドキュメントセキュリティ、およびドキュメント生成の各機能を組み合わせた、拡張性の高い統合プラットフォームです。LiveCycle ES2 を使用すると、ペーパーワークを減らし、迅速な意思決定と確実な規制順守を実現する、機能豊富で優れたアプリケーションを作成し、配布できます。LiveCycle ES2 には、J2EE との互換性がある標準ベースの統合環境でアプリケーションを開発および配布するためのツールが備わっており、LiveCycle ES2 のアプリケーションを既存の IT アセットと簡単に統合できます。

企業や組織は、LiveCycle ES2 を使用して、データ収集から複雑なビジネス変革に至るまで、様々な問題を解決できます。例えば、政府機関では、税の電子申告や eGrant 申請管理などの処理をサポートしています。金融サービスソリューションには、高度に商品化されたマーケットで金融機関を差別化するのに役立つ、ローン申請処理や口座開設処理が含まれています。

LiveCycle ES2 の提供する統合されたツールセットによって容易に開発できるため、ビジネスアナリスト、開発者、フォームデザイナーなどが、共有の、再利用可能なリソースを使用して共同作業を行うことができます。LiveCycle ES2 の中核を成すのは、呼び出し、イベント管理、プロセス管理など、基幹機能を各モジュールに提供する、単一のランタイム環境です。Flex のリモート呼び出しや Web サービスなど、共通の呼び出し方法および要求方法によって、ユーザーや外部プロセスとの統合も容易になります。

サービス指向設計を採用した LiveCycle ES2 は、顧客エンゲージメントアプリケーションの構築、デプロイ、および管理を大幅に簡略化し、豊富な機能を持つ効率的なユーザー体験を、サービスとして公開されるバックエンドプロセスと一貫した方法で結びつけます。LiveCycle ES2 プラットフォームで緩やかに結合されたサービスから顧客エンゲージメントアプリケーションを構築することで、開発、デプロイ、および管理に伴う複雑さを軽減し、IT システムをビジネス上の優先順位に合わせて調整し、コストを削減し、組織のアジリティ（俊敏性）と投資収益率の向上を実現することができます。



アドビ システムズ 株式会社
〒141-0032 東京都品川区大崎 1-11-2
ゲートシティ大崎イーストタワー
www.adobe.com/jp
Adobe Systems Incorporated
345 Park Avenue, San Jose, CA 95110-2704 USA
www.adobe.com

Adobe, the Adobe logo, Acrobat, ActionScript, Adobe AIR, the Adobe AIR logo, the Adobe PDF logo, Flash, the Flash logo, Flex, Flash Builder, LiveCycle, PostScript, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries. Java is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries. All other trademarks are the property of their respective owners.

© 2010 Adobe Systems Incorporated. All rights reserved. Printed in the Japan.
03/10