



OutSystems でのアプリ運用と保守

システム開発の生産性、保守性の向上やデジタルトランスフォーメーション推進の手段として注目されているローコード開発。中でも「OutSystems」は、アプリケーション開発の生産性だけでなく、開発基盤として、開発後の運用・保守における生産性向上にも貢献できる製品として注目されています。

本書では、運用・保守における効率化をポイントに OutSystems の機能をご紹介します。既に OutSystems をご導入いただいているお客様、今後導入をご検討されているお客様のご参考になれば幸いです。

内容

OutSystems における運用機能	エラー! ブックマークが定義されていません。
● アプリの環境移行	3
● アプリのバージョン管理	3
● モニタリング機能	4
OutSystems における保守機能	5
● Architecture Dashboard による解析	5
● アプリ間の依存関係分析	6
● 作成したアプリのフィードバック機能	8
終わりに	8

OutSystems における運用機能

OutSystems で開発したアプリを運用する機能についてご紹介します。OutSystems は、開発環境、環境、本番環境の3つの環境と、それらを運用、保守する環境である「Lifetime 環境」で構成されます。

OutSystems のアプリ開発では、「小さい機能単位」で開発することが推奨されています。運用機能においてもこの特性をカバーする機能が備わっており、Lifetime 環境がその役割を担っています。アプリ開発完了後の検証環境、本番環境への移行、バージョン管理、そしてそれぞれのアプリの監視方法についてご紹介します。

アプリの環境移行

Lifetime 環境の機能により、開発環境から検証環境、検証環境から本番環境へのアプリ移行を GUI ベースで実行できます。移行の際は、対象アプリが参照する環境の解析が行われ、不足しているモジュールがあれば警告表示されるため、移行漏れを抑止できます。アプリ開発をサイクル的に行っている場合は、開発環境、検証環境、本番環境で稼働しているアプリのバージョンがそれぞれ異なります。そのような場合、それぞれの環境にどのバージョンのアプリが配置されているか分かるため、環境間の差異を視覚的に確認できます。

Asset	development	Testing	Production
Asset_共通部品_FW	0.3 1/7, 09:21	0.3 1/7, 09:25	0.3 1/7, 10:01
Asset_共通部品_TW	1.0 2021-01-15, 09:24		
Asset_動作確認_FW	1.0 1/11, 08:11		
Asset_動作確認_TW	1.0+ 1/12, 07:59		

図 1. 開発環境、検証環境、本番環境で稼働しているアプリのバージョン

アプリのバージョン管理

OutSystems では、モジュール単位でアプリの更新履歴を保持しています。OutSystems の開発機能に「One-Click Publish」とよばれる、コンパイル、デプロイを

ワンクリックで行える機能があります。管理コンソールである Service Center の機能となり、「One-Click Publish」を実施した単位でバージョン管理され、モジュールのダウンロード、開発環境、検証環境、開発環境へのデプロイができます。

仕様齟齬などがあり、「以前のバージョンに戻したい」という場合でも Service Center の GUI で変更できます。

Module Inventory

Back to Modules List General Log Screen Log Error Log Service Actions Log Application Objects

Application: Stock Keeper

Description: (No description available)

Status: The Module is OK. No messages to show.

Last Published: 2021-12-20 00:30:16 by [user]

Redeploy Published Version Download Published Version Publish All Consumers Test App Delete

Versions Dependencies Single Sign-On Solutions Integrations Tenants Site Properties Timers Operation

Version	Uploaded	Published	Service Studio Version	Verified	Actions
4868	2021-12-20 00:30:00 by [user]	✓	11.12.5 (Build 51077)	✓	Publish Download
4867	2021-11-24 03:04:39 by [user]		11.12.7 (Build 51940)	✓	Publish Download
4866	2021-11-24 02:16:54 by [user]		11.12.7 (Build 51940)	✓	Publish Download
4865	2021-11-24 02:11:32 by [user]		11.12.7 (Build 51940)	✓	Publish Download
4864	2021-11-16 03:31:21 by [user]		11.12.7 (Build 51940)	✓	Publish Download
4863	2021-11-15 09:47:52 by [user]		11.10.16 (Build 40208)	✓	Publish Download
4862	2021-11-15 09:41:57 by [user]		11.10.16 (Build 40208)	✓	Publish Download
4861	2021-11-15 09:39:06 by [user]		11.10.16 (Build 40208)	✓	Publish Download
4860	2021-11-15 08:59:45 by [user]		11.10.16 (Build 40208)	✓	Publish Download
4859	2021-11-15 08:51:28 by [user]		11.10.16 (Build 40208)	✓	Publish Download

図 2. アプリのバージョン一覧

モニタリング機能

Service Center の機能には、Error Log、General Log、Environment Health（サービスの死活監視）など、各環境の情報を収集、閲覧できるモニタリング機能があります。

Error Log は、ログイン時の ID/パスワード間違いに関するログや、データ取得時の実行エラー、タイマー処理の実行エラーなどのログが出力されます。作成するアプリでログを埋め込むこともでき、そのログを出力し Service Center で確認することもできます。

General Log では、API 処理の送受信ログ、実行されているモジュール単位で処理が遅い SQL や画面表示がある場合などにログが出力されます。処理の遅い SQL ログや画面表示などは、保守・運用時の改善活動に役立ちます。

Environment Health では、アプリの死活監視だけでなく、アプリケーションサーバー、DB サーバーのプロセスの死活監視も行うことができます。ただし、ログのエラー検出時、アプリの停止時の通知機能はないため、監視ツールと連携した運用環境を構築する必要があります。

Time of Log	Module	Message	Source	Server
2022-01-17 09:04:29	Inventory (Stock Keeper)	修理済みのレコードが存在しません。		
2022-01-17 09:04:07	Inventory (Stock Keeper)	StoreManager role required		
2022-01-17 09:01:13	Mobile_Sample (Mobile_Sample)	The INSERT statement conflicted with the FOREIGN KEY constraint "OSFRK_OSUSR_ZIH_ATTENDANCE_OSUSR_ZIH_EMPLOYEE_ID". The conflict occurred in database "Devoutsystems", table "dbo.OSUSR_ZIH_EMPLOYEE", column "ID". The statement has been terminated.		
2022-01-17 09:01:12	Mobile_Sample (Mobile_Sample)	The INSERT statement conflicted with the FOREIGN KEY constraint "OSFRK_OSUSR_ZIH_ATTENDANCE_OSUSR_ZIH_EMPLOYEE_ID". The conflict occurred in database "Devoutsystems", table "dbo.OSUSR_ZIH_EMPLOYEE", column "ID". The statement has been terminated.		
2022-01-17 09:00:20	InventoryApproval (Stock Keeper Web)	SalesDepartment role required		
2022-01-17 09:00:05	Inventory	販売済みのレコードが存在しません。		

図 3. Error Log 一覧

OutSystems における保守機能

OutSystems は、リリースしたアプリの保守機能を備えており、リリースしたアプリの問題点を発見するだけでなく、アプリの要件が変更になり、修正する場合に他のアプリへの影響度などを把握することができます。

Architecture Dashboard による解析

Architecture Dashboard は、OutSystems が示すベストプラクティスに基づき、アプリ内を解析し、「technical debt（技術的負債）」のレベルを判定します。「technical debt」は、ベストプラクティスに従っていない実装による問題点を、パフォーマンス、セキュリティ、アーキテクチャ、保守性の4つの観点で検知し、改善方法を提示してくれます。

図 4.は、アプリごとに「technical debt」を判定した結果のサンプルです。アプリ単位に解析され、赤色になるほど危険度が高く、緑色に近づくほど危険度が低いことを示します。

Apps						
Sync is in progress...						
End user						
Independent Modules Reactive Web	Clone_StockKeeper Mobile	Asset_BDDテスト_TW Traditional Web	20211202_AD_Test Mobile	Data Grid Sample Traditional Web		
Mobile_Sample Mobile	TDI OSPT Reactive Web	Asset_Tipsサブ... Traditional Web	Asset_Tipsサブ... Reactive Web	Stock Keeper Web Reactive Web	IdP Traditional Web	
Google Services OAuth2 Traditional Web	ReactiveWeb開発... Reactive Web	HeaderFixed Traditional Web	人財育成_デモアプリ Reactive Web	TDI_OSPT_Certific... Reactive Web	Asset_共通部品_RW Reactive Web	
商品マスタアプリ Traditional Web	BDDFramework Traditional Web	BDDTset Mobile	zTree -Demo Traditional Web	Add To Calendar Traditional Web		
インターネット用カメラアプリ Mobile	Employees (Integrating with ... Traditional Web	Omnnext Uploader App Traditional Web	TDI_OSPT_QA Reactive Web	Implementing a Good Timer ... Reactive Web		
Web_demo_20211... Reactive Web	UserManagement Traditional Web	Mobile Study Mobile				
人財育成_Pro課題 Reactive Web		TimeTest Traditional Web	DeployTest_02 Reactive Web	web_demo_20210... Traditional Web		

図 4. アプリ毎の technical debt 解析結果

各アプリの解析結果を確認することで、ベストプラクティスの観点に合わないコード解析パターンごとに、その問題点の「technical debt（技術的負債）」のレベル、及び、その該当箇所を示してくれます。コード解析パターンについては OutSystems 社から示されています。

コード解析パターン：

https://success.outsystems.com/ja-jp/Documentation/11/Managing_the_Applications_Lifecycle/Manage_technical_debt/Code_Patterns

Architecture Dashboard の解析結果に従い修正を行うことで OutSystems 社が示すベストプラクティスを習得することができます。それを応用してアプリ開発の生産性を向上するサイクルを回すことで、保守効率の高いアプリ作成に繋がります。また、解析は Life Time サーバーが起動していれば毎日実施されるため、問題が大きくなる前にアプリを修正することができます。

アプリ間の依存関係分析

アプリ間の依存関係（結合関係）は Forge で取得できる「Discovery」アプリで解析できます。Discovery による解析は、アーキテクチャの検証を行い、ルール違反の検出を支

援します。適切に設計されたアーキテクチャを実現するには、下記 URL に記載されているルールに従うことが推奨されています。

適切に設計されたアーキテクチャの実現 参考 URL :

https://success.outsystems.com/ja-jp/Support/Enterprise_Customers/Maintenance_and_Operations/Designing_the_Architecture_of_Your_OutSystems_Applications/Validating_your_application_architecture

Discovery による解析で、ルールに従ってアプリが構築されているかを自動的に確認し、モジュール間の実際の依存関係を分析することで、ルールに適合しないアプリを特定できます。また、アプリ間の依存関係を視覚的に表示することも可能です。アプリ修正時の影響度調査は、アプリが増えるほど難しくなります。OutSystems では、Discovery ツールを使うことで、アプリ間の依存関係を視覚的に確認でき、保守運用時に役立ちます。

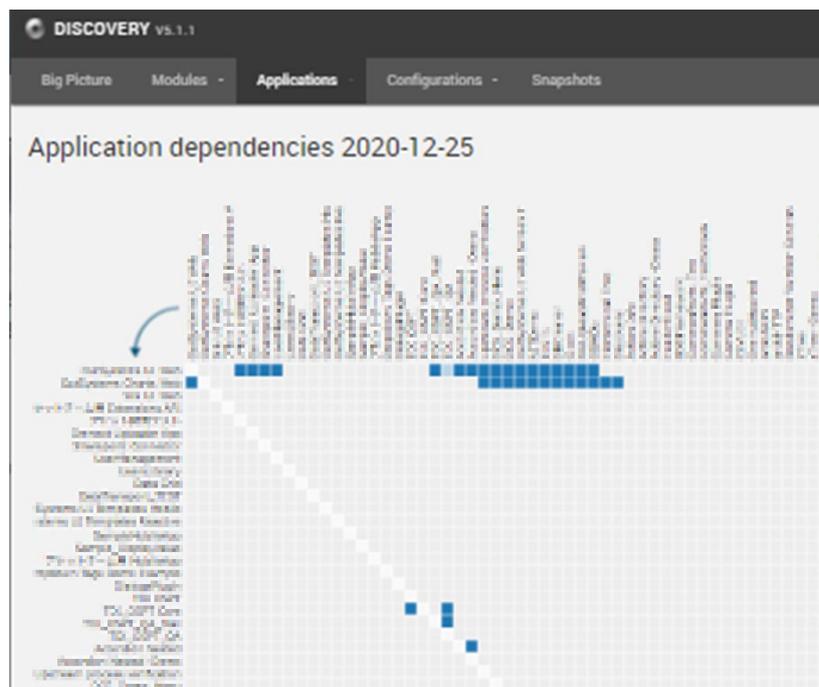


図 5. Discovery による解析結果

アプリ間ができるだけ疎結合であることが、保守性の観点で望ましいとされています。Discovery による解析を利用、確認することで、保守性を考慮したアプリ開発を行うことができます。

作成したアプリのフィードバック機能

「ECT_Provider」と呼ばれる機能を用いることで、作成したアプリに対するユーザーからのフィードバックを受け取れるようになります。画面上の使い勝手などを、ユーザーが直接アプリ上の画面にコメントを書き込めるため、開発者も同じ画面でその内容を確認することができます。従来の、スクリーンショットを取り、Excelなどにコメントを記載して管理する方法より、ユーザーの指摘も分かりやすく、管理の手間が省けます。ユーザーの使用している環境（デバイス、ブラウザなど）の情報も同時に確認できるため、機種依存、ブラウザ依存の問題を特定しやすく、修正をスムーズに行うことができます。



図 6. ユーザーフィードバック機能の利用イメージ

終わりに

本書では、OutSystemsにおける運用、保守機能の一部をご紹介いたしました。ローコード開発製品では、開発フェーズの生産性に着目されることが多いですが、運用、保守の機能の充実度は、社会の変化に対応した柔軟なアプリの提供に不可欠と考えています。

tdiでは、OutSystemsの機能や技術について十分な知識を持った多くの技術者を有しており、資格保有者数は国内トップクラスです。ローコード開発が一般的に注目される以前（2016年）から重ねたOutSystems開発の実績をもとに、IT戦略コンサルティングやOutSystems導入から運用までをトータルサポートします。また、お客様に合わせた人材育成や内製化もご支援いたします。

お困りのご担当者の方は、どうぞお気軽にお問合せください。

※OutSystems® は OutSystems-Software Em Rede S.A.の登録商標です。



【ローコード開発基盤「OutSystems」】

<https://www.tdi.co.jp/outsystems/>

【お問い合わせ】

<https://tdi.smktg.jp/public/application/add/1095>

tdi 情報技術開発株式会社 営業本部

東京: 〒163-1332 東京都新宿区西新宿六丁目5番1号 新宿アイランドタワー32階

TEL.03-5325-4811(代表) FAX.03-5325-4812

中部: 〒451-6027 愛知県名古屋市西区牛島町6番1号 名古屋ルーセントタワー27階

TEL.052-571-6871(代表) FAX.052-571-3856

関西: 〒530-0005 大阪府大阪市北区中之島二丁目2番7号 中之島セントラルタワー20階

TEL.06-6201-7739(代表) FAX.06-6201-7740

九州: 〒812-0013 福岡県福岡市博多区博多駅東二丁目10番1号 福岡ビルS館7階

TEL.092-451-8218(代表) FAX.092-474-7379