

# The Speed of Change:

## 今求められるスピード

新型コロナウイルスの感染拡大の危機と、それに伴うロックダウンの影響により、世界経済と人々の生活は一変しました。「新しい生活様式」が形成されつつありますが、当面の間は混乱が続くことは間違いないでしょう。

これからは、発想力と適応力が組織の運命を左右する重要な要素になります。あらゆる業界の組織で、かつてないスピードで変化が起こっています。OutSystemsは今春、ITプロフェッショナルとシニアITリーダー2,200名を対象に、各組織が変化にどの程度迅速に対応できているのかを調査し、変化への対応力が高い（または低い）要因を探りました。

今、デジタル化が急がれる新たな時代が始まろうとしています。本レポートでは、スピードと適応力において他社をリードしている組織に着目し、そこから得られるインサイトを紹介します。

## 組織の自己評価

調査データの出発点は自己評価です。今回の評価には、変更管理を専門とするProsciの「アジリティ属性評価」に基づくマトリックスを使用しました<sup>1</sup>。回答者には、変化に対する組織の適応力、組織の柔軟性に対する自信、他社と比較した場合の組織のアプローチに関する見解を尋ねました。その結果、自社の俊敏性に自信を持っている回答者はごく一部であることがわかりました。

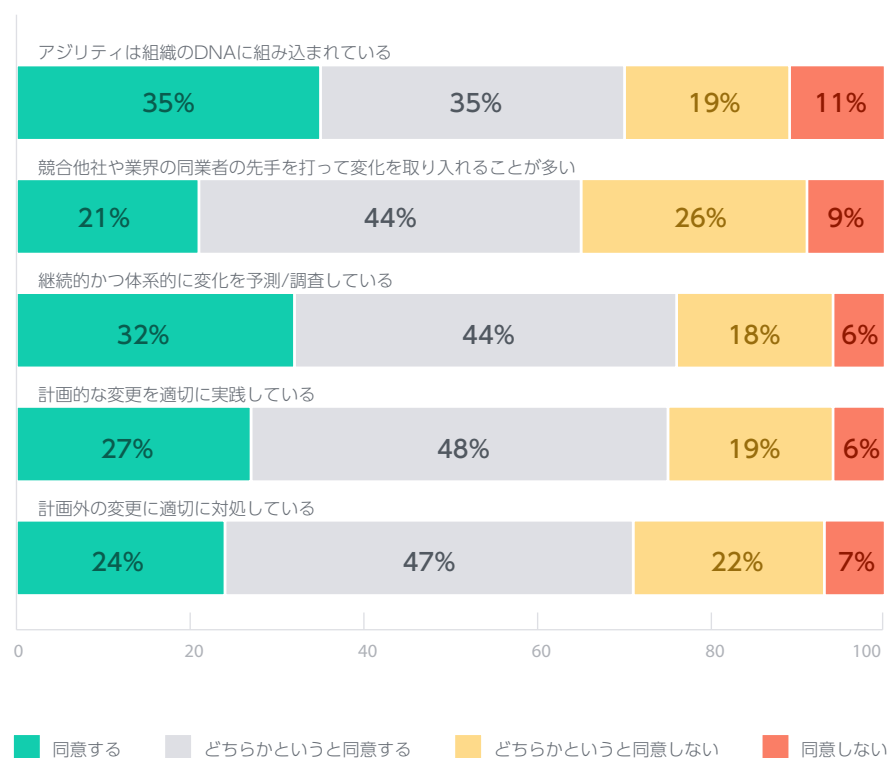


図1：組織のアジリティ評価

注目すべきなのは、他社と比較して変化への対応力が優れていると回答したシニアITリーダーやITプロフェッショナルはわずか21%にとどまっているという点です。

COVID-19による混乱が各方面に広がっている昨今の状況では、計画外の変更が歓迎されないのはなおさらです。そのことは、変更に対する自信を持つ回答者が4分の1に満たないという数字にも表れています。

アプリケーションの迅速かつ柔軟なデリバリーを実現する方法は多くありますが、今回は変化に対する取り組みを評価する指標として、IT組織におけるアジャイル開発手法の導入レベルを回答してもらいました。評価には以下の5つの成熟度を設定しました。

- 1 初期段階：**一貫性に欠けており、全員の連携をとるためのトレーニングが必要である。
- 2 開始間もない段階：**プロセスが完全に定義されていない。アジャイル導入の基本レベル。
- 3 定義された段階：**チーム全体がアジャイルを使用している。スプリントごとに継続的なデリバリーを実現している。
- 4 定量的に管理された段階：**コード品質などの主要指標を測定している。
- 5 最適化された段階：**KPIに基づいて、持続可能で継続的な改善を行っている。

調査の結果、「定量的に管理された段階」または「最適化された段階」に達している企業は25%未満であり、過半数は取り組みの「初期段階」または「開始間もない段階」にあることが明らかになりました。つまり、先進的な組織がアジャイルを導入してさらに先へと進んでいるのに対し、圧倒的多数のIT部門は、いまだに旧来の方法を用いてデリバリーの高速化に向けて変革を模索している段階にあるのです。

<sup>1</sup>Prosci「アジリティ属性評価」

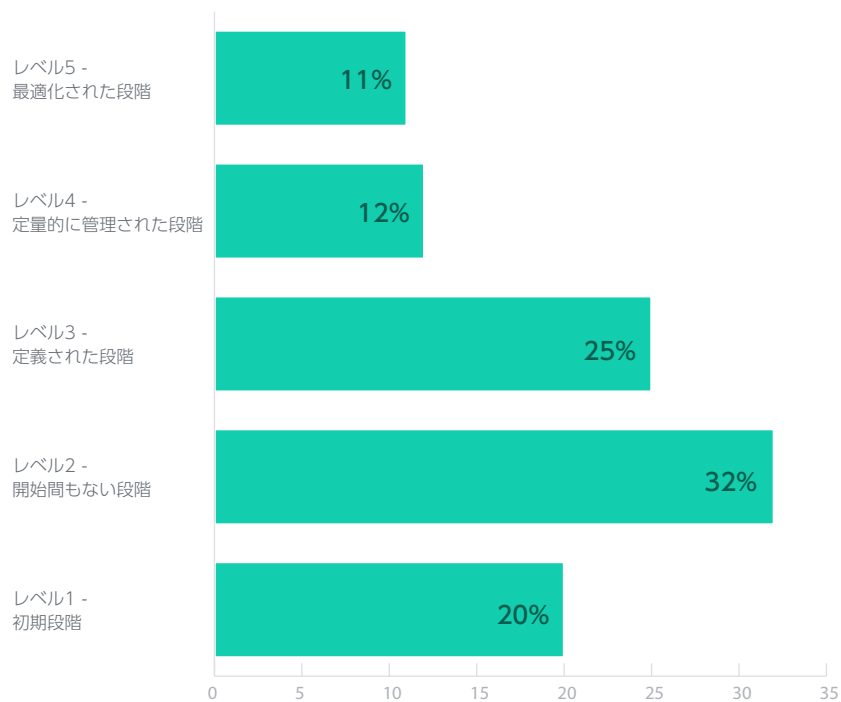


図2：アジャイルの成熟度

## 開発期間の実態

会議で経営陣から「デリバリーに時間がかかりすぎている」と指摘された経験は誰にでもあるのではないのでしょうか。そこで本調査では、組織が新しいWebアプリケーションやモバイルアプリケーションのデリバリーに要する平均時間を回答者に尋ねました。調査結果によると、デリバリーのタイムラインと世界情勢との間には大きなずれが生じています。データを見ると、現在進行中のプロジェクトの65～75%は、COVID-19対策として米国全体でロックダウンが実施されるよりも前に開始されていたということが読み取れます。これはある意味では、各組織が先見の明を持ってプロジェクトを早めに開始していたといえるかもしれません。

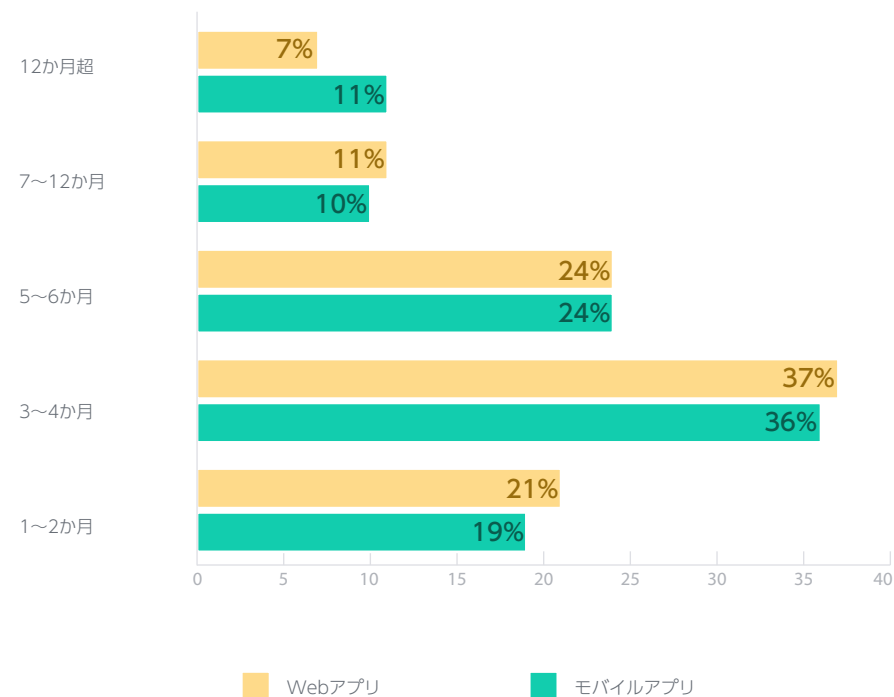


図3：アプリケーションの開発期間

とはいえ、アプリケーションのデリバリーに3～6か月を要するというのは、今の時勢には合わないように感じられます。しかし、これが過半数の組織の実情なのです。すべてのIT組織は、目まぐるしく変化する状況と喫緊の重要課題に対応するために、開発期間の短縮方法を徹底的に検討する必要があります。

では、直近1か月間でソフトウェアプロジェクトの開始から終了まで遂行した数少ない企業とはどのような組織でしょうか。また、それほどのスピードを実現したソフトウェア開発のアプローチとはどのようなものなのでしょうか。本調査では、こうした質問に対する回答に基づいて、組織を「リーダー」と「ラガード」に分類しました。

## リーダーとラガードの比較

リーダーは、変化への対応力に自信を持っています。アジャイルのような開発手法の導入を着実に推進し、ソフトウェアプロジェクトのデリバリーを短期間で実現しています。また、ソフトウェアのアップデートやリリースを毎日または週に複数回行っています。リーダーとラガードとの間には明らかに開きがあり、そこから学ぶことでチームのスピードと変化への対応力を高めることができます。

### デリバリーを高速化するためのアプローチとテクノロジー

表1と表2は、アプリケーションのデリバリーを高速化するために、リーダーとラガードが最近投資したアプローチとテクノロジーを示したものです。

| アプリ開発を高速化するためのアプローチへの投資 | ラガード | リーダー | 差    |
|-------------------------|------|------|------|
| デザイン思考/デザインスプリント        | 13%  | 50%  | 37%  |
| カスタマージャーニーマッピング         | 11%  | 37%  | 26%  |
| アジャイル                   | 32%  | 56%  | 24%  |
| DevOpsまたはSecDevOps      | 22%  | 45%  | 23%  |
| Lean UX                 | 6%   | 13%  | 7%   |
| どれも該当しない                | 29%  | 10%  | -19% |
| 開発の部分的なアウトソーシング         | 47%  | 21%  | -26% |

表1：アプリ開発を高速化するためのアプローチへの投資

| アプリ開発を高速化するためのテクノロジーへの投資 | ラガード | リーダー | 差   |
|--------------------------|------|------|-----|
| 新しいプログラミング言語またはフレームワーク   | 15%  | 29%  | 14% |
| コンテナ/マイクロサービス            | 15%  | 29%  | 14% |
| ローコード + MXDプラットフォーム      | 30%  | 44%  | 14% |
| デジタルプロセスオートメーション/RPA     | 24%  | 29%  | 5%  |
| BPMプラットフォーム              | 7%   | 11%  | 4%  |
| クラウド                     | 58%  | 61%  | 3%  |

表2：アプリ開発を高速化するためのテクノロジーへの投資

リーダーは、デザイン思考やデザインスプリント、カスタマージャーニーマッピングなど、顧客中心かつエクスペリエンスを重視したアプローチに重点的に投資しています。リーダーにおいては、DevOps、コンテナ、マイクロサービス、ローコードのほか、新しいプログラミング言語やフレームワークに投資する割合がラガードの2倍にのぼっています。

ラガードはアジャイル開発手法に関心を持っており、RPAへの投資はリーダーに近い割合になっているものの、その他の項目における差がパフォーマンス上の格差につながっています。これは次の3つの点で顕著に表れています。

### アプリケーション開発のイノベーションとバックログ

リーダーが他社に差をつけている1つ目の要因は、古いアプリケーションの維持やリプレースに時間をかけるのではなく、新しいアプリケーションの構築に時間を割いている点です。一方、ラガードは機能の修正や「現状維持」のための保守、様々な技術的負債に多くの時間を割いていると回答しています。

ラガードの大半はアプリケーションのバックログにほとんど変化がないと述べている一方、3分の1近くは増加したと答えています。また、アジリティリーダーにおいては、バックログの削減に成功している割合が高くなっています。

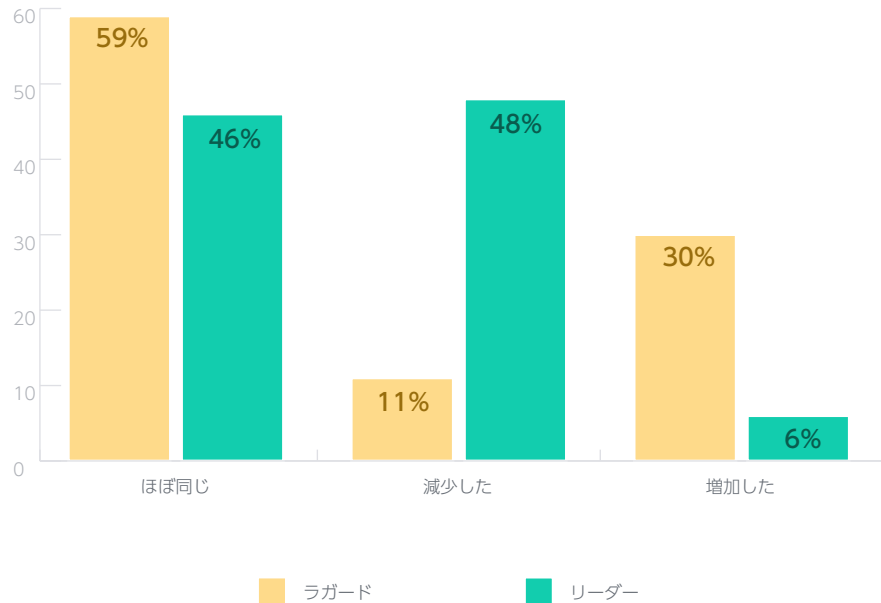


図4：直近1年間でのアプリ開発のバックログ

イノベーションとバックログに関してこれほど明白な違いが生じているのはなぜでしょうか。既存のソフトウェアやレガシーソフトウェアを維持するプロセスには多大な労力を要します。集中的な取り組みやスキルを必要とし、コストも膨大にかかるため、イノベーションに回せるIT予算が少なくなります。そしてそのしわ寄せがアプリケーションのバックログにいくのです。一方、リーダーには変更や更新をすばやく処理するための手法やツール、先進的なアーキテクチャがあります。そのため、バックログの削減に手間取らずに、アプリケーションの新規開発に取り組むことができます。

## アプリ開発のスピードと頻度

各組織においてアプリケーションをデリバリーする標準的な開発スピードが過去1年間で向上したかどうかを問う質問に対して、1年前と比べて「やや向上した」または「とても向上した」と答えたリーダーの割合はラガードの3倍近くにのぼりました。

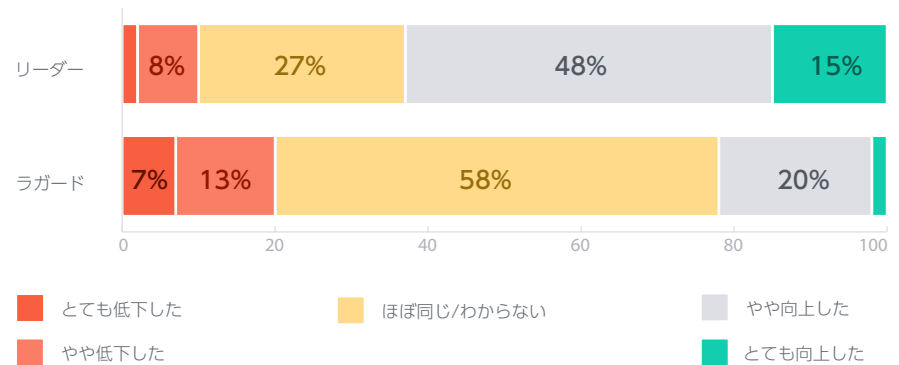


図5：直近1年間でのアプリ開発のスピード

リーダーは、スピードを高めるための新しいツールや手法に注力することで、アジリティへの投資を効率化し、継続的な改善を行い、開発のさらなる高速化を実現しています。

ラガードはソフトウェアの保守に非常に多くの時間を費やしています。また、ツールやスキル、プロセスが欠如しているため、開発の加速に苦慮しています。技術的負債はイノベーションにとって重荷となります。一度後れをとってしまうと、現状を変えない限り挽回するのは難しくなります。ラガードの中には、何とか対応して1年前よりも開発を高速化しているケースもありますが、それがいつまで続くかはわかりません。

## アプリ開発の人材とスキル

人材のスキル不足と適性の不一致は、アジリティに必要となる先進的なITアーキテクチャを迅速に採用するうえで妨げとなります。これはリーダーとラガードの両方に影響を及ぼしていますが、フルスタック開発者の採用が「難しい」または「とても難しい」と回答したラガードの割合は、リーダーと比べて21%も多くなっています。

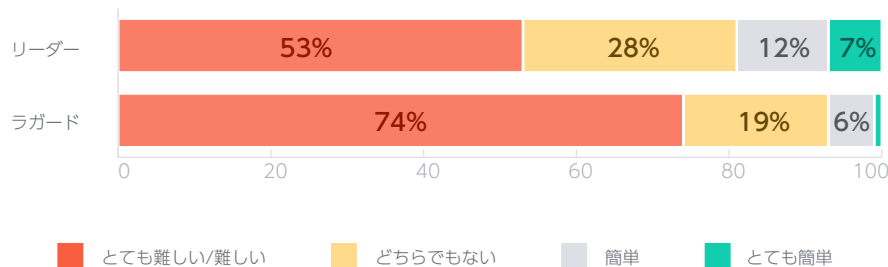


図6：フルスタック開発者の採用の難しさ

リーダーにとって、フルスタック開発者の採用はそれほど難しいことではありません。これには、使用するアプローチやツールのほか、業務内容が関係しています。つまり、リーダーの場合、レガシーシステムの維持よりも、新しいプロジェクトに取り組む機会が多いのです。このような情報は開発者コミュニティにすぐに広がり、リーダーの採用活動を後押しすることになります。

## リーダーか、ラガードか

本調査で使用した項目をチェックすることで、自分の組織がリーダーかラガードかを判断することができます。この1年をめどに達成すべき目標を踏まえ、組織の現状を確認しましょう。現在、アプリケーションやソリューションのデリバリーをどれだけ迅速に実行できているでしょうか。競合他社と比較してどうでしょうか。また、開発を高速化し、精度を高めるためにどのようなツールや手法を使用しているでしょうか。

こうした質問に対する回答を、本レポートに取り上げたリーダーの特徴と比較してみてください。リーダーに分類されるのは少数であり、ほとんどの組織はその分類から外れます。しかし、あきらめる必要はありません。今からでもリーダーになる方法があるからです。

## リーダーに追いつくには

この新たな時代においては、デジタルファーストとクラウドファーストを念頭に置いた変革が喫緊の課題となっています。では、どのようにすればスピードとアジリティを高めることができるのでしょうか。その答えを探るために、ここではWeb/モバイルアプリケーションの新規開発に関して回答者から挙げられた4つの主要課題に注目します。図7を見ると、上位2つの課題はラガードとリーダーで共通しています。

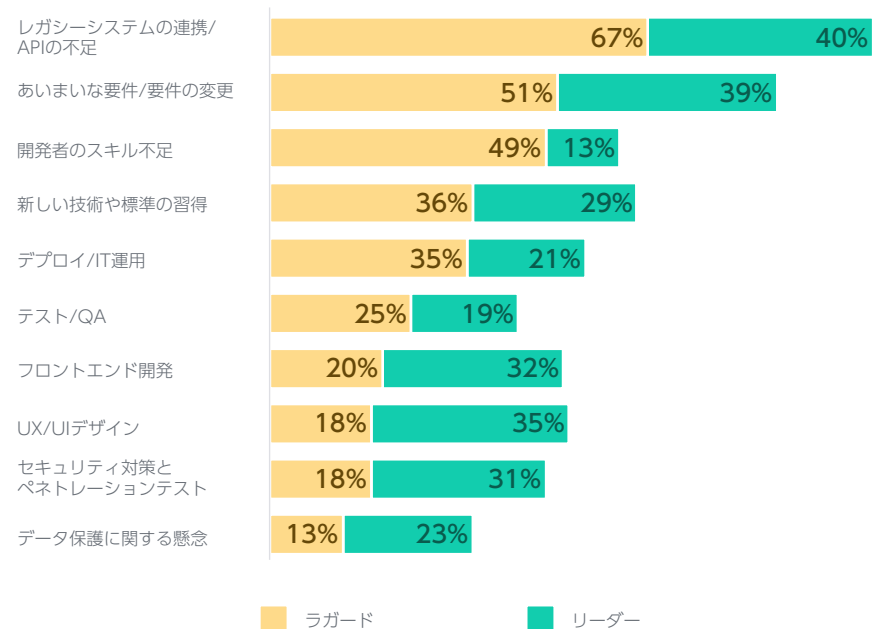


図7：アプリケーションデリバリーの遅延を引き起こす主な原因

しかし、他の課題を見ていくと両者の違いが明らかになります。ラガードの2倍近い数のリーダーが、UX/UIデザインを最大の課題として挙げています。リーダーのもう1つの主要課題はフロントエンドです。一方、開発者のスキル不足を主要課題の1つに挙げるラガードの数は、リーダーの4倍近くにのぼっています。ラガードの主要課題として4番目に多かったのは、新しい技術や標準の習得です。

ラガードの課題は内部的な問題を反映しています。人材の新規採用や、新しい言語やフレームワークに関連するトレーニングに時間を奪われ、アプリケーションを迅速に顧客に提供できていないのです。これに対して、リーダーは顧客とユーザーエクスペリエンスに意識を向けています。リーダーの目標は、顧客への価値提供を可能な限り早めることです。これを実現するには、次のような取り組みが有効です。



#### UXから着手する

カスタマージャーニーマッピングやデザインスプリントを使用し、ユーザーを中心とした開発プロセスを実施します。



#### 変更を想定して構築する

反復的なアジャイル開発手法を採用し、不確実性や不明確な方向性、要求の変更などに対応できるようにします。



#### 新しいスキルを追加する

Web、モバイルバックエンド、最新のスタックなど、チームが今後必要とするスキルを模索します。



#### 継続的デリバリーに重点を置く

テクノロジーを適宜導入し、チームが一連のDevOpsツールやスキルをそろえなくても継続的デリバリーを実現できるようにします。



#### レガシーシステムと連携する

ビルトイン/DIYコネクタを備えたツールを活用し、エンタープライズシステムやデータベース、Webサービスと簡単に連携できるようにします。

## まとめ

今、スピードが至上命題になっています。顧客のニーズに目を向け、開発者をイノベーションへとすばやく導くことのできる組織は、変化に伴う痛みにも対処できます。少数ながら、すでにその段階に達している組織は、さらなる高速化と改善のために力を尽くしています。その他多くの組織には取り組むべき課題が残されていますが、解決の糸口は確実にあります。リーダーであってラガードであっても、スピードをただ競うだけでなく、その競争を勝ち抜いていくことが可能なのです。

## 次のステップ

アジリティリーダーは、顧客とユーザーエクスペリエンスに強い関心を抱いています。アジリティリーダーになるには、ビジネスと顧客への価値提供をさらに意識する必要があります。顧客中心思考を身に付けるためのアプローチやテクノロジーは様々です。まずは、どのような選択肢があるのかを把握しましょう。

そのための第一歩として、ここでは以下の3つのリソースを紹介します。



「Digital Urgency: Agile Leaders in Action」 (ブログ)  
[記事を読む](#)



「Ready or Not: The Sudden Urgency of Digital」 (ウェビナー)  
[ウェビナーを見る](#)



Personal Editionのサインアップ  
[OutSystemsの無償トライアル](#)

## OutSystemsについて

OutSystemsは、アプリケーションポートフォリオ全体でビジュアル開発を実現し、既存のシステムとも容易に統合できる唯一のソリューションとして、世界中のお客様から信頼を獲得しています。詳細については、[www.outsystems.com/ja-jp](http://www.outsystems.com/ja-jp)をご覧ください。