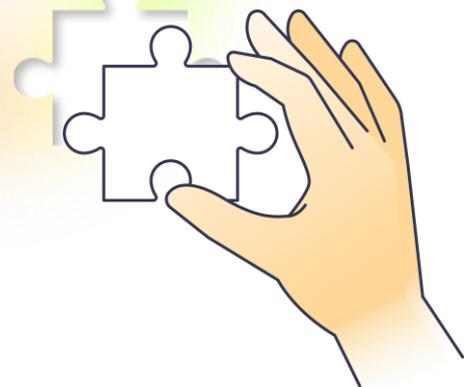


あなたが創る、豊かな未来

—Software-Defined Societyの実現に向かって—



IPA 独立行政法人
情報処理推進機構

はじめに

人口減少、高齢化、気候変動、エネルギー問題・・・私たちの社会は今、大きな転換点を迎えています。これらの課題に立ち向かい、豊かで持続可能な社会を実現するカギは「ソフトウェア」にあります。ソフトウェアの力を最大限引き出すのは、特別な誰かではなく、あなた自身です。

この資料の目的



この資料では、来るべきソフトウェア社会の中で私たち一人一人が実践しなければならないこと、そしてソフトウェアが実現する新しい社会「Software-Defined Society」について紹介しています。この資料を読むことで、あなたの日常の行動が、あなたの声が、豊かな未来を創るのだということに気づいてほしいと願っています。

この資料の対象読者

この資料は、日本の将来に関心がある社会人を対象としています。IT技術に詳しくない方でも理解できるように、身近な例をもとに解説しています。



それでは一緒に、豊かな未来を創るための一歩を踏み出していきましょう。

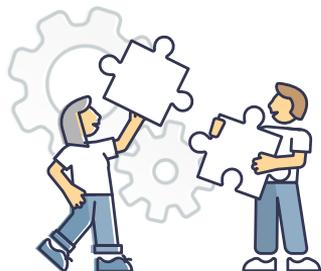


CONTENTS

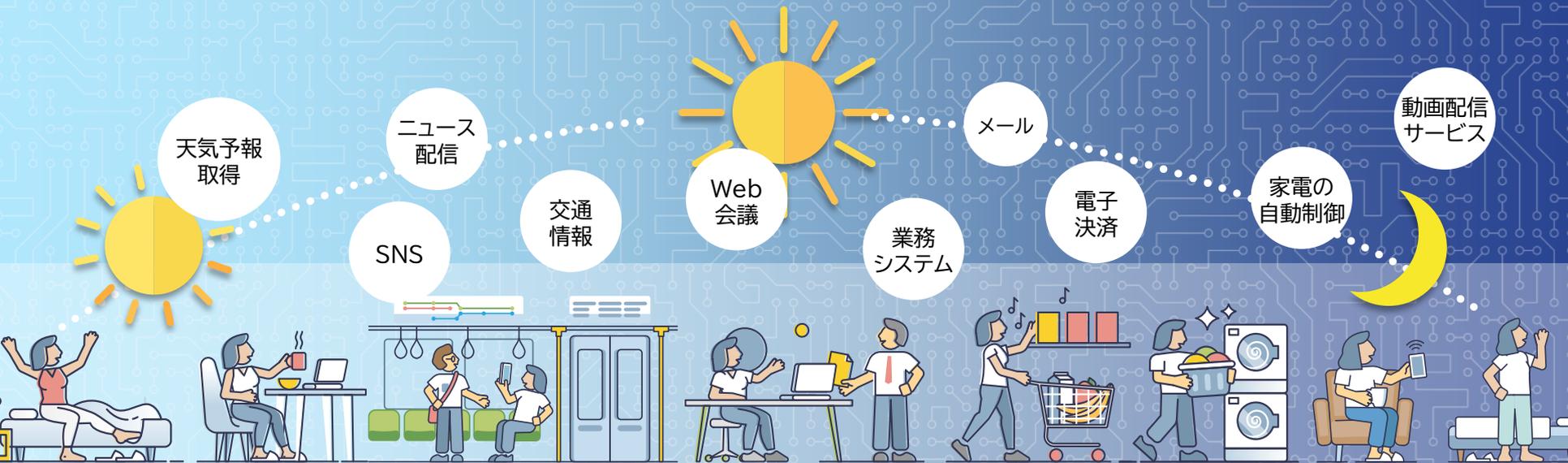
1. あなたの毎日を支えるソフトウェア
2. 変化し続ける社会の中で
3. アジリティの実現に向けて
4. ソフトウェアが導くしなやかな社会
～Software-Defined Society～
5. 私たちで創る、豊かな未来

1

あなたの毎日を支えるソフトウェア



私たちの日常生活は、気づかないうちに数多くのソフトウェアに支えられています。



「買ったら終わり」ではなく「買ってから進化」する時代

ソフトウェアといっても、業務用のシステムやスマホのアプリだけではありません。例えば自動車。購入時点で性能や機能が固定されるのではなく、購入後も新しい機能を追加したり性能を向上させたりできる自動車もあります。

例えば…

従来の自動車



ハードウェア主導の設計

購入時点で性能・機能が決まり、機能追加には部品交換やディーラーでの修理が必要。

最新の自動車(SDV※)

※ Software-Defined Vehicleの略。



ソフトウェア主導の設計

ソフトウェアをアップデートすることで、リモートで機能追加や性能向上が可能。

価値を生み出し続けるソフトウェア

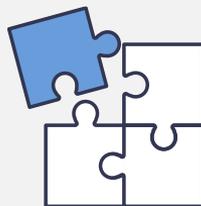
なぜこれほどまでにソフトウェアが広く活用されているのでしょうか。それは、ソフトウェアには価値を生み出し続ける3つの特長があるからです。

ソフトウェアの3つの特長



使っても減らない

何度使っても劣化せず、同時に何人でも使用できます。コピーも簡単です。



柔軟に改変可能

物理的な製造工程が不要です。不具合の修正はもちろん、機能の追加や修正も素早くできます。



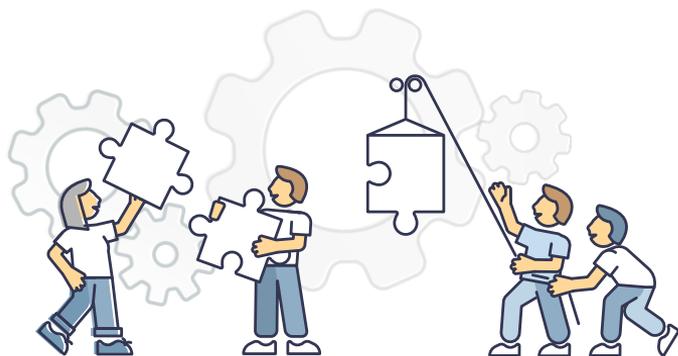
継続的な改善

データを収集しながら現場の声を繰り返し反映することで、継続的に機能を改善できます。

このソフトウェアの特長が、
昨今の変化の激しい時代を生き抜くためのカギになります。

2

変化し続ける社会の中で



予測困難な時代

現代は「VUCA(変動性・不確実性・複雑性・曖昧性)の時代」と言われ、変化が激しく将来の予測が困難です。ほんの10年前には予測できなかった変化が、今、当たり前の日常になっています。

VUCAとは…

Volatility

変動性

激しく予測不能な変化

Uncertainty

不確実性

将来の予測が困難

Complexity

複雑性

様々な要因が複雑に絡み合う

Ambiguity

曖昧性

因果関係が不明確

10年前には予測できなかった変化



働き方革命

- ・在宅勤務が当たり前。
- ・オフィス不要の働き方。



キャッシュレス社会

- ・スマホ決済が主流。
- ・外出の際でも財布を持たない生活。



AI時代

- ・スマートデバイスとの会話が日常に。
- ・生成AIが仕事のパートナー。

長期的な見通しをもとに計画することが困難

変化の激しいVUCAの時代では、詳細に計画を立て、完璧を目指すという従来の計画重視のアプローチは遅すぎます。計画している間に状況が変わり、完成したときには既に時代遅れになってしまいます。

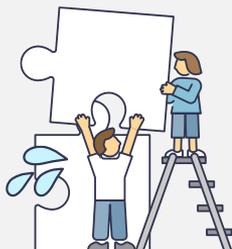
従来の計画重視のアプローチ

社会情勢の分析



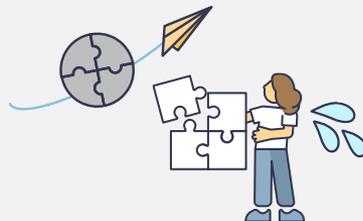
社会情勢の変動が激しくて、
長期での見通しが立たない…

詳細な計画の立案



計画中に状況が
ドンドン変わってしまう！

計画通りに実行



完璧な製品にこだわっているうちに
海外企業が先行している…

完成品を市場投入



ついに市場投入！だけど時代遅れで、
仕様変更できない…

「これまでこうしてきたから」は、もう通用しません。

VUCA時代に求められるものは

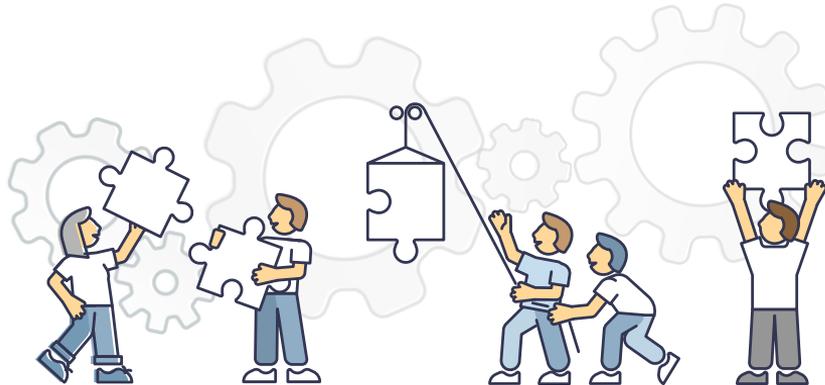
変化の激しいVUCAの時代に求められるもの、それは状況に合わせて柔軟に素早く対応する力、すなわちアジリティ(機敏性)です。アジリティの実現には、変化を前提に考え、仮説ベースで素早く実行し、多様な関係者との対話を通して共創する力が不可欠です。



高いアジリティによってVUCA時代の価値を創出

3

アジリティの実現に向けて



アジリティを実現する考え方

高いアジリティを持つ変化に強い社会を実現するには、アジリティを実現する技術をベースとした3つの考え方が重要です。これらは政府や経営者など一部の人だけでなく、社会を構成する私たち一人一人が意識し、日々の行動の中で実践する必要があります。

変化に強い社会



一人一人が意識・実践

変化を前提に

将来的な変化を前提に設計します。



まずは仮説ベースで

完璧なモノを目指すのではなく、まずは仮説ベースで実装します。



対話による共創

多様な関係者との対話を通してモノの価値を進化させます。



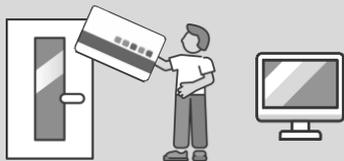
アジリティを実現する技術

変化を前提に

モノを作るときには、将来的な変化を前提に考えます。作った時点で機能を決定、固定するのではなく、変化に合わせて柔軟に機能を追加、更新できるようにすることで、変化に強く柔軟な社会を実現します。

会社の入退室管理の例

現状のセキュリティゲート



将来的な変化を想定せず、カードキーを配布してハードウェアで管理

- 頻繁に人員が増減するとカードキーの管理が大変。
- 想定していなかった時間外シフトが発生すると自由に入室できない。

変化に弱く、柔軟性がありません

これから



- ① 入退室する人員などの変化を前提に検討しよう
- ② ソフトウェアを活用して動的に管理できるようにしよう

例えば…



スマホアプリによる入退室管理

Software-Defined

「Software-Defined」とは、ハードウェアに依存していた機能や制御をソフトウェアで実現し、柔軟に更新・拡張できるようにする考え方です。ソフトウェアを継続的に更新し続けることで、不確実性や変化するニーズに柔軟かつ俊敏に対応できます。1.2の「最新の自動車(SDV)」も「Software-Defined」の一例です。

Software-Definedの考え方で、変化に強く柔軟な社会を実現

まずは仮説ベースで

最初から完璧なモノを目指すのではなく、まずは仮説ベースでいいので、素早く実装します。実装した結果を実際に使ってもらい、改善点があれば次の改善につなげます。このサイクルを高速で回すことが、価値を生み出す社会につながります。



仮説と検証を高速で繰り返すことで、価値を創出

対話による共創

様々な価値観を持つ人々との対話が、自分たちだけでは生み出せない価値の創出につながります。自組織内はもちろん、他業種との対話や利用者からのフィードバック・・・あなたの一声が改善につながり、社会を変える、それが新しい時代の在り方です。



他業種

(バンダーなど)



対話



提供者

(企業・公官庁、など)



**フィードバック
／モニタリング**



利用者

他業種との 対話の例

- 他業種間コラボレーションによる新しいサービスの創出
- 自社とは別の価値を提供するバンダーとの共創活動の推進

利用者からの フィードバックの例

- 新製品のアンケートに回答
- ユーザーコミュニティに参加して開発者と対話
- 応援したいクリエイターやプロジェクトに少額の支援(クラウドファンディング)

利用者の モニタリングの例

- ストリーミング配信の視聴率を自動で収集
- 駅や店舗などでの人間の動線をセンサーで検知

創る人と使う人が、一体となって価値を創出、向上

アジリティを実現する技術

価値創出サイクルを回すために、ソフトウェア開発の分野では様々な技術が活用されています。これらの技術を知っておくことが、アジリティをより一層高めることにつながります。

アジャイル

変化に合わせてこまめに改善

小さく試して学び、こまめに作り直していく進め方です。「一度決めて終わり」ではなく、変化に合わせて更新し続けるという考え方は、アジリティの土台です。

DevOps

作る人と使う人の団結

作る人と使う人が一体となって改善していくという考え方です。運用しながら機能変更を素早く安全に届けるといふ、対話が重要な仕組みです。

モデリング

仕組みの見える化

複雑なことを「要素」と「関係」に整理・可視化する方法です。目に見えることで設計のブレを減らし、共通理解を得やすくします。

デジタルツイン

デジタル空間での仮説検証

現実の状態をデジタル上で再現し、状況を把握しやすくする考え方です。デジタル上で仮説を検証できるので、改善や最適化を安心して進められます。

ビルディングブロック

部品の交換・組合せ

よく使う機能やルールを「部品」としてそろえ、組み合わせてモノを作る考え方です。変更があってもゼロから作り直さずに済み、修正や機能拡張を素早く安全に実施できます。

AI活用

人間の作業・判断のサポート

様々な工程やタスクで、人の作業や判断を補助する技術です。人間の負担を減らして改善のスピードを上げることで、変化への対応力が高まります。

データ利活用

データに基づく判断

データを集めて整理し、状況を理解するための根拠とする取り組みです。データに基づいて判断することで、意思決定と改善を適切に実施できます。

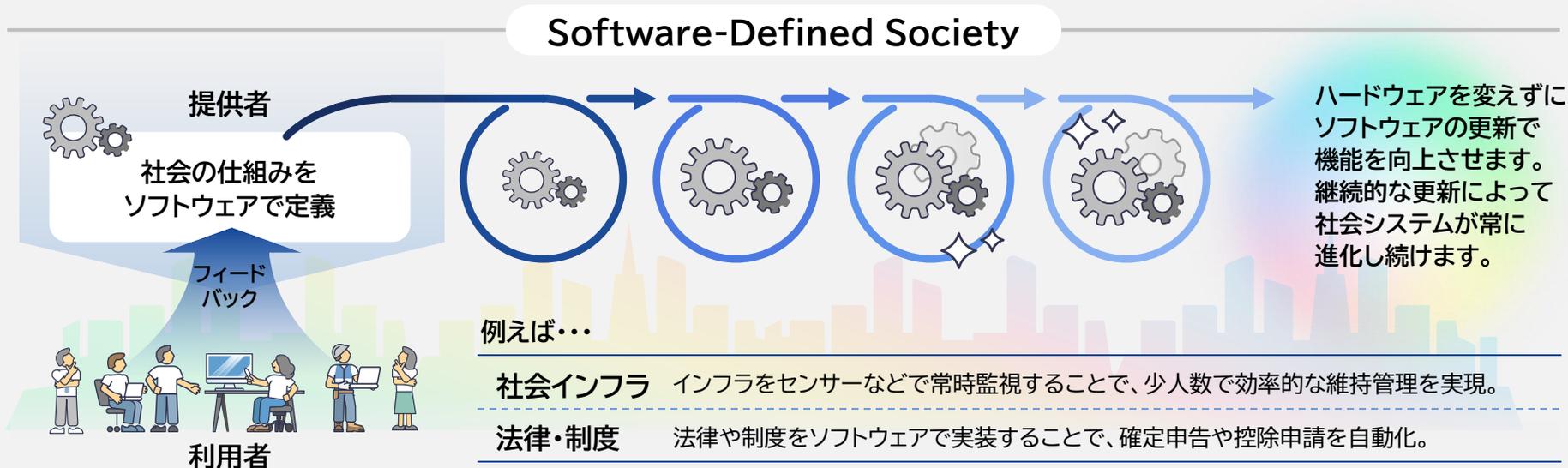
4

ソフトウェアが導くしなやかな社会
~Software-Defined Society~



Software-Defined Society(SDS)とは？

Software-Defined Society(SDS)とは、Software-Definedと「変化を前提に」「まずは仮説ベースで」「対話による協創」の『3つの柱』を社会全体に広げた、アジリティの高い社会のことです。



社会制度や環境が変化しても、状況に応じて最適なサービスが届く社会に

4.2 社会インフラのSDS革命

インフラをソフトウェアで管理する時代へ

今後は、橋梁やトンネルなどのインフラもソフトウェアで管理する時代になります。将来的な人手不足が懸念される日本だからこそ、SDSの実現による効率的な維持・管理が必要です。

従来

まずはインフラを設計・建設します。管理方法は後から検討するため、センサーが必要になったら後付けで対応します。



インフラ建設当時の管理技術がベースの設計になるため、時代の流れとともに管理技術と設計がちぐはぐになります。

これから～50年後を見据えた管理

ソフトウェアでインフラをどう管理するかを先行して検討します。インフラそのものは長く使用しますが、ソフトウェアは随時アップデートし、最新のものに更新し続けます。



Point

最新技術の適用

インフラの設計・建設時には存在しなかった最新技術を、ソフトウェアアップデートによって適用します。

事故の未然防止

インフラのデータを収集してソフトウェアで解析することで、事故の発生を予防できます。

具体例

遠隔監視(IoTセンサー):橋梁・トンネルにセンサーを取り付けてデータを収集。劣化をリアルタイムで検知することで迅速な異常検知や予兆保全を可能にします。

AI予兆保全:収集したデータをAIで解析し、故障を事前に予測。補修コストの削減やインフラの長寿命化を実現します。

ハードウェアは変わらずとも、ソフトウェアで進化するインフラへ

4.3 法律・制度のSDS革命

法律もソフトウェアで

法律や制度を制定する段階で、ソフトウェアによる実装を前提に条文を設計します。法律をソフトウェアで運用することで、法改正による事前シミュレーションや個人最適化が可能になり、面倒な申請が不要になる未来が到来します。

従来

法律は人間による運用が前提。文字の状態から人間が内容を解釈し、そこからシステム開発していくには開発期間に数年が必要。



これから

法律・制度を、ソフトウェアで実装することを前提に設計。法律自体がソフトウェアに適した形で制定されるため、システム開発が容易になります。



プログラム化した法律



システム

Point

事前シミュレーション

法律改正の際に、その変更によって誰が、どれだけ影響を受けるか、を事前にシミュレーションできます。

個人最適化

個人の状況に応じて控除や支援が自動で適用できます。確定申告など個別に対応が必要だった処理が自動適用されます。

迅速な制度改善

法律改正によるシステム開発の期間を短縮。さらに、データをもとに継続的に制度の最適化を実施できます。

具体例

- 税制: 収入や家族構成から最適な控除が自動計算。
- 社会保障: 自分が受けられる支援が自動通知。
- 補助金: 条件を満たせば自動適用。

法律のプログラム化で、最適、柔軟、効率的な制度運用を実現

5

私たちが創る、豊かな未来



あなたに求められること

基本となる3つの柱を個々人が実現していくためには、自ら学び、挑戦し、成長していく意識が必要です。毎日の生活の中で、少しずつ意識を変えていきましょう。

変化を前提に

変化はチャンス

変化は必ず起こるものと考えて行動します。変化を恐れてはいけません。むしろ、改善のためのチャンスととらえてください。

発想の転換

モノを作るときは、まずは目的や機能をどうやってソフトウェアで実装するか、から考えます。ハードウェアによる制限にとらわれず、自由に発想してください。



まずは仮説ベースで

脱・完璧主義

完璧なモノを目指す、という完璧主義にとらわれてはなりません。提供する側は仮説をもとに素早く実装し、利用する側は不完全でも許容する、この考え方が大切です。

素早い検証と改善

実装された結果を実際に使ってもらい、その結果をもとに素早く改善します。検証、改善のサイクルこそが、価値を生み出します。



対話による共創

フラットなコミュニケーション

多様な関係者と率直・真摯・フラットにコミュニケーションをとってください。対話こそが、価値を進化させるカギになります。

積極的なフィードバック

受け身の姿勢でモノを使うだけの時代は終わりました。能動的にフィードバックを行い、共に育てるという意識こそが豊かな未来につながります。



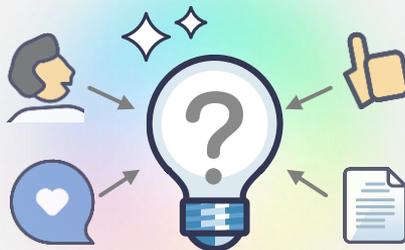
今日からできること

豊かな未来の実現に向けて、毎日の生活の中で、少しずつでも意識を変えていくことが重要です。ただ待つのではなく、動くべきは今このときです。



変化を前提に

- 顧客への提案資料をソフトウェアで管理します。製品仕様に変更があってもパーツの修正で最新化できるようになります。
- 問合せ用のAIチャットボットの実装を検討します。日々変化する問合せに自動で応答できるようになります。



まずは仮説ベースで

- アンケートやヒアリングなどから得た顧客の声を、製品・サービスの開発部門と一緒に分析します。
- すべての機能をそろえてからリリースするのではなく、仮説をもとにして、最も効果のある機能から段階的にリリースします。



対話による共創

- 他部署と共同で製品・サービスを開発します。お互いの業務を理解し合い、本当に必要な機能を一緒に考えます。
- 社内の業務システムが使いにくいとき、情報システム部門に具体的な改善案をフィードバックします。

10年後、あなたはどんな社会で暮らしたいですか？

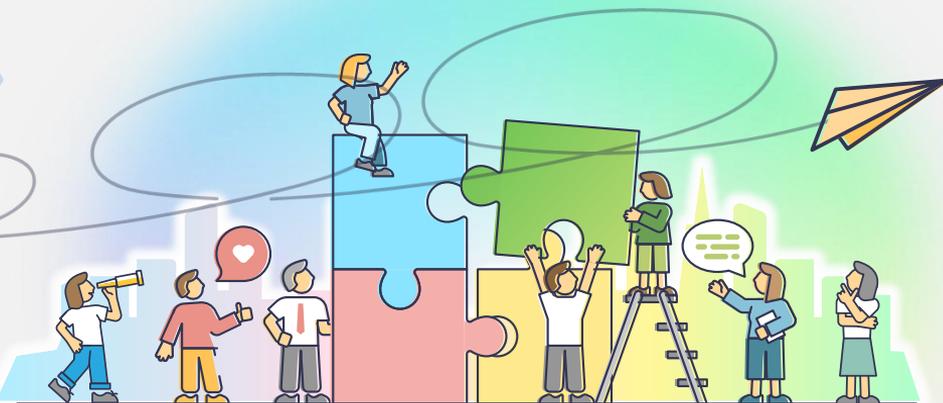
高いアジリティ

- 将来の変化を前提に検討、設計
- 多様な関係者と対話
- 仮説ベースで素早く改善

- ✓ 自ら学び、挑戦し、成長していく意識
- ✓ よりよい社会をただ待つのではなく、自ら行動

日本ならではの強み

- 品質へのこだわり
- おもてなしの文化
- 対話できる社会
- 現場主導のカイゼン



豊かな社会の実現

その未来は、今日のあなたから始まります。



〒113-6591 東京都文京区本駒込二丁目28番8号 文京グリーンコートセンターオフィス(総合受付13階)

さらに詳しい情報に関しては、次のウェブサイトで御確認ください。



[システム／ソフトウェア開発の革新 | 社会・産業のデジタル変革 | IPA 独立行政法人 情報処理推進機構](#)



[ソフトウェアモダナイゼーション委員会 | デジタル基盤センター | IPA 独立行政法人 情報処理推進機構](#)

IPA デジタル基盤センター
デジタルエンジニアリング部
ソフトウェアエンジニアリンググループ
E-mail:disc-info@ipa.go.jp