

# アジャイルにおける価値共創プロセス

## —「交換価値」から「使用価値」へ—

平井直樹

Value co-creation process in agile:  
From “value in exchange” to “value in use”

HIRAI, Naoki

本研究では、アジャイルの特徴の一つである顧客との共創とそこで生じる価値について、価値共創に関する研究、とりわけマーケティング分野で研究が進んでいる G-D ロジック、S-D ロジックを元に明らかにすることを目的としている。

これまでのソフトウェア開発であるウォーターフォール・モデルでは、当初の計画通りに完成したソフトウェアを顧客に受け渡すことを重要視しており、G-D ロジックで説明される交換価値の最大化を目指すものと捉えることができる。一方、アジャイルは、単に素早く開発を行うことではなく、反復活動を通じたフィードバックなどの顧客の参画の度合いが強く、顧客と共に、そのビジネスの成長に合わせて最初から最後まで価値のある製品やサービスを継続的に提供し続ける。製品の提供よりも実際にそのソフトウェアを利用し、顧客が目的を果たすことができること、つまりビジネス上の価値を生むことが重要であり、S-D ロジックで説明される使用価値の最大化であり、そこに至る共創プロセスでは文脈価値が生じていると捉えることができる。

アジャイルは、フィードバックといった顧客の参画の度合いが強く、ゴールを共有しており、そのため、人と人とのコミュニケーションやコラボレーション、「共創」を重視し、顧客の利用まで見据えた使用価値とそれに伴う文脈価値を実現しているのである。

キーワード：アジャイル（Agile）、共創（Co-Creation）、サービス・ドミナント・ロジック（Service-Dominant (S-D) Logic）、グッズ・ドミナント・ロジック（Goods-Dominant (G-D) Logic）

## 1. はじめに

経済産業省や 2021 年 9 月に発足したデジタル庁で、「アジャイル・ガバナンス」という言葉が使われているように、近年、アジャイル（Agile）がソフトウェア開発以外の分野で取り沙汰されることが多くなってきている。また、デジタル・トランスフォーメーション（DX）に注目が集まっており、IT などのデジタルを前提とし、ビジネスモデルや組織を変革し、顧客価値の実現を目指そうという動きが活発化しているが、そうした DX の変革の一手段として、アジャイルが検討されることも多い。ア

ジャイルは、ラジオ番組の企画や、新しい機械の開発、戦闘機の生産、ワインの生産のほか（Rigby, Sutherland, and Takeuchi, 2016）、BMW や米国トヨタ自動車（TMS）といった自動車メーカーの自動車そのものの開発（日経ビジネス編, 2019）、GE や IBM（Cappelli and Tavis, 2018）など幅広い業界や部門で広く取り入れられている。

このアジャイルは、2001 年に「Manifesto for Agile Software Development（アジャイルソフトウェア開発宣言）」として、その考え方が公開された。これまで主流であったウォーターフォール型のソフトウェア開発とは異なる

手法を実践していた 17 名のソフトウェア開発者が、それぞれの主義や手法についての議論を行い、公開されたものである。アジャイルは、ソフトウェアの仕様について厳密な決定をせず、開発プロセスを進める中で仕様を擦り合わせ、ある程度決まった部分だけを先に開発し、リリースするものであり、ユーザーのフィードバックをもらい、次の開発に生かしていくソフトウェア開発の手法である。

こうしたアジャイルの仕組みは、これまでのソフトウェア開発の方法とは異なる。顧客の要望に沿って完成したソフトウェアを提供するだけでなく、顧客がそうしたソフトウェアを実際に利用する際のことを想定したものである。顧客の利用時点に生じる価値の実現を行うためには、顧客とのコミュニケーションやフィードバック、さらには、顧客との共創が重要となる。しかし、こうしたソフトウェア産業について、特にアジャイルにおける「共創」を対象とした研究は少なく、その「共創」がいかなるものかといった分析も少ない。平井（2020a）では、開発企業と顧客の間でコミュニケーションやフィードバックがもらえにくくなる構造的な問題が多くあることについて論じたが、ソフトウェア開発における共創の詳細やその価値の生ずる部分についてまでは踏み込んでいない。

一方で、こうした顧客との共創については、これまでも多くの研究が蓄積されており、たとえば、Pralhalad and Ramaswamy（2004）による価値共創についての研究のほか、コミュニケーションを行い相互に理解し働きかけ合い、心理的に刺激する状況の枠組みとしての「場」の研究（伊丹，1999，2005）のほか、とりわけ顧客価値と結び付けられたマーケティング分野において、G-D ロジック（グッズ・ドミナント・ロジック）や S-D ロジック（サービス・ドミナント・ロジック）（Vargo and Lusch, 2004）を中心に研究が進んでいる。

本研究では、アジャイルにおける「共創」とそこで生じる価値について、こうした先行研究

を元に具体的な内容について検討することを目的とする。

## 2. アジャイルの特徴

### 2.1 アジャイルの要素

アジャイルは、正確には特定の手法を指しているのではなく、スクラム（Scrum）や XP（extreme programming）と呼ばれる、幾つかの軽量なソフトウェア開発プロセスの総称であり、ウォーターフォール・モデルと呼ばれる多くの手順に従って進んでいく重量級の開発手法と比較される（妹尾，2001）。

これまでのソフトウェア開発は、1970～1980 年代の市場の変化や成長が比較的安定していた時代から、計画とその遂行、管理を中心としたウォーターフォール・モデルと呼ばれる定型化、標準化した開発プロセスが合致していた（Cusumano, 1991, 2004; 妹尾，2001 など）。しかし、予め決められた計画と目的に従うという制約のため、顧客の要望が途中で変わっても対応することができない。外注先に任せてしまうことで、顧客はソフトウェア開発企業やそのエンジニアと一緒に自分たちでソフトウェアを作り上げるという能動的な姿勢は少なく、開発されたソフトウェアを納品してもらうという考え方に陥りやすく、顧客自らが本当に必要とするソフトウェアの開発は難しくなる。ウォーターフォールのような方法では、変化の速い市場に対応できず、試行錯誤を経て魅力的な機能を持つようなイノベティブなソフトウェアは期待することができないのである（Cusumano, 2004）。アジャイルは、激しい変化を伴う競争環境に対応しようとするものであり、特に解決すべき問題が複雑であったり、解決方法が不明であったりするなか、イノベーションを起こすために、より良いプロダクトやサービスを作ることを目指すものである（平井，2020b）。

ビジネス環境の変化と速さに対応し、顧客の要望を反映させた真に価値のあるソフトウェアを

作り上げるためには、顧客自身の協力が必要となる。ソフトウェアにおいて、顧客を中心とした反復型の開発がアジャイルである。アジャイルは、その特徴からウォーターフォール・モデルと比較され、反復型のプロセスや効率に焦点が当てられやすいが、その本質は人、組織（チーム）、文化であり、「組織的な学びの仕組み・改善のための「ふりかえり（Retrospective）」、「顧客との共創（Co-Creation）」、「自己組織化（Self-organizing Team）」「機能横断的（Cross Functional Team）」の4つの要素があげられる（表1）。

アジャイルは、ソフトウェアに対する顧客のニーズを理解することは難しく、技術の移り変わりも早いいため、そのような複雑なソフトウェアを前もって完全に設計することは無理があるという考え方である。顧客にとってより良いものをつくるため、顧客との共創を前提とした発見と解決のサイクルを繰り返す組織的な学習を行い、成功や失敗から学び、カイゼンしていく開発手法である（平井、2019）。

## 2.2 反復と共創

アジャイルは、開発プロジェクトを進めていくなか、顧客からの要求に基づくソフトウ

ェアを作成し、その検査とフィードバックをもとに新たな要求の設計や解析といった作業をPDCA サイクルのように反復（iteration）を繰り返す。

アジャイルではすべての機能を一度に取り込むのではなく、幾つかの機能を選択して開発を行う。さらに、その開発作業も1週間から1か月程度の非常に短い期間で反復して行い、それぞれの反復期間の終了ごとにソフトウェアを本番稼働させることを目指している。完成してからソフトウェアを動かすのではなく、ある程度動くソフトウェアを成長させながら作成する、反復、且つ漸進型であることが大きな特徴であり、顧客の要望をリスト化し、その中から優先度の高いものをできるだけ早く作成し、少しでもできあがったソフトウェアを顧客に確認してもらうことで、早期にフィードバックを得ることを目的としている（平井、2020a）。

ウォーターフォール・モデルのような従来の手続き型の手法に対し、アジャイルのような反復型のプロセスに注目が集まっている背景には、ビジネス環境の変化がある。急速にグローバル化が進み、市場も急激な変化を遂げており、VUCA と呼ばれるような不確実性や不透

表1 アジャイルの要素

特徴	概要
Self-organizing Team (自己組織化)	開発チームが状況に応じてミッションを実現するための選択を自分たちが決定し、行動できる。
Cross Functional Team (機能横断的)	外部に頼らず開発をやり遂げるために必要なあらゆるスキルを持つチーム、メンバーが複数の役割を果たすことができる。
Co-Creation (顧客との共創)	フィードバックといった顧客の参画の度合いが強く、ゴールを共有しており、そのため、人と人とのコミュニケーションやコラボレーション、「共創」を重視。
Retrospective (組織的な学びの仕組み・改善のための「ふりかえり」)	単なる試行錯誤ではなく、失敗からいかに学び、成長できるか。「反復」活動を小さく短く繰り返すことで、小さな失敗を繰り返し、多くを学んでいく。

出所：西村・永瀬・吉羽（2013）、市谷（2019）、Rigby, Sutherland, and Takeuchi（2016）、平井（2019, 2020a, 2020b）を元に作成。

明性が増した環境となっている。従来の市場のニーズに応えるために長い時間をかけて調査や分析を行い、製品・サービスを開発していくような方法は過去のものとなり、市場の不確実性の攻略と顧客への価値提供スピードに大きくシフトしているのである（田村・和田, 2015）。

どのようなプロダクトやサービスを創り出すのかわからない状況に対応するには、ユーザーの要望を見つけ出し、ユーザーにとって価値のあるものを作ろうとする顧客を中心とした考え方が重要となり、リスクを削減するために少しずつ作って検証するといったプロセスが不可欠である（渡辺・ブラウン・小俣, 2019）。そのためには、仮説とその検証、学習を繰り返す反復型のプロセスが有効となるのである。

アジャイルにとって重要なことは、失敗から学び、成長できるかである。反復活動を小さく短く繰り返すことは、大きな失敗を避け、リスクを最小化するという目的もあるが、それ以上に小さな失敗を繰り返すことで、多くを学んでいくことが求められる。これは単なる試行錯誤ではなく、失敗をふりかえり、学習できる環境が必要となる。

また、失敗を分析するような反省会ではなく、改善のための集まりの場を用意し、開発プロセスや技術についてのほか、チーム自体もふりかえりの対象となる。心理的な安全のもと、チームメンバーが自由に発言し、共感をもって意見を聞く（渡辺・ブラウン・小俣, 2019）。失敗が許され、それを糧に同じような失敗を繰り返さないようにメンバーが成長していくことで、最終的に高いパフォーマンスや難易度を達成していくのである（西村・永瀬・吉羽, 2013）。チームが最終的な結果を改善するために素早いサイクルで学習し、変化に迅速に適応できるようにすることが重要であり、そのためにチームや企業の境界を超え、顧客と会話をし、共創の場の中でイノベーションを作り出していくのである。

このように、アジャイルは、単に素早く開発

を行うことではない。アジャイルは、反復活動を通じたフィードバックといった顧客の参画の度合いが強く、顧客との対話や交流、人と人とのコミュニケーションやコラボレーション、共創を重視している（平井, 2019）。製品開発当初の計画に従うよりも顧客の要望やビジネスの変化への対応を価値としており、最後まで少しでも顧客のために改善していくことがアジャイルの目的なのである。

### 3. 共創

#### 3.1 価値の共創

これまでの工業化を前提とした計画駆動型の方法は規模の効率性を高めたが、デジタル化には顧客志向のビジネスモデルと価値のある共創サービスプロセスが必要となる（Kuula, Haapasalo, and Tolonen, 2018）。共創という概念自体は新しいものではなく、これまでも「場」（伊丹, 1999, 2005）や、企業と顧客による価値共創（Prahalad and Ramaswamy, 2004）、マーケティング分野における顧客価値創造や共創マーケティング（Vargo and Lusch, 2004, 2008, 2012; 藤川, 2012, 2017; 村松編, 2015 など）として研究が蓄積されてきている。

こうした既存の研究では、共創における価値がどこで生じるのかが焦点とされてきた。たとえば、池田・山崎（2014）は、共創を「企業と顧客が中長期的な関係を築き深く理解し合う中で、予期せぬ、または期待を超えた新しい価値を生み出すこと」（池田・山崎, 2014, p.13）と定義している。Grönroos and Gummerus（2014）は、共創について、2つ以上の行為者（actors）が、直接的に相互作用するプロセスの中で、1つの協働的（collaborative）、対話的なプロセス（dialogical process）に統合される、共に何かを創り出すプロセスと定義し、そのうえでサービス提供者と顧客がそうした共創のプラットフォームで行う共同のプロセスから価値が創造されると述べている。

また、共創における企業と顧客の関係につい

ては、Prahalad and Ramaswamy (2000) が、企業の新たなコンピタンス (competence) の源泉として顧客を捉え、顧客が消費者という従来の役割から一歩踏み出して価値の共創者という役割を担いつつあるとし、受動的な観客としての顧客 (passive audience) から、積極的な役者としての顧客 (active player) へと進化していると指摘している。さらに、Prahalad and Ramaswamy (2004) は、顧客との価値共創こそ企業の競争優位の源泉であるとし、顧客も企業と同様にイノベーションの主体と位置付けている。これは、それまでのイノベーションの主体を企業自身に位置付け、顧客の役割を副次的に取り扱ってきた発想とは異なるものであり、企業と顧客が一体となってイノベーションを実現すること (co-innovation) を目指すべきとしている (Prahalad and Ramaswamy, 2004)。

また、Prahalad and Ramaswamy (2004) は、従来の価値創造プロセスにおいて、企業と消費者が生産と消費のように明確に役割が分かれており、市場を通してそうした生産者と消費者間で価値が含まれた製品やサービスが交換されているとしている。一方で、Prahalad and Ramaswamy (2004) は、価値を定義したり、想像したりする価値共創のプロセスに消費者が徐々にかわりを強めることで、従来の役割分担は消えていくとし、BtoB や BtoC といった区別をせず、自社と関わりのある人すべてを消費者とみなすべきとも述べている。

伊丹 (2005) は、「人々がそこに参加し、意識・無意識のうちに相互に観察し、コミュニケーションを行い、相互に理解し、相互に働きかけ合い、相互に心理的刺激をする、その状況の枠組み」(伊丹, 2005, p.103) として「場」について述べており、そうした「場」の創発には、自由や信頼、情報共有といった条件を備えている必要性を指摘している。

共創のデザインについては、サービスマーケティングの分野において、企業が規定したパラメータの範囲内での顧客の参加を共同生

産 (co-production) と呼んでおり、さらにオペレーションの生産性や品質に対して影響を及ぼす要因としてみなしている (小野・藤川・阿久津・芳賀, 2014)。共同生産では、企業がサービスの生産プロセスの中に、変動要因ともなりうる顧客の参加の方法をいかにデザインし、運営するかが主要な課題であり、顧客の使用プロセスを企業はコントロールできないが、時間と空間を共有するなかで提供されるサービス財の共同生産を通して顧客行動に働きかけ、仕掛けることで、企業は顧客が使用価値を作るプロセスに関与することができる (小野・藤川・阿久津・芳賀, 2014)。

一方で、共創は、企業と顧客による共同生産だけを意味するものではない。村松 (2015) によると、企業と顧客の直接的な相互作用によって価値が共創されるのであり、その共創による価値は、顧客によって判断される。そのため、どのように消費されるのかを事前に配慮するようなモノづくりや、生産プロセスに顧客を取り込んだ顧客参加型の製品開発などの顧客との共同だけでは価値共創とはいえないと指摘している。さらに、村松 (2015) はこうした取り組みは、より良い交換時点の価値を実現しようとするものであり、その交換価値を共同で作成しているに過ぎないことを指摘し、企業が購入後に行うアフターサービスについても、モノの購入時の交換価値に戻そうとしているに過ぎないとも述べている。

つまり、顧客参加型の製品開発や、サービス提供プロセスへの顧客参加、協働型マーケティングなどは、顧客とのより良い交換を成功させるために事前に価値を埋め込むという点に重きを置いた企業主導型のものであり、顧客と共に経験を通した価値や製品やサービスを使用する時点での価値といった文脈が共創されるわけではないのである (Vargo and Lusch, 2008)。

### 3.2 G-D ロジック

共創について、製品とサービスの交換タイ

ミングの価値に着目した理論が S-D ロジック（サービス・ドミナント・ロジック）（Vargo and Lusch, 2004）である。S-D ロジックは、これまでのサービスの交換様式を G-D ロジック（グッズ・ドミナント・ロジック）として捉えている。G-D ロジックと S-D ロジックの比較が、表 2 である。

G-D ロジックは、価値について企業が製品やサービスを生産し、顧客への販売などにより受け渡す部分に存在するとしている。顧客はその価値を消費するだけの存在として捉えられ、顧客は価値創造プロセスの外側に存在する（大藪, 2015）。この受け渡す際に価値が生じることを「交換価値」（value in exchange）と表し、この交換価値の金額、つまり、販売価格を最大化することを目的としている。

G-D ロジックでは、企業の目的を顧客が製品の価値を認識し、金銭を支払って購入することであるとし、企業はそのより良い交換価値のために、製品を開発し、販売する。その結果、他社の製品よりも性能やブランドなどで優劣を競い、差別化を行うことになる。顧客は、そうした企業の製品に対し、交換価値と価格、自身の

経験、他社の評価などの諸条件を考慮し、製品を購入するか判断する（清野, 2015）。市場で行われるのは、すでに出来上がった価値の取引、すなわち交換価値だけであり、そうした状況下では、バリューチェーンから生まれた価値と消費者行動が生み出す需要を効率的にマッチングさせることが極めて重要となる（Prahalad and Ramaswamy, 2004）。

このように G-D ロジックが、交換価値を最大化することを目的とすることで、企業、顧客共に製品の交換時点に重点が置かれることになる。顧客は自身の製品の使用を想定しながら購入すべき製品を見極めようとし、企業も顧客の使用のシーンを見据えた顧客価値を想定して製品開発を行うことになる（清野, 2015）。しかし、こうした顧客の行動は、購入時点・交換時点想定しているに過ぎず、製品の使用時に生じる価値、つまり、製品に含まれた価値を実際に利用することで引き出すことができるかは未知数となる（清野, 2015）。

以上のように、G-D ロジックは製品の使用について企業も顧客も事前に想定する範囲に留まっており、企業が一方的に価値をつくり込む

表 2 G-D ロジックと S-D ロジックの比較

	G-D ロジック	S-D ロジック
顧客像 (消費者)	企業がつくり出す価値を消費する存在 価値の破壊者	消費者であると同時に生産者としての役割を担い、企業と協働して価値をつくり出す価値共創者
企業の役割	価値の役割を決め、消費者に提供する	個々の顧客を巻き込んで独自の価値を共創する
交換対象 (サービス観)	小麦や魚といった有形なグッズ（モノやサービス）	農耕と漁獲といったサービス（経済活動すべてをサービスと捉える）
価値	交換価値（金額）	使用価値（使用と経験）・文脈価値
価値創出者	企業による創出	企業と顧客の相互作用、双方向的・協業的な価値共創
価値創造プロセス	生産から販売まで	文脈価値の認識まで

出所：Prahalad and Ramaswamy（2004）p.241, Vargo and Lusch（2004, 2008, 2012）、田口（2010）、藤川（2012, 2017）、Akaka, Vargo, and Lusch（2013）、池田・山崎（2014）、村松（2015）を元に筆者作成。

ものであり、企業による価値生産と顧客による価値消費が分業されているのである（藤川, 2017）。

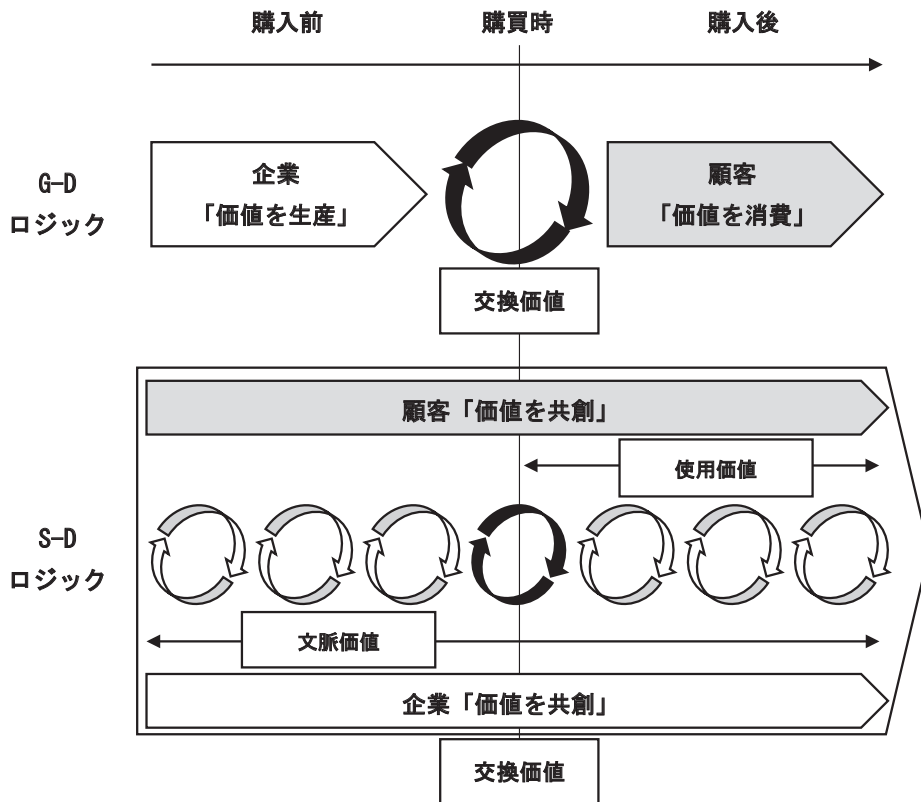
### 3.3 S-D ロジック

G-D ロジック に対し、Vargo and Lusch (2004) による S-D ロジックは、顧客を単なる受け手や使用者としてではなく、価値共創者として捉えている（神田, 2021）。S-D ロジックでは、これまでの G-D ロジックのような製品の売買取立したときに受け取る目に見える商品の価値（交換価値）ではなく、顧客と共に経験を通じた価値や、製品やサービスを使用する時点での使用価値（value in use）まで踏み込んだ文脈価値（value in context）を創出することの重要性が主張されている（Vargo and Lusch,

2004, 2008; 池田・山崎, 2014; 神田, 2021）。この文脈価値とは、個別的で状況依存的な性質を持つものであり、ある顧客にとって価値があるものでも他の顧客にはそれほど価値が無い場合や、同一の顧客であっても状況によって認識する文脈価値が異なる可能性がある（大藪, 2015）。

G-D ロジックと S-D ロジックの価値づくりの違いを表したものが図 1 である。

S-D ロジックでは、使用価値は顧客が利用することによって初めて認知される。Vargo and Lusch (2008) は、価値を最終的な受益者である顧客が主観的に認識するものであるとしている。S-D ロジックは、企業を主体とするのではなく、顧客や企業、ステークホルダーが自ら価値を創造する主体として捉えられており、現代



出所：西山・藤川（2016）p.48, 藤川（2017）を元に筆者作成。

図1 G-D ロジックと S-D ロジックの価値づくりの違い

のような複雑で不確実性の高いビジネス環境に適応した見方を提供している（神田，2021）。

また，S-D ロジックでは，顧客の受動的な参加である共同生産（co-production）と能動的，積極的な参加である共同創造（co-creation）が区別されており，共創によって生まれる価値について，企業が製品・サービスを通じて提供するものから企業と顧客が共に創り出すものという考え方へと転換することを促す（池田・山崎，2014）。

企業と顧客は相互作用を通じて，それぞれが価値を生み出すものであり，双方向的，協業的な価値創造を前提とし，製品やサービスの購買時だけでなく，その前後にも様々なやりとりをする文脈の中で実現する使用価値を重視するのである（藤川，2012）。

たとえば，自動車の場合，完成した車はそれ自体には価値が無く，実際に運転するなど利用されることで価値が創造されるのであり，自動車メーカーと利用者による価値共創として捉えられる（Vargo, Maglio, and Akaka, 2008）。製品は製造されても，購入され，利用されなければ価値は生じないのである（Gummesson, 1998）。

さらに，S-D ロジックを前提とし，価値を顧客の使用経験を通して時間とともに蓄積されるとするものが S ロジック（サービス・ロジック）である（Grönroos, 2008; Grönroos, 2011）。Grönroos（2011）によると，企業および顧客ともに価値の共創者であるという S-D ロジックの主張は，企業と顧客の役割を曖昧にしていまい，価値創造のプロセスを不透明にしていまいとも指摘している。S ロジックでは，顧客は価値の共創者ではなく，創造者として捉えており，企業と顧客間において，現場従業員と顧客の直接的コミュニケーションといった対話的プロセスである直接的相互作用が無ければ，共創は実現しないとされる（大藪，2019）。

以上のように，共創で生ずる価値は，これまでの製品の完成やサービスの提供によって完結

していた取引が，そこに至るプロセスと製品の利用やサービスの提供によって生じる顧客にとっての使用価値までもが含まれることになり，生産と消費が一体化した文脈価値（池田・山崎，2014）なのである。

顧客にとって本来重要となることは，製品を使用することではなく，製品を使用することで顧客自身のニーズを満たすことであり，それによって価値が生ずるのである。S-D ロジックの世界観に基づけば，企業のみでは価値の最大化を実現できず，企業と顧客がともに価値をつくり出す「価値共創」が重要となるのである（藤川，2017）。

## 4. アジャイルにおける共創

### 4.1 ソフトウェア開発における価値共創

ここまで述べられてきた企業と顧客の共創関係，さらには，G-D ロジック，S-D ロジックなどで述べられた顧客との交換価値と使用価値，そして文脈価値について，ソフトウェア開発では，どのように捉えることができるのであろうか。

ソフトウェア開発では，既存のウォーターフォール・モデルのような取り組みは G-D ロジックで，アジャイルのように反復且つ漸進型の取り組みは S-D ロジックで説明することができる。

ウォーターフォール・モデルでは，開発したソフトウェアを納品した時点でその取引は終了する。つまり，計画通りに開発することに価値の主軸が置かれており，そのソフトウェアやシステムを顧客がどのように使用するかは顧客の問題であって，企業側には関係がないといえる。仮に，全く使えないようなソフトウェアであっても，顧客の依頼にあった通りにソフトウェア製品を開発し，納品さえすれば済む。現実には役に立たないソフトウェアは顧客の不満を生み，企業の信用を落とし，継続的な契約は期待できないであろうし，状況によっては損害賠償を請求される恐れもある。そのため，アフ

ターサービスや2次開発などを行うこともある。

こうしたウォーターフォール・モデルは、交換価値を最大化するものであり、G-D ロジックの考え方に近い。また、下請け企業や受発注関係に基づく開発体制は、顧客が一切参加しない共同生産ですらない場合もある。ソフトウェアの仕様を決める要件定義で顧客のニーズを確認していくが、あくまで完成したソフトウェア製品の納品、つまり交換価値の最大化が目的であり、共創関係にはほど遠いのが現状である。

こうした既存の手法では、成果物と、計画通りに進んだかどうかの定量的なプロジェクトの成功や進捗が重視されてきた。しかし、顧客の真に望むものは、顧客自身もわかっていないことも多く、当初の計画通り作成したものが交換時点のニーズを満たしているとは限らない。アジャイルは、こうした従来型のアプローチが犯してしまいがちな顧客ニーズとは異なるものを作ってしまうことを防ごうとするものであり、検証による学習の量や質を重視し、実際の製品やサービスを顧客に提示、評価してもらう現物主義と顧客志向である（菊池，2013）。

既に述べたように、アジャイルの反復型プロセスは、顧客の要望を少しずつ実装し、実際の製品・サービスを使ってもらうことで、早期にフィードバックを貰い、顧客のビジネスに合わせ、動くソフトウェアを成長させながら作成する、反復、且つ漸進型である。顧客の参画の度合いが強く、ゴールを共有しており、そのため、人と人とのコミュニケーションやコラボレーション、共創を重視している。

顧客からの要求一覧（バックログ）に対し、優先事項をつけて開発を行い、顧客にとって価値があるものを優先して開発していく。顧客からのフィードバックや市場の変化に応えるような開発中の設計の変更は、成功に結び付くようなものであれば、必ずしも悪いものではない。製品開発当初の計画に従うよりもユーザーの要望やビジネスの変化への対応を価値としている

のである。

このアジャイルの変化していく顧客のニーズに応え、常に顧客のビジネスに合致したソフトウェアを提供する方法は、S-D ロジックで説明された顧客にとっての使用価値が、常に最大化されることを目的としているといえる。

S-D ロジックの考え方に基づくのであれば、アジャイルは、顧客が利用を可能とする製品やサービスのリリース、つまり交換価値の最大化ではなく、実際に利用された反応を見て改善していく使用価値、さらに顧客を巻き込んで共にソフトウェアを作りこむという点から文脈価値を実現しているのである。

## 4.2 事例：ソニックガーデン

実際に、アジャイル開発により、共創と顧客の価値を実現している事例をとりあげる。

ソフトウェア開発会社である株式会社ソニックガーデンでは、「納品のない受託開発」と銘打ち、アジャイル開発を行っている。「納品のない受託開発」は、既存のウォーターフォール・モデルのように一度に完成まで作り上げる一括請負での受託開発、つまり納品して終わるものではなく、月額定額で顧客とエンジニアが共にソフトウェアを最初から最後まで、開発から運用に至るまで区切りなく関係性を継続するサービスであるとしている。

これまでのウォーターフォール・モデルのような開発手法では、工程や役割ごとに異なる企業に外注されることも多く、顧客とベンダーの関係性が分断されていることも珍しくない。「納品のない受託開発」では、こうした問題の解消を図るため、顧客の専属エンジニア（顧問エンジニア）を用意し、エンジニアと顧客が直接コミュニケーションをとることで顧客ニーズとの齟齬や取りこぼしを防ぎ、さらにビジネスの発展と共に開発内容も変えていくことを可能としている。ソニックガーデンは、ソフトウェアの質が顧客の利用するシステムあるいは業務システムの質を左右し、結果的に事業の成否を

決めるとし、事業とソフトウェアは表裏一体で切っても切れない関係であり、ソフトウェアを「事業の写像」として捉えているのである。

さらに、ソニックガーデンは、ソフトウェアは使い始めることで初めて価値が生じるものであり、完成しただけのソフトウェアはその開発投資分のマイナス、つまり負債が生じた状態であることを指摘する（倉貫、2014）。

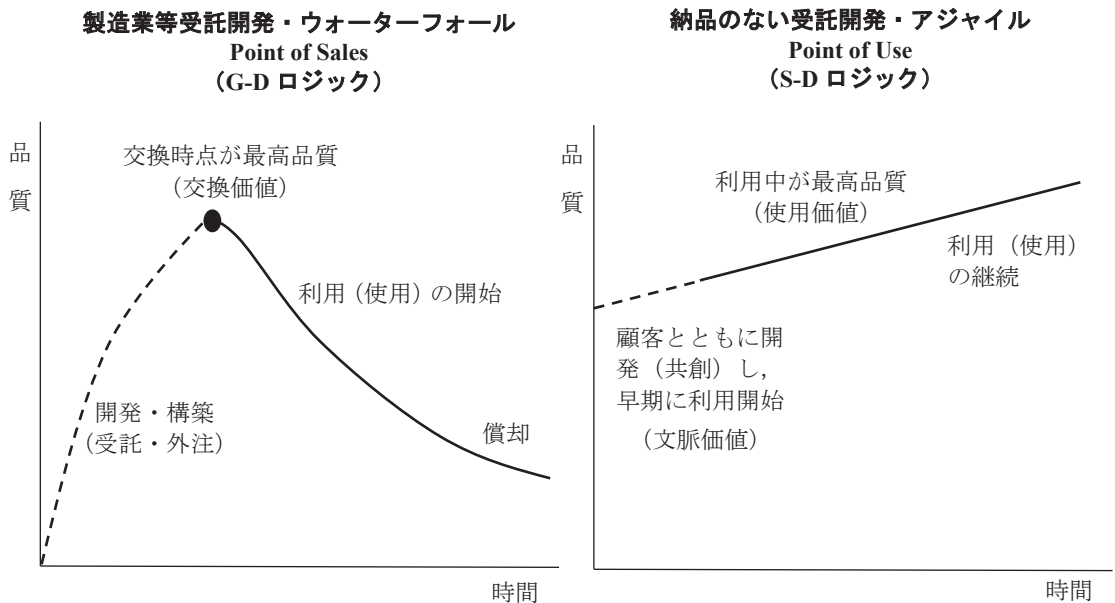
発注元の企業は、自身のビジネスのために、つまり事業の成長や発展のためにソフトウェアが必要なのであり、一方で、開発企業はソフトウェアを完成させ、納品することだけを目的としてしまうところに齟齬が生じるのであり、ソニックガーデンのこれら取り組みはその解消を目指したものと見える。

ソニックガーデンの指摘する、ソフトウェアをアップデートし続けることを事業をアップデートし続けることとみなす考えは、アジャイルの漸進的な開発手法を前提とする。「納品のない受託開発」は、アジャイル開発に基づき、まずは1か月程度でリリースを行い、顧客

やエンドユーザーの反応、顧客のビジネスの成長に合わせて細かく修正を行い、リリースを繰り返していく。ソフトウェアの開発は、これまでのウォーターフォール・モデルのような手法では、数か月から数年かかる場合もあり、その間に顧客のビジネスが発展することで追加の機能や内容自体を変更する必要が生ずることも多い。アジャイルによる顧客ニーズの反映とこまめなリリースはこうした問題を解決することが可能となるのである。

ソニックガーデンは、従来の製造業の受託開発による購入時点が最高品質となるもの（Point of Sales）と、納品のない受託開発、つまりアジャイル型の開発による利用される瞬間を最高にするもの（Point of Use）とを比較している（図2）。

前者はG-D ロジックの交換価値を最大化することであり、後者はS-D ロジックの使用価値と文脈価値に当てはめることができる。ソニックガーデンの Point of Sales の製造業等の開発手法は、いわゆるウォーターフォール・モデル



出所：倉貫（2014）p.151, 今村（2020）, 「株式会社ソニックガーデン」ホームページを元に、筆者加筆・修正。

図2 製造業と受託開発と納品のない受託開発（アジャイル）の価値

ルであり<sup>1)</sup>、受託や外注により開発・構築が行われ、その期間の間、完成するまでソフトウェアを顧客は使用できない。また、完成品を最後に受け渡すという方法のため、交換時点が想定される最高品質の交換価値となるが、その後の状況に合わせた変更は行われない。そのため、時間が経つにつれて、図2のように品質やニーズへの対応といった顧客にとっての価値は逡減していく。

一方で、Point of Use の納品のない受託開発によるアジャイルの方法は、顧客とともに開発し、素早くリリースを行い、継続的に修正を行っていく。顧客が利用している状況を常に最新、且つ顧客ニーズを満たした最高品質とみなすことで、顧客の価値は Point of Sales のように逡減せず、むしろ継続的な修正により、品質は向上していくことになる。こうした使用価値に加え、企業と顧客が最初から最後まで共創していくことで、そのプロセスも含め、生産と消費が一体化した文脈価値を含んでいるのである。

以上のように、ソニックガーデンのアジャイルに基づく「納品のない受託開発」は、G-D ロジックとS-D ロジックで説明されてきた共創に当てはめることができる。アジャイル開発は、ウォーターフォール・モデルのような交換価値の最大化ではなく、その共創で生じた価値が顧客と共に最初から最後まで開発を行っていくような生産と消費が一体化した文脈価値をもち、さらに常に顧客のニーズに合った利用を可能とする使用価値の実現を図るものなのである。

## 5. おわりに

本研究では、アジャイルの特徴の一つである顧客との共創について、その共創がいかなるものなのか、既存の共創研究を中心に説明を試みてきた。

共創とは、単に何かを顧客と一緒に共同ですることではなく、企業と顧客が中長期的な関係を築き、一体となってイノベーションを実現す

ることであり、そこに価値が生じる。

しかし、これまでのウォーターフォール・モデルに代表されるようなソフトウェア開発では、そうした共創やそこから生じる価値ではなく、完成したソフトウェアを予定通りリリースすることが目的であり、G-D ロジックで説明された製品の販売、つまり交換価値の最大化を目指すものとして捉えることができる。

一方、アジャイルでは、反復型の改善手法を用い、顧客と共に、そのビジネスの成長に合わせて最初から最後まで価値のある製品やサービスを継続的に提供し続ける。一方的な価値の提供ではなく、双方向に意見を交えながら、実際に顧客がそのソフトウェアを利用して目的を果たすことができること、つまり価値を生むことが重要であり、これはS-D ロジックで説明された使用価値であり、そこに至る共創のプロセスでは文脈価値が生じていると捉えることができる。

このように考えることで、アジャイルにおける共創とは、反復型のプロセスのもと、「フィードバック」といった顧客の参画の度合いが強く、ゴールを共有しており、そのため、人と人とのコミュニケーションやコラボレーション、「共創」を重視し、さらに、顧客の利用まで見据えた使用価値とそのプロセスに伴う文脈価値を実現することなのである。

## 注

- 1) ウォーターフォール・モデルはソフトウェア開発の手法であるが、そのソフトウェアはハードウェア開発に付随するものとして扱われ、その後分離した経緯があり、そのため製造業と同じような手法となっている（平井、2018）。

## 参考文献

- Akaka, Melissa Archpru., Vargo, Stephen L., and Lusch, Robert F. (2013) "The Complexity of Context: A Service Ecosystems Approach for International Marketing", *Journal of International Marketing*, 21 (4), pp.1-20.

- Cappelli, Peter. and Tavis, Anna. (2018) "HR Goes Agile", *Harvard Business Review*, March-April 2018, Harvard Business School Publishing.
- Cusumano, Michael A. (1991) *Japan's software factories: a challenge to U.S. management*, New York; Tokyo: Oxford University Press. (富沢宏之・藤井留美訳 (1993)『日本のソフトウェア戦略：アメリカ式経営への挑戦』三田出版会.)
- Cusumano, Michael A. (2004) *The business of software: what every manager, programmer and entrepreneur must know to thrive and survive in good times and bad*, Simon and Schuster. (サイコムインターナショナル訳 (2004)『ソフトウェア企業の競争戦略』ダイヤモンド社.)
- Grönroos, Christian (2008) "Service Logic Revisited: Who Creates Value? And Who Co-creates?", *European Business Review*, 20 (4), pp.298-314.
- Grönroos, Christian (2011) "Value Co-creation in Service Logic: A Critical Analysis", *Marketing Theory*, 11 (3), pp.279-301.
- Grönroos, Christian and Gummerus Johanna (2014) "The service revolution and its marketing implications: Service logic vs service-dominant logic", *Journal of Service Theory and Practice*, 24 (3), pp.206-229.
- Gummesson, Evert (1998) "Implementation Requires a Relationship Marketing Paradigm", *Journal of the Academy of Marketing Science*, 26 (3), pp.242-249.
- Kuula, Seppo., Haapasalo, Harri., and Tolonen, Arto. (2018) "Cost-efficient co-creation of knowledge intensive business services", *Service Business*, vol.12, pp.779-808, Springer Nature.
- Prahalad, C. K. and Ramaswamy, Venkatram (2000) "Co-opting Customer Competence", *Harvard Business Review*, 78 (1), pp.79-87. (中島由利訳 (2000)『顧客と共に競争優位を築くカスタマー・コンピタンス経営』『ハーバード・ビジネス・レビュー』25 (6), pp.116-128, ダイヤモンド社.)
- Prahalad, C. K. and Ramaswamy, Venkatram. (2004) *The Future of Competition*, Harvard Business School Press. (有賀裕子訳 (2013)『コ・イノベーション経営』東洋経済新報社.)
- Rigby, Darrell K., Sutherland, Jeff., and Takeuchi, Hirotaka. (2016) "Embracing Agile", *Harvard Business Review*, MAY 2016 ISSUE, Harvard Business School Publishing. (倉田幸信訳 (2016)「アジャイル開発を経営に活かす 6 つの原則 臨機応変のマネジメントで生産性を劇的に高める」『ハーバード・ビジネス・レビュー』41 (9), pp.92-106, ダイヤモンド社.)
- Vargo, Stephen L. and Lusch, Robert F. (2004) "Evolving to a New Dominant Logic", *Journal of Marketing*, 68 (1), pp.1-17.
- Vargo, Stephen L. and Lusch, Robert F. (2008) "Service-Dominant Logic: Continuing the Evolution", *Journal of the Academy of Marketing Science*, 36 (1), pp.1-10.
- Vargo, Stephen L. and Lusch, Robert F. (2012) "The nature and understanding of value: A service-dominant logic perspective", *Review of Marketing Research*, 9, pp.1-12.
- Vargo, Stephen L., Maglio, Paul P., and Akaka, Melissa Archpru. (2008) "On value and value co-creation: A service systems and service logic perspective", *European Management Journal*, Volume.26, Issue.3, pp.145-152.
- 池田紀行・山崎晴夫 (2014)『次世代共創マーケティング』SB Creative.
- 伊丹敬之 (1999)『場のマネジメントー経営の新パラダイムー』NTT 出版.
- 伊丹敬之 (2005)『場の論理とマネジメント』東洋経済新報社.
- 市谷聡啓 (2019)「正しいものを正しくつくるープロダクトをつくるとはどういうことなのか、あるいはアジャイルのその先についてー」株式会社 BNN 新社.
- 今村一真 (2020)「ソニックガーデンー価値共創に基づくリレーションシップと組織文化ー」村松潤一・藤岡芳郎・今村一真編著『ケースで学ぶ 価値共創マーケティングの展開ー新たなビジネス領域への挑戦ー』pp.165-177, 同文館出版.
- 大藪 亮 (2015)「サービス・ドミナント・ロジックと価値共創」村松潤一編『価値共創とマーケティング』pp.54-69, 同文館出版.
- 大藪 亮 (2019)「サービス視点のマーケティング研究：サービス・ドミナント・ロジックとサービス・ロジックにおける価値共創」『経営とデータサイエンス』1, pp.1-18, 岡山理科大学マネジメント学会.
- 小野譲司・藤川佳則・阿久津聡・芳賀麻誉美 (2014)「共創志向性ー事後創発される価値の原動力ー」『マーケティングジャーナル』33 (3), pp.5-31.
- 神田正樹 (2021)「エンゲージメントの創発と価値の共創」『明大商學論叢』103 (4), pp.105-119, 明治大学.

- 菊池隆裕 (2013)「広がるリーンスタートアップ 大企業の新商品開発に威力」『日経コンピュータ』(838), pp.62-67, 日経 BP 社.
- 倉貫義人 (2014)『「納品」をなくせばうまくいく』日本実業出版社.
- 清野 聡 (2015)「サービス・ドミナント・ロジックと価値共創」村松潤一編『価値共創とマーケティング』pp.205-220, 同文館出版.
- 妹尾 大 (2011)「ソフトウェア開発の新潮流 一状況論的リーダーシップの胎動一」『組織科学』(35), pp.65-80, 組織学会.
- 田口尚史 (2010)「サービス・ドミナント・ロジック: 間接的サービス供給における4つの価値共創パターン」『横浜商大論集』43 (2), pp.90-121.
- 田村昌也・和田 洋 (2015)「新規事業のイノベーションを加速するリーン・スタートアップ適用とアジャイル開発: ビジネス開発とソフトウェア開発両面からのアプローチ 実践のポイント」『Provision』(87), pp.20-25, 日本アイビーエム.
- 西村直人・永瀬美穂・吉羽龍太郎 (2013)『SCRUMBOOT CAMP THE BOOK』翔泳社.
- 西山浩平・藤川佳則 (2016)「サービス・イノベーションの社会受容デザイン —価値共創の第三のアクター「レギュレーター」の役割—」『マーケティングジャーナル』35(3), pp.45-62.
- 日経ビジネス編 (2019)「BMW (独自動車大手) アンチ巨人のスピード経営」『日経ビジネス』2019年2月25日号, pp.70-74, 日経 BP 社.
- 平井直樹 (2018)「ソフトウェア開発プロセスにおける分業構造と知識労働 —日本の受託ソフトウェア開発の組織問題—」博士論文, 立教大学大学院ビジネスデザイン研究科.
- 平井直樹 (2019)「アジャイルの導入と本質 —開発プロセスから学習する組織活動へ—」『立教 DBA ジャーナル』(10), pp.35-44, 立教大学大学院ビジネスデザイン研究科.
- 平井直樹 (2020a)「ソフトウェア開発におけるアジャイルに向けた共創の課題 —顧客の構造問題の観点から—」『現代ビジネス研究所紀要』2019年度, 昭和女子大学現代ビジネス研究所.
- 平井直樹 (2020b)「顧客志向の反復型プロセス —リーン・スタートアップとアジャイルの組織的仕組み—」『立教 DBA ジャーナル』(11), pp.45-58, 立教大学大学院ビジネスデザイン研究科.
- 藤川佳則 (2012)「製造業のサービス化: サービス・ドミナント・ロジックによる考察」『パナソニック技報』58 (3), pp.168-173, パナソニックコーポレート R&D 戦略室.
- 村松潤一 (2015)「価値共創の論理とマーケティング研究との接続」村松潤一編『価値共創とマーケティング』pp.129-149, 同文館出版.
- 渡辺 隆・ブラウンザック・小俣剛貴 (2019)「Pivotal 流イノベーションを起こす開発チームの作り方」『日経 systems』(319), pp.36-43, 日経 BP 社.

## インターネット資料

- Manifesto for Agile Software Development (アジャイルソフトウェア開発宣言)  
<https://agilemanifesto.org/> (2021 年 9 月 24 日閲覧).
- 株式会社ソニックガーデン  
<https://www.sonicgarden.jp/> (2021 年 8 月 28 日閲覧).
- 藤川佳則 (2017)「サービス・マネジメントの変遷を学ぶための論文」DIAMOND ハーバード・ビジネス・レビュー, 2017.10.10  
<https://www.dhbr.net/articles/-/5052> (2021 年 9 月 17 日閲覧).

