

10 ソフトウェア的発想で ハードウェアを作る

～プログラミング経験をハードウェアの企画・
開発に活かす～



田中章愛 | ソニー・インタラクティブエンタテインメント

ソフトとハード、越境はチャンス

就職やキャリアチェンジにより、専攻していたソフトウェア分野から他分野へ転向し、“越境”した方も少なくないでしょう。また近年 IoT やロボットなどソフトとハードの分野横断的な業務に取り組む方も増加傾向にあるのではないのでしょうか。

私は学生時代、趣味のプログラミングやロボコン出場を経て制御・システム工学を専攻し、研究ではロボットのソフト開発に情熱を注ぎました。ソニー入社後は、配属先の状況や偶然もありメカエンジニアに転向しロボットの研究開発を行うことになりました。その後 IoT 系ハードウェアの試作～量産を経て、現在はソフトウェアがバックグラウンドの研究者・エンジニアが私を含め 3 名集まって始まった「toio™」（以降 toio）¹⁾ というロボットの商品企画や事業開発に携わっています。toio のハードについては、私は最初のプロトタイプからかわり、ソフト専門の同僚とともに商品像を探索しながら企画・開発してきました。

このように、ソフトからハードに専門の軸足を移し、その後それらを横断した商品開発・企画の立場になるうちに、その経歴がソフト専門家の考えの理解・共感・協業に役立つのではないかと、そしてソフトウェア工学で触れた概念やノウハウはハードに広く応用可能で、まだまだ異分野に応用されていない発想の種が眠っているのではないかと、思うようになりました。以来ソフト的発想をハードウェアや仕

事自体に積極的に活用しています。

本稿では特にソフトを専攻した後ハードに深くかかわることになった方へ向けて、ハード的にもソフト基点の発想を盛り込んで企画・開発された toio の事例を述べ、ソフトの経験をどう活用したかについて紹介します。就職・キャリアチェンジ等で分野を“越境”された方々が専門性を新しい業務に活かすヒントとなればと考えています。

ソフト的発想でハードを作る： toio の事例

toio は「創意工夫を引き出すロボット玩具」として、ロボット 2 台入りの「本体セット」とカートリッジや周辺部品が詰まった「専用タイトル」を組み合わせ遊ばすという、ゲーム機にも似た一風変わった商品構成（図-1）を持つおもちゃのためのプラットフォームを目指した商品です。

キューブ型ロボット「toio コア キューブ」



■図-1 toio の商品構成

(図-2)には子どもたちが好きなおもちゃや自作の工作を載せることができ、特殊印刷されたマットと絶対位置を検出できる読み取りセンサを用いた複数台の協調制御によりシナリオに沿ってキャラクタを演じたり、ゲーム性のあるインタラクティブな遊びを楽しんだりできます。

この、おもちゃとしては一見変わった構成は、企画からアーキテクチャ・開発手法に至るまでソフト的な発想をハードに逆輸入した形で色濃く反映しています。以下に代表事例を紹介しします(表-1)。

プラットフォーム型の商品構成

toio はロボットを使った実世界の体験が中心の製



■図-2 おもちゃや工作を載せてキャラクタが自在に変わるキューブ型ロボット toio コアキューブ

■表-1 toio に活用したソフト的発想・参照した概念とハード企画・開発における実現・実施形態の例

項目	ハード企画・開発の実現・実施形態	ソフト的発想・参照した概念
企画方針	商品構成	本体セット +専用タイトル
	技術特徴	シナリオ・ルール+ 複数台協調制御
設計開発方針	制御構造	プログラム+玩具+ キューブ
	制御手法	絶対位置検出・ 閉ループ制御
検証手法	体験価値	UX 主導開発・ MVP デザイン思考
	事業	リインスタートアップ
	設計・ 実現性	コンカレントエンジ ニアリング
		テスト駆動開発・ アジャイル開発

品ですが、汎用性と遊び方の多様化の両立のためにビデオゲームやPCのように基本部分とアプリを分離したプラットフォーム型モデルを、ロボットと専用タイトルという切り口で適用しています。これにより「共通要素を凝縮して量産し低コスト化と高機能・高性能・高品質を両立する」「きめ細かなニーズや趣向に合った鮮鋭なコンテンツを提供する」ということを同時に実現させることを指向しています。開発の分業・効率化も可能となりさまざまなクリエイターとのコラボレーションが可能になる上、商品説明においても「ゲーム機的」ということでコミュニケーションしやすくなっています。ソフトでは当たり前になった機能分離のアプローチをハードにも適用することでこうした効果を生み出しています。

ソフト基点の制御構造

toio の設計開発方針の中でも、特に制御構造については、実世界で起こるインタラクションをコンピュータ上でビデオゲームのように扱うなど「ソフト基点」の発想で設計・実装しています(図-3)。技術的には特に、実世界のキャラクタ(ロボット)の位置を知ること(自己位置同定)が鍵となり、独自のセンサを用いています。

この制御構造で特筆すべき点として、ロボットの自己位置同定によく使われるデッドレコニングやSLAM^{☆1}等の誤差蓄積を伴う手法を使わず、特殊

☆1 Simultaneous Localization and Mapping



■図-3 toio の制御構造

パターンが印刷されたマットと光学センサにより絶対位置をリアルタイムに直接検出する手法を構築したことが挙げられます。これにより実世界のロボットをタブレット上のオブジェクトのように、ドリフトやキャリブレーションを考慮することなくコンピュータモデルで容易に扱え、複数ロボットの協調制御もビデオゲームのキャラクターのように単純なモデルとしてアルゴリズムを適用できます。そのシンプルさから、小学生から意図したロボットの行動制御プログラムが直感的に書けることをユーザテストにて確認しています。

キャラクタ／ルール・シナリオ／ロボット制御を分離したアーキテクチャはUI設計における基本的モデルである「Model-View-Controller (MVC)」に近いとも言えます。作り手はそれぞれの範囲での作り込みに集中でき、さらにたとえばView部分の一部であるキャラクタの形状や表現は子どもをはじめとするユーザに開放し自由に創作してもらうこともできるようになりました。

このように、toioではソフト開発者がシンプルにプログラムを作る点を重視し、通常のロボットの定石にとらわれない独自のアーキテクチャや商品像としたことで、ソフト的には非常に直感的にロボットを扱えるようになりました。

テスト駆動型の検証手法

toioの商品像は技術シーズ基点ではなく「子どもが自分で作ったおもちゃが動き出し、まるでテレビゲームの主人公になったようにインタラクティブに遊べる」という体験を提供したいという動機から始まっています。このような特異なコンセプトは創案当初から自信を持つのは難しく、また対象とするユーザは主に小学生であり開発者自身で価値を包括的に自己評価することができないため、中身をいきなり開発せずまずUX（ユーザ体験）の価値から検証することとしました。

体験価値検証のための手法としては「デザイン思

考」、事業構築までを含めると「リーンスタートアップ」という手法が確立されていますが、考え方の根本は外部インタフェースの成立・整合を確保してから中身を開発する「テスト駆動開発」や、仕様を徐々に詳細化しながらソフト構築する「アジャイル開発」と同等です。UXや事業性の価値を検証する場合は製品やビジネスモデルの表層部分のプロトタイプ（最小ではキャッチコピーやカタログ案だけでもOK）を構築してユーザに理解および満足していただけるか・対価をいただけるか、をテストケースとして繰り返し検証します。こうした検証スタイルはソフト開発では一般的であり、プログラミング等で類似の経験があればその型を流用することができます。toioでも、実際に小学生や親御さんに対して数多く開催した体験会から意見を反映し徐々に製品像を固め、そこから必要な技術開発や製品設計を行うことで自信を深めつつ着実に製品化を進められました。

以上のように、toioはソフト的な発想や作り方をハードに適用することで、これまででない実世界でのインタラクティブな遊びを実現しています。

ソフト的発想は日常にも役立つ

ソフト的な発想やノウハウは、必ずしもハードウェアの企画・開発だけでなく、さまざまな仕事や普段の生活の中でも役に立ちます。ソフトウェアで培われたノウハウやアルゴリズムの利点を人の日常に取り入れ生産的な行動を心がけるような考え方は、Wing²⁾により“Computational Thinking”（邦訳：計算論的思考）として提唱されています。

私自身の経験としても、自分で実装し腹落ちしたアルゴリズムやロジックは生活や仕事にも転用しやすく、たとえば自分がある系の変数やエンティティの1つとすれば、過度に主観に陥らず客観的に物事を考えるきっかけになると感じています。

少々飛躍した例ですが、何かアイディアを探索する際、狭い視野（局所最適）に陥らないよう検討範

圏（解空間）全体から俯瞰し見直すことで最良と思える案にたどり着くことがあります。これは焼きなまし法等の全体最適化に似ているなど感じています。ほかに、たとえば事業構築において次の一步の最適解が見えない状況で次々行動を決断すべき際は、手堅い一手（線形予測）だけでなく将来化けるかもしれないものにも多少タネを蒔きながらリソースを配分し、結果を事後評価して改善しようなどと考えるのですが、これはロボット制御でも使われるカルマンフィルタ等の推定手法や機械学習にも一部似た部分があるので、こうしたアルゴリズムの発想を取り入れたりしています。

以上はやや突飛な個人的事例ですが、このようにソフト的な発想を日常に取り入れ、ある種のモデル化を行って自分や周囲の状況を観察してみると、客観的で納得のいく思考や判断の助けになる、というのもソフトウェアの経験を活かす1つの方法だと思います。このほか、プログラミング経験を仕事に活用する論考³⁾があり、参考になります。

製品も仕事もソフト・ハードが不可分な時代に

私は学生時代、指導教官とのディスカッションの中で出てきた「ハードウェアはポテンシャルを生み出す。ソフトウェアはポテンシャルを引き出す。」といった内容の言葉が深く心に刺さりました。以来、私はハードウェアとソフトウェアは表裏一体であり、双方を深く理解して使いこなすことで持てる力を十分に引き出そうと意識しています。

これは例えば人間の身体・精神と同じで、体格が良くても運動神経抜群とは限らず、その逆もしかり、というのと似ているかもしれません。

また、元来生物や自然にはハード・ソフトのような明確な境界はなく、ロボット工学やAIの研究においてはこれを踏まえた「身体性」が重視されつつあるように、技術的にはソフト・ハードを統合的に、もしくはさらに上位のシステムや概念で扱うことの重要性がますます大きくなっています。製品開発や企画の仕事においても、ITやSNSの発展により専門性や人とのつながりも多様化する中では、分野を超えた協業の重要性が増しています。体験価値としても、ソフト・ハードの境目はなくなり、統合的なデザインの視点が重要になっています⁴⁾。多様な視点や共感力が必要な時代には、キャリアの中で幾度か分野を越境することはより重要性を増していると思います。

仕事内容が短期で劇的に変化し、過去存在しなかった職種が次々と現れる時代が到来しています。キャリア上の専門性が変わる理由やきっかけは自発的・偶然含めさまざまかもしれませんが、ぜひそのチャンスを活かし、大きな目標に向かって、同じ社会の一員としてともに価値を作っていきましょう。

参考文献

- 1) toio™ : <https://toio.io>
- 2) Wing, J. M. (著), 中島秀之 (訳) : Computational Thinking 計算論的思考, 情報処理, Vol.56, No.6, pp.584-587 (June 2015).
- 3) 久野 靖 : プログラミング教育/学習の理念・特質・目標, 情報処理, Vol.57, No.4, pp.340-343 (Apr. 2016).
- 4) 渡邊恵太 : 融けるデザイン ハード×ソフト×ネット時代の新たな設計論, ビー・エヌ・エヌ新社 (2015).
(2019年3月5日受付)

■田中章愛 Akichika.Tanaka@sony.com

2006年筑波大学大学院修了, ソニー(株)入社。2013年スタンフォード大学訪問研究員。ロボットの研究開発や新規事業創出プログラムの企画運営を経てロボットtoio™を提案, 2018年よりソニー・インタラクティブエンタテインメントT事業企画室課長。商品企画と事業開発を担当。