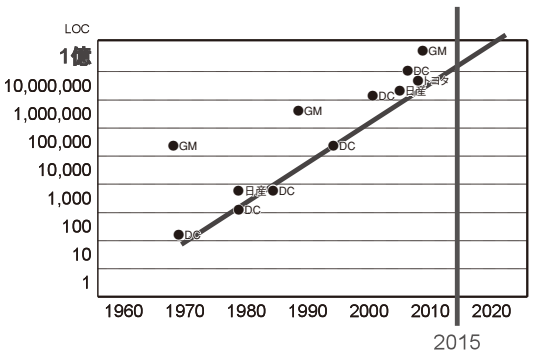
組込みソフトウェアアセット

CATS社　取締役副社長
渡辺　政彦

1．組込みソフトウェアの現状

モノ造り製造業において、組込みソフトウェア開発費の割合が増加しています。経済産業省の組込みソフトウェア産業実態調査では、全開発費におけるソフトウェア開発費の割合は06年度に40%、07年度では46.2%に増加しています。全開発費ではなく、組込み関連だけにすると、実に59.7%がソフトウェア開発費で、残り20.7%がハードウェア（電子系）開発費、11.9%がハードウェア（機械系）開発費になっています。

開発費が増大していることから、組込みソフトウェアの規模も増大しているのでしょうか。2007年5月に行われた九州組込みソフトウェア研究会１周年記念講演会でJasPar運営委員長が、2015年には車載ソフトウェアが１億行を超えると予測しています（図１）。

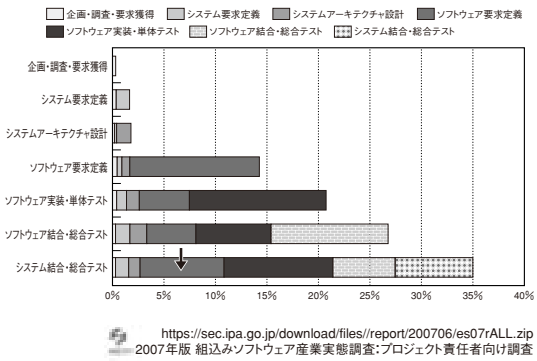


車両の現状及び将来技術とエレクトロニクス
May 24th 2007 NISSAN MOTOR CO.,LTD Kazutaka ADACHI

図１　車載電子システムのソフトウェア量の推移

このような急激な規模拡大は、組込み製品の品質問題を表面化させています。最近、携帯電話や自動車のリコール原因が組込みソフトにあると報道されることが多くなりました。07年経産省の組込みソフトウェア産業実態調査では、製品出荷後の品質問題の原因としてソフトウェア不具合率は43.8%を占め、ハードウェア不具合率の16.9%に比べて非常に高い割合になっています。品質問題発生による対策費用は、１件あたり100万円以下が５割程度ですが、２億円以上の企業も数%存在しています。

ソフトウェア不具合発見工程別件数比較と発生工程の内訳（図２）を見ますと、下流工程のシステム結合・総合テストで、上流工程のソフトウェア要求定義を原因とする不具合が多く見られます。



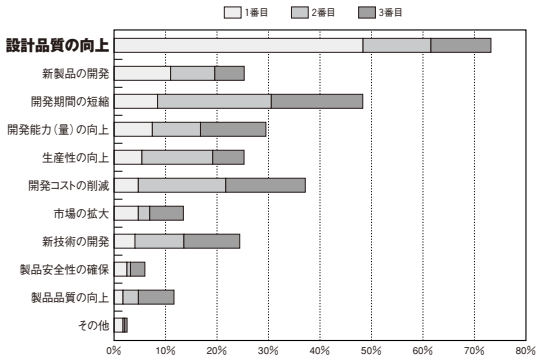
https://sec.ipa.go.jp/download/files//report/200706/es07rALL.zip
2007年版 組込みソフトウェア産業実態調査:プロジェクト責任者向け調査

図２　ソフトウェア不具合発見工程別件数比率と発生工程の内訳

上流工程のあいまいさが下流工程で露呈するという良くないソフトウェア開発スタイルから未だに脱却できていないことを数字が示しています。そのためでしょうか、組込みソフトウェアに課せられている課題として断トツ１位なのが設計品質の向上です（図３）。

2．組込みソフトウェアの課題

こうした状況を組込み分野のソフトウェア・クライシスと呼ぶ人がいます。20年以上前、携帯電話もカーナビも身近にない時代に、ソフトウェア・クライシスのパネルディスカッションが行われて、その様子が情報処理Vol.26 No.9（Sep.1985）に記載されています。現在では想像



https://sec.ipa.go.jp/download/files//report/200706/es07rALL.zip
2007年版 組込みソフトウェア産業実態調査:経営者・事業責任者向け調査

図３　組込みソフトウェアに課せられている課題

もつきませんが、当時はソフトウェア技術者なる職業が一般的に通用しない社会的問題や、漢字コードが統一化されていない技術的問題があったことを知ることができます。この問題は現在までに解決されています。しかし、次の３点については、20数年来抱えている課題といえます。

①ソフトウェアに従事している人たちは、自らの自動化が行えていない。

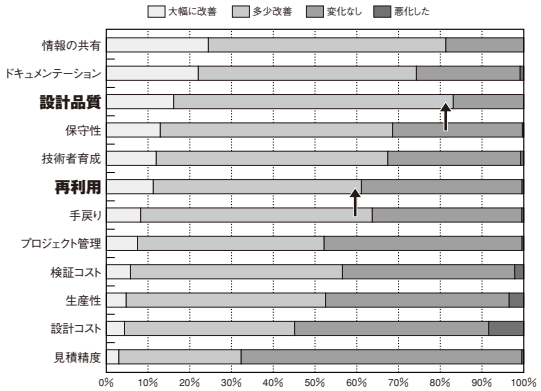
②ソフトウェア部品を再利用するために、ソフトウェアの設計法を抽象化して蓄積し、再利用時には、具体化の技法の開発が必要である。知識工学的手法というものをソフトウェア工学と融合させる必要がある。何を知識とするかについては、抽象化技法あるいは具体化技法と密接に結びついてくると考えられる。

③ハードウェアの設計に１台何千万円もする機械を技術者に与えるのに、ソフトウェアの設計者のためには、せいぜい100万円ぐらいの端末を与えて、しかもそれを数人でシェアさせる。このことは、生産性の向上を妨げている。

これらの問題は、まったく解決されていないわけではありません。特に①と②の項目は、20数年前から着実に進化をしています。

３．モデルベース開発

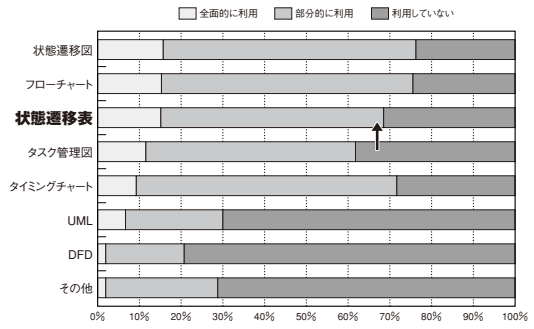
①と②の問題はモデルベース開発が、その解決方法の１つです。状態遷移表モデルという抽象化からCコードなる具体化を自動的に生成するモデルベース開発は、すでに数多くの企業で適用しています。ZIPC WATCHERSのバックナンバーで毎年増えている適用事例を参照してください（http://www.zipc.com/support/download/ instance.html）。07年経済産業省組込みソフトウェア産業実態調査では、モデリング手法の導入により設計品質に改善効果があったとの回答は８割を超えています（図４）。モデリング手法により再利用に効果があったとの回答は６割になっています。



https://sec.ipa.go.jp/download/files//report/200706/es07rALL.zip
2007年版 組込みソフトウェア産業実態調査:経営者・事業責任者向け調査

図４　モデリング手法の導入効果

モデリング手法の利用状況では状態遷移表を全面的・部分的に利用する割合は７割近くです（図５）。



https://sec.ipa.go.jp/download/files//report/200706/es07rALL.zip
2007年版 組込みソフトウェア産業実態調査:経営者・事業責任者向け調査

図５　モデリング手法の利用状況

およそ20年前のパネルディスカッションの指摘では、抽象化と具体化ができれば、再利用が促進すると指摘しています。

4．組込みソフトウェアアセット

完全なソフトウェアを再利用できることは、4年後のROIが30倍になることが示されています（図6）。

技術	4年利用後の1ドルあたりの投資効果($)
完全なソフトウェア再利用	30
I-CASE	25
ソフトウェアの品質測定	17
ソフトウェアの見積もりツール	17
正規の設計インスペクション	15
正規のコードインスペクション	15
オブジェクト指向プログラミング	12
ソフトウェアの生産性測定	10
ソフトウェアプロセスアセスメント	10
機能的尺度	8

Assessment and Control of Software Risks : Prince-Hall, 1994,
邦訳:島崎・富野「ソフトウェア病理学」共立出版,1995

図6　ソフトウェア技術対投資効果

『完全な』とは難しい定義ですが、今までのコードレベルやライブラリィレベルの再利用から、今後はモデルレベルでの再利用、プロダクトラインやRASを用いることで、より『完全な』再利用になると考えます。

4.1　再利用資産仕様

OMGが規格するRAS（Reusable Asset Specification）では、ソフトウェアの成果物であるソースコード、モデル、テストケース、ドキュメントなどを成果物として定義しています。これらの成果物だけでは完全な再利用は行えないとして、『分類』、『ソリューション』、『使用法』、そして『他のアセットとの関係』の4つをコアRASとして定義しています（図7）。土地の資産価値をあげるには、駅、病院、学校などのインフラを整備することと同じように、成果物のソフトウェアアセットの価値を引き上げるには、それが有効に活用されるためのインフラ、ここでは再利用のための情報が必要になります。

こうしたRASや後述するプロダクトラインなどを導入する場合、ソフトウェア技術者にPCとIDEを与えるだけの投資では済みません。ソフトウェアアセットを管理するための大規模なリポジトリシステムの構築や、ソトウェアアセットを構成する各種のモデルベースツールの整備などの投資が必要になります。

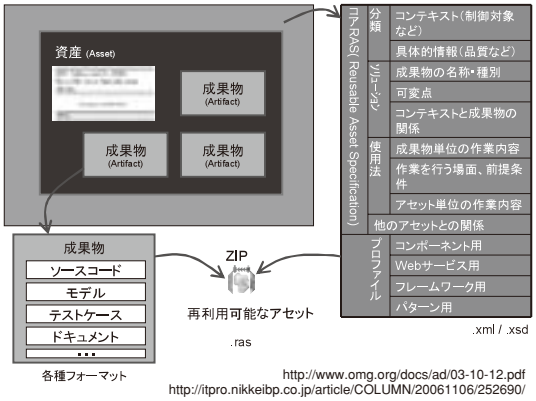


図7　ソフトウェアアセットとは

4.2　知的資産経営

ソフトウェアは、『知的労働』の成果物なのですが、『知的』と『労働』とのどちらに比重をおくかで、経営方針は変わります。『労働』に力点をおくと従来型のレイバーベース開発になり、『知的』の方におくとアセットベース開発になります。レイバーベースとは労働集約型ソフトウェア開発で、見積もりは人月を基本とし、ソフトウェアの資産化を行うことには消極的です。どんどん作ることが生産活動であるととらえるので、コードクローン分析を行うと、品質に悪い影響を与えるコピーペーストコードが大量に見つかったりします。アセットベース開発とは、資産活用によって利益を生み出すことを目的としますから、ソフトウェアの資産化を積極的に進めます。

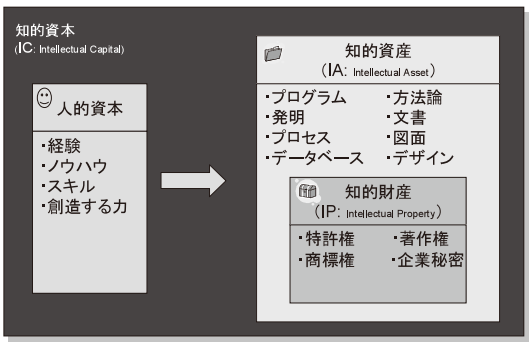
A･トフラーは、第1の波『暴力』、第2の波『お金』から、第3の波『知識』へとパワーシフトが起こると述べています。P･ドラッカーは、基本的な経済資源は、『資本（お金）』、『天然資本（土地）』でも、『労働』でもなく、それは『知識』となると述べています。経済産業省は「知的資産経営の開示ガイドライン」を平成17年に公表しました。企業の競争力の源泉として、「知的資産」を活用した経営の果たす役割が重要

となってきており、資本市場側も企業等の評価において、こうした無形の資産をより重視する傾向が国際的に強まっているとガイドラインでは指摘しています。

4.3　ソフトウェア資産

経営者はレイバーベースでは未来がないことは気づいています。しかしながら、ソフトウェア開発の実態が理解できないため、改革を行えずにいます。そこで、ソフトウェア管理職が、もう少し経営に分かる言葉で話しをしてみてはいかがでしょうか。プロジェクトマネジメントでデスマーチを食い止めるのに精一杯だとの声も聞こえますが、もしかすると経営者を巻き込んだ、コーポレートマネジメントレベルでプロジェクトマネジメントできれば、デスマーチがなくなるかもしれません。ソフトウェアの再利用、つまり資産化に投資してもらえるのですから。

ソフトウェアは知的労働の産物であり、知的資産です。知的資本は知的労働者である人的資本と、知的資本から作り出されるソフトウェア（プログラム、文書、図面）、プロセスやブランドなどの無形の資産である知的資産を含めたものです。知的資産はさらに法律によって保護される知的財産になるものがあります。知的資本、人的資本、知的資産、知的財産の関係を図示したものが図8です。



パトリック・サリヴァン：知的経営の真髄：東洋経済新報社：2002

図８　知的資産とは

プロセス、ブランド、そして顧客とのネットワークといった知的資産からソフトウェアに注目したものがソフトウェア資産です。

ソフトウェア資産にはWord、Excelといった事務職や技術職も使うようなアプリケーションソフトや、Visual Studio、ZIPCといった技術職が使うアプリケーションがあります。このようなソフトウェア資産をどのように管理するかといった基準を、ソフトウェア資産管理コンソーシアムではCMMのような成熟度を入れて管理目標を制定しています（図9）。

Ver. 1.0　平成14年10月30日

管理目標 / 成熟度（ソフトウェア資産管理コンソーシアム http://www.samconsortium.org）	【方針】自組織に適したソフトウェア資産管理の方針・規定等が作成され、遵守されていること	【体制】体制が整備されていること	【所有】所有ライセンスの種類・数量が把握されていること	【導入】どのハードウェアにどのソフトウェアがインストールされているか、また、そのソフトウェアがどのライセンスに基づいているか把握されていること	【証明】所有ライセンスが証明可能であること	【認識】ソフトウェア不正使用は不法行為であるとの認識が浸透していること	【理解】ライセンス内容が理解されていること	【環境】ユーザが不正使用しにくい環境が構築されていること	【購コ】購入コストの削減	【管コ】管理の効率化、コスト削減	【セキ】セキュリティ上の配慮がなされていること
レベル5 最適化されている段階											
レベル4 管理されている段階											
レベル3 定義されている段階											
レベル2 反復可能な段階											
レベル1 初期／場当たり的な段階											
レベル0 管理が存在しない段階											

図９　ソフトウェア資産管理基準

4.4　組込みソフトウェアアセットの特性

本稿で取り扱うソフトウェアアセットは、購入して使用するパッケージソフトウェアの資産ではなく、開発した組込みソフトウェアの資産を意味します。区別するために、意識的に資産をアセットと英語読みしてカナにしています。ソフトウェアアセットとは、ソフトウェア開発という知的労働の成果物のことです。ソフトウェア開発の成果物には、ソースコードはもちろんのこと、モデル、テストケース、自然言語によるドキュメントなどが含まれます。OMGのRAS（再利用資産仕様）では、ソフトウェアアセットの再利用を促進するために、ソフトウェ

アアセットの分類、ソリューション、使用法を成果物にラップして、圧縮保存できる形式を提供しています（図7）。

ソフトウェアアセットの種類の１つに組込みソフトウェアアセットがあります。組込みソフトウェアの特徴をイメージしたものが図10です。

組込みソフトウェアアセットの特性としては、次の３つがあります。

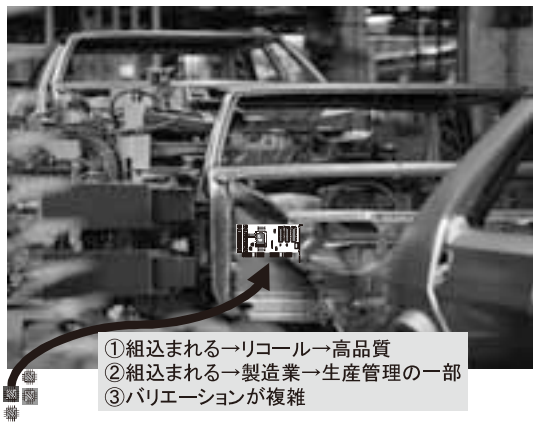


図10　組込みソフトウェアとは

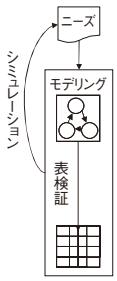
①組込まれる、つまり、最終製品で不具合があると、リコールとなる場合があり、改修に多大のコストがかかる。また、不具合による安全性が人命に及ぶ場合もあるので、高品質、高信頼性が求められる。

②組込まれる、つまり、最終製品は、製造業における生産管理システムの一部である。このため、他の活動との連携が求められる。

③最終製品の仕向地や、オプションなど多彩な製品ラインナップをソフトウェアが対応するため、バリエーションが多く、かつ、これを動的に可変にするほどリソースをもらえない。つまりメモリ容量をけちるために、コンパイルオプションで静的にプログラムの構造を変える。

4.4.1　高品質

リコールなどをおこさない高品質なソフトウェアであるかを知る方法の１つに、ISO9126（JIS X0129）のソフトウェア品質特性があります。状態遷移表モデルにすることで、漏れ抜けを防止することは、機能性という品質特性の正確性なる副特性に効果があります。そのほか、



品質特性	副特性	定義	技法	測定方法
機能性	合目的性	明示された仕事に対する機能の集合が存在し、適切であることをもたらすソフトウェアの属性 **仕様通りに動くか？**	ニーズをモデル化し、このモデルをシミュレーションすることで、ニーズと合致しているかをテストする	テストケース
	正確性	正しい結果もしくは正しい効果、または同意できる結果もしくは同意できる効果をもたらすソフトウェアの属性 **仕様の抜け・漏れが防止できるか？**	状態図から状態表に変換し、空白セルを定義する	空白定義

JIS X 0129-1994（ISO/IEC 9126:1991）ソフトウェア製品の評価-品質特性及びその利用要領

図11　高品質設計（１／２）

品質特性	副特性	定義	技法	測定方法
信頼性	障害許容性	ソフトウェアの障害部分を実行した場合、又は仕様化されたインタフェース条件に違反が発生した場合に、仕様化された達成のレベルを維持する能力をもたらすソフトウェアの属性 **想定外のことに対応できているか？**	想定外事象をelseイベントとする	モデル内のelseイベント定義数
保守性	解析性	欠陥もしくは故障の原因の診断又は改訂すべき部分の認識に必要な労力に影響するソフトウェアの属性 **不具合発生時に解析しやすいか？**	事象No、状態Noのリングバッファログ	解析時間
	変更性	改訂、障害の除去又は環境への対応への対応に必要な労力に影響するソフトウェアの属性 **変更しやすいか？**	表による変更容易性	変更時間
	試験性	改訂したソフトウェアの妥当性の確認に必要な労力に影響するソフトウェアの属性 **試験しやすいか？**	表によるチェックリストの作成容易性	試験時間

JIS X 0129-1994（ISO/IEC 9126:1991）ソフトウェア製品の評価-品質特性及びその利用要領

図12　高品質設計（２／２）

『品質特性・信頼性：副特性・障害許容性』、『品質特性・保守性：副特性・解析性』、『品質特性・保守性：副特性・変更性』、『品質特性・保守性：副特性・試験性』に有効です。図14、図15に示すような状態遷移モデルベースの品質に対する効果をソフトウェア技術者、ソフトウェア管理者は、経営者に示し、投資を促したいものです。

不具合が生命の危険にまでおよぶような場合、ISO15026（JIS X0134）のソフトウェアの信頼性等級、完全性水準を利用し、そのILを高く設定することが求められます（図13）。

リスク抑制完全性水準の手法に準形式的記法とあるのが状態モデルやUMLのことで、形式的記法とは、CafeOBJやVDMなどの形式仕様言語のことです。

アクティビティ	特性	ソフトウェアにおいて個々の完全性水準（IL: Integrity Level）の信頼等級を達成する手法	
		IL	手法
ソフトウェア要求分析	・精度 ・完備製 ・正しさ ・抽象度 ・一貫性 ・検証性	1 2 3 4	構造化アプローチ IL1+準形式的記法 IL1+**形式的記法** IL3+**形式的証明**
ソフトウェア設計	・要求事項への追跡可能性 ・検証性 ・モジュール性 ・抽象度	1 2 3 4	構造化アプローチ IL1+準形式的記法 IL1+**形式的記法** IL3+**形式的証明**
ソフトウェアコーディング	・設計記述への追跡可能性 ・プログラム構造・構成の非曖昧性 ・プログラム言語の標準 ・保守性	1 2 3 4	構造化プログラミング IL1+型制約言語 IL2+言語の安全なサブセットに制限 IL3+形式的証明

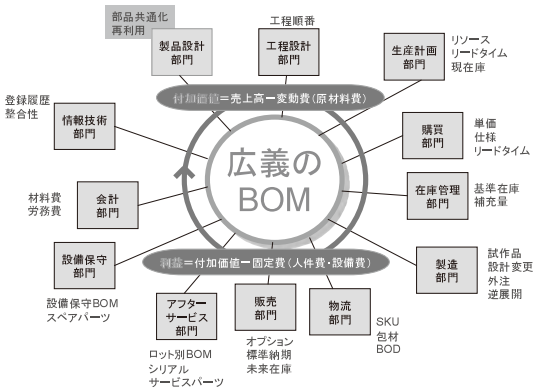
システム及びソフトウェアに課せられたリスク抑制の完全性水準（JIS X 0134:ISO/IEC 15026）

図13　ソフトウェアの信頼等級及び完全性水準

4.4.2　生産管理の一部

組込みソフトウェアは、製造業における作業工程のなかの一部です。製造業では通常、部品表（BOM）を中心に生産を管理するシステムになっています。このBOMに組込みソフトウェアが入ってないことが、経営者を組込みソウトウェアから遠ざけている理由の１つではないでしょうか（図14）。

全開発費におけるソフトウェア開発費が６割近くを占め、ソフトウェアの不具合の率は４割を超える状況です。早急に組込みソフトウェアをBOMに組み入れ、生産管理システムの中で組込みソフトウェアに対する投資を判断する必要があります。しかしながら、従来のBOMが取り扱ってきたマテリアルと組込みソフトウェアとでは、特質が異ります。例えば、目に見えない、バージョン・リビジョンがたくさんある、1つでも大量生産でもコピーすればよいなどです。組込みソフトウェアを生産管理システムにどのように組み入れるかについて、組込みソフトウェア管理職の協力が欠かせないのです。



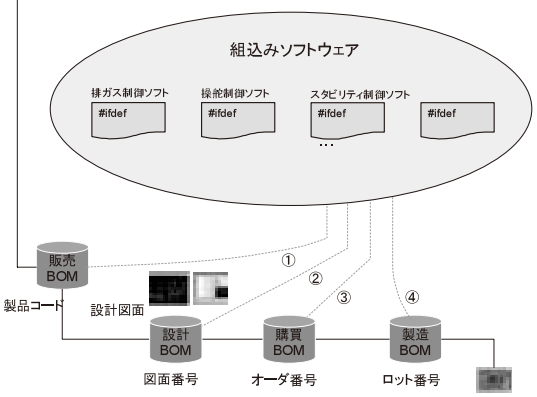
佐藤知一・山崎誠：BOM/部品表入門：日本能率協会マネジメントセンター

図14　BOM（Bill of Material）とは

BOMは複数のBOMから構成されています。それぞれのBOMにはそれぞれのステークホルダーがいます。ですから、組込みソフトウェアには組込みソフトウェアBOMが構築されるべきでしょう。それぞれのBOM間には関連があります。組込みソフトウェアBOMが関連する既存BOMは次の４つです（図15）。

製品仕様

主要装備一覧表 http://lexus.jp/models/ls600h/spec/equiment07.html		LS600h 1	S	S1	U	U1	LS600hL 後
走行性能	ステアリング						
	ギヤ比可変ステアリング［VGRS］	●	●	●	●	●	●
	電動パワーステアリング［EPS］	●	●	●	●	●	●
	サスペンション						
	前後マルチリンク式サスペンション	●	●	●	●	●	●
	電子制御式エアサスペンション（AVS機能付）	●	●	●	●	●	●
	スタビライザー（フロント・リア）	●	-	●	●	●	●
	アクティブスタビライザー（フロント・リア）	-	●	-	-	-	-
	ブレーキ						
	フロント：対向4ポッドキャリパー・φ357mmスパイラルフィン式ベンチレーテッドディスク	●	●	●	●	●	●
	リヤ：対向2ポッドキャリパー・φ335mmスパイラルフィン式ベンチレーテッドディスク	●	●	●	●	●	●
	電子制御ブレーキ［ECB］	●	●	●	●	●	●
安全性能	VDIM（アクティブステアリング統合制御付）	●	●	●	●	●	●
	ABS（電子制御力配分制御付）	●	●	●	●	●	●
	ブレーキアシスト	●	●	●	●	●	●
	VSC	●	●	●	●	●	●
	TRC	●	●	●	●	●	●
	ヒルスタートアシストコントロール	●	●	●	●	●	●
	プリクラッシュセーフティシステム ミリ波レーダー方式	△	△	△	△	△	▲
	プリクラッシュセーフティシステム ドライバーモニター付ミリ波レーダー・ステレオカメラフュージョン方式	△ ▲	▲	▲	△	△	●



①販売BOM　②設計BOM　③購買BOM　④製造BOM

図15　組込みソフトウェアとBOM（２／２）

4.4.2.1　販売BOMと組込みソフトウェアBOM

販売BOMは、企画や営業が製品の仕様を考えるためのものです。製品仕様に、組込みソフトウェアが大きく関与しており、製品コードと組込みソフトウェアとの紐付けが必要になってきています。

4.4.2.2　設計BOMと組込みソフトウェアBOM

設計BOMは、製品設計部門の図面が管理されます。製品設計部門では部品共通化が重要な設計業務の１つです。2015年には１億行を超えるといわれるソフトウェアの設計において、製造業伝統の部品化再利用設計に学ぶところは多いはずです。モデルといった準形式言語による設計図面番号と機械設計や電子設計の図面番号と紐付けを行います。こうすることで、電子部品を変更した場合、影響する組込みソフトのモデルが何かを即座に分かる仕組みができあがります。

4.4.2.3　購買BOMと組込みソフトウェアBOM

購買BOMでは、従来の人月単価購入から、組込みソフトウェア部品購入へと変わることでしょう。納期、費用の見積もりに高いリスクがある現状では、グローバルな消費者ニーズに合わせて、製品仕様変更に即時対応できなければ、競争力が失われます。

4.4.2.4　製造BOMと組込みソフトウェアBOM

製造BOMでは、調達したソフトウェア部品を用いて、何を試験し、何を試験しなくて良いかといった組込みソフトウェアの動作保証が必要になります。ソフトウェアの完全性を証明できれば、その資産価値は高まります。

組込みソフトウェアBOMの構築には、モデルベース開発に対する投資以上を覚悟する必要があります。ですから、経営者の合意がなければ進めません。また、これをおざなりにしておくと、そのツケは数年後に、企業の競争力低下として現れることでしょう。組込みソフトウェア管理職の皆さんは、社保庁のように問題を先送りしてはいけません。幸い、最新経営の関心は「知的経営」にあり、固定資産より無形資産、それも知識による資産がこれからの企業の競争力の源泉になることを経営者は勉強しています。ですから、経営者の言葉で、組込みソフトウェアアセットの重要性を進言すればよいのです。

4.4.3　バリエーション

プロダクトラインでは、製品開発とコア資産開発を明確に分けることで、部品共通化という抽象化を行い、再利用を促進させます。部品共通化のためにはバリエーションモデルをどのように取り扱うかが重要です。バリエーションの抽象化技法として、フィーチャ図（図16）やADL（図17）などが現在提唱されています。

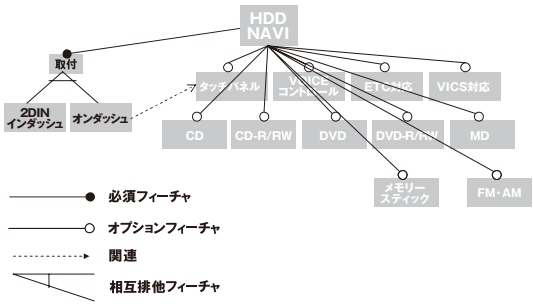


図16　フィーチャ図

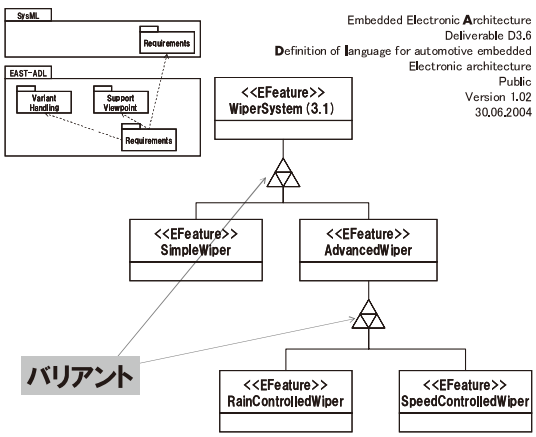


図17　ＡＤＬ

バリエーションの具体化技法としては、継承やパラメタなどプログラミング言語環境で実現されています（図18）。

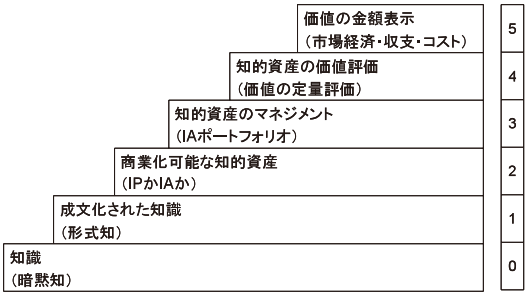
バリエーションの型		
メカニズム	特化する時	可変性の型
継承 Inheritance	クラス定義時	既存定義に追加、変更して特化させる 例）LongDistanceCall inherits from PhoneCall.
拡張 Extension	要求時	システムの使い方に別に使い方の定義を追加する 例）WithdrawalTransaction extends BasicTransaction.
使用	要求時	別の使い方の機能が含まれることを定義する 例）WithdrawalTransaction uses the Authentication use.
コンフィグレーション	ランタイム前	コンポーネント特化のためリソースファイルなどを分ける 例）JavaBeans properties file
パラメタ	コンポーネント実装時	限定できない機能定義を実際の使用時点で定義する 例）calculatePriority(Rule)
テンプレートインスタンス化	コンポーネント実装時	限定できない仕様を実際の使用時点で仕様の型を決める 例）ExceptionHandler<Container>
生成	ランタイム前またはその間	ユーザ入力から定義をツールが生成する 例）コンフィグレーションウィザード

http://www.sei.cmu.edu/productlines/frame_report/comp_dev.htm

図18　バリエーションの型

4.5　組込みソフトウェアアセットマネジメント

P・サリバンは知的資産マネジメントを6段階に分けています（図19）。



パトリック・サリヴァン：知的経営の真髄：東洋経済新報社：2002x

図19　知的資産マネジメントとは

レベル 0 は知識が暗黙知として個人の頭の中に存在している状態です。レベル 1 では、この暗黙知を形式知として成文化された知識とすることです。レベル 2 の成文化することで、その知識が商業化可能な知的資産とするかどうか、つまり知的財産（IP）にするか、知的資産（IA）にすることができます。レベル 3 では知的資産を管理し、レベル 4 でその知的資産の価値評価を行い、レベル 5 では具体的な金額としてその価値を表示します。

4.5.1　知識

組込みソフトウェアにこの知的資産マネジメントを適用すると、レベル 0 は、組込みソフトウェアが個人に依存しており、その個人がいなくなるとメンテ不能になる可能性がある状態です。

4.5.2　成文化された知識

レベル 1 では、組込みソフトウェアに関して形式知としてモデル化が行われています。モデルの完全性を証明するなど、形式手法がモデルベース開発に取り入れられます。

知識が人の頭の中にあるレベルから文書によって継承されるレベルになります。

4.5.3　商業化可能な知的資産

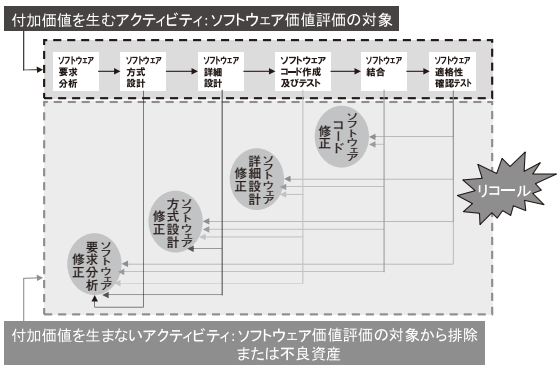
レベル 2 は、モデルで特許をとる（IA）か、とらないか（IP）を検討できます。

4.5.4　知的資産のマネジメント

レベル 3 では組込みソフトウェアアセット単位ごとに投資（もっとテストをする、基本性能を上げる、拡張性を高めるなど）するか否かを分析し判断できます。

4.5.5　知的資産の価値評価

レベル 4 では、組込みソフトウェアの価値評価を行えるようになります。ソフトウェアの価値評価については、2007年に経済産業省のエンタープライズ系ソフトウェア開発力強化推進ＴＦの見積もり手法部会でソフトウェア価値評価WGが発足しました。横浜国際社会科学研究第７巻で、『知的無形資産の価値評価に関する研究―ソフトウェアの価値評価を中心にして―』では、ソフトウェア開発プロセス中のフィードバック工数を不良資産としてみる考えを示しています（図20）。



横浜国立大学大学院　前田公彦：『横浜国際社会科学研究』第7巻
知的無形資産の価値評価に関する研究―ソフトウェアの価値評価を中心として―　平成14年8月29日

図20　修正コスト・アプローチ

ソフトウェア開発の費用増大やソフトウェア不具合に対する会社の利益損失について、会社の経営責任を問うべきではないかと問う株主。近い将来、デスマーチを恐れるのはソフトウェアプロジェクトリーダだけでなく、経営者自らが対策を講じるようになるかもしれません。

4.5.6　価値の金額表示

レベル5では、組込みソフトウェアアセットに価格がつき、市場に流通します。07年経産省の組込みソフトウェア産業実態調査では、ソフトウェア開発費の内訳で、ソフトウェア購入費が1割未満ですが、将来、3割、4割となることで、組込みソフトウェア業界の新たな魅力と国際的な競争力が生まれると思います。

5．おわりに

組込みソフトウェアをレイバーベース開発のままでいると、新3K（「厳しい」、「きつい」、「帰れない」）となるのでしょう。何故なら、労働力を提供してお金を得るとなれば、この労働力にかかる費用をより削減できれば、より利益となる構造なのですから。高齢化社会を迎えて、看護がどれだけ重要になるといっても、利益重視を株主が主張すれば、そこに働く人たちの対価は、できるだけ安くするように経営者は舵をきります。

一方で、組込みソフトウェアを資産として考えるアセットベース開発を導入したとて、やはり3Kになると思います。でも、その内容が違います。再利用してもらえるソフトウェア部品を創造し、改良し、保守することは、「厳しい」ことです。ソフトウェア部品が売れないという「きつい」リスクを負うことになります。それでも、常にアドホックに開発現場をたらい回しされ、体力や交渉力が創造力や知識力より重視されることがなくなって、無我夢中でソフトウェア部品のことを考えていて「帰る」時間を忘れてしまうくらい楽しい業界になるのではないでしょうか。

だって、モノ造りは本来楽しいことで、エンジニアはそれを追及する職業なのですから。