

The logo features the text "IAR visualSTATE" in a bold, sans-serif font. The "IAR" is in a larger, bolder font weight than "visualSTATE". The background of the entire page is a vibrant yellow, decorated with abstract circuit-like patterns, including a large circular shape on the left and a series of smaller circles connected by a line on the right. The bottom right corner contains three overlapping circles with the numbers 16, 32, and 8 inside them.

IAR visualSTATE®

アイディアからターゲットへ！

 **IAR**
SYSTEMS

DIFFERENT ARCHITECTURES.
ONE SOLUTION.

www.iar.com

IAR visualSTATEのコンセプト

IAR visualSTATEは設計、テスト、状態/イベント・モデルをベースとしたリアルタイム・アプリケーション作成のためのパワフルなテクノロジーです。IAR visualSTATEツールは複数のテクノロジーを単一のソリューションへと統合。コーディング、テスト、文書化が大幅に簡略化されますから、最も大切なこと、すなわちシステム構築に専念できます。

IAR visualSTATEツールを使用すれば、開発サイクル中の様々なアクティビティ間を簡単にナビゲートすることが可能です。目標とする製品のシンプルなアウトラインから始め、累積的に特徴、機能を追加してゆくインタラクティブ開発モデルをサポートします。いつでも、その時点までの作業を検証し、実証することができます。

ヒューマン・コミュニケーション

グラフィカルにデザインされたフォーマットを使用することにより、エンジニアであろうとなかろうと、ソフトウェアの振る舞いがより簡単に理解できるようになります。新しくプロジェクトチームに加わった人も、従来必要とされたよりずっと短時間でソフトウェアの全容をつかむことが可能です。顧客と問題点を話し合う場合も、このフォーマットにより簡単に理解し合えるようになります。

プロトタイプング・コンセプトにより、製品、あるいはその一部のプロトタイプ作成を迅速に行うことが可能になりました。これは様々なソリューションをテスト、評価するときにも使用でき、ユーザーフレンドリーにプロトタイプを作成できるため、エンドユーザーも評価に参加することが可能です。顧客や販売担当者は、要件スペック作成が完了する以前でさえ、製品を試してみることができるのです。

ツールが完全に統合されているので、同じ作業を繰り返すことはなくなりました。設計したすべての要素はテスト、シュミレート、プロトタイプ化され、ターゲット日に稼働可能です。

傑出したテクノロジー

IAR visualSTATEはアプリケーション実装に関する従来の多くの問題を技術的に解決しました。IAR visualSTATEは自動的にターゲット・システム用コードを生成し、そしてこのコードは設計と完全に一貫性があります。生成されたコードは完全に確定的に実行されるため、アプリケーションは常に一定の振る舞いをします。競争状態、その他の「ランダム」な行動はあり得ません。

確定モデルのおかげで、アプリケーションの完全なDynamic Formal Verificationを行うことが可能です。IAR visualSTATE Verificatorがシステムのすべての組み合わせをチェックし、エラーを報告します。

最新の文書

現行の実行状態を示した最新の文書にアクセス可能です。設計説明書とテストレポートにはアプリケーションがどのように設計され、どのようにテストされたかが記述されます。

文書

フレキシブルな実装

IAR visualSTATEで作成されたアプリケーションは、リアルタイム・オペレーティングシステムの有無に関わらず、実質的にすべてのプラットフォームに導入可能です。通常、手作業で書かれたアプリケーションよりもコード数、メモリ容量ともに少なく済み、8ビット・システムに対してさえも適合します。

実装



グラフィカル・デザイン

IAR visualSTATEはコンカレントおよび階層両状態マシンを伴うUML準拠のステートチャートを提供します。これにより、マルチタスク・アプリケーションにおける並行スレッドをより視覚的に表現することが可能です。

プロトタイピング

迅速なプロトタイプ作成

製品が最終的にどのようなものになるか、他の人にも見せることができます。IAR visualSTATEは設計をもとにしたWindowsプロトタイプを簡単に生成します。これはマーケティングとエンジニアリング間のコミュニケーション強化に大いに役立ちます。

テスト

広範にわたるテスト

IAR visualSTATEにはきわめてパワフルな検証、実証ツールがあります。完全ダイナミック・フォーマル検証テクニックを採用し、対話式シミュレーション、テストスクリプト実行、高度なブレイクポイント、静的、動的解析などが可能です。

コーディング

自動コード生成

IAR visualSTATEはターゲット・プラットフォームへの統合用の「生産開始準備完了」状態のコードを生成します。自動的に生成されたコードはターゲット・アプリケーションが設計通りに動作することを確認し、手作業によるコードエラーの可能性を排除します。

「IAR visualSTATEは高品質の包丁のようなもの。一度慣れてしまうと、それなしには仕事できません。迅速、正確、完璧なやり方でタスク実行を可能にしてくれます。」

Anders Gunnarsson,
Viking Sewing Machines

ソフトウェア設計だけに専念できます

今日の複雑なソフトウェア・システムを扱う際、視覚的な文書の助けを借りず、ソフトウェアの全容を理解することは不可能です。IAR visualSTATEはアプリケーション作成にグラフィック・フォーマットを使用します。常に視覚的にアプリケーションの全体図を見ることができ、文書と実際のアプリケーションは完全に一致しています。

状態チャート

状態チャートはシステムの動作モデルを記述する際に使用される確固たる手法です。このフォーマットは同時に進行し、かつ相互関連のある数多くのプロセスや実行スレッドを伴うリアルタイム・システムに特に適しています。それぞれのプロセスあるいはスレッドは状態チャートで記述することが可能で、ヒエラルキー・モデルを使用することにより、様々なレベルで最大限に明瞭な設計が可能になります。

状態チャートを使用することにより、アプリケーションのアウトラインから開始し、徐々に詳細レベルの機能性を加えてゆくといった、より相互的で反復的なワーキングモデルを構築することが可能です。IAR visualSTATE Validatorを使用すれば、いつでもモデル動作をシミュレートすることができ、さらに、いつでもプロトタイプやターゲット実装モデルを作成することが可能です。

先端GUI

visualSTATE Designerのグラフィカル・ユーザー・インタフェース(GUI)が状態チャート作成用の先端ツールを提供します。ウィンドウレイアウトは完全にユーザー構築可能で、複数の状態チャートを編集あるいは参照用に同時に開くことが可能です。

プロジェクト・ブラウザ

ツリー・ブラウザを使用し、簡単にプロジェクトの全体像を概観し、ナビゲートすることが可能です。システムのすべての要素はブラウザで見て、アクセスすることができ、該当するエレメントをダブルクリックするだけで見たい図面を表示することができます。

色分けとサイズの変更

状態チャートを最大限に明瞭にするため、すべての図表オブジェクトは個別に色とサイズを変更することができます。これにより、アプリケーション内における通常動作とエラーハンドリングを区別することも可能です。エラー処理に含まれる遷移を赤で、その他を黒で表示すれば、誰にでも一目瞭然です。

基本的なエレメントは以下の通りです：

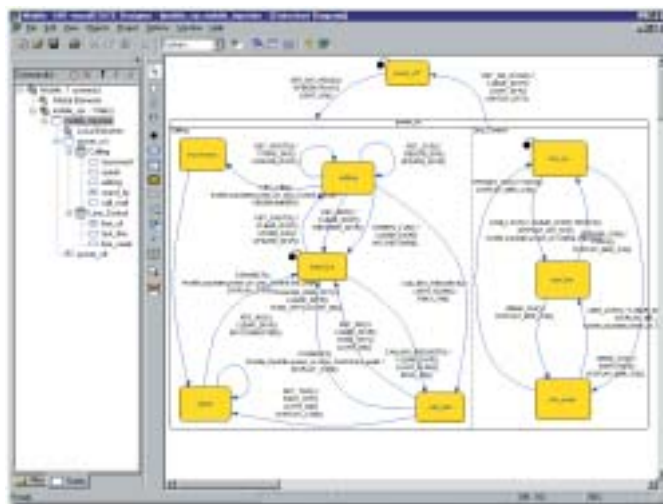
ステート

「ステート」はシステムが何をしているかを説明し、通常はイベント発生を待っています。待機中にタスク実行も可能です。

遷移

「遷移」はどのような条件下であるステートが他のステートに変化するかを記述します。「遷移」はイベントによって引き起こされ、遷移中に一つ以上のアクションを実行することが可能です。

ユーザーフレンドリーなドラッグ&ドロップ式のインタフェースだから、ステートマシンモデルの意図された動作だけに専念することが可能。



イベント

「イベント」は、「ボタンが押された」、「時間切れ」、「コミュニケーション受信」等、何かが発生したことをステートマシンに伝えるメッセージです。

アクション

「アクション」は外部アプリケーション中のC関数実行の引き金になります。この関数は例えば、出力設定、LCDへのテキスト書き出し、メッセージ送信などを行います。

IAR visualSTATEには以下も含まれます：

- コンカレント リージョン
- ガード、シグナル、パラメータ
- ユニットステート、イニシャルステート
- タイマアクション、開始/終了アクション
- ヒストリステート、ディープヒストリーステート
- 内部および外部変数

完全なテスト

従来の手作業による実装戦略では、アプリケーションの隅から隅まで完全にテストすることは不可能でした。また、オペレーションのすべての組み合わせを識別することさえも困難です。

visualSTATEの基本テクノロジーによって、アプリケーションを網羅した完全テストが可能になりました。これには相互シミュレーション、プロトタイプ化、ターゲット・テスト用のReal-Link、完全ダイナミック・フォーマル検証、静的、動的解析が含まれます。

検証

対話式シミュレーションにより、アプリケーションがイベントにどのように反応するか見ることが可能です。シミュレータはアプリケーションに関する全情報、すべてのカレント・イベント、アクティブ・イベント、ガード、変数、始動されたアクション、シグナルを表示します。

時間ベースの動作テストのために、シミュレータにはスケール調整可能なタイマーがついています。リアルタイムの1秒をシミュレータで10秒にするといった設定ができるため、動作を観察することが可能です。

プロトタイプ作成

visualSTATE Prototyperを使用することにより、カスタマイズされたシミュレーション環境を設計することができます。製品のグラフィック・モデルを設計し、その後の動作を相互的にテストすることも可能です。シミュレータ、ログファイル機能へのアクセスもできます。

Real-Link

ターゲット・システム上でアプリケーションをテストするために提供されているのがReal-Linkで、RS-232、CAN、Ethernet等、標準的なコミュニケーション・インタフェースを使用してターゲットと通信することが可能です。このリンクを通してシステムの動作を観察、ログすることができるだけでなく、ホスト・コンピュータからテスト・ベクトルを実行することも可能です。

ログ

シミュレータを使用したすべてのテストは単一のログファイルに記録することができます。同一のログファイル中にいくつかの別個のシーケンスを定義することも可能です。ログファイルは文書化のために保存したり、繰り返しテストのためのスクリプト・ファイルとして使用することもできます。

回歸テスト

設計変更後、以前行ったすべてのテストを繰り返し、自動的に結果を比較することが可能です。これは以前のテストのログファイルを使って行われます。まったく同一のイベント・シーケンスはその後評価されます。visualSTATE Validatorが以前のテストと結果を比較し、違いを報告します。



静的解析

静的解析ツールを使って、システムの動作を調べることができ、イベントまたはアクション・サイドからシステムを解析することが可能です。例えば、「ボタンAが押されたらどうなるか」と質問すると、解析ツールが状態遷移、特定のイベントによって引き起こされたアクションを報告します。同じツールはまた、特定のアクションを割り出すためにも使用できます。例えば、「何が表示をクリアするか」という質問に対し、visualSTATEは特定のアクションの原因となるすべての状態遷移を即座に報告します。

動的解析

動的解析ツールを使用して、アプリケーションの正確なテスト対象範囲を計算し、アプリケーション・プロフィールを作成することが可能です。この解析はシミュレーション、プロトタイプ作成、ターゲット実行のログファイルをもとに行われます。動的解析ツールを起動させることによって、visualSTATEは特定のイベント、アクション、シグナル、変数、状態遷移が何度生じたか計算し、アプリケーション中の実行されなかった部分を識別します。正確なテスト対象範囲も計算します。

ダイナミック・フォーマル検証

visualSTATEはアプリケーションのランタイム・モデルを検証するため特許技術を採用しています。この技術は複数の同時進行中ステートマシンに対応し、すべての組み合わせのテストを実行します。したがって、visualSTATEはアプリケーションが完全に論理的に一貫性があるか、ランタイム実装と設計に矛盾がないかを検証可能です。

実装

visualSTATE Verificatorは明示的に表現された動作のみが存在すること、すべての定義された状態遷移が行われることをチェックします。それはまたネガティブ・チェックも実行し、設計、実装の暗黙的結果として引き起こされる遷移の可能性もチェックします。

到達できない遷移

遷移が保護されている場合は、visualSTATE Verificatorはシステムのステート組み合わせの少なくとも一つに届くことをチェックします。これにより、アプリケーション中の動作不可能な箇所を識別することが可能です。

デッドエンド

visualSTATE Verificatorはシステム中を検索し、デッドエンドを発見します。ある状態に入り、そこから出口がない場合はデッドエンドとして報告されます。デッドエンドは局部レベルとシステムレベルの両方で報告されます。したがって、それぞれは適切に動作していても、同時に実行されると問題を起こす複数のステートマシンを認識することも可能です。

対立する動作

あるイベントによっていくつもの遷移が引き起こされた場合は、それらが互いに対立している可能性があります。visualSTATE Verificatorはそのような対立がないことをチェックします。

プロトタイプ作成

visualSTATEのプロトタイピング・コンセプトはエンジニアリング、マーケティング、カスタマー間のコミュニケーションを促進します。

開発プロジェクトのきわめて早い段階であっても、様々な人に新製品のプロトタイプを試し、貴重なフィードバックを提供してもらうことが可能です。これにより、土壇場の変更という事態が大幅に削減します。

設計作業は一度ですみます。プロトタイプで使用される設計モデルは最終製品で使用されるものと全く同一です。visualSTATE独自のテクノロジーがプロトタイプと最終製品の同一動作を保証します。

仕様書の検証

visualSTATEを使用することによって、仕様書作成のプロセスが反復的かつ対話式になります。作業はまず製品のアウトラインから始まり、その後、visualSTATE Prototyperを使用して迅速に仮想的プロトタイプを作成、それを仕様書に照らし合わせて検証することができます。

仕様書には詳細事項が持続的に付加され、必要とあれば変更されます。これらは設計に実装され、即時に新しい仕様と照らし合わせて評価、検証されます。

このような作業方法を採用することにより、従来の作業方法よりも早い段階で仕様要求を最終決定することが可能になります。この段階でvisualSTATEでなされたすべての作業は、プロジェクトの残り段階で、変更なしに再使用することができます。



ユーザー動作分析

消費者市場向け製品では特に、消費者が実際にどのように製品を使用するかというフィードバック収集が欠かせません。visualSTATEを使用することによって、ユーザー動作を記録するプロトタイプを作ることができます。こうして集められた情報はvisualSTATE Testerで動的解析することが可能です。

高度なシミュレーション

複雑なシステムを開発する場合、カスタマイズされたシミュレーション環境を生成する必要性が生じることはよくあります。visualSTATEを使用することで、一台のパソコン上での分散型システム、またはコミュニケーション・プロトコルをシミュレートすることが可能になりました。カスタマイズされたテストスクリプト実行のためのプラットフォームを創出できる可能性もあります。

visualSTATE Prototyperを使用して、特定のニーズを満たす環境を作り出すこともできます。標準的なWindows開発ツールのシミュレータGUIを定義すると、visualSTATE PrototyperはvisualSTATE Designer、visualSTATE Validatorとグラフィック的に相互作用します。

自動コード生成

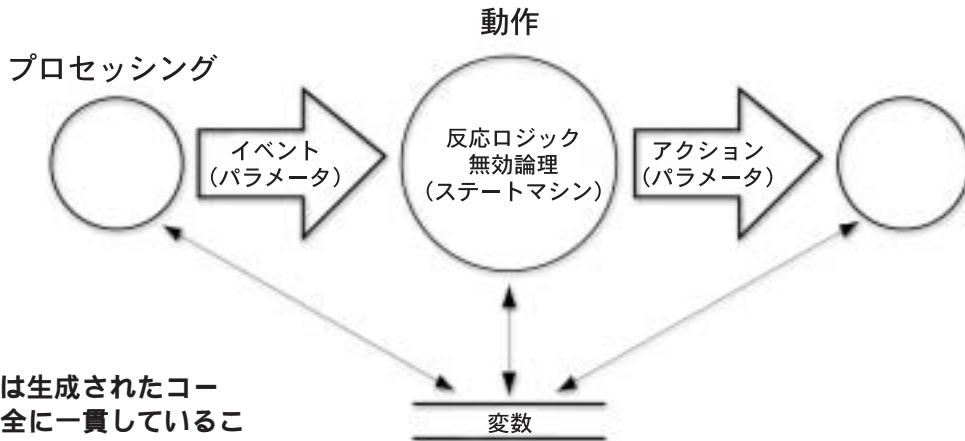
IAR visualSTATEはターゲット・システム用のコードを自動的に生成します。設計と完璧に一貫したプログラム・コードが生成されます。手作業で行うコーディングで生じるエラーがないわけですから、エラーそのものがあり得ません。

生成されるのはANSI-Cソースコードできわめて効率的です。手作業で作成されたアプリケーションに比べ、コード、データスペースが特に少なくすみずみます。

フレキシブルな統合

visualSTATEが生成したコードは、リアルタイム・オペレーティングシステムの有無に関わらず、すべてのマイクロプロセッサ上で実行可能です。

ターゲットの実装は「アプリケーションに固有でターゲットから独立している」アプリケーション部分と、「ターゲットに固有で、アプリケーションの動作から独立している」ファームウェア部分という二つの部分から成り立っています。



visualSTATEは生成されたコードが設計と完全に一貫していることをチェックします。コード生成のためにvisualSTATE Generatorを起動する度に、コードは現行の設計に基づいて一から生成され直されます。旧バージョンが残ることはありません。

visualSTATEは不確定要素のないコードの実行を保証するテクノロジーを採用しています。実行時間、シーケンスともに確定的です。

visualSTATE実装モデルで同時進行のプロセスを実行中に競合状態が生じる可能性はまったくありません。これは、製品が常に一定の振る舞いをするを意味します。

デバイス・ドライバとアプリケーションのほかの部分との相互作用においても、visualSTATEはよく定義されたAPIを搭載しています。これにより、きわめて構造的で移植性のあるアプリケーションが作成できるようになります。

visualSTATEはアプリケーションの動的動作に必要なコードを完全に生成します。しかし、ターゲット固有のデバイス・ドライバはご自身で書かなければなりません。こうすることにより、visualSTATEとハードウェアやアプリケーションのほかの部分のやり取りを完全にコントロールすることができます。

デバイス・ドライバがそれぞれ独立しているため、visualSTATEで作成されたアプリケーションは移植性に優れています。同一のアプリケーションを、設計し直すことなく、複数のハードウェア・プラットフォームに実装することも可能です。

visualSTATEアプリケーションの実装の基本的な方法が、アプリケーション・ロジックを一つのタスクとして動作させるものです。ファームウェア層、アプリケーションのその他の部分は同一、あるいは別のタスクとして動作させることができます。選択はご自由です。

より高度な実装方法が、生成されたvisualSTATEアプリケーションを複数のタスクで動作させてみる方法です。こうすることにより、各サブシステムを特定タスクへと割り当てることが可能になります。

visualSTATEのマルチタスクAPIには複数のモデルをロード可能なエントリーがあり、さまざまな部分で共通するコードのリエントラントな実行を可能にします。

ネイティブ・コードCコンパイラを使用することで、生成されたコードとファームウェアをコンパイルし、リンクし、アドレスを決定します。この結果生成されるのが、ターゲット・プラットフォーム向けの完全に実行可能なアプリケーションです。

文書の作成

最適なフォーマットで文書を作成することができます。visualSTATEアプリケーションからテキスト、MS-Word、RTF、HTMLの文書を作成することが可能です。

visualSTATEを使用することにより、文書の大半は自動的に作成されます。確実に文書を作成できるだけでなく、文書は常に更新された最新版です。

なぜ組み込み開発にIARを使うのでしょうか？

異なるアーキテクチャ、一つのソリューション IAR Systems、ソリューションのプロバイダ

過去20年間で、IAR Systemsは世界市場におけるマイクロプロセッサ用C/C++コンパイラ・ツールの主力サプライヤーとしての地位を確立しました。「異なるアーキテクチャ、一つのソリューション」をモットーに、私たちは8、16、32ビットの各セグメントで、約30種のプロセッサ/コントローラ・アーキテクチャ用プログラミング・ツールを開発してきました。一つのアーキテクチャから別のアーキテクチャへと更新する際に、なるべく簡単にアプリケーション・コードを再使用していただけるようお手伝いしたいと当社は考えます。IAR Systemsのツールを使用することにより、カスタマーは慣れた開発環境で快適に、より効率的に作業を進めることが可能です。チップ別のサポート体制も整っています。つまり、お客様は山積みの問題、遅延に直面することなく、異なるチップ・アーキテクチャにコードを簡単に移せるようになったのです。IARは8、16、32ビットの各セグメントでこのソリューションを提供する組み込みシステム業界唯一の会社です。

お客様すべてのニーズにお応えするため、私たちの事業の形態も変化しています。IARはコード効率の良い開発ツールを提供しているだけではありません。特定のニーズに対応したカスタムメイドのツール開発、設計に関する専門アドバイスも提供しています。大型プロジェクトの場合は、標準的あるいはカスタムメイドのツールを供給するだけでなく、戦略的パートナーとして貴社製品の開発をお手伝いすることも可能です。組み込みシステム・ソフトウェアの開発をさらに向上させ、さらにシンプルにする。そんなIAR Systemsの使命達成に、IAR visualSTATEテクノロジーは大きな役割を果たしています。



DIFFERENT ARCHITECTURES.
ONE SOLUTION.

www.iar.com

版權 ©2000 IAR Systems

販売元・お問い合わせ先
株式会社プロトン
〒169-0073 東京都新宿区
百人町1-22-17
電話：03-5337-6430
Fax：03-5337-6130
E-mail: es@sb.proton.co.jp

スウェーデン
IAR Systems AB
PO Box 23051
Islandsgatan 2
SE-750 23 Uppsala
電話：+46 18 16 78 00
ファックス：+46 18 16 78 38
電子メール：info@iar.se

米国
IAR Systems
(US HQ-West Coast)
One Maritime Plaza
San Francisco, CA 94111
電話：+1 415 765 5500
ファックス：+1 415 765 5503
電子メール：info@iar.com

ドイツ
IAR Systems AG
Posthalterring 5
D-85599 Parsdorf
電話：+49 89 90 06 90 80
ファックス：+49 89 90 06 90 81
電子メール：info@iar.de

英国
IAR Systems Ltd.
9 Spice Court, Ivory Sq.
Plantation Wharf, York Rd.
London SW11 3UE
電話：+44 20 7924 3334
ファックス：+44 20 7924 5341
電子メール：info@iarsys.co.uk

デンマーク
IAR Systems A/S
Elkjærvej 30-32
8230 Åbyhøj
電話：+45-8625-1111
ファックス：+45-8625-1191
電子メール：info@iar.dk