(19)日本国特許庁(JP)　(12)公表特許公報(A)　(11)特許出願公表番号
特表2011-504621
(P2011-504621A)
(43)公表日　平成23年2月10日(2011.2.10)

(51)Int.Cl.		F I			テーマコード（参考）
G06F 11/00	(2006.01)	G06F	9/06	630J	5B376

審査請求 未請求　予備審査請求 未請求　（全 13 頁）

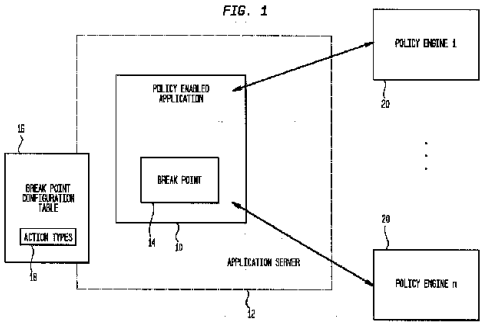
(21)出願番号　特願2010-532317 (P2010-532317)
(86)(22)出願日　平成20年11月3日 (2008.11.3)
(85)翻訳文提出日　平成22年6月30日 (2010.6.30)
(86)国際出願番号　PCT/US2008/082240
(87)国際公開番号　WO2009/059291
(87)国際公開日　平成21年5月7日 (2009.5.7)
(31)優先権主張番号　11/982,675
(32)優先日　平成19年11月2日 (2007.11.2)
(33)優先権主張国　米国 (US)

(71)出願人　509229887
テルコーディア　ライセンシング　カンパニー，　リミテッド　ライアビリティ　カンパニー
ＴＥＬＣＯＲＤＩＡ　ＬＩＣＥＮＳＩＮＧ　ＣＯＭＰＡＮＹ，　ＬＬＣ
アメリカ合衆国　０８８５４　ニュージャージー州　ピスカタウェイ　ワン　テルコーディア　ドライブ　（番地なし）　５ジー１１６
(74)代理人　100077481
弁理士　谷　義一
(74)代理人　100088915
弁理士　阿部　和夫

最終頁に続く

(54)【発明の名称】ポリシーが与えられたプログラミングのための方法及びシステム

(57)【要約】
アプリケーションコードへの修正なしに通信システムにおけるアプリケーションの振る舞いの外部の実行時間の適応を可能にするシステム及び方法は、少なくとも１つのブレイクポイントを有し、アプリケーションサーバ上にあるアプリケーションと、少なくとも１つの識別可能な判定エンジンと、ブレイクポイントの１においてアプリケーションがブレイクポイントのリストにアクセスするように、各ブレイクポイントについて判定エンジンの少なくとも１つの識別子を有し、ブレイクポイントに対応する判定エンジンのリストされたインスタンスを呼び出し、判定エンジンに基づいてアプリケーション振る舞いを適応するブレイクポイントのリストとを有するアプリケーションを備え、さらに、いくつかのブレイクポイントについての２つのエントリが等しいシーケンス番号を有する場合にこれらのエントリにおいて特定された判定エンジンを平行で呼び出すことができるように、ブレイクポイントのリストにおける各エントリがシーケンス番号を有すことができる。



【特許請求の範囲】
【請求項１】
通信システムにおけるアプリケーションのアプリケーション振る舞いの外部の実行時間の適応を可能にするシステムであって、
アプリケーションサーバ上にあり、少なくとも１つのブレイクポイントを有するアプリケーションと、
少なくとも１つの識別可能な判定エンジンと、
各ブレイクポイントについての前記判定エンジンの少なくとも１つの識別子を備えるブレイクポイント構成のリストとを備え、
前記ブレイクポイントの１において、前記アプリケーションはブレイクポイント構成の前記リストにアクセスし、前記１つのブレイクポイントについての前記判定エンジンの前記リストされたインスタンスを呼び出し、前記少なくとも１つの判定エンジンに基づいてアプリケーションの振る舞いを適応させることを特徴とするシステム。
【請求項２】
ブレイクポイント構成の前記リストは、前記アプリケーションの外部にあることを特徴とする請求項１に記載のシステム。
【請求項３】
ブレイクポイント構成の前記リストは、各識別子についてマッピングする属性及び各識別子についての動作タイプをさらに備えることを特徴とする請求項１に記載のシステム。
【請求項４】
ブレイクポイント構成の前記リストは、前記リストにおける各エントリに関連するシーケンス番号をさらに備えることを特徴とする請求項１に記載のシステム。
【請求項５】
ブレイクポイント構成の前記リストは、前記ブレイクポイントを前記判定エンジンに動的にマッピングすることを特徴とする請求項１に記載のシステム。
【請求項６】
前記１つのブレイクポイントの各エントリについての前記シーケンス番号が等しい場合、呼び出しが平行でなされることを特徴とする請求項１に記載のシステム。
【請求項７】
通信システムにおけるアプリケーションの振る舞いの外部の実行時間の適応を可能にする方法であって、
アプリケーションにおける少なくとも１つのブレイクポイントのそれぞれにおいてブレイクポイント構成のリストにアクセスするステップであって、前記テーブルが前記少なくとも１つのブレイクポイントのそれぞれについての判定エンジンに対する少なくとも１つの識別子を有する、ステップと、
前記ブレイクポイントに対応する前記判定エンジンのすべてを呼び出すステップと、
前記判定エンジンに基づいて前記アプリケーションの振る舞いを適応させるステップとを備えたことを特徴とする方法。
【請求項８】
ブレイクポイント構成の前記リストは、前記アプリケーションの外部にあることを特徴とする請求項７に記載の方法。
【請求項９】
ブレイクポイント構成の前記リストは、各識別子についてマッピングする属性及び各識別子についての動作タイプをさらに備えたことを特徴とする請求項７に記載の方法。
【請求項１０】
ブレイクポイント構成の前記リストは、前記リストにおける各エントリに関連するシーケンス番号をさらに備えたことを特徴とする請求項７に記載の方法。
【請求項１１】
ブレイクポイント構成の前記リストは、前記ブレイクポイントを前記判定エンジンに動的にマッピングすることを特徴とする請求項７に記載の方法。

【請求項１２】
前記１つのブレイクポイントの各エントリについての前記シーケンス番号が等しい場合、呼び出しが平行でなされることを特徴とする請求項７に記載の方法。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、概して、通信、情報及びエンターテイメントサービスのプログラミングに関する。
【背景技術】
【０００２】
従来方式で書かれたソフトウェアアプリケーションはアプリケーションコード内に定義されたすべての機能を有する。このようなアプリケーションが一度展開されると、アプリケーションを提案する提供者は、アプリケーションの振る舞いに影響する余地をほとんど有さない。このアプリケーションは調整できる２，３のパラメータを有する可能性があるが、一般的にアプリケーションの振る舞いは、アプリケーションコードが修正される場合を除いて固定である。
【０００３】
動的な市場に取り組むことを望むサービス提供者は、迅速に新しいサービスを提案するために、柔軟性を必要とする。アプリケーションコードを修正することに関連するサイクル時間は、受け入れ難いほど処理を遅くする可能性がある。したがって、ソフトウェアリリースを待つことなく振る舞いを修正できるアプリケーションコードの必要性がある。
【０００４】
電話網サービスを構築するためのインテリジェントネットワーク（Intelligent Network）及び高度インテリジェントネットワーク（Advanced Intelligent Network）概念の開発において同様の動機を伴った。電話交換機内のコードを変えることにより新しいサービスを形成することは可能であったが、これはむしろ遅く、扱いにくかった。この選ばれたアプローチは、２者間の接続の確立において電話交換機ソフトウェアの状態を指定する呼モデルを正確に定義することになった。呼モデルにおける様々なポイントにおいて、電話交換機を、どのように進めるかの指示のための外部のサービス制御ポイントを要求するように構成することができる。交換機システム自体内でソフトウェアに変更を行うことなく、サービス制御ポイントにおけるロジックの振る舞いを変えることにより、新しいサービス機能を作り出すことができる。
【０００５】
このアプローチは、交換機システムの状態を指定する共通の呼モデル及び期待された振る舞いに高く依存する。呼の確立の過程は、あるシステムから別のシステムまで一貫しているため、このような呼モデルを定義しうる。呼モデルの定義及び合意はかなりの時間量をとり、電話呼の処理は時間が経つと比較的に静的であったため、この時間は耐えられた。提供者が原則頻繁に新しく多様なサービスを提案する環境において、サービスロジックの内部の状態についてこのような詳細化されたモデルを定義することは不可能となるだろう。この一般的な呼モデル技法が望ましい特性を有する一方で、新興で新しく多様なサービスの処理に直接適用できない。
【０００６】
電話通信装置ベンダ及びオペレータは、呼処理システムへの特徴の追加において、似たような問題に取り組んだ。高度インテリジェントネットワーク（ＡＩＮ）に基づくソリューションの上での合意は、電話呼の確立の処理に基づいて別の共通の呼モデルを定義する。この共通の呼モデルは、判定のために外部システムにクエリできるイベントのあらかじめ定義されたセットを提供する。例えば、特許文献１「Programmable call processing system and method」（プログラム可能な呼処理システム及び方法）において、Ｂｕｎｃｈなどは、ＡＩＮに結合された分散電話通信交換システム上でこのアプローチを説明している。呼を処理するサービス交換ポイント（交換機）及びサービスロジックを処理するサー

ビス制御ポイントのこの分離は、交換機内のソフトウェアに変更を行うことなく新しい呼サービスを定義することを可能にする。このアプローチは回路交換ネットワークにおいて電話呼を制御するのに良く、Parlay Group仕様書及びＪＣＰ（Java Community Process）のＪＡＩＮ(Java Advanced Integrated Network)仕様書などの工業フォーラムの結果を通して次世代パケット交換ネットワークへ拡張された。特定の機能（呼制御）を扱い、機能への高度な合意及び／又は標準化を扱うという点で、このような標準化に必要性があるという事実は、このアプローチの限界を示す。したがってこのアプローチは、時間が経って静的であるアプリケーションの機能に適している。

【０００７】

特許文献２「Device and method for swapping out a part of a service logic program」（サービスロジックプログラムの一部をスワップアウトするデバイス及びシステム）において、Ｍｏｒｉｔｚは、複数の実体にまたがる分散サービスロジックのためのメカニズムを説明する。サービスロジックの一部をクライアント端末へ分散させることにより、クライアント端末として動作するインテリジェント端末デバイスの能力を増加するのに活用することに、Ｍｏｒｉｔｚは特に焦点を合わせる。Ｍｏｒｉｔｚは、この分散アプローチを課金関連情報の判定について使用するための方法を開示する。クライアントデバイスへの分散はパーソナライズ化を可能にできるが、変更をすべてのクライアント端末デバイスに伝播しなければならないであろうから、全体サービスの振る舞いの修正を間単に可能にしない。通信メカニズム（Ｍｏｒｉｔｚの「課金チケット（charge ticket）」）が前もって定義されコード化されているので、課金などの単一のトピックについての分散を可能にすることは単純化された。したがってＭｏｒｉｔｚは柔軟性のあるサービスロジックプログラムの必要性に取り組むが、とても限定された領域においてである。

【０００８】

特許文献３「Architecture for the rapid creation of telephony services in a next generation network」（次世代ネットワークにおける電話サービスの迅速な生成のためのアーキテクチャ）において、Ａｎｊｕｍなどは、「hides the detail of the underlying call-state management protocols and hardware from applications」（下層の呼状態管理プロトコル及びハードウェアの詳細をアプリケーションから隠す）オブジェクト指向の呼モデルを説明する。特に、Ａｎｊｕｍなどは、回路交換及びパケット交換電話ネットワークの両方における呼制御を表すのに十分な摘要となるように設計された新しい呼モデルを説明する。このような呼モデルは、ＪＴＡＰＩ（Java Telephony Application Programming Interface）モデルよりも柔軟性があり、ＪＴＡＰＩモデルに基づいたが単一の機能をまだ表す。したがってこのモデルは、振る舞いが同様に定義されず又は前もって明確に知られないより一般的な場合に適用可能である。

【０００９】

ドラフト要件文書「Policy Evaluation, Enforcement and Management Architecture」（ポリシーの評価、実行及び管理アーキテクチャ）（OMA-AD- Policy_Evaluation_Enforcement_ Management-Vl_0-200600625-D）において指定されているように、ＯＭＡ（Open Mobile Alliance）内のＰＥＥＭ（Policy Evaluation Enforcement Management）の取り組みは、ＯＭＡのサービスイネーブラのサポートとしてポリシー評価及び実行のためのアーキテクチャを説明する。これらのイネーブラ（enabler）は、グループリスト管理、メッセージング及び位置などの機能を含む。これらのイネーブラがポリシー判定をクエリするための共通フレームワークを提供するように、このアーキテクチャを設計する。このような能力は、システム間のマッピングインターフェースの処理を単純にするであろうことから、ここで説明されるポリシーが与えられたプログラミング（Policy-enabled Programming）の構成ブロックとして必要とはされていないが、役に立つであろう。しかしＯＭＡワークは、ポリシーエンジン（ＰＥＥＭイネーブラ）をいつクエリするか又はどのポリシーエンジンがクエリするかを、どのようにイネーブラが判定することになるかは指定しない。

【００１０】

別のアプローチがワークフローシステムによりサポートされる。複合的な命令処理にしばしば適用されるこれらのシステムにおいて、処理ステップのセットは、ソフトウェアコードの方式に従うよりむしろ、実行時間で解釈されるテキストの様式で定義される。これは、ワークフローエンジンのコードを変えることなくワークフローを修正できる点で柔軟性を提供する。しかし解釈の操作は非効率的であり、命令処理に適切であるが実際のサービスの実行に一般的に不十分である性能に導く。

【００１１】

データベースシステムは、トリガする別のメカニズムをまだ提供し、ロジックを実行するために格納された手続きを呼び出すことができ、大いに柔軟性を提供する。しかし格納された手続きを実行できる操作は、データベースにおけるデータ上で挿入、削除、更新及び選択のデータベースの操作のためだけに存在する。このアプローチでの問題はデータベース操作についての制限であり、この仕様全体がデータベースシステム内にあり、データ及び関連するトリガ及び格納された手続きを介して誘導しなければ、簡単に参照可能又は修正可能ではない。

【００１２】

したがって、現在の処理はソフトウェアアプリケーションの特定のタイプに制限され、これはたとえば呼処理であり、これらが使用されうる前にアプリケーション処理モデルの高度な合意及び又は標準化を必要とした。他の現在の処理は、データベース又はワークフローシステムなどのミドルウェア内にのみ存在し、それらの範囲を限定し、サービス実行のための性能要件を満たさない。

【００１３】

したがって、ポリシーエンジンなどの外部の判定ポイントをいつ及びどこでクエリするべきかを動的に判定することができるシステムの必要性がある。制御された選択肢のセットに構成されうるブレイクポイントにより特徴付けられた、このような動的なクエリを構成するための構造化された効率的なメカニズムもまた必要性がある。

【先行技術文献】

【特許文献】

【００１４】

【特許文献１】米国特許第５，９４０，４８７号明細書
【特許文献２】米国特許第６，９７０，９０１号明細書
【特許文献３】米国特許第６，９６７，９５７号明細書

【発明の概要】

【００１５】

本発明は、アプリケーションコードに変更を行うことなく実行の間に、アプリケーションの振る舞いを適応又は構成することを可能にするアプリケーションソフトウェアを形成及び展開するための設計アーキテクチャ及び方法論を有利に提供する。アプリケーションの構成又は適応により異なる目的のために単一のソフトウェアアプリケーションの複数回の再利用が提供される。さらに本発明のシステム及び方法は、ソフトウェア開発サイクルに関連して遅延することなく遂行できるので、新しいサービス振る舞いの迅速な開発を可能とする。

【００１６】

通信システムにおけるアプリケーションの振る舞いの外部の実行時間の適応を可能にするシステム及び方法は、少なくとも１つのブレイクポイントを有し、アプリケーションサーバ上にあるアプリケーションと、少なくとも１つの識別可能な判定エンジンと、各ブレイクポイントについて判定エンジンの少なくとも１つの識別子を有するブレイクポイントのリストとを備え、ブレイクポイントの１つにおいて、アプリケーションは、ブレイクポイント構成のリストにアクセスし、ブレイクポイントに対応する判定エンジンのリストされたインスタンスを呼び出し、判定エンジンに基づいてアプリケーションの振る舞いを適応させる。

【００１７】

ブレイクポイントのリストはアプリケーションの外部のファイルであることができ、各識別子についての属性マッピング及び各識別子についての動作タイプを含むことができ、加えてブレイクポイントを判定エンジンに動的にマッピングできる。

【００１８】

本発明は、本発明の限定ではない例示の実施形態のために記された図面を参照することにより、下記の発明を実施するための形態においてさらに説明され、図面全体で似た参照番号は同様の部分を表す。しかし本発明は示された正確な編成及び道具に限定されないことを理解されたい。

【図面の簡単な説明】

【００１９】

【図１】ポリシーが与えられたプログラムの実行のためのアーキテクチャを示す図である。

【図２】ブレイクポイント構成の例示的なテーブルを示す図である。

【図３】ブレイクポイント構成テーブルにおける処理エンティティのアプリケーションの振る舞いを示すフローチャートの図である。

【発明を実施するための形態】

【００２０】

アプリケーションの振る舞いの外部の実行時間の適応を可能にするシステム及び方法が示されている。このアプローチの主要な特徴は、コンパイルされたコードに関連する性能に加えて、アプリケーションフロー内の判定ポイント及び可能性のある動作を定義する柔軟性をアプリケーション開発者に提供し、外部の評価エンジンにこれらの判定ポイントの動的なマッピングを可能にし、アプリケーションが開発されるにつれて又はアプリケーションの設計及び開発の間だけでなくアプリケーションを実行している間にさえ、アプリケーションの振る舞いを構成することを可能にすることである。

【００２１】

このようなシステム及び方法をサポートする２つの具体的なメカニズムがある。第一のメカニズムは、ブレイクポイント構成テーブルなどの外部テーブルである。アプリケーションは、実行中に定義されたブレイクポイントに到達するとき、このテーブルを読み込み、どの動作がとられるかを判定する。このテーブルにおけるエンティティを、アプリケーションから分離して定義でき、所与のブレイクポイントで何の属性又は変数をアプリケーションが定義したかの情報に基づいて、アプリケーションが実行している間に変更さえする。第２の適応方法は、外部システムに含まれる。これらのシステム内のロジック、ポリシー及び／又はルールは、続くフロー及びアプリケーションの振る舞いに影響することになる応答を作成することになる。これら２つのメカニズムでアプリケーションの振る舞いは、アプリケーションコードを再書き込み又は修正する必要なしに大きく修正できる。

【００２２】

図１は、本発明のシステムの一実施形態を示す。ポリシーが与えられたアプリケーション１０は、通信システム（図示されていない）内のアプリケーションサーバ１２内にある。ポリシーが与えられたアプリケーション１０は、アプリケーションロジックのフロー内の構成可能なブレイクポイント１４の指定されたセットで構築されている。アプリケーションの実行がこれらのブレイクポイント１４の１つに到達するとき、アプリケーションコードは、ブレイクポイントの構成のリスト又は動作タイプ１８を含むデータを包含する、ブレイクポイント構成テーブルなどの外部テーブル１６に目を向ける。このデータを評価することにより、アプリケーション１０はブレイクポイント１４でとるべき動作を判定できる。テーブルの複雑性を管理するために、可能性のある動作タイプの数を限定できる。一般的な動作は、アプリケーションが外部システム２０にデータを送信し、アプリケーションの次の動作を判定するために結果の応答を使用することであろう。そのコンパイルされたコード内で定義されるように、関連した性能で、アプリケーション１０の一部として、これらの続く動作は実行される。外部システム２０は、判定エンジン、ポリシーエンジン又はアプリケーション１０からの入力を受け取り、１又複数のポリシーの定義されたセ

ットを適用し、応答を提供するであろうポリシー判定ポイントでありうる。アプリケーションのコードに変更を行うことに代えて、１又は複数のポリシーを修正することにより、アプリケーションの振る舞いを変える。代替の実施形態において、ジャンプテーブル（図示されていない）を、アプリケーション１０内に実装できる。このような場合に、外部の判定ポイントの処理を修正することにより、又はアプリケーション１０を修正及び再コンパイルすることにより、アプリケーションの振る舞いを変更できるであろう。

【００２３】

外部システム２０は、ワークフローシステム、データベースシステム又は別のソフトウェアアプリケーションでさえもあり得ることに留意されたい。判定ポイントがポリシーが与えられたアプリケーション１０においてハードコードされているとしても、マッピングを、異なるソフトウェアアプリケーションを指し示すように変更することができる事実は、柔軟性を提供する。

【００２４】

ポリシーが与えられたアプリケーション１０が外部のポリシーエンジン２０から受け取る応答に対処できる多くの方法がある。その続く振る舞いの判定において外部システム２０に完全な柔軟性を与えるアプリケーション１０を形成することは極めて難しく、エラーとなる傾向があるであろう。しかし動作タイプ又は選択肢１８の範囲上の適切な限定にも関わらず、複雑性は管理可能なレベルのままでありえた。具体的には、外部システム２０により提供された応答に基づいて動作タイプ１８の限定された数をサポートするようにアプリケーション１０を設計できた。動作タイプ１８は分岐又は二択の判定ポイント動作、ｎｕｌｌ動作及び変数置き換え動作を含むことができた。

【００２５】

二択の判定動作１８の場合において、アプリケーション１０は外部システム２０にメッセージを送り、外部システムは２値、つまり「はい／いいえ」又は「真／偽」応答を提供する。アプリケーション１０はその後、応答が「はい」である場合に１つの規定のパスに進み、応答が「いいえ」である場合に別のパスに進む。アプリケーション１０がこの判定動作１８において２つの選択肢のうち１つのみを実行する一方で、全体の振る舞いは、判定をするのに外部システム２０によって使用されるポリシー又は処理に基づいて広範囲に変わることができる。これは、複数のパスが可能である複数方式の判定動作に拡張することができる。

【００２６】

例として、デジタルコンテンツアイテムをユーザに配信又は送信するよう設計されたアプリケーション１０は、コンテンツを送信するちょうど前にブレイクポイント１４を含みうる。このブレイクポイント１４の応答において、アプリケーション１０は、クエリ内のコンテンツアイテムのユーザ及びリストの両方を特定する情報を外部ポリシーエンジン２０に供給する。アプリケーション１０は、応答が「はい」の場合にコンテンツの送信を進め、応答が「いいえ」の場合に応答をキャンセルするようにプログラムされている。ある状況において、外部システム２０は、ユーザがアイテムに支払いするのに十分な残高を有するかどうかを判定するリアルタイム課金エンジンでありうる。ユーザの残高が十分の場合、外部システム２０はユーザの口座から引き落として「はい」を返す。残高が不十分の場合、外部システム２０は「いいえ」を返す。別の状況において、外部システム２０は、例えば会社のセキュリティ基準として又は訪問者を引き付けるレストランの手段としてある物理的な場所内でユーザにコンテンツをダウンロードさせることのみを可能にする認証システムかもしれない。ユーザがコンテンツアイテムについて指定された場所の境界内である場合、外部システム２０は「はい」を返し、そうでなければシステム２０は「いいえ」を返す。したがって、個別の外部システム２０を採用する２つの異なるサービスは、同じコンテンツ配信ポリシーが与えられたアプリケーション１０を利用してきた。

【００２７】

ｎｕｌｌ動作１８の場合において、アプリケーション１０は単にブレイクポイント１４を実施し、実行を続ける。Ｎｕｌｌ動作はアプリケーションが外部システム２０にいくつ

かのイベントを知らせる必要が単にある場合に適している。上述のコンテンツダウンロードサービスの例において、コンテンツを送信するちょうど前にｎｕｌｌ動作１８を含むブレイクポイント１４を実行するようにアプリケーション１０に指示しうる。後払い口座（post paid accounts）を持つユーザが次の請求サイクルの間コンテンツに課金しうるように、ブレイクポイント１４の実行は、外部システム２０がコンテンツアイテムのリストを含むダウンロードされたイベントを記録することを引き起こしうる。

【００２８】

変数置き換え動作１８において、アプリケーション１０は外部システム２０からの応答を使用して、アプリケーション１０により処理されている変数の値を変える。上のコンテンツダウンロードサービスの例において、コンテンツの送信のちょうど前に変数置き換え動作１８を含むブレイクポイント１４を実行するようアプリケーション１０に指示しうる。ブレイクポイント１４を実行することは、コンテンツアイテムのリストをフィルタすることを外部システム２０に引き起こす可能性があり、おそらく特定されたユーザの歳又は嗜好に基づいて、それらの格付け情報がユーザにとって許容できないことを示すコンテンツアイテムを取り除く。外部システム２０はその後、アプリケーション１０にフィルタされたリストを返し、アプリケーション１０はコンテンツアイテムの初期のリストをフィルタされたリストで置き換え、ダウンロードを進める。

【００２９】

他の動作タイプ１８は、アプリケーションフローの制御においてより大きな柔軟性を提供するように定義されうる。アプリケーション１０の構築において過度の複雑性を避けるために、動作タイプ又は選択肢１８の数を限定するのが望ましいことに留意されたい。

【００３０】

ポリシーが与えられたアプリケーション１０は、アプリケーションサーバ１２又はサービス配信プラットフォームにおいて実行し、各ブレイクポイント１４に関連するアプリケーションの振る舞いを指定するであろうブレイクポイント構成テーブル１６へのローカルアクセスを有するであろう。図２は、製品／サービスを展開する責任のあるプロビジョニングシステムにより追加投入されうる例示的テーブル１６を示す。テーブル１６は、上述の動作タイプ１８だけでなく続くフィールドも含み、続くフィールドは以下でより詳細に説明されるように、ポリシーエンジン又は外部システム２０のためのアドレス２２、シーケンス番号２４及び属性マッピング２６である。ポリシーが与えられたアプリケーション１０は、１又は複数の外部ポリシー判定ポイントと対話するであろう。構成テーブル１６は、対話するための特定の判定ポイントで又は（送信のための属性など）対話するため方式で対話が実施できるプログラムフロー内でポイントをリンクする。

【００３１】

この議論がポリシー判定ポイントを説明する一方で、外部システム２０を限定するようにここで定義されたメカニズムにおいてポリシー評価するものでありえない。定義された呼び出しインターフェースを有する任意の外部システム２０をポリシー判定ポイントに置き換えて使用しうる。

【００３２】

単純化されたサンプル構成テーブル１６が表１に示されている。この場合に、ブレイクポイント１　１４に関連した２つのブレイクポイントエントリがある。第１のエントリについて、アプリケーション１０は、テーブル１６においてシステムアドレス２２を介してポリシーエンジン１に属性Ａ、Ｂ及びＣを送ることにより、属性マッピング２６を実施し、属性Ｃの値を置き換えるために応答を使用しなければならない。第２のエントリについて、アプリケーション１０は、テーブル１６においてそのアドレス２２を介してポリシーエンジン１に属性２６　Ａ，Ｃ及びＤを送り、「はい／いいえ」応答に基づいて判定動作１８を実施しなければならない。シーケンス列におけるシーケンス番号２４は、単一のブレイクポイント１４に関連した呼び出しが実行される順を示す。シーケンス番号２４を共有する呼び出しは平行で実行されうる。シーケンス番号が使用されない場合において、アプリケーション１０はファイルで見つかった順にクエリする。

【００３３】
【表１】
表１　ブレイクポイント構成テーブル

ﾌﾞﾚｲｸﾎﾟｲﾝﾄ ID	ﾌﾞﾚｲｸﾎﾟｲﾝﾄ ﾀｲﾌﾟ	ｼｰｹﾝｽ	ｼｽﾃﾑｱﾄﾞﾚｽ	属性ﾏｯﾋﾟﾝｸﾞ	応答 置き換え
１	変数置き換え	１	ポリシーエンジン１	Ａ，Ｂ，Ｃ	Ｃ
１	分岐	２	ポリシーエンジン１	Ａ，Ｃ，Ｄ	
２	ｎｕｌｌ	１	ポリシーエンジン２	Ｘ，Ｙ，Ｚ	

【００３４】
図３は、図１で示されるアプリケーション１０の振る舞いを示すフローチャートである。図の左側を参照して、ステップＳ１で初期化の上、アプリケーション１０はステージ１を通してその設計された活動を実施する。ステップＳ２において、アプリケーションはブレイクポイント１　１４に到達する。このポイントにおいてアプリケーション１０は、ブレイクポイント構成テーブル１６を調べて、ブレイクポイント１　１４に関連する任意のエントリがあるかどうかを見る。
【００３５】
図３の右側は、これらのエントリを処理するためにブレイクポイント１モジュール内で使用されるフローを示す。ステップＳ３におけるエントリの上で、アプリケーション１０は、テーブル１６において定義された属性マッピング２６を使用して、ステップＳ４で外部システム２０を呼び出す。外部システム２０が応答するとき、ステップＳ５で、アプリケーション１０は、ブレイクポイントのタイプ又は処理されている動作タイプ１８を判定する。動作ブレイクポイントタイプ１８が変数置き換えである場合、ステップＳ６で、アプリケーション１０は、構成テーブル１６において定義されるように置き換えを実施し、処理されるべきブレイクポイントエントリ１４がさらにあるかをテストするためにステップＳ７に進む。ブレイクポイントタイプ１８がｎｕｌｌである場合、ステップＳ７で、さらなるブレイクポイントエントリ１４を処理する必要がある場合、アプリケーション１０は応答を待つ必要はなく、テストに直接進む。ブレイクポイントタイプ１８が分岐である場合、アプリケーション１０は、ステップＳ８で外部システム２０からの応答をテストする。外部システム２０が「はい」を応答する場合、アプリケーション１０はステップＳ７でさらなるブレイクポイントエントリ１４をテストするために進む。外部システム２０が「いいえ」を応答する場合、アプリケーションはこれ以上ブレイクポイントエントリ１４を処理することなくブレイクし、ステップＳ９でメインのアプリケーションフローに、「いいえ」の終了状態を返す。処理されるべきさらなるブレイクポイントエントリ１４がない場合、ステップＳ１０でアプリケーション１０は、メインのアプリケーションフローに「はい」の終了状態を返す。
【００３６】
外部システム２０との対話における例外及びエラー状態の対処は示されていない。アプリケーション１０は、返されたエラー又は応答無しでのタイムアウトを扱うさまざまな例外の対処の振る舞いでプログラムされうる。代替として、ある対処された例外の場合にとられることになる動作を示すためにテーブルを拡張しうる。アプリケーション１０がこれらの状態を扱う方式は、このメカニズムに重要ではない。
【００３７】
この動的なプログラミングアプローチの主要な態様は、判定ポイントシステム２０及び判定ポイントシステム２０内の特定のポリシーに、アプリケーションブレイクポイント１４を動的にマッピングできることである。マッピングが実行できるであろう多くの方法が

ある。おそらく最も直接的な方法は手動アプローチであり、手動アプローチでは、人の分析者がブレイクポイント構成テーブル１６を作成し、各ブレイクポイント１４に対応するポリシーを定義し、構成テーブル１６をアプリケーションサーバ１２にロードし、判定ポイントにポリシーをロードする。アプリケーションブレイクポイント１４及びポリシーが構造化された様式で可視にできる場合、このマッピング処理における自動化及び妥当性の検証が可能である。

【００３８】

設計環境は、ユーザにより選択されたアプリケーション１０におけるすべてのブレイクポイント１４の構造化された表現を取り込みうる。この環境はそこに知られている判定ポイントからポリシーを取り出しうる。人のユーザが、テーブル２に示されている簡単な表現であるブレイクポイント１４を選択するとき、この環境は有効なこれらのポリシーのみ取り出し表示しうる。例えば、操作を分岐又は二択の動作１８のみをブレイクポイント１４に許可する場合、はい／いいえ又は真／偽を返すこれらのポリシーは適切であろう。ポリシーを選択した後、テーブル３において示されるように、ユーザは、選択されたブレイクポイント１４で、ソフトウェアアプリケーション２０からの利用可能な変数を、ポリシーへのインターフェースにおいて指定された変数にマッピングするであろう。この環境はその後、例えば、文字列、整数などのタイプが一致することを保障するマッピングを有効化しうる。すべてのマッピングがこの方式で成功して定義された後、環境は一致するブレイクポイント構成テーブル１６を作成し、アプリケーションサーバ１２上にインストールしうる。

【００３９】

【表２】

表２　省略されたブレイクポイント表現構造

ブレイクポイント
名前
記述
許可されたタイプ（ｎｕｌｌ、変数置き換え、分岐）
利用可能な変数（名前及びタイプ）
・・・

【００４０】

【表３】

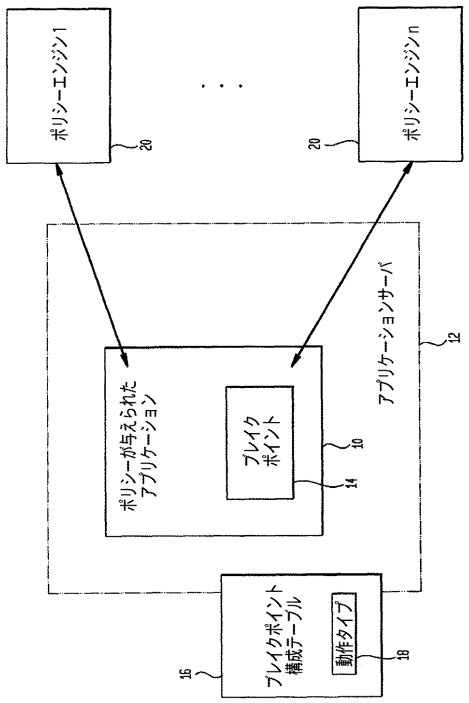
表３　省略されたポリシー呼び出しインターフェース構造

ポリシー
名前
記述
入力［変数１（名前／タイプ）、変数２・・・］
出力［変数１（名前／タイプ）、変数２・・・］
・・・

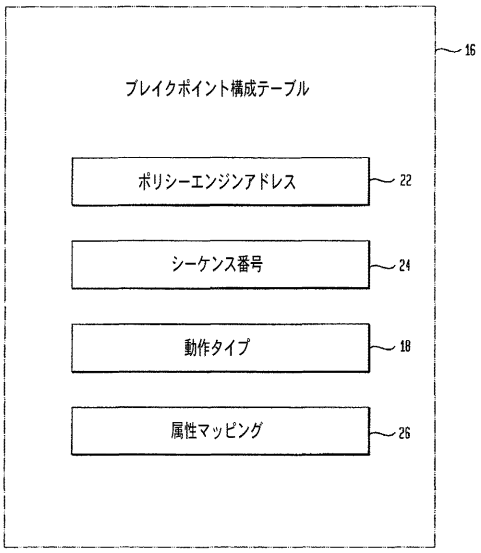
【００４１】

本発明は特定の実施形態において説明されてきたが、本発明がこのような実施形態により限定されて構築すべきではなく、むしろ以下の特許請求の範囲により構築されるべきであることを理解されたい。

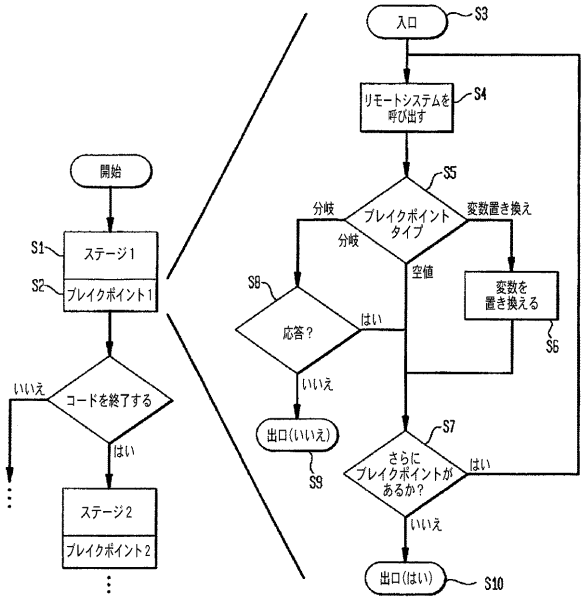
【図１】


ポリシーエンジン1
20
ポリシーエンジンn
20
アプリケーションサーバ
12
ポリシーが与えられた
アプリケーション
ブレイク
ポイント
10
14
ブレイクポイント
構成テーブル
動作タイプ
16
18

【図２】


16
ブレイクポイント構成テーブル
ポリシーエンジンアドレス
22
シーケンス番号
24
動作タイプ
18
属性マッピング
26

【図３】


開始
S1
ステージ1
S2
ブレイクポイント1
いいえ
コードを終了する
はい
ステージ2
ブレイクポイント2
入口
S3
リモートシステムを
呼び出す
S4
S5
分岐
ブレイクポイント
タイプ
変数置き換え
分岐
空値
変数を
置き換える
S6
S8
応答?
はい
いいえ
出口(いいえ)
S9
S7
さらに
ブレイクポイントが
あるか?
はい
いいえ
出口(はい)
S10

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US 08/82240

A. CLASSIFICATION OF SUBJECT MATTER
IPC(8) - G06F 15/177 (2008.04)
USPC - 709/220
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
USPC: 709/220

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
USPC: 709/203, 220, 223, 224, 230; 370/254, 255, 257 keyword limited - see search terms below

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
PubWest (PGPB,USPT,USOC,EPAB,JPAB), Google Scholar, Google Patent
Search terms: telecommunication, telecom, application, programming, dynamic, adaptation, configurable, policy, decision, breakpoint, table, driven, interpretive, list, command, action, state, machine, external, parameter, attribute, primitive, element etc.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5,724,406 A (JUSTER) 03 March 1998 (03.03.1998), col 2, ln 3-43, col 2, ln 62 to col 3, ln 2, col 3, ln 15-32	1-12
A	US 5,317,757 A (MEDICKE et al.) 31 May 1994 (31.05.1994), entire document	1-12
A	US 7,111,053 B1 (BLACK et al.) 19 September 2006 (19.09.2006), entire document	1-12

☐ Further documents are listed in the continuation of Box C. ☐

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
18 December 2008 (18.12.2008)	29 DEC 2008
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-3201	Authorized officer Lee W. Young PCT Helpdesk: 571-272-4300 PCT OSP: 571-272-7774

Form PCT/ISA/210 (second sheet) (April 2007)

フロントページの続き

(81)指定国 AP(BW,GH,GM,KE,LS,MW,MZ,NA,SD,SL,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV,MC,MT,NL,NO,PL,PT,RO,SE,SI,SK,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AO,AT,AU,AZ,BA,BB,BG,BH,BR,BW,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DO,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IS,JP,KE,KG,KM,KN,KP,KR,KZ,LA,LC,LK,LR,LS,LT,LU,LY,MA,MD,ME,MG,MK,MN,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PG,PH,PL,PT,RO,RS,RU,SC,SD,SE,SG,SK,SL,SM,ST,SV,SY,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,ZA,ZM,ZW

（特許庁注：以下のものは登録商標）
１．ＪＡＶＡ

(72)発明者 ムニール コシャンワラ
アメリカ合衆国 ０７９２０ ニュージャージー州 バスキング リッジ サウス アルワード アベニュー １５４
(72)発明者 ジョセフィン ミカレフ
アメリカ合衆国 ０７０４０ ニュージャージー州 メープルウッド ミッドランド ブールバード ７８
(72)発明者 ジョン アール. ウラート ザ セカンド
アメリカ合衆国 ０８８３６ ニュージャージー州 マーチンズビル メイフラワー コート １１０６

Ｆターム(参考) 5B376 CA08 FA13