

【公報種別】特許法第 17 条の 2 の規定による補正の掲載

【部門区分】第 6 部門第 3 区分

【発行日】平成28年10月6日(2016.10.6)

【公表番号】特表2015-526821(P2015-526821A)

【公表日】平成27年9月10日(2015.9.10)

【年通号数】公開・登録公報2015-057

【出願番号】特願2015-528554(P2015-528554)

【国際特許分類】

G 0 6 F 9/50 (2006.01)

G 0 6 F 12/08 (2016.01)

【F I】

G 0 6 F 9/46 4 6 5 C

G 0 6 F 12/08 5 5 1 C

G 0 6 F 12/08 5 6 5

G 0 6 F 12/08 5 1 3

【手続補正書】

【提出日】平成28年8月17日(2016.8.17)

【手続補正 1】

【補正対象書類名】特許請求の範囲

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【特許請求の範囲】

【請求項 1】

処理リソース割り当て方法であって、
 予め選択されたデータ場所を有するデータにマッピングされた命令を識別する工程と、
処理リソース割り当て装置により、前記予め選択されたデータ場所に基づいて第 1 の処理装置（P U）から第 2 の P U に前記命令のコールを転送するか否かを判定する工程と、
 を含む処理リソース割り当て方法。

【請求項 2】

前記第 1 の P U に関連付けられたキャッシュ・メモリ装置を選択する工程と、
 前記命令を前記キャッシュ・メモリ装置にロードする工程と、
 前記命令に関連付けられた境界において、前記キャッシュ・メモリ装置に、余分なコードの組をインジェクトする工程であって、前記命令がターゲット命令であり、前記余分なコードの組は、前記コールを特定の P U に転送するか否かを前記第 1 の P U が判定することを可能にするよう構成される工程と、
 を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 3】

予め選択された 1 つ又は複数の実行可能なルーチンそれぞれに対応する 1 つ又は複数の実行可能なルーチン名を識別する工程と、
 前記 1 つ又は複数の実行可能なルーチン名に基づいて、前記余分なコードの組を選択して、参照渡しコールから値渡しコールに前記コールを変換する工程と、
 を更に含む請求項 2 記載の処理リソース割り当て方法。

【請求項 4】

前記第 1 の P U 及び前記第 2 の P U それぞれに関連付けられた第 1 のキャッシュ・メモリ装置及び第 2 のキャッシュ・メモリ装置を識別する工程と、
 第 1 のデータの組及び第 2 のデータの組それぞれに関連付けられた第 1 のデータ場所及び第 2 のデータ場所を識別する工程であって、前記第 1 のデータ場所及び前記第 2 のデー

タ場所は前記第 1 のキャッシュ・メモリ装置内及び前記第 2 のキャッシュ・メモリ装置内それぞれに配置される、工程と、

前記第 1 のデータ場所及び前記第 2 のデータ場所に基づいて前記第 1 のデータの組及び前記第 2 のデータの組それぞれに対する前記第 1 のキャッシュ・メモリ装置及び前記第 2 のキャッシュ・メモリ装置の割り当てを行う工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 5】

前記命令にマッピングされたデータの組を識別する工程と、

前記第 1 のキャッシュ・メモリ装置でなく前記第 2 のキャッシュ・メモリ装置に前記データの組が現在記憶されていることを判定して、それにより、前記第 1 の P U から前記第 2 の P U に前記コールを転送することを決定する工程と、

を更に含む請求項 4 記載の処理リソース割り当て方法。

【請求項 6】

前記第 1 の P U 及び前記第 2 の P U それぞれに関連付けられた第 1 のキャッシュ・メモリ装置及び第 2 のキャッシュ・メモリ装置を識別する工程と、

前記第 1 の P U により、前記コールにアクセスする工程であって、前記コールは、前記予め選択されたデータ場所に前記命令がマッピングされている旨を示す引数を含む、工程と、

前記第 1 の P U によって前記引数を解析する工程であって、前記予め選択されたデータ場所が前記第 1 のキャッシュ・メモリ装置内に配置されていない旨の判定を行う、工程と、

前記判定に基づいて潜在的なりダイレクト・コールとして前記コールを識別する工程と

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 7】

前記予め選択されたデータ場所が前記第 2 のキャッシュ・メモリ装置に配置されていない旨を判定する工程であって、それにより、前記第 1 の P U により、前記潜在的なりダイレクト・コールに関連付けられたターゲット・コードを実行することの決定を行う、工程と、

前記決定に基づいて前記第 1 の P U により、前記ターゲット・コードを実行する工程と

を更に含む請求項 6 記載の処理リソース割り当て方法。

【請求項 8】

前記予め選択されたデータ場所が前記第 2 のキャッシュ・メモリ装置内に配置されていることを判定する工程であって、それにより、前記潜在的なりダイレクト・コールを実際のリダイレクト・コールとして識別する、工程と、

前記第 1 の P U から前記第 2 の P U に前記実際のリダイレクト・コールを転送する工程と、

を更に含む請求項 6 記載の処理リソース割り当て方法。

【請求項 9】

前記命令に関連付けられたコード又はデータ・メモリ場所を識別する工程と、

前記第 2 の P U に関連付けられた前記コード又はデータ・メモリ場所に基づいて前記第 1 の P U から前記第 2 の P U に前記コールを転送する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 10】

前記コード又はデータ・メモリ場所が前記第 2 の P U 及び複数の他の P U に関連付けられていることを判定する工程と、

前記第 2 の P U に対応する現在利用可能な処理機能を反映する数値測定を定義する工程と、

複数の他の現在利用可能な処理機能それぞれを反映する複数の他の数値測定を定義する

工程であって、前記複数の他の現在利用可能な処理機能は前記複数の他の P U それぞれに対応する、工程と、

前記数値測定が前記他の数値測定それぞれよりも大きいことに基づいて前記第 1 の P U から前記第 2 の P U に前記コールを転送する工程と、

を更に含む請求項 9 記載の処理リソース割り当て方法。

【請求項 1 1】

特定のリソースを選択する工程と、

前記第 2 の P U に割り当てられている前記特定のリソースの制御の度合いに基づき、前記特定のリソースへのアクセスを前記第 1 の P U が欠いていることに基づき、かつ、前記特定のリソースへのアクセスを 1 つ又は複数の他の P U が欠いていることに基づいて前記第 1 の P U から前記第 2 の P U に前記コールを転送する工程と、

を更に含む請求項 9 記載の処理リソース割り当て方法。

【請求項 1 2】

前記命令を記憶させた共有ランダム・アクセス・メモリ (R A M) 装置を選択する工程と、

前記共有 R A M 装置のセクタをリダイレクト・セクタとして識別する工程と、

前記命令が前記リダイレクト・セクタに記憶されているか否かを判定する工程であって、それにより、前記第 1 の P U から前記第 2 の P U に前記コールを転送するか否かを判定する、工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 1 3】

共有 R A M 装置を選択する工程と、

前記共有 R A M 装置の 1 つ又は複数のセクタを 1 つ又は複数のリダイレクト・セクタそれぞれとして識別する工程と、

前記命令を含むプログラムを識別する工程と、

前記プログラムに対応するスタティック・コードにアクセスする工程と、

前記スタティック・コードのサイズに基づいて前記 1 つ又は複数のリダイレクト・セクタの全体としてのサイズを定義することにより、前記共有 R A M 装置をパーティショニングする工程と、

前記パーティショニング及び P U 仕様に基づいて前記 1 つ又は複数のリダイレクト・セクタのうちの 1 つ又は複数の選択されたリダイレクト・セクタに前記スタティック・コードの量をロードする工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 1 4】

前記共有 R A M 装置の 1 つ又は複数の別々のセクタを 1 つ又は複数の非リダイレクト・セクタそれぞれとして識別する工程と、

前記プログラムに対応する少なくとも 1 つの共有ライブラリの 1 つ又は複数の部分にアクセスする工程と、

前記少なくとも 1 つの共有ライブラリの 1 つ又は複数の部分及び前記スタティック・コードの全体としてのサイズに基づいて前記 1 つ又は複数のリダイレクト・セクタの前記全体としてのサイズを定義することにより、前記共有 R A M 装置をプロポーショニングする工程と、

前記プロポーショニング及び前記 P U 仕様に基づいて前記 1 つ又は複数の非リダイレクト・セクタのうちの 1 つ又は複数の選択された非リダイレクト・セクタに、又は前記 1 つ又は複数のリダイレクト・セクタのうちの 1 つ又は複数の選択されたリダイレクト・セクタに前記スタティック・コードの 1 つ又は複数の部分をロードする工程、及び／又は、

前記プロポーショニング及び前記 P U 仕様に基づいて前記 1 つ又は複数の非リダイレクト・セクタのうちの 1 つ又は複数の選択された非リダイレクト・セクタに、又は 1 つ又は複数のリダイレクト・セクタのうちの 1 つ又は複数の選択されたリダイレクト・セクタに前記少なくとも 1 つの共有ライブラリの前記 1 つ又は複数の部分をロードする工程と、

を更に含む請求項 1 3 記載の処理リソース割り当て方法。

【請求項 1 5】

前記第 1 の P U に関連付けられたキャッシュ・メモリ装置を選択する工程と、

前記命令に関連付けられたコード又はデータを前記キャッシュ・メモリ装置にフェッチする工程と、

前記第 1 の P U に関連付けられた P U データ・バスを識別する工程と、

前記 P U データ・バス内のシャドー・レジスタ値を、前記コード又はデータに関連付けられた少なくとも 1 つの P U を示す P U 識別子を前記シャドー・レジスタ値が反映するように規定する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 1 6】

前記命令に関連付けられたコードを識別する工程と、

少なくとも 1 つの共有メモリ・コヒーレンス・マシン命令を含む実行可能なルーチンを識別する工程と、

前記実行可能なルーチンが前記第 2 の P U に関連付けられていることに基づき、かつ、前記コードが、前記少なくとも 1 つの共有メモリ・コヒーレンス・マシン命令にマッピングされていることに基づき、前記第 1 の P U から前記第 2 の P U に前記コールを転送する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 1 7】

前記命令に関連付けられた大域メモリ場所を識別する工程と、

第 3 の P U が前記大域メモリ場所へのアクセスを欠いていること、及び前記第 1 の P U が前記大域メモリ場所へのアクセスを有していることの結果として、前記第 3 の P U から前記第 1 の P U へ前記コールを自動的に転送する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 1 8】

特定の P U に関連付けられたキャッシュ・メモリ装置を識別する工程と、

前記予め選択されたデータ場所が前記キャッシュ・メモリ装置内に配置されていることに基づいて前記データに関連付けられた特定のデータ・タイプに対する前記キャッシュ・メモリ装置の割り当てを行う工程と、

前記命令を含むプログラムにアクセスする工程と、

前記プログラムに対応するプログラミング言語を識別する工程と、

前記プログラミング言語に対応するライブラリにアクセスする工程であって、前記ライブラリは、メモリ割り当てコールを含むコンストラクタを規定し、前記メモリ割り当てコールは前記特定のデータ・タイプを有する情報にマッピングされる、工程と、

前記割り当てに基づいて、前記メモリ割り当てコールに対する修正を実現する工程であって、それにより、前記プログラムの修正バージョン及び修正メモリ割り当てコールを作成する、工程と、

前記修正の結果として前記情報が前記キャッシュ・メモリ装置にロードされるように、前記修正されたメモリ割り当てコールを行う工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 1 9】

前記コールを行う工程であって、それにより、前記割り当ての結果として前記特定の P U により、前記プログラムの前記修正バージョンを実行する、工程と、

を更に含む請求項 1 8 記載の処理リソース割り当て方法。

【請求項 2 0】

1 つ又は複数のコード・ライブラリを識別する工程と、

前記 1 つ又は複数のコード・ライブラリに関連付けられた参照渡しソース・コードにアクセスする工程と、

前記参照渡しソース・コードを値渡しサポート・コードに変換する工程であって、それ

により、転送ライブラリを生成する、工程と、

前記転送ライブラリを利用して、前記データの参照渡しでなく、値渡しを行う工程と、
を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 2 1】

前記コール内の 1 つ又は複数のポインタを識別する工程と、

前記 1 つ又は複数のポインタを変換する工程であって、それにより、前記データの参照渡しでなく値渡しを行うよう前記コールを構成する、工程と、

前記第 1 の P U から前記第 2 の P U への前記コールの転送を実行する工程と、

前記転送中に前記データの値渡しを行う工程であって、それにより、前記データの局所の複製を第 1 のキャッシュ・メモリ装置に記憶し、前記第 2 の P U が前記第 1 のキャッシュ・メモリ装置を含む、工程と、

前記 1 つ又は複数のポインタを前記局所の複製にリマッピングする工程であって、それにより、第 2 のキャッシュ・メモリ装置にアクセスすることなく、前記コールを実行することを可能にし、前記第 1 の P U が、前記第 2 のキャッシュ・メモリ装置を含む、工程と

、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 2 2】

遠隔プロシージャ・コール（R P C）を起動させる工程であって、それにより、前記第 1 の P U から前記第 2 の P U に前記コールを転送する、工程、
を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 2 3】

前記第 1 の P U 及び前記第 2 の P U それぞれに関連付けられた第 1 の局所メモリ・マップ及び第 2 の局所メモリ・マップを識別する工程と、

前記第 1 の局所メモリ・マップと前記第 2 の局所メモリ・マップとの間の区別を識別する工程と、

前記区別に基づいてメモリ・アドレス変換を生成する工程と、

前記メモリ・アドレス変換を利用して前記 R P C を構成する工程と、

を更に含む請求項 2 2 記載の処理リソース割り当て方法。

【請求項 2 4】

前記第 1 の P U 及び前記第 2 の P U それぞれに関連付けられた第 1 のキャッシュ・メモリ装置及び第 2 のキャッシュ・メモリ装置を識別する工程と、

前記第 1 のキャッシュ・メモリ装置に記憶され、前記命令に関連付けられたデータ値を識別する工程と、

前記 R P C の結果として前記第 2 のキャッシュ・メモリ装置に前記データ値を記憶する工程と、

を更に含む請求項 2 2 記載の処理リソース割り当て方法。

【請求項 2 5】

前記第 2 のキャッシュ・メモリに前記データ値が記憶されるまで前記第 2 の P U による前記コールの実行を遅延させる工程と、

を更に含む請求項 2 4 記載の処理リソース割り当て方法。

【請求項 2 6】

制御の第 1 のスレッドに関連付けられた実行可能なルーチン及び前記命令を含むプログラムにアクセスする工程と、

前記コール内の 1 つ又は複数のポインタを識別する工程と、

前記 1 つ又は複数のポインタを変換する工程であって、前記データの参照渡しでなく値渡しを行うよう前記コールを構成して、それにより、制御の第 2 のスレッドを作成する、工程と、

前記第 2 の P U に関連付けられたキャッシュ・メモリ装置を識別する工程と、

前記 R P C の結果として前記データの値渡しを行う工程であって、それにより、前記データの局所の複製を前記キャッシュ・メモリ装置に記憶し、前記第 2 の P U に前記制御の

第 2 のスレッドを供給する、工程と、

前記制御の第 1 のスレッドが前記制御の第 2 のスレッドと無関係であることの結果として、前記第 2 の P U が前記命令の実行を前記制御の第 2 のスレッドに基づいて行う一方で、前記制御の第 1 のスレッドに基づいて前記第 1 の P U により、前記実行可能なルーチンを行う工程であって、前記実行可能なルーチンが、前記実行中に生成されたデータ値に対するリターン要求を欠いている、工程と、

を更に含む請求項 2 2 記載の処理リソース割り当て方法。

【請求項 2 7】

前記データにマッピングされた第 1 の実行可能なルーチンを行う第 1 のプロセッシング・コアを識別する工程であって、前記第 1 の実行可能なルーチンは共有メモリ・コヒーレンス・マシン命令を含む、工程と、

第 2 のプロセッシング・コアにおいて第 2 の実行可能なルーチンにアクセスする工程と、

プロシージャ・コールを起動する工程であって、それにより、前記第 2 の実行可能なルーチンが前記データにマッピングされていることに基づき、かつ、前記共有メモリ・コヒーレンス・マシン命令を含む前記第 2 の実行可能なルーチンに基づき、前記第 2 のプロセッシング・コアから前記第 1 のプロセッシング・コアに前記第 2 の実行可能なルーチンを転送する、工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 2 8】

別々の P U は、前記第 1 のプロセッシング・コア及び前記第 2 のプロセッシング・コアそれぞれを含み、前記プロシージャ・コールが R P C である、

請求項 2 7 記載の処理リソース割り当て方法。

【請求項 2 9】

単一の P U は、前記第 1 のプロセッシング・コア及び前記第 2 のプロセッシング・コアを何れも含み、前記プロシージャ・コールが R P C である、

請求項 2 7 記載の処理リソース割り当て方法。

【請求項 3 0】

前記命令を含むプログラムにアクセスする工程と、

前記プログラムに対応するプログラミング言語を識別する工程であって、1 つ又は複数の仮想関数を有するクラスに属するデータ項目が、仮想関数テーブル (V F T) にマッピングされた 1 つ又は複数の V F T ポインタを含むように前記 V F T を実現するよう前記プログラミング言語が構成される、工程と、

前記プログラミング言語に関連付けられたコンストラクタを識別する工程であって、前記コンストラクタは前記 1 つ又は複数の V F T ポインタを規定するよう構成される、工程と、

前記プログラムのロード中に前記コンストラクタ周りにラップ・コールをインジェクトする工程であって、それにより、前記 1 つ又は複数の仮想関数に対する 1 つ又は複数の R P C 決定ラップを含む、V F T へのポインタで、前記 1 つ又は複数の V F T ポインタを置き換える、工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 3 1】

前記命令を含むプログラムにアクセスする工程と、

前記プログラムに対応するプログラミング言語を識別する工程であって、1 つ又は複数の仮想関数を有するクラスに属するデータ項目が、V F T にマッピングされた 1 つ又は複数の V F T ポインタを含むように前記 V F T を実現するよう前記プログラミング言語が構成される、工程と、

R P C 決定コードがコールされることなく、前記 1 つ又は複数の仮想関数が 1 つ又は複数の R P C それぞれに自動的に誘導されるように複数の R P C の実現の結果として前記 V F T を修正する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 3 2】

前記命令を含むプログラムにアクセスする工程と、

前記プログラムに対応するプログラミング言語を識別する工程であって、1 つ又は複数の仮想関数を有するクラスに属するデータ項目が、V F T にマッピングされた 1 つ又は複数の V F T ポインタを含むように前記 V F T を実現するよう前記プログラミング言語が構成される、工程と、

R P C 決定コードがコールされることなく、前記 1 つ又は複数の仮想関数が前記プログラムに関連付けられた局所コードに自動的に誘導されるように 1 つ又は複数の R P C の実現がないことの結果として前記 V F T を修正する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 3 3】

複数の条件分岐にマッピングされたコードを含む前記命令の予め定義されたバージョンにアクセスする工程と、

前記命令の複数の修正バージョンを生成する工程であって、前記修正バージョンはそれぞれ、前記複数の条件分岐のうちの別の条件分岐をたどるよう修正されたコードを含む工程と、

前記複数の修正バージョンそれぞれに関連付けられた複数の挙動を予測する工程と、

前記命令に関連付けられた実行可能なルーチンにアクセスする工程と、

前記実行可能なルーチンに関連付けられたランタイム挙動を識別する工程と、

前記ランタイム挙動に対応する前記修正バージョンの予測された挙動に基づいて前記実行可能なルーチンの動作中の実行のために前記複数の修正バージョンのうちの修正バージョンを選択する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 3 4】

前記命令に関連付けられた実行可能なルーチンにアクセスする工程と、

仮想関数の第 1 の実現形態及び第 2 の実現形態を識別する工程であって、前記仮想関数は前記第 1 の実現形態を指し示す V F T エントリを含み、前記仮想関数は更に、複数の V F T のうちの第 1 の V F T にマッピングされた V F T ポインタを含む、工程と、

前記第 1 の実現形態により、前記実行可能なルーチンの一部分の動作を起動させる工程と、

前記動作中の前記第 1 の実現形態によって表される分岐挙動を識別する工程と、

前記分岐挙動の動作解析を行う工程であって、それにより、前記第 2 の実現形態が前記分岐挙動に対応する度合いが、前記第 1 の実現形態が前記分岐挙動に対応する度合いよりも大きいことを判定する、工程と、

前記動作解析に基づいて前記複数の V F T のうちの第 2 の V F T に前記 V F T ポインタをリマッピングする工程であって、前記第 2 の V F T が、前記第 2 の実現形態にマッピングされたポインタを含む、工程、又は、

前記動作解析に基づいて前記第 2 の実現形態を指し示すよう前記 V F T エントリを更新する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 3 5】

前記命令に関連付けられたプログラムにアクセスする工程であって、前記プログラムは、予め定義されたコール・サイトにマッピングされた予め定義された関数を含む、工程と、

前記予め定義されたコール・サイトにより、前記予め定義された関数のコールを行う工程と、

前記コールの結果として、前記予め定義された関数によって表される分岐挙動を識別する工程と、

前記分岐挙動に基づいて、前記予め定義されたコール・サイトを修正する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 36】

前記命令に関連付けられた実行可能なルーチンにアクセスする工程と、

仮想関数の第 1 の実現形態及び第 2 の実現形態を識別する工程であって、前記仮想関数は前記第 1 の実現形態を指し示す V F T エントリを含み、前記仮想関数は更に、複数の V F T のうちの第 1 の V F T にマッピングされた V F T ポインタを更に含む、工程と、

前記第 1 の実現形態により、前記実行可能なルーチンの一部分の動作を起動させる工程と、

前記動作中の前記第 1 の実現形態によって表される分岐挙動を識別する工程と、

前記分岐挙動に基づいて前記仮想関数の考えられる将来の挙動を予測する工程と、

前記考えられる将来の挙動の動作解析を行う工程であって、それにより、前記第 2 の実現形態が前記考えられる将来の挙動に対応する度合いが、前記第 1 の実現形態が前記考えられる将来の挙動に対応する度合いよりも大きいことを判定する、工程と、

前記動作解析に基づいて前記複数の V F T のうちの第 2 の V F T に前記 V F T ポインタをリマッピングする工程であって、前記第 2 の V F T が、前記第 2 の実現形態にマッピングされたポインタを含む、工程、又は、

前記動作解析に基づいて前記第 2 の実現形態を指し示すよう前記 V F T エントリを更新する工程と、

を更に含む請求項 1 記載の処理リソース割り当て方法。

【請求項 37】

コンピュータ・システムに、請求項 1 乃至 36 のいずれか一項に記載の方法を実行させるコンピュータ・プログラム。

【請求項 38】

請求項 37 に記載のコンピュータ・プログラムを記録したコンピュータ可読記憶媒体。