

【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6月16日(2016.6.16)

【公表番号】特表2015-516098(P2015-516098A)

【公表日】平成27年6月4日(2015.6.4)

【年通号数】公開・登録公報2015-036

【出願番号】特願2015-511521(P2015-511521)

【国際特許分類】

G 0 6 F 9/44 (2006.01)

G 0 6 F 9/54 (2006.01)

G 0 6 F 9/48 (2006.01)

【F I】

G 0 6 F 9/44 5 3 0 M

G 0 6 F 9/06 6 2 0 A

G 0 6 F 9/06 6 4 0 C

G 0 6 F 9/46 4 5 7

G 0 6 F 9/44 5 3 0 P

【手続補正書】

【提出日】平成28年4月19日(2016.4.19)

【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】

1 つ以上のマイクロプロセッサ上で動作するオブジェクト指向プログラミング言語環境において遅延参照をサポートするための方法であって、

将来利用可能になるオブジェクトに対して遅延参照を提供するステップと、

前記遅延参照に基づいて前記オブジェクトを得るように呼び出し側から要求を受取った後、

前記オブジェクトが利用可能である場合には前記オブジェクトを前記呼び出し側に返すこと、ならびに、

前記オブジェクトが現在のところ利用可能でない場合および前記オブジェクトが利用可能にならない場合のうちの少なくとも 1 つの場合には前記呼び出し側に通知すること、のうちの 1 つを実行するステップとを備える、方法。

【請求項 2】

オブジェクト指向プログラミング言語環境において遅延参照をサポートするためのシステムであって、

1 つ以上のマイクロプロセッサと、

前記 1 つ以上のマイクロプロセッサ上で起動するインターフェースとを備え、前記インターフェースは、

将来利用可能になるオブジェクトに対して遅延参照を提供するステップと、

前記遅延参照に基づいて前記オブジェクトを得るように呼び出し側から要求を受取った後、

前記オブジェクトが利用可能である場合には前記オブジェクトを前記呼び出し側に返すこと、ならびに、

前記オブジェクトが現在のところ利用可能でない場合および前記オブジェクトが利

用可能にならない場合のうちの少なくとも1つの場合には前記呼び出し側に通知すること、のうちの1つを実行するステップとを実行するように動作する、システム。

【請求項3】

前記インターフェースは、前記オブジェクトが利用可能になるのを待っている間は、前記遅延参照に基づいて前記オブジェクトを要求するために前記呼び出し側が用いる呼び出しスレッドをブロックすることを回避できる、請求項2に記載のシステム。

【請求項4】

前記インターフェースは、前記オブジェクトが利用可能になる前に、前記オブジェクトがどのクラスに関連付けられるかについて前記呼び出し側に知らせることができる、請求項2または3に記載のシステム。

【請求項5】

前記インターフェースは、前記オブジェクトが現在のところ利用可能でない場合に、ランタイム例外を実行するか、またはヌル値を返すことができる、請求項2～4のいずれか1項に記載のシステム。

【請求項6】

前記インターフェースは、前記オブジェクトが利用可能にならない場合に、オブジェクト利用不可例外を実行することができる、請求項2～5のいずれか1項に記載のシステム。

【請求項7】

前記オブジェクトが利用可能になるのを前記呼び出し側が最大期間だけ待つことを確実にすることができるヘルパーをさらに備える、請求項2～6のいずれか1項に記載のシステム。

【請求項8】

前記オブジェクトに関連付けられるクラス定義に基づいてオブジェクトに対する遅延参照を作成することができるヘルパーをさらに備える、請求項2～7のいずれか1項に記載のシステム。

【請求項9】

前記インターフェースは、遅延参照を別の遅延参照にネスティングすることができる、請求項2～8のいずれか1項に記載のシステム。

【請求項10】

前記オブジェクトの記録動的プロキシを得ることができる、

前記オブジェクトの前記記録動的プロキシによって、最後に記録された呼び出しの遅延表現を提供することができるヘルパーをさらに備える、請求項2～9のいずれか1項に記載のシステム。

【請求項11】

命令を含む機械読取可能なプログラムであって、前記命令は、実行されると、コンピュータに

将来利用可能になるオブジェクトに対して遅延参照を提供するステップと、

前記遅延参照に基づいて前記オブジェクトを得るように呼び出し側から要求を受取った後、

前記オブジェクトが利用可能である場合には前記オブジェクトを前記呼び出し側に返すこと、ならびに、

前記オブジェクトが現在のところ利用可能でない場合および前記オブジェクトが利用可能にならない場合のうちの少なくとも1つの場合には前記呼び出し側に通知すること、のうちの1つを実行するステップとを、実行させる、プログラム。

【請求項12】

請求項1に記載の方法をコンピュータに実現させるためのコンピュータプログラム。

【手続補正2】

【補正対象書類名】明細書

【補正対象項目名】0011

【補正方法】変更

【補正の内容】

【0011】

OOP環境100において、呼び出し側102は、遅延参照101に基づいて将来のオブジェクト103を得るように要求、例えばgetメソッドを送り得る。指示対象または将来のオブジェクト103を決定するための計算は、getメソッドが呼び出されるまでは開始しないであろう。一方、呼び出し側102に遅延参照101を返すことは、直ちに行われる。

【手続補正3】

【補正対象書類名】明細書

【補正対象項目名】0014

【補正方法】変更

【補正の内容】

【0014】

本発明のさまざまな実施の形態によれば、OOP環境は、将来のオブジェクト103が利用可能になるのを待っている間は、将来のオブジェクト103を要求するために呼び出し側102が用いる呼び出しスレッド104をブロックすることを回避し得る。呼び出しスレッド104が他の重要なタスクに関与し得るので、このような非同期モデルは有益であり得る。

【手続補正4】

【補正対象書類名】明細書

【補正対象項目名】0017

【補正方法】変更

【補正の内容】

【0017】

将来のオブジェクト103の遅延参照101を用いて、OOP環境は、JAVA Futureクラスに関連するさまざまな問題を回避することができる。例えば、値を決定するための計算は、将来のクラスが作成されると開始し得る。さらに、将来のクラスは、呼び出しスレッドを強制的に待機させ得る。例えば、JAVA Futureクラスは、値が生成されるまでは永久に待機するように呼び出しスレッドをブロックしなければならないFuture.get()メソッド、または、呼び出しスレッドを特定の時間ブロックするFuture.get(time)メソッドを提供し得る。これらのメソッドは両方とも、非同期でないブロッキングアルゴリズムを開発者に書かせる。

【手続補正5】

【補正対象書類名】明細書

【補正対象項目名】0073

【補正方法】変更

【補正の内容】

【0073】

システム500は、ヘルパー520をさらに備え得る。ヘルパー520は、オブジェクトが利用可能になるのを呼び出し側502が最大期間だけ待つことを確実にすることができる。また、ヘルパー520は、オブジェクトに関連付けられるクラス定義に基づいてオブジェクトに対する遅延参照を作成することができる。さらに、ヘルパー520は、オブジェクトの記録動的プロキシを得て、オブジェクトの記録動的プロキシによって、最後に記録された呼び出しの遅延表現を提供することができる。