(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2020年7月23日(23.07.2020)

WIPO | PCT

(10) 国際公開番号
WO 2020/148815 A1

(51) 国際特許分類:
G06F 11/14 (2006.01)　*G06F 9/50* (2006.01)

(21) 国際出願番号：　PCT/JP2019/000977

(22) 国際出願日：　2019年1月15日(15.01.2019)

(25) 国際出願の言語：　日本語

(26) 国際公開の言語：　日本語

(71) 出願人: 富士通株式会社(**FUJITSU LIMITED**) [JP/JP]; 〒2118588 神奈川県川崎市中原区上小田中４丁目１番１号 Kanagawa (JP).

(72) 発明者: 中川　岳 (**NAKAGAWA, Gaku**); 〒2118588 神奈川県川崎市中原区上小田中４丁目１番１号 富士通株式会社内 Kanagawa (JP). 金政 泰彦(**KANEMASA, Yasuhiko**); 〒2118588 神奈川県川崎市中原区上小田中４丁目１番１号 富士通株式会社内 Kanagawa (JP).

(74) 代理人: 酒井 昭徳(**SAKAI, Akinori**); 〒1020094 東京都千代田区紀尾井町３番１２号 紀尾井町ビル７階 酒井総合特許事務所 Tokyo (JP).

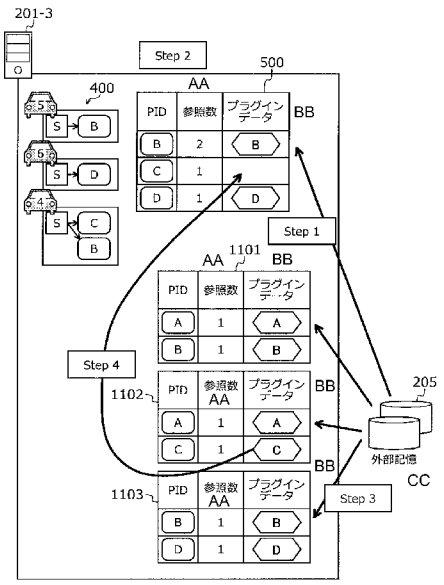
(81) 指定国(表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) 指定国(表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ,

(54) Title: TAKEOVER PROGRAM, TAKEOVER METHOD, INFORMATION PROCESSING DEVICE, AND PARALLEL DISTRIBUTED PROCESSING SYSTEM

(54) 発明の名称： 引継プログラム、引継方法、情報処理装置、および並列分散処理システム

[図11]



1, 2, 3, 4　Step
AA　Reference number
BB　Plug-in data
CC　External storage

(57) Abstract: A distributed node (201-3) acquires state information relating to plug-ins for implementing each of one or more functions allocated to an external storage device (205), including a takeover function, from the external storage device (205), and regenerates a state table (410) list (400). The distributed node (201-3) calculates a reference number for each plug-in, on the basis of the regenerated list (400), and regenerates a plug-in repository table (500). The distributed node (201-3) acquires snapshots (1101 to 1103) of the plug-in repository table (500) stored by each distributed node (201), from the external storage device (205), and generates list information summarizing plug-in information. If the plug-in repository table (500) includes plug-in information having empty plug-in data, the distributed node (201-3) acquires and sets the plug-in data from the list information.

WO 2020/148815 A1

［続葉有］

DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

添付公開書類：

一　国際調査報告（条約第21条(3)）

(57) 要約：分散ノード（２０１－３）は、外部記憶装置（２０５）から、引き継ぐ機能を含む自装置に割り振られた１以上の機能のそれぞれの機能を実現するプラグインに関するステート情報を取得し、ステートテーブル（４１０）のリスト（４００）を再生成する。分散ノード（２０１－３）は、再生成したリスト（４００）に基づいて、プラグインごとの参照数を算出し、プラグインリポジトリテーブル（５００）を再生成する。分散ノード（２０１－３）は、外部記憶装置（２０５）から、それぞれの分散ノード（２０１）が格納したプラグインリポジトリテーブル（５００）のスナップショット（１１０１～１１０３）を取得し、プラグイン情報を纏めた一覧情報を生成する。分散ノード（２０１－３）は、プラグインリポジトリテーブル（５００）のうち、プラグインデータが空であるプラグイン情報があれば、一覧情報からプラグインデータを取得して設定する。

明　細　書

発明の名称 ：

引継プログラム、引継方法、情報処理装置、および並列分散処理システム

技術分野

[0001]　本発明は、引継プログラム、引継方法、情報処理装置、および並列分散処理システムに関する。

背景技術

[0002]　従来、複数のノードを含み、複数のノードのそれぞれのノードにデータを処理する機能を実現するプラグインを実装し、複数のノードで分散して複数の端末装置のそれぞれの端末装置のデータを処理する並列分散処理システムがある。

[0003]　先行技術としては、例えば、機能セットの内のローカルにサポートされていない機能を装置ドライバがサポートすることを可能とする１以上のプラグインを識別し、利用者からの許可を得て、プラグインをダウンロードするものがある。また、例えば、複数の周辺機器に共通する第１のプラグインを介して、周辺機器のそれぞれに対応する第２のプラグインをアプリケーションの実行環境に接続されるように配置する技術がある。また、例えば、所定のコンテンツに必要なプラグインモジュールを判定する技術がある。

先行技術文献

特許文献

[0004] 特許文献1：特開２０１０－９８７３２号公報

特許文献2：特開２０１５－２０７２７３号公報

特許文献3：特開２００３－３１６５９１号公報

発明の概要

発明が解決しようとする課題

[0005]　しかしながら、従来技術では、並列分散処理システムのノード間において機能を引き継ぐことが難しいことがある。例えば、特定のノードで障害が発

生した場合、特定のノードから他のノードに機能を引き継ぐことが望まれるが、障害により特定のノードが有するプラグインが失われると、機能を引き継ぐことが難しくなることがある。

[0006] １つの側面では、本発明は、機能を引継可能にすることを目的とする。

課題を解決するための手段

[0007] １つの実施態様によれば、複数の情報処理装置のうち、自装置とは異なる他の情報処理装置が所定の端末装置に提供していた所定の機能を引き継ぐことの要求を受け付け、前記要求を受け付けた場合、前記複数の情報処理装置のそれぞれの情報処理装置によって、当該情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを集約したバックアップデータが格納される外部記憶装置から、前記他の情報処理装置によって格納されたバックアップデータを取得し、取得した前記バックアップデータを解釈し、前記他の情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードのうち、前記所定の機能を実現するプラグインのコードを抽出し、抽出した前記コードに基づいて、前記所定の機能を前記所定の端末装置に提供する引継プログラム、引継方法、情報処理装置、および並列分散処理システムが提案される。

発明の効果

[0008] 一態様によれば、機能を引継可能にすることが可能になる。

図面の簡単な説明

[0009] [図1]図１は、実施の形態にかかる引継方法の一実施例を示す説明図である。

[図2]図２は、並列分散処理システム２００の一例を示す説明図である。

[図3]図３は、分散ノード２０１のハードウェア構成例を示すブロック図である。

[図4]図４は、ステートテーブル４１０の記憶内容の一例を示す説明図である。

[図5]図５は、プラグインリポジトリテーブル５００の記憶内容の一例を示す説明図である。

[図6]図６は、情報処理装置１１０の機能的構成例を示すブロック図である。

[図7]図７は、ステートテーブル４１０を更新する一例を示す説明図である。

[図8]図８は、プラグインデータを管理する一例を示す説明図である。

[図9]図９は、ステートテーブル４１０とプラグインリポジトリテーブル５００とをバックアップする一例を示す説明図である。

[図10]図１０は、障害発生時の分散ノード２０１の動作を示す説明図（その１）である。

[図11]図１１は、障害発生時の分散ノード２０１の動作を示す説明図（その２）である。

[図12]図１２は、障害発生時の分散ノード２０１の動作を示す説明図（その３）である。

[図13]図１３は、障害発生時の分散ノード２０１の動作を示す説明図（その４）である。

[図14]図１４は、障害発生時の分散ノード２０１の動作を示す説明図（その５）である。

[図15]図１５は、障害発生時の分散ノード２０１の動作を示す説明図（その６）である。

[図16]図１６は、並列分散処理システム２００の適用例を示す説明図である。

[図17]図１７は、適用例におけるメッセージフローを示す説明図である。

[図18]図１８は、適用例における計算機１６０１の具体的な機能的構成例を示す説明図（その１）である。

[図19]図１９は、適用例における計算機１６０１の具体的な機能的構成例を示す説明図（その２）である。

[図20]図２０は、各種メッセージの一例を示す説明図である。

[図21]図２１は、全体処理手順の一例を示すフローチャートである。

[図22]図２２は、イベントメッセージ処理手順の一例を示すフローチャートである。

[図23]図２３は、ステート更新処理手順の一例を示すフローチャートである。

[図24]図２４は、プラグイン処理手順の一例を示すフローチャートである。

[図25]図２５は、制御系イベント処理手順の一例を示すフローチャートである。

[図26]図２６は、プラグイン追加処理手順の一例を示すフローチャート（その１）である。

[図27]図２７は、プラグイン追加処理手順の一例を示すフローチャート（その２）である。

[図28]図２８は、プラグイン削除処理手順の一例を示すフローチャート（その１）である。

[図29]図２９は、プラグイン削除処理手順の一例を示すフローチャート（その２）である。

[図30]図３０は、チェックポイント処理手順の一例を示すフローチャートである。

[図31]図３１は、チェックポイント保存処理手順の一例を示すフローチャートである。

[図32]図３２は、回復処理手順の一例を示すフローチャートである。

[図33]図３３は、再生成処理手順の一例を示すフローチャート（その１）である。

[図34]図３４は、再生成処理手順の一例を示すフローチャート（その２）である。

[図35]図３５は、再生成処理手順の一例を示すフローチャート（その３）である。

発明を実施するための形態

[0010] 以下に、図面を参照して、本発明にかかる引継プログラム、引継方法、情報処理装置、および並列分散処理システムの実施の形態を詳細に説明する。

[0011] （実施の形態にかかる引継方法の一実施例）

図１は、実施の形態にかかる引継方法の一実施例を示す説明図である。図１において、情報処理装置１１０は、並列分散処理システムに含まれる複数のノードのいずれかのノードとして動作することができるコンピュータである。以下の説明では、異なる情報処理装置を区別する場合、番号ｉを用いて「情報処理装置１１０－ｉ」と表記する場合がある。

[0012] それぞれのノードは、処理対象のデータを処理する処理プログラムを有し、処理対象のデータを処理する。処理プログラムは、例えば、処理対象のデータを処理する機能を実現するプラグインを呼び出すことができるメインプログラムを含む。処理プログラムは、例えば、プラグインを含むことができる。プラグインは、例えば、サービス開発者のコンピュータからノードに配信され、処理プログラムに追加される。

[0013] ここで、並列分散処理システムでは、ノード間において、処理対象のデータを処理する機能を引き継ぐことが望まれる場合がある。例えば、特定のノードで障害が発生した場合、特定のノードが担当していた機能を、他のノードに引き継ぐことが望まれる。また、例えば、並列分散処理システムに新たなノードが追加された場合、新たなノードに、他のノードが担当していた機能を引き継ぎ、負荷分散を図ることが望まれる場合がある。

[0014] しかしながら、ノードは、他のノードが担当していた機能を引き継ぐことが難しいことがある。例えば、ノードは、他のノードが提供していた機能を引き継ぐ際、機能を実現するプラグインを取得する必要があるが、プラグインを取得することができず、機能を引き継ぐことができない場合がある。

[0015] 具体的には、引き継ぐ機能を実現するプラグインを配信した、サービス開発者のコンピュータで、プラグインが既に失われている場合、ノードは、サービス開発者のコンピュータからプラグインを取得することができない。また、障害により特定のノードが有するプラグインが失われている場合、ノードは、特定のノードからプラグインを取得することができない。また、並列分散処理システムが、ノード間で直接通信可能に形成されていない場合、ノードは、他のノードから直接プラグインを取得することができない。

[0016] そこで、本実施の形態では、並列分散処理システムのノードが、処理対象のデータを処理する他のノードが担当していた機能を、他のノードが外部記憶装置に格納したバックアップデータに基づいて引継可能にする引継方法について説明する。

[0017] 図１において、並列分散処理システム１００は、３つの情報処理装置１１０を含む。以下の説明では、３つの情報処理装置１１０をそれぞれ区別する場合、「情報処理装置１１０－１」、「情報処理装置１１０－２」、「情報処理装置１１０－３」と表記する場合がある。

[0018] 情報処理装置１１０は、処理対象のデータを処理する処理プログラムを有する。処理プログラムは、例えば、処理対象のデータを処理する機能を実現するプラグインを呼び出すことができるメインプログラムを含む。処理プログラムは、例えば、プラグインを含むことができる。情報処理装置１１０は、プラグインが実現する機能を、複数の端末装置のいずれかの端末装置に提供する。機能の提供は、例えば、機能が用いるデータを入力可能にすること、または、機能が生成したデータを取得可能にすることである。

[0019] 情報処理装置１１０は、プラグインのコードを含む制御メッセージを受信したことに応じて、プラグインの実行条件とプラグインのコードとを記憶し、処理プログラムにプラグインを追加する。制御メッセージは、例えば、追加するプラグインの実行条件と、プラグインのコードとを対応付けて表す。プラグインの実行条件は、特定の端末装置での特定のイベントの発生である。イベントは、例えば、人や物のステートの変化である。ステートは、状態であり、例えば、速度である。制御メッセージは、サービス開発者のコンピュータ１２０から、情報処理装置１１０に送信される。

[0020] 図１の例では、情報処理装置１１０－１は、プラグインＰ１のコードやプラグインＰ２のコードを記憶する。情報処理装置１１０－２は、プラグインＰ３のコードやプラグインＰ４のコードを記憶する。情報処理装置１１０－３は、プラグインＰ５のコードやプラグインＰ６のコードを記憶する。

[0021] 情報処理装置１１０は、例えば、プラグインの実行条件を、図４に後述す

るステートテーブルを用いて記憶する。情報処理装置１１０は、例えば、プラグインのコードを、図５に後述するプラグインリポジトリテーブルに集約する。情報処理装置１１０は、所定のタイミングで、自装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを集約したバックアップデータを、外部記憶装置１３０に格納する。バックアップデータは、例えば、プラグインリポジトリテーブルのスナップショットである。

[0022] 図１の例では、情報処理装置１１０－１は、プラグインＰ１のコードやプラグインＰ２のコードを集約したバックアップデータｂ１を、外部記憶装置１３０に格納する。情報処理装置１１０－２は、プラグインＰ３のコードやプラグインＰ４のコードを集約したバックアップデータｂ２を、外部記憶装置１３０に格納する。情報処理装置１１０－３は、プラグインＰ５のコードやプラグインＰ６のコードを集約したバックアップデータｂ３を、外部記憶装置１３０に格納する。

[0023] ここで、情報処理装置１１０は、自装置とは異なる他の情報処理装置１１０が所定の端末装置に提供していた所定の機能を引き継ぐことの要求を受け付ける。図１の例では、情報処理装置１１０－３は、情報処理装置１１０－２が所定の端末装置に提供していた、プラグインＰ４により実現される所定の機能を引き継ぐことの要求を受け付ける。

[0024] 情報処理装置１１０は、要求を受け付けた場合、外部記憶装置１３０から、少なくとも、他の情報処理装置１１０によって格納されたバックアップデータを取得する。図１の例では、情報処理装置１１０－３は、要求を受け付けた場合、外部記憶装置１３０から、情報処理装置１１０－２によって格納されたバックアップデータｂ２を取得する。

[0025] 情報処理装置１１０は、取得したバックアップデータを解釈し、他の情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードのうち、所定の機能を実現するプラグインのコードを抽出する。解釈は、データ形式に従って、バックアップデータに含まれる各種データを区別可能にすることである。解釈は、例えば、プラグイン単位で、プ

ラグインのコードを区別可能にする。解釈は、復号を含んでもよい。図１の例では、情報処理装置１１０－３は、バックアップデータｂ２を解釈し、情報処理装置１１０－２が提供していた所定の機能を実現するプラグインＰ４のコードを抽出する。

[0026] 情報処理装置１１０は、抽出したコードに基づいて、所定の機能を所定の端末装置に提供する。図１の例では、情報処理装置１１０－３は、プラグインＰ４のコードに基づいて、所定の機能を所定の端末装置に提供する。情報処理装置１１０は、具体的には、図７～図１５に後述する動作例に示すような動作を行う。

[0027] これにより、情報処理装置１１０は、他の情報処理装置１１０が提供していた機能を実現するプラグインのコードを取得することができ、他の情報処理装置１１０が提供していた機能を引き継ぐことができる。情報処理装置１１０は、例えば、他の情報処理装置１１０が提供していた機能を実現するプラグインのコードを、プラグインの配信元から取得困難な状況であっても、プラグインのコードを取得することができる。情報処理装置１１０は、例えば、他の情報処理装置１１０が提供していた機能を実現するプラグインのコードを、他の情報処理装置１１０から取得困難な状況であっても、プラグインのコードを取得することができる。

[0028] また、情報処理装置１１０は、外部記憶装置１３０が、プラグインのコードを集約したバックアップデータを記憶しており、プラグイン単位でプラグインのコードを区別して取得困難な状況であっても、プラグインのコードを取得することができる。このため、並列分散処理システム１００は、外部記憶装置１３０に特別な機能を追加しなくても、実現することができる。

[0029] 並列分散処理システム１００は、例えば、外部記憶装置１３０が、バックアップデータを解釈し、プラグイン単位でプラグインのコードを区別可能にする機能を有していなくても、実現することができる。このため、並列分散処理システム１００は、例えば、外部記憶装置１３０が、バックアップデータ内を検索し、プラグイン単位でプラグインのコードを抽出することが難し

い状況でも、実現することができる。結果として、並列分散処理システム１００は、外部記憶装置１３０として、クラウド上の記憶領域などを利用することができる。

[0030] （並列分散処理システム２００の一例）

次に、図２を用いて、図１に示した情報処理装置を適用した、並列分散処理システム２００の一例について説明する。

[0031] 図２は、並列分散処理システム２００の一例を示す説明図である。図２において、並列分散処理システム２００は、分散ノード２０１－１～２０１－Ｎと、管理ノード２０２と、端末装置２０３と、開発ノード２０４と、外部記憶装置２０５とを含む。以下の説明では、分散ノード２０１－１～２０１－Ｎを区別せずに「分散ノード２０１」と表記する場合がある。

[0032] 並列分散処理システム２００において、分散ノード２０１と、管理ノード２０２と、端末装置２０３と、開発ノード２０４と、外部記憶装置２０５とは、有線または無線のネットワーク２１０を介して接続される。ネットワーク２１０は、例えば、ＬＡＮ（Ｌｏｃａｌ　Ａｒｅａ　Ｎｅｔｗｏｒｋ）、ＷＡＮ（Ｗｉｄｅ　Ａｒｅａ　Ｎｅｔｗｏｒｋ）、インターネットなどである。

[0033] 分散ノード２０１は、図１に示した情報処理装置として動作するコンピュータである。分散ノード２０１は、制御メッセージを受信すると、制御メッセージに応じてプラグインを追加または削除する。制御メッセージは、例えば、追加するプラグインの実行条件と、プラグインのコードとを対応付けて表す。プラグインの実行条件は、特定の端末装置での特定のイベントの発生である。イベントは、例えば、人や物のステートの変化である。ステートは、状態であり、例えば、速度である。プラグインのコードは、実行形式である。

[0034] 分散ノード２０１は、例えば、制御メッセージを受信すると、プラグインの実行条件と、プラグインのコードとを対応付けて、図４および図５に後述する各種テーブルを用いて記憶する。分散ノード２０１は、具体的には、プ

ラグインの実行条件とプラグインＩＤとを対応付けて図４に後述するステートテーブル４１０を用いて記憶する。分散ノード２０１は、具体的には、ステートテーブル４１０を複数含む、図４に後述する、ステートテーブル４１０のリスト４００を記憶する。また、分散ノード２０１は、具体的には、プラグインＩＤとプラグインのコードとを対応付けて図５に後述するプラグインリポジトリテーブル５００を用いて記憶する。分散ノード２０１は、図４および図５に後述する各種テーブルを、外部記憶装置２０５にバックアップする。

[0035] 分散ノード２０１は、イベントメッセージを受信すると、イベントメッセージに応じてプラグインを実行する。イベントメッセージは、イベントと、イベントの発生元の人や物に対応する端末装置とを表す。イベントメッセージは、例えば、人や物のステートの変化などのイベントがあったタイミングにおける、人や物のステートを表す。分散ノード２０１は、例えば、サーバやＰＣ（Ｐｅｒｓｏｎａｌ　Ｃｏｍｐｕｔｅｒ）などである。

[0036] 管理ノード２０２は、端末装置２０３からイベントメッセージを受信し、分散ノード２０１に送信するコンピュータである。また、管理ノード２０２は、開発ノード２０４から制御メッセージを受信し、制御メッセージを分散ノード２０１に送信する。管理ノード２０２は、例えば、サーバやＰＣなどである。

[0037] 端末装置２０３は、イベントの発生元の人や物に対応するコンピュータである。端末装置２０３は、例えば、人に保持される装置である。端末装置２０３は、例えば、移動体に搭載された装置である。移動体は、例えば、車である。端末装置２０３は、イベントの発生に応じてイベントメッセージを生成し、管理ノード２０２に送信する。端末装置２０３は、例えば、ＰＣ、タブレット端末、スマートフォン、ウェアラブル端末などである。

[0038] 開発ノード２０４は、制御メッセージを管理ノード２０２に送信するコンピュータである。開発ノード２０４は、例えば、分散ノード２０１にプラグインを追加または削除させるときに、管理ノード２０２に制御メッセージを

送信する。開発ノード２０４は、例えば、サーバ、ＰＣなどである。

[0039] 外部記憶装置２０５は、分散ノード２０１から、図４および図５に後述する各種テーブルをバックアップされる。外部記憶装置２０５は、例えば、ストレージ装置である。外部記憶装置２０５は、例えば、クラウド上の記憶領域などにより実現されてもよい。

[0040] ここでは、管理ノード２０２が、分散ノード２０１とは異なる装置である場合について説明したが、これに限らない。例えば、管理ノード２０２が、いずれかの分散ノード２０１と一体である場合があってもよい。また、例えば、管理ノード２０２が、開発ノード２０４と一体である場合があってもよい。

[0041] ここでは、分散ノード２０１と管理ノード２０２と端末装置２０３と開発ノード２０４とが、ネットワーク２１０を介して接続される場合について説明したが、これに限らない。例えば、分散ノード２０１と管理ノード２０２と開発ノード２０４とを接続するネットワークと、管理ノード２０２と端末装置２０３とを接続するネットワークとが異なる場合があってもよい。

[0042] （分散ノード２０１のハードウェア構成例）

次に、図３を用いて、分散ノード２０１のハードウェア構成例について説明する。

[0043] 図３は、分散ノード２０１のハードウェア構成例を示すブロック図である。図３において、分散ノード２０１は、ＣＰＵ（Ｃｅｎｔｒａｌ Ｐｒｏｃｅｓｓｉｎｇ Ｕｎｉｔ）３０１と、メモリ３０２と、ネットワークＩ／Ｆ（Ｉｎｔｅｒｆａｃｅ）３０３と、記録媒体Ｉ／Ｆ３０４と、記録媒体３０５とを有する。また、各構成部は、バス３００によってそれぞれ接続される。

[0044] ここで、ＣＰＵ３０１は、分散ノード２０１の全体の制御を司る。メモリ３０２は、例えば、ＲＯＭ（Ｒｅａｄ Ｏｎｌｙ Ｍｅｍｏｒｙ）、ＲＡＭ（Ｒａｎｄｏｍ Ａｃｃｅｓｓ Ｍｅｍｏｒｙ）およびフラッシュＲＯＭなどを有する。具体的には、例えば、フラッシュＲＯＭやＲＯＭが各種プログ

ラムを記憶し、ＲＡＭがＣＰＵ３０１のワークエリアとして使用される。メモリ３０２に記憶されるプログラムは、ＣＰＵ３０１にロードされることで、コーディングされている処理をＣＰＵ３０１に実行させる。メモリ３０２は、例えば、図４および図５に後述する各種テーブルを記憶する。

[0045] ネットワークＩ／Ｆ３０３は、通信回線を通じてネットワーク２１０に接続され、ネットワーク２１０を介して他のコンピュータに接続される。そして、ネットワークＩ／Ｆ３０３は、ネットワーク２１０と内部のインターフェースを司り、他のコンピュータからのデータの入出力を制御する。ネットワークＩ／Ｆ３０３は、例えば、モデムやＬＡＮアダプタなどである。

[0046] 記録媒体Ｉ／Ｆ３０４は、ＣＰＵ３０１の制御に従って記録媒体３０５に対するデータのリード／ライトを制御する。記録媒体Ｉ／Ｆ３０４は、例えば、ディスクドライブ、ＳＳＤ（Ｓｏｌｉｄ　Ｓｔａｔｅ　Ｄｒｉｖｅ）、ＵＳＢ（Ｕｎｉｖｅｒｓａｌ　Ｓｅｒｉａｌ　Ｂｕｓ）ポートなどである。記録媒体３０５は、記録媒体Ｉ／Ｆ３０４の制御で書き込まれたデータを記憶する不揮発メモリである。記録媒体３０５は、例えば、ディスク、半導体メモリ、ＵＳＢメモリなどである。記録媒体３０５は、分散ノード２０１から着脱可能であってもよい。記録媒体３０５は、例えば、図４および図５に後述する各種テーブルを記憶してもよい。

[0047] 分散ノード２０１は、上述した構成部の他、例えば、キーボード、マウス、ディスプレイ、プリンタ、スキャナ、マイク、スピーカーなどを有してもよい。また、分散ノード２０１は、記録媒体Ｉ／Ｆ３０４や記録媒体３０５を複数有していてもよい。また、分散ノード２０１は、記録媒体Ｉ／Ｆ３０４や記録媒体３０５を有していなくてもよい。

[0048] （ステートテーブル４１０の記憶内容）

次に、図４を用いて、ステートテーブル４１０の記憶内容の一例について説明する。ステートテーブル４１０は、例えば、図３に示した分散ノード２０１のメモリ３０２や記録媒体３０５などの記憶領域により実現される。

[0049] 図４は、ステートテーブル４１０の記憶内容の一例を示す説明図である。

図４に示すように、ステートテーブル４１０は、ＩＤと、ステート名と、ステート内容と、プラグインＩＤとのフィールドを有する。ステートテーブル４１０は、各フィールドに情報を設定することにより、ステート情報がレコードとして記憶される。

[0050] ＩＤのフィールドには、イベントの発生元の人や物に対応する端末装置を識別するＩＤが設定される。ステート名のフィールドには、イベントの種別を示す情報としてステート名が設定される。ＩＤとステート名とのフィールドにより、プラグインの実行条件が設定される。ステート内容のフィールドには、イベントの内容を示す情報としてステート内容が設定される。プラグインＩＤのフィールドには、プラグインを識別する情報としてプラグインＩＤが設定される。プラグインＩＤは、例えば、プラグインのハッシュ値である。

[0051] ステートテーブル４１０は、例えば、ＩＤごとに生成される。情報処理装置は、例えば、ＩＤごとにステートテーブル４１０を生成し、ステートテーブル４１０のリスト４００を記憶する。また、以下の説明では、ステートテーブル４１０を、簡略化した図形４２０を用いて図示する場合がある。図４の例では、図形４２０は、例えば、ＩＤ「５」とステート名「Ｓ」とプラグインＩＤ「Ｂ」とを対応付けるステートテーブル４１０を表す。

[0052] （プラグインリポジトリテーブル５００の記憶内容）

次に、図５を用いて、プラグインリポジトリテーブル５００の記憶内容の一例について説明する。プラグインリポジトリテーブル５００は、例えば、図３に示した分散ノード２０１のメモリ３０２や記録媒体３０５などの記憶領域により実現される。

[0053] 図５は、プラグインリポジトリテーブル５００の記憶内容の一例を示す説明図である。図５に示すように、プラグインリポジトリテーブル５００は、プラグインＩＤと、参照数と、プラグインデータとのフィールドを有する。プラグインリポジトリテーブル５００は、各フィールドに情報を設定することにより、プラグイン情報がレコードとして記憶される。

[0054] プラグインＩＤのフィールドには、プラグインを識別する情報としてプラグインＩＤが設定される。参照数のフィールドには、ステートテーブル４１０に設定済みのプラグインの実行条件の数が設定される。プラグインデータのフィールドには、プラグインデータが設定される。プラグインデータは、実行形式のプラグインのコードである。

[0055] （管理ノード２０２のハードウェア構成例）

管理ノード２０２のハードウェア構成例は、例えば、図３に示した分散ノード２０１のハードウェア構成例と同様であるため、説明を省略する。

[0056] （端末装置２０３のハードウェア構成例）

端末装置２０３のハードウェア構成例は、図３に示した分散ノード２０１のハードウェア構成例と同様であるため、説明を省略する。端末装置２０３は、さらに、センサ装置を有してもよい。

[0057] センサ装置は、例えば、端末装置２０３のステートを検出する。センサ装置は、具体的には、端末装置２０３の位置、動き、および向きのうち少なくともいずれかを検出する。センサ装置は、具体的には、加速度センサ、地磁気センサ、光センサ、振動センサなどの少なくともいずれかにより実現される。センサ装置は、ＧＰＳ（Ｇｌｏｂａｌ Ｐｏｓｉｔｉｏｎｉｎｇ Ｓｙｓｔｅｍ）受信機により実現され、端末装置２０３のＧＰＳ座標を検出してもよい。

[0058] センサ装置は、例えば、生体情報を取得してもよい。センサ装置は、例えば、生体情報として、体温、脈拍、心拍、呼吸などに関する情報を取得する。センサ装置は、例えば、生体情報として、指紋、静脈、顔貌、声紋、掌形、網膜、虹彩などに関する情報を取得してもよい。センサ装置は、例えば、画像情報を取得してもよい。これにより、センサ装置は、ステートを検出することができる。

[0059] （開発ノード２０４のハードウェア構成例）

開発ノード２０４のハードウェア構成例は、例えば、図３に示した分散ノード２０１のハードウェア構成例と同様であるため、説明を省略する。

[0060] （情報処理装置１１０の機能的構成例）

次に、図６を用いて、情報処理装置１１０の機能的構成例について説明する。

[0061] 図６は、情報処理装置１１０の機能的構成例を示すブロック図である。情報処理装置１１０は、記憶部６００と、取得部６０１と、更新部６０２と、抽出部６０３と、実行部６０４と、出力部６０５とを含む。

[0062] 記憶部６００は、例えば、図３に示したメモリ３０２や記録媒体３０５などの記憶領域によって実現される。以下では、記憶部６００が、情報処理装置１１０に含まれる場合について説明するが、これに限らない。例えば、記憶部６００が、情報処理装置１１０とは異なる装置に含まれ、記憶部６００の記憶内容が情報処理装置１１０から参照可能である場合があってもよい。

[0063] 取得部６０１～出力部６０５は、制御部の一例として機能する。取得部６０１～出力部６０５は、具体的には、例えば、図３に示したメモリ３０２や記録媒体３０５などの記憶領域に記憶されたプログラムをＣＰＵ３０１に実行させることにより、または、ネットワークＩ／Ｆ３０３により、その機能を実現する。各機能部の処理結果は、例えば、図３に示したメモリ３０２や記録媒体３０５などの記憶領域に記憶される。

[0064] 記憶部６００は、各機能部の処理において参照され、または更新される各種情報を記憶する。記憶部６００は、プラグインの実行条件とプラグインのプラグインデータとを記憶する。プラグインは、メインプログラムから呼出可能なサブプログラムである。プラグインの実行条件は、特定の端末装置２０３での特定のイベントの発生である。プラグインの実行条件は、例えば、端末装置２０３のＩＤと、ステート名とにより表現される。イベントは、例えば、人や物のステートの変化である。ステートは、状態であり、例えば、速度である。プラグインデータは、実行形式のプラグインのコードである。プラグインデータは、例えば、符号化して記憶されてもよい。

[0065] 記憶部６００は、例えば、プラグインに対応付けて、プラグインの実行条件を、図４に示したステートテーブル４１０を用いて記憶する。記憶部６０

０は、具体的には、端末装置２０３のＩＤと、ステート名と、ステート内容と、プラグインＩＤとを対応付けて、図４に示したステートテーブル４１０を用いて記憶する。

[0066] 記憶部６００は、例えば、プラグインデータを、図５に示したプラグインリポジトリテーブル５００に集約する。記憶部６００は、具体的には、プラグインＩＤと、参照数と、プラグインデータとを対応付けて、図５に示したプラグインリポジトリテーブル５００に集約する。これにより、記憶部６００は、イベントに対応するプラグインを実行可能にすることができる。

[0067] 取得部６０１は、各機能部の処理に用いられる各種情報を取得する。取得部６０１は、取得した各種情報を、記憶部６００に記憶し、または、各機能部に出力する。また、取得部６０１は、記憶部６００に記憶しておいた各種情報を、各機能部に出力してもよい。取得部６０１は、例えば、利用者の操作入力に基づき、各種情報を取得する。取得部６０１は、例えば、自装置とは異なる装置から、各種情報を受信してもよい。

[0068] 取得部６０１は、第１制御メッセージを受信する。第１制御メッセージは、プラグインを追加することの要求である。第１制御メッセージは、例えば、追加するプラグインの実行条件と、プラグインデータとを対応付けて表す。第１制御メッセージは、例えば、管理ノード２０２から、情報処理装置１１０として動作する分散ノード２０１に送信される。第１制御メッセージは、送信後、管理ノード２０２に残されないことがある。

[0069] 取得部６０１は、第２制御メッセージを受信する。第２制御メッセージは、プラグインを削除することの要求である。第２制御メッセージは、削除するプラグインの実行条件と、プラグインデータとを対応付けて表す。第２制御メッセージは、例えば、管理ノード２０２から、情報処理装置１１０として動作する分散ノード２０１に送信される。

[0070] 取得部６０１は、イベントメッセージを受信する。イベントメッセージは、イベントと、イベントの発生元の人や物に対応する端末装置２０３とを表す。イベントメッセージは、例えば、端末装置２０３のＩＤと、ｋｅｙ－ｖ

ａｌｕｅペアとを含む。ｋｅｙは、イベントの種別を示す情報である。ｋｅｙは、例えば、ステート名に対応する。ｖａｌｕｅは、イベントの内容を示す情報である。ｖａｌｕｅは、例えば、ステート内容に対応する。

[0071] 取得部６０１は、定期的に発行される同期メッセージを受信する。同期メッセージは、例えば、バリアメッセージである。バリアメッセージは、記憶部６００の記憶内容をバックアップする指示を表す。バリアメッセージは、具体的には、分散チェックポイントによる障害復旧のためのバックアップの指示を表す。バックアップは、具体的には、チェックポイントとして外部記憶装置２０５に保存することである。バリアメッセージは、例えば、管理ノード２０２から送信される。

[0072] 取得部６０１は、他の情報処理装置１１０が所定の端末装置２０３に提供していた所定の機能を引き継ぐことの要求を受信する。以下の説明では、所定の機能を引き継ぐことの要求を「引継要求」と表記する場合がある。引継要求は、例えば、所定の端末装置２０３の識別子を含む。識別子は、例えば、ＩＤである。引継要求は、具体的には、情報処理装置１１０に割り振る１以上の機能のそれぞれの機能を実現するプラグインの実行条件を含む。プラグインの実行条件は、機能を提供する相手となる端末装置２０３のＩＤを含む。引継要求は、例えば、管理ノード２０２から送信される。

[0073] 更新部６０２は、プラグインを記憶部６００に記憶、または、プラグインを記憶部６００から削除する。更新部６０２は、例えば、第１制御メッセージを受信した場合、受信した第１制御メッセージから特定される、プラグインの実行条件とプラグインデータとを対応付けて、記憶部６００に記憶する。更新部６０２は、具体的には、受信した第１制御メッセージから、端末装置２０３のＩＤと、ステート名と、プラグインデータとを特定し、プラグインＩＤと、参照数を算出する。プラグインＩＤは、プラグインデータのハッシュ値である。そして、更新部６０２は、端末装置２０３のＩＤと、ステート名と、プラグインＩＤとを対応付けたステート情報を、図４に示したステートテーブル４１０を用いて記憶する。

[0074] また、更新部６０２は、例えば、プラグインＩＤと、参照数と、プラグインデータとを対応付けたプラグイン情報を、図５に示したプラグインリポジトリテーブル５００に記憶する。これにより、更新部６０２は、第１制御メッセージを受信したことに応じて処理プログラムに新たなプラグインを追加することができ、新たなプラグインにより実現される新たな機能を端末装置２０３に提供することができる。

[0075] 更新部６０２は、例えば、第２制御メッセージを受信した場合、受信した第２制御メッセージから特定されるプラグインの実行条件に対応付けて記憶部６００に記憶されたプラグインデータを削除する。更新部６０２は、具体的には、受信した第２制御メッセージから、端末装置２０３のＩＤと、ステート名と、プラグインデータとを特定し、プラグインＩＤを算出する。そして、更新部６０２は、端末装置２０３のＩＤと、ステート名と、プラグインＩＤとを対応付けているステート情報を、図４に示したステートテーブル４１０から削除する。また、更新部６０２は、削除に応じて参照数をデクリメントする。

[0076] ここで、更新部６０２は、参照数が０になれば、プラグインＩＤと、０になった参照数と、特定したプラグインデータとを対応付けているプラグイン情報を、図５に示したプラグインリポジトリテーブル５００から削除する。これにより、更新部６０２は、処理プログラムからプラグインを削除することができる。

[0077] 更新部６０２は、同期メッセージを受信すると、記憶部６００の記憶内容をバックアップする。バックアップは、具体的には、チェックポイントとして外部記憶装置２０５に保存することである。更新部６０２は、例えば、情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインの識別子とプラグインデータとを対応付けた対応情報を集約したバックアップデータを、外部記憶装置２０５に格納する。プラグインの識別子は、例えば、プラグインＩＤである。

[0078] バックアップデータは、例えば、情報処理装置１１０が提供していた１以

上の機能のそれぞれの機能を実現するプラグインのプラグインデータを、所定のデータ形式に沿って区切って集約される。バックアップデータは、例えば、情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインのプラグインデータを、暗号化して集約される。更新部６０２は、具体的には、プラグインリポジトリテーブル５００のスナップショットを、チェックポイント番号と、自装置の識別子とに対応付けて、外部記憶装置２０５に保存する。

[0079] また、更新部６０２は、例えば、情報処理装置１１０が提供していた１以上の機能のそれぞれの機能について、機能を実現するプラグインのプラグインＩＤと、機能が提供される端末装置２０３のＩＤとを対応付けた対応情報を、外部記憶装置２０５に格納する。更新部６０２は、具体的には、ステートテーブル４１０を、チェックポイント番号と、自装置の識別子と、ステートテーブル４１０のＩＤとに対応付けて、外部記憶装置２０５に保存する。

[0080] 更新部６０２は、具体的には、ステートテーブル４１０とプラグインリポジトリテーブル５００のスナップショットとを保存すると、チェックポイント番号をインクリメントする。これにより、更新部６０２は、バックアップを実施し、ステートテーブル４１０とプラグインリポジトリテーブル５００とを取得可能にすることができる。

[0081] また、更新部６０２は、同期メッセージに関わらず、定期的に記憶部６００の記憶内容をバックアップしてもよい。更新部６０２は、具体的には、所定時間が経過する都度、ステートテーブル４１０とプラグインリポジトリテーブル５００のスナップショットとを、外部記憶装置２０５に保存する。これにより、更新部６０２は、バックアップを実施し、ステートテーブル４１０とプラグインリポジトリテーブル５００とを取得可能にすることができる。

[0082] 抽出部６０３は、外部記憶装置２０５の記憶内容を取得する。抽出部６０３は、引継要求を受け付けた場合、外部記憶装置２０５から、引き継ぐ機能を提供していた他の情報処理装置１１０によって格納されたバックアップデ

ータを取得する。抽出部６０３は、例えば、外部記憶装置２０５から、引き継ぐ機能を提供していた他の情報処理装置１１０によって格納された、第１の対応情報を集約したバックアップデータを取得する。第１の対応情報は、他の情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインのプラグインＩＤとプラグインデータとを対応付ける。第１の対応情報は、例えば、プラグインリポジトリテーブル５００のスナップショットである。

[0083] 抽出部６０３は、具体的には、外部記憶装置２０５から、引き継ぐ機能を提供していた他の情報処理装置１１０の識別子と、直近のチェックポイント番号との組み合わせに対応付けられたリポジトリテーブルのスナップショットを取得する。これにより、抽出部６０３は、引き継ぐ機能を実現するプラグインのプラグインデータを取得可能にすることができる。

[0084] また、抽出部６０３は、外部記憶装置２０５から、自装置によって格納されたバックアップデータを取得する。抽出部６０３は、具体的には、外部記憶装置２０５から、自装置の識別子と、直近のチェックポイント番号との組み合わせに対応付けられたリポジトリテーブルのスナップショットを取得する。これにより、抽出部６０３は、自装置が提供していた機能を実現するプラグインのプラグインデータを取得可能にすることができる。

[0085] また、抽出部６０３は、引き継ぐ機能を提供していた他の情報処理装置１１０によって格納された、プラグインＩＤと、端末装置２０３のＩＤとを対応付けた対応情報を取得する。抽出部６０３は、例えば、引き継ぐ機能を提供していた他の情報処理装置１１０によって格納された第２の対応情報を取得する。第２の対応情報は、他の情報処理装置１１０が提供していた１以上の機能のそれぞれの機能について、機能を実現するプラグインのプラグインＩＤと、機能を提供する端末装置２０３のＩＤとを対応付ける。第２の対応情報は、例えば、ステートテーブル４１０またはステートテーブル４１０のステート情報である。

[0086] 抽出部６０３は、具体的には、外部記憶装置２０５に保存された、直近の

チェックポイント番号と、引き継ぐ機能を提供していた他の情報処理装置１１０の識別子と、引き継ぐ機能を提供する端末装置２０３のＩＤとの組み合わせに対応するステート情報を取得する。これにより、抽出部６０３は、自装置に割り振られた機能のうち引き継ぐ機能についてのステート情報を、ステートテーブル４１０に反映可能にすることができる。

[0087] また、抽出部６０３は、自装置によって格納された、プラグインＩＤと、端末装置２０３のＩＤとを対応付けた対応情報を取得する。抽出部６０３は、具体的には、外部記憶装置２０５に保存された、直近のチェックポイント番号と、自装置の識別子と、自装置に割り振られた機能を提供する端末装置２０３のＩＤとの組み合わせに対応するステート情報を取得する。これにより、抽出部６０３は、自装置に割り振られた機能についてのステート情報を、ステートテーブル４１０に反映可能にすることができる。

[0088] 抽出部６０３は、取得したステート情報を、ステートテーブル４１０に反映する。抽出部６０３は、例えば、取得したステート情報を、ステートテーブル４１０のステート情報に上書き、または、ステートテーブル４１０に追加する。抽出部６０３は、ステートテーブル４１０を初期化し、取得したステート情報に基づいてステートテーブル４１０を再作成してもよい。これにより、抽出部６０３は、ステートテーブル４１０に、引き継ぐ機能についてのステート情報を反映することができる。また、抽出部６０３は、ステートテーブル４１０のステート情報を、バックアップの時点の状態に戻すことができる。

[0089] 抽出部６０３は、バックアップデータを解釈し、プラグインデータを抽出し、記憶部６００に記憶する。抽出部６０３は、例えば、取得したバックアップデータを解釈し、他の情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインのプラグインデータのうち、所定の機能を実現するプラグインのプラグインデータを抽出する。抽出部６０３は、具体的には、取得したバックアップデータを解釈し、取得した第２の対応情報を参照し、引継要求に含まれる所定の端末装置２０３のＩＤに基づいて

、第１の対応情報のうち、所定の機能を実現するプラグインのプラグインデータを抽出する。

[0090] 抽出部６０３は、より具体的には、取得したバックアップデータを解釈し、他の情報処理装置１１０が有していたリポジトリテーブルを生成する。抽出部６０３は、ステート情報を反映したステートテーブル４１０を参照し、引継要求に含まれる所定の端末装置２０３のＩＤと、ステート名とに対応付けられたプラグインＩＤを特定する。抽出部６０３は、他の情報処理装置１１０が有していたプラグインリポジトリテーブル５００のうち、特定したプラグインＩＤに対応付けられたプラグインデータを抽出する。抽出部６０３は、抽出したプラグインデータを、自装置のプラグインリポジトリテーブル５００に設定する。これにより、抽出部６０３は、引き継ぐ所定の機能を実現可能にすることができる。

[0091] また、抽出部６０３は、例えば、取得したバックアップデータを解釈し、自装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのプラグインデータを抽出し、記憶部６００に記憶してもよい。

[0092] 抽出部６０３は、具体的には、取得したバックアップデータを解釈し、自装置が有していた過去のプラグインリポジトリテーブル５００を生成する。抽出部６０３は、ステート情報を反映したステートテーブル４１０を参照し、引継要求に含まれる端末装置２０３のＩＤと、ステート名とに対応付けられたプラグインＩＤを特定する。抽出部６０３は、自装置が有していた過去のプラグインリポジトリテーブル５００のうち、特定したプラグインＩＤに対応付けられたプラグインデータを抽出する。抽出部６０３は、抽出したプラグインデータを、自装置のプラグインリポジトリテーブル５００に設定する。これにより、抽出部６０３は、自装置に割り振られた機能を実現可能にすることができる。

[0093] 抽出部６０３は、外部記憶装置２０５から、それぞれの情報処理装置１１０によって格納されたバックアップデータを取得してもよい。抽出部６０３は、取得したバックアップデータを解釈し、それぞれの情報処理装置１１０

が提供していた１以上の機能のそれぞれの機能を実現するプラグインのプラグインデータを、重複するプラグインのプラグインデータを統合して纏めた一覧情報を生成する。抽出部６０３は、一覧情報の中から、所定の機能を実現するプラグインのプラグインデータを抽出し、記憶部６００に記憶する。

[0094] 抽出部６０３は、例えば、一覧情報の中から、自装置に割り振られた１以上の機能のそれぞれの機能を実現するプラグインのプラグインデータを抽出し、記憶部６００に記憶する。これにより、抽出部６０３は、自装置に割り振られた機能を実現可能にすることができる。

[0095] 実行部６０４は、記憶部６００に記憶されたプラグインデータに基づいて、プラグインを実行する。実行部６０４は、例えば、イベントメッセージを受信した場合、受信したイベントメッセージから特定されるイベントに対応付けて記憶部６００に記憶されたプラグインデータに基づいて、プラグインを実行する。

[0096] 実行部６０４は、具体的には、ステートテーブル４１０を参照し、イベントメッセージの端末装置２０３のＩＤと、ｋｅｙ－ｖａｌｕｅペアのｋｅｙに対応するステート名とに対応付けられたプラグインＩＤを特定する。実行部６０４は、プラグインリポジトリテーブル５００を参照し、プラグインＩＤに対応付けられたプラグインデータに基づいて、プラグインを実行する。これにより、実行部６０４は、イベントメッセージを処理することができる。また、実行部６０４は、抽出部６０３が抽出したプラグインデータが、記憶部６００に記憶されているため、引き継いだ所定の機能を実現することができ、所定の機能を所定の端末装置２０３に提供することができる。

[0097] 出力部６０５は、プラグインの実行結果を出力する。出力形式は、例えば、ディスプレイへの表示、プリンタへの印刷出力、ネットワークＩ／Ｆ３０３による外部装置への送信、または、メモリ３０２や記録媒体３０５などの記憶領域への記憶である。これにより、出力部６０５は、プラグインの実行結果を利用可能にすることができる。

[0098] 出力部６０５は、いずれかの機能部の処理結果を出力してもよい。これに

より、出力部６０５は、いずれかの機能部の処理結果を利用者に通知可能にし、情報処理装置１１０の管理や運用、例えば、情報処理装置１１０の設定値の更新などを支援することができ、情報処理装置１１０の利便性の向上を図ることができる。

[0099] （並列分散処理システム２００の動作例）

次に、図７～図１５を用いて、並列分散処理システム２００の動作例について説明する。まず、図７～図９を用いて、分散ノード２０１の基本動作について説明する。

[0100] ここで、図７の説明に移行し、並列分散処理システム２００において、分散ノード２０１がプラグインを追加される際、ステートテーブル４１０を更新する一例について説明する。

[0101] 図７は、ステートテーブル４１０を更新する一例を示す説明図である。図７において、開発ノード２０４は、サービス開発者の操作入力に基づき、制御メッセージを、管理ノード２０２を介して、分散ノード２０１に送信する。図７の例では、制御メッセージは、プラグイン追加を表し、プラグインＤを分散ノード２０１に追加することの要求である。

[0102] 分散ノード２０１は、制御メッセージを受信すると、制御メッセージに基づいて、プラグインＤの実行条件と、プラグインＤのプラグインＩＤとを対応付けたステート情報を、ステートテーブル４１０に追加する。図７の例では、実行条件は、ＩＤ「Ｘ」の端末装置２０３におけるイベントである。イベントは、例えば、運転時間の変化である。このため、イベントの種別を示すステート名は、運転時間である。

[0103] これにより、分散ノード２０１は、所定の端末装置２０３におけるステートとプラグインとを対応付けて、ステートテーブル４１０を更新することができる。そして、分散ノード２０１は、所定の端末装置２０３におけるステートの変化に対応した、プラグインによるサービスを実装することができる。サービスは、例えば、速度の警告、長時間運転の警告などである。

[0104] また、分散ノード２０１は、プラグインデータを、プラグインリポジトリ

テーブル５００に記憶して管理する。図７の例では、分散ノード２０１は、プラグインＤのプラグインデータを、プラグインリポジトリテーブル５００に記憶する。

[0105] 次に、図８の説明に移行し、分散ノード２０１が、プラグインリポジトリテーブル５００によりプラグインデータを管理する一例について説明する。

[0106] 図８は、プラグインデータを管理する一例を示す説明図である。図８に示すように、分散ノード２０１は、プラグインＩＤと、参照数と、プラグインデータとを対応付けたプラグイン情報を、プラグインリポジトリテーブル５００に記憶し、プラグインデータを管理する。参照数は、ステートテーブル４１０のリスト４００全体で、プラグインＩＤに対応付けられた実行条件の数を計数して得られる。プラグインリポジトリテーブル５００は、分散ノード２０１ごとに１つ記憶される。

[0107] 次に、図９の説明に移行し、分散ノード２０１が、ステートテーブル４１０とプラグインリポジトリテーブル５００とをバックアップする一例について説明する。

[0108] 図９は、ステートテーブル４１０とプラグインリポジトリテーブル５００とをバックアップする一例を示す説明図である。図９において、分散ノード２０１は、一定時間ごと、または、管理ノード２０２の指示により、ステートテーブル４１０のリスト４００と、プラグインリポジトリテーブル５００のスナップショットとを、外部記憶装置２０５に保存する。これにより、分散ノード２０１は、分散ノード２０１間において機能を引継可能にすることができる。

[0109] 次に、図１０～図１５の説明に移行し、並列分散処理システム２００に障害が発生した場合の、分散ノード２０１の動作について説明する。

[0110] 図１０～図１５は、障害発生時の分散ノード２０１の動作を示す説明図である。図１０において、分散ノード２０１－２において障害が発生し、機能を提供することができなくなる。このため、分散ノード２０１－１と分散ノード２０１－３とで、分散ノード２０１－２が提供していた機能を引き継ぐ

ことが望まれる。

[0111] このため、管理ノード２０２は、分散ノード２０１－１に、分散ノード２０１－２が提供していた、端末装置２０３のＩＤ「３」に対応する機能を引き継ぐことの要求を送信する。また、管理ノード２０２は、分散ノード２０１－３に、分散ノード２０１－２が提供していた、端末装置２０３のＩＤ「４」に対応する機能を引き継ぐことの要求を送信する。ここで、図１１の説明に移行し、要求を受け付けた分散ノード２０１－３の動作について説明する。

[0112] 図１１において、分散ノード２０１－３は、Ｓｔｅｐ１として、外部記憶装置２０５から、引き継ぐ機能を含む自装置に割り振られた１以上の機能のそれぞれの機能を実現するプラグインに関するステート情報を取得する。分散ノード２０１－３は、取得したステート情報から、ステートテーブル４１０のリスト４００を再生成する。Ｓｔｅｐ１の具体例は、図１２を用いて後述する。

[0113] 分散ノード２０１－３は、Ｓｔｅｐ２として、再生成したステートテーブル４１０のリスト４００に基づいて、プラグインごとの参照数を算出し、プラグインリポジトリテーブル５００を再生成する。Ｓｔｅｐ２の具体例は、図１３を用いて後述する。

[0114] 分散ノード２０１－３は、Ｓｔｅｐ３として、外部記憶装置２０５から、それぞれの分散ノード２０１が格納したプラグインリポジトリテーブル５００のスナップショット１１０１～１１０３を取得し、プラグイン情報を纏めた一覧情報を生成する。Ｓｔｅｐ３の具体例は、図１４を用いて後述する。

[0115] 分散ノード２０１－３は、Ｓｔｅｐ４として、プラグインリポジトリテーブル５００のうち、プラグインデータが空であるプラグイン情報があれば、一覧情報からプラグインデータを取得して設定する。Ｓｔｅｐ４の具体例は、図１５を用いて後述する。ここで、図１２の説明に移行する。

[0116] 図１２において、分散ノード２０１－３は、外部記憶装置２０５から、自装置に割り振られていた端末装置２０３のＩＤ「５，６」に対応するステー

ト情報を取得する。また、分散ノード２０１－３は、引き継ぐ機能を実現するプラグインに関する、端末装置２０３のＩＤ「４」に対応するステート情報を取得する。分散ノード２０１－３は、取得したステート情報に基づいて、ステートテーブル４１０のリスト４００を再生成する。次に、図１３の説明に移行する。

[0117] 図１３において、分散ノード２０１－３は、再生成したステートテーブル４１０のリスト４００に基づいて、プラグインＩＤ「Ｂ」に対応する参照数「２」を算出し、プラグインリポジトリテーブル５００に設定する。分散ノード２０１－３は、例えば、プラグインリポジトリテーブル５００に、既に、プラグインＩＤ「Ｂ」に対応するプラグイン情報があるため、プラグイン情報に参照数「２」を上書きする。

[0118] また、分散ノード２０１－３は、再生成したステートテーブル４１０のリスト４００に基づいて、プラグインＩＤ「Ｃ」に対応する参照数「１」を算出し、プラグインリポジトリテーブル５００に設定する。分散ノード２０１－３は、例えば、プラグインリポジトリテーブル５００に、まだプラグインＩＤ「Ｃ」に対応するプラグイン情報がないため、プラグインＩＤ「Ｃ」と参照数「１」とを対応付けたプラグイン情報を追加する。

[0119] また、分散ノード２０１－３は、再生成したステートテーブル４１０のリスト４００に基づいて、プラグインＩＤ「Ｄ」に対応する参照数「１」を算出し、プラグインリポジトリテーブル５００に設定する。分散ノード２０１－３は、例えば、プラグインリポジトリテーブル５００に、既に、プラグインＩＤ「Ｄ」に対応するプラグイン情報があるため、プラグイン情報に参照数「１」を上書きする。ここで、分散ノード２０１－３は、プラグインＩＤ「Ｃ」のプラグインデータを有していない。次に、図１４の説明に移行する。

[0120] 図１４において、分散ノード２０１－３は、外部記憶装置２０５から、それぞれの分散ノード２０１が格納したプラグインリポジトリテーブル５００のスナップショット１１０１～１１０３を取得し、プラグイン情報を纏めた

一覧情報を生成する。分散ノード２０１－３は、自装置で有していないプラグインＩＤ「Ｃ」のプラグインデータを、一覧情報から探索する。次に、図１５の説明に移行する。

[0121] 図１５において、分散ノード２０１－３は、プラグインＩＤ「Ｃ」のプラグインデータを発見すると、プラグインリポジトリテーブル５００のプラグインＩＤ「Ｃ」に対応するプラグイン情報に設定する。これにより、分散ノード２０１－３は、分散ノード２０１－２が提供していた機能を実現するプラグインのプラグインデータを取得することができ、分散ノード２０１－２が提供していた機能を引き継ぐことができる。

[0122] ここでは、並列分散処理システム２００に障害が発生したことに応じて、分散ノード２０１が、他の分散ノード２０１が提供していた機能を引き継ぐ場合について説明したが、これに限らない。例えば、並列分散処理システム２００に分散ノード２０１が増設されたことに応じて、分散ノード２０１が、他の分散ノード２０１が提供していた機能を引き継ぐ場合があってもよい。

[0123] ここでは、分散ノード２０１－３が、障害発生前のプラグインリポジトリテーブル５００を流用し、プラグインリポジトリテーブル５００に設定されたプラグインデータを利用する場合について説明したが、これに限らない。例えば、分散ノード２０１－３が、障害発生前のプラグインリポジトリテーブル５００を削除し、プラグインリポジトリテーブル５００を再生成する場合があってもよい。

[0124] （並列分散処理システム２００の適用例）

次に、図１６～図２０を用いて、並列分散処理システム２００の適用例について説明する。

[0125] 図１６は、並列分散処理システム２００の適用例を示す説明図である。図１６に示すように、並列分散処理システム２００は、例えば、フレームワークとしてＡｐａｃｈｅ　Ｆｌｉｎｋを利用する。並列分散処理システム２００は、上述した分散ノード２０１に対応する計算機１６０１－１～１６０１

－Ｎと、マスタノードである計算機１６０２と、上述した管理ノード２０２に対応する計算機１６０３とを含む。以下の説明では、計算機１６０１－１～１６０１－Ｎを区別せずに単に「計算機１６０１」と表記する場合がある。

[0126] 計算機１６０１は、Ｆｌｉｎｋ Ｓｌａｖｅ Ｒｕｎｔｉｍｅを動作させ、分散ノードソフトウェアを動作させる。計算機１６０１は、Ａｐａｃｈｅ Ｆｌｉｎｋのジョブとして、上述した動作例に示したような動作を実現する。計算機１６０２は、Ｆｌｉｎｋ Ｓｌａｖｅ Ｒｕｎｔｉｍｅを動作させる。計算機１６０３は、メッセージキューを有し、イベントメッセージを受け付け、計算機１６０１に送信する。

[0127] 計算機１６０２間は、例えば、クラスタ間通信ネットワーク１６１０により接続される。計算機１６０２と計算機１６０３とは、例えば、クラスタ間通信ネットワーク１６１０により接続される。次に、図１７の説明に移行する。

[0128] 図１７は、適用例におけるメッセージフローを示す説明図である。図１７に示すように、計算機１６０３からイベントメッセージや制御メッセージが、計算機１６０１に送信される。計算機１６０１は、文字列解釈ブロック１７０１と仮想オブジェクトブロック１７０２とを有する。

[0129] 文字列解釈ブロック１７０１は、受け付けたイベントメッセージや制御メッセージの文字列を解釈し、受け付けたイベントメッセージや制御メッセージの宛先を判別する。そして、文字列解釈ブロック１７０１は、受け付けたイベントメッセージや制御メッセージを、自装置の仮想オブジェクトブロック１７０２、または他装置の仮想オブジェクトブロック１７０２に送信する。

[0130] 仮想オブジェクトブロック１７０２は、イベントメッセージや制御メッセージを処理する。仮想オブジェクトブロック１７０２は、具体的には、上述した動作例に示したような動作を実現するように、イベントメッセージや制御メッセージを処理する。次に、図１８および図１９の説明に移行し、上述

した動作例に示したような動作を実現する、具体的な機能的構成例について説明する。

[0131] 図１８および図１９は、適用例における計算機１６０１の具体的な機能的構成例を示す説明図である。図１８に示すように、文字列解釈ブロック１７０１は、文字列解釈モジュール１８０１と、宛先決定モジュール１８０２とを有する。

[0132] 文字列解釈ブロック１７０１は、受け付けたイベントメッセージや制御メッセージのＪＳＯＮ形式の文字列を内部表現に変換し、宛先決定モジュール１８０２に出力する。宛先決定モジュール１８０２は、内部表現に基づいて、受け付けたイベントメッセージや制御メッセージの宛先を判別し、自装置の仮想オブジェクトブロック１７０２、または他装置の仮想オブジェクトブロック１７０２に送信する。次に、図１９の説明に移行する。

[0133] 図１９に示すように、仮想オブジェクトブロック１７０２は、ステートテーブル１９２１のリスト１９２０と、プラグインリポジトリテーブル１９３０とを有する。ステートテーブル１９２１は、図４に示したステートテーブル４１０と同様である。プラグインリポジトリテーブル１９３０は、図５に示したプラグインリポジトリテーブル５００と同様である。

[0134] 仮想オブジェクトブロック１７０２は、メッセージ判別部１９１１と、制御メッセージ処理部１９１２と、イベントメッセージ処理部１９１３と、チェックポイント管理部１９１４と、プラグインリポジトリ再生成部１９１５とを有する。

[0135] メッセージ判別部１９１１は、イベントメッセージ１９０１、プラグインを追加または削除する制御メッセージ１９０２のいずれかのメッセージの入力を受け付ける。メッセージ判別部１９１１は、入力されたメッセージの種別を判別し、制御メッセージ処理部１９１２またはイベントメッセージ処理部１９１３に引き渡す。

[0136] 制御メッセージ処理部１９１２は、プラグインを追加または削除する制御メッセージ１９０２を解釈する。制御メッセージ処理部１９１２は、例えば

、プラグインを追加する制御メッセージ１９０２から、ＩＤと、対象ステート名と、追加するプラグインのプラグインデータとを抽出する。制御メッセージ処理部１９１２は、抽出した情報に基づいて、ステートテーブル１９２１のリスト１９２０と、プラグインリポジトリテーブル１９３０とを更新する。

[0137] 制御メッセージ処理部１９１２は、例えば、プラグインを削除する制御メッセージ１９０２から、ＩＤと、対象ステート名と、削除するプラグインのプラグインデータとを抽出する。制御メッセージ処理部１９１２は、抽出した情報に基づいて、ステートテーブル１９２１のリスト１９２０と、プラグインリポジトリテーブル１９３０とを更新する。

[0138] イベントメッセージ処理部１９１３は、イベントメッセージ１９０１を解釈し、ＩＤと、１以上のｋｅｙ－ｖａｌｕｅペアを抽出する。イベントメッセージ処理部１９１３は、抽出したＩＤに対応するステートテーブル１９２１を特定する。イベントメッセージ処理部１９１３は、特定したステートテーブル１９２１を、受け付けたｋｅｙ－ｖａｌｕｅペアで更新する。イベントメッセージ処理部１９１３は、いずれかのステート内容が変更された場合、ステートが変化したと判定する。イベントメッセージ処理部１９１３は、ステートが変化すると、プラグインデータに基づいて、プラグインを実行する。

[0139] チェックポイント管理部１９１４は、一定時間ごとに、チェックポイント番号をインクリメントする。チェックポイント管理部１９１４は、ステートテーブル１９２１のリスト１９２０と、プラグインリポジトリテーブル１９３０のスナップショットとを、チェックポイント番号と、自装置の識別子と対応付けて、外部記憶装置２０５に保存する。

[0140] プラグインリポジトリ再生成部１９１５は、外部記憶装置２０５に保存された、ステートテーブル１９２１のリスト１９２０と、プラグインリポジトリテーブル１９３０のスナップショットを取得する。プラグインリポジトリ再生成部１９１５は、取得した情報に基づいて、ステートテーブル１９２１

のリスト１９２０と、プラグインリポジトリテーブル１９３０のスナップショットを再生成する。次に、図２０の説明に移行し、各種メッセージの具体例について説明する。

[0141] 図２０は、各種メッセージの一例を示す説明図である。イベントメッセージ１９０１は、ＩＤと、１以上のステートとのフィールドを有する。ステートは、ｋｅｙ－ｖａｌｕｅペアである。ＩＤのフィールドには、イベントの発生元の端末装置２０３を識別する情報としてＩＤが設定される。ｋｅｙ－ｖａｌｕｅペアのフィールドには、ｋｅｙ－ｖａｌｕｅペアが設定される。ｋｅｙは、ステート名に対応し、イベントの種別を示す情報である。ｖａｌｕｅは、ステート内容に対応し、ステートの状態を示す特徴量である。ステートは、例えば、速度である。特徴量は、例えば、速度の値である。

[0142] 制御メッセージ１９０２は、ＩＤと、対象ステート名と、操作種別と、プラグインデータとのフィールドを有する。ＩＤのフィールドには、イベントの発生元の端末装置２０３を識別する情報としてＩＤが設定される。対象ステート名のフィールドには、プラグインの実行条件となるイベントの種別を示す情報として対象ステート名が設定される。操作種別のフィールドには、メッセージの種別として「プラグイン追加」または「プラグイン削除」が設定される。プラグインデータのフィールドには、プラグインの実行形式データであるプラグインデータが設定される。

[0143] （全体処理手順）

次に、図２１を用いて、分散ノード２０１が実行する、全体処理手順の一例について説明する。全体処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0144] 図２１は、全体処理手順の一例を示すフローチャートである。図２１において、分散ノード２０１は、メッセージの到着まで待機する（ステップＳ２１０１）。

[0145] 次に、分散ノード２０１は、受け付けたメッセージが制御メッセージであ

るか否かを判定する（ステップＳ２１０２）。ここで、制御メッセージではなく、イベントメッセージである場合（ステップＳ２１０２：Ｎｏ）、分散ノード２０１は、ステップＳ２１０３の処理に移行する。一方で、制御メッセージである場合（ステップＳ２１０２：Ｙｅｓ）、分散ノード２０１は、ステップＳ２１０４の処理に移行する。

[0146] ステップＳ２１０３では、分散ノード２０１は、図２２に後述するイベントメッセージ処理を実行する（ステップＳ２１０３）。そして、分散ノード２０１は、ステップＳ２１０１の処理に戻る。

[0147] ステップＳ２１０４では、分散ノード２０１は、図２５に後述する制御系イベント処理を実行する（ステップＳ２１０４）。そして、分散ノード２０１は、ステップＳ２１０１の処理に戻る。

[0148] （イベントメッセージ処理手順）

次に、図２２を用いて、分散ノード２０１が実行する、イベントメッセージ処理手順の一例について説明する。イベントメッセージ処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0149] 図２２は、イベントメッセージ処理手順の一例を示すフローチャートである。図２２において、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、まだ選択していないステートテーブル４１０を選択する（ステップＳ２２０１）。

[0150] 次に、分散ノード２０１は、選択したステートテーブル４１０のＩＤが、受け付けたイベントメッセージのＩＤと一致するか否かを判定する（ステップＳ２２０２）。ここで、一致する場合（ステップＳ２２０２：Ｙｅｓ）、分散ノード２０１は、ステップＳ２２０６の処理に移行する。一方で、一致しない場合（ステップＳ２２０２：Ｎｏ）、分散ノード２０１は、ステップＳ２２０３の処理に移行する。

[0151] ステップＳ２２０３では、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、すべてのステートテーブル４１０を選択したか否か

を判定する（ステップＳ２２０３）。ここで、未選択のステートテーブル４１０がある場合（ステップＳ２２０３：Ｎｏ）、分散ノード２０１は、ステップＳ２２０１の処理に戻る。一方で、すべてのステートテーブル４１０を選択している場合（ステップＳ２２０３：Ｙｅｓ）、分散ノード２０１は、ステップＳ２２０４の処理に移行する。

[0152] ステップＳ２２０４では、分散ノード２０１は、新しいステートテーブル４１０を生成し、ステートテーブル４１０のリスト４００に登録する（ステップＳ２２０４）。次に、分散ノード２０１は、登録したステートテーブル４１０のＩＤに、イベントメッセージのＩＤを設定する（ステップＳ２２０５）。そして、分散ノード２０１は、ステップＳ２２０６の処理に移行する。

[0153] ステップＳ２２０６では、分散ノード２０１は、図２３に後述するステート更新処理を実行する（ステップＳ２２０６）。そして、分散ノード２０１は、イベントメッセージ処理を終了する。

[0154] （ステート更新処理手順）

次に、図２３を用いて、分散ノード２０１が実行する、ステート更新処理手順の一例について説明する。ステート更新処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0155] 図２３は、ステート更新処理手順の一例を示すフローチャートである。図２３において、分散ノード２０１は、イベントメッセージのうち、まだ選択していないｋｅｙ－ｖａｌｕｅペアを選択する（ステップＳ２３０１）。次に、分散ノード２０１は、ステートテーブル４１０のうち、まだ選択していないステート情報を選択する（ステップＳ２３０２）。

[0156] そして、分散ノード２０１は、選択したｋｅｙ－ｖａｌｕｅペアのｋｅｙと、選択したステート情報のステート名とが一致するか否かを判定する（ステップＳ２３０３）。ここで、一致する場合（ステップＳ２３０３：Ｙｅｓ）、分散ノード２０１は、ステップＳ２３０６の処理に移行する。一方で、

一致しない場合（ステップＳ２３０３：Ｎｏ）、分散ノード２０１は、ステップＳ２３０４の処理に移行する。

[0157] ステップＳ２３０４では、分散ノード２０１は、ステートテーブル４１０のうち、すべてのステート情報を選択したか否かを判定する（ステップＳ２３０４）。ここで、未選択のステート情報がある場合（ステップＳ２３０４：Ｎｏ）、分散ノード２０１は、ステップＳ２３０２の処理に戻る。一方で、すべてのステート情報を選択している場合（ステップＳ２３０４：Ｙｅｓ）、分散ノード２０１は、ステップＳ２３０５の処理に移行する。

[0158] ステップＳ２３０５では、分散ノード２０１は、ステートテーブル４１０に、選択したｋｅｙ－ｖａｌｕｅペアのｋｅｙを、ステート名として設定したステート情報を追加する（ステップＳ２３０５）。次に、分散ノード２０１は、選択したｋｅｙ－ｖａｌｕｅペアのｖａｌｕｅを、ステート内容として、ステート情報に設定する（ステップＳ２３０６）。

[0159] そして、分散ノード２０１は、ステート情報にプラグインＩＤが含まれるか否かを判定する（ステップＳ２３０７）。ここで、ステート情報にプラグインＩＤが含まれない場合（ステップＳ２３０７：Ｎｏ）、分散ノード２０１は、ステップＳ２３０９の処理に移行する。一方で、ステート情報にプラグインＩＤが含まれる場合（ステップＳ２３０７：Ｙｅｓ）、分散ノード２０１は、ステップＳ２３０８の処理に移行する。

[0160] ステップＳ２３０８では、分散ノード２０１は、図２４に後述するプラグイン処理を実行する（ステップＳ２３０８）。そして、分散ノード２０１は、ステップＳ２３０９の処理に移行する。

[0161] ステップＳ２３０９では、分散ノード２０１は、イベントメッセージのうち、すべてのｋｅｙ－ｖａｌｕｅペアを選択したか否かを判定する（ステップＳ２３０９）。ここで、未選択のｋｅｙ－ｖａｌｕｅペアがある場合（ステップＳ２３０９：Ｎｏ）、分散ノード２０１は、ステップＳ２３０１の処理に戻る。一方で、すべてのｋｅｙ－ｖａｌｕｅペアを選択している場合（ステップＳ２３０９：Ｙｅｓ）、分散ノード２０１は、ステート更新処理を

終了する。

[0162] （プラグイン処理手順）

次に、図２４を用いて、分散ノード２０１が実行する、プラグイン処理手順の一例について説明する。プラグイン処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0163] 図２４は、プラグイン処理手順の一例を示すフローチャートである。図２４において、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、まだ選択していないプラグイン情報を選択する（ステップＳ２４０１）。

[0164] 次に、分散ノード２０１は、選択したプラグイン情報のプラグインＩＤが、実行対象のプラグインＩＤと一致するか否かを判定する（ステップＳ２４０２）。ここで、一致しない場合（ステップＳ２４０２：Ｎｏ）、分散ノード２０１は、ステップＳ２４０３の処理に移行する。一方で、一致する場合（ステップＳ２４０２：Ｙｅｓ）、分散ノード２０１は、ステップＳ２４０４の処理に移行する。

[0165] ステップＳ２４０３では、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、すべてのプラグイン情報を選択したか否かを判定する（ステップＳ２４０３）。ここで、未選択のプラグイン情報がある場合（ステップＳ２４０３：Ｎｏ）、分散ノード２０１は、ステップＳ２４０１の処理に戻る。一方で、すべてのプラグイン情報を選択している場合（ステップＳ２４０３：Ｙｅｓ）、分散ノード２０１は、プラグイン処理を終了する。

[0166] ステップＳ２４０４では、分散ノード２０１は、プラグイン情報のプラグインデータに基づいて、プラグインを実行する（ステップＳ２４０４）。そして、分散ノード２０１は、プラグイン処理を終了する。

[0167] （制御系イベント処理手順）

次に、図２５を用いて、分散ノード２０１が実行する、制御系イベント処理手順の一例について説明する。制御系イベント処理は、例えば、図３に示

したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0168] 図２５は、制御系イベント処理手順の一例を示すフローチャートである。図２５において、分散ノード２０１は、制御メッセージがプラグイン追加を表すか否かを判定する（ステップＳ２５０１）。ここで、プラグイン追加を表す場合（ステップＳ２５０１：Ｙｅｓ）、分散ノード２０１は、ステップＳ２５０２の処理に移行する。一方で、プラグイン追加を表さない場合（ステップＳ２５０１：Ｎｏ）、分散ノード２０１は、ステップＳ２５０３の処理に移行する。

[0169] ステップＳ２５０２では、分散ノード２０１は、図２６および図２７に後述するプラグイン追加処理を実行する（ステップＳ２５０２）。そして、分散ノード２０１は、制御系イベント処理を終了する。

[0170] ステップＳ２５０３では、分散ノード２０１は、制御メッセージがプラグイン削除を表すか否かを判定する（ステップＳ２５０３）。ここで、プラグイン削除を表す場合（ステップＳ２５０３：Ｙｅｓ）、分散ノード２０１は、図２８および図２９に後述するプラグイン削除処理を実行する（ステップＳ２５０４）。そして、分散ノード２０１は、制御系イベント処理を終了する。一方で、プラグイン削除を表さない場合（ステップＳ２５０３：Ｎｏ）、分散ノード２０１は、制御系イベント処理を終了する。

[0171] （プラグイン追加処理手順）

次に、図２６および図２７を用いて、分散ノード２０１が実行する、プラグイン追加処理手順の一例について説明する。プラグイン追加処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0172] 図２６および図２７は、プラグイン追加処理手順の一例を示すフローチャートである。図２６において、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、まだ選択していないステートテーブル４１０を選択する（ステップＳ２６０１）。

[0173] 次に、分散ノード２０１は、選択したステートテーブル４１０のＩＤが、受け付けた制御メッセージのＩＤと一致するか否かを判定する（ステップＳ２６０２）。ここで、一致する場合（ステップＳ２６０２：Ｙｅｓ）、分散ノード２０１は、ステップＳ２６０４の処理に移行する。一方で、一致しない場合（ステップＳ２６０２：Ｎｏ）、分散ノード２０１は、ステップＳ２６０３の処理に移行する。

[0174] ステップＳ２６０３では、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、すべてのステートテーブル４１０を選択したか否かを判定する（ステップＳ２６０３）。ここで、未選択のステートテーブル４１０がある場合（ステップＳ２６０３：Ｎｏ）、分散ノード２０１は、ステップＳ２６０１の処理に戻る。一方で、すべてのステートテーブル４１０を選択している場合（ステップＳ２６０３：Ｙｅｓ）、分散ノード２０１は、プラグイン追加処理を終了する。

[0175] ステップＳ２６０４では、分散ノード２０１は、ステートテーブル４１０のうち、まだ選択していないステート情報を選択する（ステップＳ２６０４）。

[0176] 次に、分散ノード２０１は、選択したステート情報のステート名と、受け付けた制御メッセージのステート名とが一致するか否かを判定する（ステップＳ２６０５）。ここで、一致する場合（ステップＳ２６０５：Ｙｅｓ）、分散ノード２０１は、図２７のステップＳ２７０１の処理に移行する。一方で、一致しない場合（ステップＳ２６０５：Ｎｏ）、分散ノード２０１は、ステップＳ２６０６の処理に移行する。

[0177] ステップＳ２６０６では、分散ノード２０１は、ステートテーブル４１０のうち、すべてのステート情報を選択したか否かを判定する（ステップＳ２６０６）。ここで、未選択のステート情報がある場合（ステップＳ２６０６：Ｎｏ）、分散ノード２０１は、ステップＳ２６０４の処理に戻る。一方で、すべてのステート情報を選択している場合（ステップＳ２６０６：Ｙｅｓ）、分散ノード２０１は、プラグイン追加処理を終了する。ここで、図２７

の説明に移行する。

[0178] 図２７において、分散ノード２０１は、制御メッセージから、追加するプラグインのプラグインデータを取得する（ステップＳ２７０１）。次に、分散ノード２０１は、プラグインＩＤとして、取得したプラグインデータのＭＤ５ハッシュ値を算出する（ステップＳ２７０２）。そして、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、まだ選択していないプラグイン情報を選択する（ステップＳ２７０３）。

[0179] 次に、分散ノード２０１は、算出したプラグインＩＤと、選択したプラグイン情報のプラグインＩＤとが一致するか否かを判定する（ステップＳ２７０４）。ここで、一致する場合（ステップＳ２７０４：Ｙｅｓ）、分散ノード２０１は、ステップＳ２７０７の処理に移行する。一方で、一致しない場合（ステップＳ２７０４：Ｎｏ）、分散ノード２０１は、ステップＳ２７０５の処理に移行する。

[0180] ステップＳ２７０５では、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、すべてのプラグイン情報を選択したか否かを判定する（ステップＳ２７０５）。ここで、未選択のプラグイン情報がある場合（ステップＳ２７０５：Ｎｏ）、分散ノード２０１は、ステップＳ２７０３の処理に戻る。一方で、すべてのプラグイン情報を選択している場合（ステップＳ２７０５：Ｙｅｓ）、分散ノード２０１は、ステップＳ２７０６の処理に移行する。

[0181] ステップＳ２７０６では、分散ノード２０１は、プラグインリポジトリテーブル５００に新たなプラグイン情報を追加し、新たなプラグイン情報に算出したプラグインＩＤと、取得したプラグインデータとを設定し、参照数を０に初期化する（ステップＳ２７０６）。

[0182] ステップＳ２７０７では、分散ノード２０１は、ステートテーブル４１０のプラグインＩＤに、算出したプラグインＩＤを設定する（ステップＳ２７０７）。次に、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、追加するプラグインに対応する参照数をインクリメントする（ステ

ップＳ２７０８）。そして、分散ノード２０１は、プラグイン追加処理を終了する。

[0183] （プラグイン削除処理手順）

次に、図２８および図２９を用いて、分散ノード２０１が実行する、プラグイン削除処理手順の一例について説明する。プラグイン削除処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0184] 図２８および図２９は、プラグイン削除処理手順の一例を示すフローチャートである。図２８において、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、まだ選択していないステートテーブル４１０を選択する（ステップＳ２８０１）。

[0185] 次に、分散ノード２０１は、選択したステートテーブル４１０のＩＤが、受け付けた制御メッセージのＩＤと一致するか否かを判定する（ステップＳ２８０２）。ここで、一致する場合（ステップＳ２８０２：Ｙｅｓ）、分散ノード２０１は、ステップＳ２８０４の処理に移行する。一方で、一致しない場合（ステップＳ２８０２：Ｎｏ）、分散ノード２０１は、ステップＳ２８０３の処理に移行する。

[0186] ステップＳ２８０３では、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、すべてのステートテーブル４１０を選択したか否かを判定する（ステップＳ２８０３）。ここで、未選択のステートテーブル４１０がある場合（ステップＳ２８０３：Ｎｏ）、分散ノード２０１は、ステップＳ２８０１の処理に戻る。一方で、すべてのステートテーブル４１０を選択している場合（ステップＳ２８０３：Ｙｅｓ）、分散ノード２０１は、プラグイン削除処理を終了する。

[0187] ステップＳ２８０４では、分散ノード２０１は、ステートテーブル４１０のうち、まだ選択していないステート情報を選択する（ステップＳ２８０４）。

[0188] 次に、分散ノード２０１は、選択したステート情報のステート名と、受け

付けた制御メッセージのステート名とが一致するか否かを判定する（ステップＳ２８０５）。ここで、一致する場合（ステップＳ２８０５：Ｙｅｓ）、分散ノード２０１は、図２９のステップＳ２９０１の処理に移行する。一方で、一致しない場合（ステップＳ２８０５：Ｎｏ）、分散ノード２０１は、ステップＳ２８０６の処理に移行する。

[0189] ステップＳ２８０６では、分散ノード２０１は、ステートテーブル４１０のうち、すべてのステート情報を選択したか否かを判定する（ステップＳ２８０６）。ここで、未選択のステート情報がある場合（ステップＳ２８０６：Ｎｏ）、分散ノード２０１は、ステップＳ２８０４の処理に戻る。一方で、すべてのステート情報を選択している場合（ステップＳ２８０６：Ｙｅｓ）、分散ノード２０１は、プラグイン削除処理を終了する。ここで、図２９の説明に移行する。

[0190] 図２９において、分散ノード２０１は、制御メッセージから、削除するプラグインのプラグインデータを取得する（ステップＳ２９０１）。次に、分散ノード２０１は、プラグインＩＤとして、取得したプラグインデータのＭＤ５ハッシュ値を算出する（ステップＳ２９０２）。そして、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、まだ選択していないプラグイン情報を選択する（ステップＳ２９０３）。

[0191] 次に、分散ノード２０１は、算出したプラグインＩＤと、選択したプラグイン情報のプラグインＩＤとが一致するか否かを判定する（ステップＳ２９０４）。ここで、一致する場合（ステップＳ２９０４：Ｙｅｓ）、分散ノード２０１は、ステップＳ２９０６の処理に移行する。一方で、一致しない場合（ステップＳ２９０４：Ｎｏ）、分散ノード２０１は、ステップＳ２９０５の処理に移行する。

[0192] ステップＳ２９０５では、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、すべてのプラグイン情報を選択したか否かを判定する（ステップＳ２９０５）。ここで、未選択のプラグイン情報がある場合（ステップＳ２９０５：Ｎｏ）、分散ノード２０１は、ステップＳ２９０３の処

理に戻る。一方で、すべてのプラグイン情報を選択している場合（ステップＳ２９０５：Ｙｅｓ）、分散ノード２０１は、プラグイン削除処理を終了する。

[0193] ステップＳ２９０６では、分散ノード２０１は、ステートテーブル４１０のプラグインＩＤを空に設定する（ステップＳ２９０６）。次に、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、削除するプラグインに対応する参照数をデクリメントする（ステップＳ２９０７）。

[0194] そして、分散ノード２０１は、プラグインリポジトリテーブル５００の参照数が０であるか否かを判定する（ステップＳ２９０８）。ここで、参照数が０ではない場合（ステップＳ２９０８：Ｎｏ）、分散ノード２０１は、プラグイン削除処理を終了する。一方で、参照数が０である場合（ステップＳ２９０８：Ｙｅｓ）、分散ノード２０１は、ステップＳ２９０９の処理に移行する。

[0195] ステップＳ２９０９では、分散ノード２０１は、プラグインリポジトリテーブル５００から、参照数が０であるプラグイン情報を削除する（ステップＳ２９０９）。そして、分散ノード２０１は、プラグイン削除処理を終了する。

[0196] （チェックポイント処理手順）

次に、図３０を用いて、分散ノード２０１が実行する、チェックポイント処理手順の一例について説明する。チェックポイント処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0197] 図３０は、チェックポイント処理手順の一例を示すフローチャートである。図３０において、分散ノード２０１は、チェックポイント番号を０に初期化する（ステップＳ３００１）。次に、分散ノード２０１は、図３１に後述するチェックポイント保存処理を実行する（ステップＳ３００２）。

[0198] そして、分散ノード２０１は、チェックポイント番号をインクリメントする（ステップＳ３００３）。その後、分散ノード２０１は、一定時間待機し

（ステップＳ３００４）、ステップＳ３００２の処理に戻る。

[0199] （チェックポイント保存処理手順）

次に、図３１を用いて、分散ノード２０１が実行する、チェックポイント保存処理手順の一例について説明する。チェックポイント保存処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0200] 図３１は、チェックポイント保存処理手順の一例を示すフローチャートである。図３１において、分散ノード２０１は、ステートテーブル４１０のリスト４００から、まだ選択していないステートテーブル４１０を選択する（ステップＳ３１０１）。

[0201] 次に、分散ノード２０１は、選択したステートテーブル４１０を、チェックポイント番号と、自ノードの識別子と、選択したステートテーブル４１０のＩＤとに対応付けて、外部記憶装置２０５に保存する（ステップＳ３１０２）。

[0202] そして、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、すべてのステートテーブル４１０を選択したか否かを判定する（ステップＳ３１０３）。ここで、未選択のステートテーブル４１０がある場合（ステップＳ３１０３：Ｎｏ）、分散ノード２０１は、ステップＳ３１０１の処理に戻る。一方で、すべてのステートテーブル４１０を選択している場合（ステップＳ３１０３：Ｙｅｓ）、分散ノード２０１は、ステップＳ３１０４の処理に移行する。

[0203] ステップＳ３１０４では、分散ノード２０１は、プラグインリポジトリテーブル５００のスナップショットを、チェックポイント番号と、自ノードの識別子とに対応付けて、外部記憶装置２０５に保存する（ステップＳ３１０４）。そして、分散ノード２０１は、チェックポイント保存処理を終了する。

[0204] （回復処理手順）

次に、図３２を用いて、分散ノード２０１が実行する、回復処理手順の一

例について説明する。回復処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0205] 図３２は、回復処理手順の一例を示すフローチャートである。図３２において、分散ノード２０１は、外部記憶装置２０５に保存された、直近のチェックポイント番号と、自ノードの識別子と、自ノードに割り当てられたＩＤとの組み合わせに対応する、ステート情報を取得する（ステップＳ３２０１）。次に、分散ノード２０１は、取得したステート情報に基づいて、ステートテーブル４１０を更新する（ステップＳ３２０２）。

[0206] そして、分散ノード２０１は、他ノードから引き継ぐステート情報が存在するか否かを判定する（ステップＳ３２０３）。ここで、存在しない場合（ステップＳ３２０３：Ｎｏ）、分散ノード２０１は、回復処理を終了する。一方で、存在する場合（ステップＳ３２０３：Ｙｅｓ）、分散ノード２０１は、ステップＳ３２０４の処理に移行する。

[0207] ステップＳ３２０４では、分散ノード２０１は、引き継ぐステート情報を、外部記憶装置２０５から取得する（ステップＳ３２０４）。次に、分散ノード２０１は、取得したステート情報に基づいて、ステートテーブル４１０を更新する（ステップＳ３２０５）。そして、分散ノード２０１は、図３３～図３５に後述する再生成処理を実行する（ステップＳ３２０６）。その後、分散ノード２０１は、回復処理を終了する。

[0208] （再生成処理手順）

次に、図３３～図３５を用いて、分散ノード２０１が実行する、再生成処理手順の一例について説明する。再生成処理は、例えば、図３に示したＣＰＵ３０１と、メモリ３０２や記録媒体３０５などの記憶領域と、ネットワークＩ／Ｆ３０３とによって実現される。

[0209] 図３３～図３５は、再生成処理手順の一例を示すフローチャートである。図３３において、分散ノード２０１は、プラグインリポジトリテーブル５００の参照数を０に初期化する（ステップＳ３３０１）。

[0210] 次に、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、まだ選択していないステートテーブル４１０を選択する（ステップＳ３３０２）。そして、分散ノード２０１は、選択したステートテーブル４１０のうち、まだ選択していないステート情報を選択する（ステップＳ３３０３）。

[0211] 次に、分散ノード２０１は、選択したステート情報にプラグインＩＤが含まれるか否かを判定する（ステップＳ３３０４）。ここで、プラグインＩＤが含まれる場合（ステップＳ３３０４：Ｙｅｓ）、分散ノード２０１は、図３４のステップＳ３４０１の処理に移行する。一方で、プラグインＩＤが含まれない場合（ステップＳ３３０４：Ｎｏ）、分散ノード２０１は、ステップＳ３３０５の処理に移行する。

[0212] ステップＳ３３０５では、分散ノード２０１は、選択したステートテーブル４１０のうち、すべてのステート情報を選択したか否かを判定する（ステップＳ３３０５）。ここで、未選択のステート情報がある場合（ステップＳ３３０５：Ｎｏ）、分散ノード２０１は、ステップＳ３３０３の処理に戻る。一方で、すべてのステート情報を選択している場合（ステップＳ３３０５：Ｙｅｓ）、分散ノード２０１は、ステップＳ３３０６の処理に移行する。

[0213] ステップＳ３３０６では、分散ノード２０１は、ステートテーブル４１０のリスト４００のうち、すべてのステートテーブル４１０を選択したか否かを判定する（ステップＳ３３０６）。ここで、未選択のステートテーブル４１０がある場合（ステップＳ３３０６：Ｎｏ）、分散ノード２０１は、ステップＳ３３０２の処理に戻る。一方で、すべてのステートテーブル４１０を選択している場合（ステップＳ３３０６：Ｙｅｓ）、分散ノード２０１は、図３５のステップＳ３５０１の処理に移行する。まず、図３４の説明に移行する。

[0214] 図３４において、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、まだ選択していないプラグイン情報を選択する（ステップＳ３４０１）。

[0215] 次に、分散ノード２０１は、選択したステート情報のプラグインＩＤと、選択したプラグイン情報のプラグインＩＤとが一致するか否かを判定する（ステップＳ３４０２）。ここで、一致する場合（ステップＳ３４０２：Ｙｅｓ）、分散ノード２０１は、ステップＳ３４０５の処理に移行する。一方で、一致しない場合（ステップＳ３４０２：Ｎｏ）、分散ノード２０１は、ステップＳ３４０３の処理に移行する。

[0216] ステップＳ３４０３では、分散ノード２０１は、すべてのプラグイン情報を選択したか否かを判定する（ステップＳ３４０３）。ここで、未選択のプラグイン情報がある場合（ステップＳ３４０３：Ｎｏ）、分散ノード２０１は、ステップＳ３４０１の処理に戻る。一方で、すべてのプラグイン情報を選択している場合（ステップＳ３４０３：Ｙｅｓ）、分散ノード２０１は、ステップＳ３４０４の処理に移行する。

[0217] ステップＳ３４０４では、分散ノード２０１は、プラグインリポジトリテーブル５００に、選択したステート情報のプラグインＩＤが設定された新しいプラグイン情報を追加し、参照数を０に初期化する（ステップＳ３４０４）。そして、分散ノード２０１は、ステップＳ３４０５の処理に移行する。

[0218] ステップＳ３４０５では、分散ノード２０１は、プラグイン情報の参照数をインクリメントする（ステップＳ３４０５）。そして、分散ノード２０１は、図３３のステップＳ３３０５の処理に戻る。次に、図３５の説明に移行する。

[0219] 図３５において、分散ノード２０１は、外部記憶装置２０５から、直近のチェックポイント番号と対応付けられたリポジトリテーブルのスナップショットを取得する（ステップＳ３５０１）。次に、分散ノード２０１は、取得したスナップショットに基づいて、プラグインＩＤとプラグインデータとを対応付けた一覧情報を生成する（ステップＳ３５０２）。そして、分散ノード２０１は、プラグインリポジトリテーブル５００のうち、まだ選択していないプラグイン情報を選択する（ステップＳ３５０３）。

[0220] 次に、分散ノード２０１は、選択したプラグイン情報のプラグインデータ

が空であるか否かを判定する（ステップＳ３５０４）。ここで、空ではない場合（ステップＳ３５０４：Ｎｏ）、分散ノード２０１は、ステップＳ３５０６の処理に移行する。一方で、空である場合（ステップＳ３５０４：Ｙｅｓ）、分散ノード２０１は、ステップＳ３５０５の処理に移行する。

[0221] ステップＳ３５０５では、分散ノード２０１は、空であるプラグインデータを、一覧情報から探索し、選択したプラグイン情報に設定する（ステップＳ３５０５）。そして、分散ノード２０１は、ステップＳ３５０６の処理に移行する。

[0222] ステップＳ３５０６では、分散ノード２０１は、すべてのプラグイン情報を選択したか否かを判定する（ステップＳ３５０６）。ここで、未選択のプラグイン情報がある場合（ステップＳ３５０６：Ｎｏ）、分散ノード２０１は、ステップＳ３５０３の処理に戻る。一方で、すべてのプラグイン情報を選択している場合（ステップＳ３５０６：Ｙｅｓ）、分散ノード２０１は、再生成処理を終了する。

[0223] 以上説明したように、情報処理装置１１０によれば、自装置とは異なる他の情報処理装置１１０が所定の端末装置に提供していた所定の機能を引き継ぐことの要求を受け付けることができる。情報処理装置１１０によれば、要求を受け付けた場合、外部記憶装置１３０から、他の情報処理装置１１０によって格納されたバックアップデータを取得することができる。情報処理装置１１０によれば、取得したバックアップデータを解釈し、他の情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードのうち、所定の機能を実現するプラグインのコードを抽出することができる。情報処理装置１１０によれば、抽出したコードに基づいて、所定の機能を所定の端末装置に提供することができる。これにより、情報処理装置１１０は、他の情報処理装置１１０が提供していた機能を実現するプラグインのコードを取得することができ、他の情報処理装置１１０が提供していた機能を引き継ぐことができる。

[0224] 情報処理装置１１０によれば、外部記憶装置１３０から、他の情報処理装

置１１０によって格納された、第１の対応情報を集約したバックアップデータ、および、第２の対応情報を取得することができる。情報処理装置１１０によれば、取得したバックアップデータを解釈し、取得した第２の対応情報を参照し、要求に含まれる所定の端末装置の識別子に基づいて、第１の対応情報のうち、所定の機能を実現するプラグインのコードを抽出することができる。これにより、情報処理装置１１０は、所定の端末装置の識別子に基づいて、他の情報処理装置１１０が提供していた機能を実現するプラグインのコードを特定しやすくすることができる。

[0225] 情報処理装置１１０によれば、外部記憶装置１３０から、自装置によって格納されたバックアップデータを取得することができる。情報処理装置１１０によれば、取得したバックアップデータを解釈し、自装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを抽出することができる。これにより、情報処理装置１１０は、自装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを、バックアップ時の状態に戻すことができる。

[0226] 情報処理装置１１０によれば、外部記憶装置１３０から、それぞれの情報処理装置１１０によって格納されたバックアップデータを取得することができる。情報処理装置１１０によれば、取得したバックアップデータを解釈し、それぞれの情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを、重複するプラグインのコードを統合して纏めることができる。情報処理装置１１０によれば、所定の機能を実現するプラグインのコードを抽出することができる。これにより、情報処理装置１１０は、引き継ぐ機能を提供していた情報処理装置１１０を特定しなくても、引き継ぐ機能を実現するプラグインのコードを抽出することができる。

[0227] 情報処理装置１１０によれば、管理装置から、プラグインのコードを受信して記憶することができる。これにより、情報処理装置１１０は、プラグインのコードを記憶することができる。また、情報処理装置１１０は、管理装

置がプラグインのコードを送信した後、破棄していても、外部記憶装置１３０に基づいて、プラグインのコードを抽出することができる。

[0228] 情報処理装置１１０によれば、いずれかの情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを、所定のデータ形式に沿って区切って集約するバックアップデータを取得することができる。そして、情報処理装置１１０によれば、バックアップデータを解釈し、プラグイン単位でプラグインのコードを取得することができる。これにより、情報処理装置１１０は、外部記憶装置１３０がバックアップデータを解釈する機能を有していなくても、正常に動作することができる。

[0229] 情報処理装置１１０によれば、いずれかの情報処理装置１１０が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを、暗号化して集約するバックアップデータを取得することができる。そして、情報処理装置１１０によれば、バックアップデータを解釈し、プラグイン単位でプラグインのコードを取得することができる。これにより、情報処理装置１１０は、外部記憶装置１３０がバックアップデータを解釈し、復号する機能を有していなくても、正常に動作することができる。また、情報処理装置１１０は、バックアップデータを暗号化してセキュリティの向上を図ることができる。

[0230] なお、本実施の形態で説明した引継方法は、予め用意されたプログラムをパーソナル・コンピュータやワークステーション等のコンピュータで実行することにより実現することができる。本実施の形態で説明した引継プログラムは、ハードディスク、フレキシブルディスク、ＣＤ－ＲＯＭ、ＭＯ、ＤＶＤ等のコンピュータで読み取り可能な記録媒体に記録され、コンピュータによって記録媒体から読み出されることによって実行される。また、本実施の形態で説明した引継プログラムは、インターネット等のネットワークを介して配布してもよい。

符号の説明

[0231] １００，２００　並列分散処理システム

１１０　情報処理装置

１２０　コンピュータ

１３０，２０５　外部記憶装置

２０１　分散ノード

２０２　管理ノード

２０３　端末装置

２０４　開発ノード

２１０　ネットワーク

３００　バス

３０１　ＣＰＵ

３０２　メモリ

３０３　ネットワークＩ／Ｆ

３０４　記録媒体Ｉ／Ｆ

３０５　記録媒体

４００，１９２０　リスト

４１０，１９２１　ステートテーブル

４２０　図形

５００，１９３０　プラグインリポジトリテーブル

６００　記憶部

６０１　取得部

６０２　更新部

６０３　抽出部

６０４　実行部

６０５　出力部

１１０１～１１０３　スナップショット

１６０１～１６０３　計算機

１６１０　クラスタ間通信ネットワーク

１７０１　文字列解釈ブロック

１７０２　仮想オブジェクトブロック

１８０１　文字列解釈モジュール

１８０２　宛先決定モジュール

１９０１　イベントメッセージ

１９０２　制御メッセージ

１９１１　メッセージ判別部

１９１２　制御メッセージ処理部

１９１３　イベントメッセージ処理部

１９１４　チェックポイント管理部

１９１５　プラグインリポジトリ再生成部

請求の範囲

[請求項1] 端末装置に機能を提供する複数の情報処理装置のうち第１の情報処理装置に、

前記複数の情報処理装置のうち前記第１の情報処理装置とは異なる第２の情報処理装置が所定の端末装置に提供していた所定の機能を引き継ぐことの要求を受け付け、

前記要求を受け付けた場合、前記複数の情報処理装置のそれぞれの情報処理装置によって、前記情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを集約したバックアップデータが格納される外部記憶装置から、前記第２の情報処理装置によって格納されたバックアップデータを取得し、

取得した前記バックアップデータを解釈し、前記第２の情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードのうち、前記所定の機能を実現するプラグインのコードを抽出し、

抽出した前記コードに基づいて、前記所定の機能を前記所定の端末装置に提供する、

処理を実行させることを特徴とする引継プログラム。

[請求項2] 前記要求は、前記所定の端末装置の識別子を含み、

前記バックアップデータは、前記情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインの識別子とコードとを対応付けた対応情報を集約し、

前記外部記憶装置は、さらに、前記情報処理装置が提供していた１以上の機能のそれぞれの機能について、前記機能を実現するプラグインの識別子と、前記機能が提供される端末装置の識別子とを対応付けた対応情報が格納され、

前記取得する処理は、

前記外部記憶装置から、前記第２の情報処理装置によって格納され

た、前記第２の情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインの識別子とコードとを対応付けた第１の対応情報を集約したバックアップデータ、および、前記第２の情報処理装置が提供していた１以上の機能のそれぞれの機能について、前記機能を実現するプラグインの識別子と、前記機能を提供する端末装置の識別子とを対応付けた第２の対応情報を取得し、

前記抽出する処理は、

取得した前記バックアップデータを解釈し、取得した前記第２の対応情報を参照し、前記要求に含まれる前記所定の端末装置の識別子に基づいて、前記第１の対応情報のうち、前記所定の機能を実現するプラグインのコードを抽出する、ことを特徴とする請求項１に記載の引継プログラム。

[請求項3] 前記取得する処理は、

前記外部記憶装置から、さらに、前記第１の情報処理装置によって格納されたバックアップデータを取得し、

前記抽出する処理は、

取得した前記第１の情報処理装置によって格納されたバックアップデータを解釈し、前記第１の情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを抽出する、ことを特徴とする請求項１または２に記載の引継プログラム。

[請求項4] 前記取得する処理は、

前記外部記憶装置から、前記それぞれの情報処理装置によって格納されたバックアップデータを取得し、

前記抽出する処理は、

取得した前記それぞれの情報処理装置によって格納されたバックアップデータを解釈し、前記それぞれの情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを、重複するプラグインのコードを統合して纏め、前記所定の機能を実現する

プラグインのコードを抽出する、ことを特徴とする請求項１～３のいずれか一つに記載の引継プログラム。

[請求項5] 前記情報処理装置は、前記情報処理装置にプラグインのコードを送信した後で前記プラグインのコードを破棄する管理装置から、前記プラグインのコードを受信して記憶する、ことを特徴とする請求項１～４のいずれか一つに記載の引継プログラム。

[請求項6] 前記バックアップデータは、前記情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを、所定のデータ形式に沿って区切って集約する、ことを特徴とする請求項１～５のいずれか一つに記載の引継プログラム。

[請求項7] 前記バックアップデータは、前記情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを、暗号化して集約する、ことを特徴とする請求項１～６のいずれか一つに記載の引継プログラム。

[請求項8] 端末装置に機能を提供する複数の情報処理装置のうち第１の情報処理装置が、

前記複数の情報処理装置のうち前記第１の情報処理装置とは異なる第２の情報処理装置が所定の端末装置に提供していた所定の機能を引き継ぐことの要求を受け付け、

前記要求を受け付けた場合、前記複数の情報処理装置のそれぞれの情報処理装置によって、前記情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードを集約したバックアップデータが格納される外部記憶装置から、前記第２の情報処理装置によって格納されたバックアップデータを取得し、

取得した前記バックアップデータを解釈し、前記第２の情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードのうち、前記所定の機能を実現するプラグインのコードを抽出し、

抽出した前記コードに基づいて、前記所定の機能を前記所定の端末装置に提供する、

処理を実行することを特徴とする引継方法。

[請求項9] 端末装置に機能を提供する複数の情報処理装置のいずれかの情報処理装置であって、

前記複数の情報処理装置のうち、自装置とは異なる他の情報処理装置が所定の端末装置に提供していた所定の機能を引き継ぐことの要求を受け付け、

前記要求を受け付けた場合、前記複数の情報処理装置のそれぞれの情報処理装置によって、当該情報処理装置が提供していた1以上の機能のそれぞれの機能を実現するプラグインのコードを集約したバックアップデータが格納される外部記憶装置から、前記他の情報処理装置によって格納されたバックアップデータを取得し、

取得した前記バックアップデータを解釈し、前記他の情報処理装置が提供していた1以上の機能のそれぞれの機能を実現するプラグインのコードのうち、前記所定の機能を実現するプラグインのコードを抽出し、

抽出した前記コードに基づいて、前記所定の機能を前記所定の端末装置に提供する、

制御部を有することを特徴とする情報処理装置。

[請求項10] 端末装置に機能を提供する複数の情報処理装置を含む並列分散処理システムであって、

前記複数の情報処理装置のそれぞれの情報処理装置は、

前記複数の情報処理装置のうち、自装置とは異なる他の情報処理装置が所定の端末装置に提供していた所定の機能を引き継ぐことの要求を受け付け、

前記要求を受け付けた場合、前記複数の情報処理装置のそれぞれの情報処理装置によって、当該情報処理装置が提供していた1以上の機

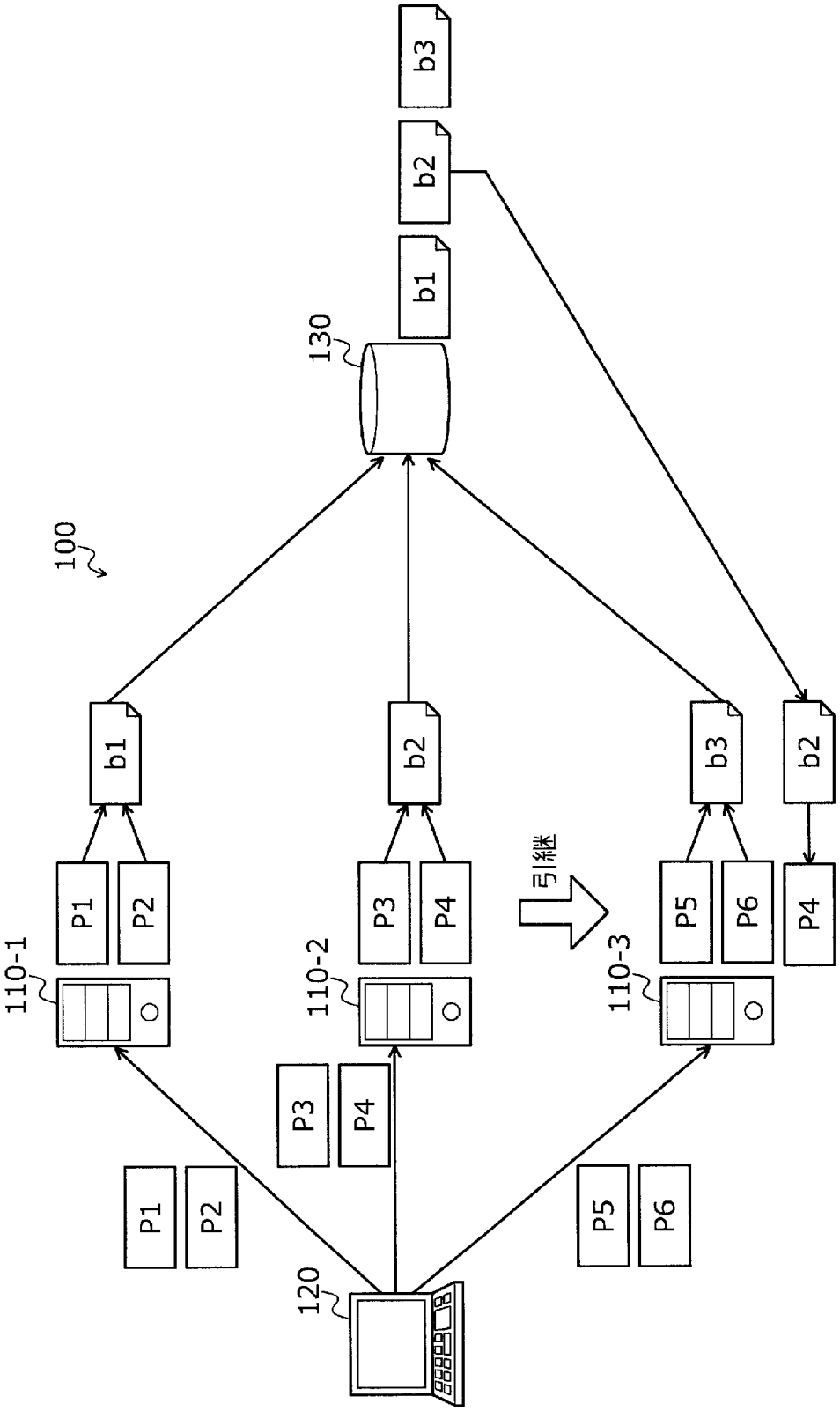
能のそれぞれの機能を実現するプラグインのコードを集約したバックアップデータが格納される外部記憶装置から、前記他の情報処理装置によって格納されたバックアップデータを取得し、

取得した前記バックアップデータを解釈し、前記他の情報処理装置が提供していた１以上の機能のそれぞれの機能を実現するプラグインのコードのうち、前記所定の機能を実現するプラグインのコードを抽出し、

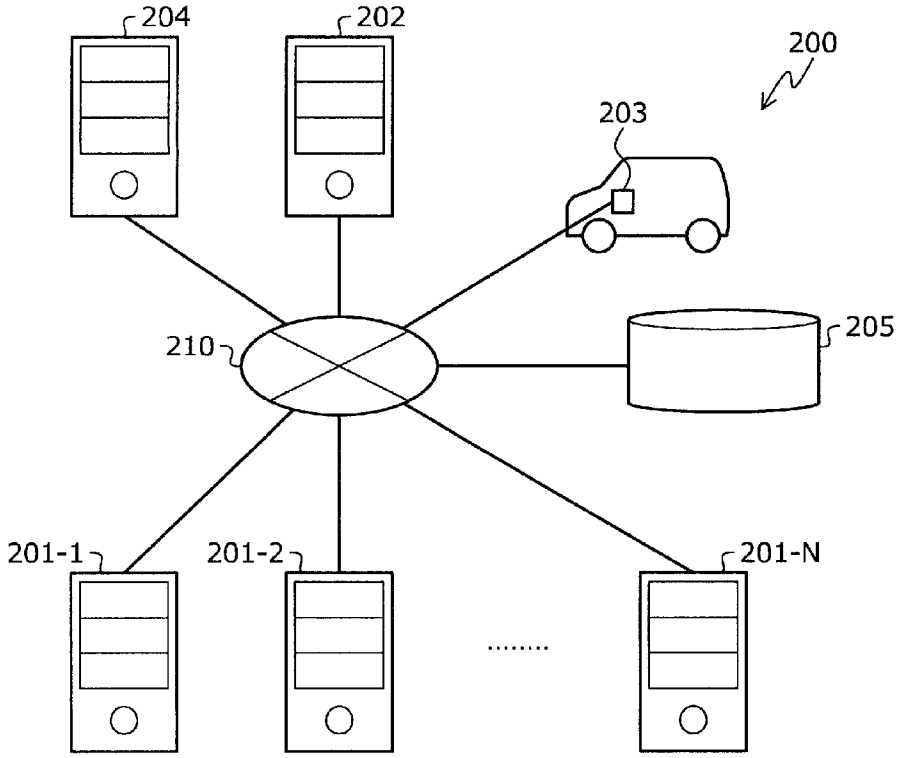
抽出した前記コードに基づいて、前記所定の機能を前記所定の端末装置に提供する、

ことを特徴とする並列分散処理システム。

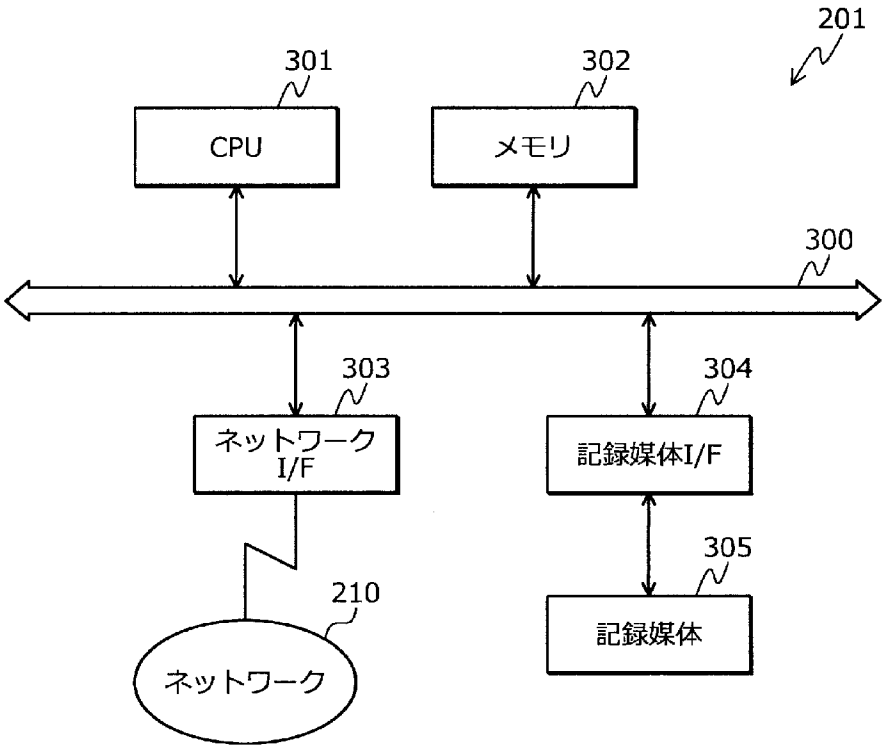
[図1]



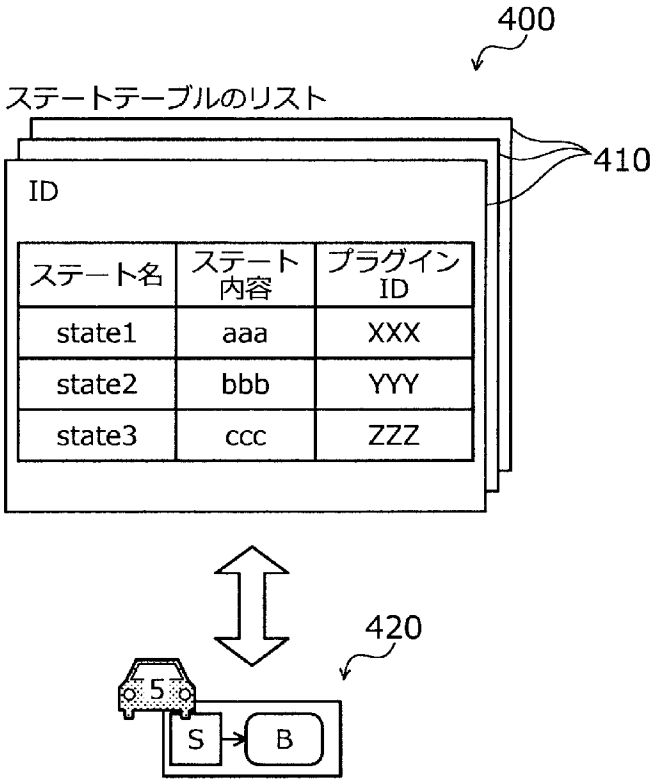
[図2]


204
202
200
203
210
205
201-1
201-2
201-N
.....

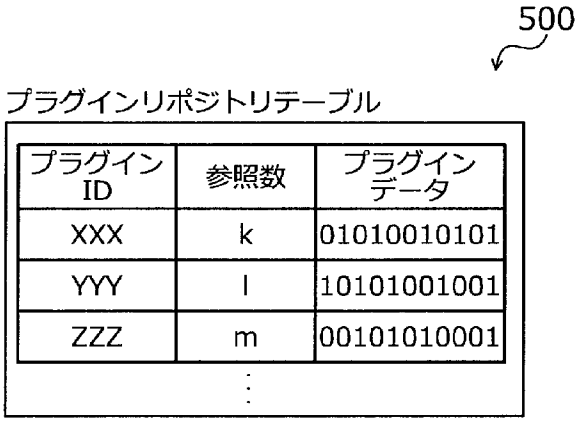
[図3]


201
301
302
CPU
メモリ
300
303
304
ネットワーク
I/F
記録媒体I/F
305
210
記録媒体
ネットワーク

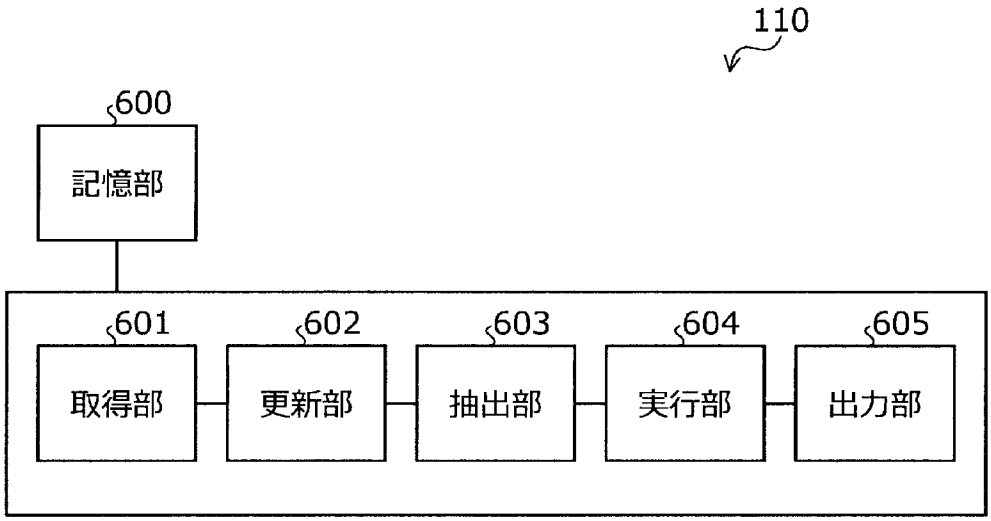
[図4]


400
ステートテーブルのリスト
410
ID
ステート名
ステート内容
プラグインID
state1
aaa
XXX
state2
bbb
YYY
state3
ccc
ZZZ
420
5
S
B

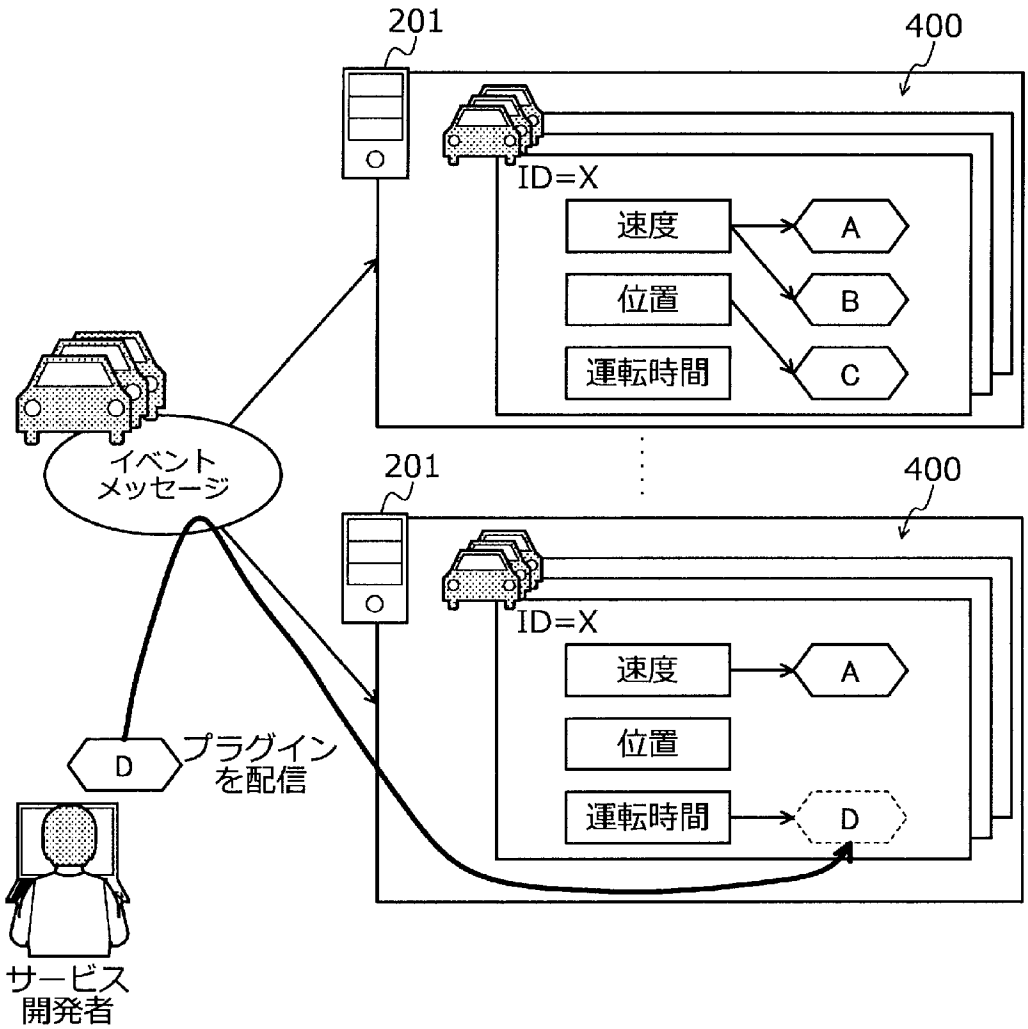
[図5]


500
プラグインリポジトリテーブル
プラグインID
参照数
プラグインデータ
XXX
k
01010010101
YYY
l
10101001001
ZZZ
m
00101010001

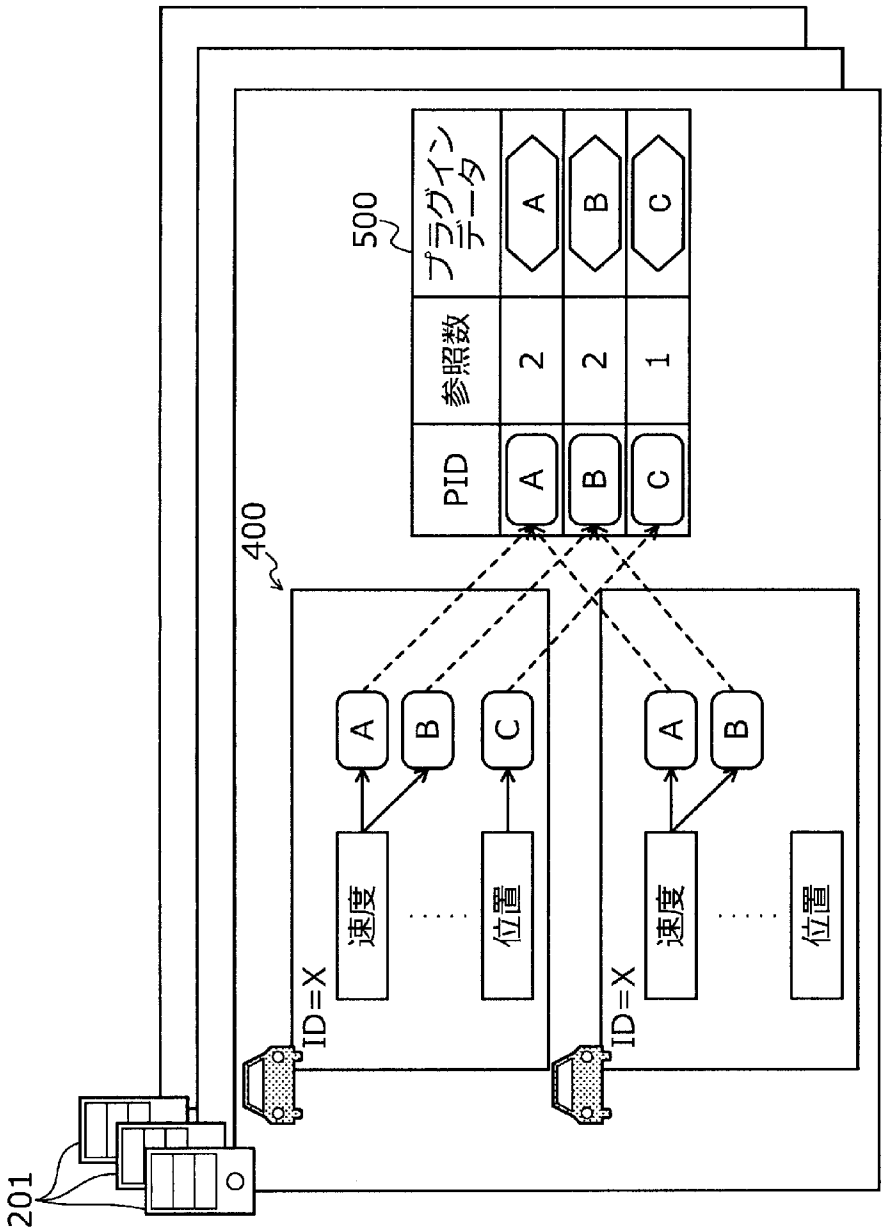
[図6]



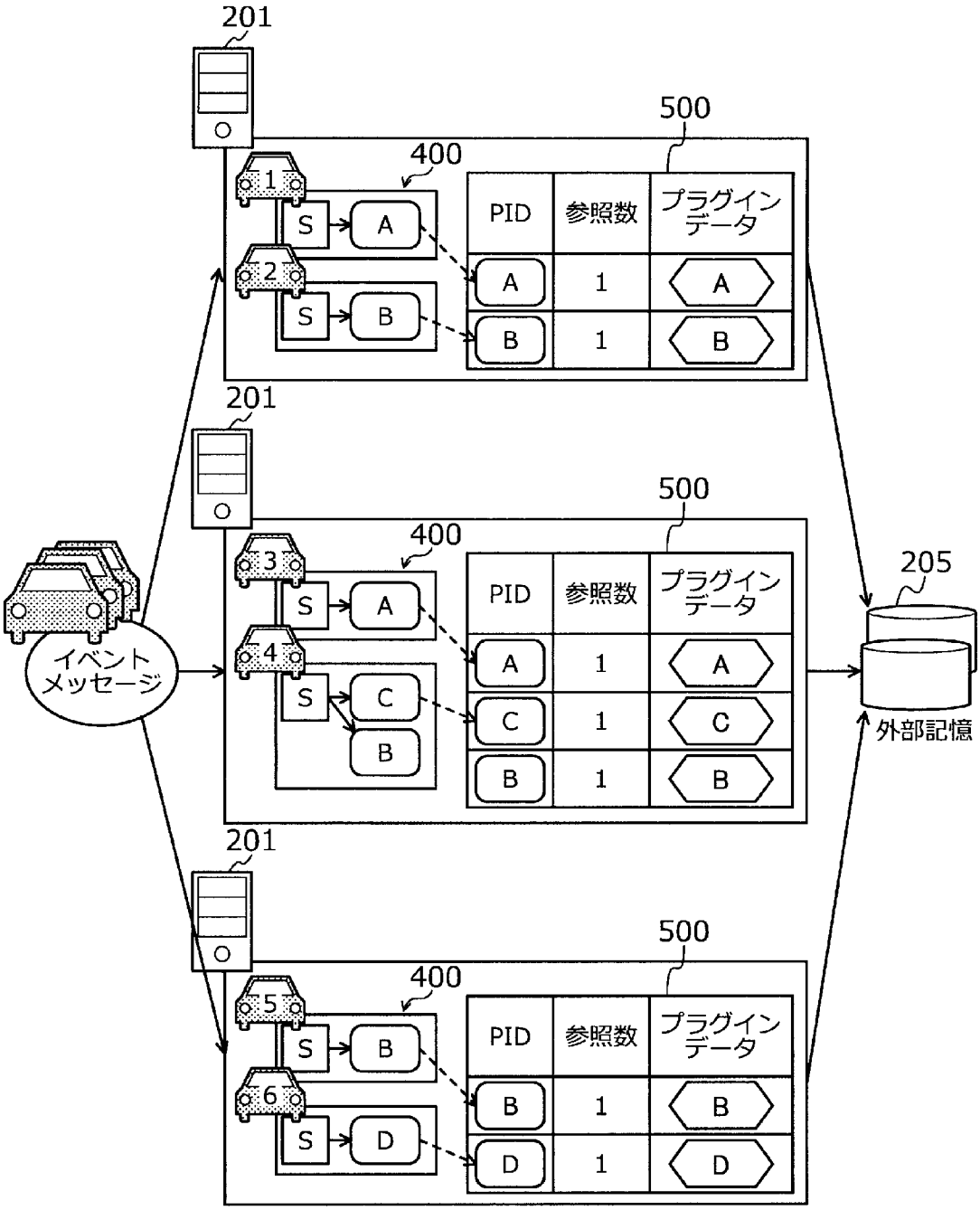
[図7]


201
400
ID=X
速度
A
位置
B
運転時間
C
イベント
メッセージ
201
400
ID=X
速度
A
位置
運転時間
D
D
プラグイン
を配信
サービス
開発者

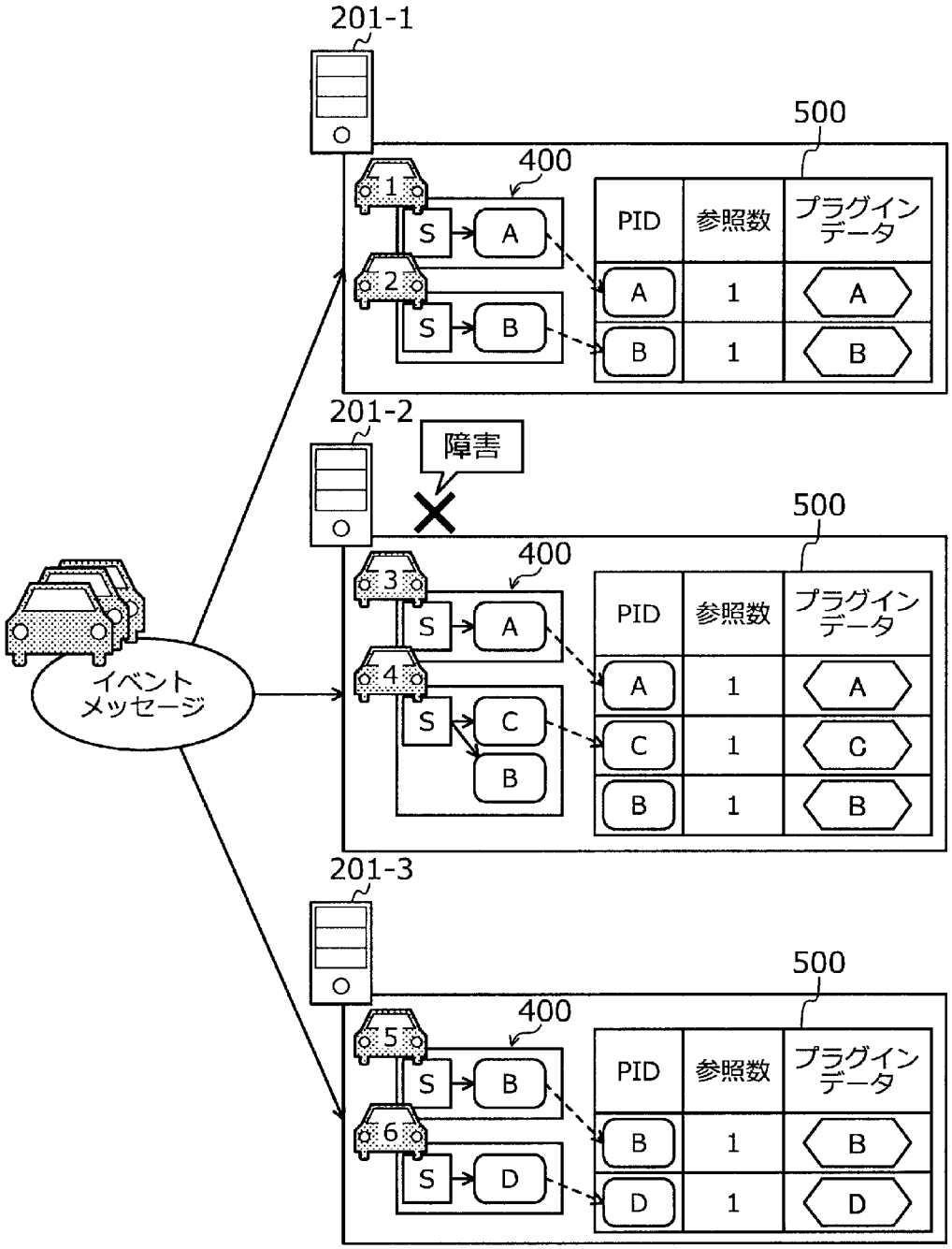
[図8]


201
400
ID=X
速度
位置
A
B
C
ID=X
速度
位置
A
B
500
PID
参照数
プラグイン
データ
A
2
A
B
2
B
C
1
C

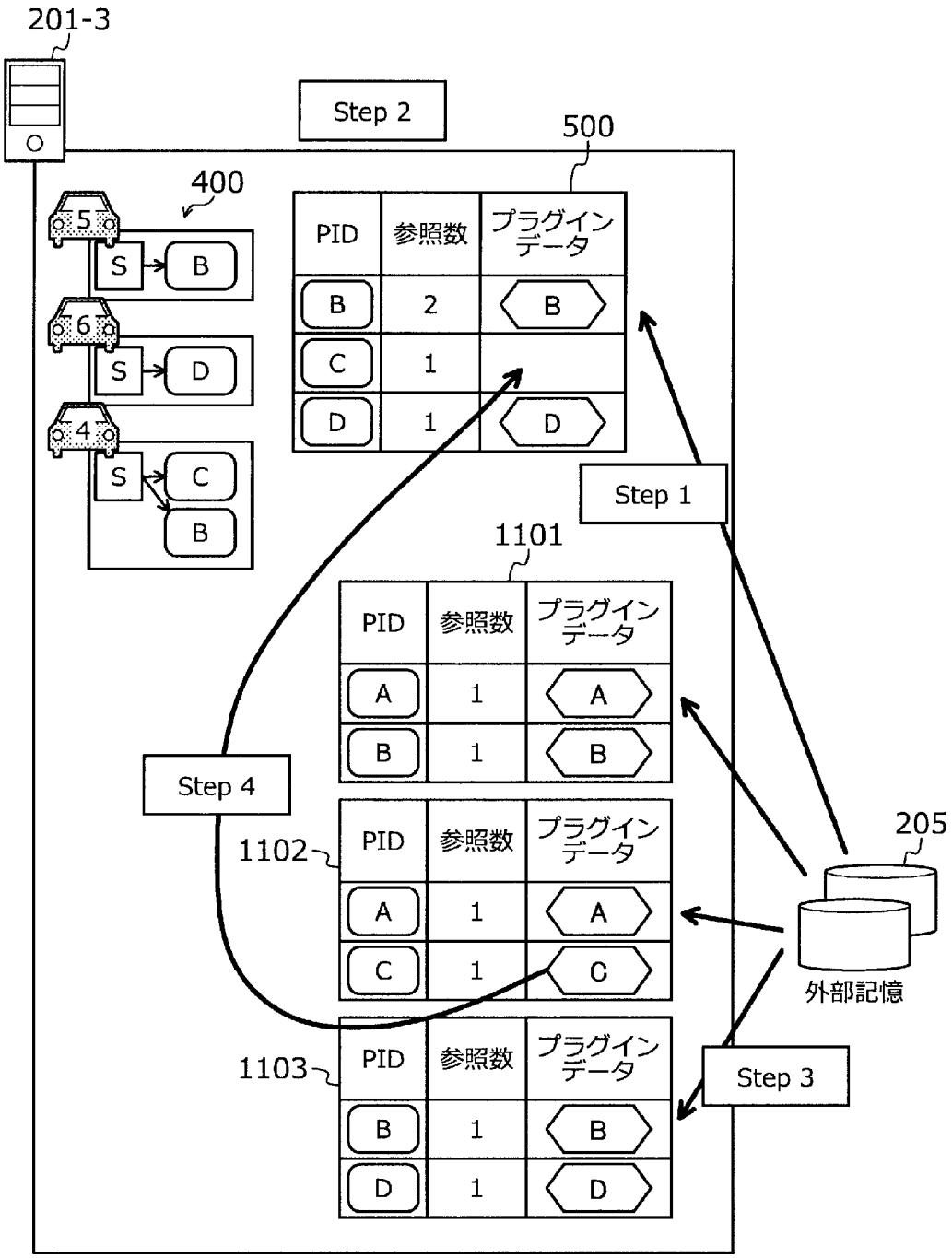
[図9]


201
500
400
1
S
A
2
S
B
PID
参照数
プラグイン
データ
A
1
A
B
1
B
201
500
400
3
S
A
4
S
C
B
PID
参照数
プラグイン
データ
A
1
A
C
1
C
B
1
B
205
外部記憶
イベント
メッセージ
201
500
400
5
S
B
6
S
D
PID
参照数
プラグイン
データ
B
1
B
D
1
D

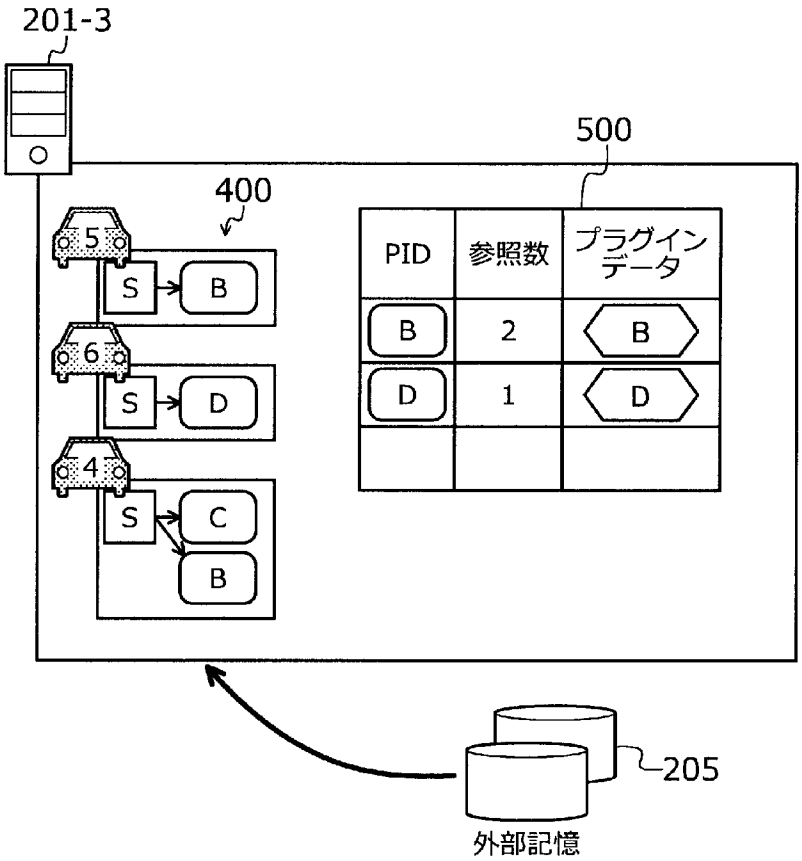
[図10]


201-1
500
400
1
S
A
2
S
B
PID
参照数
プラグイン
データ
A
1
A
B
1
B
201-2
障害
500
400
3
S
A
4
S
C
B
PID
参照数
プラグイン
データ
A
1
A
C
1
C
B
1
B
イベント
メッセージ
201-3
500
400
5
S
B
6
S
D
PID
参照数
プラグイン
データ
B
1
B
D
1
D

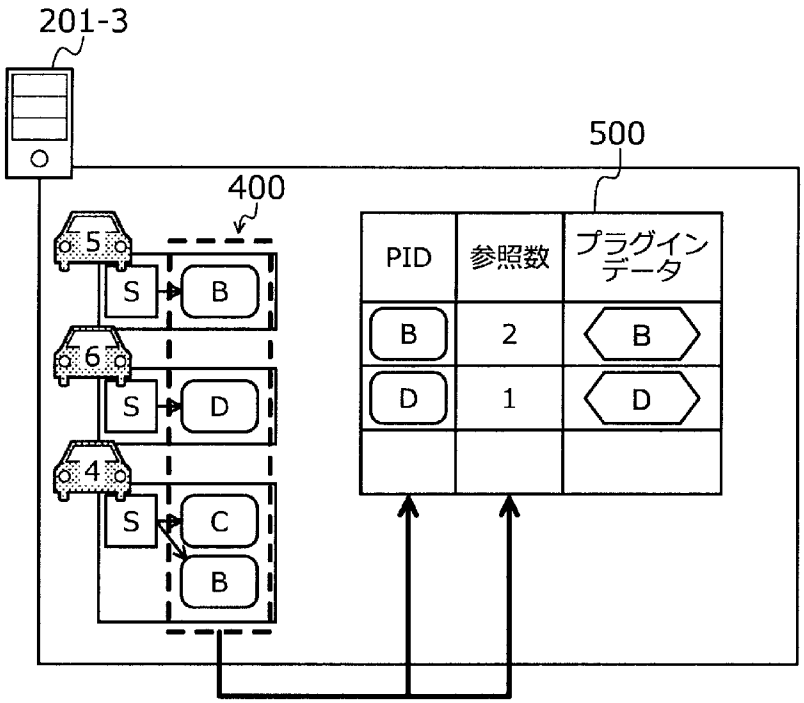
[図11]


201-3
Step 2
500
400
5
S
B
6
S
D
4
S
C
B
PID
参照数
プラグインデータ
B
2
B
C
1
D
1
D
Step 1
1101
PID
参照数
プラグインデータ
A
1
A
B
1
B
Step 4
1102
PID
参照数
プラグインデータ
A
1
A
C
1
C
205
外部記憶
1103
PID
参照数
プラグインデータ
B
1
B
D
1
D
Step 3

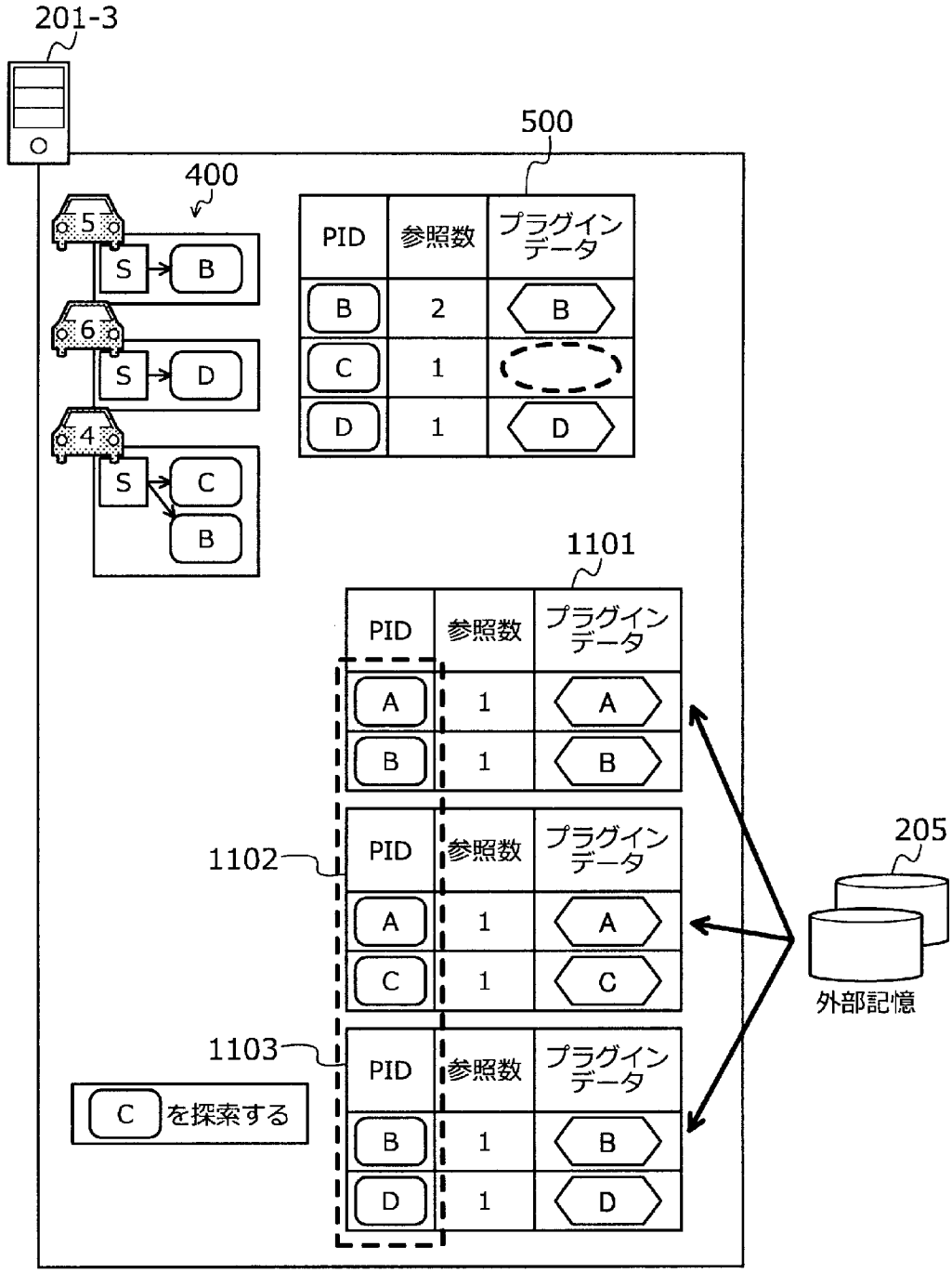
[図12]


201-3
500
400
5
S
B
6
S
D
4
S
C
B
PID
参照数
プラグインデータ
B
2
B
D
1
D
205
外部記憶

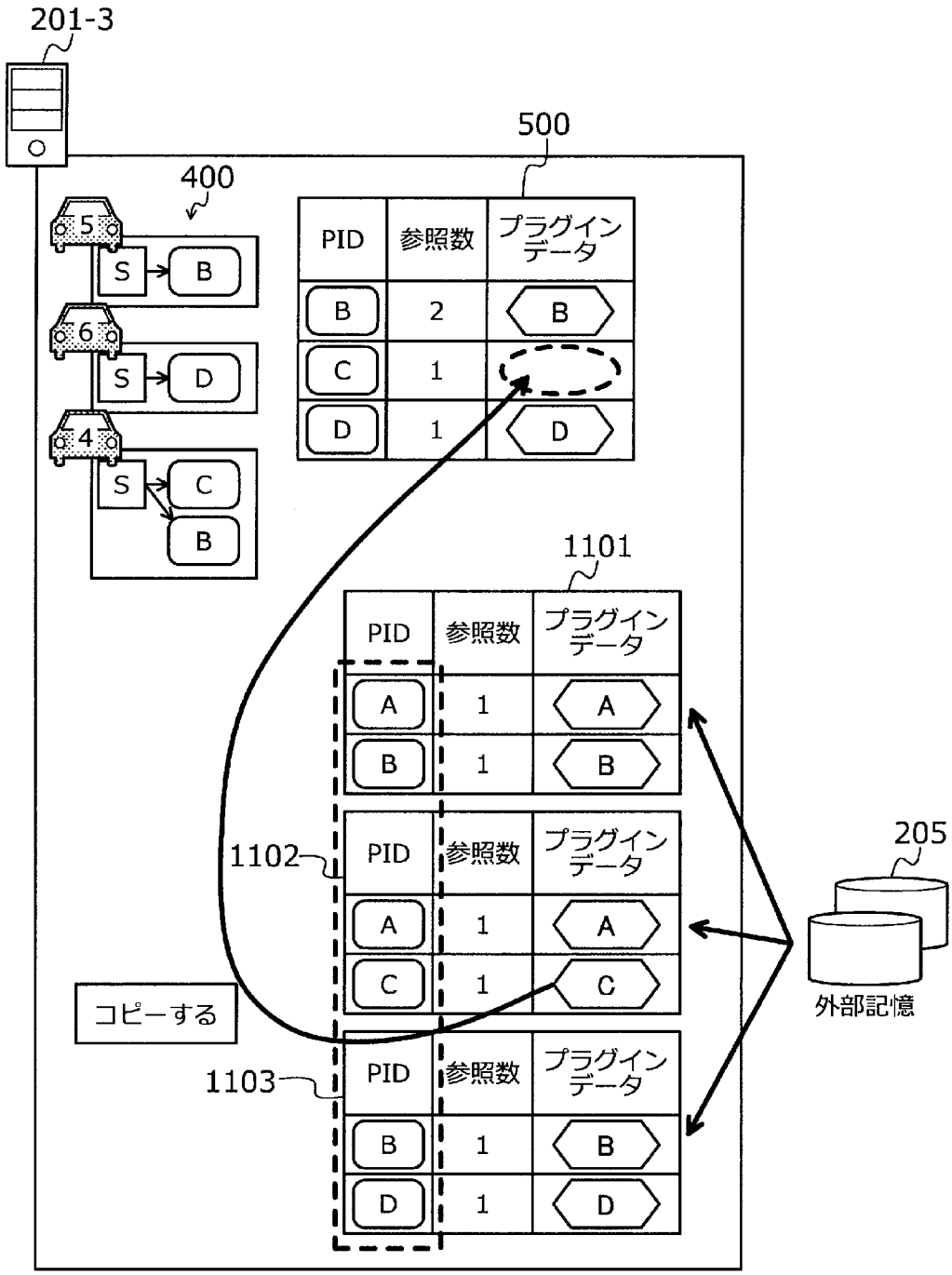
[図13]


201-3
500
400
5
S
B
6
S
D
4
S
C
B
PID
参照数
プラグインデータ
B
2
B
D
1
D

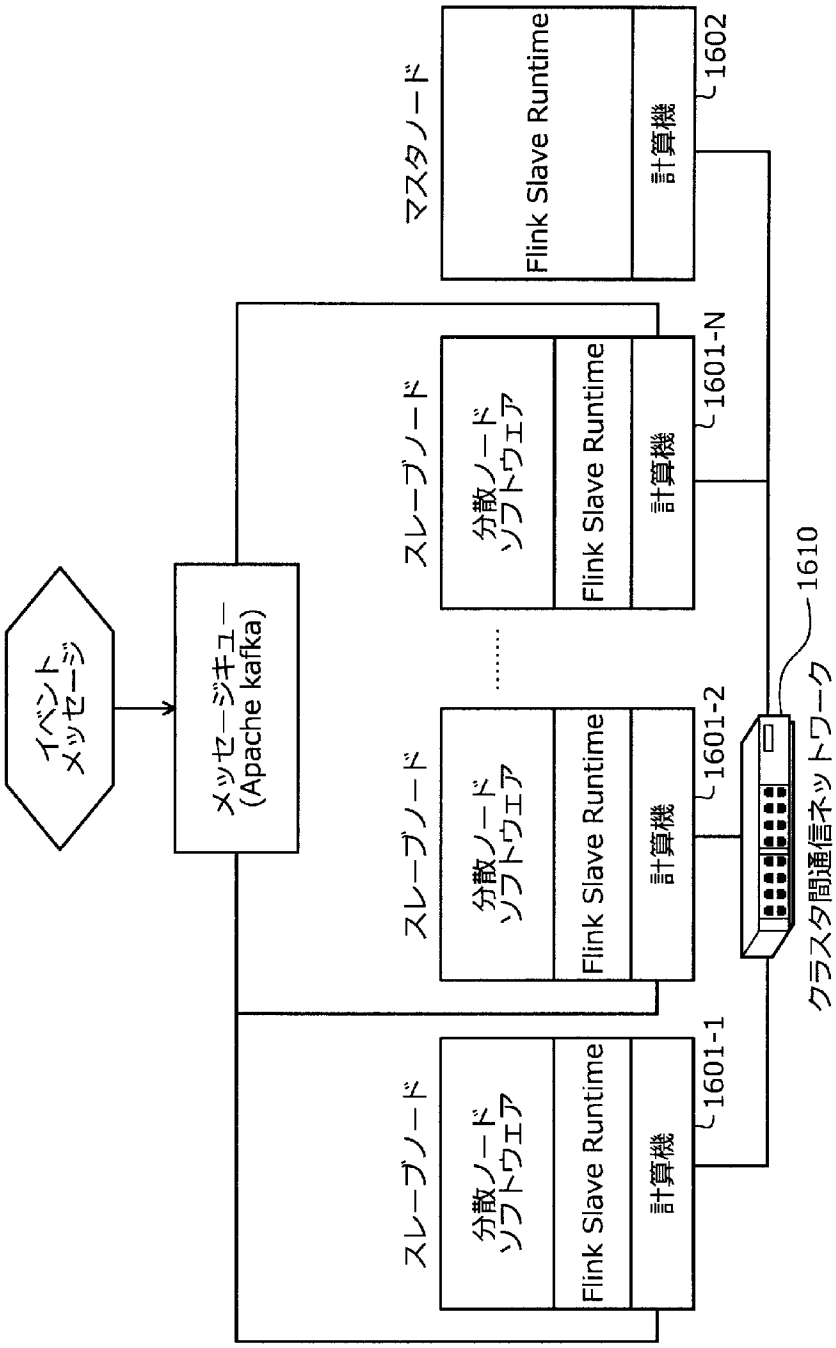
[図14]


201-3
500
400
5
S
B
6
S
D
4
S
C
B
PID
参照数
プラグインデータ
B
2
B
C
1
D
1
D
1101
PID
参照数
プラグインデータ
A
1
A
B
1
B
1102
PID
参照数
プラグインデータ
A
1
A
C
1
C
1103
PID
参照数
プラグインデータ
B
1
B
D
1
D
C を探索する
205
外部記憶

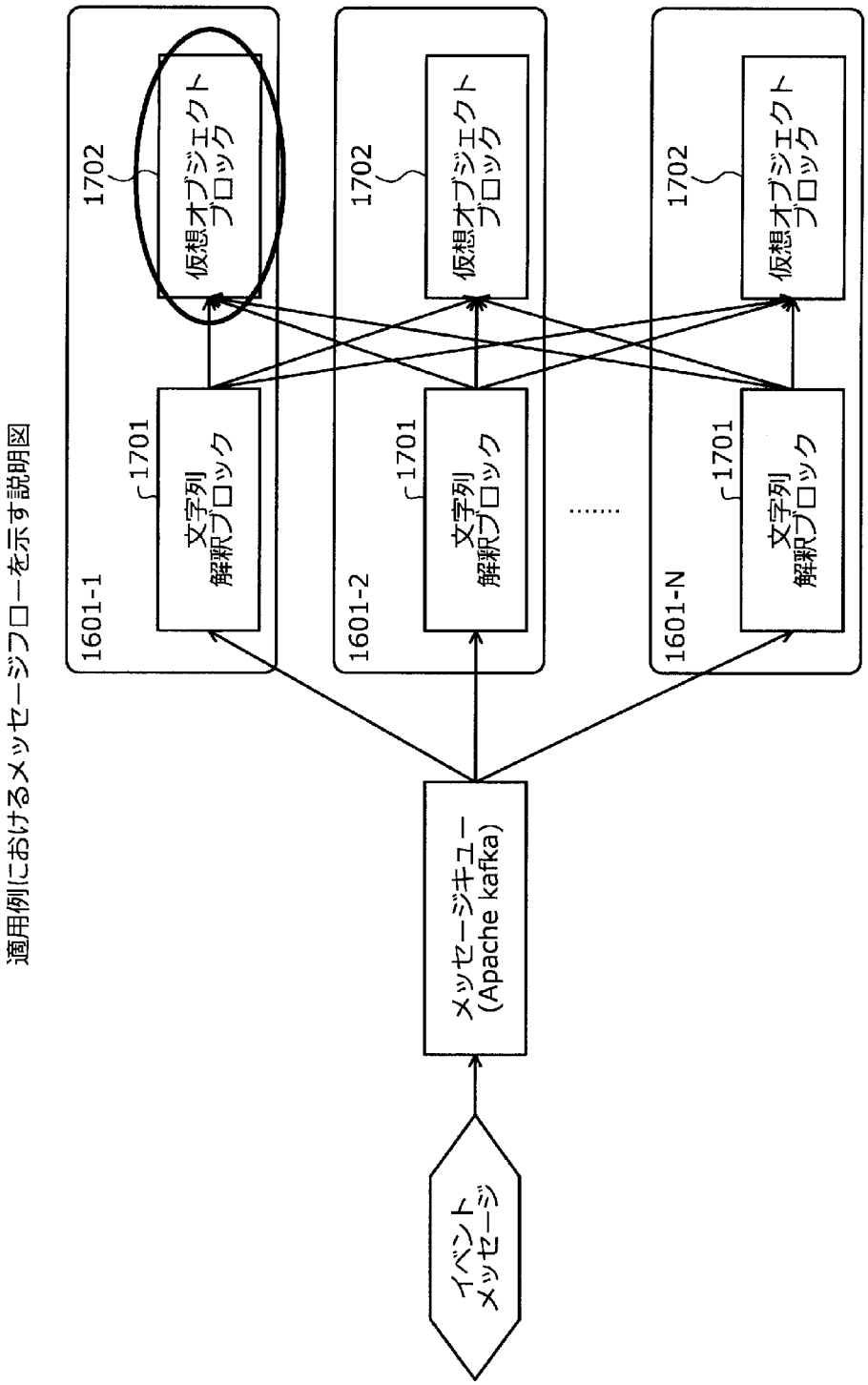
[図15]


201-3
500
400
5
S
B
6
S
D
4
S
C
B
PID
参照数
プラグインデータ
B
2
B
C
1
D
1
D
1101
PID
参照数
プラグインデータ
A
1
A
B
1
B
1102
PID
参照数
プラグインデータ
A
1
A
C
1
C
コピーする
1103
PID
参照数
プラグインデータ
B
1
B
D
1
D
205
外部記憶

[図16]

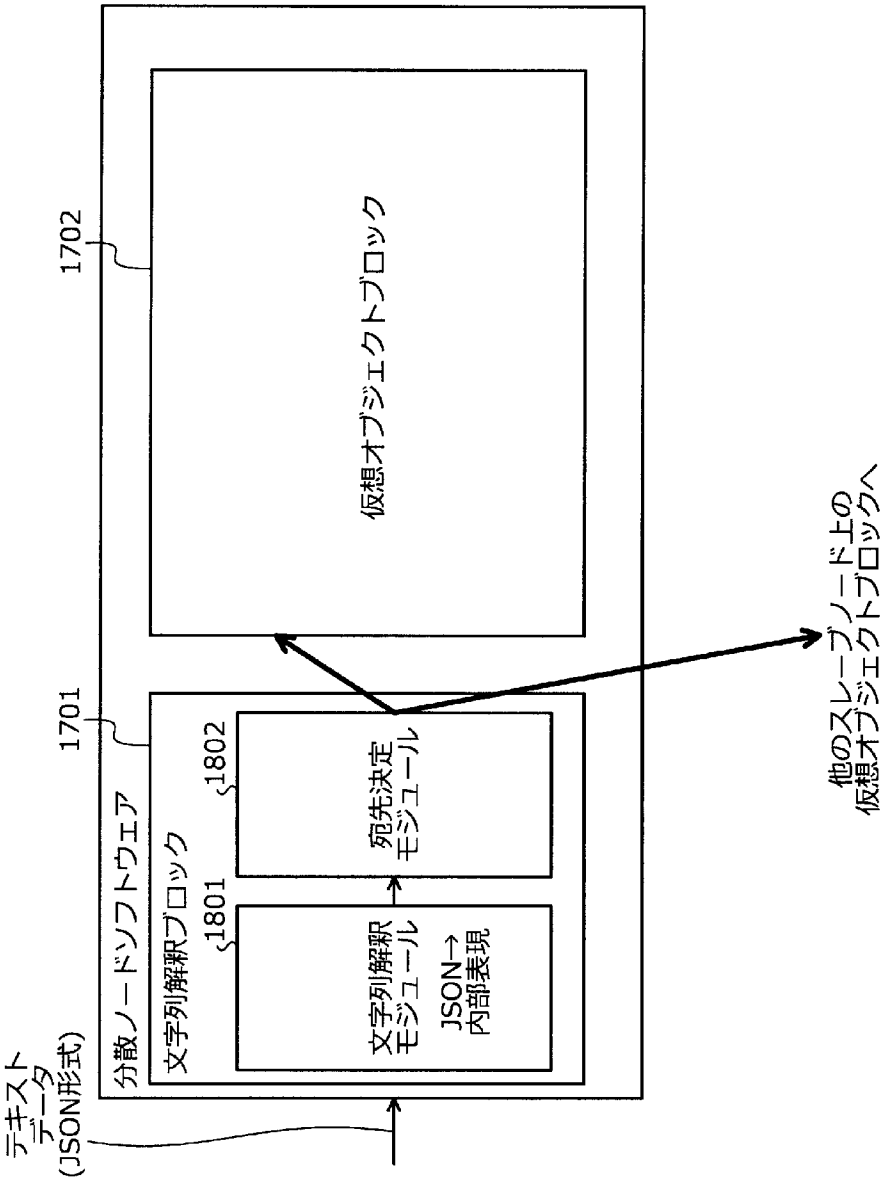

イベント
メッセージ
メッセージキュー
(Apache kafka)
マスタノード
Flink Slave Runtime
計算機
1602
スレーブノード
分散ノード
ソフトウェア
Flink Slave Runtime
計算機
1601-N
スレーブノード
分散ノード
ソフトウェア
Flink Slave Runtime
計算機
1601-2
スレーブノード
分散ノード
ソフトウェア
Flink Slave Runtime
計算機
1601-1
1610
クラスタ間通信ネットワーク

[図17]

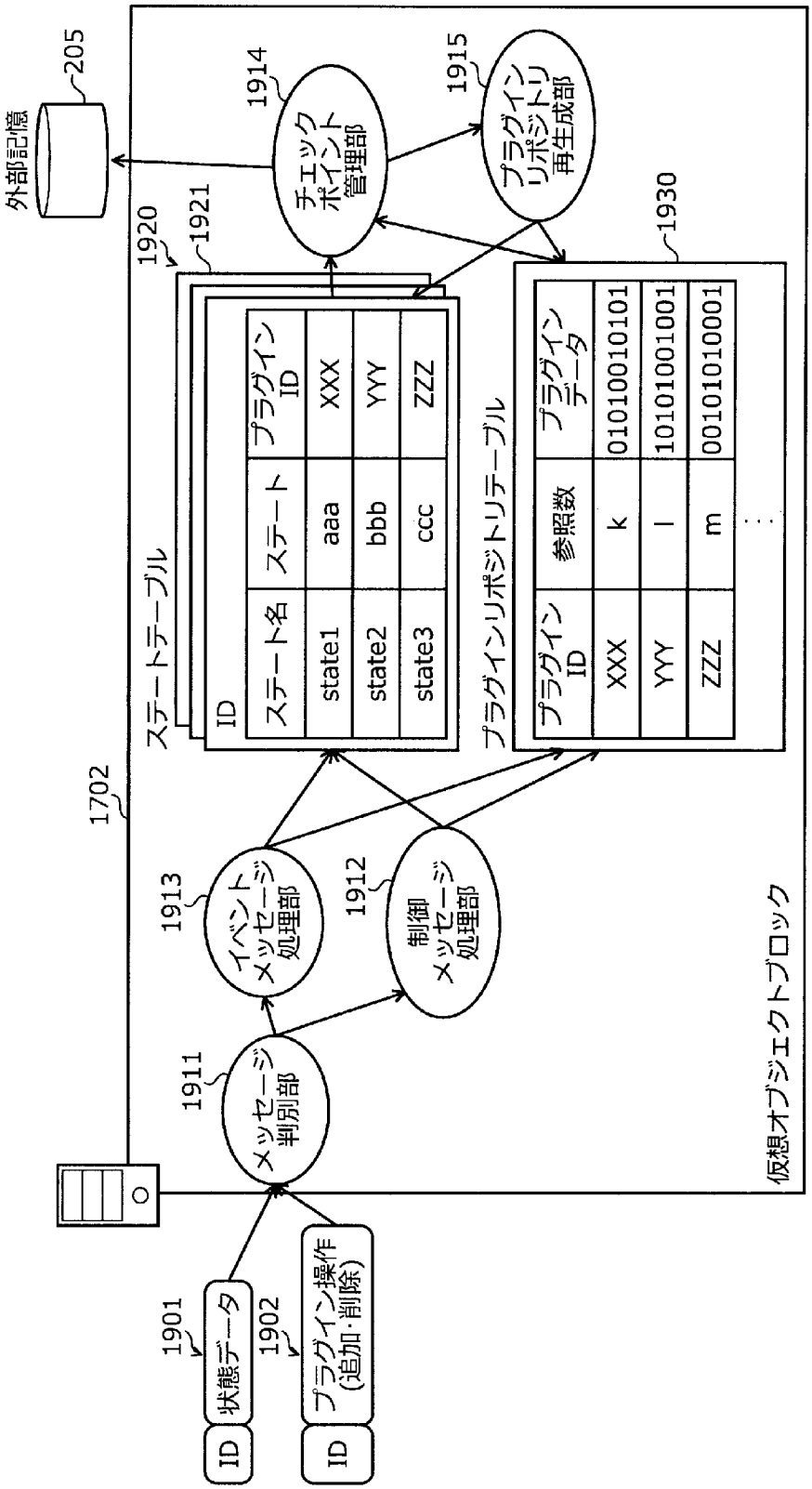


適用例におけるメッセージフローを示す説明図

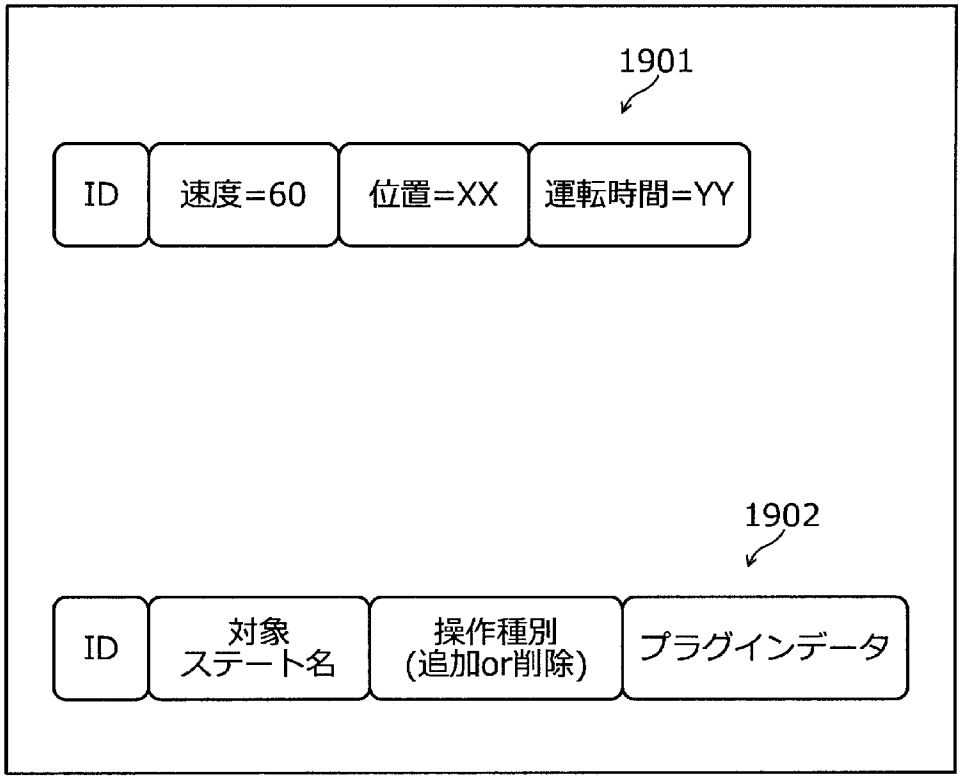
[図18]


テキスト
データ
(JSON形式)
1701
分散ノードソフトウェア
文字列解釈ブロック
1801
文字列解釈
モジュール
JSON→
内部表現
1802
宛先決定
モジュール
1702
仮想オブジェクトブロック
他のスレーブノード上の
仮想オブジェクトブロックへ

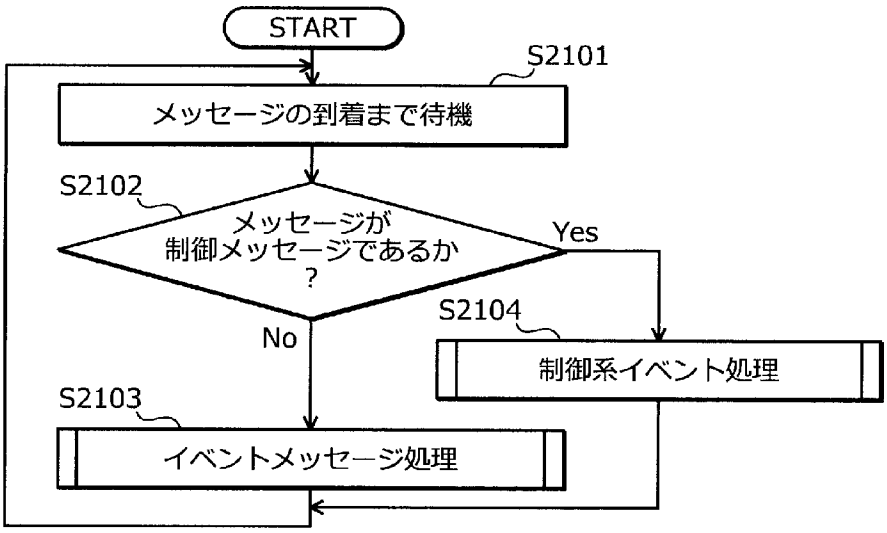
[図19]


外部記憶
205
1914
チェックポイント管理部
1915
プラグインリポジトリ再生成部
1920
1921
1930
ステートテーブル
ID
ステート名
ステート
プラグインID
state1
aaa
XXX
state2
bbb
YYY
state3
ccc
ZZZ
プラグインリポジトリテーブル
プラグインID
参照数
プラグインデータ
XXX
k
01010010101
YYY
l
10101001001
ZZZ
m
00101010001
1702
1913
イベントメッセージ処理部
1912
制御メッセージ処理部
1911
メッセージ判別部
仮想オブジェクトブロック
1901
状態データ
ID
1902
プラグイン操作(追加・削除)
ID

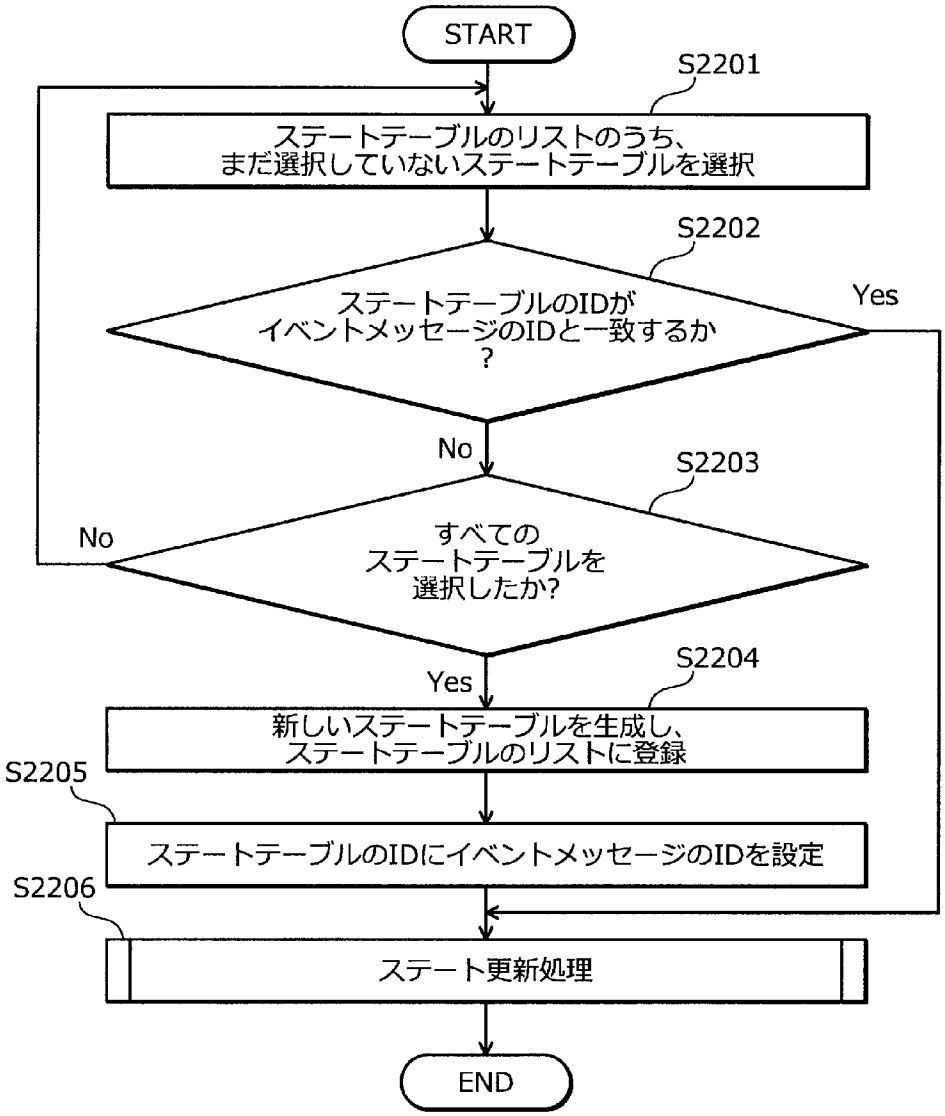
[図20]


1901
ID
速度=60
位置=XX
運転時間=YY
1902
ID
対象
ステート名
操作種別
(追加or削除)
プラグインデータ

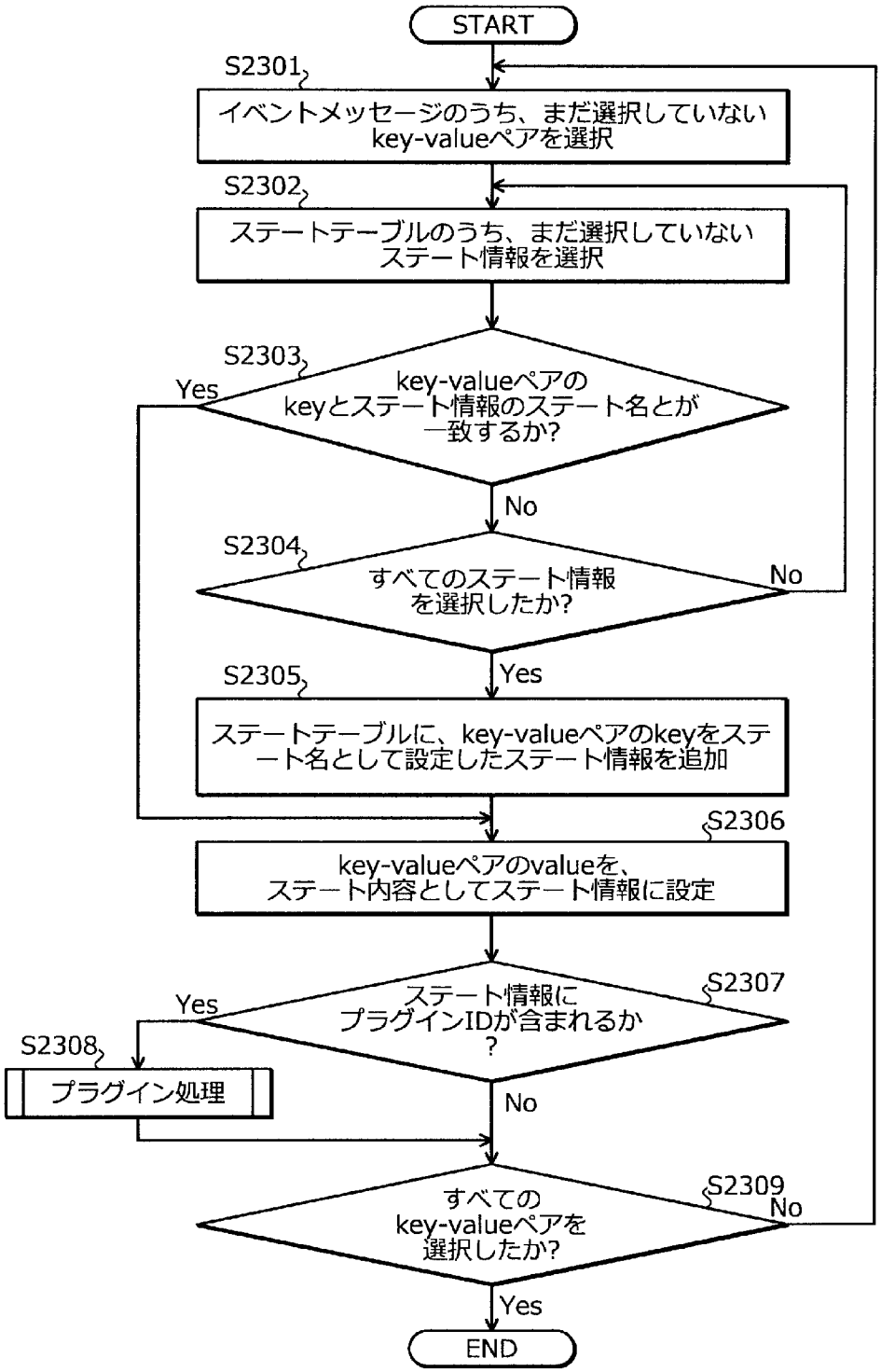
[図21]


START
S2101
メッセージの到着まで待機
S2102
メッセージが
制御メッセージであるか
?
Yes
No
S2104
制御系イベント処理
S2103
イベントメッセージ処理

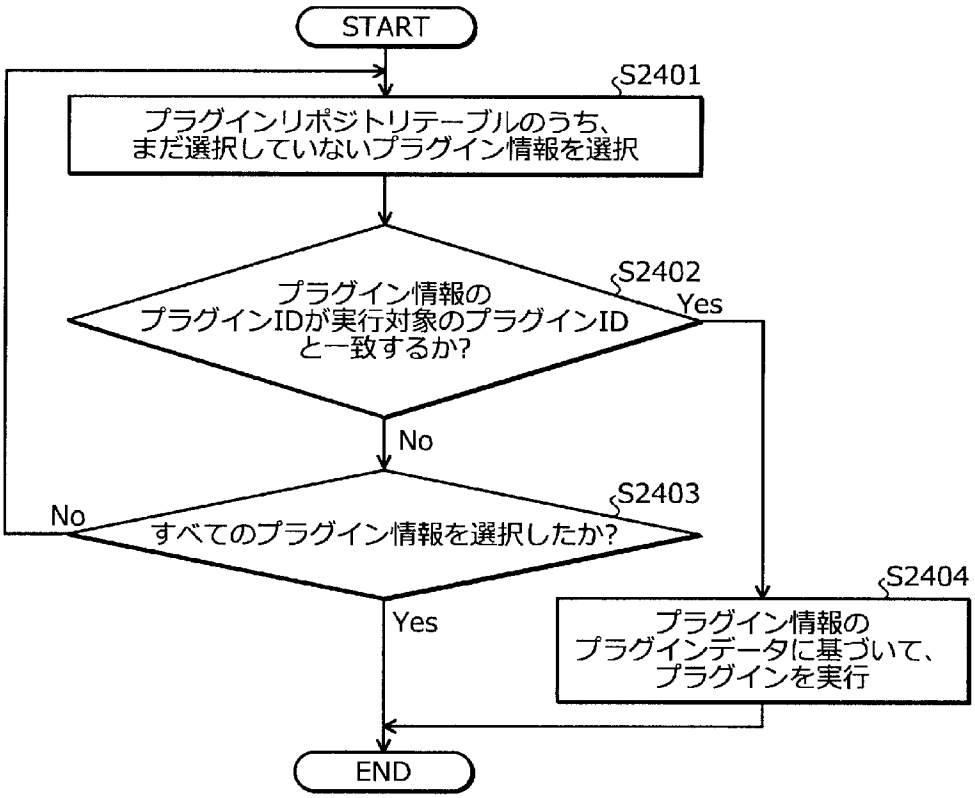
[図22]


START
S2201
ステートテーブルのリストのうち、
まだ選択していないステートテーブルを選択
S2202
ステートテーブルのIDが
イベントメッセージのIDと一致するか
?
Yes
No
S2203
すべての
ステートテーブルを
選択したか?
No
Yes
S2204
新しいステートテーブルを生成し、
ステートテーブルのリストに登録
S2205
ステートテーブルのIDにイベントメッセージのIDを設定
S2206
ステート更新処理
END

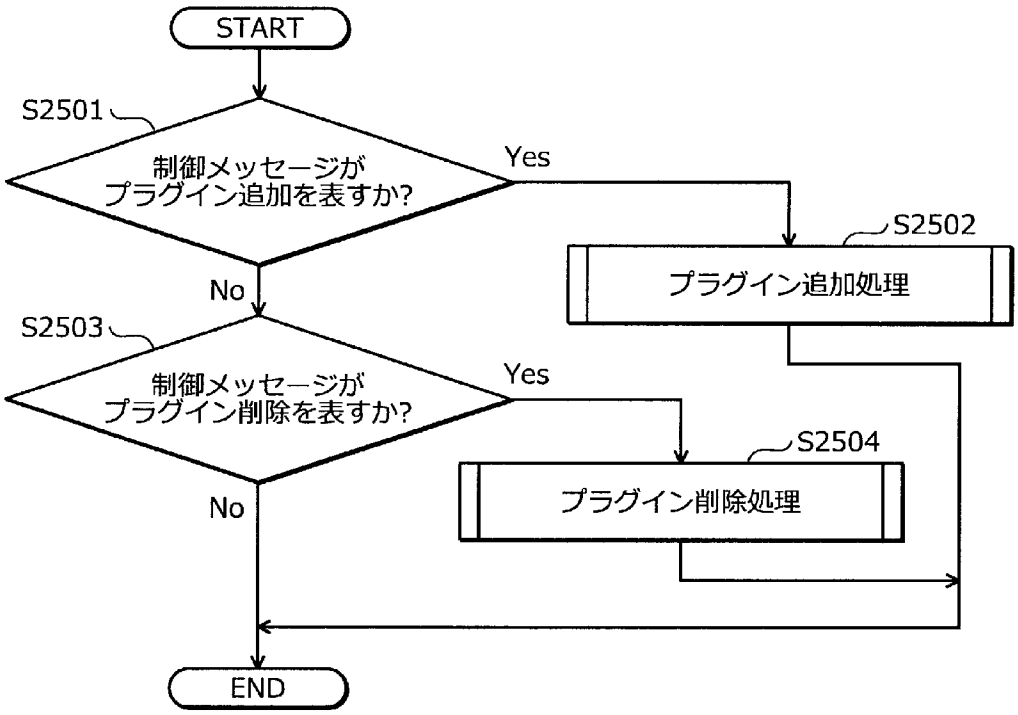
[図23]


START
S2301
イベントメッセージのうち、まだ選択していない key-valueペアを選択
S2302
ステートテーブルのうち、まだ選択していない ステート情報を選択
S2303
key-valueペアの keyとステート情報のステート名とが 一致するか?
Yes
No
S2304
すべてのステート情報 を選択したか?
No
Yes
S2305
ステートテーブルに、key-valueペアのkeyをステート名として設定したステート情報を追加
S2306
key-valueペアのvalueを、 ステート内容としてステート情報に設定
S2307
ステート情報に プラグインIDが含まれるか ?
Yes
S2308
プラグイン処理
No
S2309
すべての key-valueペアを 選択したか?
No
Yes
END

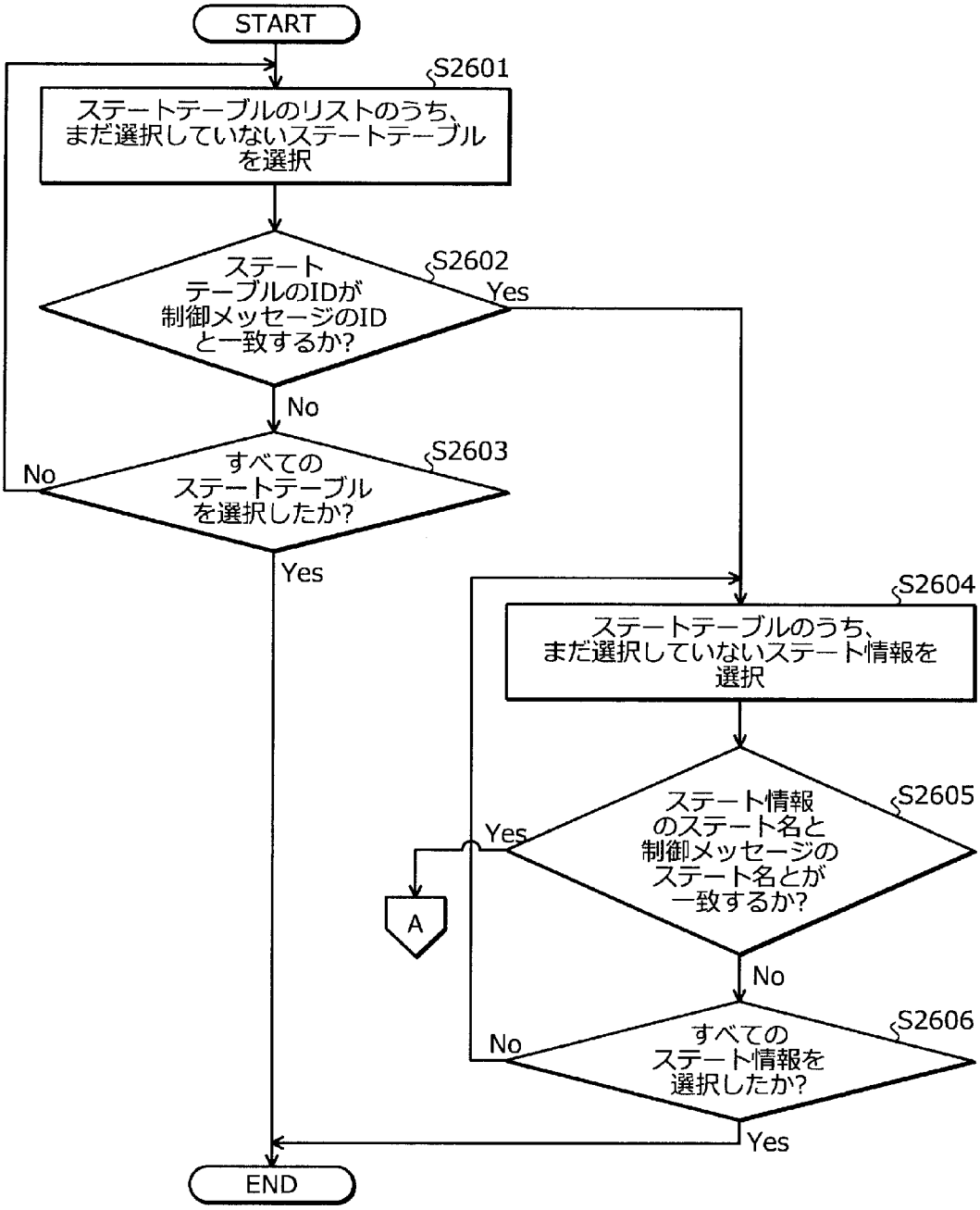
[図24]


START
S2401
プラグインリポジトリテーブルのうち、
まだ選択していないプラグイン情報を選択
S2402
プラグイン情報の
プラグインIDが実行対象のプラグインID
と一致するか?
Yes
No
S2403
No
すべてのプラグイン情報を選択したか?
Yes
S2404
プラグイン情報の
プラグインデータに基づいて、
プラグインを実行
END

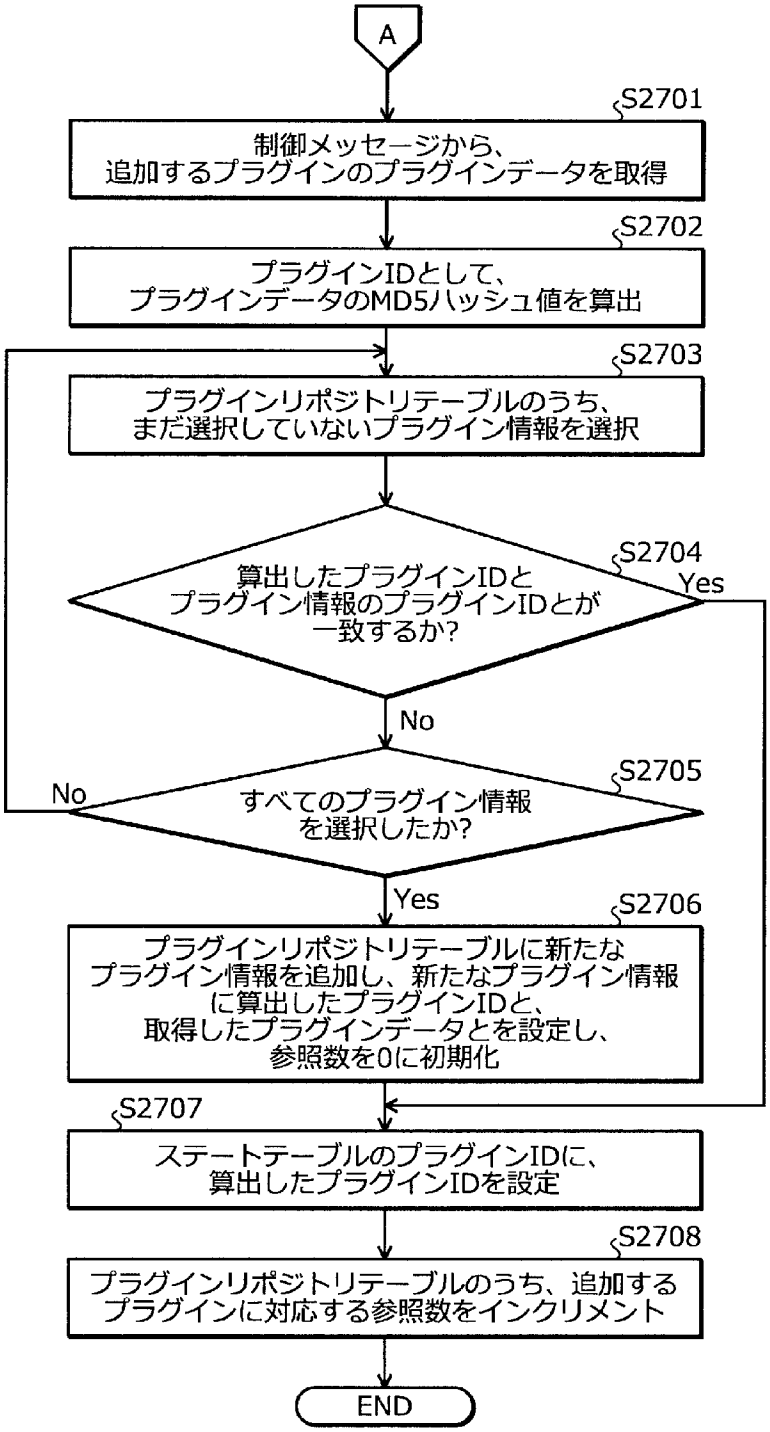
[図25]


START
S2501
制御メッセージが
プラグイン追加を表すか?
Yes
S2502
プラグイン追加処理
No
S2503
制御メッセージが
プラグイン削除を表すか?
Yes
S2504
プラグイン削除処理
No
END

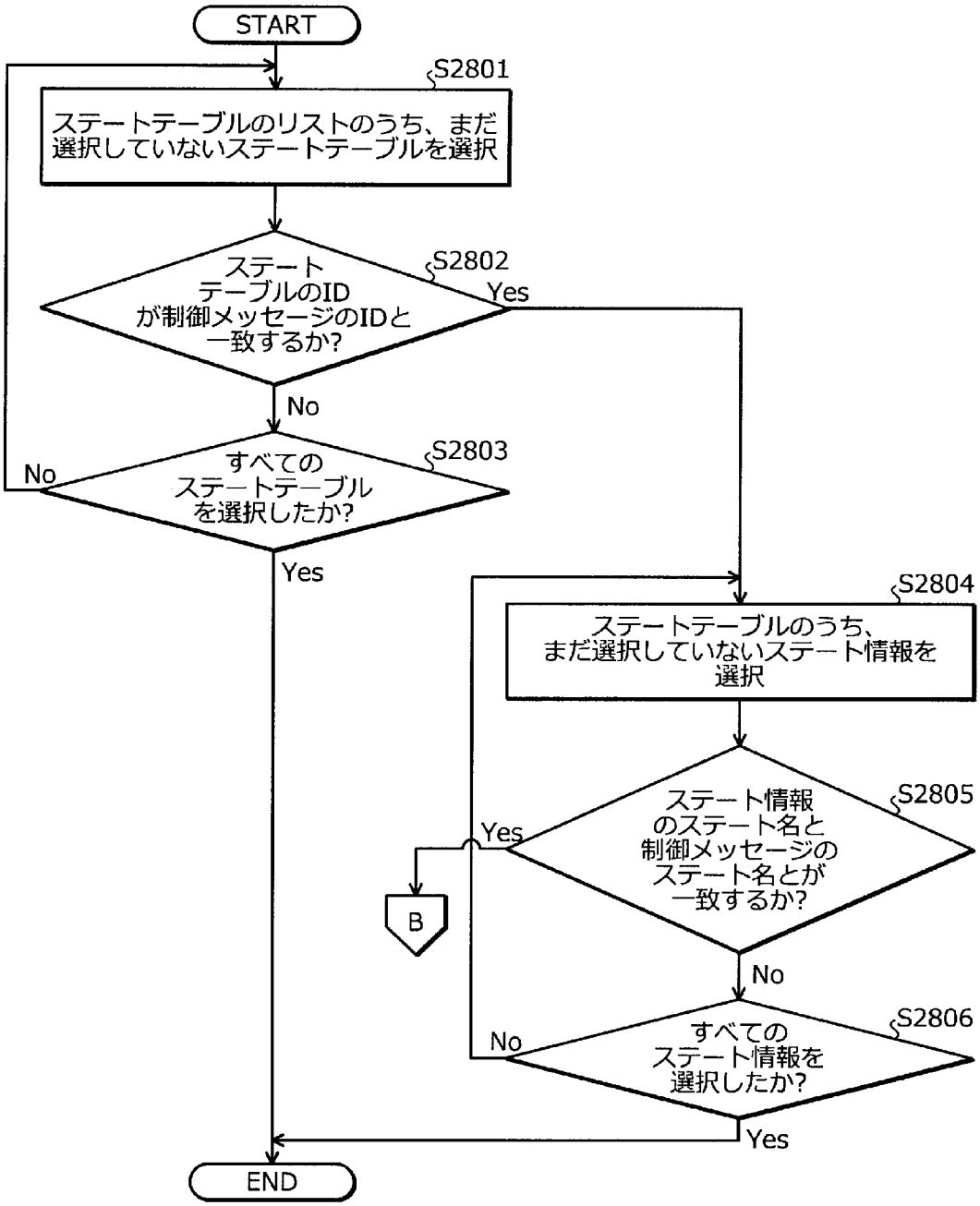
[図26]


START
S2601
ステートテーブルのリストのうち、まだ選択していないステートテーブルを選択
S2602
ステートテーブルのIDが制御メッセージのIDと一致するか?
Yes
No
S2603
すべてのステートテーブルを選択したか?
No
Yes
S2604
ステートテーブルのうち、まだ選択していないステート情報を選択
S2605
ステート情報のステート名と制御メッセージのステート名とが一致するか?
Yes
A
No
S2606
すべてのステート情報を選択したか?
No
Yes
END

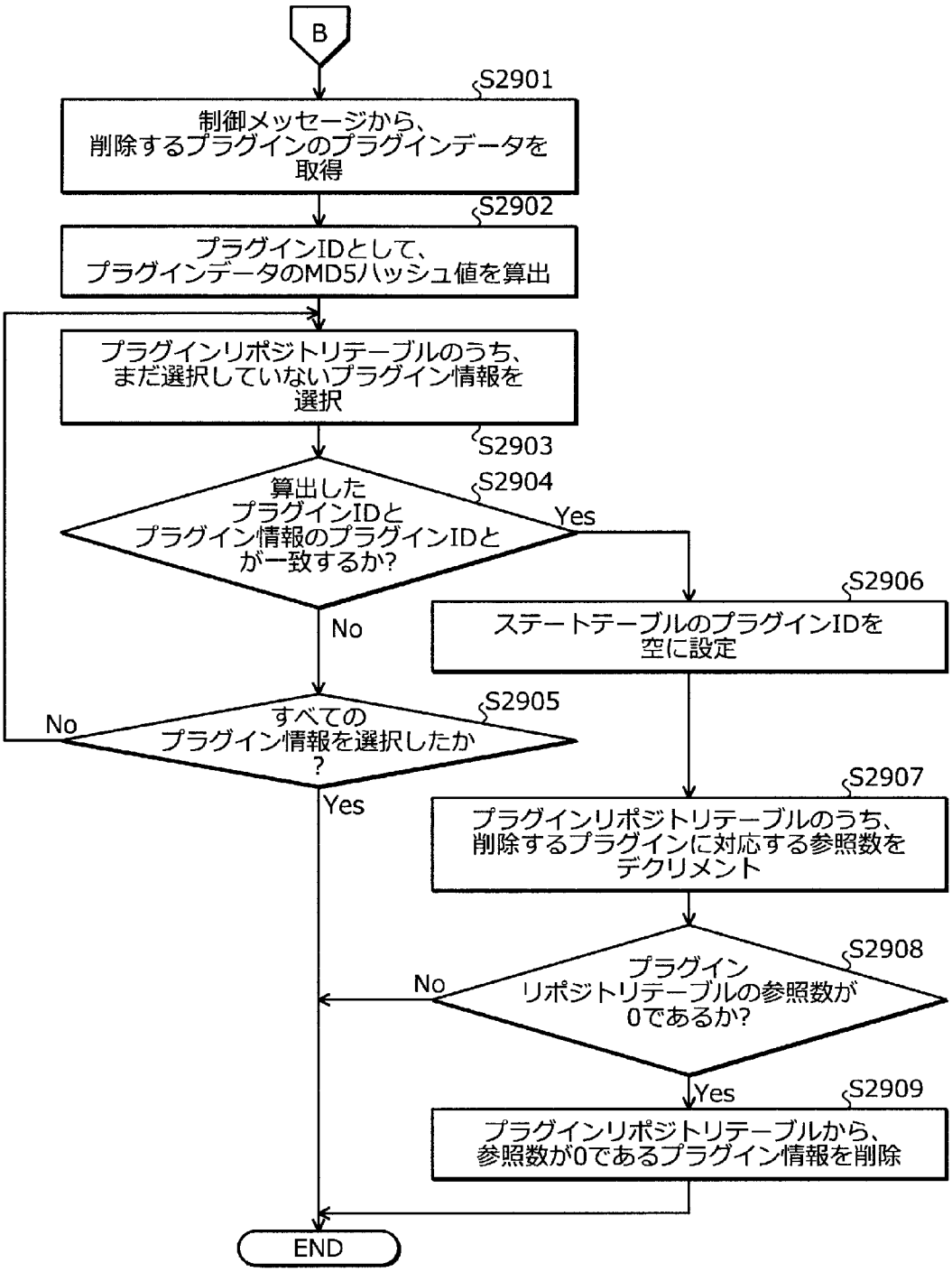
[図27]


A
S2701
制御メッセージから、
追加するプラグインのプラグインデータを取得
S2702
プラグインIDとして、
プラグインデータのMD5ハッシュ値を算出
S2703
プラグインリポジトリテーブルのうち、
まだ選択していないプラグイン情報を選択
S2704
算出したプラグインIDと
プラグイン情報のプラグインIDとが
一致するか?
Yes
No
S2705
すべてのプラグイン情報
を選択したか?
No
Yes
S2706
プラグインリポジトリテーブルに新たな
プラグイン情報を追加し、新たなプラグイン情報
に算出したプラグインIDと、
取得したプラグインデータとを設定し、
参照数を0に初期化
S2707
ステートテーブルのプラグインIDに、
算出したプラグインIDを設定
S2708
プラグインリポジトリテーブルのうち、追加する
プラグインに対応する参照数をインクリメント
END

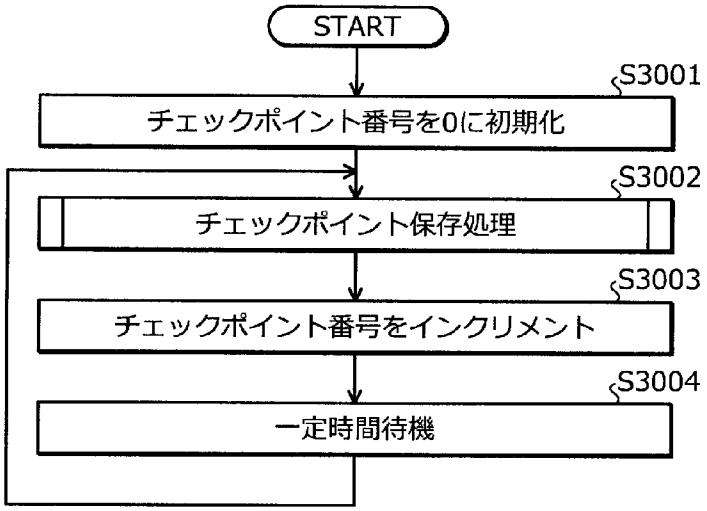
[図28]


START
S2801
ステートテーブルのリストのうち、まだ選択していないステートテーブルを選択
S2802
ステートテーブルのIDが制御メッセージのIDと一致するか?
Yes
No
S2803
すべてのステートテーブルを選択したか?
No
Yes
S2804
ステートテーブルのうち、まだ選択していないステート情報を選択
S2805
ステート情報のステート名と制御メッセージのステート名とが一致するか?
Yes
B
No
S2806
すべてのステート情報を選択したか?
No
Yes
END

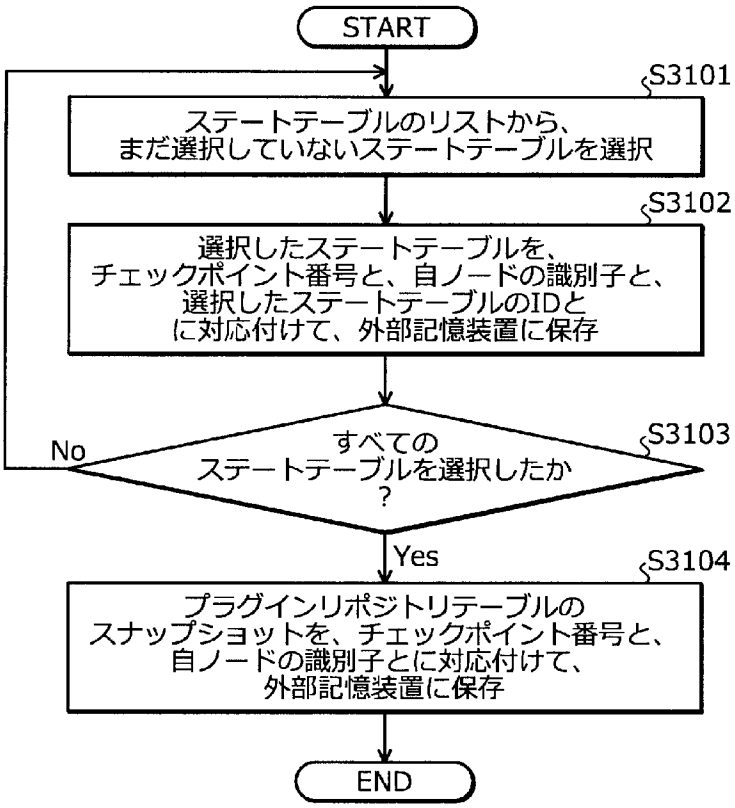
[図29]


B
S2901
制御メッセージから、
削除するプラグインのプラグインデータを
取得
S2902
プラグインIDとして、
プラグインデータのMD5ハッシュ値を算出
プラグインリポジトリテーブルのうち、
まだ選択していないプラグイン情報を
選択
S2903
S2904
算出した
プラグインIDと
プラグイン情報のプラグインIDと
が一致するか?
Yes
No
S2906
ステートテーブルのプラグインIDを
空に設定
S2905
すべての
プラグイン情報を選択したか
?
No
Yes
S2907
プラグインリポジトリテーブルのうち、
削除するプラグインに対応する参照数を
デクリメント
S2908
プラグイン
リポジトリテーブルの参照数が
0であるか?
No
Yes
S2909
プラグインリポジトリテーブルから、
参照数が0であるプラグイン情報を削除
END

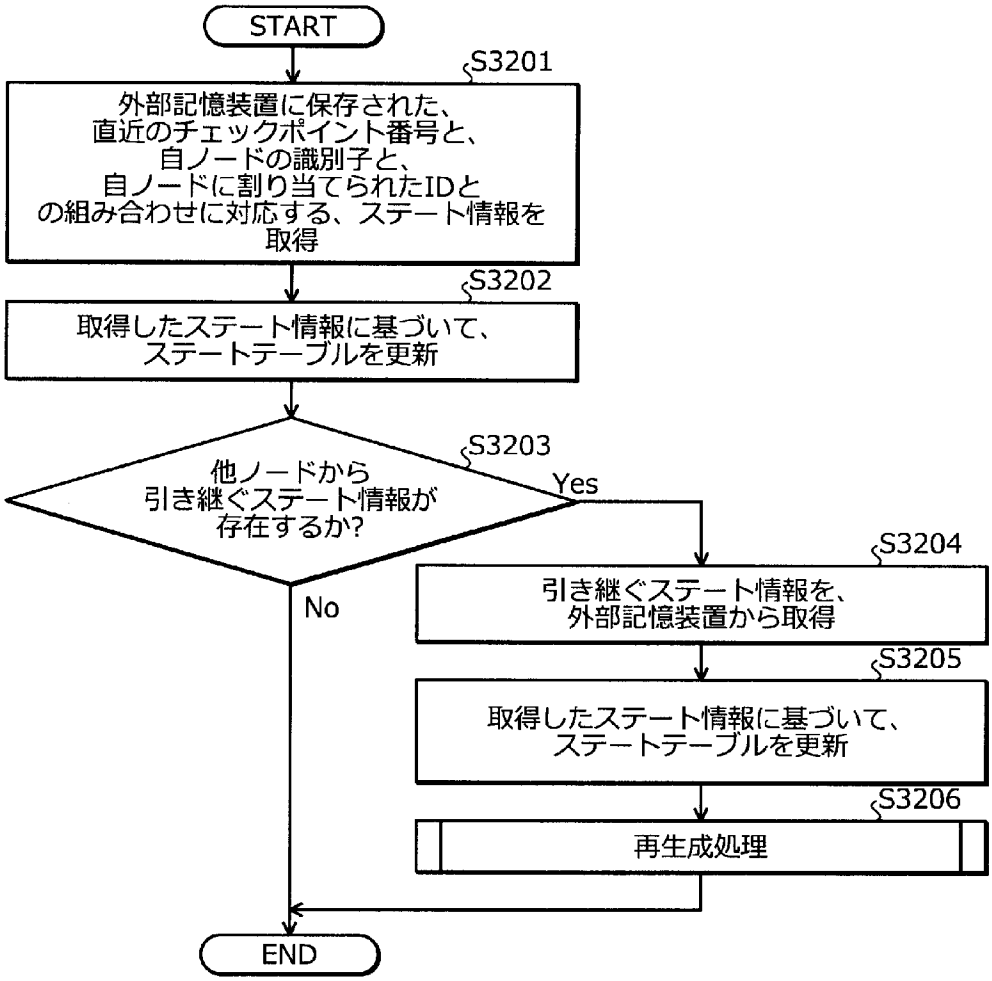
[図30]


START
S3001
チェックポイント番号を0に初期化
S3002
チェックポイント保存処理
S3003
チェックポイント番号をインクリメント
S3004
一定時間待機

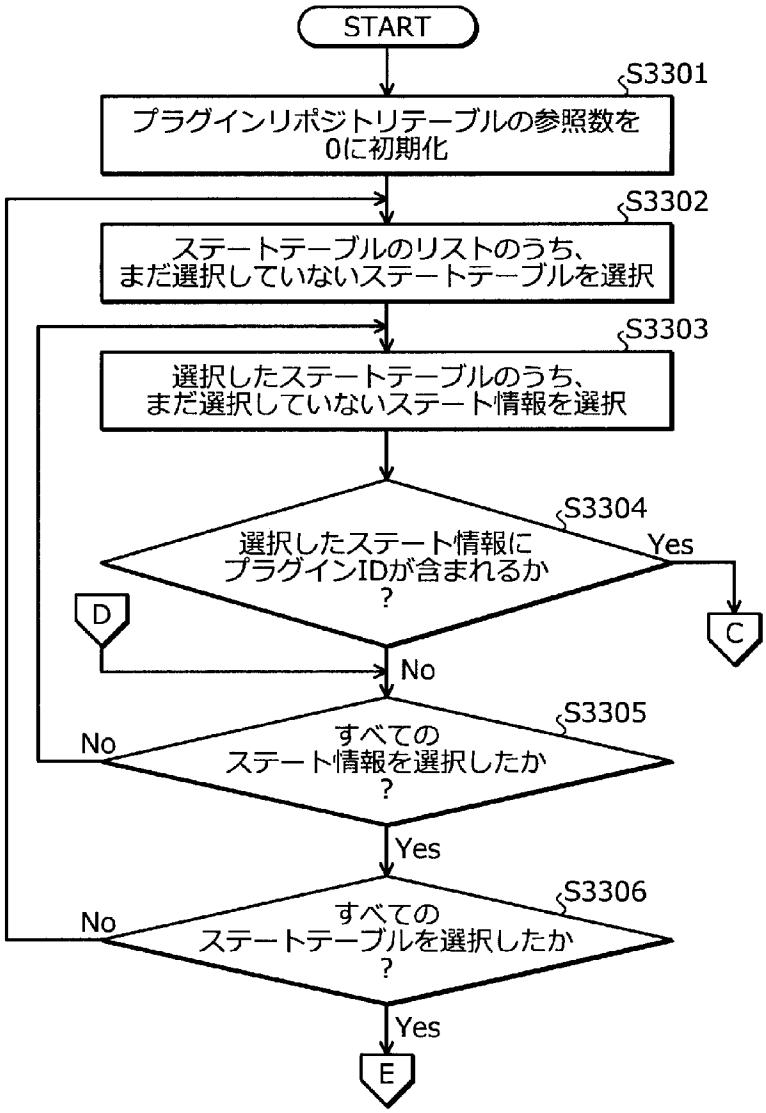
[図31]


START
S3101
ステートテーブルのリストから、
まだ選択していないステートテーブルを選択
S3102
選択したステートテーブルを、
チェックポイント番号と、自ノードの識別子と、
選択したステートテーブルのIDと
に対応付けて、外部記憶装置に保存
S3103
すべての
ステートテーブルを選択したか
?
No
Yes
S3104
プラグインリポジトリテーブルの
スナップショットを、チェックポイント番号と、
自ノードの識別子とに対応付けて、
外部記憶装置に保存
END

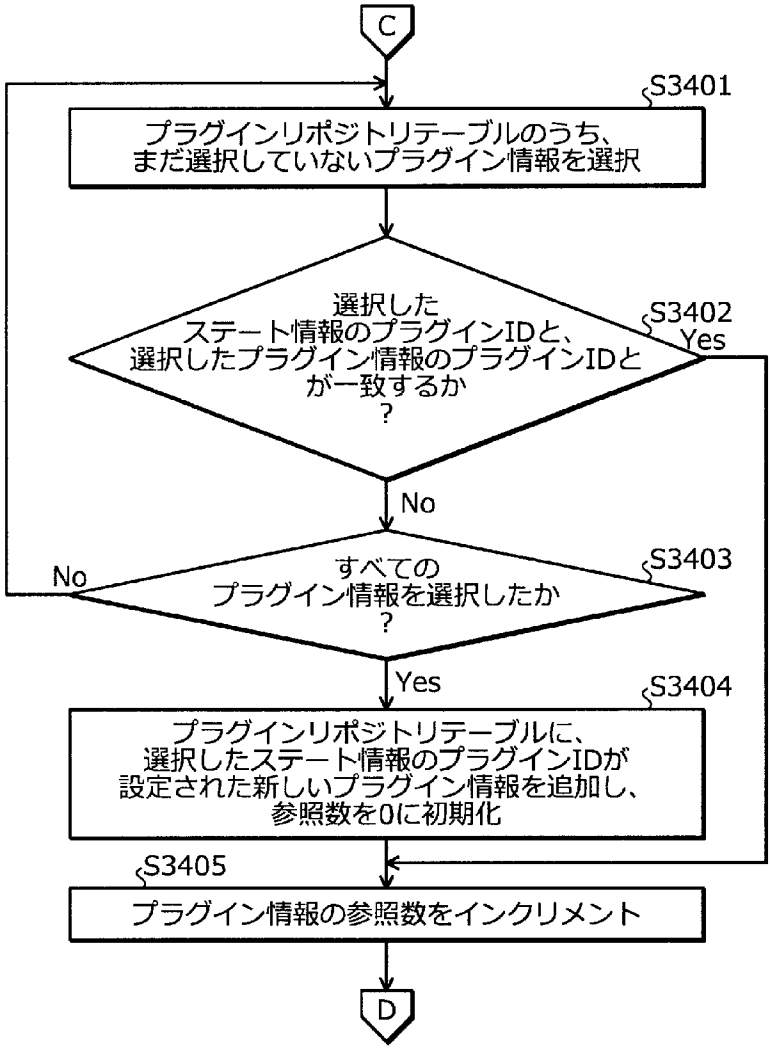
[図32]


START
S3201
外部記憶装置に保存された、
直近のチェックポイント番号と、
自ノードの識別子と、
自ノードに割り当てられたIDと
の組み合わせに対応する、ステート情報を
取得
S3202
取得したステート情報に基づいて、
ステートテーブルを更新
S3203
他ノードから
引き継ぐステート情報が
存在するか?
Yes
No
S3204
引き継ぐステート情報を、
外部記憶装置から取得
S3205
取得したステート情報に基づいて、
ステートテーブルを更新
S3206
再生成処理
END

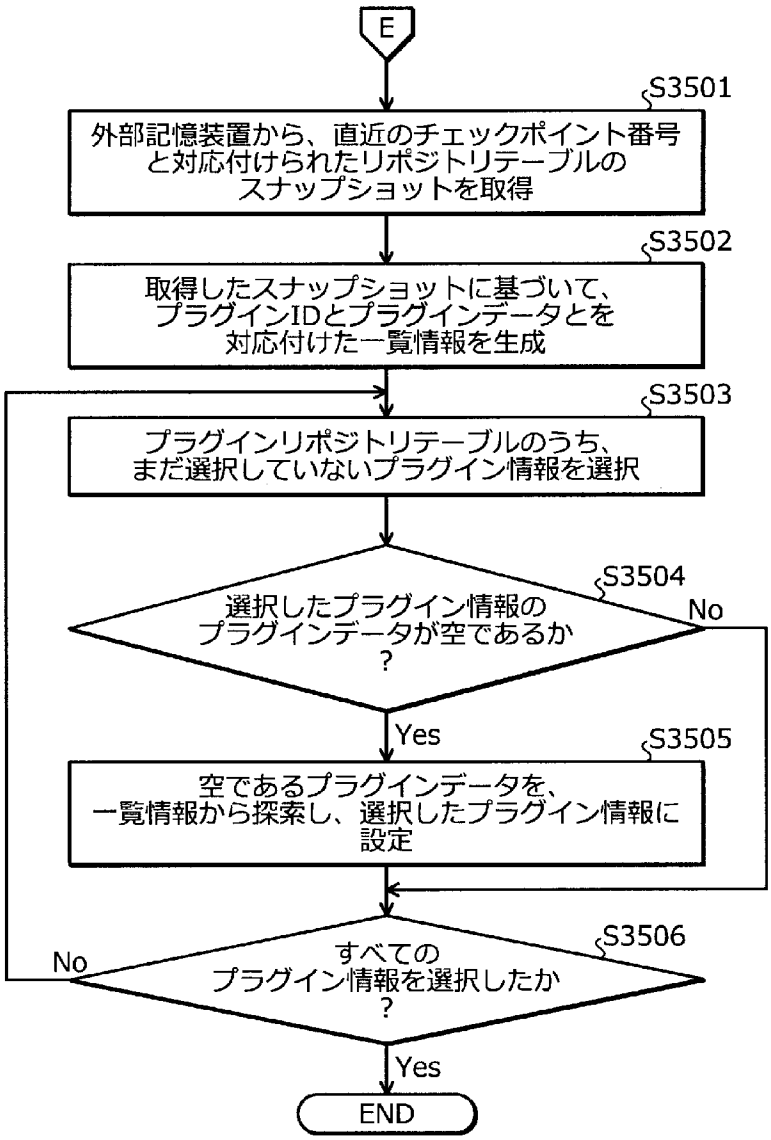
[図33]


START
S3301
プラグインリポジトリテーブルの参照数を
0に初期化
S3302
ステートテーブルのリストのうち、
まだ選択していないステートテーブルを選択
S3303
選択したステートテーブルのうち、
まだ選択していないステート情報を選択
S3304
選択したステート情報に
プラグインIDが含まれるか
?
Yes
C
D
No
S3305
すべての
ステート情報を選択したか
?
No
Yes
S3306
すべての
ステートテーブルを選択したか
?
No
Yes
E

[図34]


C
S3401
プラグインリポジトリテーブルのうち、
まだ選択していないプラグイン情報を選択
S3402
選択した
ステート情報のプラグインIDと、
選択したプラグイン情報のプラグインIDと
が一致するか
?
Yes
No
S3403
すべての
プラグイン情報を選択したか
?
No
Yes
S3404
プラグインリポジトリテーブルに、
選択したステート情報のプラグインIDが
設定された新しいプラグイン情報を追加し、
参照数を0に初期化
S3405
プラグイン情報の参照数をインクリメント
D

[図35]


E
S3501
外部記憶装置から、直近のチェックポイント番号と対応付けられたリポジトリテーブルのスナップショットを取得
S3502
取得したスナップショットに基づいて、プラグインIDとプラグインデータとを対応付けた一覧情報を生成
S3503
プラグインリポジトリテーブルのうち、まだ選択していないプラグイン情報を選択
S3504
選択したプラグイン情報のプラグインデータが空であるか？
No
Yes
S3505
空であるプラグインデータを、一覧情報から探索し、選択したプラグイン情報に設定
S3506
すべてのプラグイン情報を選択したか？
No
Yes
END

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP2019/000977

A. CLASSIFICATION OF SUBJECT MATTER
Int.Cl. G06F11/14(2006.01)i, G06F9/50(2006.01)i

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)
Int.Cl. G06F11/14, G06F9/50

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Published examined utility model applications of Japan	1922-1996
Published unexamined utility model applications of Japan	1971-2019
Registered utility model specifications of Japan	1996-2019
Published registered utility model applications of Japan	1994-2019

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2011-76128 A (HITACHI, LTD.) 14 April 2011, paragraphs [0060]-[0064], [0205], [0217], fig. 1, 4-5 & US 2011/0078494 A1, paragraphs [0099]-[0103], [0256], [0269], fig. 1, 4-5	1-10
A	JP 2007-310807 A (RICOH CO., LTD.) 29 November 2007, paragraphs [0002], [0010], [0023], [0026]-[0027], [0033], [0036]-[0041] (Family: none)	1-10

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

☐ See patent family annex.

* 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
04 April 2019 (04.04.2019)	16 April 2019 (16.04.2019)
Name and mailing address of the ISA/ Japan Patent Office 3-4-3, Kasumigaseki, Chiyoda-ku, Tokyo 100-8915, Japan	Authorized officer Telephone No.

国際調査報告

国際出願番号　ＰＣＴ／ＪＰ２０１９／０００９７７

Ａ．　発明の属する分野の分類（国際特許分類（ＩＰＣ））

Int.Cl.　G06F11/14(2006.01)i, G06F9/50(2006.01)i

Ｂ．　調査を行った分野

調査を行った最小限資料（国際特許分類（ＩＰＣ））

Int.Cl.　G06F11/14, G06F9/50

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	１９２２－１９９６年
日本国公開実用新案公報	１９７１－２０１９年
日本国実用新案登録公報	１９９６－２０１９年
日本国登録実用新案公報	１９９４－２０１９年

国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）

Ｃ．　関連すると認められる文献

引用文献のカテゴリー＊	引用文献名　及び一部の箇所が関連するときは、その関連する箇所の表示	関連する請求項の番号
A	JP 2011-76128 A (株式会社日立製作所) 2011.04.14, 段落[0060]-[0064],[0205],[0217],図1,図4-5 & US 2011/0078494 A1, 段落[0099]-[0103],[0256],[0269], 図1,図4-5	1-10
A	JP 2007-310807 A (株式会社リコー) 2007.11.29, 段落[0002],[0010],[0023],[0026]-[0027],[0033],[0036]-[0041] (ファミリーなし)	1-10

☐　Ｃ欄の続きにも文献が列挙されている。

☐　パテントファミリーに関する別紙を参照。

＊　引用文献のカテゴリー

「Ａ」特に関連のある文献ではなく、一般的技術水準を示すもの

「Ｅ」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの

「Ｌ」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）

「Ｏ」口頭による開示、使用、展示等に言及する文献

「Ｐ」国際出願日前で、かつ優先権の主張の基礎となる出願の日の後に公表された文献

「Ｔ」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「Ｘ」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Ｙ」特に関連のある文献であって、当該文献と他の１以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」同一パテントファミリー文献

国際調査を完了した日	国際調査報告の発送日
０４．０４．２０１９	１６．０４．２０１９

国際調査機関の名称及びあて先
日本国特許庁（ＩＳＡ／ＪＰ）
郵便番号１００－８９１５
東京都千代田区霞が関三丁目４番３号

特許庁審査官（権限のある職員）　５Ｂ　６２９６
加藤　優一
電話番号　０３－３５８１－１１０１　内線　３５４５