障害事例から教訓を導く例

～保守関連の障害～

独立行政法人情報処理推進機構（ＩＰＡ）
技術本部 ソフトウェア高信頼化センター（ＳＥＣ）

Information-technology Promotion Agency, Japan
Software Reliability Enhancement Center (SEC)

注) 本資料の内容は，実例をベースにしていますが，一部，推測を含みます．

障害事例A（1/6）

【問題（障害内容）】

（個別）コンテキスト

◆概要

20xx年m月dd日，ある鉄道会社にて，列車運行を管理するシステムのうち，ダイヤに基づき各駅の信号機を自動制御する「自動進路制御装置(PRC)」の保守時に異常が発生．その後バックアップ系統への切替えに失敗し、装置が停止．管轄する3路線の鉄道が2時間以上運行できなくなり，385本が運休，約11.1万人に影響．

◆状況

午前4時，PRCの部品（今回の要因となった部品とは別のもの）の交換作業を始めると，アラームが鳴動し動作停止．

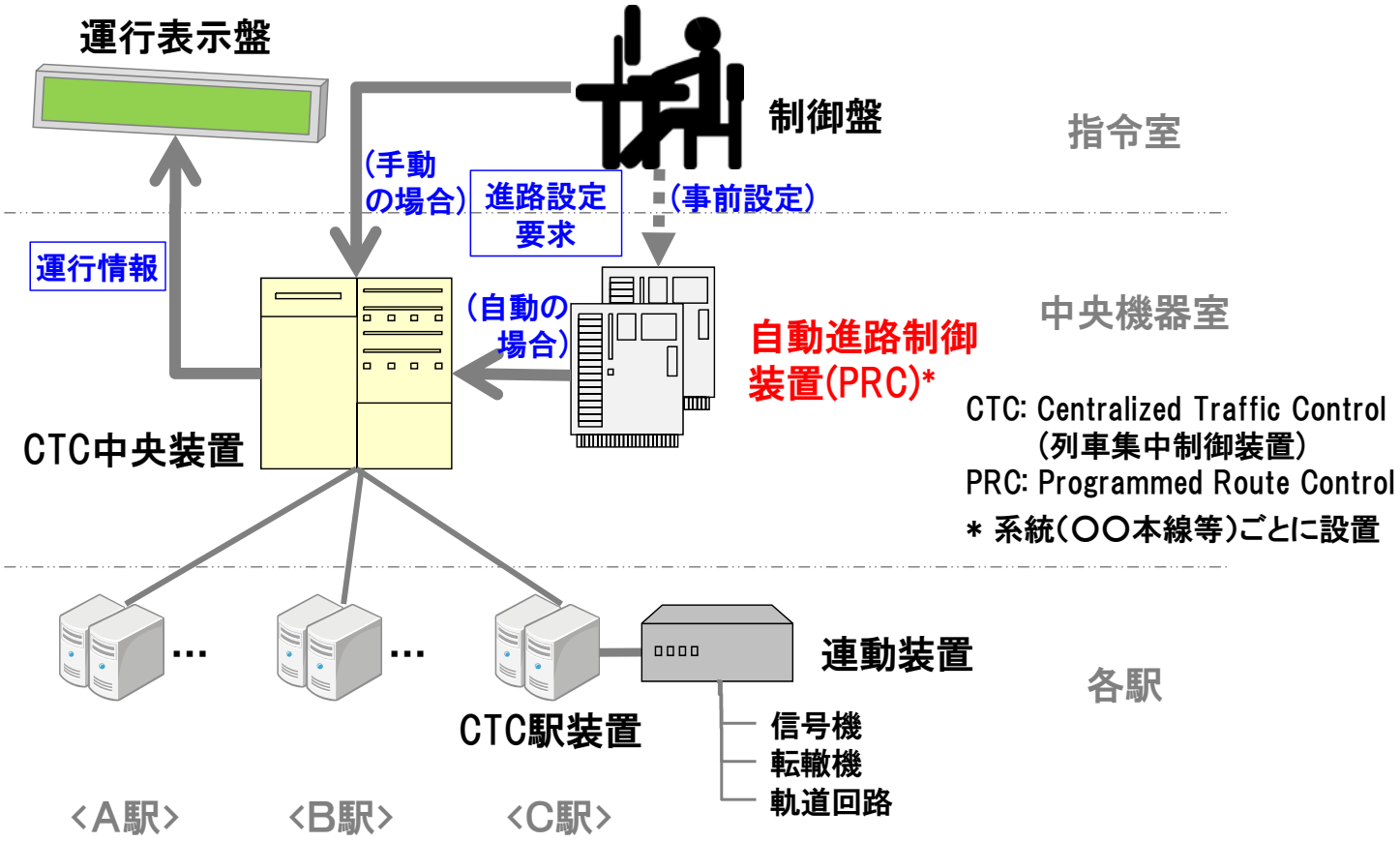
復旧のため，機器内に２つある処理系統のうち上記部品交換を行ったのとは別の系統（バックアップ系統）を起動させようとしたものの，起動せず．

交換対象の部品を戻して再起動する等を試みたが，PRCは復旧せず．

５時半頃に予備のPRCを使ってシステム全体の復旧に取り掛かり，６時54分に復旧．７時半頃から順次運転を再開．

＜参考＞
新聞等の報道記事

障害事例A (2/6)　列車運行制御システムの概要



障害事例A (3/6)　直接原因

【（直接の）原因】　**(個別) コンテキスト**

PRCのディスクとして，当初はHDDが使用されていたが，3年後に，インタフェース互換のあるSSDに交換された．制御OSの変更はなかった．【設計】

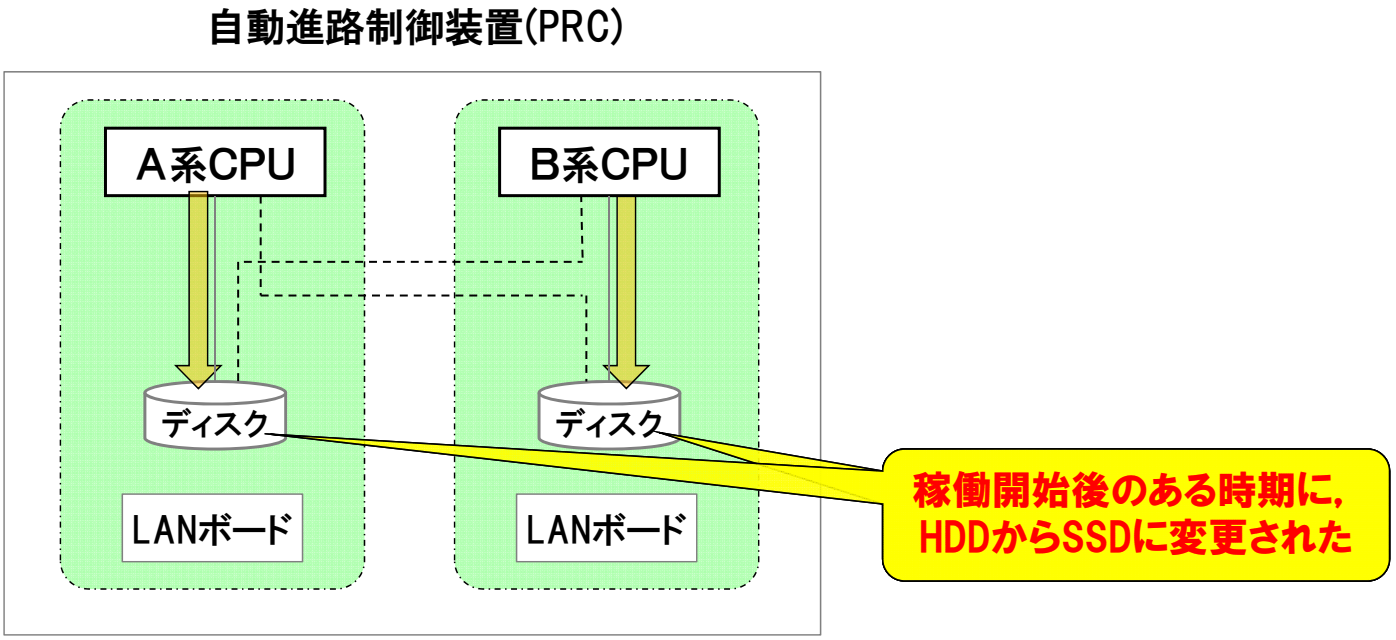
交換当時は，現用・待機両系停止の状態で装置を起動する試験のみを実施し，潜在リスクを発見できなかった．【試験】

その3年後，PRC内部のある故障部品を交換する必要が生じ，現用系のみを停止して実施することとした．工場での事前確認では，同じ装置がなかったため，HDD搭載装置での試験を行い，SSD搭載装置ではできなかった．【運用】

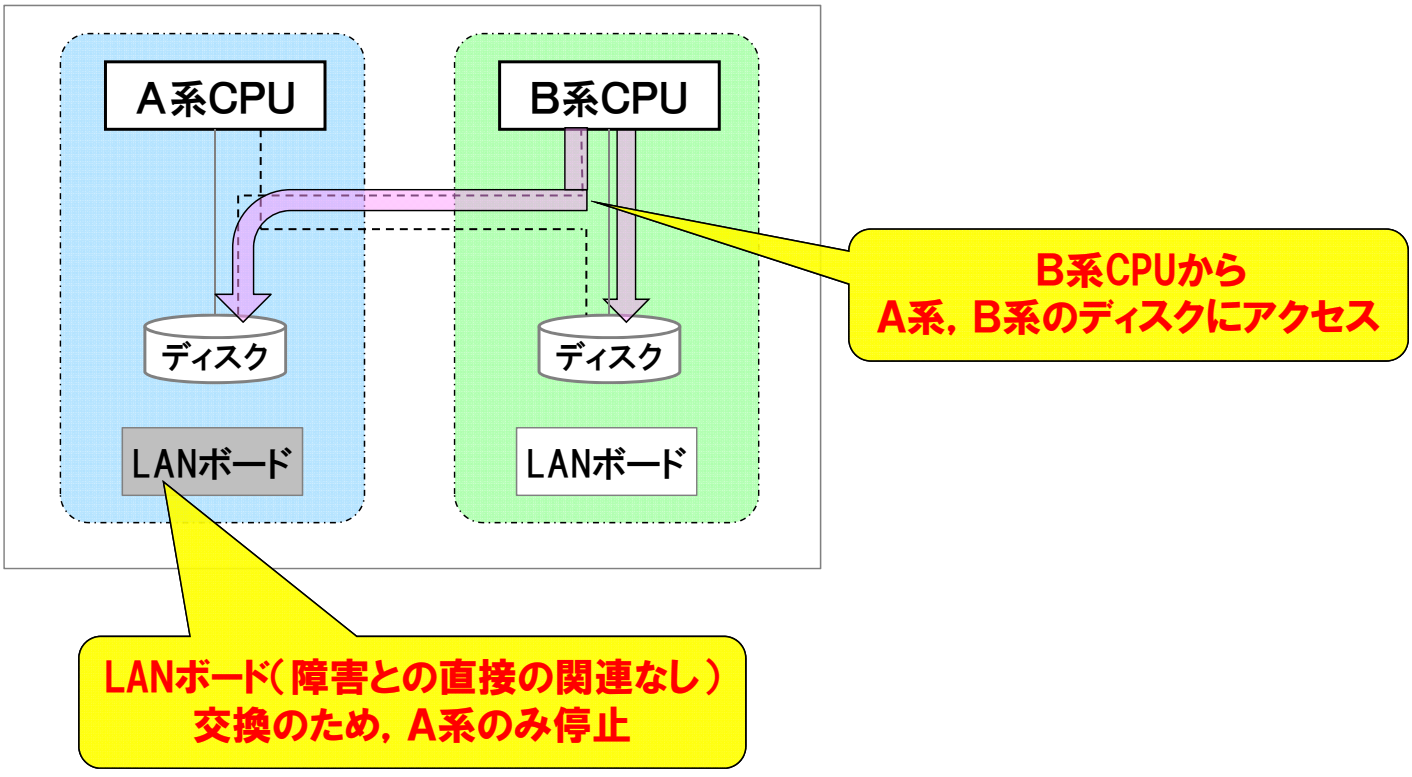
このような状況で，現場での実際の交換作業において，PRCの現用系停止状態で待機系を起動させた．この場合，装置の仕様上，初期起動時に，CPU上のOSがディスクにリセット要求を行い，その完了をタイマ監視（タイマ値=200ms）で待つ．HDDの場合には1msで完了するが，今回の装置に搭載されていたSSDでは300msが必要であった．

したがって，タイムアウトを検出したOSはディスクの異常と判断して停止要求を行い，SSDはそれを不正コマンドとして受け付けず，外部からのコマンドを拒否するモードに移行した．

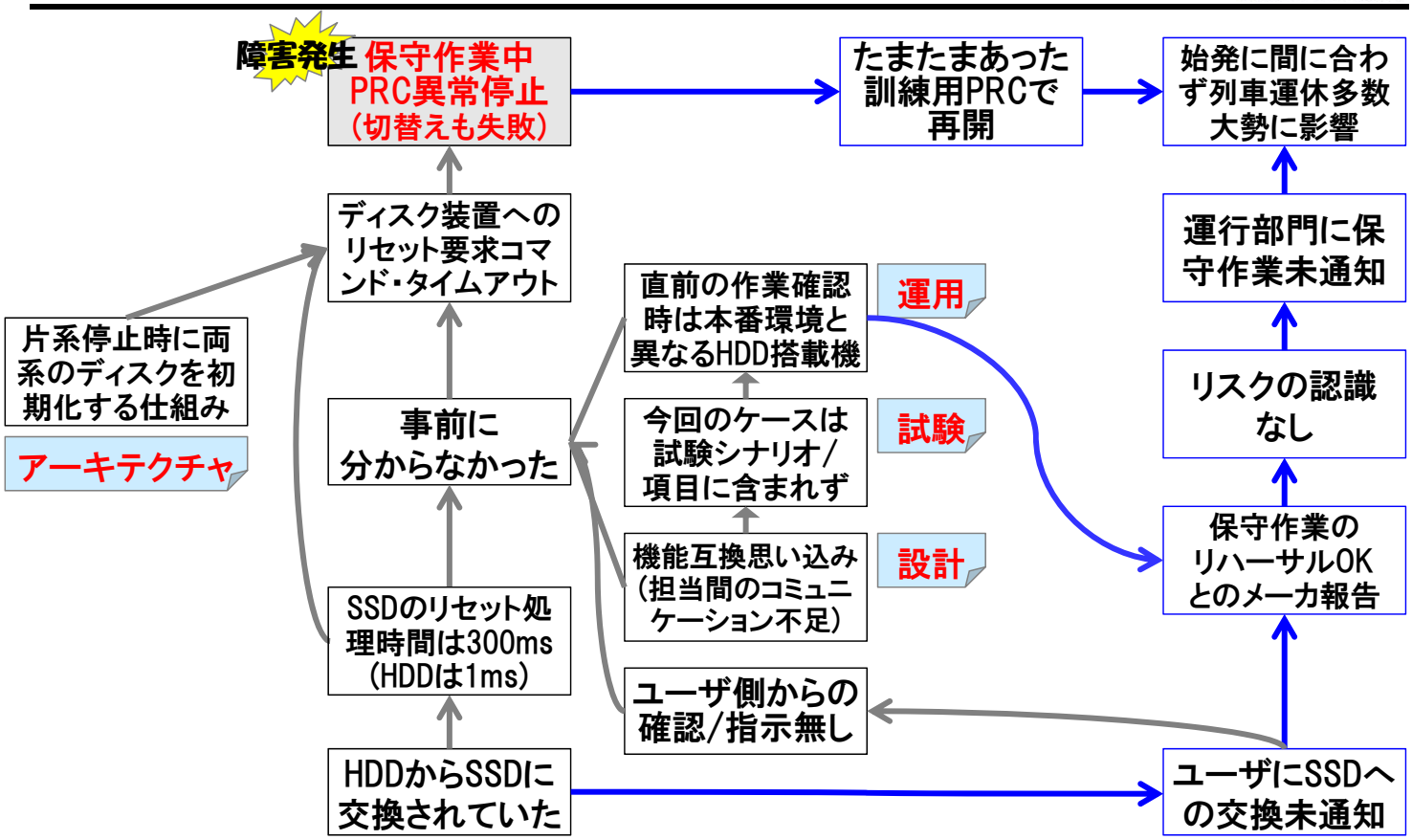
障害事例A(4/6)　自動進路制御装置(PRC)の構成概要



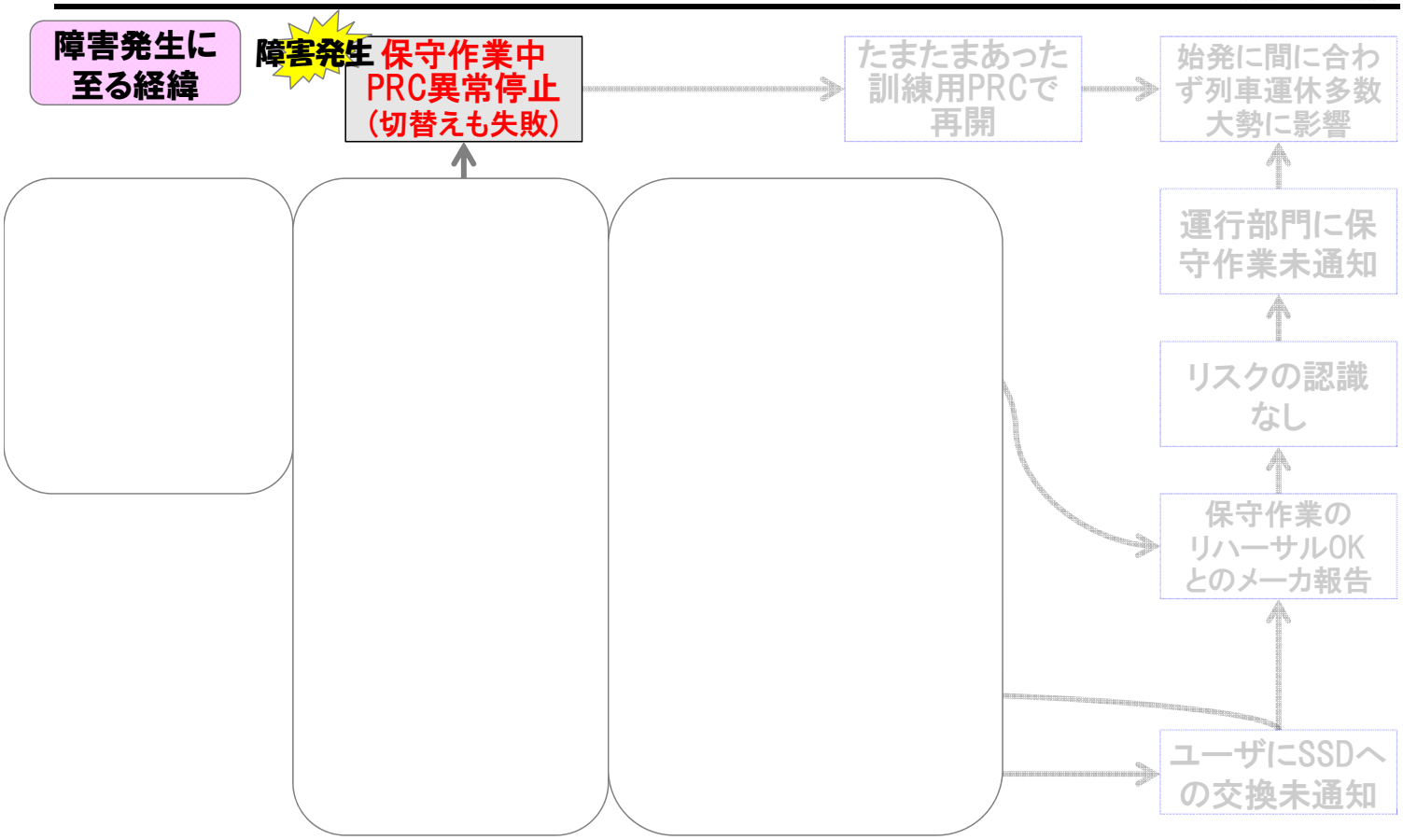
障害事例A(5/6)　自動進路制御装置(PRC)保守時の動作概要



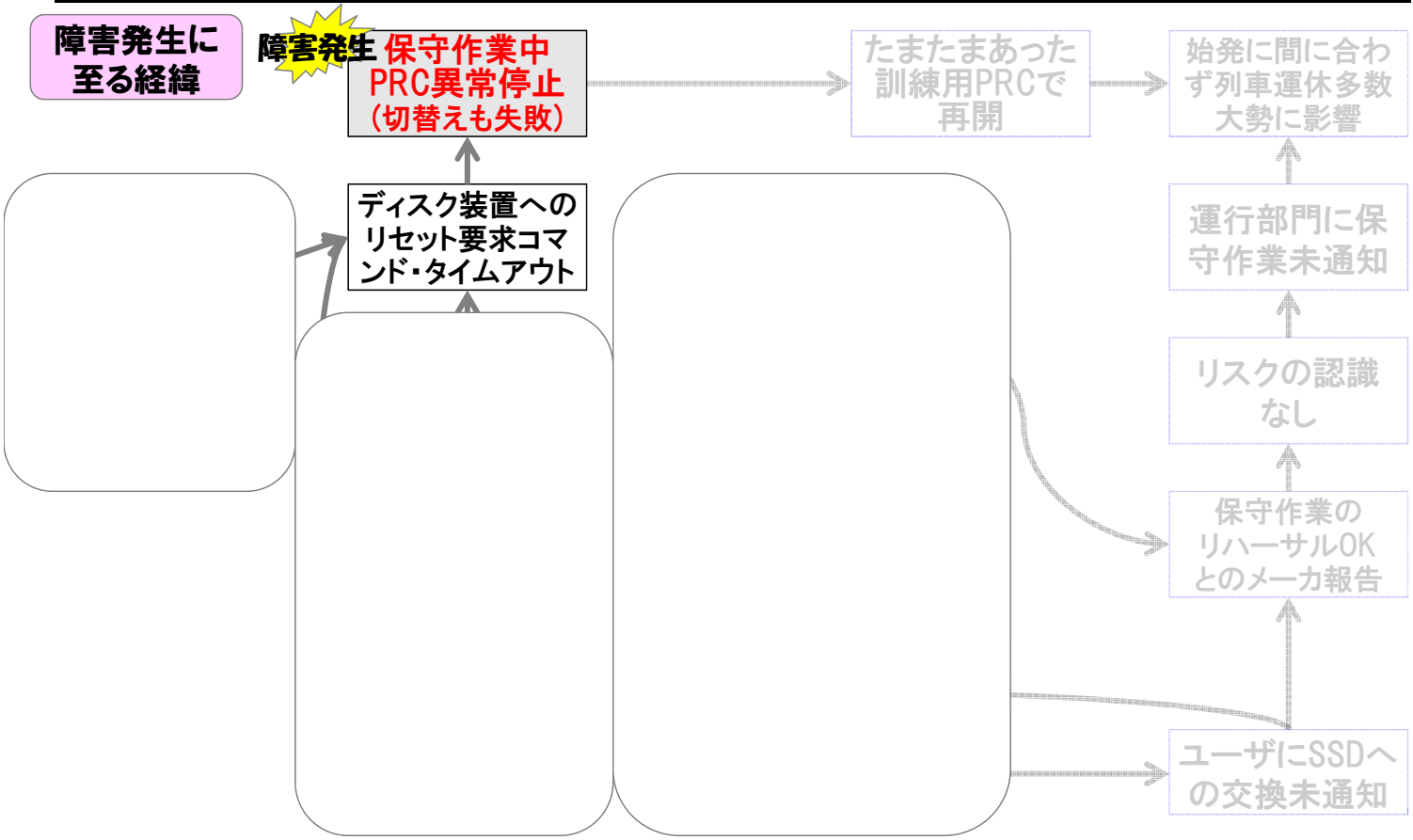
障害事例A(6/6)　障害事例の分析例



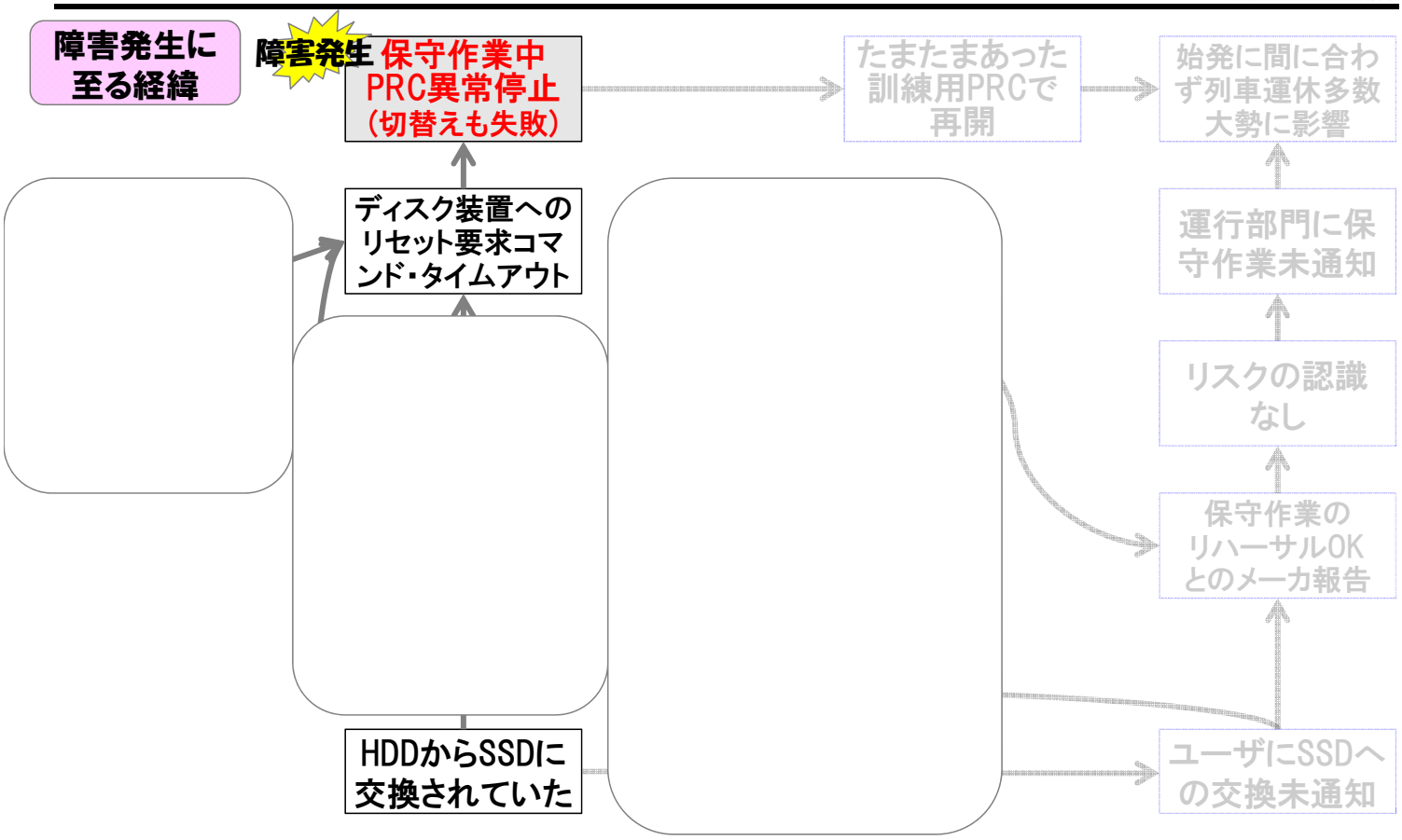
障害事例の分析例:障害発生に至る経緯(01/11)



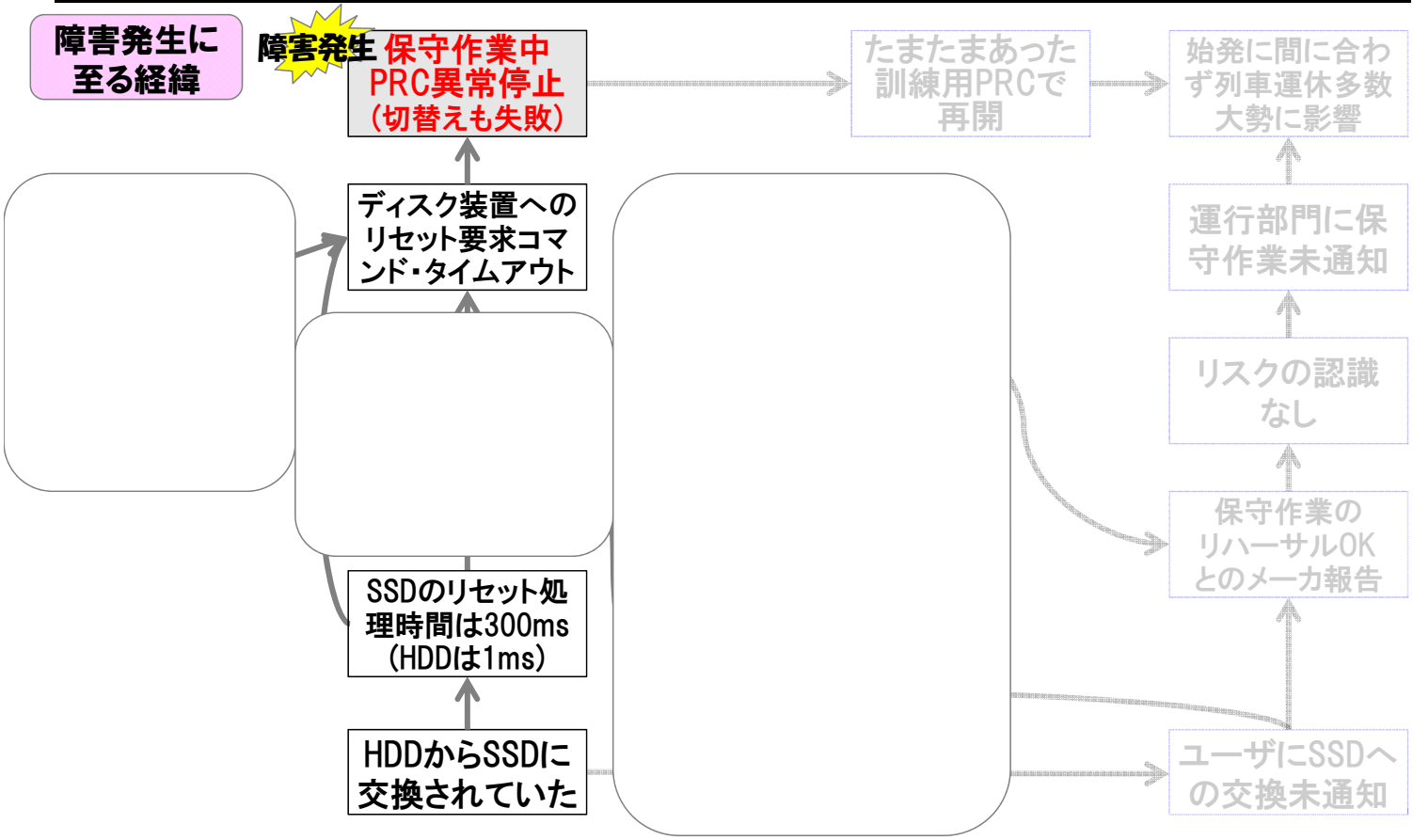
障害事例の分析例：障害発生に至る経緯(02/11)



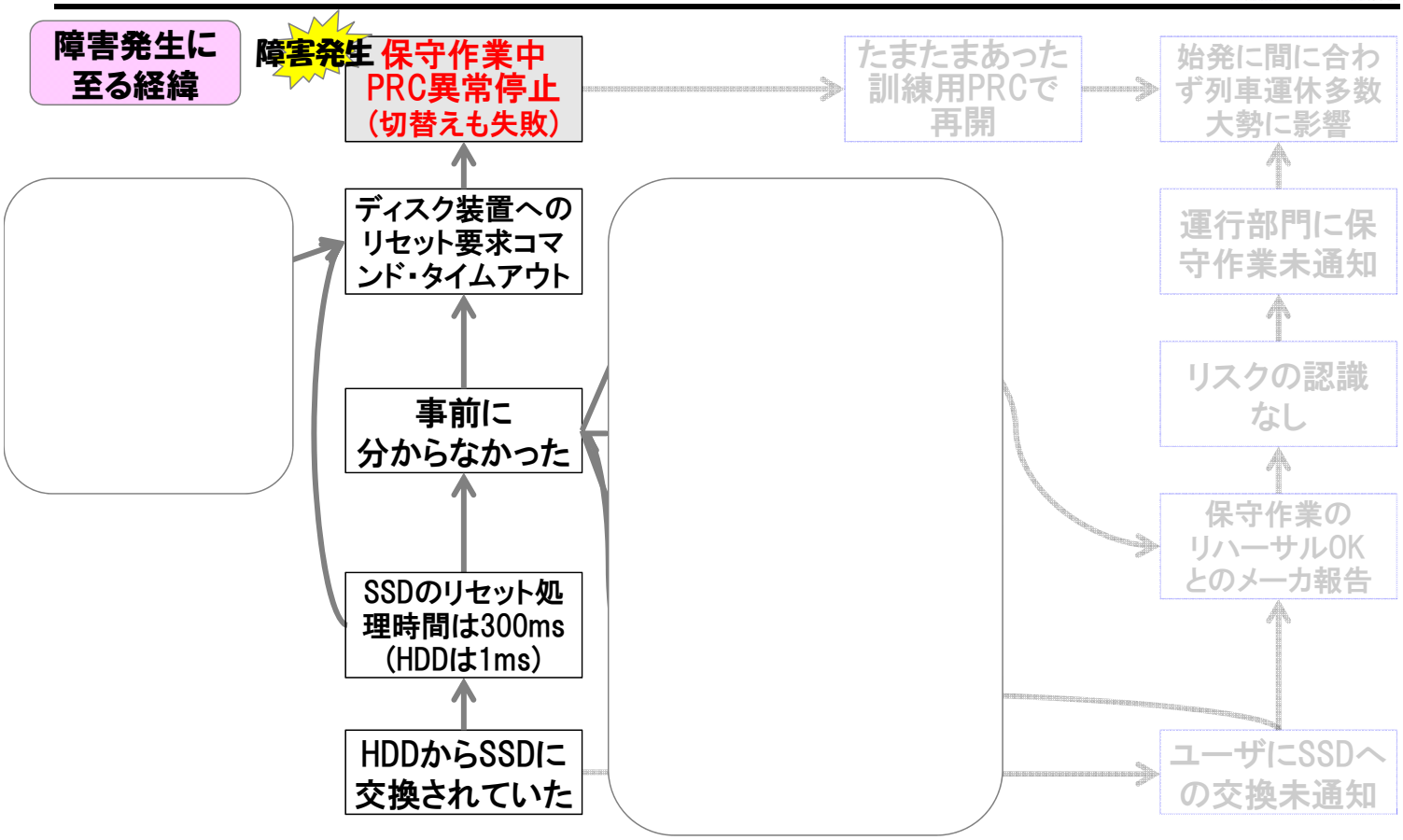
障害事例の分析例：障害発生に至る経緯(03/11)



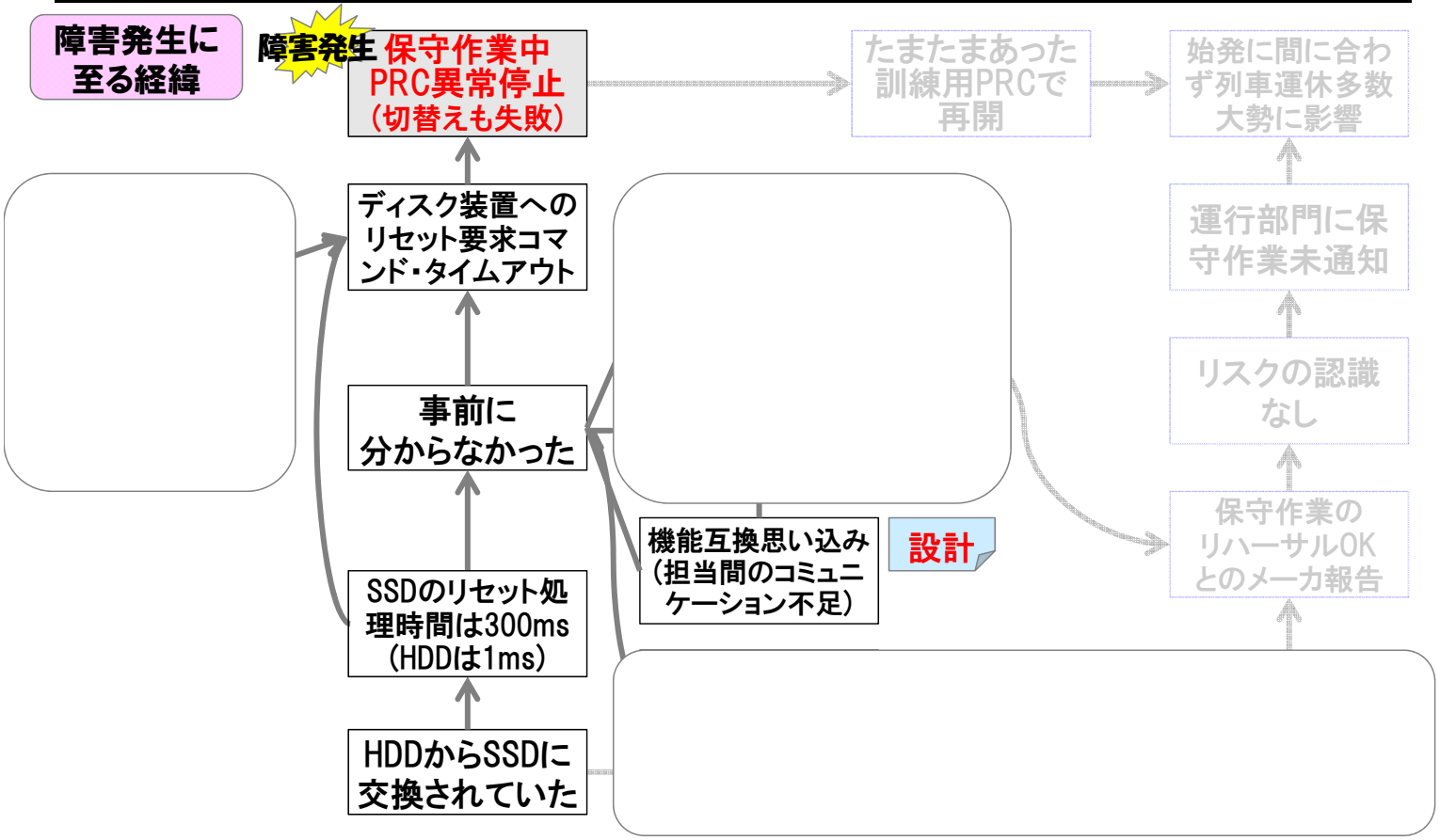
障害事例の分析例：障害発生に至る経緯(04/11)



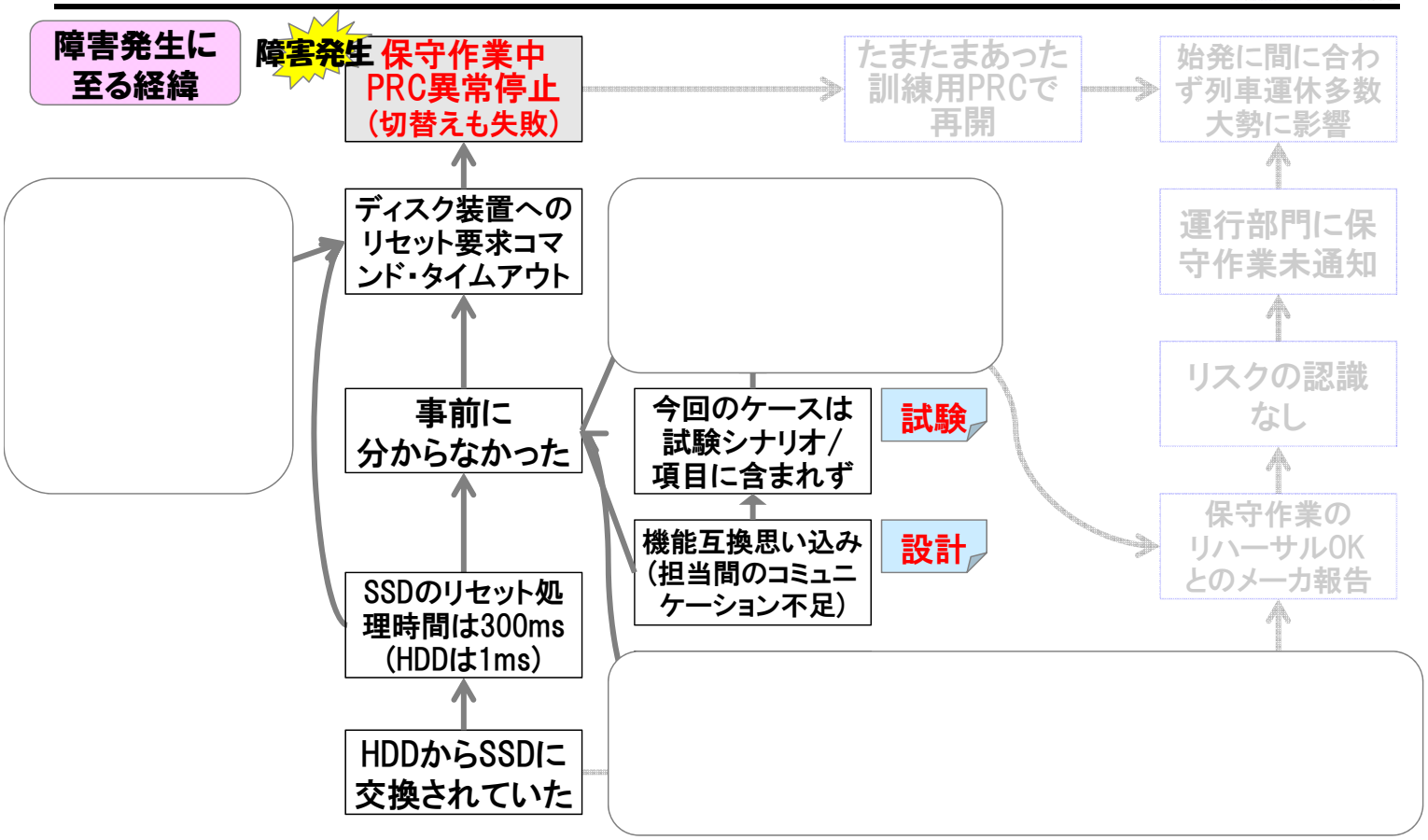
障害事例の分析例：障害発生に至る経緯(05/11)



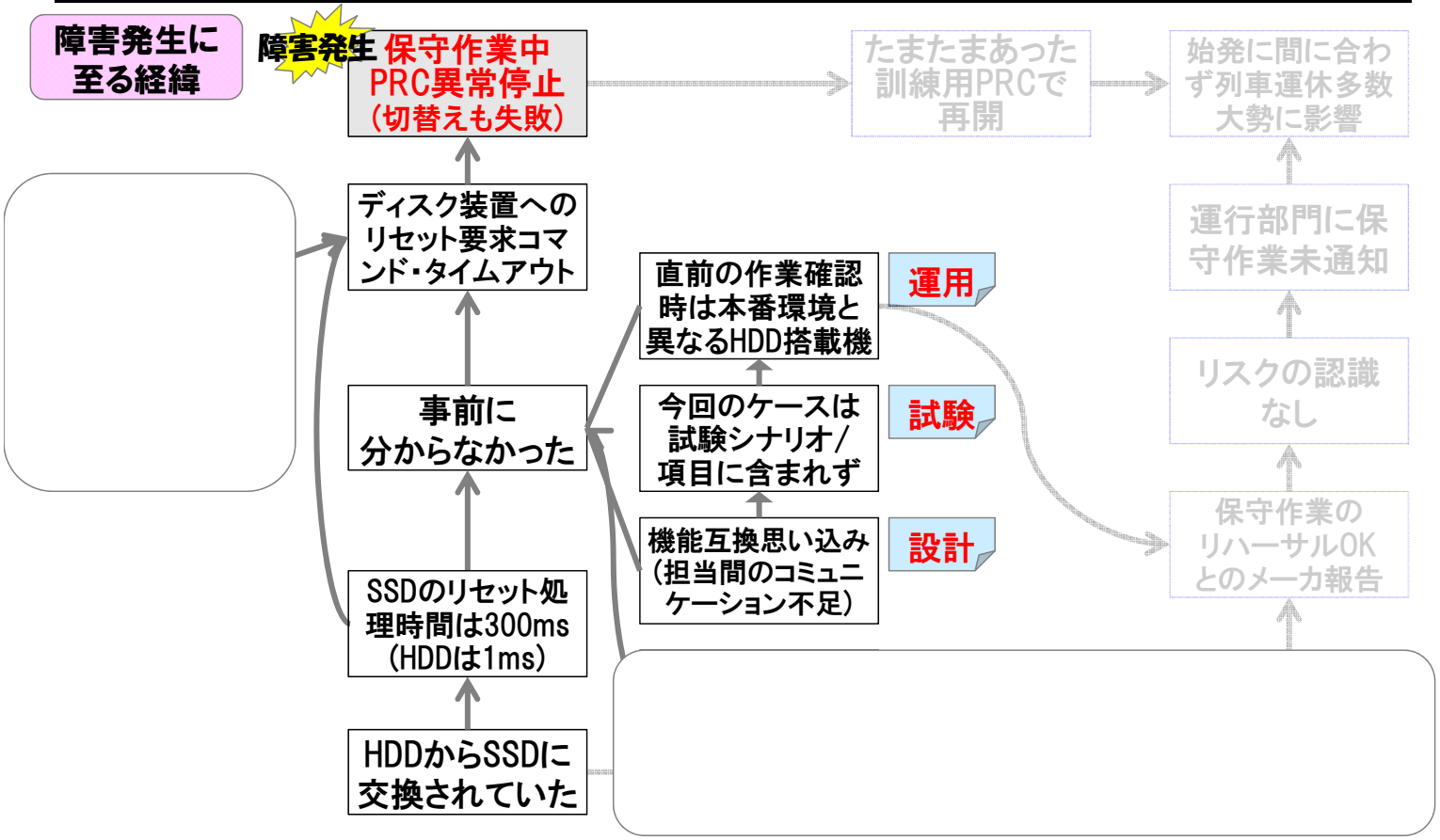
障害事例の分析例：障害発生に至る経緯(06/11)



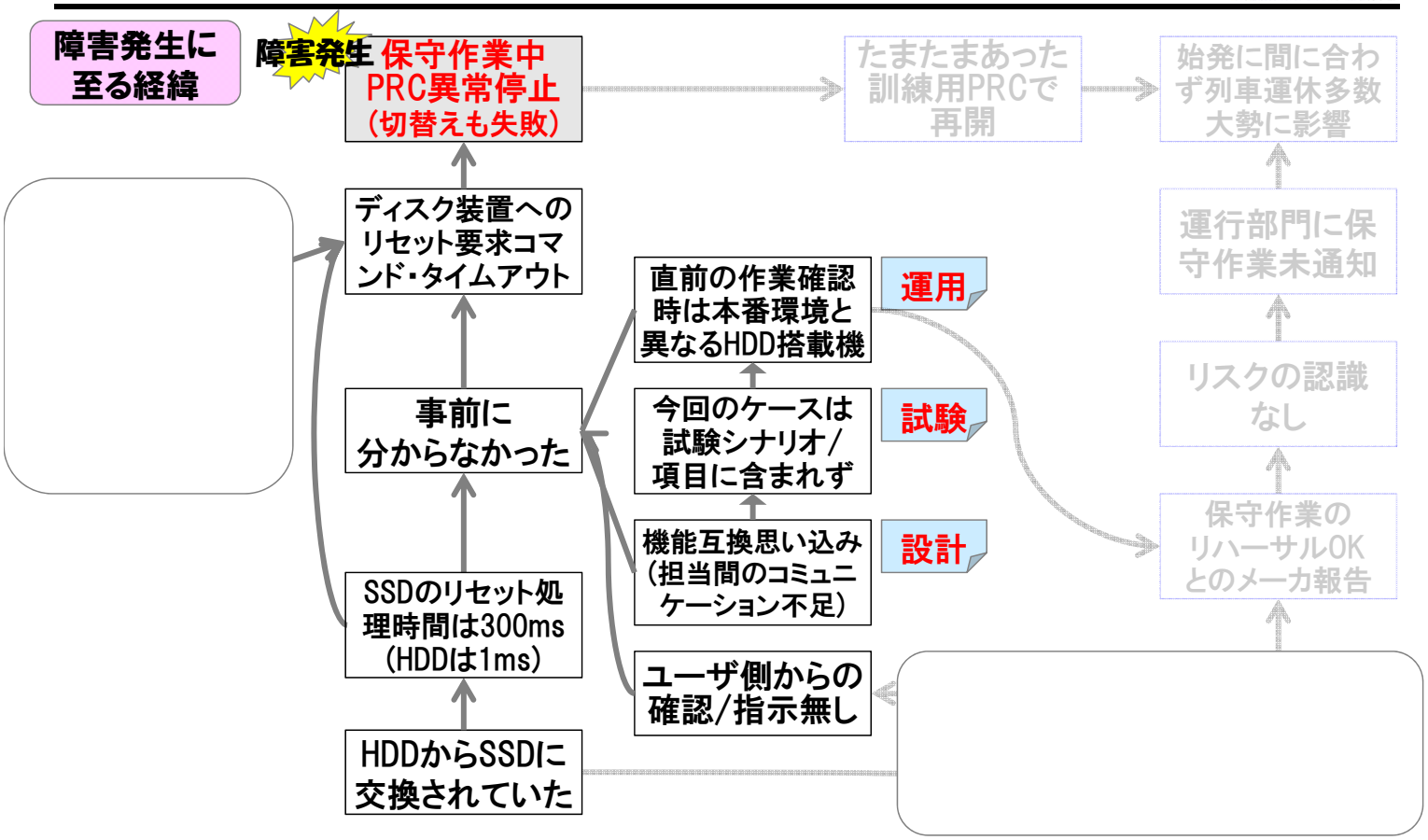
障害事例の分析例：障害発生に至る経緯(07/11)



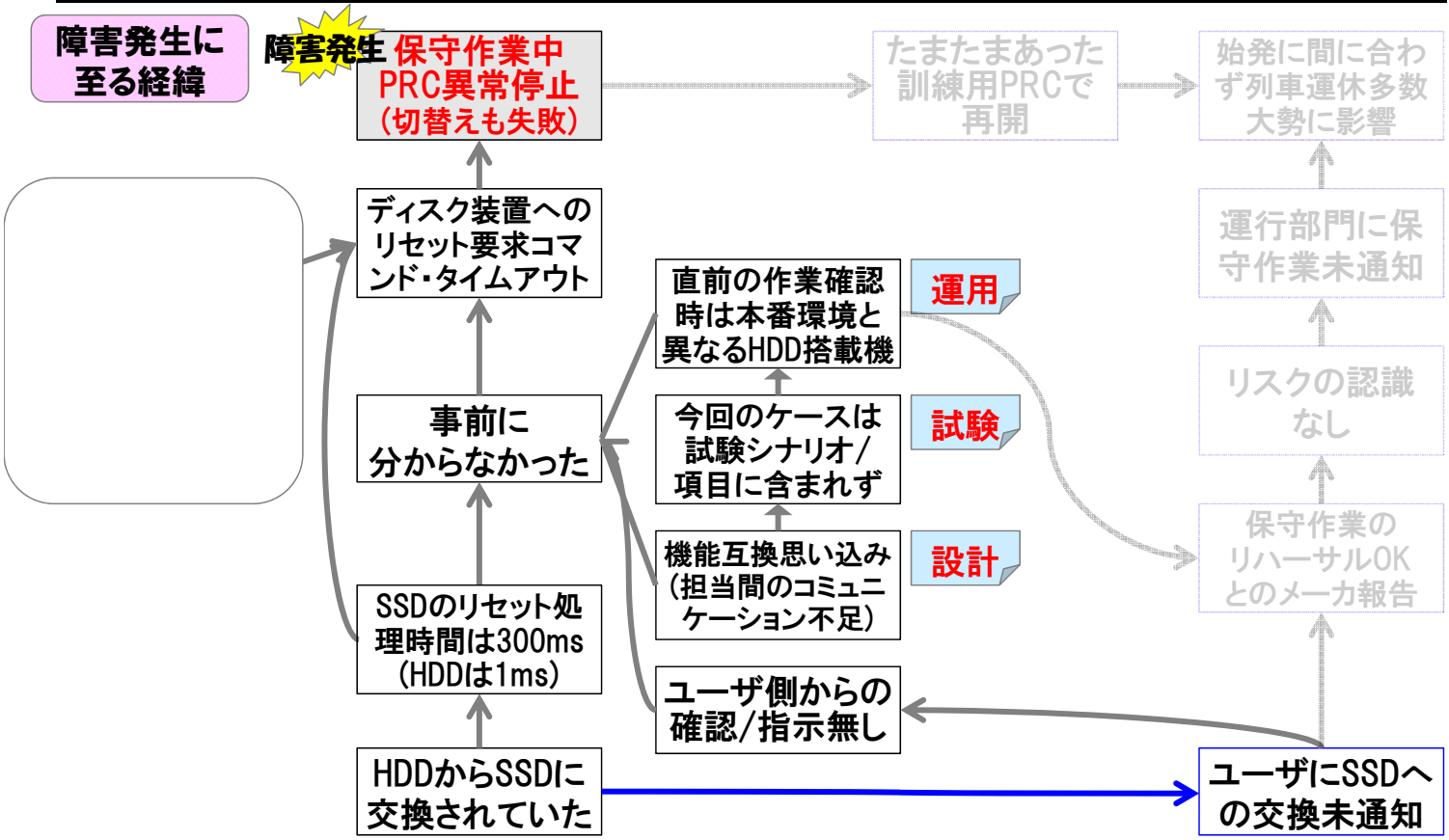
障害事例の分析例：障害発生に至る経緯(08/11)



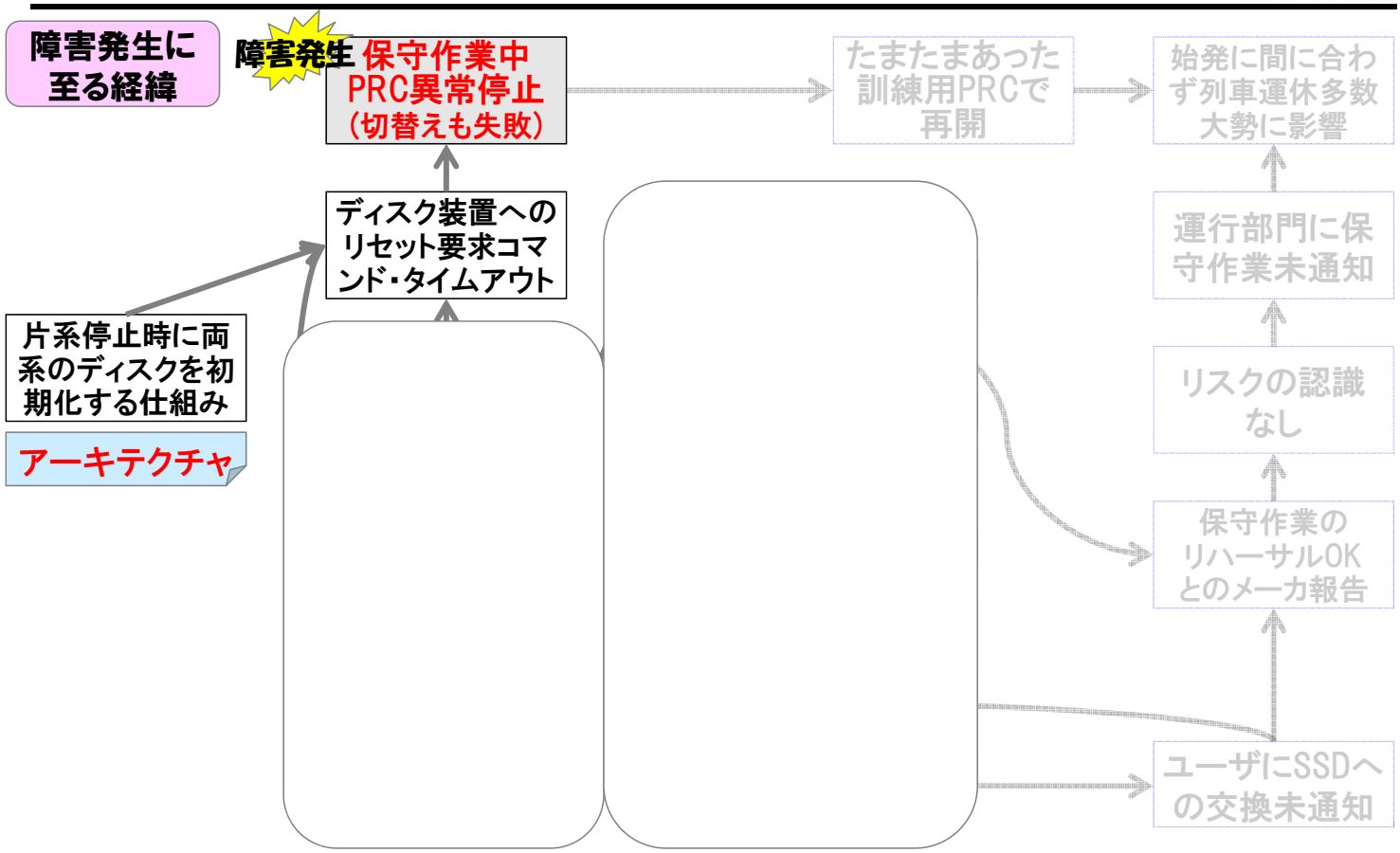
障害事例の分析例：障害発生に至る経緯(09/11)



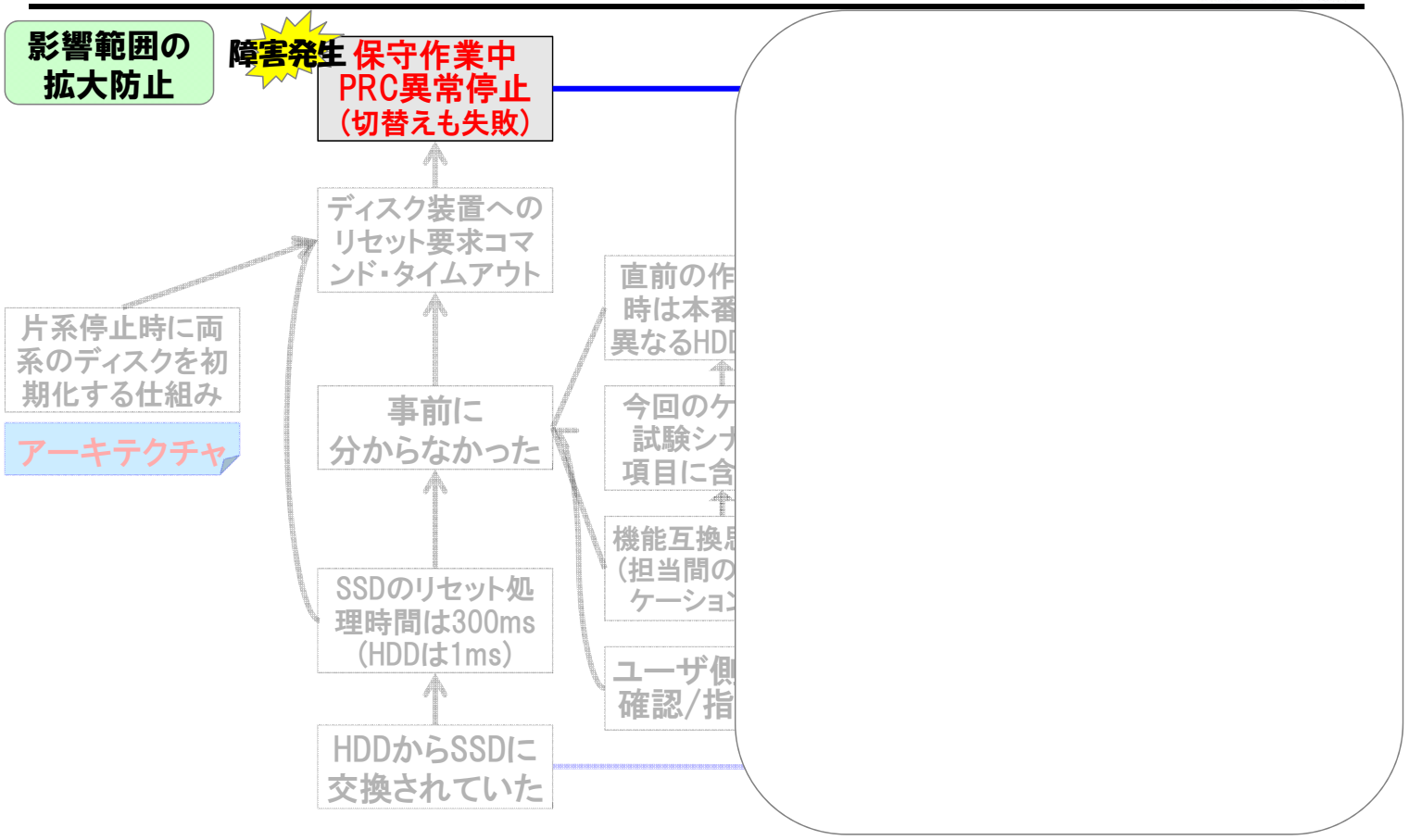
障害事例の分析例：障害発生に至る経緯(10/11)



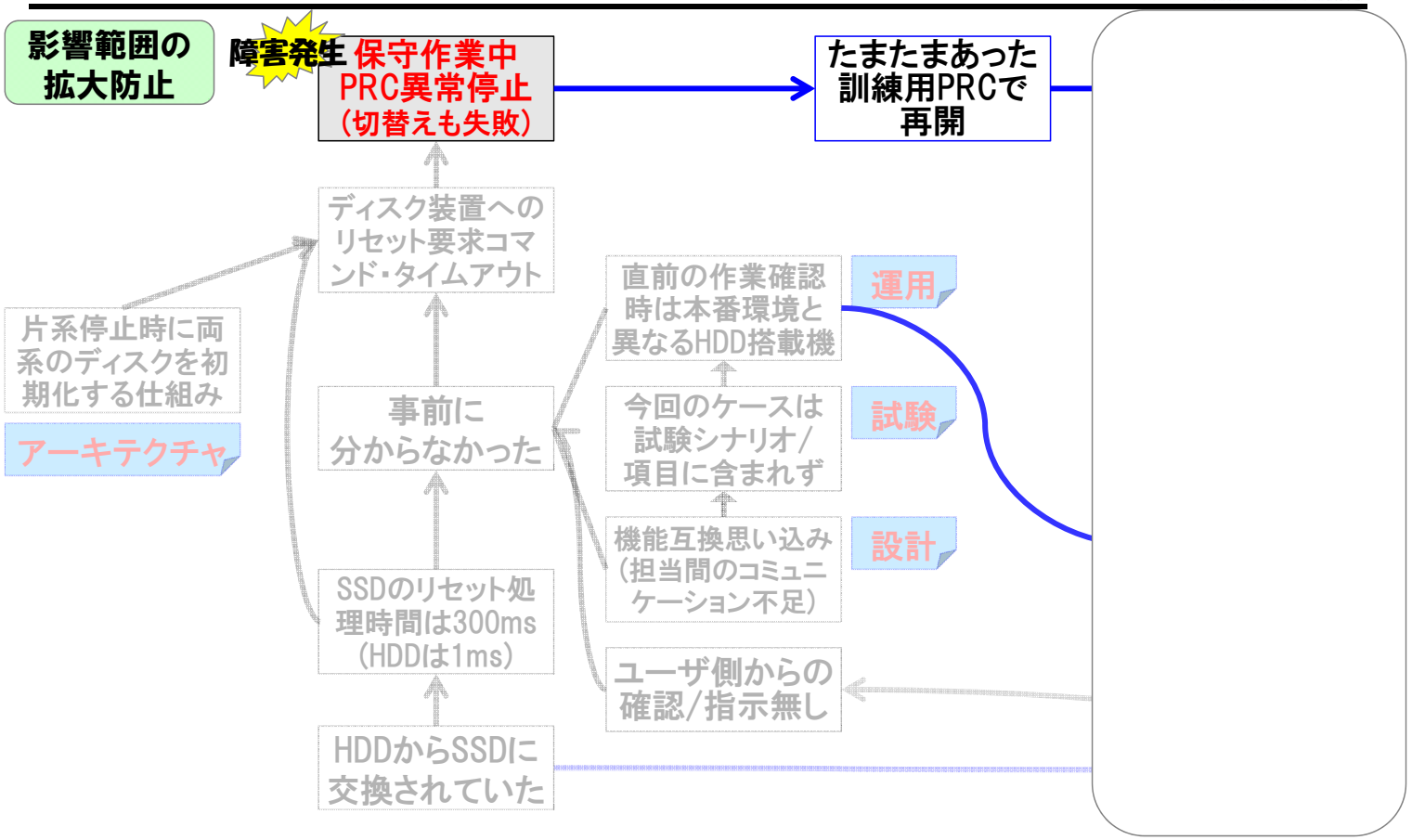
障害事例：障害発生に至る経緯の分析例(11/11)



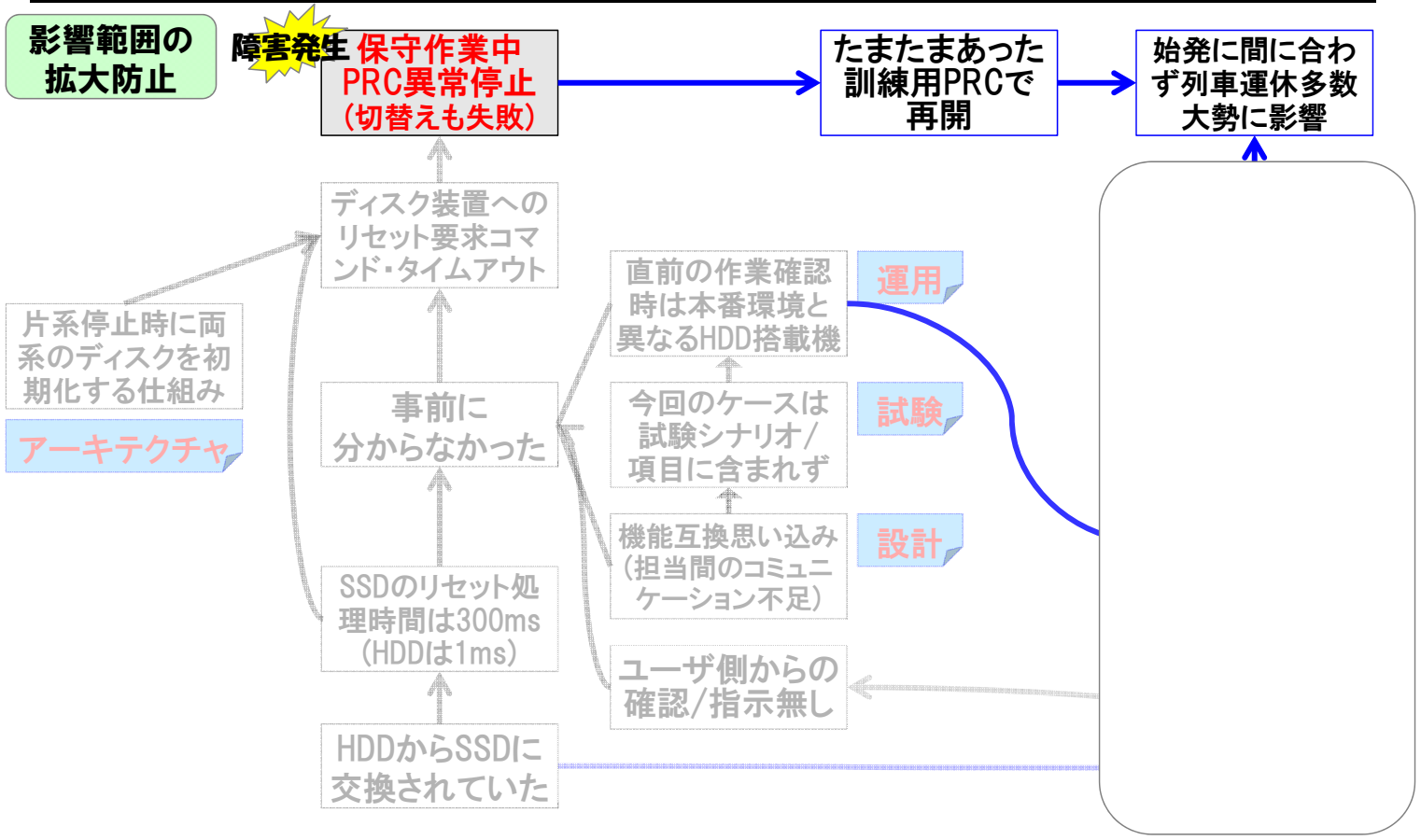
障害事例の分析例：影響範囲の拡大防止(01/07)



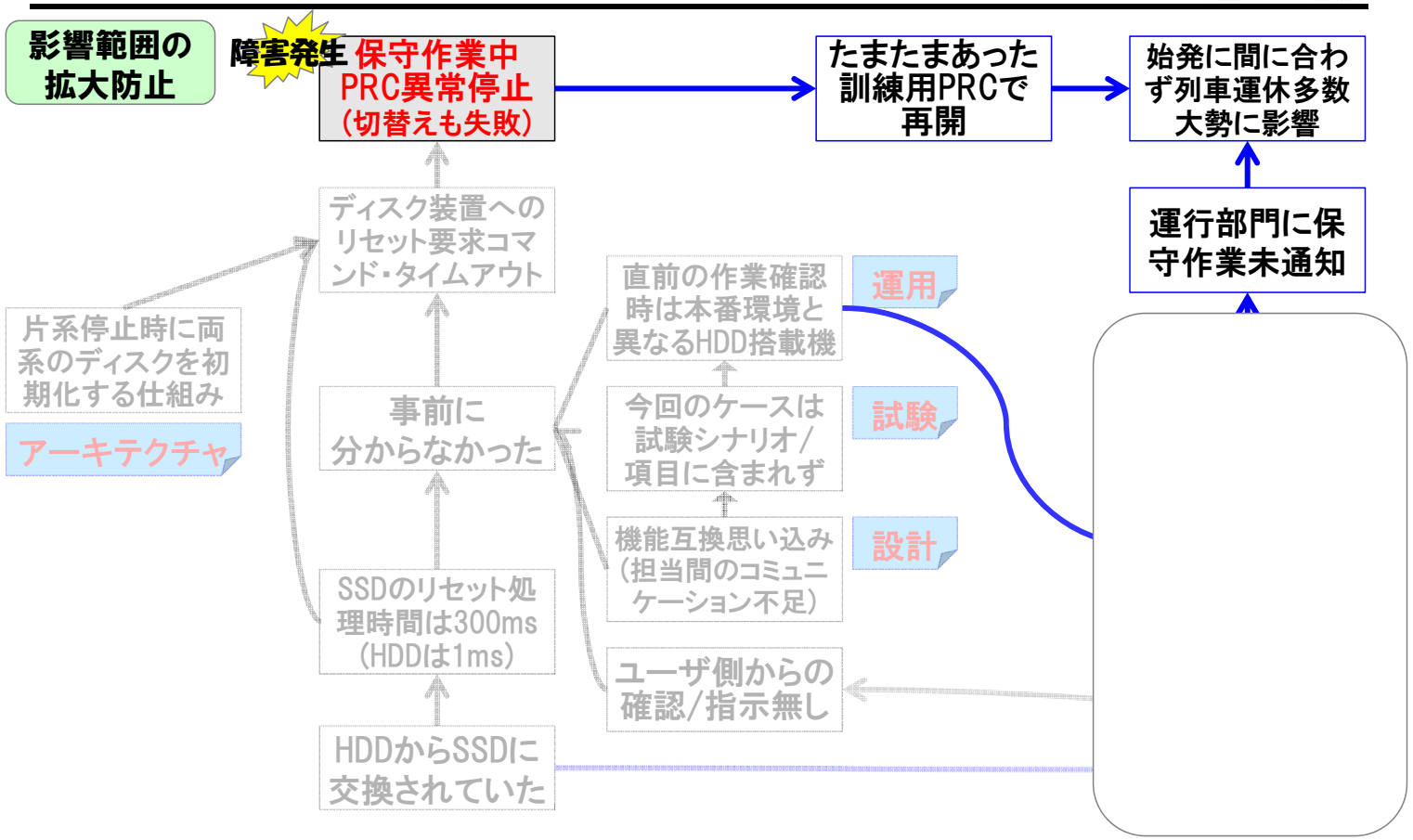
障害事例の分析例：影響範囲の拡大防止(02/07)



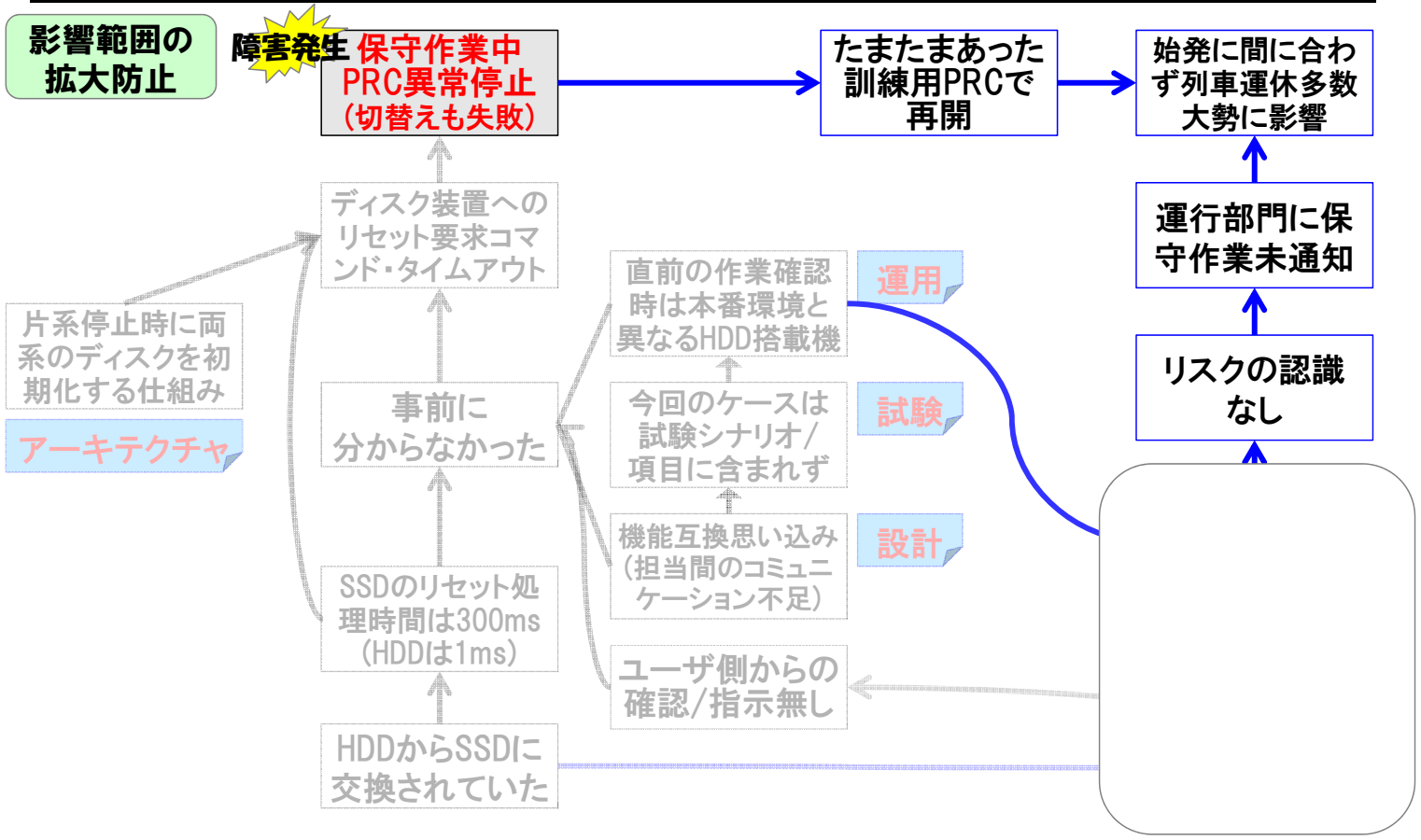
障害事例の分析例：影響範囲の拡大防止(03/07)



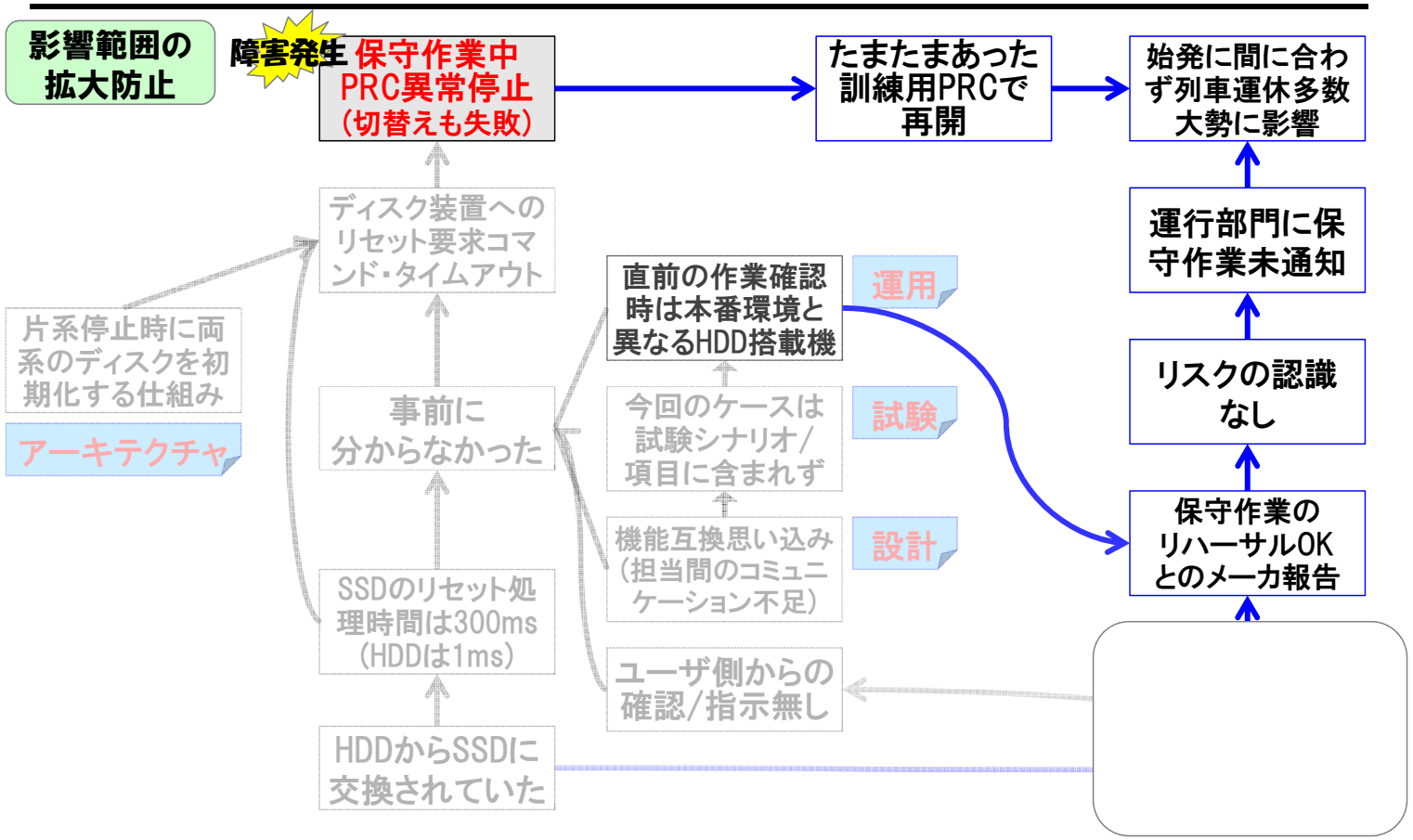
障害事例の分析例：影響範囲の拡大防止(04/07)



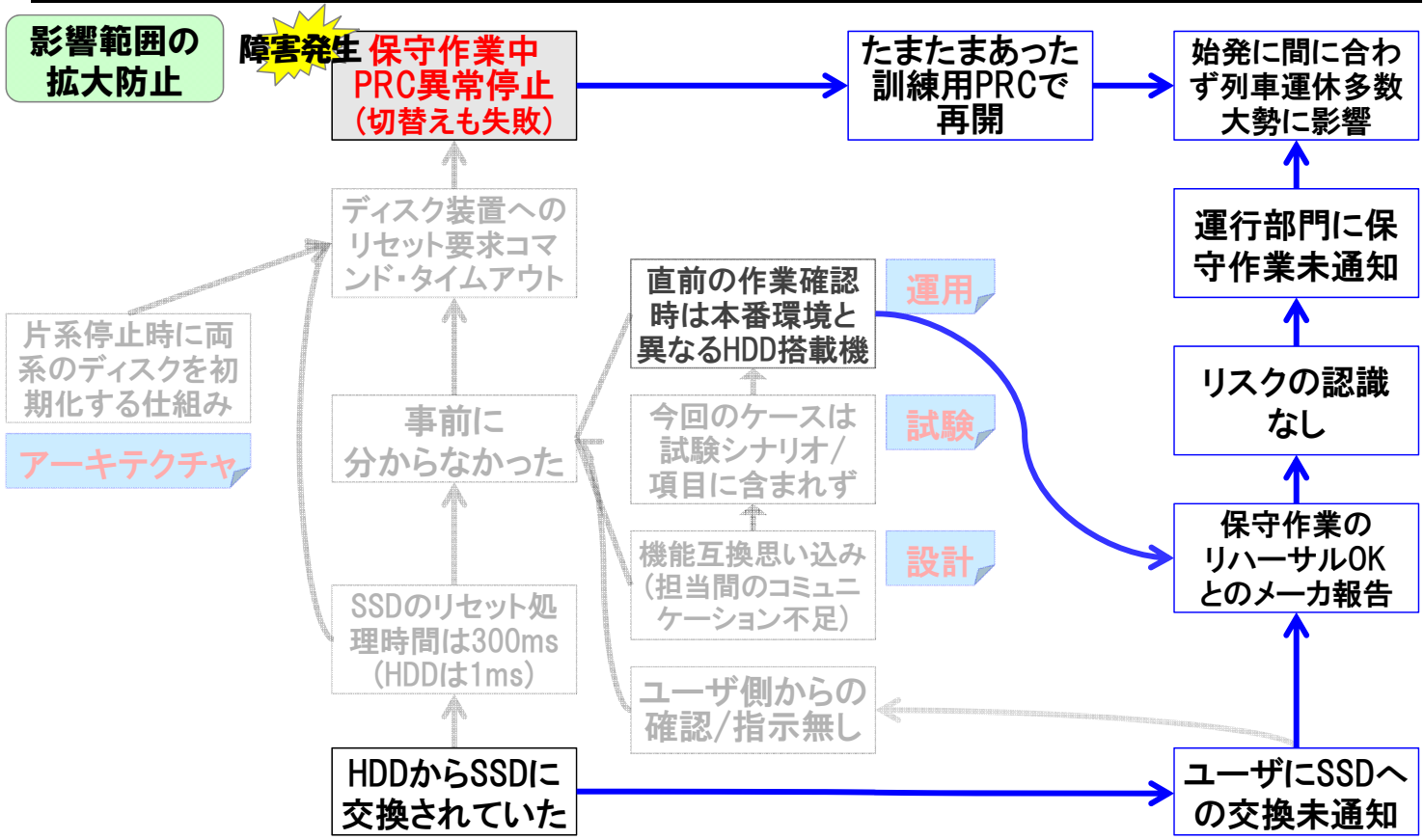
障害事例の分析例：影響範囲の拡大防止(05/07)



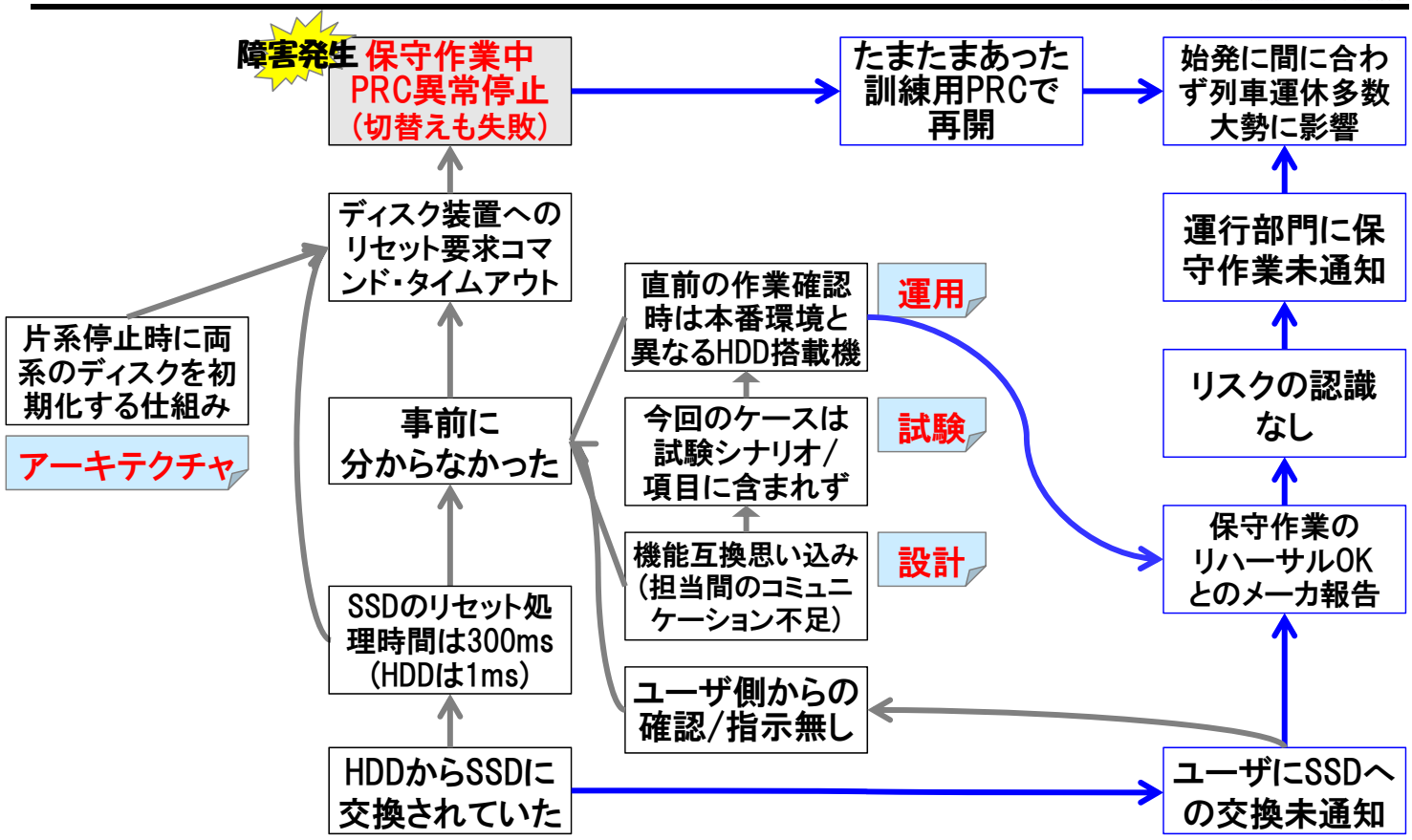
障害事例の分析例：影響範囲の拡大防止(06/07)



障害事例の分析例：影響範囲の拡大防止(07/07)



障害事例の分析例：詳細



詳細分析例

HDDからSSDへの交換

＜一般論＞

- 構築・完成から長時間経過すると，構成部品の保証期間が切れていることがある．
- その場合，既に生産中止となっており，しかも在庫がないことが多い．
- 自社製品であれば特別に製造できないこともないが，調達品の場合には難しい．
- 後継品は，技術の進展により，性能や品質の高いものに置き換わっている．
- 在庫があっても，コスト等を考慮して意識的に互換品を使うこともあり得る？

＜事実＞

- 「これまで付けていた装置の生産が終了していた．」
- （代替品として）取り付けた装置は，メーカから従来品と同等以上の機能があると説明を受けていた．」

＜質問＞

- 装置の交換や代替品の仕様について，メーカからユーザに，どの程度の情報が伝えられていたか？
- 同様に，メーカ内の担当者間で，どの程度の情報が共有されていたか？

ステークホルダ間のコミュニケーションを適切に！
特に，保守時において．

詳細分析例

SSDのリセット処理時間はHDDの場合よりかなり長い

＜一般論＞

- 代替品は『互換性』があるという説明になっていても，機能やインタフェースが一部異なっているかもしれない．特に，新機能が追加されていることがよくある．
- 代替品に異なる技術が用いられている場合には，設計思想そのものが異なっていることもある．また，機能は同じでも，性能等の非機能特性が異なるケースは多い．

＜事実＞

- ディスクのアクセス・インタフェースは，標準規格に準拠していた．しかし，リセット要求コマンドの実行時間は規格の規定外であり，実装に任されていた．

＜質問＞

- メーカの担当者には，『互換性』に，どの程度の認識があったのか？

「互換性」という言葉の落とし穴
非機能の違いを見逃すな！

詳細分析例

設計段階での見逃し

＜一般論＞

- システムの構成要素の一部交換という保守においては，交換前後の構成要素間の仕様の差異を確認し，仕様が異なる部分の既存システムへの影響を分析する．
- 影響分析には，対象構成要素とインタフェースする部分の担当者が関わる．
- 影響分析のためには，担当者が異動等のために既にいない場合もあるため，関連ドキュメントが揃っていなければならない．
- 特に，性能差がある場合，タイマ監視等を行う処理においては，監視タイマ値の妥当性を入念にチェックし，チューニングを実施する必要がある．特に差異のない構成要素の特性がシステムの性能に影響する場合を見逃してはならない．

「互換性」という言葉の落とし穴

差異確認と影響分析は，全関係者が参加し入念に！

＜事実＞

□（特になし）

＜質問＞

➢ 仕様の差異確認は，全コマンドについて入念に行われたか？
➢ 仕様が異なる部分の影響分析には，関連担当者（OSの担当者等）が参加したか？
➢ 保守作業の手順は，社内で整備され，マニュアル化されていたか？

詳細分析例

試験段階での見逃し

＜一般論＞

- 総合試験では，想定される運用シナリオ（試験項目）を洗い出し，テストする．
- 保守時のリグレッション（回帰）テストでは，通常，初期構築時の試験項目を一通りテストする．（確実に影響ないと判断できれば，実施しない項目もあり得る．）
- 試験環境では本番環境を忠実に再現できない場合には，類似シナリオでの試験と机上確認を行うと共に，試験未実施のリスクを評価する．

「テスト設計」を重視せよ！

戦略的なテストはコストダウンへも近道！

＜事実＞

□ HDDからSSDへのディスク交換時には，今回事象が発生した保守作業のシナリオについての試験は実施されなかった．

＜質問＞

➢ 今回のシナリオが初期構築時の試験項目に含まれていた場合，回帰テスト時にそれを実施しなかった理由は？
➢ 今回のシナリオが初期構築時の試験項目に含まれていなかった場合，試験項目の抽出の基準，考え方は？
➢ 試験項目抽出や試験実施に関する社内標準が整備されていたか？

詳細分析例

運用段階(直前の作業手順確認)での見逃し

＜一般論＞

- リハーサル作業は，本番環境と極力同じ条件(環境，時間帯等)で行う．
- リハーサル環境が本番環境を忠実に再現できない場合には，類似環境での確認と机上確認を入念に行うと共に，トラブル発生時のリスクを評価する．その結果，必要に応じ，コンティンジェンシー計画を策定する．

＜事実＞

- メーカ工場での直前のリハーサル作業では，SSD搭載装置がなかったため，HDD搭載装置を用いて行った．

＜質問＞

- メーカの担当者及び管理者(責任者)は，リハーサル環境が本番環境と異なることによるリスクをどの程度認識していたか？
- リハーサル作業の結果を「確認OK」と判断する基準は，社内で明確に規定され，関係者が認識していたか？

テスト環境と本番環境の差異を体系的に整理し，障害のリスク対策を練る

詳細分析例

事前に分からない組織的な問題

＜一般論＞

- 単一エラーの発生確率に比べ，一般に，多重エラーの確率は極めて低い．
- 同じ過ちを犯す組織には，成熟度あるいは企業風土に問題がある．

＜事実＞

- メーカでの設計，試験，運用(直前のリハーサル作業での確認)の全ての段階における誤り発見の機会をすり抜けてしまった．

＜質問＞

- メーカにおける，当該装置に関する(開発・)保守・運用の体制は？
- メーカにおける，システムの(開発・)保守・運用に関する社内標準は，どの程度整備されているか？

自律的なプロセス改善のマインドを醸成し，組織の成熟度を高めよう！

詳細分析例

ユーザ側からの確認／指示がない

＜一般論＞

- ユーザ側も，適切なポイントで，メーカ／ベンダにシステム構築・運用状況の確認を行うのがよい．
- 重要なレビューには，ユーザ側も参画する．
- ユーザ側は，過去や他所での具体的トラブル事例があれば，それをメーカ／ベンダに示して注意を喚起する．
- ユーザによるメーカ管理には相応のコストを要するため，システムの重要度に応じてその内容や程度を調節する．

＜事実＞

□ メーカからユーザに対し，ディスク交換の事実が知らされていなかったため，それに関するユーザからメーカへの特段の確認／指示はなされなかった．

＜質問＞

➢ ユーザによる普段のメーカ管理の内容は？　特になし(お任せ)？

ユーザは，システムの重要度に応じ，
適度のメーカ管理を！

詳細分析例

アーキテクチャ／方式上の問題？

＜一般論＞

- 高信頼なシステム設計では，構成要素間，システム間のインタフェース／インタラクションを極力少なくする．
- 必要のない処理は行わない．
- 容易には復旧不可能な，システム動作不可状態に陥る必要性を精査する．

＜事実＞

□ 片系の切離しが発動された場合，稼働系CPUのOSから，稼働系・停止系双方のディスク(インタフェース)に対しリセット要求コマンドが発せられる仕様となっていた．

＜質問＞

➢ 片系の切離しが発動された場合，稼働系CPUのOSからディスク(インタフェース)にリセット要求コマンドを発する理由は？

➢ リカバリによりディスクアクセスを復旧する方法を設けられなかったか？

アーキテクチャは
Simplicity is the Best!

詳細分析例

冗長化構成による可用性向上

＜一般論＞

- システムの重要度に応じた冗長構成を採る．重要システムでは予備を待機させる．
- 必要のない処理は行わない．

＜事実＞

- PRCはフォールトトレラントコンピュータを使用していたため，予備の装置を置いていなかった．
- たまたま訓練用の装置があり，それに切り替えて作業した．

＜質問＞

- PRC停止のリスクをどのように評価していたか？

障害発生のリスクを評価し，
システムの重要度に応じた冗長構成を！

詳細分析例

リスクを想定した備え

＜一般論＞

- 保守作業に対しては，リスク評価を行った上で，必要に応じ，万一のトラブルに備えた対応方法を事前に決めておく．
- 万一のトラブルに備えて，関係部門が協力する万全の態勢を整えておく．

＜事実＞

- 保守作業の実施について，運行部門に事前には知らされていなかった．
- 保守作業は，深夜でも貨物列車が運行しているので，貨物列車運行の少ない早朝の時間帯で，始発列車に間に合うよう時間をみて設定した．
- 両系全停止の作業の場合，装置の立上げに多くの時間を要することから，片系停止での作業方法を選択した．（本方法は今回初めて実施）

＜質問＞

- 保守作業トラブルによる影響をどのように評価していたか？

障害発生のリスクに備え，
関係者が協力した万全の態勢を！

再発防止策(メーカ)

SEC Software Reliability Enhancement Center

プログラム改修

- 監視タイマ値の変更
  - OSの監視タイマ値を200msから360msに変更
- リセット時のエラー検出時にも，ディスクの読み書きを不能としないように処理を変更

設計

- 部品更新時に確認不足を起こさないよう，社内ルールの見直し

運用(直前の作業手順確認)

- 工場でのリハーサル時には，現地(本番)と同一構成の装置を使用

再発防止策(ユーザ)

SEC Software Reliability Enhancement Center

保守作業に関する準備・確認の体制

- 作業による万一のシステム停止の影響範囲について，関係部門と共有すると共に，これを想定した作業計画の策定
- 機能停止を伴う部品交換作業は，開発訓練用装置を用いてオフラインで確認した後，実際の交換作業を実施

社長のことば：影響を受けた人の中には，試験に間に合わずに受験できず，その後の人生が変わってしまった人がいるかもしれない．そういうことがあってはならない．

抽象化と本質

PRC内の ディスク装置の 交換	抽象化	情報処理システム内の 構成要素の 交換
リセット要求コマンドの 処理時間が 異なる	抽象化	非機能関連の インタフェースが 異なる
事前の 列車運行部門との共有	抽象化	事前の 関係部門との共有

本質

- システムの一部構成要素の交換
- 非交換部分とのインタフェースの整合
- リスク評価に基づく事前対応

教訓分析例

教訓の一例 (1/5)

【教訓タイトル】

★:本質

＜抽象的な表現の例＞

システムの部分変更(に伴う新旧混在)時に(非変更部分との)整合性を確認する.

＜具体的な表現の例＞

変化に対応して(プログラム／システム定義データ中の)定数をチェックする.

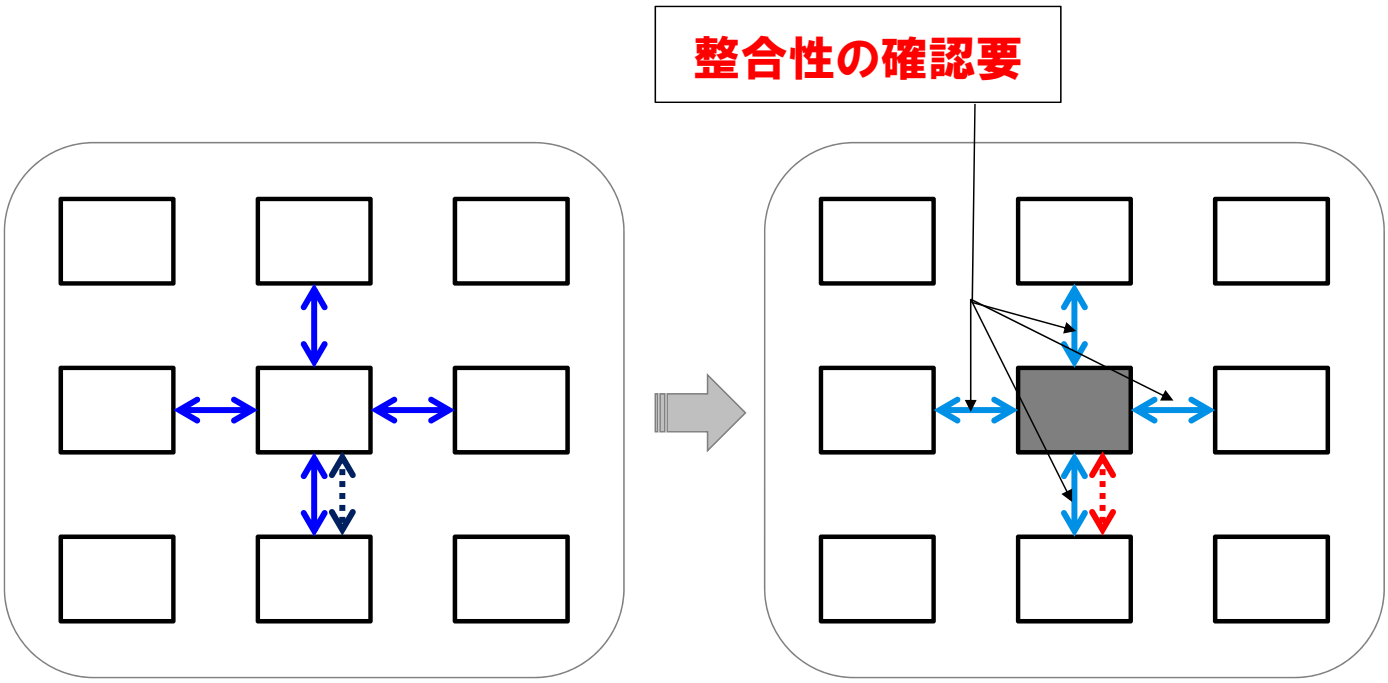
【説明】

共通コンテキスト

交換（部分変更）する構成要素（ソフトウェアを含む）は，交換前のものと互換性があるという仕様になっていても，機能やインタフェースが一部異なっているかもしれない．特に，新機能が追加されていることがよくある．異なる技術が用いられている場合には，設計思想そのものが異なっていることもある．また，機能は同じでも，性能等の非機能特性が異なるケースは多い．
したがって，交換する構成要素と，システムの他の部分（交換しない部分）とが整合するかについて，様々な視点から確認する必要がある．

特に，性能差がある場合，タイマ監視等を行う処理においては，監視タイマ値の妥当性を入念にチェックし，チューニングをやり直す必要がある．特に，新しい構成要素の特性が性能に影響する場合を見逃してはならない．

教訓の一例 (2/5)



教訓の一例 (3/5)

【対策（例）】

共通コンテキスト

◆コンポーネント・レベル
・ミスを起こさない：特にメンテナンス時における，ハード担当と制御ソフト担当とのコミュニケーションの徹底（気づかせるための会議，文書の工夫等）
・ミスを逃さない　：試験時の思い込み（“互換性”への過信）排除（第三者の関与等）．標準規格の規定事項/規定外事項の明確化

◆システム・レベル
・ミスを起こさない：変化点を捉えた，俯瞰的かつ系統的な設計レビュー
・ミスを逃さない　：試験時における，本番環境との相違点に関するリスク評価

◆環境レベル
・影響を拡げない　：代替システムの準備，リスク評価と関係者間の情報共有，トラブル発生への備え

教訓の一例(4/5)

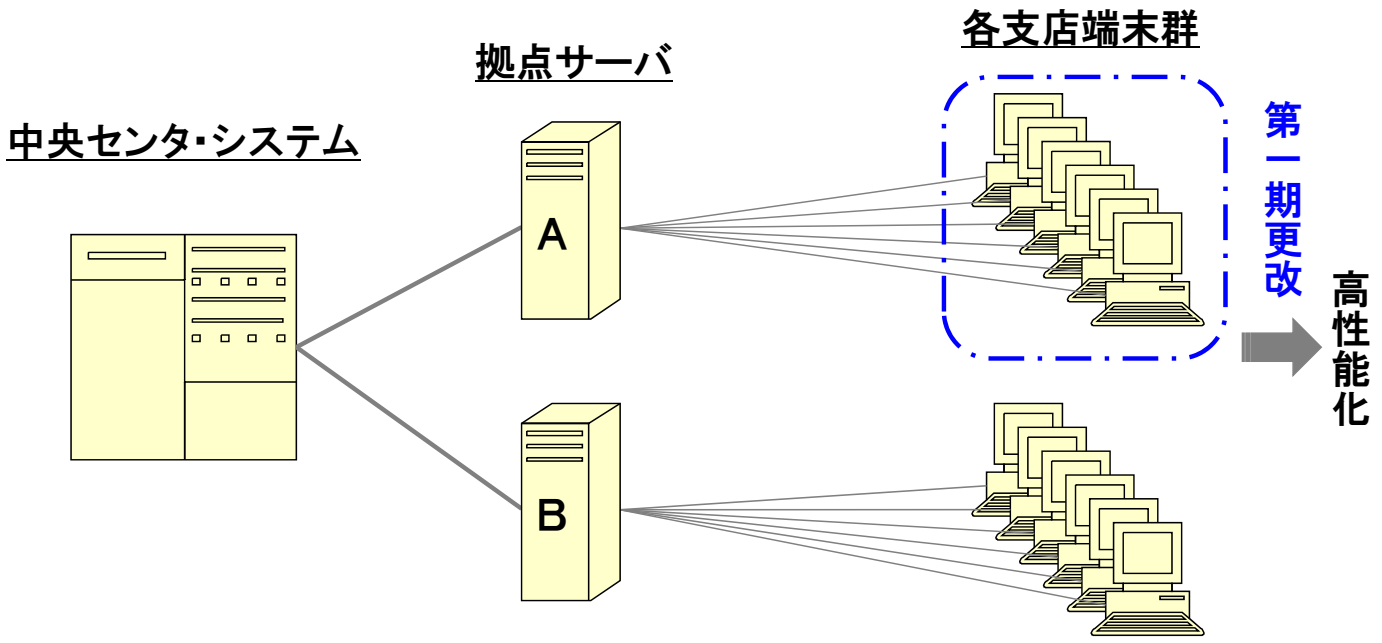
(個別)コンテキスト

【教訓の活用例(1)】

比較的大規模なシステムにおいて，システムの構築あるいは更改を段階的に行う場合，版や性能等の異なる構成要素が混在することになるケース

多数のサーバと端末装置から成るシステムにおいて，当初は全体を一括で構築したものの，その後の端末の更改を，毎年一部ずつ順に行うような場合，新規端末の性能が当初端末より高い場合には，サーバとの通信における応答待ちタイマ値等をチューニングし直す必要があるかどうか，検討.

大規模システムの段階的更改



教訓の一例(5/5)

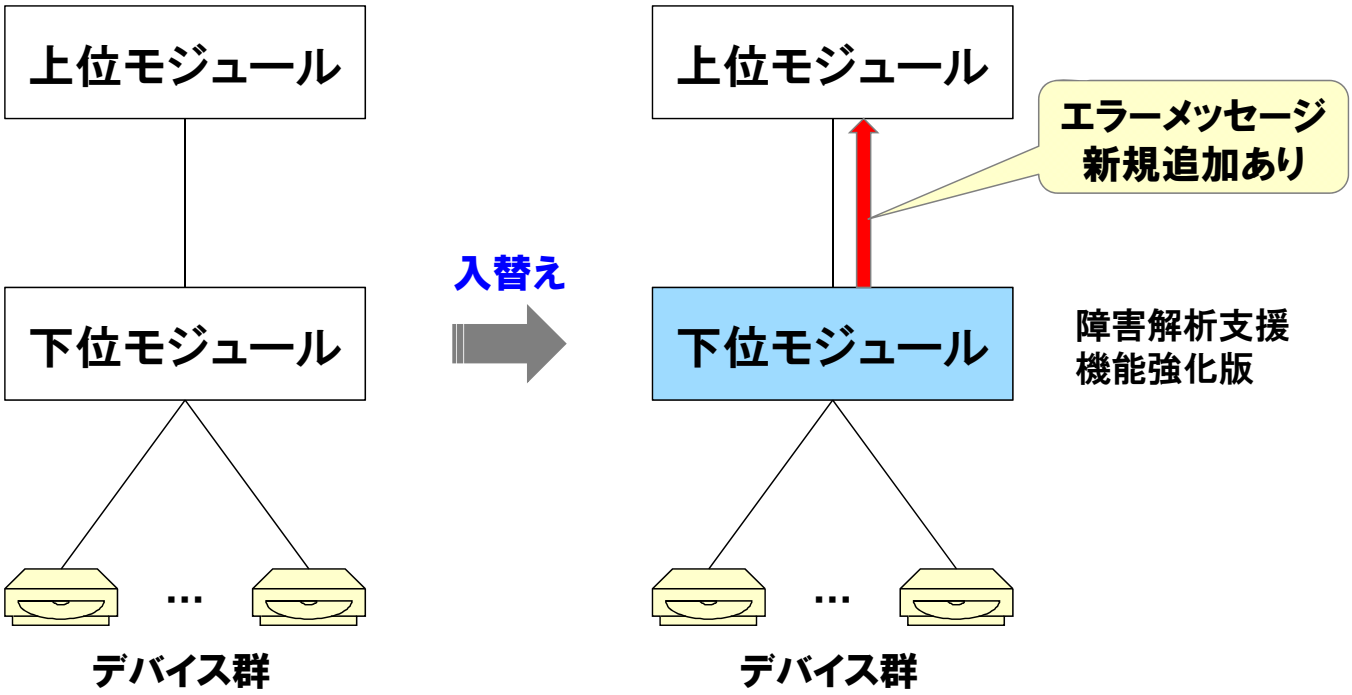
(個別)コンテキスト

【教訓の活用例(3)】

システムの定期メンテナンスにおいて，一部のハードウェアコンポーネントあるいはソフトウェアモジュールをその時の最新のものと交換するケース

交換予定の最新のハードウェアコンポーネントあるいはソフトウェアモジュールでは，従来機能に加えて，機能拡張が行われていることがある．その場合，何らかの条件で，それら新規ハードウェアコンポーネントあるいはソフトウェアモジュールから，従来インタフェースにはなかったエラーメッセージが上位ドライバモジュールに報告されると，上位ドライバモジュールが異常停止するかもしれない．そのような可能性について，確認．

機能拡張されたモジュールとの交換



終わり(障害事例から教訓を導く例)