SERCフォーラム 情報システムの障害対応とソフト ウェア保守を考える

2016年5月19日(木) 13:30-17:00 総合区民センター 第2研修室 (東京都江東区大島4-5-1)



ソフトウェア・メインテナンス研究会

SERC (Software Evolution Research Consortium)

プログラム

13:00-13:30 受付

13:30-13:35 開会

SERC 幹事 奈良 隆正

13:35-14:35 基調講演 「情報システムの障害状況ウオッチ近年2年間の解説」

独立行政法人 情報処理推進機構

技術本部 ソフトウェア高信頼化センター

システムグループ グループリーダー 山下 博之氏

14:35 - 15:35 パネルディスカッション

・パネラー SERC研究員 伊藤 順一、諸岡 隆司、三輪 東、馬場 辰男

・コーディネーター SERC代表幹事 増井 和也

15:35 - 15:50 休憩

15:50 - 16:20 研究報告(1)

情報システムの障害に対する準備と対処方法 SERC研究員 鈴木 勝彦

16:20 - 16:50 研究報告(2)

障害報告書の書き方、お詫びの仕方

SERC研究員 高橋 芳広

16:50 - 17:00 クロージング SERC研究員 大島 道夫



近年のソフトウェア障害の特徴

~IPA/SECによるシステム障害事例分析に基づく教訓~

ソフトウェア・メインテナンス研究会 (SERC) フォーラム 「情報システムの障害対応とソフトウェア保守を考える」 2016年5月19日

> 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)

Copyright © 2013-2016 IPA, All Rights Reserved

sec-sys-infb@ipa.go.jp

IPA Software Reliability Enhancement Center

本講演の趣旨



情報システムの障害は、サイバー攻撃等の情報セキュリティ事案と比べ、一般に、発生頻度は低いものの、ひとたび発生するとその影響範囲は広く、深刻度も高い、世の中を見てみると、障害発生防止のための対策が講じられていても、思わぬ状況や原因により障害は発生している。これを減らすためには、あらかじめすべてのリスク要因を想定することは不可能なため、他所で発生した障害を自システムでは発生しないように対応することが有効であり、そのためには、障害事例情報の共有が必要である。

IPA/SECでは、2013年度から、10程度の分野の事業者のIT部門からお集まり頂く委員会において、一定の守秘義務の下に、各社の障害事例を紹介して頂き、その根本原因と再発防止策等について多方面から議論している。そして、その結果を抽象化・普遍化してまとめ、「教訓(集)」として公開している。

本講演では、まず、IPA/SECにおけるシステム障害事例情報共有の<u>取組みの</u> 概要を紹介した後、これまでにまとめた障害事例の分析に基づく教訓のうち、 特に保守関連の教訓について説明する。また、時間が許せば、保守作業中に発生したある障害事例について、その分析過程を詳しく説明する。

内容



- 1. IPA/SECにおける障害事例情報分析・共有活動
 - 背景
 - 取組みと成果概要
- 2. 保守関連の障害事例の分析に基づく教訓
- 3. 障害事例から教訓を導く例
- 4. まとめ

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

3

内容



- 1. IPA/SECにおける障害事例情報分析・共有活動
 - 背景
 - 取組みと成果概要
- 2. 保守関連の障害事例の分析に基づく教訓
- 3. 障害事例から教訓を導く例
- 4. まとめ



ソフトウェアは、それ自身、複雑化・大規模化し、 システム間連携により、複雑化は一層進展



停止・異常動作等のリスクの増大

:: 市民生活や社会経済活動がITシステムに大きく依存

社会リスクに比例して、ビジネス・リスクも増大

Copyright © 2013-2016 IPA, All Rights Reserved

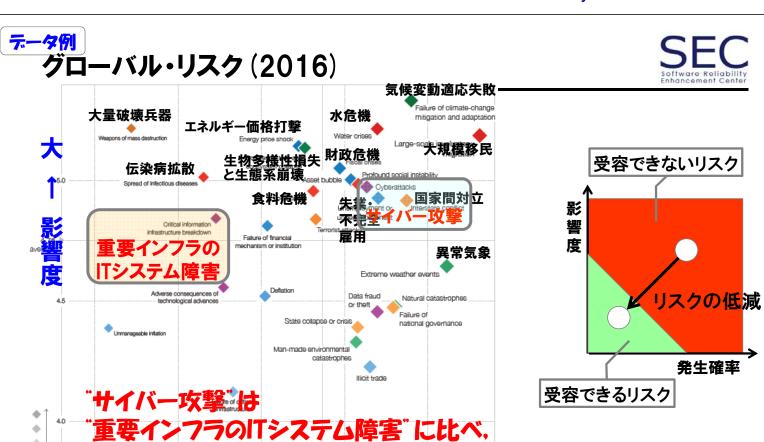
mpact

Col

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

į



発生確率、発生時の影響ともに大きい

Figure 1: The Global Risks Landscape 2016

the World Economic Forum

〈左図の出典〉

http://www.weforum.org/reports/the-global-risks-report-2016

The Global Risks Report 2016 11th Edition,

データ例

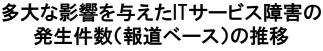
情報処理システム障害の発生状況

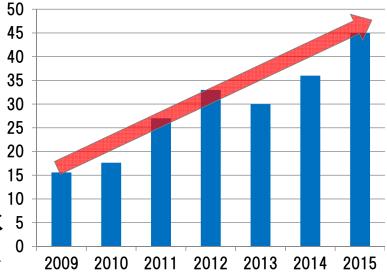


社会に大きな影響を与えた システム障害の発生件数 2009年以降で増加傾向

新聞やテレビなどのメディアでは, 幾度となく以下のようなニュースが 世間を賑わせている:

- ・△△でリコール、国内で数十万台 …理由は、制御プログラムに不具合が発見された ためという。
- ・○○システムで障害か、終日つながりにくく 5 …原因は、法律改正直前の駆け込み需要と期末の 締め処理とが重なり、想定外の大量入力にシステ ムの性能が耐えられなかった模様。
- □□システムで障害、午前中のサービス停止 …原因は、システムは本番装置の故障により予備 装置に自動的に切り替わるようになっていたが、 その切替えが失敗したためという。





(出典) SEC Journal 情報システムの障害状況



類似障害の発生

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

件

IPA Software Reliability Enhancement Center

1.1 障害事例情報分析・共有の背景

情報処理システムの信頼性向上



システムの 構築時→初期リスク(故障)回避

ソフトウェア・エンジニアリング技法の活用

はるかに長期間

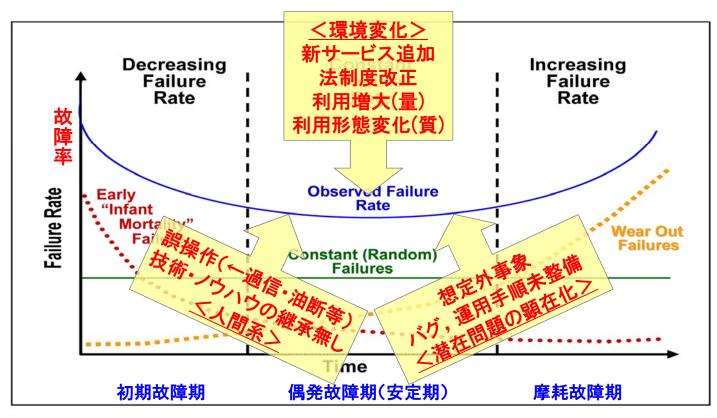
システムの 運用時→様々なリスクに対応

> 体系的な取組みが必要 着目点は...

1.1 障害事例情報分析・共有の背景

システム運用時に想定されるリスク





<故障率曲線の原図> "Bathtub curve" by en:User:Wyatts - U.S. Army document. Licensed under Public domain via ウィキメディア・コモンズ - http://commons.wikimedia.org/wiki/File:Bathtub_curve.jpg#mediaviewer/File:Bathtub_curve.jpg

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

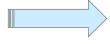
IPA Software Reliability Enhancement Center

1.1 障害事例情報分析・共有の背景

リスクへの対応



ハードウェアは劣化する→故障



冗長構成,など

ソフトウェアは文化しない

ソフトウェアは<u>相対的に劣化する</u>



使われる環境の変化

- ✓ ビジネス方針, ニーズ
- ✓ 組織・人(慣れによる過信・ 油断,交代による技術/ノ ウハウ継承無し)
- ✓ 利用者增,技術進展,他

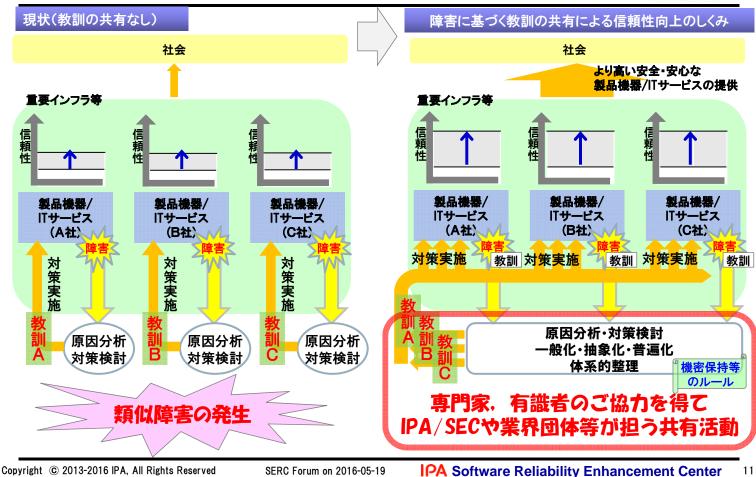
リスク要因

網羅的な事前抽出が困

1.2 IPA/SECの取組みと成果

教訓共有の取組みの目指す方向





1.2 IPA/SECの取組みと成果

(重要インフラシステム等の)ソフトウェア障害情報の収集・分析

【参画企業等】

トヨタ自動車(株)、日産自動車(株) 日本電気(株)、(株)日立製作所 三菱電機(株)、横河電機(株)

富士電機(株)、矢崎総業(株)

アイシン精機(株)

日本電気通信システム(株)

(株)日立産業制御ソリューションズ

三菱電機メカトロニクスソフトウェア(株) (株)富士通コンピュータテクノロジーズ

オムロンソーシアルソリューションズ(株)

アイシン・コムクルーズ(株)

北陸先端科学技術大学院大学

九州大学、会津大学

(一社)組込みシステム技術協会

(一社)電子情報技術産業協会

国民生活や社会・経済基盤 に関わる「障害情報」を収集

【参画企業等】

(株)三菱東京UFJ銀行 日本生命保険相互会社

東京海上日動火災保険(株)

(株)日本取引所グループ

東京電力(株) 東日本旅客鉄道(株)

KDDI(株)

(株)フジテレビジョン

(株)クロスウェーブ

(株)オリジネィション

日本大学

内閣官房情報通信技術総合戦略室

-社)日本情報システム・ユーザー協会

<特徴>

- ①業界・分野を超えて活用可能な普遍化された教訓。
- ②機密保持ルールの下で詳細情報の提供を受けた深い議論。
- ③蓄積されたソフトウェア・エンジニアリングに関する知見活用。

製品・制御(組込み)システム分野

<製品・制御システム高信頼化部会>



収集した情報を分析し



く重要インフラITサービス高信頼化部会>

ITサービス分割

普遍化 取りまとめ



情報処理システム高信頼化教訓集 (<u>ITサービス編</u>/組込みシステム編)

2015年度版:2016年3月31日公開

http://www.ipa.go.jp/sec/reports/20160331_1.html

教訓ごとの随時公開も実施中

http://www.ipa.go.jp/sec/system/lesson.html

2015年度末時点

IPA Software Reliability Enhancement Center

1.2 IPA/SECの取組みと成果

各教訓の構成



あらかじめ実施

はないか?

しておくべき対策

[教訓 I D] 教訓概要(タイトル)

問題:障害事例の内容

原因:問題を引き起こした要因の

分析結果

対策:問題の原因を取り除き再発

を防止するための方法

効果:対策の実施により見られた

/期待される効果

教訓:得られた教訓の内容説明・

補足

各教訓の説明 →後ほど

Copyright © 2013-2016 IPA, All Rights Reserved SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

類似の障害は

起きないか?

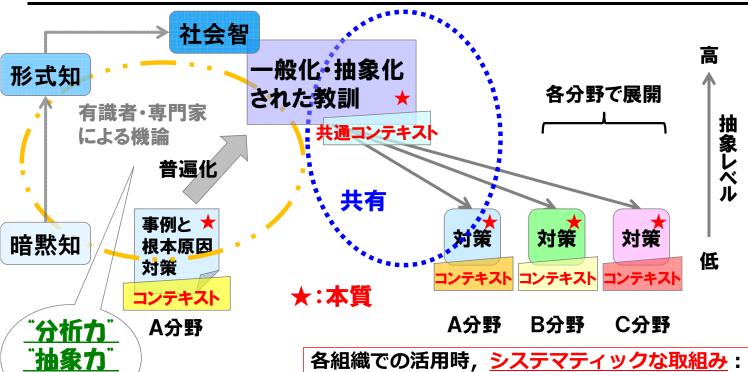
13



1.2 IPA/SECの取組みと成果

教訓の作成と活用の流れ





"想像力" が重要

教訓と共に提供されるコンテキストと 自身のコンテキストとを比較・照合し, 適用可能な教訓について, 具体的対策を検討

が重要

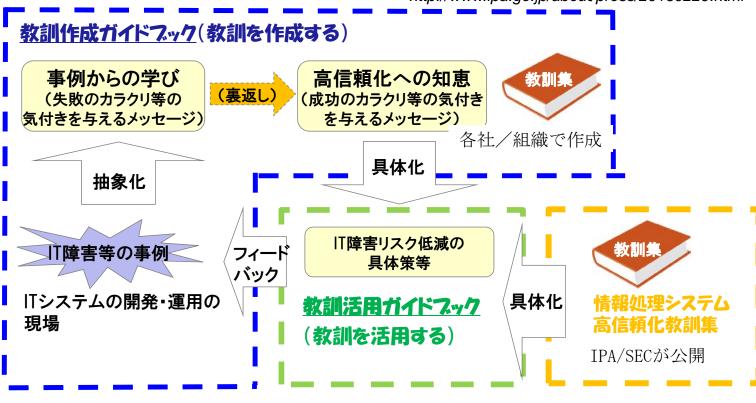
1.2 IPA/SECの取組みと成果

教訓の作成と活用のためのガイドブック



プレスリリース:システム障害を未然に防止するためのガイドブック2編を公開

http://www.ipa.go.jp/about/press/20160229.html



Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

15

内容



- 1. IPA/SECにおける障害事例情報分析・共有活動
 - 背景
 - 取組みと成果概要
- 2. 保守関連の障害事例の分析に基づく教訓
- 3. 障害事例から教訓を導く例
- 4. まとめ





障害事例の分析に基づく教訓 (ITサービス編 概要)

- 2016年 -

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)

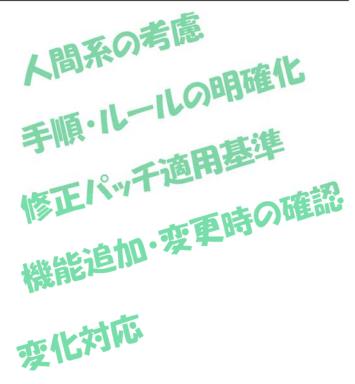
Copyright © 2013-2016 IPA, All Rights Reserved

IPA Software Reliability Enhancement Center

教訓のパターン



- > 人間系の考慮
- > 手順・ルールの明確化
- > 修正パッチ適用基準
- > 機能追加・変更時の確認
- > 変化対応



2. 保守関連の教訓

教訓一覧(ITサービス)[ガバナンス/マネジメント領域]



1)ガバナンス/マネジメント領域の教訓

No.	教訓 I D	教訓概要
1	G 1	システム開発を情シス部門だけの仕事にせず、各事業部門が自分のこととして捉える「態勢」をつくることが大切
2	G 2	発注者は要件定義に責任を持ってシステム構築にかかわるべし
3	G 3	運用部門は上流工程(企画・要件定義)から開発部門と連携して進めるべし
4	G 4	運用者は、少しでも気になった事象は放置せず共有し、とことん追求すべし
5	G 5	サービスの拡大期には業務の処理量について特に入念な予測を実施すべし
6	G 6	作業ミスとルール逸脱は、個人の問題でなく、組織の問題!
7	G 7	クラウド事業者と利用者が連携した統制がとれたトラブル対応体制を整備すべし
8	G 8	共同利用システムでは、非常時対応を含めて利用者間の情報共有を図ること
9	G 9	システム利用不可時の手作業による代替業務マニュアルを作成し定期的な訓練を行うべし
10	G 1 0	関係者からの疑義問合せは自社システムに問題が発生していることを前提に対処すべし」。
11	G 1 1	システムの重要度に応じて運用・保守の体制・作業に濃淡をつけるべし
12	G 1 2	キャパシティ管理では、業務部門とIT部門のパートナーシップを強化するとともに、管理 の 項目と閾値を設定してPDCAをまわすべし
13	G 1 3	キャパシティ管理は関連システムとの整合性の確保が大切
14	G 1 4	設計時に定めたキャパシティ管理項目は、環境の変化にあわせて見直すべし

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

9

G6:

作業ミスとルール逸脱は、個人の問題でなく、組織の問題!



【問題】運用作業者がグループウェアの全ユーザデータを削除

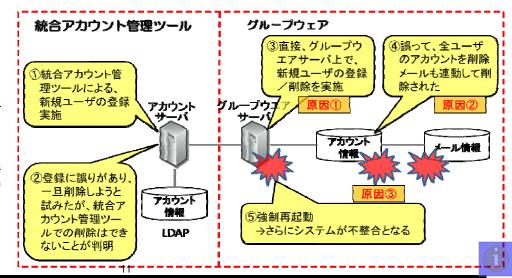
【原因】不慣れな運用作業者(新人)が、独断で、運用規定外の手段(管理ツールを介さないサーバへの直接アクセス)により、誤操作(<u>ルール逸脱</u>)

繁忙な環境下、迅速な処理が求められる状況で、各メンバがお互いの作業に追われて連携できず、不慣れな作業者は、<u>多忙な熟練者にも聞くことができず</u>、自分が業務を遅らせる原因になってはいけないというプレッシャーから、ルール逸脱

運用チーム内のスキ ルの共有も不十分

【対策】組織的な総合対策:

- ・作業を受ける場合の リスクを考慮した受 諾の判断基準作成
- ・複数名体制での作業 実施等、ルールを逸 脱しない<u>作業規定</u>の 作成
- ・普段のチーム内の<u>コ</u> ミュニケーション



G10:

関係者からの疑義問合せは

自社システムに問題が発生していることを前提に対処すべし

【問題】コールセンタにおいて電話

コールの一部が着信後に即切断されてしまう事象が発生していたがイタズラ電話との認識、また通信回線事業者からコールの接続異常が時々発生しているが問題はないかと問合せはあったが他の事業者は正常との回答であったため問題視しなかった。システム障害と気付くまでに4時間経過していた。

【原因】コールセンタ受付システムの回線収容基板上の<u>回線共通バッファがオーバフロー</u>し , コールに対する応答信号の送出が出来なくなっていた.

設定変更ミス: 一部の収容回線の廃止に伴う, 回線試験用設定の削除を忘れた.

回線試験エラー電文が回線 共通バッファに蓄積し,ついにオーバフロー.廃止回 線のため,エラーをおかしいとは思わなかった.

【対策】

- ・交代系への切替えで復旧
- ・<u>保守運用マニュアル</u>の見 直し
- ・バッファ監視機能追加
- 回線事業者連絡会の設置

コールセンタ受付システム

| 全回線の | アジタルPBX | 現用系 | 文代系 | 日線収容基板1(障害) | 上級国を電話回線 | 1 P報 | 上級国を電話回線 | 上級国を同じない |

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

21

G11:

システムの重要度に応じて

運用・保守の体制・作業に濃淡をつけるべし

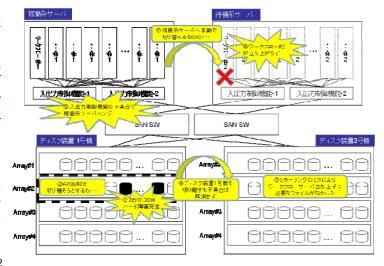
SEC

【問題】A社の社内ワークフローシステムに障害が発生し、連携している顧客向けサービスが終日全面停止した。システム関係者が集まったものの、状況を把握するのに時間が掛かり、システム停止時間は長時間に及んだ。

【原因】二重化サーバに接続されたRAID5構成のストレージ(2重化)において、同一ARRAY 内のディスク2台が同時に故障。当該ARRAYを切り離そうとするも、入出力制御製品不具合によりサーバがハングアップ。サーバを待機系に切り替えようとするも、ストレージのミラーリングの誤設定により必要なファイルが待機系になく、失敗。

ベンダから製品不具合の<u>修正パッチ</u>が提供されていたが、他社で大きな影響が出ておらず、A社に<u>知らされず</u>。 ミラーリングの誤設定は、<u>保守作業の</u> ミス。関係者間でのシステムの共通資料や障害発生時対応マニュアルがなく 、復旧に多くの時間を要した。

【対策】基幹システムを中心に、顧客への影響の有無、推定される損害額等、システムの重要度に応じたランク付けを行い、そのランクに応じてシステム保守対応を実施するルールを設定。



G12:

キャパシティ管理では、業務部門とIT部門のパートナーシップを強化する SEC とともに、管理項目と閾値を設定してPDCAサイクルをまわすべし!

【問題】A社のシステムはサービスの継続を優先するデータの非同期送受信(メッセージ交換)型のオンラインシステムである。このシステムの処理件数には、以前から全般的な増加と共に、時々突発的な事象によるデータ量の急増が見られた。このシステムにある日、処理能力を越えたデータが殺到し、サービスが一時的に停止した。

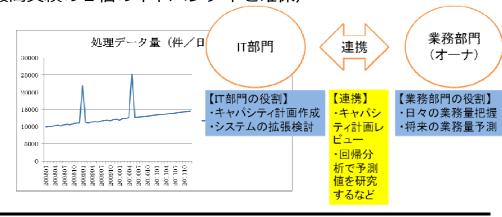
【原因】突発的事象によりデータ量が急増した時にサービス停止に至る原因は、<u>キャパシテ</u>ィに関する業務部門とIT部門との合意形成や管理方法が明確でないことによる。

【対策】①システムごとに<u>キャパシティ管理の責任を持つ業務部門</u>を決め、適材適所で役割 分担し、コミュニケーションをとる<mark>協力体制</mark>を構築。

②過去の実績を基に算出した<u>ルールに基づいて性能を拡張</u>。(例:「突発的な増加に対応可能な」過去最高実績の2倍のキャパシティを確保)

③システムごとに管理項目と閾値を設定し、キャパシティの拡張方法や拡張限界等を明確化。

④業務部門が需要の 将来予測を行い、IT 部門がシステムの拡 張を検討。



Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

23

2. 保守関連の教訓

教訓一覧(ITサービス)[技術領域]



2)技術領域の教訓

No.	教訓 I D	教訓概要
1	T 1	サービスの継続を優先するシステムにおいては、疑わしき構成要素を積極的にシステムから切り離せ
		("フェールソフト"の考え方) 蟻の目だけでなく、システム全体を俯瞰する鳥の目で総合的な対策を行うべし
2	T 2	
3	T 3	現場をよく知り、現場の知識を集約し、現場の動きをシミュレートできるようにすべし
4	T 4	ンスアム全体に影響する変化点を明確にし、その官理ルールを東正せよ
5	T 5	サービスの視点で、「変更管理」の仕組み作りと「品質管理責任」の明確化を!
6	Т 6	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
7	Т7	バックアップ切替えが失敗する場合を考慮すべし
8	T 8	仮想サーバになってもリソース管理、性能監視は運用要件の要である
9	Т 9	検証は万全?それでもシステム障害は起こる。回避策を準備しておくこと
10	T 1 0	メッシュ構成の範囲は、可用性の確保と、障害の波及リスクのバランスを勘案して決定する
11	T 1 1	サイレント障害を検知するには、適切なサービス監視が重要
12	T 1 2	新製品は、旧製品と同一仕様と言われても、必ず差異を確認!
13	T 1 3	利用者の観点に立った、業務シナリオに則したレビュー、テストが重要
14	T 1 4	Webページ更新時には、応答速度の変化等、性能面のチェックも忘れずに
15	T 1 5	緊急時こそ、データの一貫性を確保するよう注意すべし
16	T 1 6	システム構成機器の修正パッチ情報の収集は頻繁に行い、緊急性に応じて計画的に対応すべし。
17	T 1 7	長時間連続運転による不安定動作発生の回避には定期的な再起動も有効!
18	T 1 8	新たなサブシステムと老朽化した既存システムとを連携する場合は両者の仕様整合性を十分確認すべし
19	T 1 9	リレーショナルデータベース (RDBMS) のクエリ自動最適化機能の適用は慎重に!
20	T 2 0	パッケージ製品のカスタマイズはリスクを認識し特に必要十分なチェック体制やチェック手順を整備し
		て進めること
21	T 2 1	作業ミスを減らすためには、作業指示者と作業者の連携で漏れのない対策を!
22	T 2 2	隠れたバッファの存在を把握し、目的別の 闕 値設定と超過アラート監視でオーバフローを未然に防止す
/I		SC C

T4:

システム全体に影響する変化点を明確にし、 その管理ルールを策定せよ!



表示項目数が<u>システムの上限値を超えた</u>ため、全画面表示が消え、オペレータが混乱 ムシステム構築当初から決まっていた上限値について、外部仕様変更に伴う見直しを未実施 原因の本質は、全体に影響する変化点(この場合、予測時間、列車運転本数)が不明確

【原因1】予測時間を4H⇒24Hに変更した際、そのような要件変更があったにもかかわらず、「修正箇所数」の上限値の増加などシステム全体の機能要件変更を未実施

【原因2】列車の本数が年々増加しており、本来ならば(運転本数の増加の都度)上限値を超えた際のシステムの挙動を見直す必要があったにもかかわらず、未実施

【対策】 制御系システムの<mark>変化点の管理ルールを明確に</mark> し、そのルールを守る仕組みを構築

- ・システムが監視・制御する対象と仕様の変化点を網羅
- ・変化点管理のルールとそれを守る仕組みを構築
- ・変化点管理で使用する管理指標を関係部門で共有し、 「変化点の見落とし」を防止

②ダイヤ 修正発生 限値を超過 実績ダイヤ 予測ダイヤ 現在時刻 ④予測ダイヤの表示 ができなくなった

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

25

T5:

サービスの視点で、

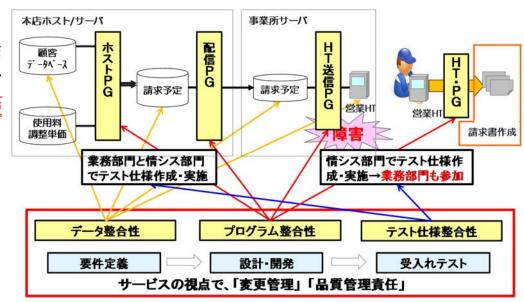
「変更管理」の仕組み作りと「品質管理責任」の明確化を



本店ホスト/サーバから請求データを端末に転送し請求書を印刷するシステムにおいて、端末として、営業員が持ち歩くHT(ハンディ・ターミナル)を新規に導入したところ、そのシステムから出力される請求書の金額が誤ったまま顧客に渡ってしまい、個別謝罪・請求書の再発行に追われた.

システムへの新たな要件追加、 使用方法の変更があると、今ま で正常に稼働していたシステム が突如障害となる(<u>追加により</u> 未使用・未確認のロジックが使 われ、不具合が顕在化)

変更があった時にシステム全体のプログラム、データ、テスト 仕様の整合性を保つための変更 管理を確実に実施 システム全体の整合性を確認す る人を決め、品質管理責任を明 確にし、開発フェーズ毎の検証 を実施



SERC Forum on 2016-05-19

T12:

新製品は、旧製品と同一仕様と言われても、必ず差異を確認し

【問題】2重化された制御系システムにおいて、部品交換の保守作業時にシステム全体の動作が停止し、短時間で復旧できずに、サービス利用者が終日影響を受けた。

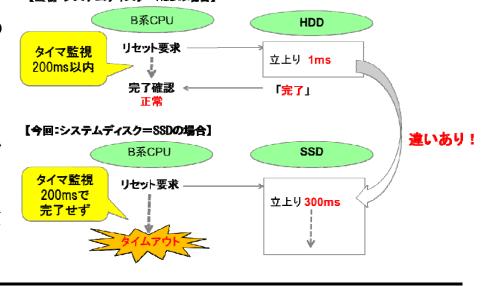
【原因】システムのディスク装置が、数年前に、当初構築時のHDDからSSDに交換されていた。部品交換作業でA系を切り離した時にB系OSから両系のディスク装置にリセット要求が発せられるが、SSDの<u>リセット要求処理時間</u>は、HDDのそれよりかなり長く、OSのタイマ監視において<u>タイムアウト</u>が発生した。その後のリカバリ処理もうまくいかなかった。 【当初:システムディスク=HDDの場合】

SSDへの交換時に、HDD と完全に<u>互換性があると</u> <u>誤認し、検証・テストが</u> 不士分であった。

(対策)・仕様上の互換性を過信

対策】・任様上の互換性を適信 せず、<mark>差異分析</mark>を必ず 実施

・ベンダとユーザの双方 が相手の役割分担を支 援し合う(ユーザ側で ハザード分析を行う)



Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

27

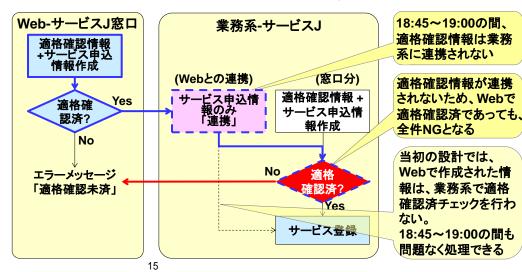
T13:

利用者の観点に立った、業務シナリオに即した レビュー、テストが重要



【問題】オフラインでの申込みのみサポートしていたところを、追加でWeb経由での申込み を可能としたサービスにおいて、特定の時間帯に限り、Webサイトからのサービス 申込みが全て不備とみなされ、登録できなかった。顧客からの連絡で判明した。

【原因】オフライン/Web経由の2系統のサービス申込みを処理するロジックにおいて、各系 統の<u>処理間でのデータの連携に誤り</u>があった。根本原因としては、全体設計が個別 システム設計に正しく引継がれなかったことと、<u>業務シナリオに即した確認が行わ</u> れず、設計後のレビューでも発見されず、対応するテストも行われなかったこと。



ようにした。

Webページ更新時には、

応答速度の変化、性能面のチェックも忘れずに

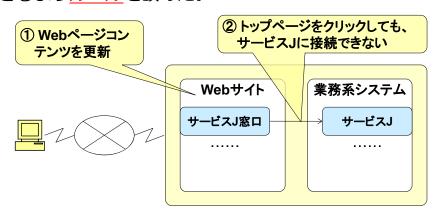


【問題】Webサイト上のあるサービスのトップページをクリックすると、 し、目的のサービスに接続できないケースが多発した。

- 【原因】業務部門がトップページのコンテンツを更新した結果、1顧客当りの<u>ダウンロード</u> サイズが4倍になったが、応答速度への影響を確認しないままリリースした。 業務部門はダウンロードサイズと応答性能との関連を意識せず、それに関するIT部 門による技術的な確認がルール化されていなかった。
- 【対策】業務部門がWebページコンテンツを更新する際には、IT部門が技術的な観点で確認 を行うことを手順書に明記するとともに、IT部門が必要と判断した場合、業務部門 に対しリリース中止を指示できるようルールを改めた。

次の直接的対策も実施:

- 当該トップページへのアク セスを高速ネットワークサ ービス経由に変更
- コンテンツ変更量の自動チ エック機能を導入し、最新 のコンテンツ量とアクセス 量を可視化



Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

T15:

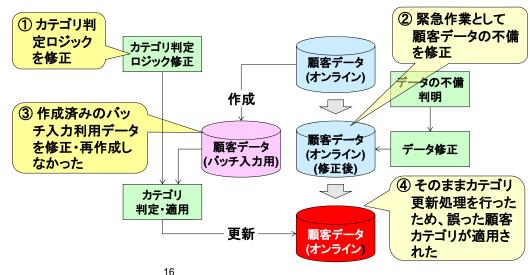
緊急時こそ、データの一貫性を確保するよう注意すべし



【問題】毎月末に、顧客のカテゴリ判定バッチ処理を、あらかじめ作成しておいた顧客デー 夕(マスタ)のコピーを用いて行う運用において、ある時、緊急の要請により、マ スタを修正して対応したことがあった。その後のオンライン処理において、誤った 顧客カテゴリが適用された。

【原因】緊急対応後に、マスタから顧客カテゴリ判定用コピーの再作成を行わなかったため 、マスタとコピーとの不整合が発生していたにも関わらず、そのまま、カテゴリ判 定処理を行ったため。

【対策】緊急時対応の影 響範囲を見極め 、対応結果が平 常時のシステム 運用の流れに確 実に繰り込まれ るよう、特に意 識するよう周知 するとともに、 作業ルール・手 順書を明確化。



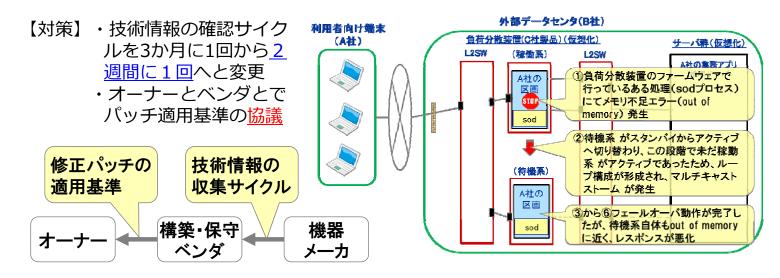
T16:

システム構成機器の修正パッチ情報の収集は頻繁に行い、 緊急性に応じて計画的に対応すべし



【問題】システムの通信機器(負荷分散装置)に障害が発生し、丸 1 日間業務が停止

【原因】システム構築・保守ベンダが外部メーカから調達した負荷分散装置のファームウェ アの既知の不具合が直接の原因であり、その修正パッチが1ヵ月前に公表されてい たが、ベンダによる修正情報の収集間隔が3ヶ月に1回程度と非常に粗く設定して いたため、その適用が間に合わなかった。システムのオーナーは、技術情報が時々 公表されていることを認識していなかった。



Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

T21:

作業ミスを減らすためには、 作業指示者と作業者の連携で漏れのない対策を!

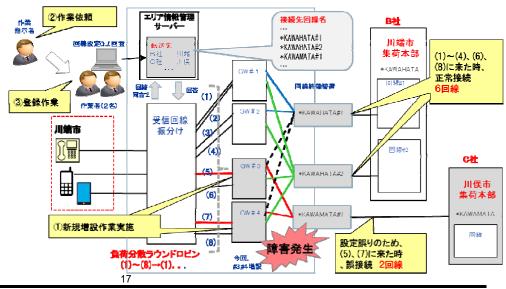


【問題】顧客からの集荷依頼を転送するサービスを行うA社は、振分けシステムの運用ミ スにより、4回に1回の割合でB社への集荷依頼を誤ってC社集荷本部に転送してい た。現場は混乱し、集荷作業漏れが多発し、顧客からの苦情が殺到していた。

【原因】システム(海外製品)のゲートウェイ装置増設に伴う転送情報登録時に、<u>作業者</u> が誤設定。("KAWAHATA"とすべきところを"KAWAMATA"と設定) 根本的には、誤りを犯し、見逃しやすい作業環境と、最後の砦となるべき作業指 示者の確認不足があった。

【対策】・作業手順の明確化 :設定値を自ら読 上げ、設定作業後 の差分チェック、 作業指示者による 確認、全ルート確 認テストの実施

・機器メーカへの依 頼:表示エリアの 限定、ルートの疑 似確認機能の追加 、等



(教訓外事例) JALの重量管理システムの障害



キャッシュへの排他制御がパッチで追加

トラブルの発端となったのは、LHSから提供を受けばALが3月23日に適用 したパッチ。・・・このパッチはJAL以外のユーザー企業からの要望で提供されたものという。・・・「なぜキャッシュの排他制御を施すことにしたのかについて、説明は受けていない。またパッチの内容についても詳しい説明は受けていない」(JAL)という。問題のパッチは「JALの要望で開発してもらったものではなく、JAL側でカスタマイズしたりもしていない」(JAL)という。

JALは、排他制御の見直し以外に、(1)待機系の処理性能を本番系と同程度まで強化する、(2)LHSとの情報共有を密にする、(3)外部ベンダーのエンジニアの協力を得つつ、パッチ適用前の検証などを強化する ――といった対策を進めるとしている。

く出典> JALシステム障害、前週に追加の排他制御がデッドロックを誘発IT Pro, 2016/04/06 http://itpro.nikkeibp.co.jp/atcl/news/16/040601011/

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

33

内容



- 1. IPA/SECにおける障害事例情報分析・共有活動
 - 背景
 - 取組みと成果概要
- 2. 保守関連の障害事例の分析に基づく教訓
- 3. 障害事例から教訓を導く例
- 4. まとめ

3. 障害事例から教訓を導く

事例を用いた説明



【教訓タイトル】

- <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.











Copyright © 2013-2016 IPA, All Rights Reserved

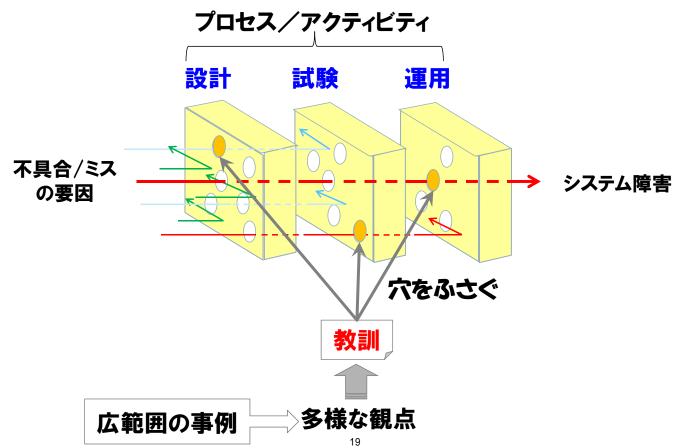
SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

参考

安全文化:スイスチーズモデル





内容



- 1. IPA/SECにおける障害事例情報分析・共有活動
 - 背景
 - 取組みと成果概要
- 2. 保守関連の障害事例の分析に基づく教訓
- 3. 障害事例から教訓を導く例
- 4. まとめ

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

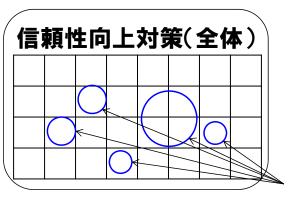
IPA Software Reliability Enhancement Center

37

4. まとめ(に代えて)

体系的な対策に向けて





教訓(再発防止策

教訓に関連するガイド 類を参照し、そのカテ ゴリ全体の信頼性向 上方法を理解する

障害事例に基づく対策は部分的 →

体系的な対策も重要



既存のガイド類に不十 分なところがあれば, その精緻化を図る

ガイド類

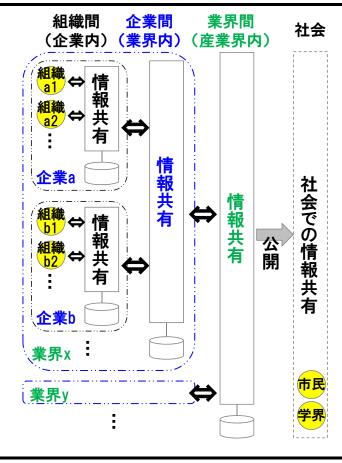
反映

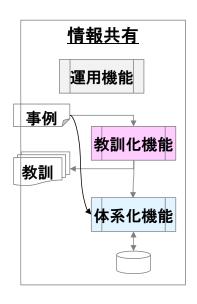
開発/運用等の標準・規定類、組織・体制等

4. まとめ(に代えて)

障害事例情報に基づく教訓共有の仕組みのモデル







各機能主体の例

- 持ち回り
- •(業界)団体
- •IPAなど

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

39



ご清聴、ありがとうございました

http://www.ipa.go.jp/sec/system/index.html

情報処理システム高信頼化教訓集(ITサービス編)-2015年度版http://www.ipa.go.jp/sec/reports/20160331_1.htm/

情報処理システム高信頼化教訓集(組込みシステム編)-2015年度版-

http://www.ipa.go.jp/sec/reports/20160331_2.html

21

以降. 参考



Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center



情報システムの障害情報データ





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

お知らせ:4月26日に書誌情報を追加しました。

詳細検索 情報システム↓ 組込みシステム↓ 統合システム↓

最近のSEC成果 製品・制御システム高信頼化↓ ITサービス高信頼化↓ メリクス分析↓ ソフトウェア品質説明力↓ ソフトウェア高信頼化↓

Home » Search results for "情報システムの障害状況"

キーワード検索 "情報システムの障害状況'の検索結果 ヒット件数:10件

発行日▼ 種別

詳細検索は ころら SEC journal最新 カテゴリ SEC journ SEC journ SEC Reports (6) SEC BOOKS (3 SECセミナー (3) SECイベント (11 **ジャンル** ガイド 調査報告 普及啓発

概要・概説

活用ドキュメント 活用事例

SEコスト本誌表紙:

活動報告

Q

55号) LIJ LI	イエグラ		21170					
ング	2015/09/01	SEC journal	報告	情報システムの障害状況2015年前半データ					
a	2015/03/01	SEC journal	報告	情報システムの障害状況2014年後半データ					
a	2014/09/30	SEC journal	報告	情報システムの事故データ 情報システムの障害状況2014年前半データ					
) 1)	2014/03/31	SEC journal	報告	情報システムの障害状況2013年後半データ					
-/	2013/09/30	SEC journal	報告	情報システムの障害状況 2013年前半デー 情報システムの事故データ					
	2013/03/08	SEC journal	報告	情報システムの障害状況・2012年後半データ					
	2012/09/28	SEC journal	報告	情報システムの障害状況 2012年前半データ					
	2012/03/30	SEC journal	報告	情報システムの障害状況2011年後半データ					
	2012/01/12	SEC journal	報告	情報システムの障害状況2011年前半データ					
ECイ ステ	2011/10/13	SEC journal	報告	情報システムの障害状況2010年データ					

2010年以降

報道された障 害事例をリスト アップし, 部を分析

情報システムの事故データ

SWEiPediaについて 📘 🔊

情報システムの障害状況 2015年後半データ

PA 顧問 松田 晃一

連載

SEC システムグループ 主任

八嶋 俊介

2015年7月から12月までに報道された情報システムの障害状況を報告する。この間に報道された 障害は合計 24 件で月平均 4.0 件となった。平均的な値に対しやや多い値である。とくに今期は の開始に伴いその関係の障害が多く発生している。また、長期間認識されずに運用されてきた不り て発覚した事故が発生している。

1. はじめに

本稿では、2015年7月から12月までの2015年後半の 半年間に報道された情報システムの障害状況をとりまとめ ア朝告する。 すず 一次音で今期の何辺について述べ 続く

ていることが特徴的である。今期から本 され今後重要な社会インフラとして運用 あるため、関連の事故の概要について次に 今期の事例の中には、以前から内在

ベント ITサービス継続 システム課題 プロジェクト管理 経営 Copyright SEC Reports

OLINO I ULUIII UII ZUTU-UU-

Sortware Nerrability ⊾nhancement Center

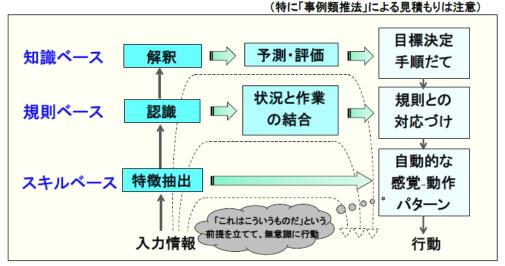
人間系の要因分析(1/2)



錬度"過信"に関わる落とし穴 参考)



■ 習熟度が上がると '思い込み' や '取り違い ' によるミスが増加する。



SRKモデル

source; J.Rasmussen

対策として多面・多重化(人'他視点'、方式)、ワーストケース思考や 標準化の推進などの取り組みを強化する。

<出典> 太田氏(株式会社ジャステック) の調査検討資料

Copyright 2014(C) JASTEC CO.,LTD.

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

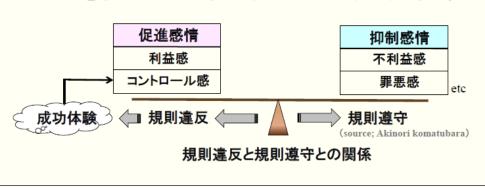
人間系の要因分析(2/2)





参考) 期間短縮に関わる留意点(人間系)

- 不合理な期間短縮を強要すると -
- 人はベテランうんぬんに関係なく、慌てている時に'取り違い'や '短期失念'などのミスが増加する。
- 人は近道心理・合理性心理および対人関係心理(答えたい) などの背後要因から'手抜き(違反)'が醸成され、結果、QCDに ダメージを与えるケースが、ベテランになるほど起こりやすい。



Copyright 2014(C) JASTEC CO.,LTD.

2

IPA Software Reliability Enhancement Center

<出典> 太田氏(株式会社ジャステック) の調査検討資料

人間系の組織能力成熟度





レベル5

(最適化)

レベル4

(管理)

レベル3

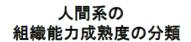
(定義)

レベル2

(反復)

(混沌)





- ■スキル開発
- 組織文化の構築
- ■要員編成
- ■動機付け

一 社員の振る舞いからみた場合の事例

透明性の高い組織文化を背景に、事業目標 達成や取り巻く環境変化を見極め、独創的で 継続的な組織標準*1の改善に挑戦している。

定量的な組織標準*1に基づき、PDCAサイクル を定量的な観点から捉え、予見的な行動様式 (5W2H)に結びついた振る舞いが定着している。

全社(又は事業部)の組織標準*1に基づき、社員は 形骸化に流されず、自ら体得工夫した振る舞いを行う "参加型文化"を形成している。

チーム内で反復可能な見積もり(WBS法等)を実践し ており、メンバーの管理はチームリーダの個性(理念型、 規範型、徒弟型、親睦型等)により統制されている。

見積もりは担当者個人の「勘と経験」や「事例類推法」に 依存され、また、技術オタクなどプロジェクト管理面に無頓 着な社員の振る舞いが根付いている。

組織標準*1:見積り方式、出来高管理手法および品質管理手法など

Copyright 2014(C) JASTEC CO.,LTD.

source: Tadao.Ota

<出典>

太田氏(株式会社ジャステック) の調査検討資料

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

(教訓関連事例)新幹線の電光掲示板不表示トラブルSEC

大型連休後半の4日の始発から, JR東日本管内の東北,上越,北 陸各新幹線の全44駅で、行き先や発車時刻を知らせる電光掲示板 が表示されないトラブルが発生.終日,復旧せず.

連休で増発した列車本数が表示システムの上限を超えたことが原因 とみられる。

同システムは2日分で計1600本が登録できるが、3~4日の列 車本数は1606本だった。

〈出典〉毎日新聞, 2016年5月4日

http://mainichi.jp/articles/20160505/k00/00m/040/017000c

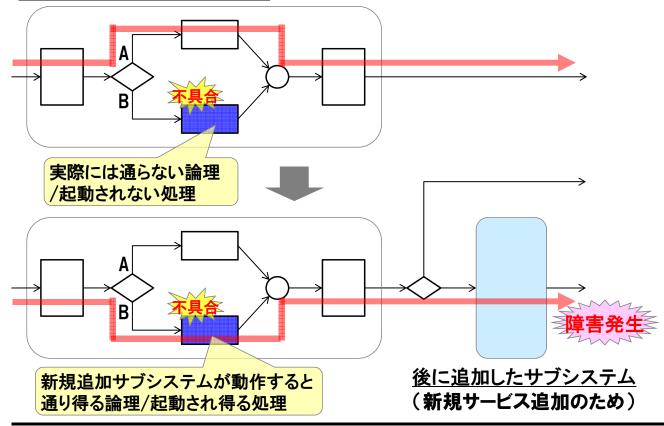
新幹線の列車運行管理システム(COSMOS)において,各駅の運行制御装置(PRC)が旅客案内装 置(PIC)に中央装置から送られてきた5月4日分の運行データを転送した時点で、PIC内のデータ 数が上限値を超えたため、その時にPICは転送されたデータ(5月4日分)をすべて廃棄した.しか し、そのことを示すメッセージ等の制御端末への表示は行われなかった。

参考

新規サービス追加により潜在不具合が顕在化(1/2)







Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

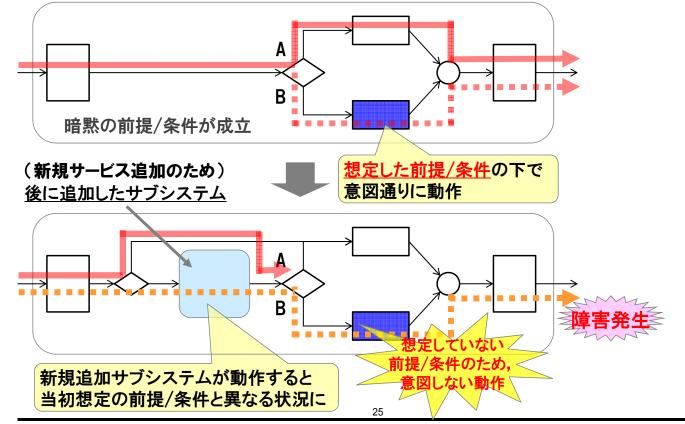
IPA Software Reliability Enhancement Center

参考

新規サービス追加により潜在不具合が顕在化(2/2)

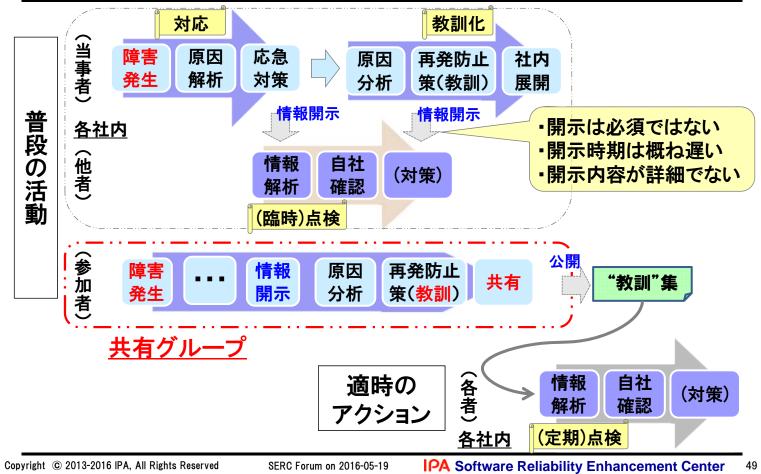


構築当初の基本サブシステム



障害事例の分析に基づく高信頼化活動の俯瞰





障害事例活用

他所でのシステム障害発生時の一般的対応



臨時点検

(他所)障害発生· 情報入手



自システムの 関連性判断



(臨時)点検

※詳細情報の入手の都度,必要に応じ,追加点検

点検や対応の範囲や観点

- …自システムのライフサイクルにおける段階により異なる
 - ●開発前:類似障害が発生しないような方式の採用・対策の実施を,プロセスを含む開発計画に盛り込む
 - 運用中:類似障害の発生するリスク要因の有無確認,発生時の影響の見積り
- 障害の原因により、点検や対応の重点、実施部門が異なる
- ・障害の発生したシステムの応用分野により、自システムと同じ分野であればより慎重になる等、点検や対応への"心構え"が異なる?

26

教訓の一般的活用方法



定期点検

教訓:障害発生から一定の時間が経過. 内容は整理されている.

社内の開発・運用標準に規定された、自システムのライフサイク ルにおける特定の段階で、自システムのリスク評価

(例:チェックリスト)

個々の教訓に関する確認の観点

- 自システムで類似の障害は起きないか?
- 類似障害の発生防止のため、あらかじめ実施しておくべき対策はないか?
- 万一類似障害が発生した場合の影響は、どの程度か?
- 類似障害発生時に影響範囲を小さくする対策はあるか?

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

51

障害事例活用

教訓の恒久的な反映



部署等/ 衫	教訓の最終的な適用先の例									
経営層	社長									
社合信	担当役員			組織に"安全文化"を醸成						
	部門長	組織•	運用 手順 の 整備							
 業務部門	業務推進担当	体制の								
**131 PP 1	システム推進担当	整備			調達		調達			
	関連会社				時の		時の	類似		
.kie 土口	部門長			開発	指示事項		確認事項	事例		
│ 情報 │ システム	システム開発担当			手順 の 整備				トラブ		
部門	システム子会社					レビュー - 章士 E会		ル発 生時		
	元請けベンダ					·試験 TE D		の		
ベンダ	アウトソーサ	社内教育				項目		原因推定		
	サブベンダ							推定		

<出典(縦軸)> SEC BOOKS 「経営者が参画する要求 品質の確保 ~ 超上流から攻めるIT化の勘どころ ~」, p.37 の 3.2(1)項、p.41 の 4.1





「他山の石」の意味※

他人の誤った言行やつまらない出来事でも それを参考にしてよく用いれば, 自分の修養の助けとなる

失敗に学ぶ → 類似障害の発生を防止できる

「対岸の火事」

※(出典) 文化庁月報 平成23年10月号(No.517)

http://www.bunka.go.jp/publish/bunkachou_geppou/2011_10/series_08/series_08.html

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

53

<mark>绣</mark> 「情けは人のためならず」



「情けは人のためならず」の意味※1

人に対して情けを掛けておけば, 巡り巡って自分に良い報いが返ってくる

"社会間接互恵性"※2

ある個体が利他行動(他者に親切にする行動)を行った結果、 その個体の評価が高まり、他者に行った利他行動が回り 回って別の他者から返ってくる仕組み

「ヒト」は、日常生活で困っている他人を見ると、それが自分の知らない人であっても助けたい衝動にかられ、多くの場合何らかの親切を行う性質を持つ

※1 (出典) 文化庁月報 平成24年3月号(No.522)

http://www.bunka.go.jp/publish/bunkachou_geppou/2012_03/series_08/series_08.html

※2 (出典) 大阪大学大学院人間科学研究科の実証実験成果から 2013年8月8日 http://www.osaka-u.ac.jp/ja/news/Resear@hRelease/2013/08/20130808_1

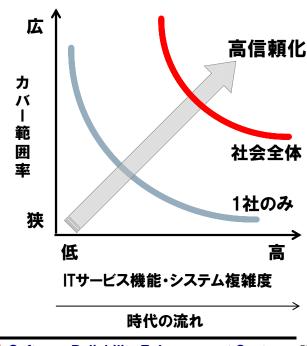
みんなの力で全体をカバー



ITサービスの機能やシステムが複雑化す ると、単一事業者のカバーする知見の範 囲は、相対的に狭くなる.



1事業者に囲われた経験と情報を幅広 く社会全体で共有し、障害対策などに 有効活用できることが重要。



Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

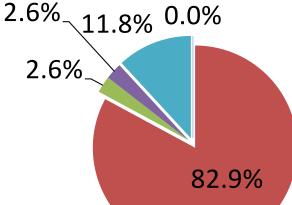
IPA Software Reliability Enhancement Center

アンケート例



障害事例情報を公開する企業に対して、どのように思 われますか?(回答者76名)

■ 1. 障害を発生させていることから、信



2. 積極的に情報公開をすることから、

公開しない企業に比べて信頼できる

- 3. 特に何とも思わない
- 4 . その他

頼できない

無回答

<u> <アンケート回答者(計76名)></u> ET2013(2013年11月) ソフトウェアジャパン(2014年2月)



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.

Copyright © 2013-2016 IPA, All Rights Reserved

IPA Software Reliability Enhancement Center

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<u>鉄道会社</u>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「<u>自動進路制御装置(PRC)</u>」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.

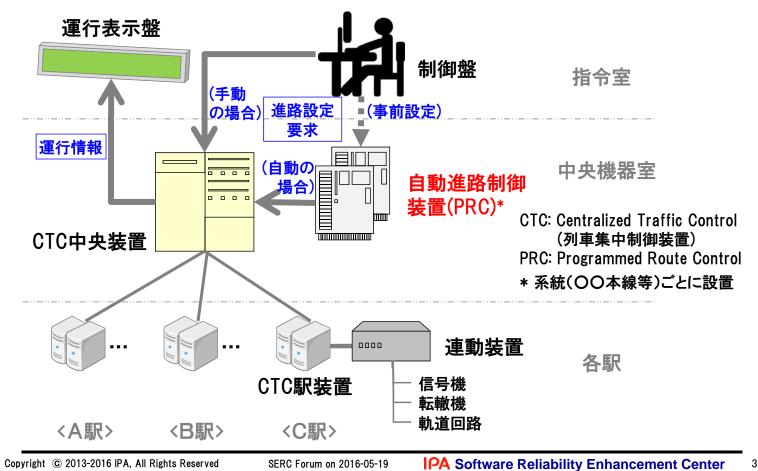
交換対象の部品を戻して再起動する等を試みたが、PRCは復旧せず.

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<u>HDD</u>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<u>SSD</u>に交換された、制御OSの変更はなかった、【設計】 交換当時は、<u>現用・待機両系停止</u>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】

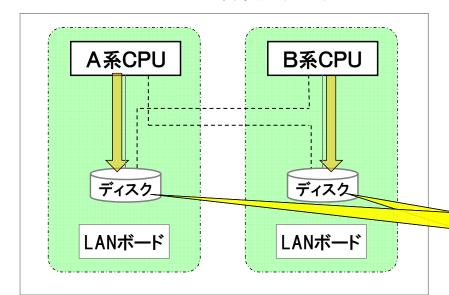
このような状況で、現場での実際の交換作業において、PRCの現用系停止状態で待機系を起動させた。この場合、装置の仕様上、初期起動時に、CPU上のOSがディスクにリセット要求を行い、その完了を<u>タイマ監視</u>(タイマ値=200ms)で待つ。HDDの場合には1msで完了するが、今回の装置に搭載されていたSSDでは300msが必要であった。

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



自動進路制御装置(PRC)



稼働開始後のある時期に、 HDDからSSDに変更された

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

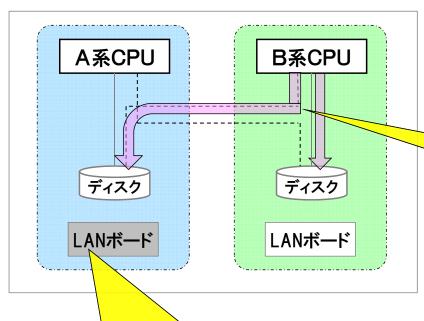
IPA Software Reliability Enhancement Center

5

障害事例A(5/6) 自動進路

自動進路制御装置(PRC)保守時の動作概要 shipper ent center

自動進路制御装置(PRC)



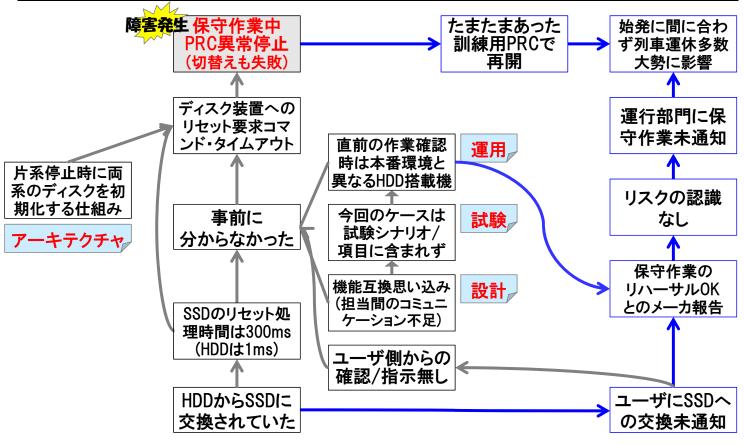
B系CPUから A系、B系のディスクにアクセス

LANボード(障害との直接の関連なし) 交換のため、A系のみ停止

32

障害事例A(6/6) 障害事例の分析例





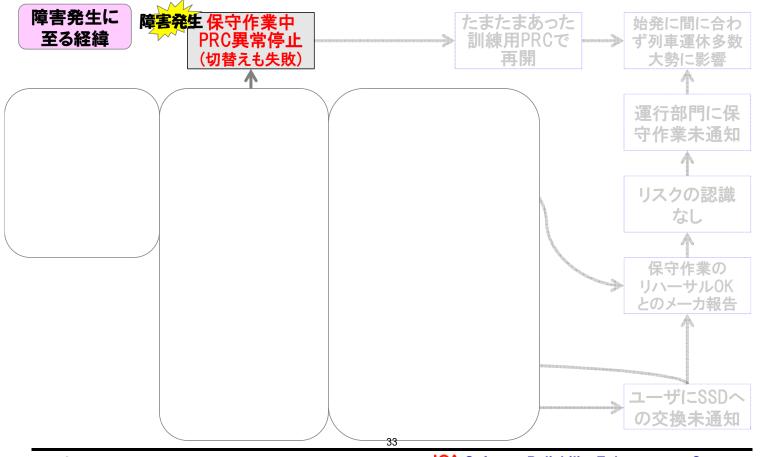
Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

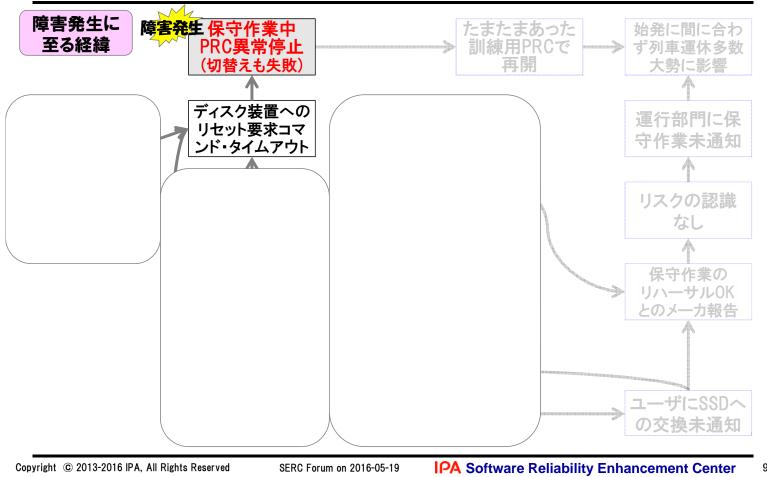
障害事例の分析例:障害発生に至る経緯(01/11)





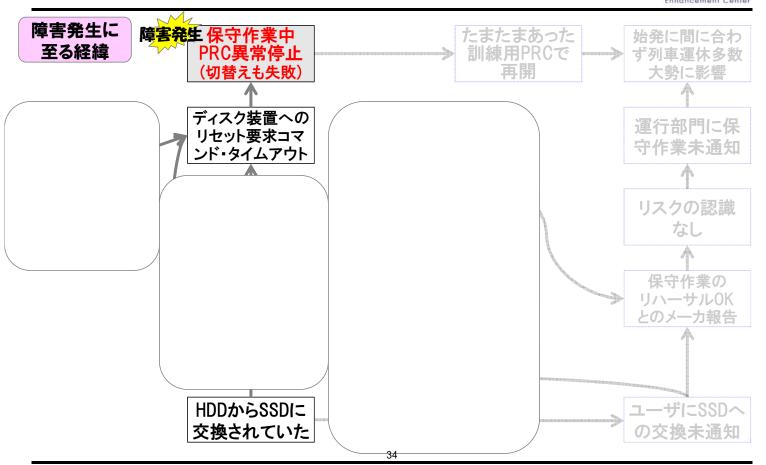
障害事例の分析例:障害発生に至る経緯(02/11)





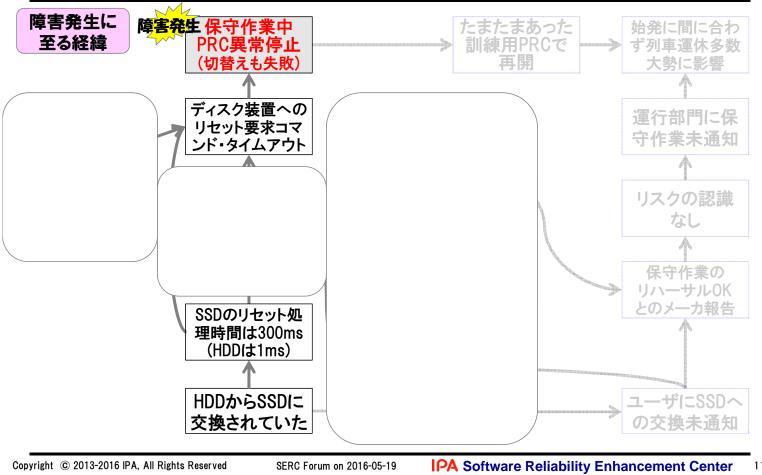
障害事例の分析例:障害発生に至る経緯(03/11)





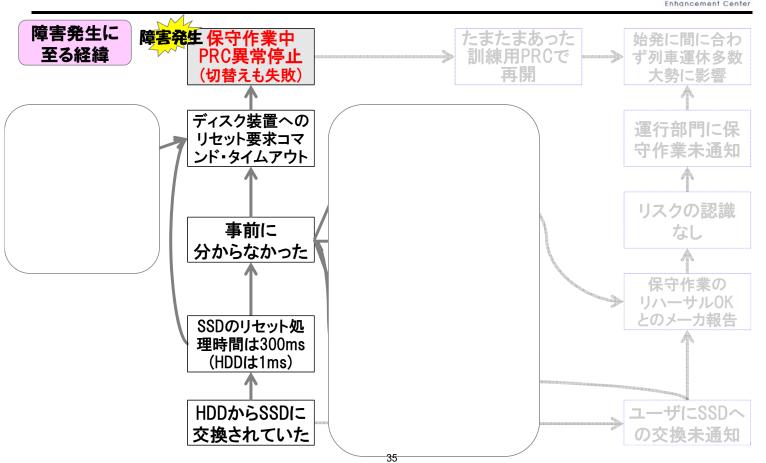
障害事例の分析例:障害発生に至る経緯(04/11)





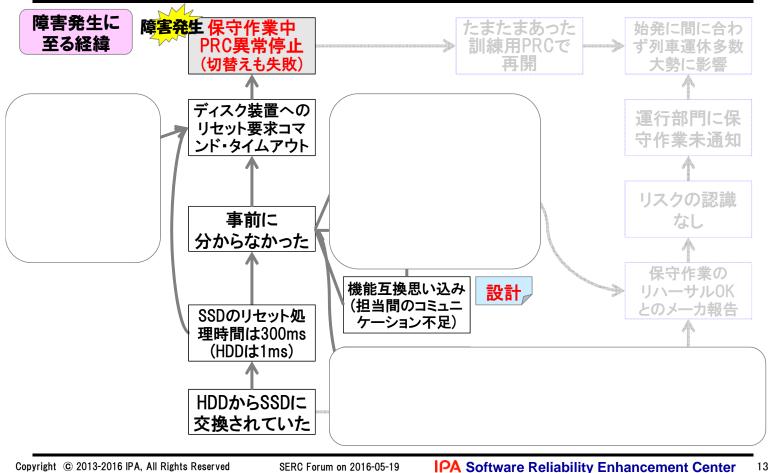
障害事例の分析例:障害発生に至る経緯(05/11)





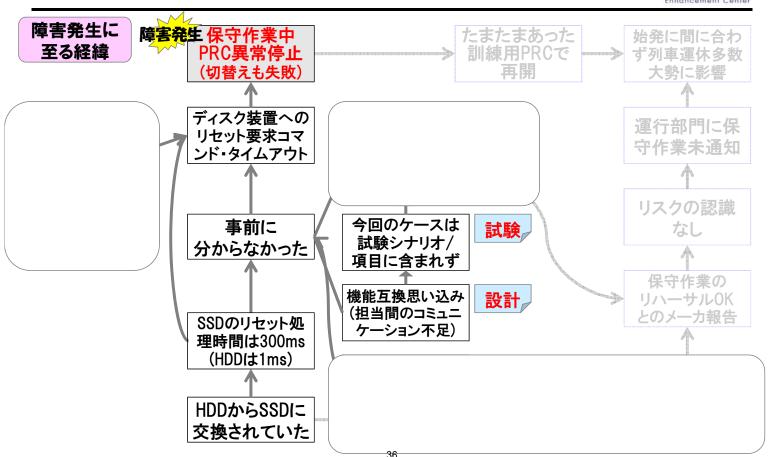
障害事例の分析例:障害発生に至る経緯(06/11)





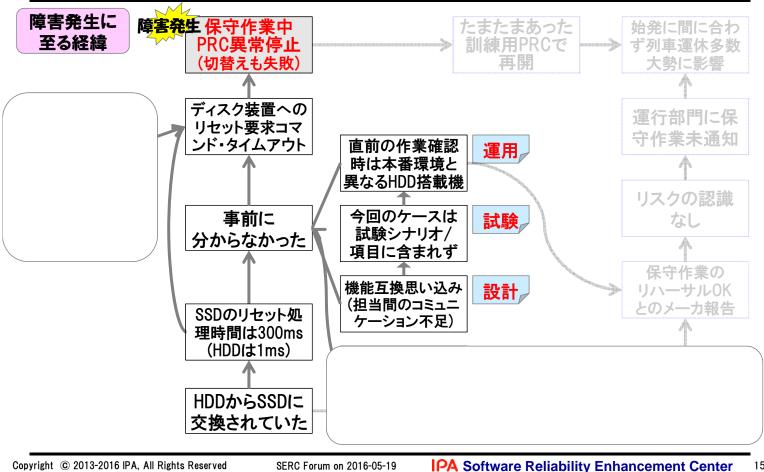
障害事例の分析例:障害発生に至る経緯(07/11)





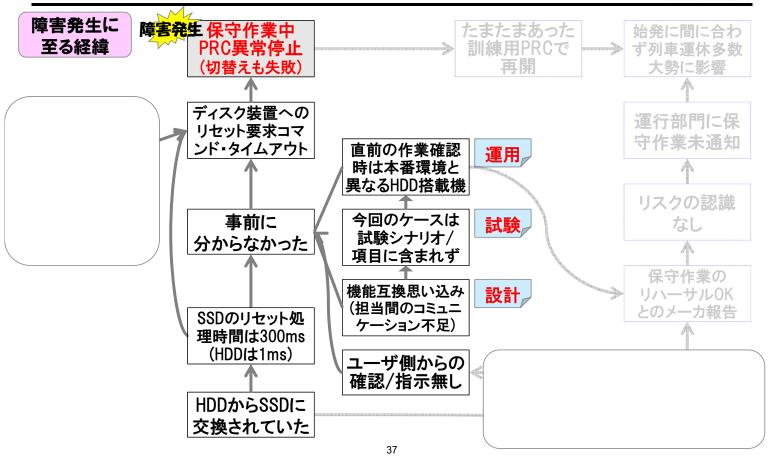
障害事例の分析例:障害発生に至る経緯(08/11)





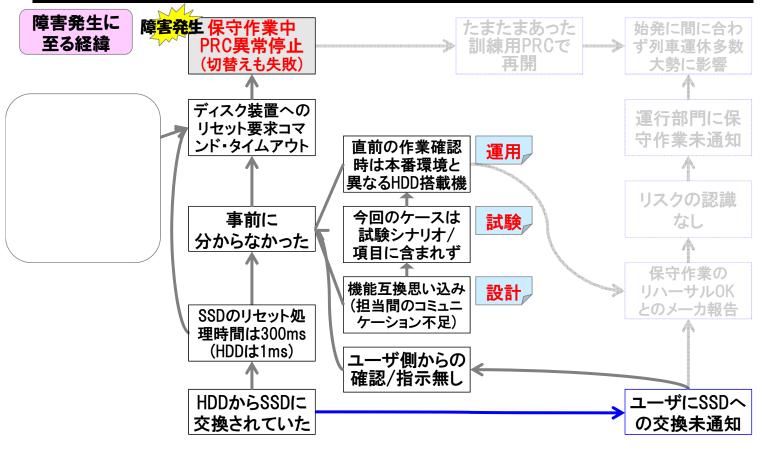
障害事例の分析例:障害発生に至る経緯(09/11)





障害事例の分析例:障害発生に至る経緯(10/11)





Copyright © 2013-2016 IPA, All Rights Reserved

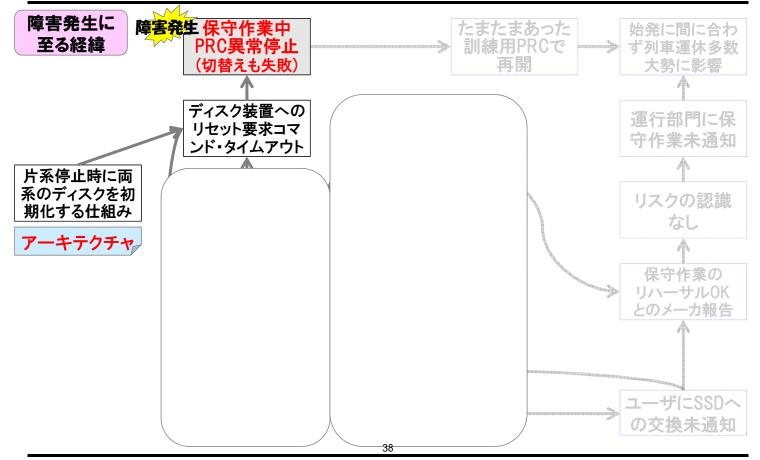
SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

17

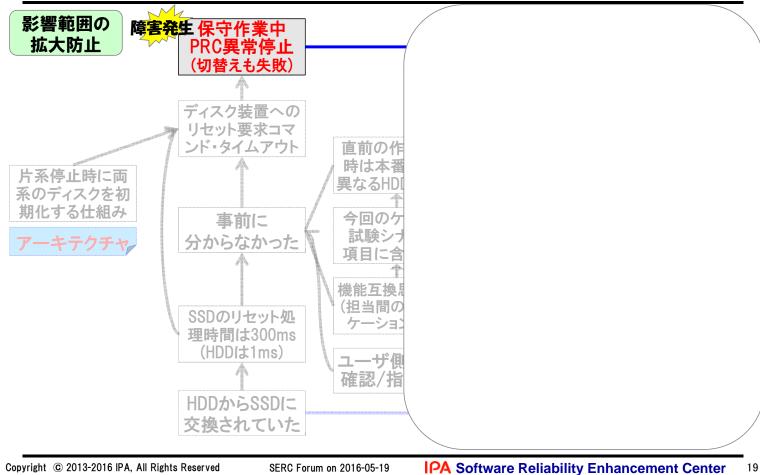
障害事例:障害発生に至る経緯の分析例(11/11)





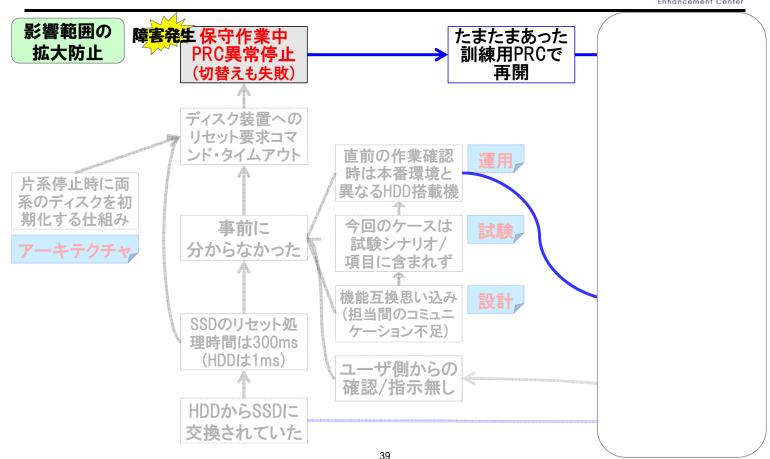
障害事例の分析例:影響範囲の拡大防止(01/07)





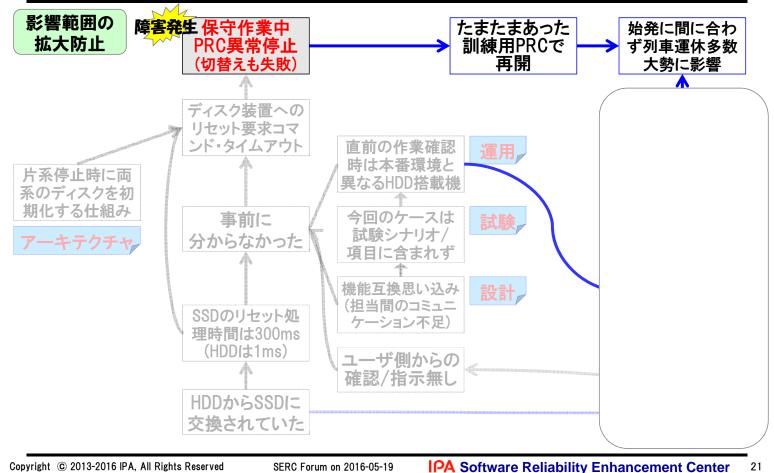
障害事例の分析例:影響範囲の拡大防止(02/07)





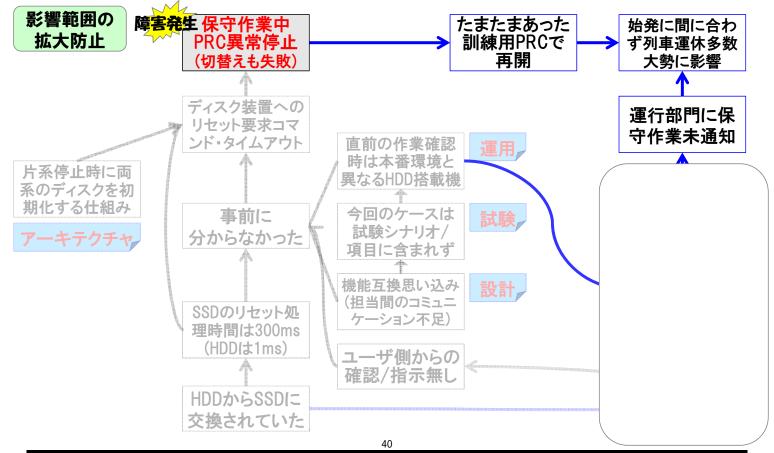
障害事例の分析例:影響範囲の拡大防止(03/07)





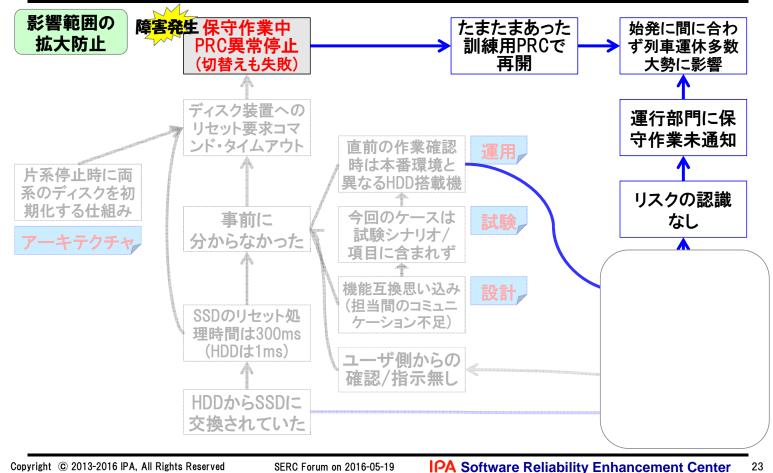
障害事例の分析例:影響範囲の拡大防止(04/07)





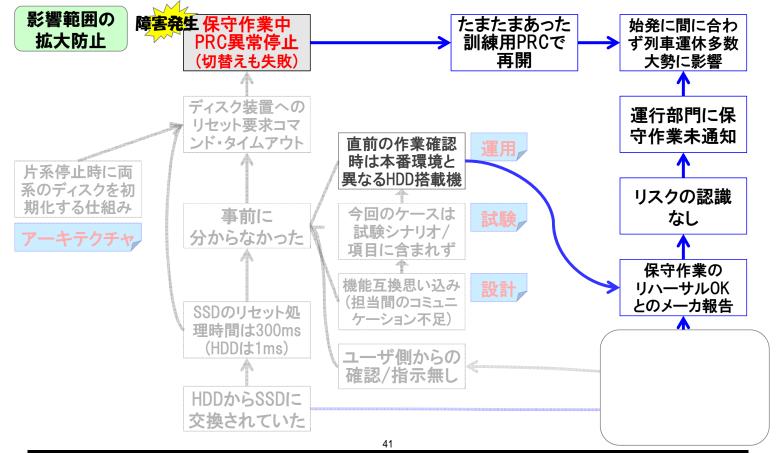
障害事例の分析例:影響範囲の拡大防止(05/07)





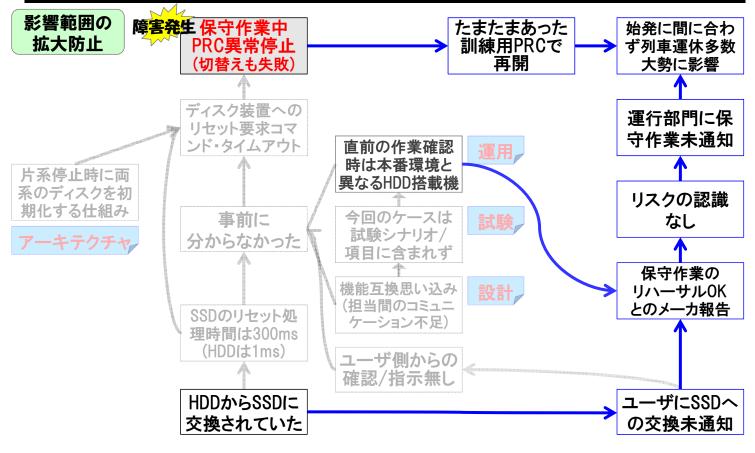
障害事例の分析例:影響範囲の拡大防止(06/07)





障害事例の分析例:影響範囲の拡大防止(07/07)





Copyright © 2013-2016 IPA, All Rights Reserved

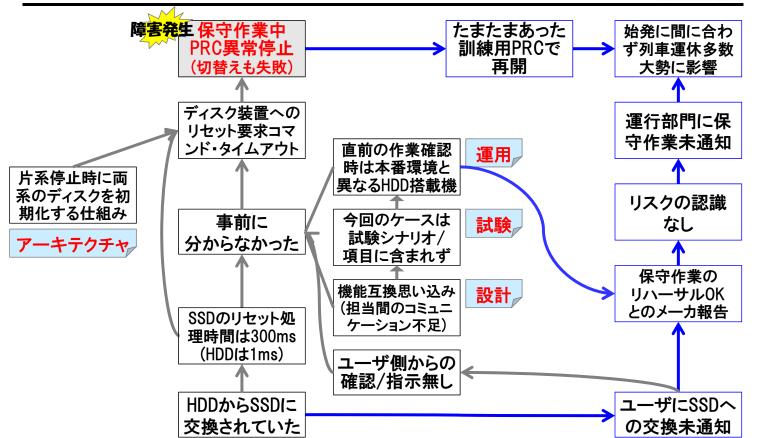
SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

25

障害事例の分析例:詳細





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

<事実>

- 口「これまで付けていた装置
- 口(代替品として)取り付)機能があると説 こ、保守

<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?

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

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。

く事実>

ロディスクのアクセス・インタフェースは、 要求コマンドの実行時間は規格 いを見消す

<質問>

> メーカの担当者には.

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 当性を入念にチェックし、チューニン と影響分析は、全関係者 素の特性がシステムの性素

く事実>

ロ(特になし)

<質問>生

- ▶ 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?

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

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。

く事実>

□HDDからSSDへのディスク交換時間 ストはコストダウング ついての試験は実施されるストラスト

<質問>

- > 今回のシナリオが 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
- > 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?

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>

ロメーカ工場での直前のリハーサル作業では、SSDを体えてかなかったため、HDD 搭載装置を用いて行った。 この情の差異なる。

<質問>

- ▶ メーカの指摘者及び管理者(する)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?

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

31

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>

ロメーカでの設計,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(のえ)保守・運用のお割は?

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/ 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 てその内容や程度を調節する

く事実>

ていなかったため、それ ロメーカからユーサン 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)?

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

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>

Simplicity is the Best! られる仕様となってい ディスク(インタフェース)に対 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?

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.

く事実>

- ロ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 いなかった.
- □ たまたま訓練用の装置があり、それに切り**巻**

<質問>

システムの重要度に応じた冗長構成を ➤ PRC停止のリスクをど

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

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。

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。
- 口保守作業は、深夜でも貨物列車が運 運行の少ない早朝 の時間帯で、始発列車に間
- 口両系全停止の作業の場 系停止での作業方法を

▶ 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?

再発防止策(メーカ)



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

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

再発防止策(ユーザ)



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

抽象化

非機能関連のインタフェースが異なる

異なる

事前の列車運行部門との共有



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

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

39

教訓分析例

教訓の一例(1/5)



【教訓タイトル】

<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

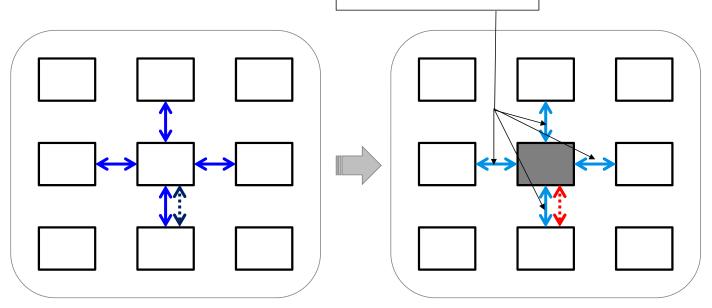
交換(部分変更)する構成要素(ソフトウェアを含む)は,交換前のものと互換性があるという仕様になっていても,機能やインタフェースが一部異なっているかもしれない.特に,新機能が追加されていることがよくある.異なる技術が用いられている場合には,設計思想そのものが異なっていることもある.また,機能は同じでも,性能等の非機能特性が異なるケースは多い.

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.







構築当初のシステム

構成要素の一部を交換

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

41

教訓分析例

教訓の一例(3/5)



【対策(例)】



- ◆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<u>タイマ値等をチューニング</u>し直す必要があるかどうか,検討.

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

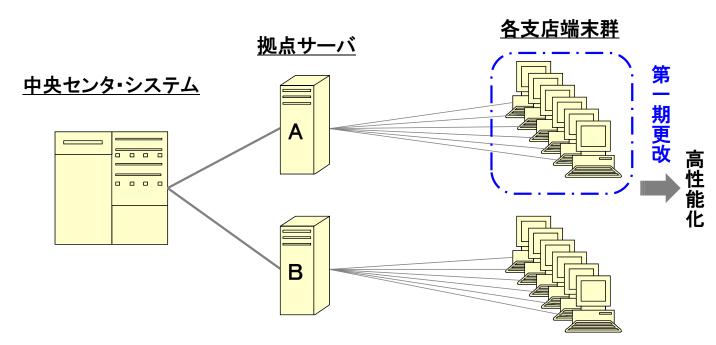
IPA Software Reliability Enhancement Center

43

教訓活用例

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、<u>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。</u>

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

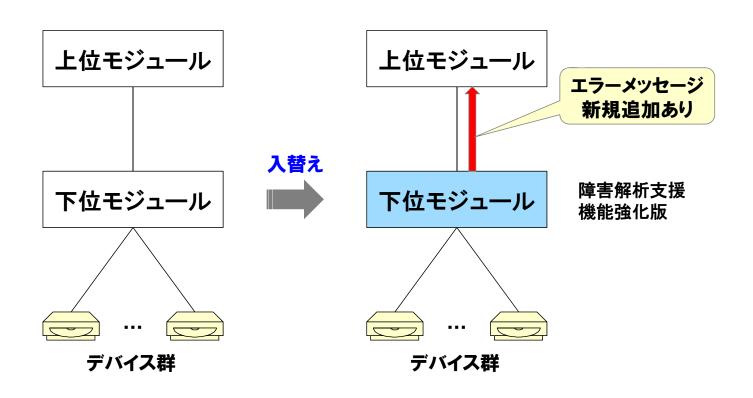
IPA Software Reliability Enhancement Center

45

教訓活用例

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)

Copyright © 2013-2016 IPA, All Rights Reserved

SERC Forum on 2016-05-19

IPA Software Reliability Enhancement Center

47



情報システムの障害に対する準備と対処方法

2016/5/19 株式会社 日立ソリューションズ

鈴木 勝彦





情報システムの障害に対する準備と対処方法

- 1. 障害にも準備が必要
- 2. 環境の変化があったか
- 3. 障害に対しての体制を事前に決めておく
- 4. 事実を正しく捉える
- 5. 事例:性能トラブル時の調査観点
- 6. 原因究明より復旧を優先する
- 7. 長期化したらKT法を使え

- 1. 障害にも準備が必要
 - 2. 障害に対しての体制を事前に決めておく
 - 3. 環境の変化があったか
 - 4. 事実を正しく捉える
 - 5. 事例:性能トラブル時の調査観点
 - 6. 原因究明より復旧を優先する
 - 7. 長期化したらKT法を使え

© Hitachi Solutions, Ltd. 2016, All rights reserved.

1. 障害にも準備が必要

HITACHI Inspire the Next

障害が発生した時に素早く対応するには、事前の準備しておくことが重要です。 準備の状況によって、初動が大きく変わります。

- (1)障害対応マニュアルを作成しておく。
 - (a) 障害の重要度の判定基準 重要度によって体制を変える
 - (b) 担当者の連絡先
 - (c) コールセンタの連絡先とOS/ミドルなどの問合せ時の契約番号
 - (d) 関連する部署の連絡先
 - (e) 社内のエスカレーションリスト
 - (f) ハードウェアの一覧
 - (g) ハードウェア構成図
 - (h) OS/ミドルなどのソフトウェア一覧
 - (i) ソフトウェア構成図
 - (j) ソフトウェアの設定パラメター覧
 - (k) ログ採取ツール(OS用、ミドルソフト用、UP用)

- 1. 障害にも準備が必要
- 2. 障害に対しての体制を事前に決めておく
 - 3. 環境の変化があったか
 - 4. 事実を正しく捉える
 - 5. 事例:性能トラブル時の調査観点
 - 6. 原因究明より復旧を優先する
 - 7. 長期化したらKT法を使え

© Hitachi Solutions, Ltd. 2016, All rights reserved.

1

2. 障害に対しての体制を事前に決めておく

HITACHI Inspire the Next

致命的でかつ緊急度の高い障害が発生した時には、事前に体制を決めておかないと混乱状態となり調査がうまくいかないことがある。緊急時には、2時間間隔ぐらいで障害対策会議を開催する。

■障害時の体制

・指揮者(正/副) :以下の担当のアサインと障害対策会議の運営を推進する。

幹部などから担当に直接指示が出ないようにコントロールする。

・記録者(正/副) :会議中は、ホワイトボードなどに記載し全員に周知できるようにする。

ホワイトボード以外の担当からの調査結果メモなども管理する。

採取したログなども管理する。

・障害回避責任者:回避方法があるかを検討する。

・障害回復責任者:回復作業が必要な障害は、回復方法があるかを検討する。

・原因調査責任者:複数の製品が関係する場合には、製品毎に調査責任者を設置する。

全体の調査責任者は、製品毎の調査結果の共有を図る。

- •再現テスト対応者:現地と同様/類似の環境を作成し、再現テストを試みる。
- ・同件調査責任者:既知のOS/ミドルソフトなどの不良の調査を実施する。
- 報告書作成者 :原因が判明しない状況でも定期的に中間報告書を作成する。
- ・現地連絡窓口 :現地との情報連絡係りを一本化し、現地作業の優先度も管理する。
- ・現地対応者 :マシンルームでは携帯電話などが使えないことがあるので、 必ず2名以上にする。進捗が無くても定期的に連絡する。

- 1. 障害にも準備が必要
- 2. 障害に対しての体制を事前に決めておく
- → 3. 環境の変化があったか
 - 4. 事実を正しく捉える
 - 5. 事例:性能トラブル時の調査観点
 - 6. 原因究明より復旧を優先する
 - 7. 長期化したらKT法を使え

© Hitachi Solutions, Ltd. 2016, All rights reserved.

6

3. 環境の変化があったか

HITACHI Inspire the Next

障害の原因には、いろいろな要因があります。

ソフトウェアの場合には、ハードウェアのような経年変化に起因することはほとんどありません。ソフトウェアは、分岐の塊のようなものなので、条件によって動作が変わります。

今まで動作していたシステムであれば、障害が発生したということは、条件が変わったことが起因している可能性が高いと考えるのが一般的です。ただし、時々ではあるが、タイミングで発生することもあります。

つまり、いつもと違う条件(=環境)の変化があったかを並行して調査することが大切です。

- (1)特定の時刻・時間(time)
 - 時刻(time points):うるう日
 - ・時間(time intervals):長時間運転
- (2)規模が大きくなった時(scale out)
 - •マシンの増設
 - ・支店の増加
- (3)データ量の増加(scale up)
 - ・ 処理件数の増加
 - ・処理データのサイズの肥大化

- (4)データの変化(data)
 - ・処理するデータの内容の変化
 - •ウイルスパターンファイルの変化
- (5)パラメタの変更(parameters)
 - •OSのパラメタを変更
 - ・ミドルソフトの環境設定の変更
 - •UPの環境設定の変更
- (6)システム構成の変更(configuration)
 - 周辺装置の変更
 - 通信経路の変更

- 1. 障害にも準備が必要
- 2. 障害に対しての体制を事前に決めておく
- 3. 環境の変化があったか

🛶 4. 事実を正しく捉える

- 5. 事例:性能トラブル時の調査観点
- 6. 原因究明より復旧を優先する
- 7. 長期化したらKT法を使え

© Hitachi Solutions, Ltd. 2016, All rights reserved.

0

4. 事実を正しく捉える

HITACHI Inspire the Next

クリティカルな障害が発生すると正しい情報だけでなく、推測の情報も混ざり、混乱状態になることがある。

【事実を正しく捉える-その1】

・現地での情報を入手する体制を確保する

日本では、エンドユーザーがプログラム開発して、運用しているケースは少ないので、 開発者が現地にいることはあまりない。クリティカルな障害が発生している時ほど、不思議 と情報が入ってこないことがある。2章で説明したような緊急時の連絡窓口が確立されて いないと、情報が発信されない事態に陥ってしまう。

このため、クリティカルな業務を運用している顧客で重大障害との一報を受けた時点で 現地に人を向かわせることが大切である。

【事実を正しく捉える-その2】

・採取する資料は事前に決めて、採取するためのツールを作成しておく プログラムのログなどは、無限に残すことができないので、いつかはラップしてしまう。 このため、障害発生時には、できるだけ速やかにログの採取が必要である。調査を開始 すると次々に必要となる資料が判明して、追加でログを採取することがあるが、想定され るログは、事前に準備して採取する。発生直後に同時にログを採取することで、同じ時間 帯のログが残り、システム全体の動作も把握できる。

- 1. 障害にも準備が必要
- 2. 障害に対しての体制を事前に決めておく
- 3. 環境の変化があったか
- 4. 事実を正しく捉える
- 5. 事例:性能トラブル時の調査観点
 - 6. 原因究明より復旧を優先する
 - 7. 長期化したらKT法を使え

© Hitachi Solutions, Ltd. 2016. All rights reserved. 10

5. 事例:性能トラブル時の調査観点

HITACHI Inspire the Next

障害調査は、現象とログを基に実施することになる。経験値の高いベテランであれば、勘で原因が究明できることもあるが、長期化することもある。

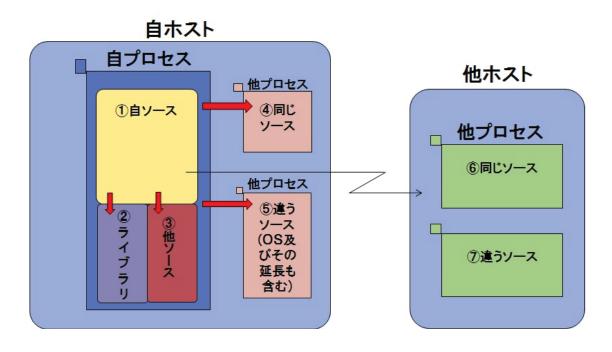
今回は、性能トラブル時の調査観点についての事例を説明する。

性能トラブルは、長期化することが多い。 性能トラブルに対して、経験豊富な特別な技術者でなく、普通の技術者でも4W1Hによる体系的な観点に基づいて調査すれば、試行錯誤することなく、解決までの時間短縮に有効である。

- 一般的に「5W1H」は、When(いつ)、Where(どこで)、Who(誰が)、What(何を)、Why(なぜ)とHow(どのように)であるが、今回は以下のように「Where」と「Why」を除き、Whomを追加して「4W1H」とした。
- -Who(誰が)は、原因となる犯人(プロセス)を究明する。
- •How(どのように)は、原因となる動作を究明する。
- What(何を)は、原因となる資源を究明する。
- •When(いつ)は、原因となるきっかけを究明する。
- •Whom(誰によって)は、原因を誘発したものを究明する。

5.1 Who (誰が) は、原因となるプロセスを究明 (1) HITACHI Inspire the Next

性能トラブルがどのプロセスでどの場所に起因して発生しているかを特定する。



© Hitachi Solutions, Ltd. 2016. All rights reserved.

5.1 Who (誰が) は、原因となるプロセスを究明 (2) HITACHI Inspire the Next

- 性能トラブルがどのプロセスに起因して発生しているかを特定する。
 - 1. プロセスを特定するには、システム全体及びプロセス単位でのCPU使用率、DISK入出力回数とバイト数、データ通信量を確認して絞り込む。
 - 1.1 自プロセスのCPU使用率が高い、DISK入出力回数が多い、または転送バイト数が多い、データ通信量が 多い場合は、自プロセスの可能性が高い時は次の3つが考えられる。
 - ・自プロセスの自分で作成したソースで発生。
 - ・自プロセスだが、取り込んだライブラリで発生。
 - ・自プロセスたが、callしているAPIの延長で発生。(システム関数の場合も)
 - 1.2 他プロセス(自分で作成したソース)のCPU使用率が高い、DISK入出力回数が多い、または転送バイト数が多い、データ通信量が多い場合は、他プロセスの可能性が高く、次の2つが考えられる。
 - ・自プロセスの延長で他プロセスが実行されるが、他プロセスからの戻りが遅いことで自プロセスの性能がでないことがある。この場合は、他プロセスに関して調査する。
 - ・自プロセスは、他プロセスと通信しながら処理しているが、他プロセスからの戻りが遅いために性能がでないことがある。この場合は、他プロセスと通信の状態に関して調査する。
 - 1.3 他プロセス(他人が作成したソース)のCPU使用率が高い、DISK入出力回数が多い、または転送バイト数が多い、データ通信量が多い場合は、他プロセスまたは、システム全体の可能性が高く次の2つが考えられる。
 - ・特定の他プロセスだけ(システムプロセスも含む)がリソースをたくさん使用している場合には、自プロセスからcallしている延長で発生しているかを確認する。
 - ・特定の他プロセスだけ(システムプロセスも含む)がリソースをたくさん使用している場合には、自プロセスが確保するリソースと競合が発生していないかを確認する。
 - 1.4 システム全体のCPU使用率が高い、DISK入出力回数が多い、または転送バイト数が多い、データ通信量が多い場合は、他プロセスまたは、システム全体の可能性が考えられる。
 - ・システム全体がリソースをたくさん使用している場合には、自プロセスが確保するリソースと競合が発生していないかを確認する。

5.1 Who (誰が) は、原因となるプロセスを究明 (3) HITACHI Inspire the Next

2. プロセスを特定した後、さらにどのソースコードで発生しているかを追究する。

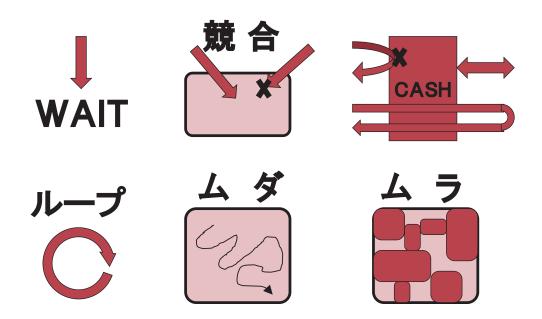
- 2.1 自プロセスでかつ自分のソースで発生 自分のソースなので、トレースログをあれば、場所の特定も容易。
- 2.2 自プロセスであるが、取り込んでいるライブラリ部分で発生ライブラリの前後でトレースログを出力すれば、絞込み可能。
- 2.3 自プロセスであるが、APIの延長(他人のソース)で発生 APIの前後でトレースログを出力すれば、絞込み可能。
- 2.4 他プロセスであるが、自分と同じソースで発生 自分のソースなので、トレースログを強化すれば、場所の特定も容易。
- 2.5 他プロセスでかつ、他人のソースの部分で発生 APIの前後でトレースログを出力し、該当する他プロセスが自分の発行しているAPIの延長(OSなども 含む)であるかを確認する。
- 他プロセスの場合は、他プロセスが特定のリソースを大量に消費していることで影響を受ける場合もある。
- 2.6 他ホストであるが、自分と同じソースで発生。
 - 通信処理の前後でトレースログを出力すれば、絞込み可能。
 - 他ホストとの通信がある場合には、他ホストからのレスポンス待ちで自ホストの自プロセスが遅く見える場合があるが、原因の特定は他ホストがキーになる。
- 2.7 他ホストでかつ、他人のソース 通信処理の前後でトレースログを出力すれば、絞込み可能。 しかし、他ホストの他人のソースが起因する場合には、原因の究明は難しい。

© Hitachi Solutions, Ltd. 2016. All rights reserved.

14

5.2 How (どうした) は、原因となる動作を究明 (1) HITACHI Inspire the Next

・性能トラブルがどうやって発生しているかを特定する。



5.2 How (どうした) は、原因となる動作を究明 (2) HITACHI Inspire the Next

性能トラブルがどうやって発生しているかを特定する。 性能トラブルの原因のメカニズムに関しては、大きく「ループ」、「WAIT」、「キャッシュ」、「ムダ」、 「ムラ」、「競合」に分類できる。性能トラブル解析時には、どのように動いているかを特定する 必要がある。

WAIT

- ・デッドロック
- ・タイマー値大
- ・タイマー値大(リトライ)
- ・タイマー設定でのリトライ多発
- ・相手の処理待ち
- ・無限待ち

競合

- •排他資源
- -CPU
- ・メモリ -DISK
- •通信

- 輻輳

キャッシュ

- ・ヒット率低下
- •キャッシュ不足
- •遅いキャッシュ
- キャッシュが無効

ムラな処理

- ・プライオリティ

- ·断片化
 - バッファサイズの不適切

その他

- ・同居製品固有の実装ロジック
- ・OS固有の実装ロジック
- ライブラリ固有の実装ロジック
- 自プロセスの延長の固有の 実装ロジック
- •ハードウェア特性

© Hitachi Solutions, Ltd. 2016, All rights reserved.

- ・無限ループ
- ・リトライ

-1/0処理を頻発 •同期処理

・ループ中のループ

・コピーの実装が悪い

・メモリ確保関数を頻発

- ・ソート、コンペアの実装が悪い
- サーチ、インデックスの実装が悪い

ムダな処理

キューなどの実装が悪い

5.3 What (何を) は、原因となる資源を究明

HITACHI Inspire the Next

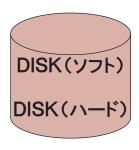
性能トラブルの原因となる資源を特定する。

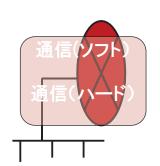






タイマー設定 のWAITなど







5.4 When (いつ) は、原因となるきっかけを究明



・性能トラブルが発生するきっかけ(いつから)を特定する。

エラー

- ・システムコールエラー
- -1/0エラー
- •通信エラー
- ・排他エラー

経年変化

- 大規模になった時
- 長時間の経過後
- データ量が多くなった時
- -機器構成を変更
- -OSのパラメタを変更
- ・ミドルソフトの環境設定を変更
- -UPの環境設定を変更
- •通信経路を変更

通信相手が起因

- ・相手がビジーによるリトライ
- •相手がビジーによる処理待ち
- ・相手がnot ready時のリトライ

DISK

- •DISKビジーによる処理待ち
- •DISKキャッシュの電池切れ

不良

- -他の業務UPの不良
- -OSの不良
- -VMウェアの不良
- ・ファイル管理ソフトの不良
- •ハードウェアの不良

© Hitachi Solutions, Ltd. 2016, All rights reserved.

5.5 Whom (誰によって) は、原因を誘発したものを究明

HITACHI Inspire the Next

・性能トラブルを誰によって誘発されたかを特定する。

ミドルウェア

- ・ウイルス監視製品
- •暗号化製品
- •バックアップ、ディザスタ製品
- ・ファイル転送製品
- ·DB製品

業務UP

- ・同様の業務UP
- ・他の業務UP

その他

- ・他ホスト
- ・その他

5.6 実践で使えるために現象から絞り込む方法



性能トラブル発生時は、4W1Hの観点で調査していくことで絞り込みが可能である。 また、現象から具体的に絞り込む方法については、以下のようなExcel形式でまとめた 資料があるので活用していただきたい。

	ito.	講覧系 (2.41)	HELDING MIT	調査ボイン十2	HEDIORIS	M(5)	訳金場所 十分録	中分數	研査場所 小公舗	MA	調査組み	44
			1			17.						1
55	他理点解,	メモル東用量の毎日	メモノ採用量が急力 に増加しているかを 確認する。	のシャジアイスの時間の経過による機能を発達する。 のスナップアンプによる機能などが原理状況の構造 のテースでよりが確保、関係支援の対し出し。 のアンテムコードトレースでより解消機能の関係の可能が のアンテムコードレースでより解消機能の関係の可能が のアンテムコードレースでより解消機能の関係の可能が のアンテムコードレースでより解消機能の関係の可能が のアンテムコードレースでより解消機能の関係の可能が のアンテムコードレースであり解消機能の関係の可能が のアンテムコードレースであり解消機能の関係の可能が のアンテムコードレースであり解消機能の関係の可能が のアンテムコード・アンテムにより のアンテムコード・アンテムにより のアンテムコード・アンテムにより のアンテムコード・アンテムにより のアンテムコード・アンテムコード・アンテム のアンテム のアンテム	01/ビデリイズの変化とシステムコールト レースから、大きなメモデリイズの機能が、 ・おなサイズを大量に確保したかを確認する。 のがシブを検討し、確保したバモデサイズと 飲を報題する。 のシースで、モリの確保時のサイズと関数を 報題する。	の3xぞ以の確保サイスが巨大。 または の3xで以の確保する概象が拡大	ケス	メモリの確保/開始のロジック		メモリの確保 /開放	終ったが急激に遅くなるような状況のあるかを確認。	OSによっては、大きなメ モバを開催しても、実際 にメモルの書き込みがないとまえせ及ががあって、 大量のメモルの書き込 みも同時に実行ないと 発生しないこともある。
54.	処理が振い	河巴地川東の柳辺	メモ!・検索量が急力 (1時間しているかを 特別する。	(1)メモリサイスの時間の経過による後移を確認する。 (のスナップサンプによる解析のそいのを向け取りの報節) (ステースマメモリの特殊、関係の様の例、出し。 (ボアステムコールトレースでメモ)角体関数の複数の構造	のンデリオスの変化とメモリの機能/開放 のシステムコールが火車に発行されていないかを検認する。 のグランプを解析し、確保したバモリサイスと 数を検討する。 のカースで、アモリの確保時のサイスと関数を 接続する。	データコピー時にコピー先の結構 を17小ずつ拡張しながらコピー している。 データサイズが大きなると指数 関数的にメジ及旧数する。また のため消費し処理に特徴がかか	ソース	メモリの職 体/関数の ロジック		メモリの確保 /開放	メモ/解除のシステムコールが大 登に得行されていないがでわかる	wenの開放などだと関 他の組長でこのような支 続かされているのでは が必要。
57	の経過と共に確くなる)	CPU使用金の確認 UPDUINNの場合に は、snoberの比喩も 確認	時間の配通と共に GPU使用室が増加 しているかを確認する。	のフェングタンプによる解析(メモリの後期は)(20) (27)ナースでメモリの機能、関連改領性の例、(20)。 (27)ナステムコールトレースでメモリ機能関表の認定の構造)	のアンプを開切し、メモル時代化していない。 いかを確認する。 ロア・ファンメモルが確認時のサイズよりいま いサイズで開放するなどの処理がないかの 確認。 「のルモルの確保サイズが可定で持続に実行 まれついるものを課題。	メモリの耐好化。 0.000パイルを確保し30Kパイルが月 開始のような対理を継い見ずと、メ モリが細数なしてメモリの確保に 時間がかかるようになる。)	シース	メモXの機 体/開放の ロジック		メモリの報係 /開放	・自分のノースのメモリの確保/育 終処理 または、使用しているライブラリの 延長で発生していることもあるの で、注意が必要。	- メモリ境開業(更た日の 使用ルモド以かなくでも7 ラブメンテーションのた のスモノデ・ジェラー12な ることがある)
55	処理が振い 個し一時的 (長時間経過法に3時や 急激に遅くなることが ある)	CPURRED ONID UPPLINE DIMEDIC IZ. and has Did TES NID	OFU使用金の時間 の経過による情勢 を確認する。	時やOPU使用面が返しな熱が発生し、やがて解析されるような値 向があるかを検討する。	の)時的に最、時に、CPU販用業が以上に 高いが毛輪語する。 CDOSがあっのようなメモリカガーページコ レクションが実行していないかを確認する。 CDグンプ解析でレモリの付置を確認する。	05のガベージコレクションが実行 されている。	os	ガベージコ レクション 処理の実 続方法			時々だが急力に強くなるような状況があるかを確認。	
50	処理が可い 但し一時的 (長時間経過)と口時か 急激)口機(なることが ある)	CPURREOWES UNCLINIONS は、INCLINOMES 知思	OPURTEO 時間 の経過による情報 を確認する。	時かCPU模能をが高い状態が発生し、やがで解消されるような傾向があるかを確認する。	の)一時的に酸、時に、CPL使用業が以上に 四、かを検討する。 の込みのメゼルカラーページコレクションが 実行していないかを検討する。 (ロタンプ解析でスキリの内容を検討する。)	Javaのガベージコレクションが実 行されている。	Jim	ガページコ レクション 処理の実 被方法			時々たが急激に遅くなるような状況があるかを確認。	使用しているAveによっ て辛飲が異なる。
60	処理が確い 変化器(なる	CPU使用室の確認	CPU使用電が高い 対策が終く	のシンテムコールトレースで通信の回数を発信する。 かつ の窓出プロセスの通信業を計画する。 のシースを解析に、通信問題の販売のロジンクを構造する。 かっ (4減信指子が会れたの場合をある)のプロセスを高臭着が使出る さかを構造する。	○法律の顕微が非常に多いかを報題する。 び課律のデーク室が多いかを報題する。 の法律相手の対して、再進の理 があり、機能は第二次るロジックがあるがを 報問する。 (に通信相手のプロセスがビジーなび続き。 をと、対象では、 をと、対象であるがを報題する。	他プロセスとのデータの受け通し があるが、他プロセスが処理しき かないデータを展する様があり。 つのプロセス間の語彙処理で編 値が発生している。	ソース	通信処理のロジック		5004	存款とサイズを変化させて協能者 変し、グラウ化する。 いっないろな存款で処理時間の計 演	通信相手のプロセスが 処理しまれている細は、 問題が発生しない。 受信プロセスの処理性 総 く送信プロセスの処理性 信 で の状態が何くと 原則を対えると参加に可 くなる。
61	処理が違い 東水道(なる	prue maio feito.	CPURRENAL (のアステムコールトレースで通信の影響を構造する。 かつ の取出プロセスの通信業を計画する。 のアースを解析に、通信処理の展示のロジックを構造する。 かつ (43番件を使れた・の場合もある)のプロセスと高臭薬状態にあ もかを構想する。	の実体の研教が非常に多いかを特別する。 の実体のデータ変が多いかを特別する。 の操作権の対象が表する。 の場合権権が関いなるロジックがあるがを 利用する。 (体験は相手のプロセスがビジーな状態で進 を受け取れない状況であるかを特別する。	他れでなりデータの受け進しが あるが、他プロセスが処理しまれ ないデータを展する理念あり、2つ のプロセス間の課性処理で解析 が発生している。	y-2	通信処理のロジック		siès	作数とサイズを変化させて目標為 更し、グラフ化する。 センス・ウエ件数で別談時間の計 選	通信相手のプロセスが 処理しまれている間は、 問題が発生しない。 受信プロセスの処理性 能 く遂位プロセスの通信性 の状態が有べと。 放果を越えると急激に対 (なる。

「性能トラブル解決の手引き-事例編」の全文は、sercのHPに掲載してある。 http://www.serc-j.jp/

© Hitachi Solutions, Ltd. 2016. All rights reserved. 20

HITACHI Inspire the Next

- 1. 障害にも準備が必要
- 2. 障害に対しての体制を事前に決めておく
- 3. 環境の変化があったか
- 4. 事実を正しく捉える
- 5. 事例:性能トラブル時の調査観点
- ▶ 6. 原因究明より復旧を優先する
 - 7. 長期化したらKT法を使え

6. 原因究明より復旧を優先する



障害が発生すると原因究明の調査を開始するが、同時に復旧も考える。

障害のレベルによって対応の方法も変える必要がある。マシンの再起動を実施すると しばらく稼働が止まるので躊躇するが、原因究明が長期化した場合のために、事前に 再起動するタイミングを決めておくとよい。

- 1. システム全体がダウンしている状態
 - (1)プロセスの状態などを確認して、資料採取してマシンのリブートをする。 OSのコマンドなども実行されない場合には、システムダンプも採取する。
- 2. ミドルソフトが動かない状態
 - (1)該当するミドルソフトに関する資料採取してミドルソフトの再起動をする。
 - (2)ミドルソフトの再起動でも回復しない場合には、マシンのリブートをする。
- 3. 部分的にエラーが発生する状態
 - (1)エラーの起因元を追究して対処する。
 - (2)ミドルソフトの再起動でも回復しない場合には、マシンのリブートをする。
- 4. マシンの再起動で復旧しない場合
 - (1)周辺機器のディスクや通信装置の再起動する

© Hitachi Solutions, Ltd. 2016. All rights reserved.



- 1. 障害にも準備が必要
- 2. 障害に対しての体制を事前に決めておく
- 3. 環境の変化があったか
- 4. 事実を正しく捉える
- 5. 事例:性能トラブル時の調査観点
- 6. 原因究明より復旧を優先する
- 7. 長期化したらKT法を使え

7. 長期化したらKT法を使え



社会心理学者のチャールズ・ケプナー(Dr. Charles Kepner:1922-)と 社会学者のベンジャミン・トリゴー(Dr. Benjamin Tregoe:1927-2005)の 名前に由来する。

KT法には、4つの手法があるが、障害調査は、 「問題の明確化と原因究明」 になるので「問題分析(PA)」を適用するのがよい。

http://jp.kepner-tregoe.com/ http://www.monodukuri.com/gihou/article/390

© Hitachi Solutions, Ltd. 2016, All rights reserved.

HITACHI Inspire the Next

ソフトウェア・メインテナンス研究会

END

情報システムの障害に対する準備と対処方法

2016/5/19 株式会社 日立ソリューションズ

鈴木 勝彦





障害報告書の書き方 お詫びの仕方

SERC研究員 トリプル・アイ企画 髙橋 芳広

目次

- 1. こんなことありませんか 3. お詫びの仕方
- 2. 障害報告書の書き方 3-1 謝罪について
- 2-1目的を理解する
- 2-2 障害報告書の分類
- 2-3 読み手の立場を理解する 4. 参考文献
- 2-4 障害報告書の構成
- 2-5「挨拶」は細心の注意
- 2-6「現象」は調査依頼に合わせる
- 2-7 「原因」と「対策」について
- 2-8「お願い事項」を忘れずに

- 3-2 謝罪の法則
- 3-3 クレームになったら

1. こんなことありませんか?

- ・障害報告書を何度も何度も書き直しになった→上長、関連部門、顧客
- 怒っている顧客に謝りにいくのが気が重い
- 怒った顧客に報告書を破られた
- ・クレームになった
- ・損害賠償を請求の裁判になった
- 謝罪の担当者がメンタルに...



2. 障害報告書の書き方

2-1 目的を理解する

- ① 情報の共有
- ② 協力して問題を解決する体制をつくる
- ③ お客様の怒りを和らげる
- ④ 信頼の回復
- ⑤ お客様へのお願い事項を伝える

2-2 障害報告書の分類

- (1) 時期
- ① 第一報
- ② 中間報告
- ③ 原因判明時
- 4 調査中断時
- (2) 原因の所在の違い
- ① 自社に原因がある場合
- ② 顧客に原因がある場合
- ③ 両者に問題ある場合(ex.操作ミス+ドキュメントが判り難い)
- ④ 原因が判明しない場合(調査中断時)
- ⑤ 他者ベンダーに問題がある場合

2-3 読み手の立場を理解する

- ① 社内の上司
- ② 社内関連部署
- ③ 顧客担当者
- ④ 顧客担当者の上長
- ⑤ 関連他社・ベンダー
- ⑥ 顧客関連部署・監督官庁
- ⑦ 裁判所

2-4 障害報告書の構成

- 1. タイトル
- 2. 宛名(※営業部門に確認要)
- 3. 発行元/発行日
- 4. 挨拶(※時候の挨拶、儀礼等、お詫びの言葉、方向性の明示)
- 5. 現象
- 6. 調査方針および状況(中間報告の場合)
- 7. 技術的原因と対策(原因が判明した場合)
- 8. 根本原因と対策(原因が判明した場合)
- 9. 今後の日程
- 10. お願い事項

2-5「挨拶」は細心の注意

- 報告会議の流れ/雰囲気が決まる
- 相手側の雰囲気(切迫度)を想定する
- 「申し訳ありません」に注意※詳細は3-6「謝罪」について
- ・誤解を受けない(罪を引き受けない)言葉使い

→原因が明確出ない場合には、原因に結びつかない言葉を使う(「システムの不具合」「動作の不具合」等現象を示す言葉)

※確定するまでは「不良」、「バグ」等は使わない

<u>2-6「現象」は調査依頼に合わせる</u>

・ついシステム側の言葉をつかいがち例)

(依頼)「〇〇コマンドが実行できなかった」

(現象)「〇〇コマンド実行時××プロセスが停止した」

- →再現等同じ現象かどうかの確認が容易
- →「依頼した問題は調査されたのか?」(不信感)
- 原因等の説明の中で正確な現象を記載する

2-7 原因と対策について

・ 原因は図で示す

※電子化されて図示に手間がかかるようになったが、省く と2度手間(再提出)になることが多い

- ・出来ない約束(対策)は提示しない
- 原因と整合性のとれた対策

※この対策でどの原因が防止できるの?

<判り難い例>

<改善例>

(原因) (対策)

(原因) (対策)

(1)000

 $(1) \bullet \bullet \bullet$

(1)000

(1)●●●(原因1)

 $(2)\Delta\Delta\Delta$

 $(2) \blacktriangle \blacktriangle \blacktriangle$

 $(2)\Delta\Delta\Delta$

(2)▲▲▲(原因2.3)

 $(3) \times \times \times$

(3) ■ ■ ■

 $(3) \times \times \times$

(3)■■■(原因4)

 $(4)\Box\Box\Box$

 $(4)\Box\Box\Box$

2-8「お願い事項」を忘れずに

- ・ 資料収集版への入れ替え
- 運用回避策
- 現象再現時の資料収集
- ・対策版への入れ替え
- ・運用上の注意事項(操作ミス等の場合) 等々の依頼事項を忘れずに
- ※詳細な手順等は別紙にする

3. お詫びの仕方

3-1 謝罪について

- ・謝罪の要否
 - →日本的にはなにかを謝る
 - →(謝らないと)心証を害して拗れることも
 - →(謝ると)不利な証拠になる
- ・謝る範囲を限定(フリーハンドで謝らない)
 - →相手に与えた不快感:「ご不快な思いをさせて」
 - →相手が感じた不満:「ご不便をおかけして」
 - →こちらの手際の悪さ:「お手間をとらせて」
- ・ 最悪が想定される場合、経営陣や法務部門に相談!!

3-2 謝罪の法則

- 1. 相手に好かれる 素早く連絡、身だしなみ、傾聴、間違いは認める、褒める 相手との接触回数が多い人
- 相手に恩を売る
 遠方から、地位の高い人、お土産(ビジネス上の)
- 3. (相手が)許す理由を説明する 再発防止策、
- 4. 小さなイエスを手に入れる 簡単に同意が得らえることから
- 5. 権威を利用する 法律判例、数字

3-3 クレームになったら

- ・組織で対応
 - →担当者を孤立させない
 - →経営、法務部門を巻き込む
- ・面会は複数人で
- 「スピード解決」を焦らない
 - →焦ると必要以上に妥協することに
 - →即答できないことは持ち帰る
- 腹をくくることも必要
 - →裁判の覚悟も

4. 参考文献

・入門ソフトウェア障害報告書(第1版)

http://serc-j.jp/archive/post_26.html

- ・「謝罪の作法」 ディスカバー携帯 増沢隆太
- •「クレーム対応の教科書」 ダイヤモンド出版 援川 聡
- うまい謝罪Nanaブックス 間川 清

情報交換会 会費 3000円

17:15 会場「和民 西大島店」へ移動

江東区大島1-34-12 菅沼ビル2·3F

TEL 03-5836-3768 禁煙席を馬場の名前で予約

