

第16回COBOLコンソーシアムセミナー

「一歩進んだ見える化の必要性、  
その4つの視点」

Ver. 0.97.0 (配布版)  
2009年2月25日



株式会社 野村総合研究所  
生産技術革新推進部  
上級テクニカルエンジニア  
木村 綾太郎

目次

1. 見える化の必要性
2. 見える化の4つの視点とその手法
3. 活用事例
4. まとめ

## 目次

---

1. 見える化の必要性
2. 見える化の4つの視点とその手法
3. 活用事例
4. まとめ

### 1. 見える化の必要性 問題を定義することの重要性

---



- **ビジネスは問題解決の繰り返し**
  - 来期、売上げを3%向上させるためには・・・？
  - 3末リリースのためには、後何人必要だろう・・・？
  - 2つの手数料計算プログラム、どうやって共通化・・・？
- **重要なのは、問題を解くことではなく定義すること**
  - 「問題を20日で解決しなければならないとしたら、私は19日かけてその問題を定義する」  
アインシュタイン

1. 見える化の必要性  
問題定義のための重要なプロセス＝「見える化」



■問題定義のための重要なプロセス＝「見える化」

- 来期、売上げを3%向上させるためには・・・？  
⇒売上げの構成要素を明らかにする・・・見える化
- 3末リリースのためには、後何人必要だろう・・・？  
⇒3末リリースのタスク・工数を列挙する・・・見える化
- 2つの手数料計算プログラム、どうやって共通化・・・？  
⇒相違点とその特性を明らかにする・・・見える化

1. 見える化の必要性  
今回、ソースコードからの見える化を中心に



■システムの「見える化」とは

- ヒト・モノ・カネの見える化  
⇒体制図、ハードウェア構成図、見積書・・・
- 運用の見える化  
⇒タイムチャート、ジョブフロー・・・
- 性能の見える化  
⇒レスポンスタイム、TPS・・・

■今回は、「ソースコードからの見える化」を中心に

- ソースコードの見える化 ≠ ソースコードからの見える化
- ソースコードから直接的・間接的に抽出可能な情報を視覚化

1. 見える化の必要性

「ソースコードからの見える化」とは・・・利用シーン

■利用シーン

	システム化計画	概要・基本設計	開発・単体テスト	連結・総合テスト
新規開発			■ 受入品質評価	■ 障害原因調査 ■ 障害同伴調査 ■ 修正影響調査
再構築	■ 現行システムの規模把握 ■ 現行システムの課題把握	■ 現行システムのデータフロー分析 ■ 現行システムのプログラム機能分析	■ 受入品質評価 ■ 現行システムのロジック分析	■ 障害原因調査 ■ 障害同伴調査 ■ 修正影響調査
保守	■ 保守費用の見積り	■ 修正影響調査	■ 受入品質評価 ■ 修正影響調査	■ 障害原因調査 ■ 障害同伴調査 ■ 修正影響調査

1. 見える化の必要性

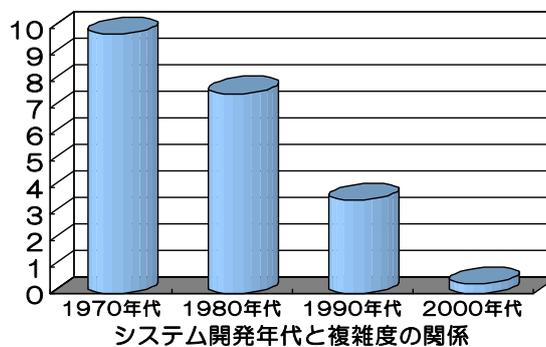
「ソースコードからの見える化」とは・・・具体例 1/3

■複雑度による見える化

非常にメンテしづらくなった古いシステムを、再構築したい

⇒「メンテしづらい」を見える化する必要がある

⇒複雑度により見える化



### 1. 見える化の必要性

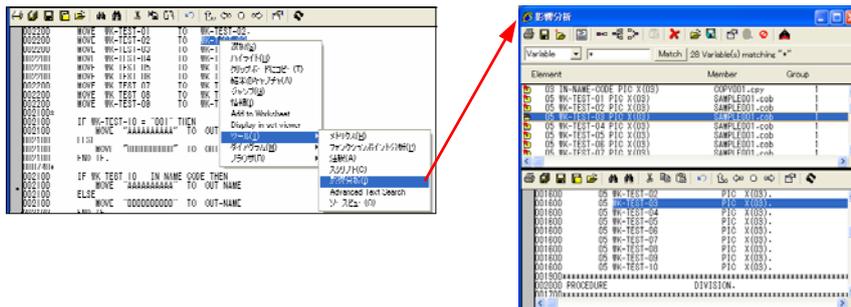
「ソースコードからの見える化」とは・・・具体例 2/3

#### ■波及分析による見える化

商品コードの1桁拡張、影響箇所をもれなく修正できるか不安

⇒「影響箇所」を見える化する必要がある

⇒波及分析ツールにより見える化



### 1. 見える化の必要性

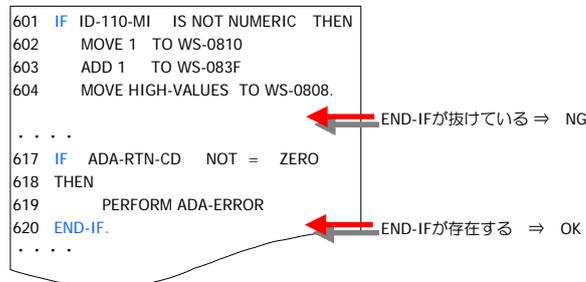
「ソースコードからの見える化」とは・・・具体例 3/3

#### ■コーディングパターンによる見える化

END-IFの書き忘れて障害発生、同件を調べたい

⇒「END-IFの無い場所」を見える化する必要がある

⇒コーディングパターン検出による見える化



## 目次

---

1. 見える化の必要性
2. 見える化の4つの視点とその手法
3. 活用事例
4. まとめ

## 2. 見える化の4つの視点とその手法

4つの視点 切り口1：マクロな視点とミクロな視点

---

### ■マクロ視点

- システム全体が見える化の対象
- 経営者やPMの視点での見える化
- 概要の把握がポイント

### ■ミクロ視点

- プログラム・コードブロックが見える化の対象
- 開発者の視点での見える化
- 詳細の把握がポイント



## 2. 見える化の4つの視点とその手法

4つの視点 切り口2：定型的な見える化と非定型的な見える化

### ■定型的な見える化

- 一般的・汎用的なもの
- プロジェクトによらず共通なもの
- ツールが得意とする分野

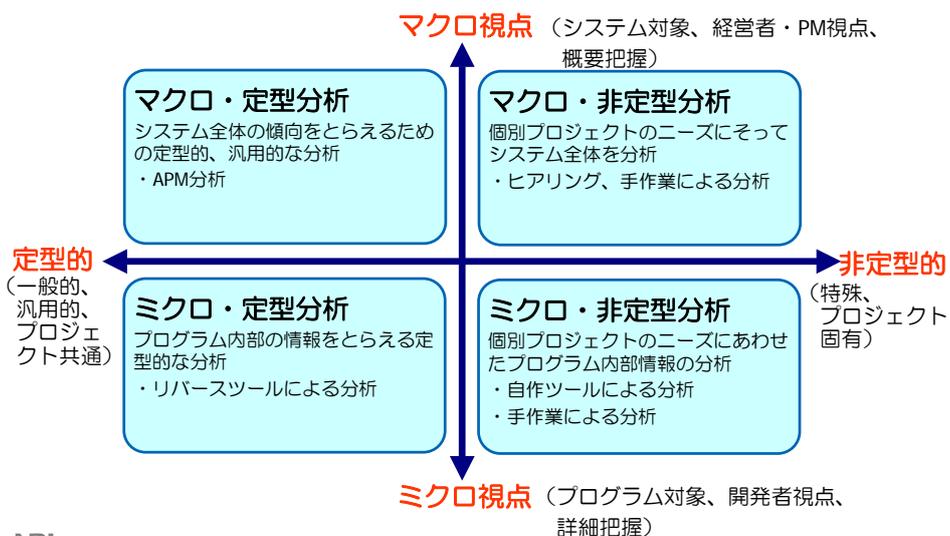
### ■非定型的な見える化

- 特殊なもの
- プロジェクト固有
- 人間が得意とする分野



## 2. 見える化の4つの視点とその手法

見える化の4つの視点＝「ソースコードからの見える化マップ」



2. 見える化の4つの視点とその手法  
マクロ・定型分析とは



■特徴

- システム全体の傾向をとらえる定型的な分析
- システム間、サブシステム間の比較が容易

■代表的な手法

⇒システム定量化分析（メトリックス分析）が中心

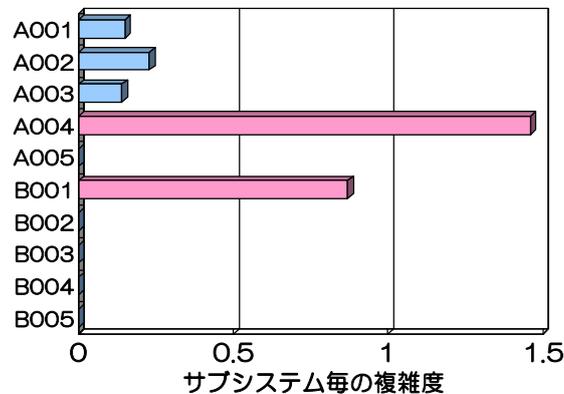
- 複雑度分析
- 重複度分析

2. 見える化の4つの視点とその手法  
マクロ・定型分析の具体例



■複雑度分析

- McCabe複雑度＝保守しにくさを表す指標 を利用
- 複雑度が閾値を超えるステップ数の割合でシステムを評価

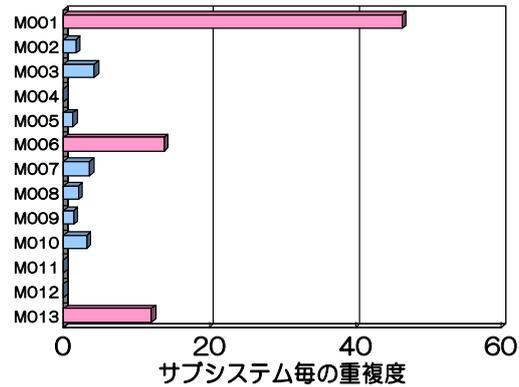


2. 見える化の4つの視点とその手法  
マクロ・定型分析の具体例



■重複度分析

- 重複度＝プログラム間の類似性を表す指標 を利用
- 重複度が閾値を超えるプログラムの割合でシステムを評価



2. 見える化の4つの視点とその手法  
マイクロ・定型分析とは



■特徴

- プログラム内部の情報をとらえる定型的な分析
- 分析を支援するツール（リバースツール）が充実

■代表的な手法

- クロスリファレンス分析
- CRUD分析
- 波及分析

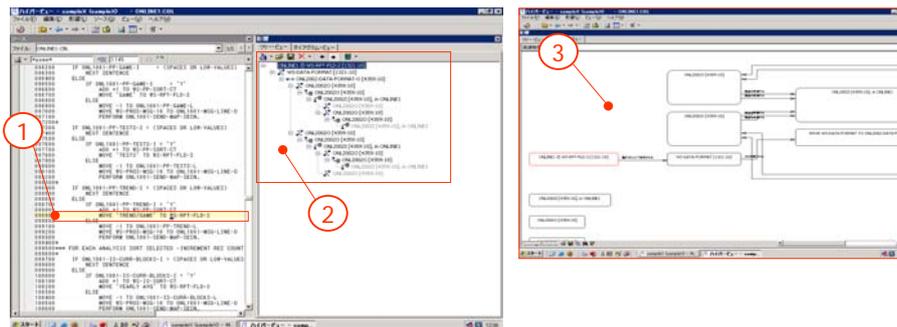


## 2. 見える化の4つの視点とその手法 分析の具体例



### ■波及分析

- データ項目変更の影響がどこまで波及するかを調査
- 2000年問題への対応を契機にツール機能が充実



## 2. 見える化の4つの視点とその手法 ミクロ・非定型分析とは



### ■特徴

- 個別ニーズにあわせたプログラム内部情報の分析  
⇒障害の同件調査・・・例) CLOSEを忘れたプログラムは？  
⇒環境移行調査・・・例) 独自ステートメントの利用場所は？
- 手作業が中心 (ツールを自作する場合も)

### ■代表的な手法

- コーディングパターン分析

## 2. 見える化の4つの視点とその手法 分析の具体例



### ■コーディングパターン分析

- 特定のコーディングパターンに一致する場所をリストアップ
- お勧めは、構文解析技術を活用した分析手法

```
PROCEDURE TERMINATE;  
BEGIN  
-----  
-- END-IFを定義しているかをチェックする  
-----  
FOR X IN ROOT.SubNodes | X IS IfStatement DO  
  IF NOT X.End THEN  
    OUT.WriteLine("END-IFが抜けている行: ", X.EndLineNr + 1);  
  END;  
END;  
END TERMINATE;
```

IFステートメントかをチェックする

END-IFが定義されていないか、  
END-IFが抜けている行番号を表示する

## 2. 見える化の4つの視点とその手法 マクロ・非定型分析とは



### ■特徴

- 個別ニーズに合わせたシステム全体の分析
- システムを俯瞰的にとらえて課題を抽出する  
⇒例) 御社のシステムの課題はx x x、解決のためにx x x  
⇒主に、システム企画・提案フェーズの活動
- 課題の抽出=人間の判断 が中心

### ■代表的な手法

- ⇒今のところ・・・決定的なものは無い
- ⇒非常に難しく、手法が確立されていない

2. 見える化の4つの視点とその手法  
 マクロ・非定型分析の具体例



■ システムフロー分析 (仮)

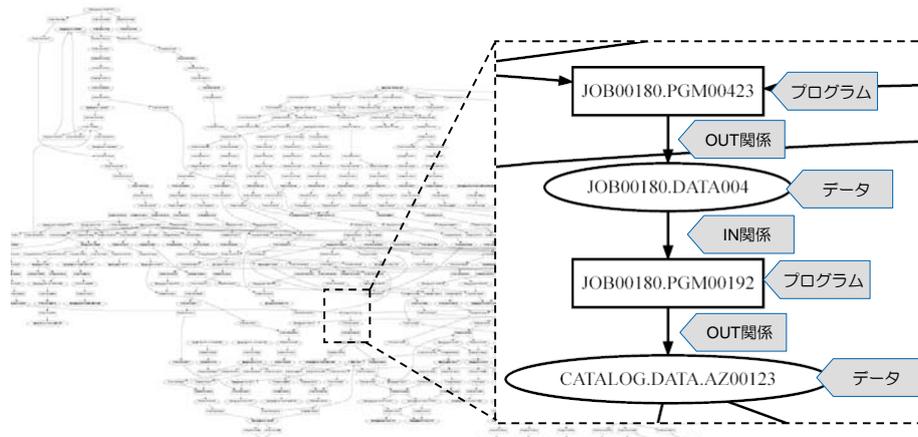
- 弊社が独自に取り組んでいる手法
- 人間の判断をサポートするための見える化に重きを置く
- 2つのポイント
  - 論理名管理
    - ⇒人間が見てわかりやすい名称を付与できる
  - 集約・階層管理
    - ⇒システム要素のグループ化・階層化ができる

2. 見える化の4つの視点とその手法  
 マクロ・非定型分析の具体例



■ システムフロー分析 (仮) Step1/4

- ジョブネット定義、JCL等から物理システムフローを描画

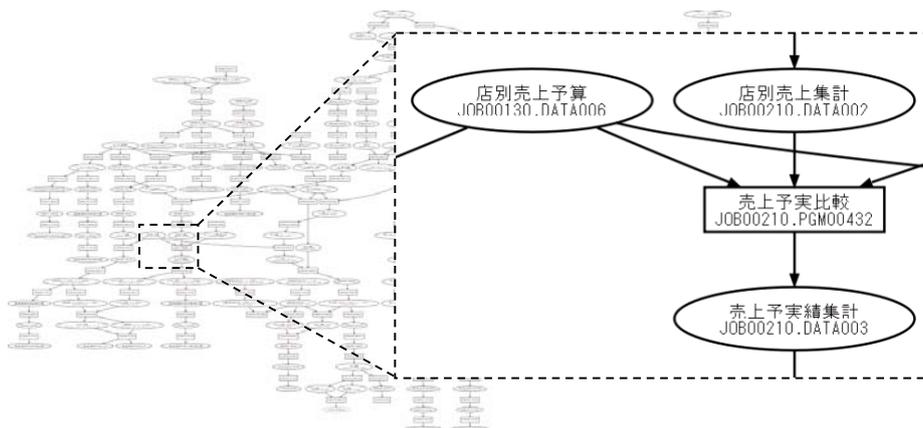


2. 見える化の4つの視点とその手法  
マクロ・非定型分析の具体例



■ システムフロー分析（仮） Step2/4

- プログラム、データに日本語名称付与⇒論理名管理

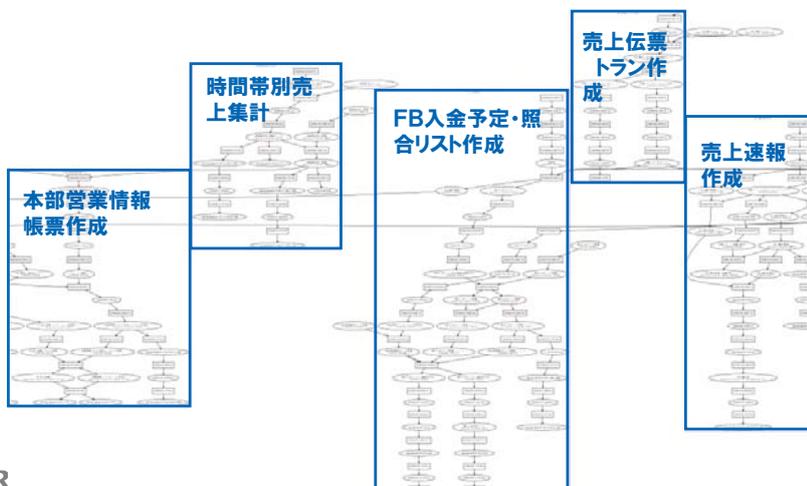


2. 見える化の4つの視点とその手法  
マクロ・非定型分析の具体例



■ システムフロー分析（仮） Step3/4

- 業務機能の単位でグループ化⇒集約・階層管理

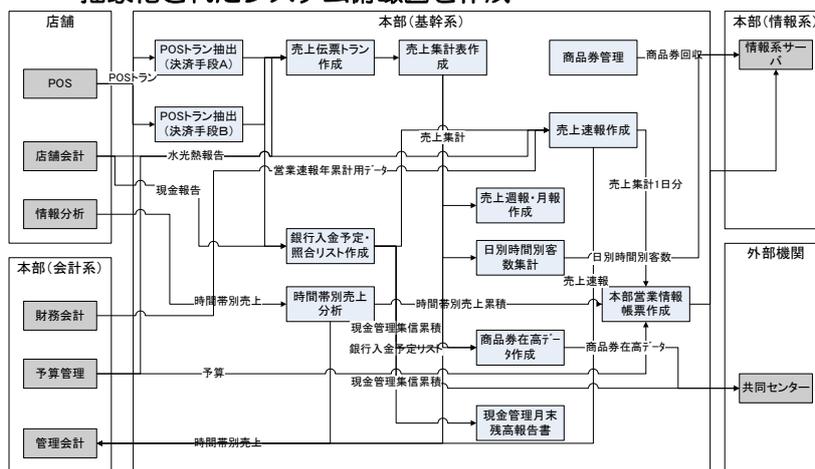


2. 見える化の4つの視点とその手法  
 マクロ・非定型分析の具体例



■ システムフロー分析（仮） Step4/4

● 抽象化されたシステム俯瞰図を作成



目次

1. 見える化の必要性
2. 見える化の4つの視点とその手法
3. 活用事例
4. まとめ

## 目次

---

1. 見える化の必要性
2. 見える化の4つの視点とその手法
3. 活用事例
4. まとめ

## 4. まとめ 見える化の必要性

---

- 重要なのは、問題を解くことではなく定義すること
- 問題定義のための重要なプロセス＝「見える化」
- 今日お話ししたのは、「ソースコードからの見える化」



#### 4. まとめ

ソースコードからの見える化 4つの視点とその手法



NRI Copyright (C) 2009 Nomura Research Institute, Ltd. All rights reserved.

33

# NRI

## 未来創発

Dream up the future.

野村総合研究所  
Nomura Research Institute