「製品開発者のためのＯＳＳライセンス･コンプライアンス」
第2部

OSSコード検出ツール「protexIP/development」のご紹介

2008年8月29日
NEC OSSプラットフォーム開発本部
山本

Contents

1. 背景と概要

2. 特長・機能ご紹介
   - デモンストレーション

3. 製品・サービス体系

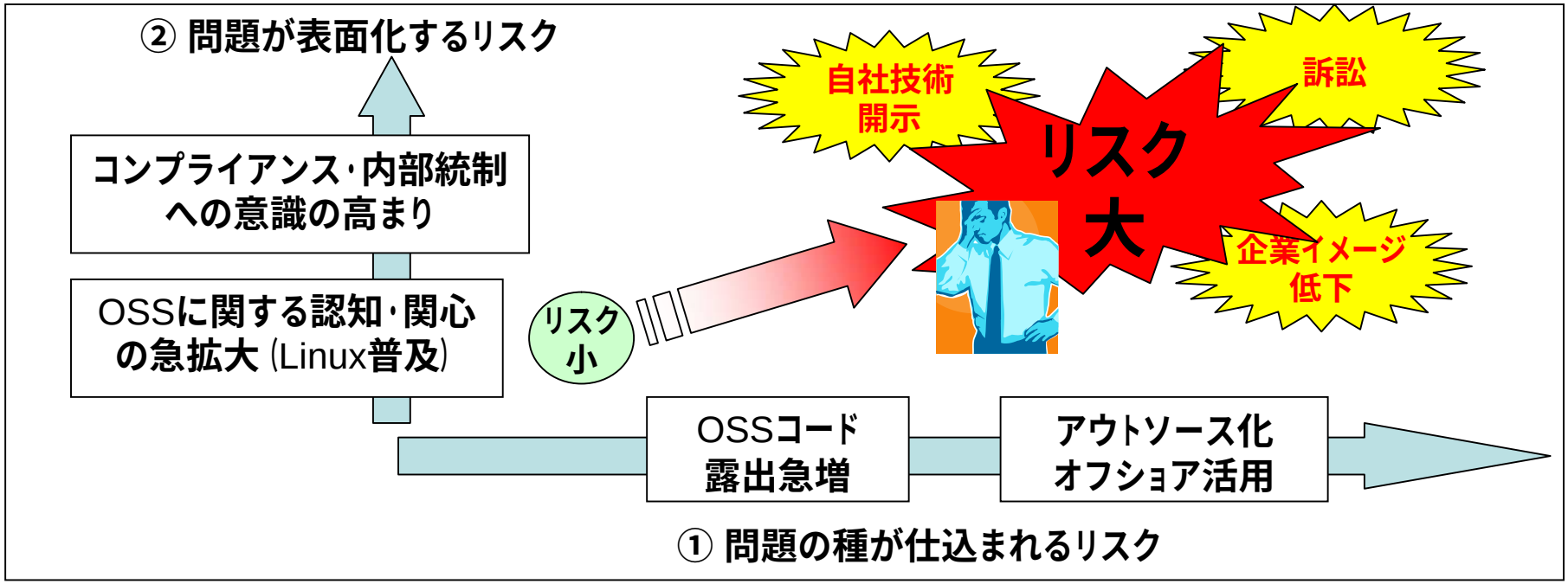
1．背景と概要

コード検査の必要性1(第1部の振り返り)

- OSSは、「単に、自由に使えるもの」ではありません。
- 正しく使うためには、ライセンスを理解し、遵守しなくてはなりません。
- ライセンス違反は、風評被害や訴訟につながりかねません。
- 正しく使うための11のチェックポイントをチェックしましょう。
- (チェックするためには、実際の物件で何を使っているか、それがライセンス的に問題無いかを知る必要があります。)
- 実際の物件で何を使っているか分からない/問題無いことを確認したいときは、「protexIP/development」で検査しましょう。
- 部門内啓発、注意事項のチェック、実際の製品についての具体的な相談の際は、「OSSライセンス・コンサルティングサービス」をご利用ください。

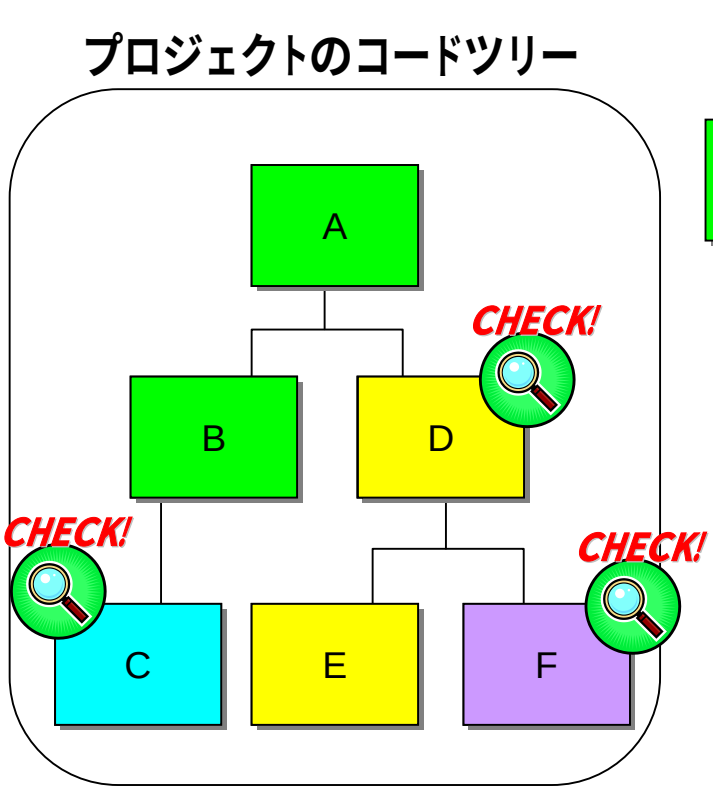
コード検査の必要性2(リスク急増の背景)

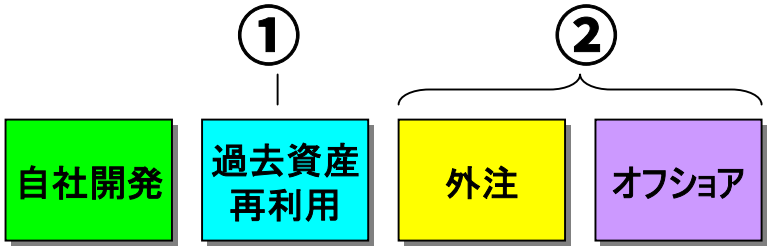
近年、2つの側面からライセンス違反リスクは急増。
① 問題の種が仕込まれるリスク
② 問題が表面化するリスク



コード検査の目的1：何を使っているかを明確化

①過去資産の再利用
②外注・オフショア物件の受入

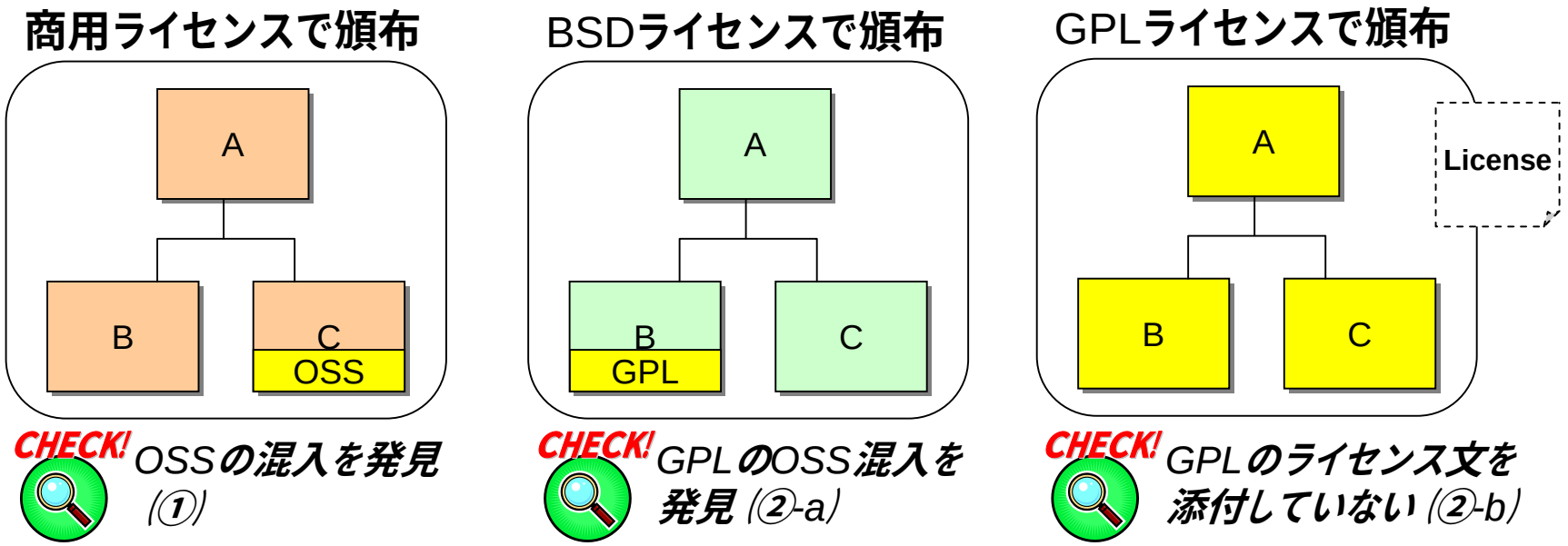




開発時の情報が残っていない物件、
開発コントロールが充分行き届かない物件
は、採用・受入時にコード検査し、
「何を使っているか」を明確にしましょう

コード検査の目的2：ライセンス上の問題確認

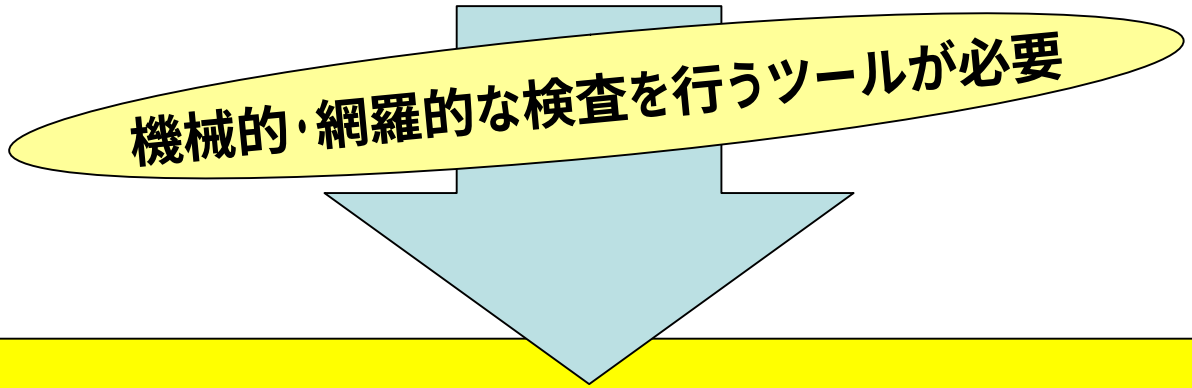
①商用ライセンスで頒布する物件にOSSが混じっていないか
②OSSライセンスで頒布する物件に
　a. 想定外のOSSが混じっていないか
　b. 当該ライセンスの義務事項を遵守しているか



いずれの場合も、ライセンス上の問題確認が必要です

なぜツールが必要か？

- コード検査の目的1、2を達成するには、**実物件の内容を精査**するレベルの検査プロセスが必要。
- しかし、開発規模の急増やオフショア活用などにより従来の**人手によるレビューで問題箇所を発見することは現実的に不可能**。



protexIP/development

2007年1月～ NEC自社導入
2008年1月～ 外販を開始

参考

(開発元) Black Duck Software社

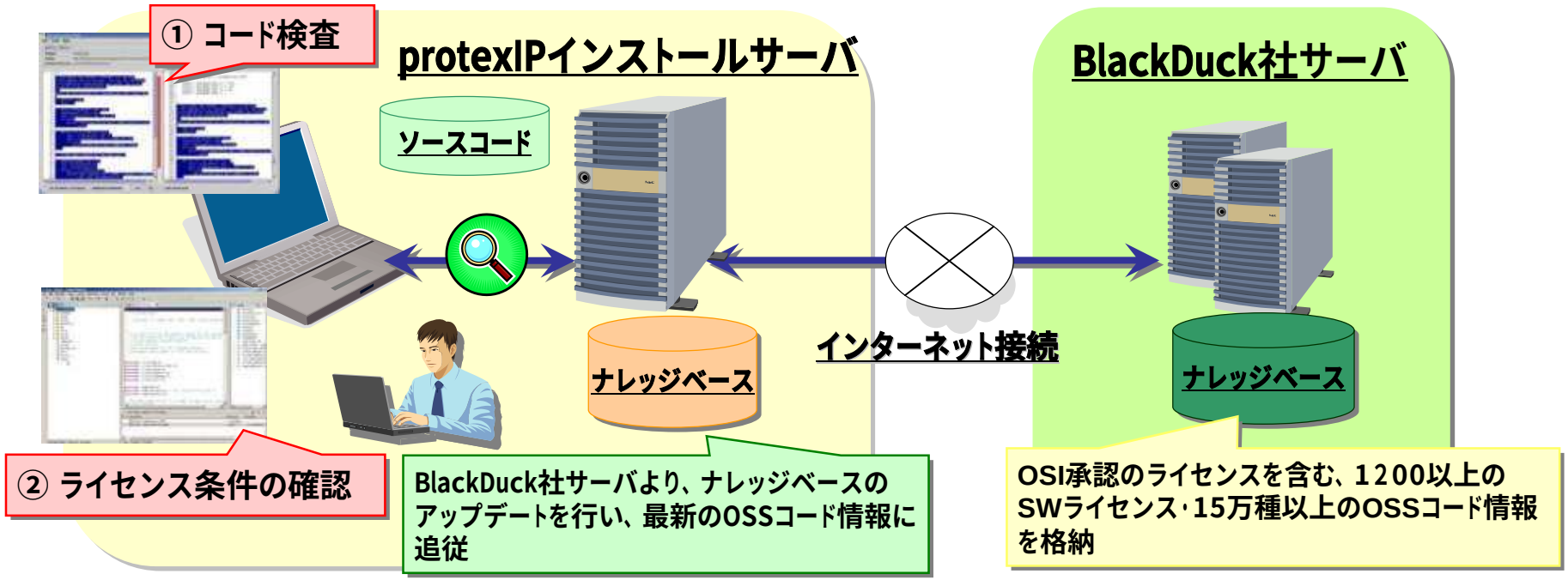
- 本社 マサチューセッツ州ウォルサム
- 西海岸セールス拠点:
  シリコンバレーオフィス
- 2002年設立
  Flagship Ventures,
  General Catalyst Partners 等
  VCのほか、Intel、Red Hat出資
- 従業員数 80名 (2008年3月現在)

blackduck™ "Know Your Code"™

「protexIP/development」機能概要

① **コード検査**
自社物件とOSSコードを比較し、混入または流用の可能性がある箇所を検出。

② **ライセンス確認**
検出されたOSSのライセンス条件を確認。



「protexIP/development」利用イメージ

2. 特長・機能ご紹介

特長1：高度な検出能力

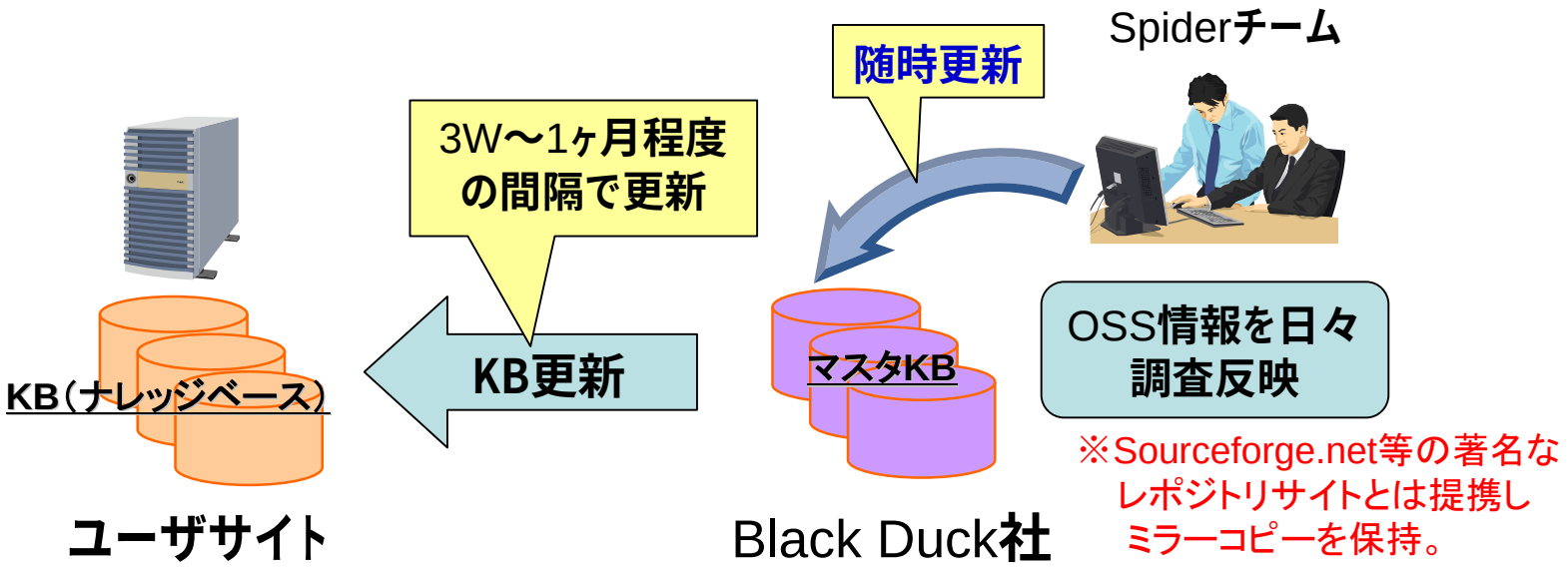
検証用DB（ナレッジベース）には豊富なOSSと
ライセンス情報を格納し、随時最新情報に更新

ソースコードだけでなく、ライブラリ、実行ファイル等の
バイナリファイルも検出

コードの特徴情報（Code Prints）を比較することにより、
部分的な流用や論理構造の一致も検出

豊富なナレッジベース

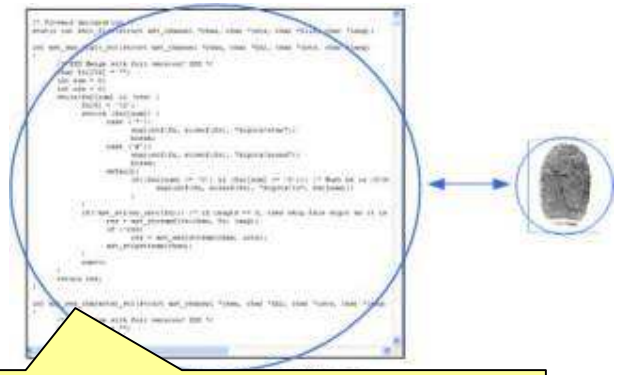
- 150,000超のOSS情報、1,200超のライセンス情報を格納、随時更新
- 各OSS情報は、ソースコードだけでなくライブラリ、実行ファイル、画像等のバイナリも含む。



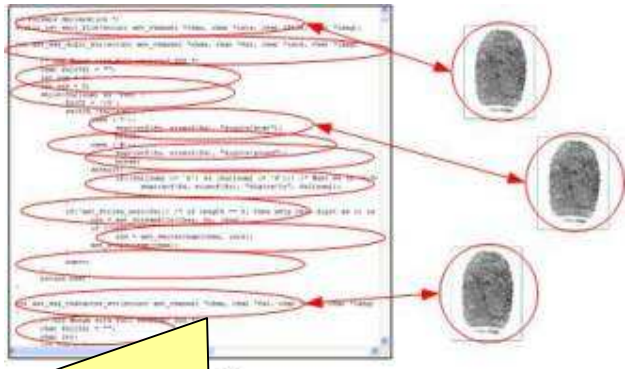
高速・柔軟な照合が可能なCode Prints

- OSSのソースファイル、バイナリファイル
  （ライブラリ、実行ファイル、画像など）の特徴を
  抽出・エンコードしたデータ。

- 効果
  - ① データサイズの大幅な縮小、高速な照合が可能。
  - ② ファイル単位の流用だけでなく、ソースファイルでは部分的な流用や論理構造の一致も検出。
    ※流用時に多少Modify（関数名や変数名の変更など）されていても検出可能。



バイナリ、ソースファイル
“Summary Print”
(md5チェックサムに類似。
原本の同一性を保証した一方向ハッシュ)



ソースファイル
“Snippet Prints”
(コード断片から特徴抽出)

特長2：ライセンス上の問題を明確に確認

ナレッジベースにはそれぞれのライセンスにおける
義務事項を属性として格納

自社物件のライセンスと、検出されたOSSライセンスの
間の競合（同時に満たせない義務事項の存在）を警告

最終的に満たすべき義務事項をチェックリスト化し、
ライセンス上の問題が無い状態を確認してから出荷

ライセンス競合

- 自社物件のライセンスと、検出されたOSSのライセンスとの間に義務事項の矛盾があれば、「競合」となる。
  - 両方のライセンスを同時に遵守することができない状態
  - そのまま出荷するとライセンス違反
  - 物件改修または自社物件のライセンス変更以外に回避手段無し

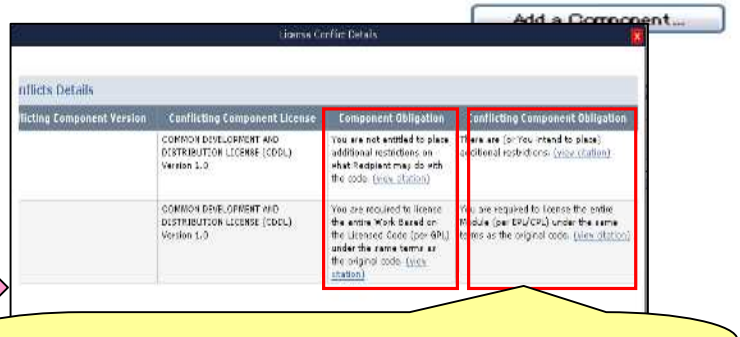
検査結果画面上の表示

Project: cddl

3 Components

Approved	License Conflict	Component	Version	License	Usage	Ship Status	# ID'd	# Depends
N/A	N/A	*cddl*	Unspecified	COMMON DEVELOPMENT	Original Code	Ship	322	0
✔		Grub	Unspecified	GPL 2.0	Component (S	Ship	247	0
✘	⚑	The GNU Ada compiler	4.0.2	GPL 2.0	Snippet	Ship	1	0

ライセンスの競合状態
赤旗: 宣言ライセンスに対してのライセンス競合あり
黄旗: 検出されたOSS間でのライセンス競合あり
旗無し: ライセンス競合なし



ライセンス中のどの条文に違反、競合するかの確認が可能

義務事項チェックリスト

- 競合が解決済みであることが前提
- 対象物件を頒布する際に遵守すべき義務事項をチェック
- 全てチェック済みでないと最終承認状態（出荷OK状態）に移行不可

例：Noticeの添付、ソースコードの公開 等

Obligations:

Fulfilled	Obligation	Category	Description
☐	Runtime Notice Instructions: If the modified program normally reads commands in(...)	Legal	If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License.
☐	Specific instructions for handling attributions: You may copy and distribute verbatim copies of the(...)	Legal	You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice.
☐	You are not entitled to impose a fee related to what Recipient may do with the code.	Legal	
☐	You are not entitled to place additional restrictions on what Recipient may do with the code.	Legal	

全ての義務事項のチェックが済みになっていないと最終承認不可

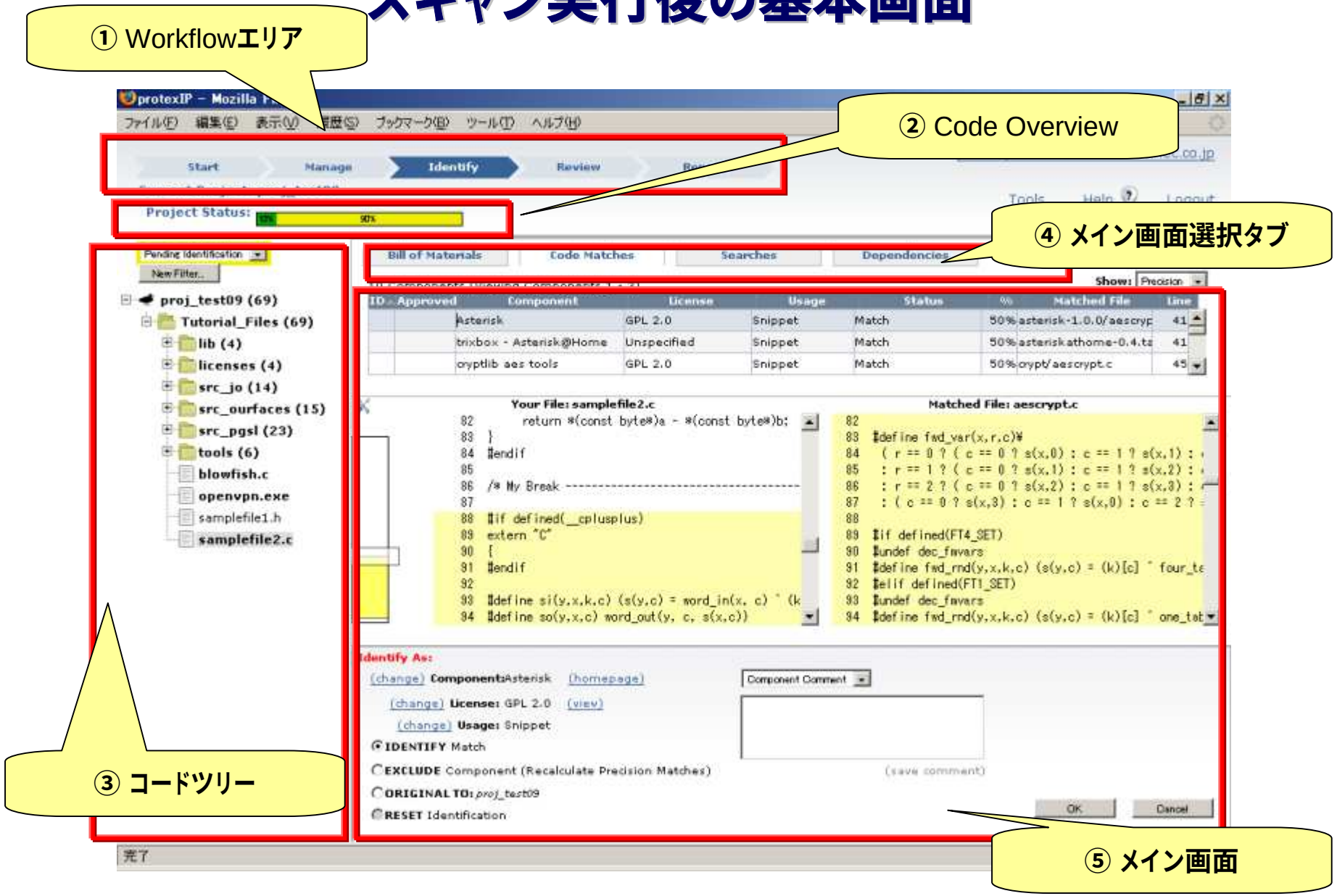
特長3：検査の形骸化を抑止するワークフロー設計

各ユーザの権限をそれぞれの役割単位で指定
（開発者、プロジェクトリーダ、品質保証責任者、弁護士、…）

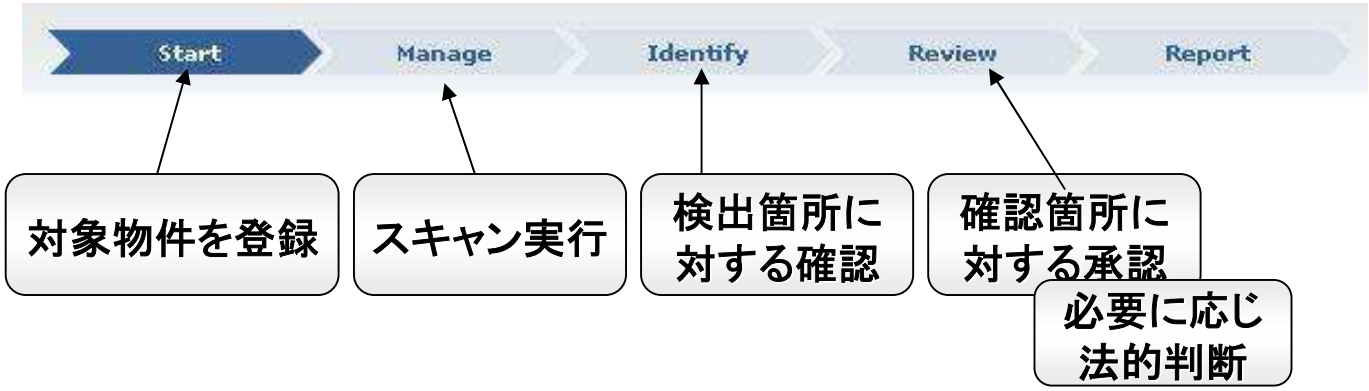
検査プロセスの各フェーズに対し、適切な役割を持ったユーザのみ作業可能

検査プロセスのワークフロー

スキャン実行後の基本画面



① Workflowエリア



② Code Overview

- プロジェクトの状態を表示
  - 検査状況を容易に把握可能

  - 緑（No Issues：問題無し）
  - 黄（Pending Identify：確認待ち）
  - 青（Pending Approval：承認待ち）
  - 赤（Violation：問題=ライセンス競合あり）

その他の機能（1）

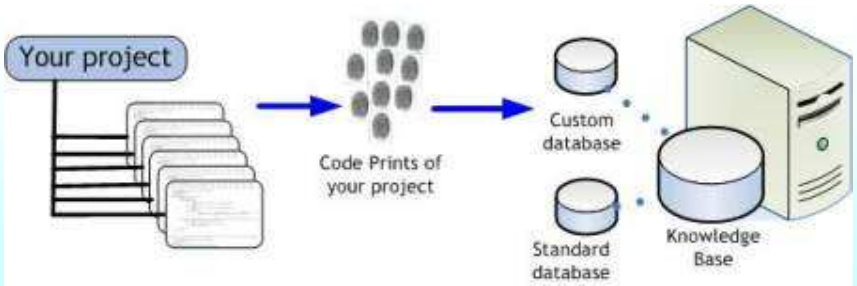
- Report機能
  - スキャン実行後はいつでもプロジェクトの検査状態の情報をファイル出力可能。
  - 対応フォーマット
    - ✓ HTML、Printable HTML
    - ✓ Excel、Word
    - ✓ Open Office – Text、Spreadsheet

開発管理記録、出荷可否判定資料等

- BDSTOOL
  - スキャン機能を呼び出すコマンドラインインタフェースを提供
  - WindowsのコマンドプロンプトまたはLinuxのシェルから起動
  - スクリプトによりスキャン機能の自動実行・スケジューリングが可能

その他の機能（2）

- カスタムDB
  - ユーザ独自のコンポーネントやライセンス定義をDBに追加
  - 独自コードやサードパーティコードの再利用を追跡可能
    例）他社との共同開発物件が契約外の製品で使われていないか検査



- Search、Pattern

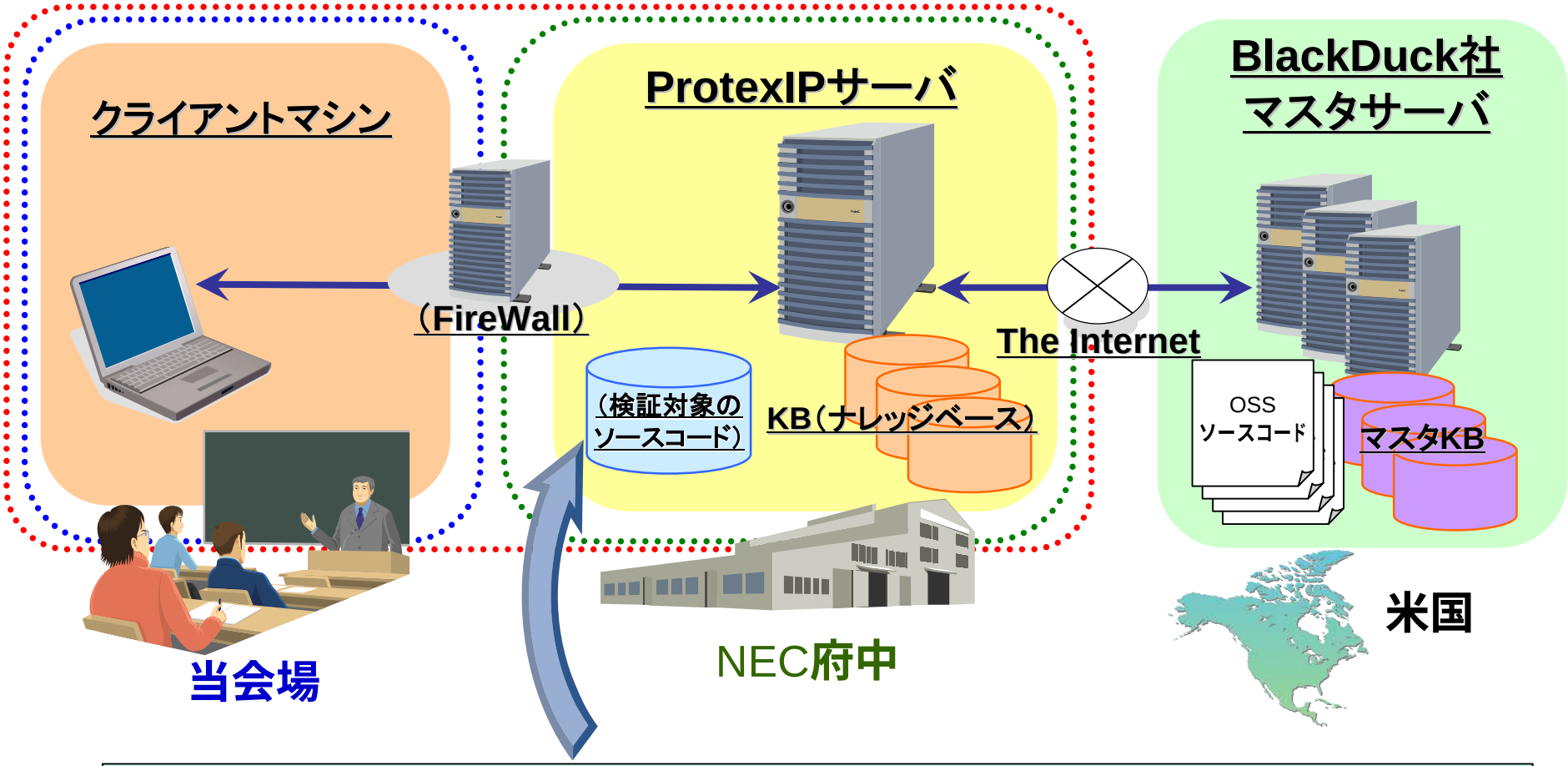
スキャン結果の確認作業における補助情報、手がかりを抽出する手段

  - Search
    - スキャン時に、特定の文字列が含まれていないかを検査することが可能
    - 例）GPL、copyright、自社名 など
  - Pattern
    - スキャン時に、特定のファイル名やフォルダ名を抽出することが可能
    - 例）readme、licence.txt など

デモンストレーション

デモンストレーション

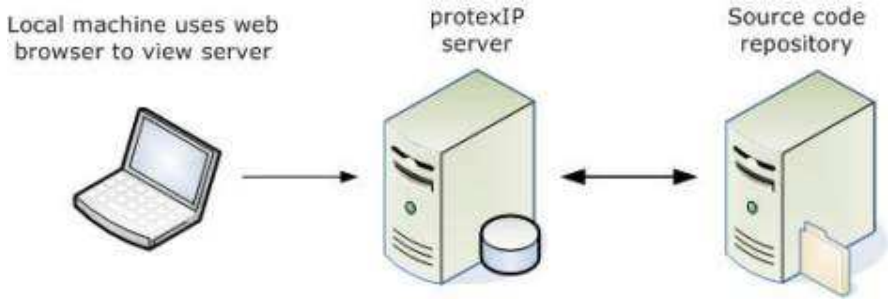
実行環境



※HP社がOSS（GPLv.2）として公開したTru64標準のファイルシステム「Advanced File System (AdvFS)」を、protexIPのナレッジベース反映前の状態で使用することにより、自社開発コードの擬似サンプルとして使用。

運用形態 (例)

(a) サーバ側にソースコードを配置
protexIPサーバ上から、ソースコードにアクセスする場合
– 又は、protexIPサーバ上に直接ソースコードを配置する場合



•パフォーマンス重視
•ソースを広く共有

(b) クライアント側にソースコードを配置
クライアントマシンから、ソースコードにアクセスする場合
– 又は、クライアントマシンに直接ソースコードを配置する場合

•ソースを局所的に管理
•高いセキュリティ
(クライアントにソースを置けばネットワークには流れない)

デモンストレーション

ケース①

自社開発コード（商用ライセンスで頒布）に
OSSコードが混入していないかを検査

デモンストレーション

ケース②

OSSライセンスで頒布する物件に他のライセンスの混入/ライセンス違反が無いかを検査

デモンストレーション

ケース③
自社物件をカスタムDBに登録し、他の開発で再利用されていないかを検査

効果的に使うためのポイント

protexIPによる検査を実行する前に、以下を実施しておくことが望ましい。（全てが難しい場合は、1、2、3の順でできるところまでやっておきましょう）

1. **対象物件のライセンス（宣言ライセンス）を明確にしましょう。**
   - どのようなライセンスで頒布するつもりなのかがはっきりしていないと、ライセンス違反を判断できません。
   - 複数のライセンスで頒布される物件は、ライセンスごとに検査単位を分けましょう。
2. **流用しているOSSがはっきりしている場合は、コードツリーのどの部分が該当するかを予め整理しておきましょう。**
   - メジャーなOSSを検査すると、検出件数が膨大になることがあります。予め流用OSSを明確にしておくことで、確認（Identify）作業が容易になります。
3. **OSSに手を加えず再頒布するだけの部分と、手を加えた部分を分離しておきましょう。**
   - 再頒布するだけの部分は、他の部分との関係に注意の上で当該ライセンスの遵守を確認すればよく、わざわざ検査する必要はありません。

3．製品･サービス体系

製品・サービス体系

➢ 製品ライセンス、サポートサービス、付加サービスで構成。

③ NEC付加サービス

・導入支援サービス

インストール、構築作業、基本トレーニング

・解析支援サービス

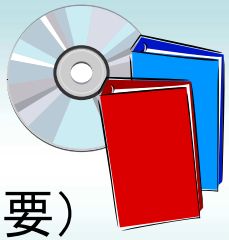
コード検査結果の解説、対処方法判断支援

② NECサポートサービス

・TEL、E-Mail、FAXによる製品使用方法のQ&A、障害調査

・Web、mailによる情報提供

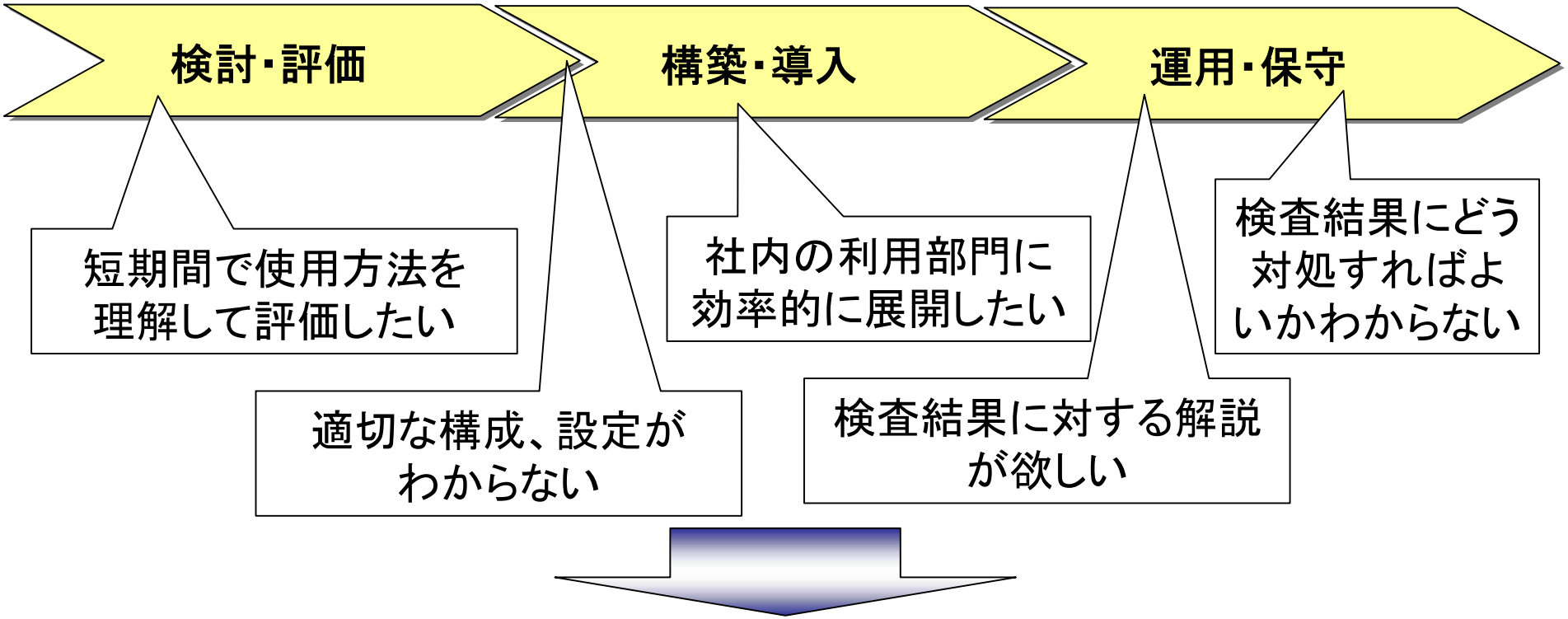
① 製品ライセンス（年間Subscription）



・ 製品使用（サーバ1台あたり1ライセンスが必要）

・ Rev.up権、ナレッジベース更新権

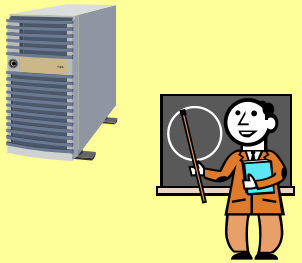
付加サービス



protexIPの製品知識に加え、OSSとライセンスのノウハウを持った
NECの技術者が支援します
導入支援サービス ・ 解析支援サービス

導入支援サービス

- 検討・評価～構築・導入フェーズにおける支援
  - ✓お客様の導入環境にあわせたprotexIP設定
  - ✓インストール、設定作業
  - ✓管理者/利用者向けのレクチャー
- 費用：個別見積り
- 内容：個別要件に応じてカスタマイズ
  - 例）ProtexIPインストール作業、環境設定作業
    - 管理者/利用者向けのレクチャー
    - ドキュメント作成（インストール作業報告書、レクチャテキスト等）
- 効果：
  - ✓短期間での効率的な評価検証
  - ✓導入コストの削減
  - ✓社内の利用部門への効率的な横展開



本サービスの一部（所定の範囲）は製品の初年度Subscriptionに含む。
- インストール ～ 基本設定（正常起動の確認まで）
- 汎用チュートリアルを用いた基本操作レクチャ

解析支援サービス

- 運用・保守フェーズにおけるprotexIPによる検査結果解析を支援
  - ✓検査結果（レポート出力）に対する解説
  - ✓検査結果に対する詳細分析・対応方法判断支援
    - *※対応方法の判断そのものは本サービスの範囲外（リスク解説まで）*

- 費用：個別見積り
- 内容：個別要件に応じてカスタマイズ
  - 例）ProtexIPのレポート出力に対する解説ドキュメント作成
    - 検出箇所に対する対応方法判断のためのリスク解説、詳細分析レポート作成
    - オンサイト対応（開発者ヒアリング等必要に応じ実施）
- 効果：
  - ✓検査結果に対する短期間でのリスク把握、対処の意思決定
  - ✓リスク認識に基づく開発プロセスへのフィードバック
    - （OSS混入・ライセンス違反を未然に防止する仕組み作り）
  - ✓解析ノウハウ蓄積、横展開

本サービスの一部（所定の範囲）は製品の初年度Subscriptionに含む。

－以下のいずれかを選択。

a) お客様の実コードでの検証・レポート作成(コードサイズ20MB以下)

b) サンプルコードによる検証実演と解析作業解説

価格体系

- 利用規模に応じた価格体系
  - 管理対象のコードサイズ　・利用ユーザ数
- 年間Subscription方式

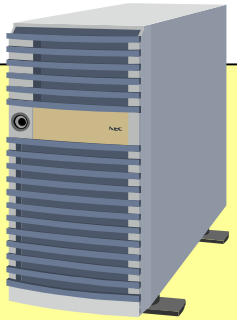
期間限定キャンペーン実施中
(2008年9月まで)
※詳しくはお問合せください。

[希望小売価格：万円/年]

※所定範囲のNEC付加サービスをバンドル。
（）内はNEC付加サービス無しの価格（原則として2年目以降の更新用）。

コードサイズ / ユーザ	50MB	100MB	200MB	500MB	1,000 MB	2,000 MB	5,000 MB	10,000 MB	Unlimited
5	570 (270)	770 (470)	1,000 (700)	1,650 (1,350)	2,650 (2,350)	3,400 (3,100)	4,250 (3,950)	4,950 (4,650)	
15	790 (490)	990 (690)	1,220 (920)	1,870 (1,570)	3,040 (2,740)	3,790 (3,490)	4,640 (4,340)	5,340 (5,040)	
50					3,430 (3,130)	4,180 (3,880)	5,030 (4,730)	5,730 (5,430)	
100						4,560 (4,260)	5,410 (5,110)	6,110 (5,810)	
Unlimited							5,800 (5,500)	6,500 (6,200)	6,800 (6,500)

動作環境



- **動作要件**

  [必須]・CPU ⇒ Pentium4 2GHz ・メモリ ⇒ 4GB
  ・HDD ⇒ 500GB(非RAID構成)

  [推奨]・CPU ⇒ Xeon ・メモリ ⇒ 8GB
  ・HDD ⇒ OS領域(ミラー構成) 80GB、DB領域(RAID構成) 1000GB

  サポートOS：RHEL4、RHEL5（x86-64推奨）

  ※NEC Express5800「Linuxサービスセット」のご利用を推奨しております。

- Express5800シリーズ：http://www.express.nec.co.jp/pcserver/
- Linuxサービスセット　：http://www.nec.co.jp/linux/linux-os/

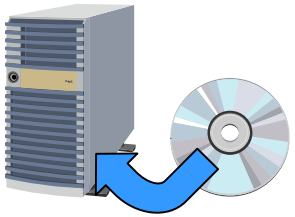
ご試用のご案内

無料

① 評価版ライセンス、② お試しレポート の2種類をご用意。

①評価版ライセンス

- 最長30日間、25MBまで、製品をご試用いただけます。
- 正規製品との機能差分はありません。
- 別途評価用サーバ、OSが必要です。



詳細な機能をじっくりとお試しになりたい方はこちら

②お試しレポート

- お客様のソースコード（最大25MB）をお預かりし、スキャン結果をお返しします。
- 正規製品の出力との差分はありません。
- 機材のご準備は不要です。

どのような結果が得られるかをお手軽に確認したい方はこちら

詳細はお手元のご紹介資料をご覧ください。

製品情報・お問い合わせ先

- 製品情報
  - http://www.nec.co.jp/oss/protexip/
- お問い合わせ先

  E-Mail:
  - protexip-info@osspf.jp.nec.com
    （NEC OSSプラットフォーム開発本部 protexIP担当）

Empowered by Innovation

NEC