

# GNATSとGnatswebの 概要とカスタマイズ方法

- Web & メールベースのBug Tracking System

EASEプロジェクト  
NTTソフトウェア(株) 岩村 聡

1

## Bug Tracking Systemとは？

- ユーザや開発者から寄せられた問題報告(バグ情報)を管理するためのシステム
  - 各問題が処理されるまでの対応状況や履歴を管理
    - 報告された問題には全て通し番号などで、問題毎に対応状況を把握することが可能
    - 忘却されたり、混同されることを阻止できる
  - 情報共有
    - 電子メールやWeb、ファイル共有などの手段で共有する
  - GNATS以外の代表的なBug Tracking System
    - Bugzilla
    - 影舞
    - Workbench
    - etc...

2

## GNATSとは？

- GNUプロジェクト関連のBug Tracking System
  - オープンソース
  - UNIX系OSで動作
  - メールを使った情報共有
  - (RDBSなどを用いずに)プレーンテキストのみで問題管理
  - 他のツールを必要としない
  - ユーザインタフェース向上のための環境が用意されている(Gnatswebなど)

3

## Gnatswebとは？

- GNATSのヒューマンインタフェースの一つで、Perlで作成されたCGIで動作する
- 設定などは特に不要で全てGNATSの設定ファイルで動作する
- 動作に必要なもの
  - CGIが動作するHTTPサーバ(Apacheなど)
  - Perl(version 5)
  - PerlのCPANモジュール(CGI, MIME::Base64)
    - 日本語対応( )には、Jcode
    - パスワードのCrypt化には、Crypt::PasswdMD5

日本語対応は、一部CGIの修正が必要

4

## GNATSのメリット

- メリット
  - 項目のカスタマイズが容易に出来る
  - バグ管理だけでなく、ドキュメントレビュー記録や課題管理・個人のTODO管理にも利用可能
  - メールインタフェース、Emacsインタフェース、Gnatswebを使うことによりWebインタフェースで問題管理が出来る(インタフェースが多様)
  - ファイルのみで管理するためバックアップやサーバの移行が容易
  - ユーザ単位でのアクセス制限が可能

5

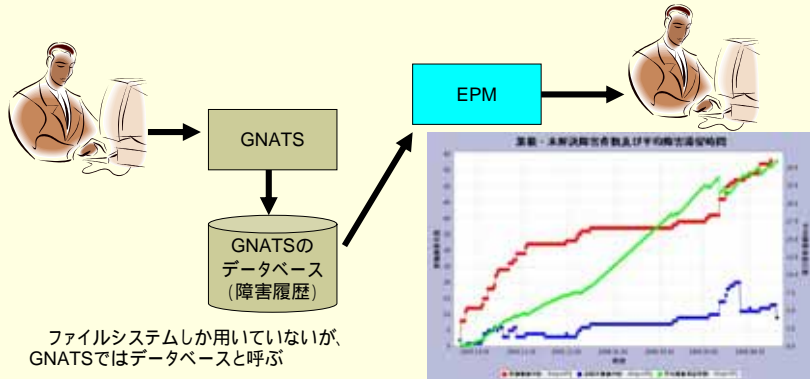
## GNATSのデメリット

- デメリット
  - 日本でのユーザが少ないため日本語での情報が少ない
  - 同一サーバ上で管理しているプロジェクトの部署単位などで分類できない(Gnatswebなどを改造すれば可能??)

6

# GNATSとEPMの関係は？

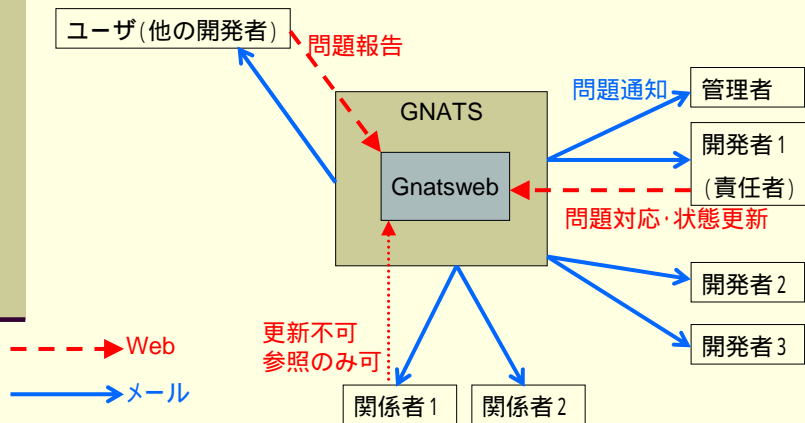
- GNATSがサーバに格納しているデータをEPMが参照し、グラフ化などを行う



7

# 基本機能

(登録 通知 更新 通知 ... 解決 通知)



8

## カスタマイズ項目 (ユーザ関連)

項目	概要	備考
認証ユーザ	Gnatswebなどからログインする時の認証に使用されるユーザ ユーザ毎に「更新・参照可能」「参照のみ可能」などの設定が出来る	「データベース毎」と「サーバ共通」を選択できる
投稿者	問題報告を投稿する側のユーザ 投稿者毎に投稿された問題報告の時間的扱いや通知先を変更できる	
責任者	問題報告を受ける側のユーザ 責任者毎に通知するメールアドレスを設定する	

「認証ユーザ」と「投稿者」と「責任者」の設定が独立しているためユーザ管理が複雑。

ユーザ管理は複雑だが、カスタマイズ次第では簡易化が可能だと思う。またファイルのみの編集で管理できるため管理プログラムの作成も難しくはないと思う。

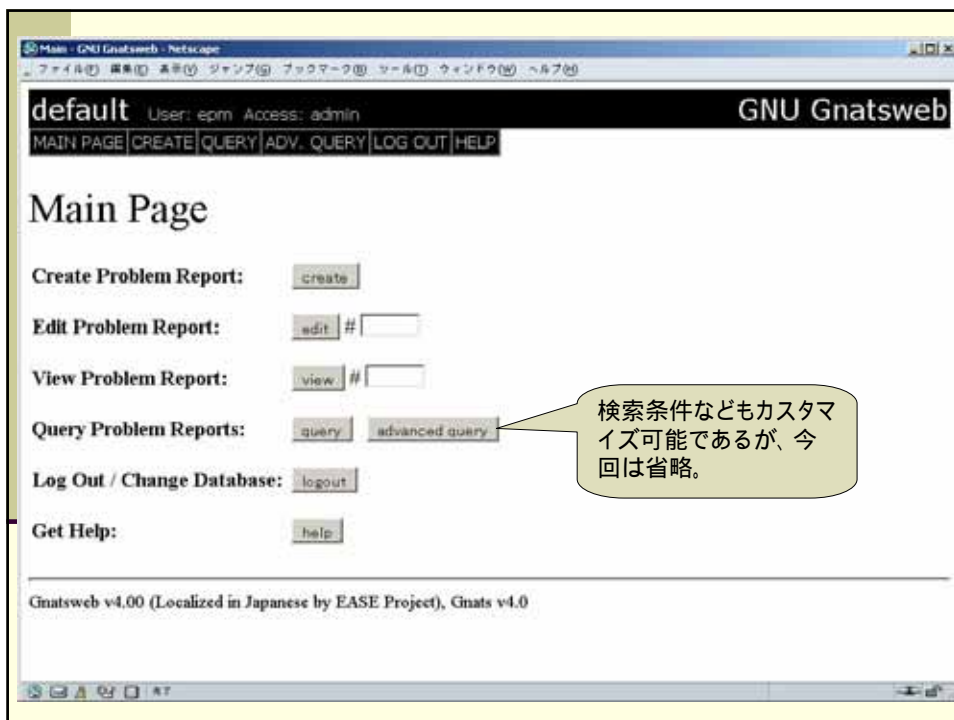
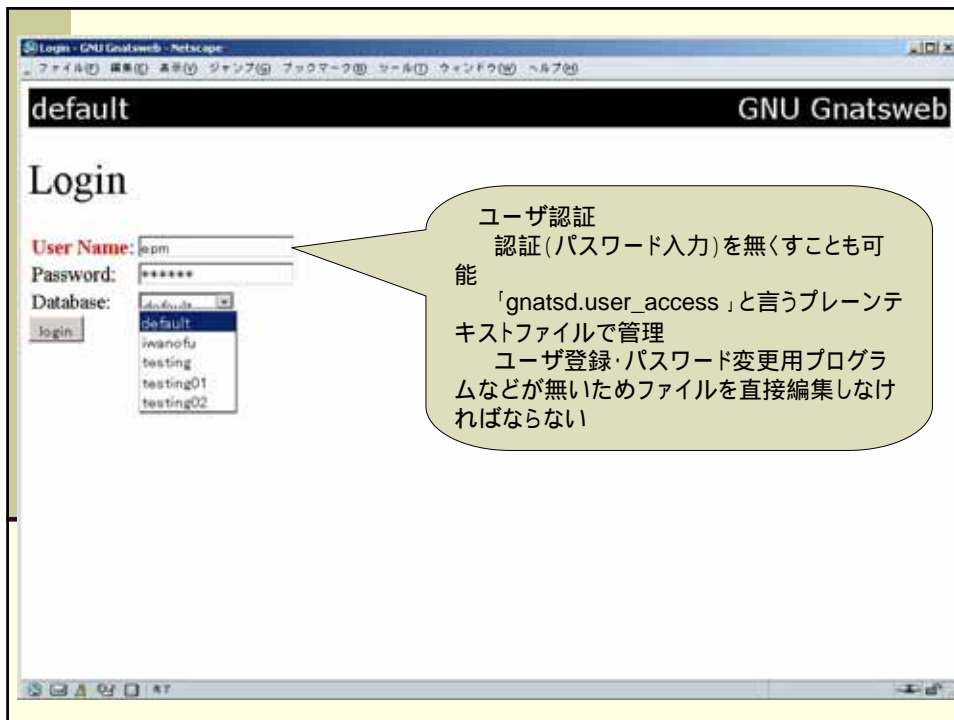
9

## カスタマイズ項目 (fieldの例)

項目	概要	備考
カテゴリ	問題の分類 (サブシステムなど) を選択	
概要	問題の概要を自由入力 (1行)	通知メールの Subject に使用
状態	問題の状態 (対応状況) を選択	
種類	問題の種類 (バグや勘違いなど) を選択	
重要度	問題の重要度を選択	
優先度	問題の優先度を選択	
最終更新時間	問題について何らかの情報が更新されたときの時間を記録	手入力不要
詳細	問題の詳細を自由入力 (複数行可)	
対処方法	問題の対処方法を自由入力 (複数行可)	

網掛け部分について、後ほど紹介

10



**状態 (State)**  
 問題の状態 (open, analyzed, suspended, feedback, closed) がデフォルト  
 open (未対応) と closed (対応終了) には、特別な意味がある  
 マニュアルでは、半角英数字のみOKと書いてあるが、実験レベルでの運用では日本語でも問題なく動いた

Number	Notify-List	Category	Synopsis	Confidential	Severity	Priority	Responsible	State	Class	Submitted
<a href="#">2 edit</a>		pending	test2	no	critical	high	<a href="#">gnats-admin</a>	analyzed	sw-bug	te
<a href="#">3 edit</a>		pending	test	no	critical	high	<a href="#">gnats-admin</a>	open	sw-bug	te
<a href="#">4 edit</a>		pending	test11	no	critical	high	<a href="#">gnats-admin</a>	open	sw-bug	te
<a href="#">6 edit</a>		pending	test32	no	critical	high	<a href="#">gnats-admin</a>	open	sw-bug	te
<a href="#">9 edit</a>		pending	aaaaaaaaa	no	critical	high	<a href="#">gnats-admin</a>	open	sw-bug	te
<a href="#">20 edit</a>		doc	誤字脱字	no	non-critical	low	<a href="#">gnats-admin</a>	open	誤字脱字	iw

**責任者 (Responsible)**  
 報告された問題の責任者を1名 (1ID) 指定

**種類 (Class)**  
 問題 (バグ) の種類 (ソフトウェアバグ・ドキュメントバグ・変更リクエストなど) を設定  
 ドキュメントレビューを管理するなら、「仕様誤解・曖昧な表現・用語不統一・誤字脱字」などで設定  
 マニュアルでは、半角英数字のみOKと書いてあるが、実験レベルでの運用では日本語でも問題なく動いた

## カスタマイズ:カテゴリ (categories)

- カテゴリを用いて問題の対策方法を分類
  - カテゴリ設定なしで報告がなされた場合は, pendingとして管理者が分類
- カテゴリにおける設定内容
  - カテゴリ名
  - 「責任者 (responsible)」 [必ず1名 (1メールアドレス)]
  - 「通知者 (notify)」 [複数登録可能]

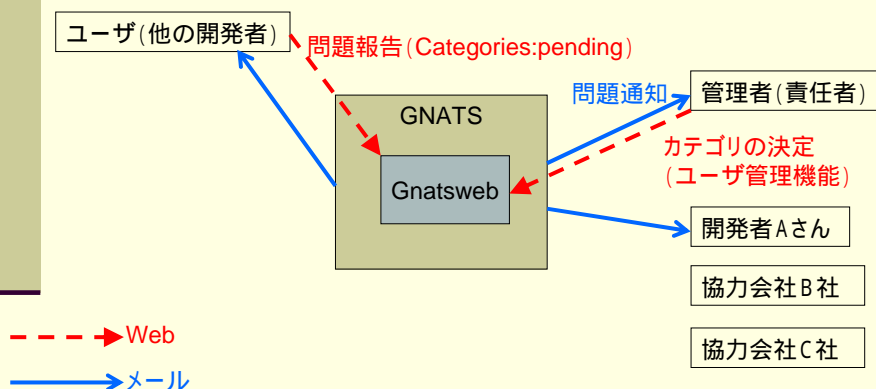
### カテゴリの設定例

カテゴリ	責任者	通知者	備考
pending (カテゴリ未確定)	管理者	なし	pendingは必須 ( デフォルトの設定)
システム共通機能	管理者	開発者Aさん 協力会社B社 協力会社C社	設定例
ユーザ管理機能	開発者Aさん	管理者	設定例
コンテンツ管理機能	協力会社B社	開発者Aさん	設定例
Webアクセス制御 機能	協力会社C社	開発者Aさん 協力会社B社	設定例

15

## カテゴリの利用例 (1)

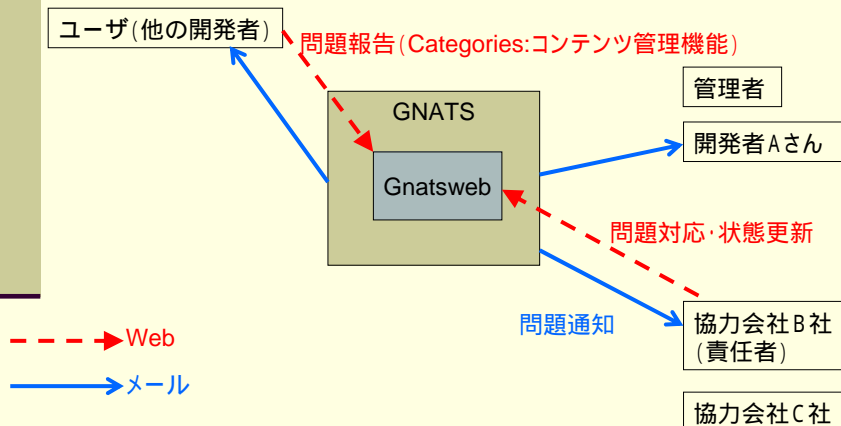
### カテゴリを決定せず報告した場合



16

## カテゴリの利用例(2)

### カテゴリを決定して報告した場合(コンテンツ管理機能の場合)



17

## カテゴリ関連のメモ

- カテゴリ毎にユーザの更新・参照権限を変えることができない
- カテゴリに関係なく責任者 (Responsible) を変えることができる (できてしまう)
  - 問題報告の初期登録時に、初期責任者をGNATSは自動で決めてくれる (だけ)
- 途中でカテゴリを変えると通知先も変わる

18

## カスタマイズ: 投稿者 (submitters)

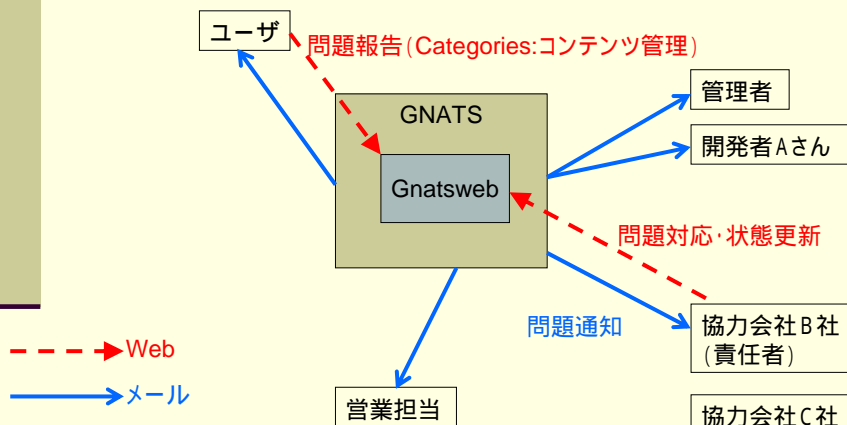
- 投稿者によって異なる対処方法を設定できる
  - 問題報告があった際にその報告先や、通知先を設定できる
- 投稿者ID毎に以下の情報を設定できる
  - 「主通知者 (contact)」【主通知者は必ず1名 (1メールアドレス)】
  - 「副通知者 (notify-other)」
  - 「レスポンスタイム」( 後で説明)

### 投稿者の設定例

投稿者ID	主通知者	副通知者	備考
unknown	管理者	-	unknownは必須ではないがデフォルト Unknownでは投稿不可
Developers	開発者	管理者	設定例
Users	営業担当	管理者・開発者	設定例
ABC_Company	ABC社開発担当	ABC社管理職	設定例

19

## 投稿者 (Users投稿)



投稿者がユーザの場合、カテゴリに関係なく「管理者」と「営業担当」へメール通知が行われる、と言う例

20

## 投稿者関連のメモ

- 「レスポンスタイム」と「ビジネスタイム」
  - 「レスポンスタイム」とは、投稿者が問題報告後、問題の状態 (States) が分析中などに変わらない時に管理者宛にメールを送信するまでの時間  
(ex: ユーザからの問題投稿の場合、顧客満足度をあげるため3時間以内に状態変更をするという運用ポリシー)
  - 「ビジネスタイム」とは、経過時間の対象とする時間
    - 曜日 (ex: 月曜から金曜) と時間 (ex: 8時から17時) で設定  
(ex: 営業時間外 (真夜中など) に投稿された場合などには営業開始時間の8時に投稿されたとして管理するという運用ポリシー)

21

## カスタマイズ: 他のフィールド (field)

- 入力項目 (フィールド) は、今まで説明した項目以外でも以下のような項目を自由に追加・設定できる
  - 「1行」または「複数行」のテキスト
  - 複数の選択項目から「1つ選択」「複数選択」
  - 日時
  - 数値
- 登録・変更時に特定の項目が空白 (未選択) の場合エラーとすることも可能
- 項目の変更可能・不可能が選択可能
- 項目の変更時にメール通知するかしないかが選択可能
- メールで通知するときのメール本文のフォーマットも変更可能

22

## インストールやカスタマイズ方法

- 以下のURLでインストール方法や具体的なカスタマイズ方法を公開しています

<http://www.empirical.jp/~iwanofu/tool/>

23