Managing the Ever-Growing To Do List

Rémy Evard

The College of Computer Science
Northeastern University

The Problem...

- Requests from users
- Long term projects
- Demands from your boss
- Systems needs
- Constant interruptions

Eventually...

- Important things go undone.
- The overhead in keeping up is too high.

Solution: A Mailing List

- **A mailing list isn't:**
  - **a login**
  - **a lot of lists**
  - **<u>me</u>**
- **A mailing list is:**
  - **a source of shared knowledge**
  - **a single place to keep things**

Not.

- Collisions
- Everyone has to read everything
- Everyone has to keep track of issues
  - Holes
  - Overhead

Problem Tracking

- Database of problems
- Ownership, priority, age, due date, history, status...
- A tool that:
  - keeps track of open and closed requests
  - allows you to prioritize issues

Req

- Our solution, for our environment:
  - email based
  - command line and X interface
  - minimally modifiable via email

Req Details

- Data stored
  - Number
  - Owner
  - Requester
  - Priority
  - Age
  - Response time
  - Status
  - Subject
  - Mail and comment history

Other Features

- FAQs
- User access to their own requests
- Statistics

How We Use It

- Hotseat
- Personal and Group to do lists
- Logging events
- Still changing

Problems

- High level problems that don’t fit into a request.
- Individual, private lists.
- Lack of time to add new features.

A Whole Solution

- A way for users to turn in requests
- Tracking user requests with minimal overhead
- Prioritizing requests
- Establishing procedures that work

Availability

- **req:**
  - **ftp.ccs.neu.edu:/pub/sysadmin/req*, tkreq***
- **status**
  - **near the first patch release**
  - **probably a month from the next major release**
  - **minimally supported**

- **paper:**
  - **http://www.ccs.neu.edu/home/remy/documents/req.html**
  - **ftp.ccs.neu.edu:/pub/people/remy/Req/req.ps**