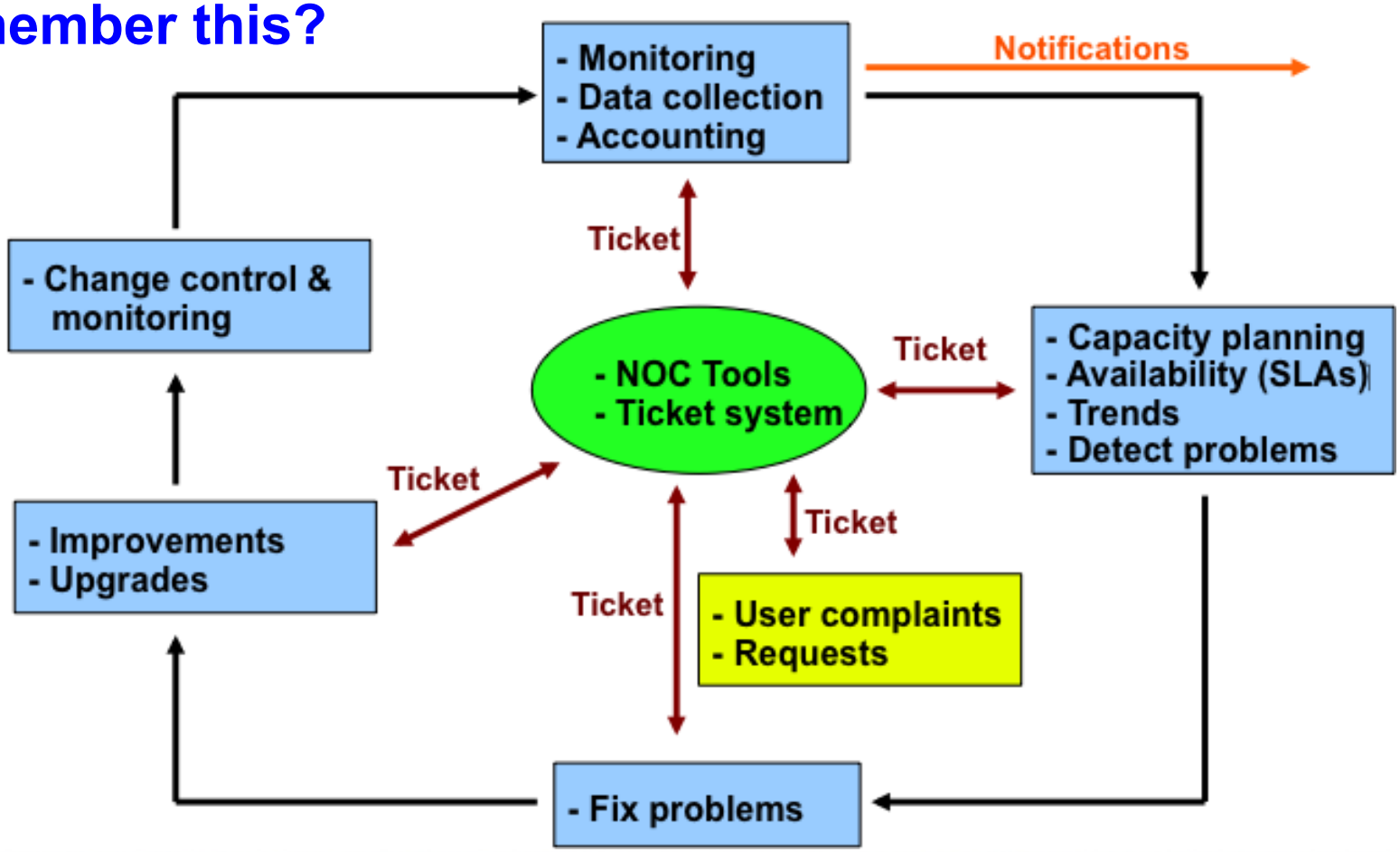
Ticketing Systems and Documentation

APRICOT 2010

February 26 – Kuala Lumpur, Malaysia

Why Ticketing Systems?

Remember this?



Ticketing Systems

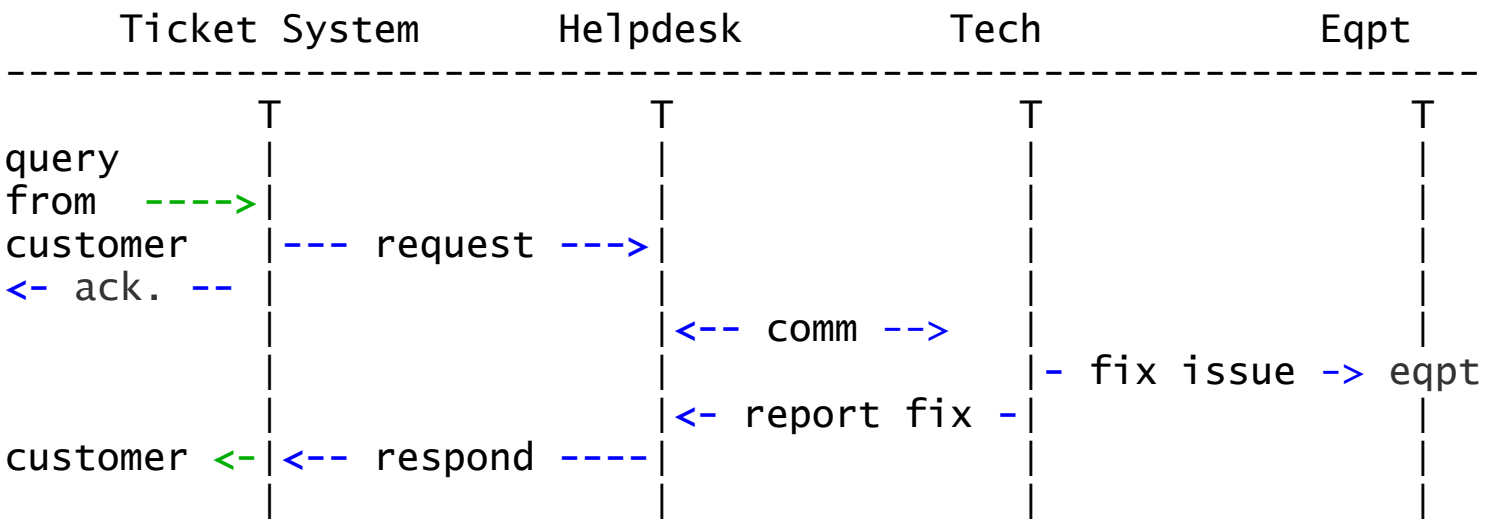
- Why are they important?
  - Track all events, failures and issues
- Focal point for help desk communication
- Use it to track all communications
  - Both internal and external
- Events originating from the outside:
  - customer complaints
- Events originating from the inside:
  - System outages (direct or indirect)
  - Planned maintenance, upgrades, etc.

Ticketing Systems

- Use ticket system to follow each case, including internal communication between technicians
- Each case is assigned a case number
- Each case goes through a similar life cycle:
  - New
  - Open
  - ...
  - Resolved
  - Closed

Ticketing Systems cont.

Help Request with Tickets



Request Tracker & Trac

RT

- Heavily used worldwide.
- Can be customized to your location.
- Somewhat difficult to install and configure.
- Handles large-scale operations.

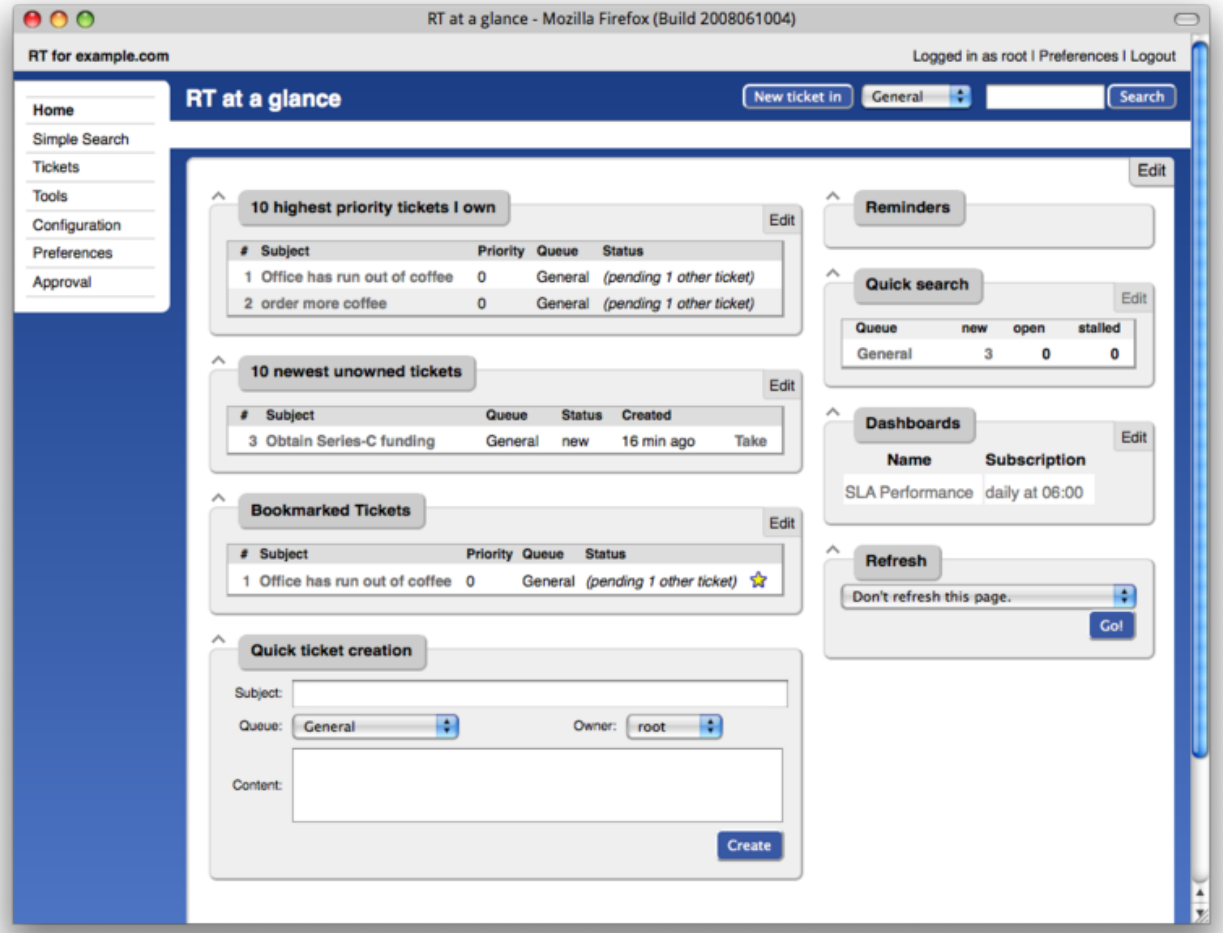
trac

- A hybrid system that includes a wiki and project management features.
- Ticketing system not as robust as rt, but works well for web-only ticket interface.
- Often used for ”trac”king group projects.
- What we're using for our class web pages:
  http://nsrc.org/workshops/2010/apricot/

RT: Request Tracker

http://bestpractical.com/rt/

What's it Look Like?



Topics

- What is a ticket management system
  - Necessities and advantages
  - Common functionalities
- Practice with RT (Request Tracker)
  - Global configuration
  - Create users
  - Create queues
  - Assign actions to the queues
  - Create message filters

Ticket Management Systems

- Why do we use the term “ticket”?
- In order to resolve a problem...
  - Who wants what?
  - Who's going to work on this?
  - When did they ask, when was it done?
  - How much time did it take (billing, hours)?
  - What's left to do?
  - Everything is summarized and presented in a simple and intuitive manner.

Applications

- User support
- Security problem management
- Issue Tracking / Incident Management

Essential Functionality

- Several interfaces
  - Web, CLI, e-mail, etc.
- Multiuser
  - At different levels: admin, general user, gues
- Authentication and authorization
- Event history
- Handles dependencies
- Notifications

Components

- Register an event (i.e., ticket creation)
- Assign an owner
- Assign interested parties
- Maintain change history
- Inform interested parties of each change
- Initiative activities based on status or priority

Example: Why Used at the UO?

- Lots of email traffic requesting help, services, etc.
- Archived as text without classification
- Very difficult to find current status or problem history.
- Sometimes problems were forgotten or never resolved.

RT: Advantages

- Open source and free
- Heavily used and tested
- Very active development
- Quite flexibile
- Web interface and via email

RT: Disadvantages

- A bit tricky to install the first time...
- It's powerful, so you'll need to spend some time learning how it works.
  - Most distributions have packages that make installation a bit easier:
    - Gentoo, Debian, FreeBSD, etc.

Problem Classification: Queues

- RT allows you to create queues so that problems are classified by type:
  - **Services:** DNS, IP addresses, Radius, LDAP
  - **Connectivity:** Communications infrastructure problems
  - **Security:** Attacks, scans, abuse, etc.
  - **Systems:** Email accounts, passwords, etc
  - General help

Web Server Configuration

- **Two Options**
  - Virtualhost
    http://rt.host.fqdn
  - Subdirectory
    http://host.fqdn/rt
- **Root user ('*root*')**
  - Change the default password on first login ('*password*')
  - Assign the complete email for the *root* account
    root@host.fqdn
  - Assign all user rights:
    Global -> User Rights

User Creation

- Create a userid for each member of your NOC team.
- Assign privileges to each user.

Create Groups

- Create groups of users:
  - Administering privileges by group is more efficient that doing so for each user.

Create Queues

- Create queues for problem categories
  - For example
    - security
    - accounts
    - connectivity
  - Assign users to each queue
    - Different between AdminCC and CC
  - Don't forget to create email *aliases* for each queue

Scrips (actions)

- For each queue create automatic actions
  - There is a group of scrips that apply to all queues.
    - Possible to customize per queue or globally
    - "*scrips*" are "snippets of Perl code"

Extensions

- You can extend the functionality of RT. For example:
  - Send daily emails to remind users of tickets that have not been “taken”
  - Send daily emails to each user reminding them of their pending tickets.
  - Periodically increment ticket priority
  - You can execute commands via email
    - **http://wiki.bestpractical.com/index.cgi?Extensions**

References

- *Best Practical* Web site

  http://bestpractical.com/rt
- *RT Essentials*. Dave Rolsky et al. O'Reilly