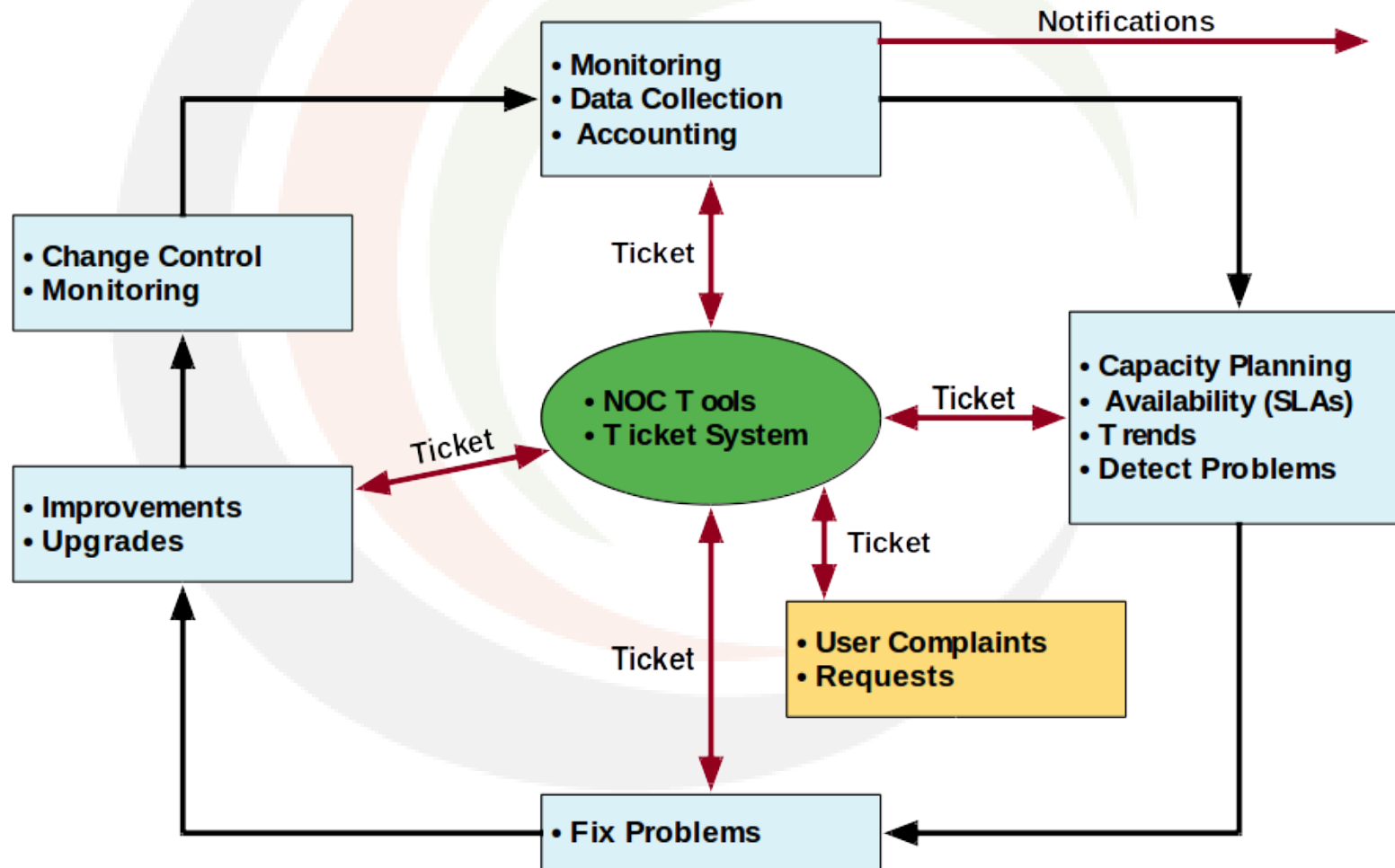# Ticketing Systems with RT

Network Startup Resource Center
www.nsrc.org

Ticketing Systems (Contd.)

- Use ticket system to follow cases, including communication between techs
- Each case is assigned a case number
- Each case goes through a similar life cycle:
  - *New – Open – … – Resolved – Closed*

Why Ticketing Systems?

Ticketing Systems

Why are they important?
- Track all events, failures and issues
- Focal point for help desk communication

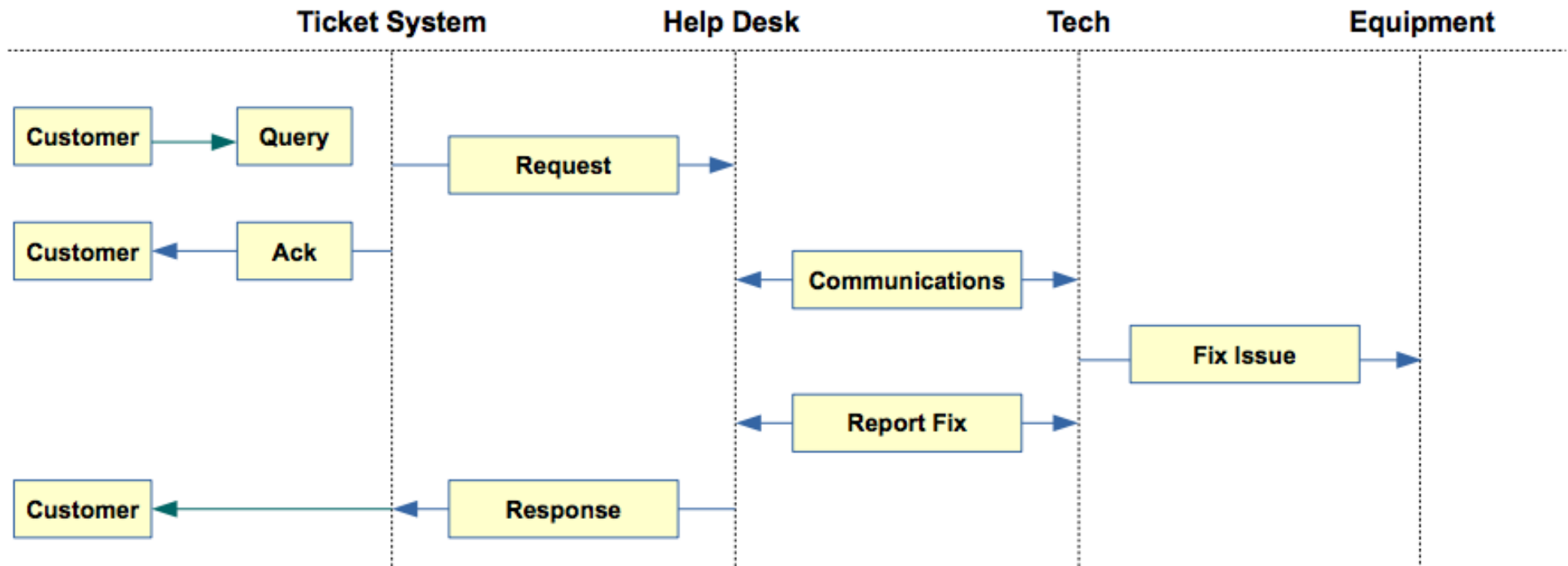**Use it to track all communications**
- Both internal and external

Events originating from the outside:
- customer complaints

Events originating from the inside:
- System outages (direct or indirect)
- Planned maintenance, upgrades, etc

Help Request with Tickets

A Few Others

- **Bugzilla:** http://www.bugzilla.org/
- **Cerberus:** http://www.cerberusweb.com/
- **Eticket:** http://www.eticketsupport.com/
- **Itracker:** http://www.itracker.org/
- **Jutda Helpdesk:** http://www.jutdahelpdesk.com/
- **Mystic:** http://www.hulihanapplications.com/projects/mystic
- **OTRS:** http://otrs.org/
- **OsTicket:** http://osticket.com/
- **Simple Ticket:** http://www.simpleticket.net/
- **Trouble Ticket Express:**
  http://www.troubleticketexpress.com/

RT: Request Tracker

Demo http://bestpractical.com/rt/

What Does It Look Like?

Ticket Management Systems

Why do we use the term "ticket"?

- https://en.wikipedia.org/wiki/Issue_tracking_system
- 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, guest
- Authentication and authorization
- Event history
- Handles dependencies
- Notification

Typical Support Scenario

- Lots of email traffic requesting help, request for 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
- Flexible
- Web interface or control via email
- Backend database (MySQL, Postgresql, Oracle, SQLite)

Components

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

RT: Disadvantages

A bit tricky to install the first time...

Most distributions have packages that make installation a bit easier:

- Red Hat, Fedora, SuSE, Debian, Ubuntu, FreeBSD, etc.

It's powerful, so you'll need to spend some time learning how it works.

Problem Classification: Queues

Services: DNS, IP addresses, Radius, LDAP
Security: Attacks, scans, abuse, etc.
Sytems: Email accounts, passwords, etc.
Networking: Network Services Group
Help Desk: Those who deal with end-users

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 team
Assign privileges to each user

Create Groups

Create groups of users:

Administering privileges by group is more efficient than doing so for each user.

Create Queues

Create queues for problem categories
For example
Security   **Accounts**   **Connectivity**
Assign users groups and groups to each queue
Different between AdminCC and CC
Don't forget to create email *aliases* for each queue

rt-mailgate

rt-mailgate facility lets us:

Define virtual users on the RT server that correspond to ticket queues in RT.

Allow third-party software (Nagios, Cacti, Smokeping, etc.) to automatically generate tickets in specified queues via email.

Provide a simple interface through which end-users can communicate with your support organization via RT.

More details at https://www.bestpractical.com/docs/rt/4.0/rt-mailgate.html

Scripts (actions)

Create automatic actions for queues

There is a group of scrips that apply to all queues.

- Possible to customize per queue or globally
- "*scrips*" are "snippets of Perl code"
- Chapter 6 of the O'Reilly "*RT Essentials*" book
- Details on how to use Scrips:
- http://requesttracker.wikia.com/wiki/Scrip

Extensions

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

References

Best Practical Web site

   http://bestpractical.com/rt

RT Essentials. Dave Rolsky et al. O'Reilly Media, Inc.

Contributions to RT:

http://requesttracker.wikia.com/wiki/Contributions

Extensions

http://requesttracker.wikia.com/wiki/Extensions
http://bestpractical.com/rt/extensions.html

Scrips

http://requesttracker.wikia.com/wiki/Scrip
http://requesttracker.wikia.com/wiki/ScripAction

Transforming education through ICT