

# ChatOps

at



# Why Group Chat?

- Searchable conversations
- Remote workers are no longer second class citizens
- You can read what you missed
- Multiple conversations at once (get more done)
- ChatOps / Robots!



# Group Chat Options?

- Slack
- Campfire
- Good Old IRC
- HipChat

*\* Make sure it has a good mobile client with push notifications*



# What is ChatOps?

*A set of tools that make operational tasks easier using Chat*

# Example Uses

- Monitoring (Graphite, logs, etc)
- Bug change notifications
- Checking production status
- Code review / Version control integration
- Agile Process automation

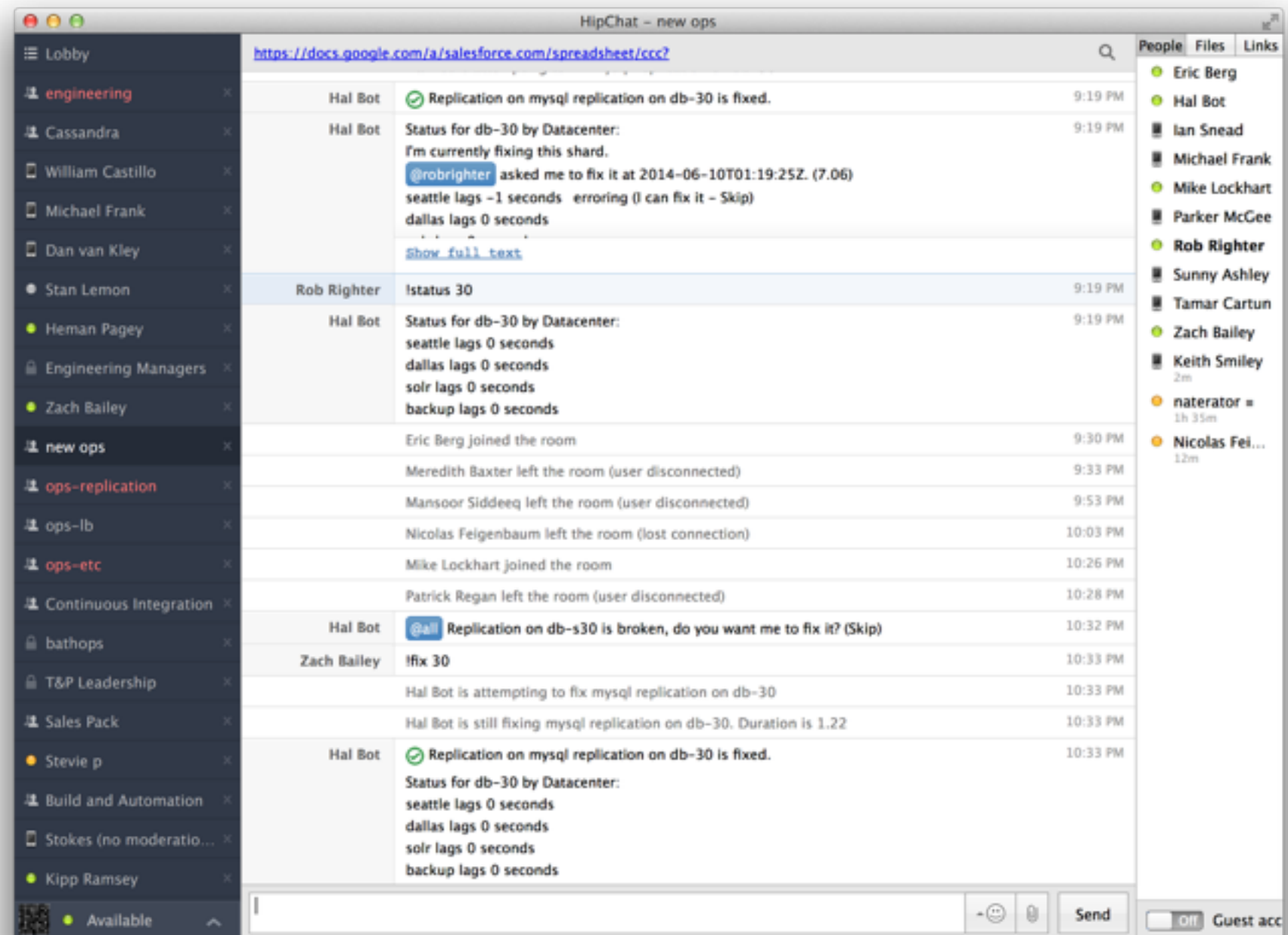
# ChatOps Platforms

- Hubot  
<https://hubot.github.com>
- Wobot  
<https://github.com/cjoudrey/wobot>
- Node-Irc  
<https://github.com/martynsmith/node-irc>



# What we do

- Hipchat
- Wobot



# A Short History of Hal

- Started as a “FedEx” project to automate mysql master/master replication errors
- Quickly turned into a key tool for day to day work
- Operationalized
  - Added a recurring job system
  - Added tests
  - Shell Script distributed with chef (telhal)
- Continues to be a growing project
- Developers love working on it. Everybody wants to pick up these stories in sprints





# Lessons Learned

# #1 Write Tests

Just because its not really in production doesn't mean  
that bugs aren't a big deal

# #2 Make it Easy for Everyone to Contribute

Adoption happens when everybody can easily add the features they want. Remove all barriers for contributions.

# #3 Make a Dev Environment

Do not let robots under development into your main chat rooms.

# #4 Don't re-invent existing tools

Chat Bots don't replace logstash, graphite, etc. They use them. If you're rebuilding an operational tool in robot code, your doing it wrong.

# #5 Take the time to refactor into reusable tools

Make sure your features are not team specific.

# Useful Patterns

# Ops Messages

- Any servers can send messages to chat rooms
- Each message is tagged with a category
- Uses the Event Emitter pattern so modules can listen for specific tags



# Ops Messages

- Notifications that Backups are Finished
- Notifications that a build has completed
- Notifications that a deployment has completed
- Great for babysitting newly deployed ops functionality

# Ops Messages

## **ops-message-handler.js**

<https://github.com/robrichter/chat-ops-helpers/blob/master/ops-messages/ops-message-handler.js>

# Ops Messages

## Usage Example:

```
opsMessageHandler.on('message', function(tag, contents) {  
  
    if(tag === MYSQL_REPLICATION_TAG) {  
        handleReplicationMessage(contents);  
    }  
  
});
```

# Ops Messages

## **tellhal bash script**

<https://github.com/robrichter/chat-ops-helpers/blob/master/ops-messages/tellhal>

# Recurring Tasks

- Similar to Cron
- Define a callback to be called at a specified:
  - Date/Time
  - OR
  - Interval

# Recurring Tasks

## Example Usage

- Call everybody to a scrum meeting
- Perform interval based monitoring
- Reminders
- Any ChatOps event not triggered by a user or ops message

# Recurring Tasks

## **Usage: Scrum Reminder**

<https://github.com/robrigher/chat-ops-helpers/blob/master/recurring-actions/jobs/scrum-message.js>

## **Usage: Graphite Monitor**

<https://github.com/robrigher/chat-ops-helpers/blob/master/recurring-actions/jobs/graphite-alerts.js>

# Recurring Tasks

## **recurring-manager.js**

<https://github.com/robrichter/chat-ops-helpers/blob/master/recurring-actions/recurring-manager.js>



# Monitoring

- Setup recurring tasks to watch for problematic events
- Use JSON api's provided by logstash and/or graphite
- Notify a chat room when a particular threshold is reached

# Monitoring

Setup items to monitor via config style files that are easy to maintain

## **Logstash Example**

<https://github.com/robrichter/chat-ops-helpers/blob/master/monitoring/logstash-queries.js>

## **Graphite Example**

<https://github.com/robrichter/chat-ops-helpers/blob/master/monitoring/graphite-queries.js>

# Monitoring

Implementation

## **Throttling**

<https://github.com/robrichter/chat-ops-helpers/blob/master/monitoring/throttled-message-handler.js>

## **Logstash**

<https://github.com/robrichter/chat-ops-helpers/blob/master/monitoring/logstash-query-listener.js>

## **Graphite**

<https://github.com/robrichter/chat-ops-helpers/blob/master/monitoring/graphite-alerts.js>

# Questions?



<https://github.com/robrighter>



@robrighter