# There Must be Rules!

By
Wolfgang Ziegler (fago) & Richard Jones (ikos)

# Who are we?

- Wolfgang Ziegler
  @the_real_fago
  Maintainer of Rules and Entity API
  CTO of drunomics

- Richard Jones
  @richardjones
  CTO of i-KOS

# Why Rules?

▪ Rules are about Reactions to events which happen on your site

▪Use it for automated workflows

▪Configure custom actions to be executed by modules

# Examples

- Notify me when a user posts a comment on my site

- or just when an anonymous user posts a comment containing some certain bad words

# Anatomy of a Rule

- A rule consists of:

  - One or more events

  - A set of actions

  - A set of conditions that determine whether the rule should execute

# Events

- A user has logged in

- Content has been updated

- State Transition

- A product has been added to the cart

# Actions

- Grant a role to a user

- Set a data value

- Create an entity

# Conditions

- Content is of type

- User has roles

- Data comparison

# Combining Rules and Data

▪Configuring data sources

▪Use the data selector to get what you need.

▪Provide some fixed configuration, e.g. select one of the

available options or provide a text.

▪Switch between both modes as you like

# Components

- Grouping a set of actions or conditions together so that they can be reused.
- Reusable on their own so they can be invoked by other rules
- Or by custom code
- Or by test cases
- Reuse
  - via actions or conditions
  - via modules, e.g. VBO
  - or by custom code
  - or by test cases

# Scheduling

- Can schedule actions to be performed at specific times

- Schedule the evaluation of a component

- With any kind of parameters!

# Loops

- Can iterate over a set of data and perform actions against multiple items

# Extending Rules

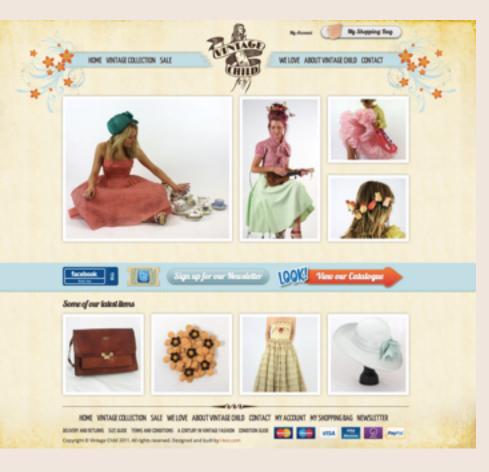- As it's Drupal, you can define you own events, conditions and actions
- Rules provides a solid API for adding event, actions and conditions
- Used by a lot of modules
- Benefits from integration with entity API
- and more

# Rules in Action

- How rules interacts with other Drupal Modules

- Using a Real-life case study

# How can we charge different VAT for different products?

- Rules' APIs allow modules to re-use it and to embed some of its UI
- Drupal Commerce does so
- Uses a generated component for applying VAT

# Video Demo

VAT-1

# I've decided to discount some products...

- 10% off
- But only this week
- And only T-Shirts
- And maybe just on products that have been on the site for a while...just so I can clear some stock

# Video Demo

# That worked great! But now I am out of stock...

- Can you set up something so people can get an email when items are back in stock?

# Add some new modules...

- Rules integrates with Flag
- Allow the customer to set a flag when an item is out of stock

# The "Back in Stock" Rule

- After updating a Commerce product
- If product is back in stock
- Fetch users who have flagged the product
- Loop over the users
- and send a mail to each of them
- reset the flag for each of them

# Video Demo

**It's working really well - I had to hire someone to load on my new stock.**

- Now I need to make sure they can't put stuff on the site before I check it

# Add some new modules

- Workbench
- with Rules!

# Workbench Moderation Rules

- Event: item move between workflow states
- Action: email is sent to site owner

# Video Demo

# Now I have loads of products... and loads of orders... how can I process them all quickly?

# Video Demo

# I want to end my promotion when I've sold 100 items... how can I schedule it to finish? Special offer Monday

- auto unpublish the offer Tuesday
- auto republish on Monday

# Video Demo

# Scheduled Rule - Abandoned Carts

- Check if user has completed
- Use Scheduler
- Send email on cart abandoned

# Video Demo

**OK I want to set up a free gift when a certain coupon is added and the customer has one at least 2 t-shirts in their cart but doesn't have any shoes and no other items in a promotion and there should only be 2 free gifts per customer.**

# Time for Questions!