

# EvQueue

Ordonnanceur

Planificateur

Job Scheduler

Moteur de Queueing

# L'UFC-Que Choisir

- Association loi 1901
- Protection des consommateurs
  - Information, tests de produits et services
  - Soutien juridique des consommateurs
  - Veille et lobbying dans l'intérêt des consommateurs
- 130 salariés (à Paris)
- 3 000 bénévoles dans 150 *associations locales*
- Des magazines, un site Internet

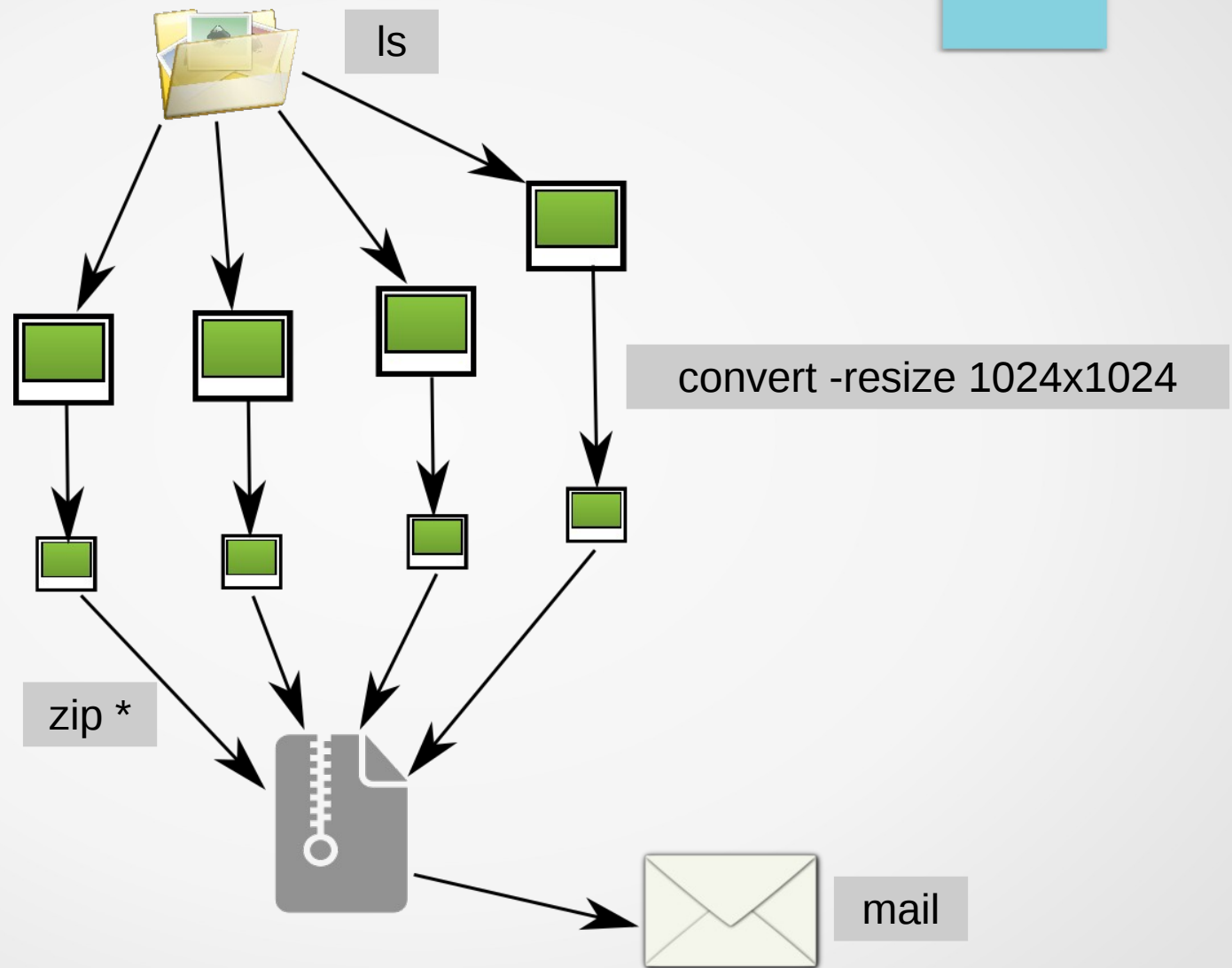
# EvQueue, la genèse

- Besoin : communication informatique avec un tiers (prestataire de service)
  - nécessité de faire de la reprise sur erreur, du suivi
- Étude de l'existant : logiciels trop complexes, non-libres, trop peu modulaires
- Publication sous licence libre : mars 2015
  - pas d'intérêt économique
  - accord de la direction
  - mais un an de latence !

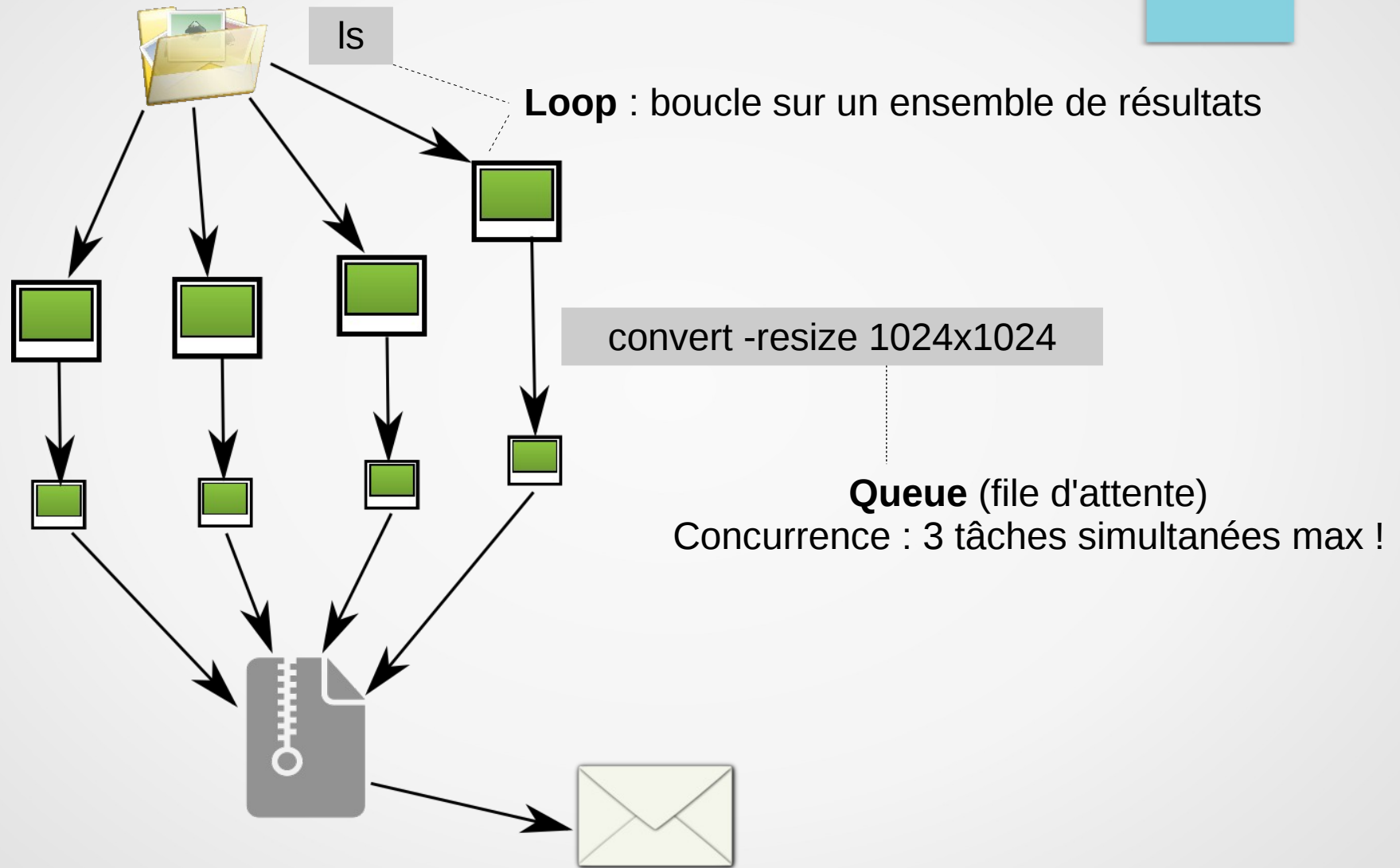
# Concepts

- La **tâche** : une brique logicielle indépendante
  - ligne de commande
  - prend des paramètres d'entrée (env, argv, stdin)
  - renvoie ou génère un résultat
  - exécutée en local ou sur une machine distante
- Le **workflow** : un enchaînement de tâches
  - dans un ordre déterminé
  - avec passage d'information de l'une à l'autre
  - dans un but plus global

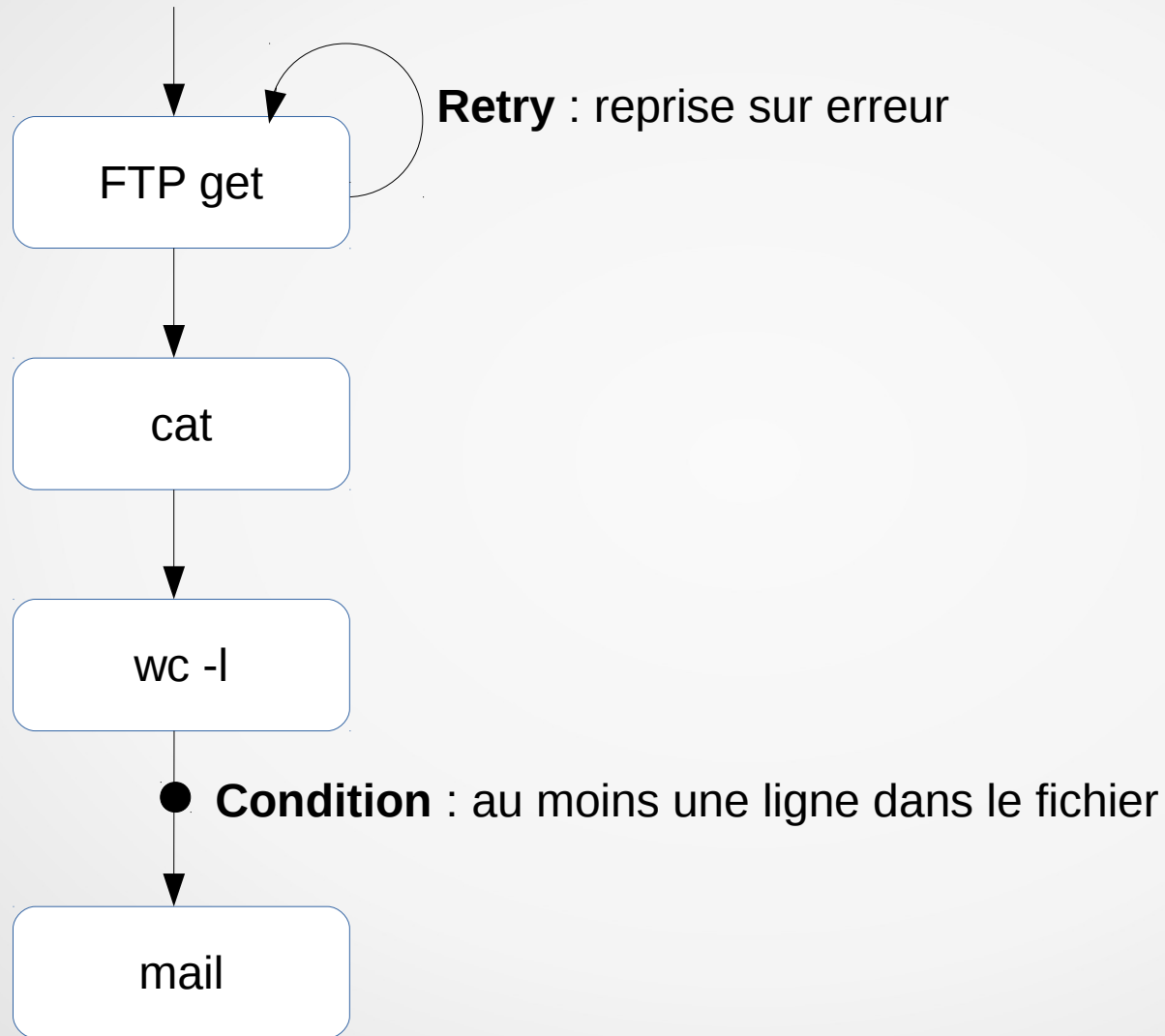
# Un workflow de traitement d'images



# Control the flow



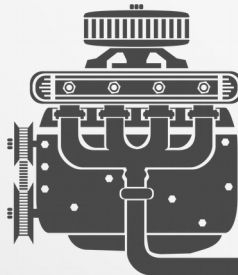
# Encore plus de contrôle



# EvQueue, moteur et interface

## evqueue-core

Ordonnance, planifie, exécute  
(c++, daemon, événementiel)



## evqueue-front-end

Interface web de gestion qui  
donne accès à l'ensemble des  
fonctionnalités

Appels sur interface réseau

The screenshot shows the EvQueue web interface. At the top, there are navigation tabs: "System state", "Settings", "Notifications", and "Logging". Below these, there are links for "Configure" and "Manage plugins". The main content area is divided into two sections. The first section is titled "3 EXECUTING Workflows" and contains a table with columns for "State", "ID - Name", "Host", "Started", and "Actions". The second section is titled "< TERMINATED Workflows 1-30 (105976 total) >" and also contains a table with similar columns. The "Actions" column in both tables contains icons for refresh, stop, and delete.

State	ID - Name	Host	Started	Finished	Actions
🔄	5165491 - order_create_and_pay (1h 53m 44s)	localhost	2015-07-01 15:38:38		🔄 ⏹ 🗑
🕒	5165245 - client_edit (2h 25m 11s)	localhost	2015-07-01 15:07:11		🔄 ⏹ 🗑
🕒	5160071 - generate_evol_expired (14h 32m 22s)	localhost	2015-07-01 09:00:00		🔄 ⏹ 🗑

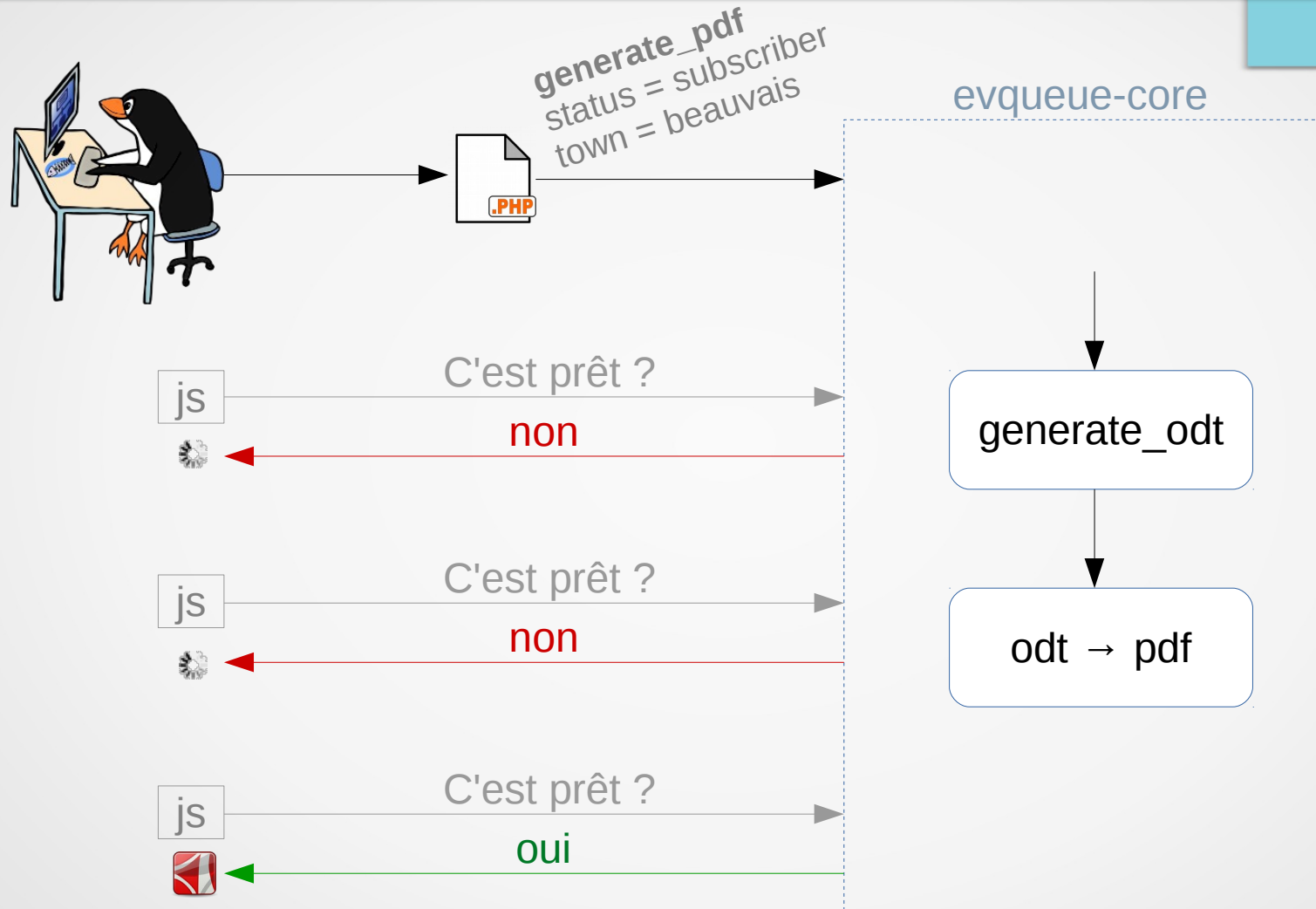
State	ID - Name	Host	Started	Finished	Actions
✅	5166279 - sij_deco (0s)	web1	2015-07-01 17:32:00	2015-07-01 17:32:00	🔄 ⏹ 🗑
✅	5166277 - od2pdf (1s)	localhost	2015-07-01 17:31:56	2015-07-01 17:31:51	🔄 ⏹ 🗑
✅	5166273 - payzen_cancel (6s)	localhost	2015-07-01 17:31:49	2015-07-01 17:31:44	🔄 ⏹ 🗑
✅	5166275 - od2pdf (0s)	localhost	2015-07-01 17:31:45	2015-07-01 17:31:45	🔄 ⏹ 🗑
✅	5166261 - majtest (1m 8s)	localhost	2015-07-01 17:30:00	2015-07-01 17:31:08	🔄 ⏹ 🗑
✅	5166271 - sij_deco (1s)	web1	2015-07-01 17:31:00	2015-07-01 17:31:01	🔄 ⏹ 🗑
✅	5166259 - search_quechoisir (24s)	localhost	2015-07-01 17:30:00	2015-07-01 17:30:24	🔄 ⏹ 🗑
✅	5166257 - import_XML (22s)	localhost	2015-07-01 17:30:00	2015-07-01 17:30:22	🔄 ⏹ 🗑
✅	5166267 - ufcezfhow (4s)	localhost	2015-07-01 17:30:00	2015-07-01 17:30:04	🔄 ⏹ 🗑
✅	5166265 - frequent (2s)	localhost	2015-07-01 17:30:00	2015-07-01 17:30:02	🔄 ⏹ 🗑



# Use case 1 : Site web et traitement lourd

- Problématique :
  - notre visiteur navigue sur le site
  - une action demande un traitement long (30s - 1min d'exécution)
- Le but du jeu :
  - laisser l'internaute à sa navigation
  - le prévenir quand le traitement est terminé
- Exemple : génération d'un fichier PDF
  - une page par adhérent de mon association
  - publipostage classique

# Use case 1 : Site web et traitement lourd






## Use case 2 : Gestion d'erreur

- Causes possibles de tâches en erreur
  - la ressource n'est pas encore présente (fichier sur ftp)
  - le service que je contacte est cassé (prestataire HS)
  - l'action entreprise n'est pas autorisée (cas non prévu)
- Dans tous les cas :
  - on réessaye plus tard => retry schedule
  - on a une vue d'ensemble dans l'interface de monitoring

# Use case 2 : Retry schedule

### Update Schedule

Schedule name:

Retry every X seconds	for Y times	
<input type="text" value="10"/>	<input type="text" value="6"/>	
<input type="text" value="60"/>	<input type="text" value="10"/>	
		

On attache un retry schedule à une tâche

# Use case 2 : Erreurs et monitoring

1 EXECUTING Workflows  Auto-refresh

State	ID - Name	Host	Started	Actions
	115733 - order_create_and_pay (3days, 15h 6m 42s)	localhost	2015-06-29 00:45:06	
	(version 2.1) real time 53s queued for 3h 5m 49s waited 3days, 12h 0m 0s for retrials			
	Task <b>client_get_by_id</b> [local-moncompte] [biwing]  (3 sub-tasks) last execution time: 00:45:06			
	Task <b>order_line_delete</b> [local-moncompte]			
	Task <b>biwing_inform_client_creation_failed</b> [local-email]			
	Task <b>client_get</b> [biwing-client-read] [biwing]  (4 sub-tasks) last execution time: 00:45:06			
	Task <b>payzen_cancel</b> [payzen] [payzen]			
	Task <b>payment_treated_by_biwing</b> [local-moncompte]			
	Task <b>send_email_client_suspended_subscription</b> [local-email]			
	Task <b>basket_create</b> [biwing-basket-write] [biwing]   2015-07-02 18:45:59 last execution time: 12:45:59			
	<pre>&lt;?xml version="1.0" encoding="UTF-8"?&gt; &lt;errors&gt; &lt;error&gt;Réabonnement déjà en attente&lt;/error&gt; &lt;/errors&gt;</pre>			
	Task <b>bill_create</b> [biwing-basket-write] [biwing] .. Waiting for the previous job to finish			
	Task <b>order_line_update</b> [biwing-basket-read] [biwing] .. Waiting for the previous job to finish			
	Task <b>bill_pay</b> [biwing-basket-write] [biwing] .. Waiting for the previous job to finish			
	Task <b>client_get</b> [biwing-client-read] [biwing] .. Waiting for the previous job to finish (3 sub-tasks)			
	Task <b>payment_treated_by_biwing</b> [local-moncompte] .. Waiting for the previous job to finish			
	Task <b>send_email_subscription_confirmed</b> [local-email] .. Waiting for the previous job to finish			

# Interface web - GUI

Updating Workflow 141: send\_email\_ask\_payment\_adh

Name:

Parameters list:

- order\_id
- email
- 

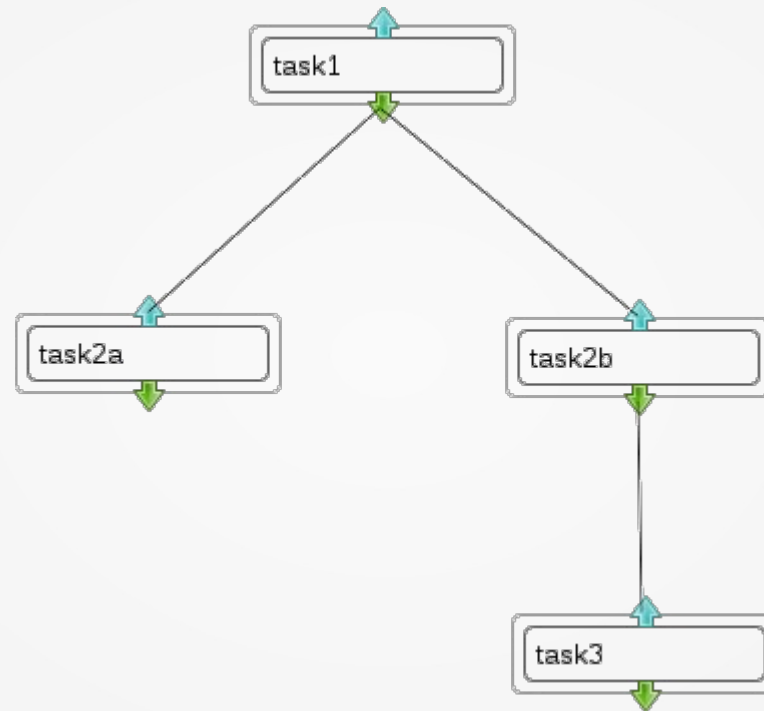
Task **adh\_cheque\_fusion** [local-email]

Task **odt2pdf** [local-odt2pdf]

Task **send\_email** [local-email]

- email:** `/workflow/parameters/parameter[@name='email']`
- order\_id:** `/workflow/parameters/parameter[@name='order_id']`
- url\_to\_pdf:** `tasks/task[@name='odt2pdf']/output/output/pdf-filename`
- STDIN** not used yet
-

# Interface web du futur - GUI



# Interface web - fonctionnalités

- Gestion des utilisateurs et droits d'accès
- Planification de tâches ou workflows
- Notification en fin de workflow sur erreur ou succès
  - par mail, extensible par plugins
- Import / export de workflows (v1.4)
  - simple fichier zip
  - contenant le workflow et ses tâches



# Conclusion

- C'est libre, c'est là, c'est utilisable
- C'est fonctionnel : 5 millions de workflows à Que Choisir
- Simple à installer – paquets debian et repo disponibles
  - et simple à utiliser !
- Il nous manque juste un logo :)
  - mais si vous voulez tester, on est preneurs de retours !

Merci !

<http://www.evqueue.net>  
[dev@evqueue.net](mailto:dev@evqueue.net)

<https://github.com/coldsource/evqueue-core>  
<https://github.com/coldsource/evqueue-frontend-php>

Sources graphiques : <https://openclipart.org>