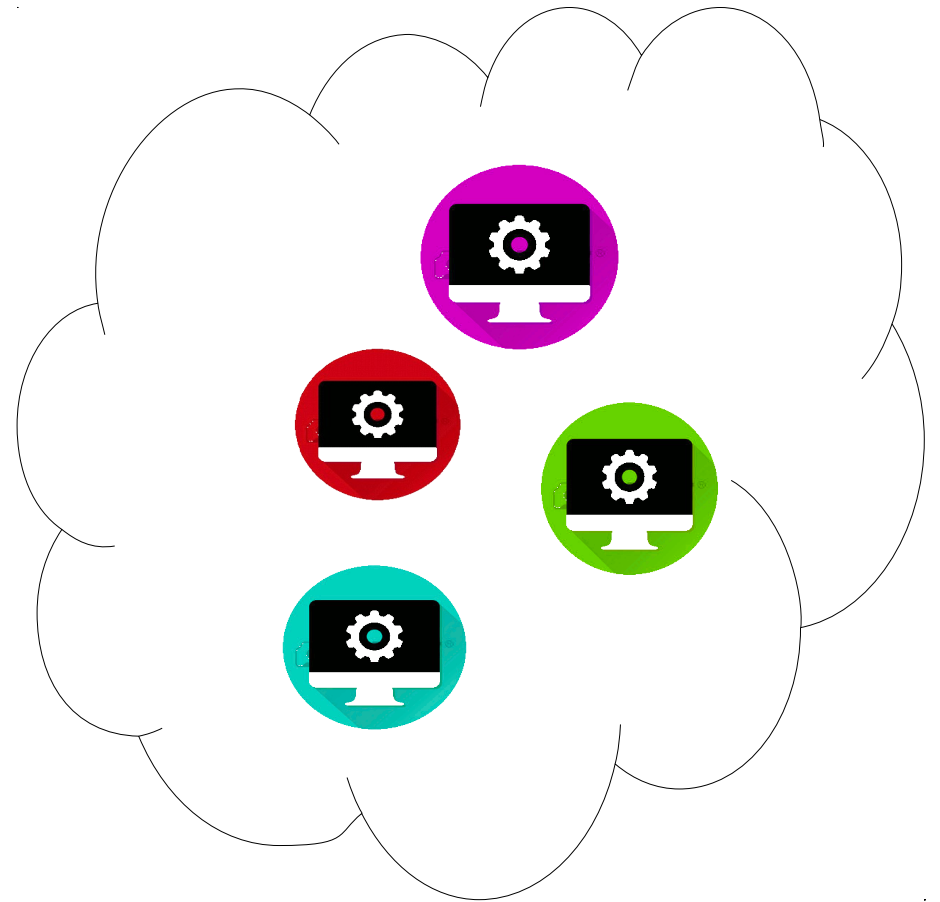


# Universal **M**iddleWare **I**ndustrialyzer

```
## Copyright (c) 2007-2008, Patrick Germain Flacidoux
## All rights reserved.
##
## This file is part of kikanf.
##
## kikanf is free software: you can redistribute it and/or modify
## it under the terms of the GNU General Public License as published by
## the Free Software Foundation, either version 3 of the License, or
## (at your option) any later version.
##
## kikanf is distributed in the hope that it will be useful,
## but WITHOUT ANY WARRANTY; without even the implied warranty of
## MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
## GNU General Public License for more details.
##
## You should have received a copy of the GNU General Public License
## along with kikanf. If not, see <http://www.gnu.org/licenses/>.
##
## Home: http://www.kikanf.org
## Contact: kikanf@gmx.com
```

# Prenez un ingénieur à la journée



Effectue un certain nombre de taches à la journée

# Des taches: des paramètres

Four overlapping windows of a Windows command prompt, each showing the usage and options for the 'ping' command. The windows are arranged in a stack, with the top window partially obscured by the others. The text in the windows is as follows:  
Top window:  
C:\WINNT\system32\cmd.exe  
C:\Documents and Settings\Administrator>ping  
Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS] [-r count] [-s count] [[-j host-list] | [-k host-list]] [-w timeout] destination-list  
Options:  
-t Ping the specified host until stopped.  
-a Resolve addresses to hostnames.  
-n count Number of echo requests to send.  
-l size Send buffer size.  
-f Set Don't Fragment flag in packet.  
-i TTL Time To Live.  
-v TOS Type Of Service.  
-r count Record route for count hops.  
-s count Timestamp for count hops.  
-j host-list Loose source route along host-list.  
-k host-list Strict source route along host-list.  
-w timeout Timeout in milliseconds to wait for each reply.  
Second window:  
C:\WINNT\system32\cmd.exe  
C:\Documents and Settings\Administrator>ping  
Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS] [-r count] [-s count] [[-j host-list] | [-k host-list]] [-w timeout] destination-list  
Options:  
-t Ping the specified host until stopped.  
Third window:  
C:\WINNT\system32\cmd.exe  
C:\Documents and Settings\Administrator>ping  
Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS] [-r count] [-s count] [[-j host-list] | [-k host-list]] [-w timeout] destination-list  
Options:  
-t Ping the specified host until stopped.  
Fourth window:  
C:\WINNT\system32\cmd.exe  
C:\Documents and Settings\Administrator>ping  
Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS] [-r count] [-s count] [[-j host-list] | [-k host-list]] [-w timeout] destination-list  
Options:  
-t Ping the specified host until stopped.  
-a Resolve addresses to hostnames.  
-n count Number of echo requests to send.  
-l size Send buffer size.  
-f Set Don't Fragment flag in packet.  
-i TTL Time To Live.  
-v TOS Type Of Service.  
-r count Record route for count hops.  
-s count Timestamp for count hops.  
-j host-list Loose source route along host-list.  
-k host-list Strict source route along host-list.  
-w timeout Timeout in milliseconds to wait for each reply.

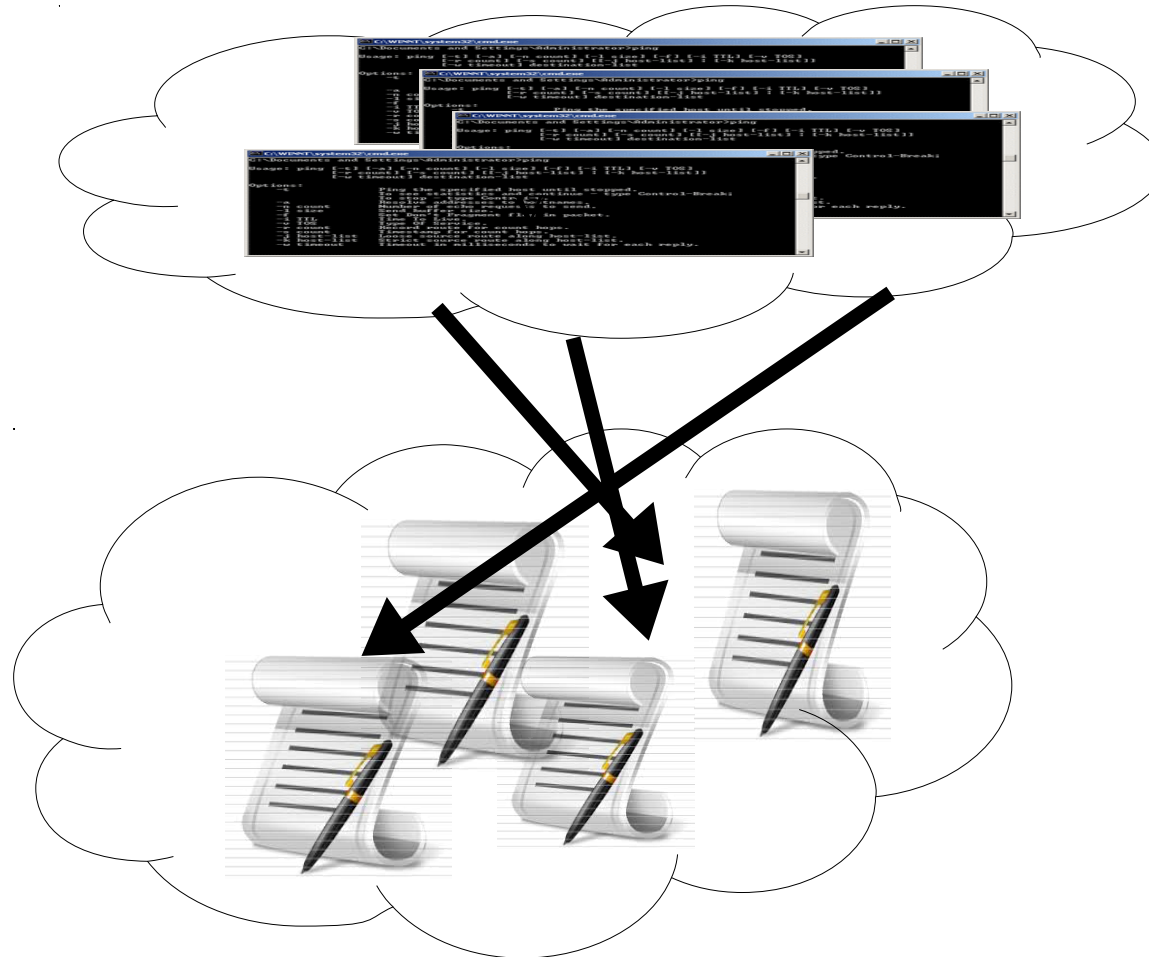
On observe que chacune de ces taches correspond à un jeu de paramètres,  
On observe qu'on en perd la trace, que ces paramètres sont peu lisibles (not human readable).

# Des taches : un ordre d'exécution



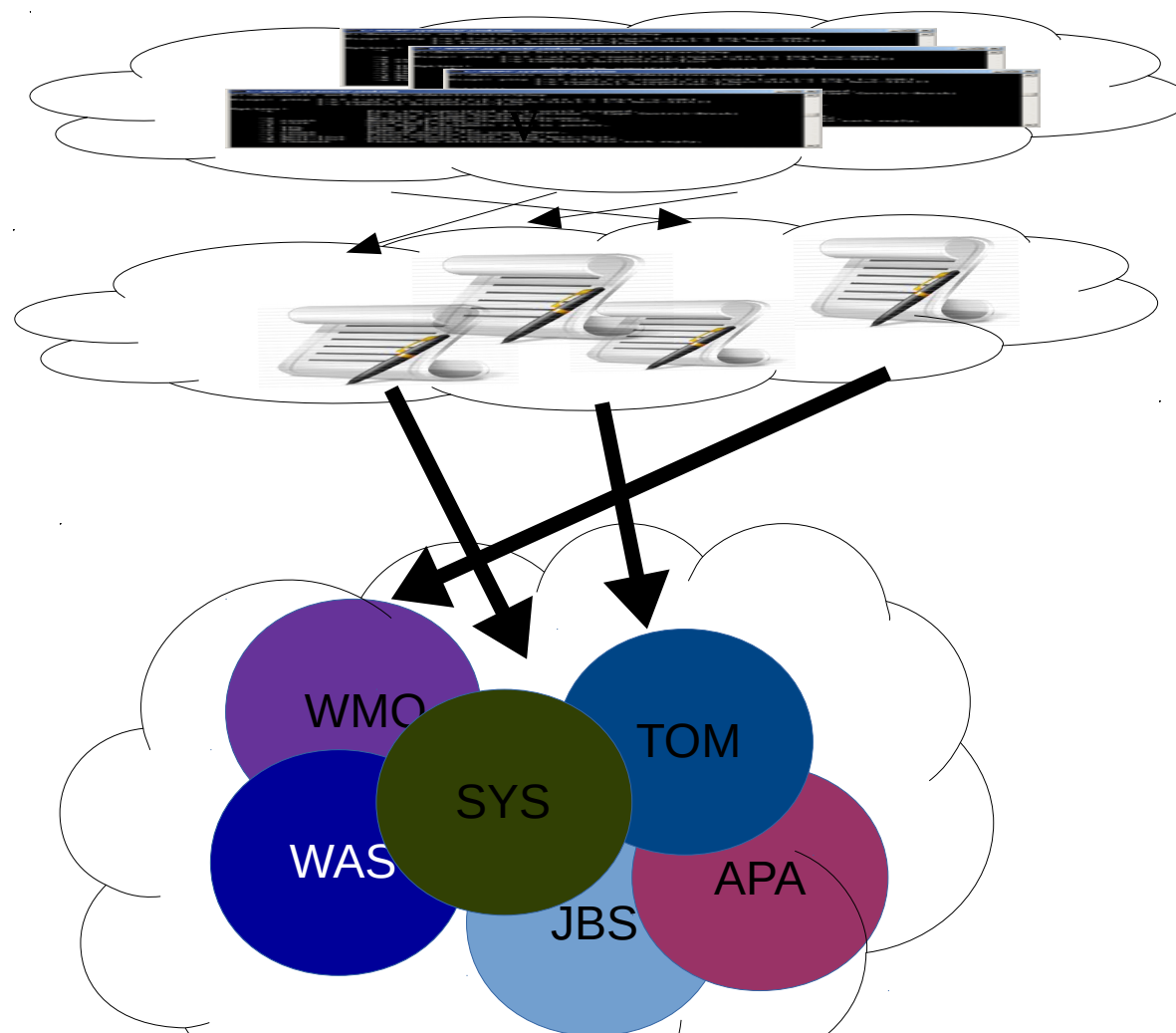
On observe que ces taches sont exécutées selon une ordre spécifique,  
On observe qu'on en perd la trace, que cet ordre est peu lisible.

# Des taches : des scripts



**On observe que ces paramètres s'appliquent à un jeu de scripts.**  
On observe qu'on en perd la trace, que leur identification est peu lisible (bloated).

# Des taches : des plateformes cibles



**On observe que ces taches s'appliquent à des plate-formes cibles.**  
On observe qu'on en perd la trace, que les plate-formes ciblées ne sont pas clairement évidentes.

De nuages scabreux en nuages scabreux  
On en arrive !

# Au bastion



De l'ingénieur qui détient un capital de commande.



Comment avoir de la lisibilité sur les  
Taches ?

# Qu'est ce qu'une tache



## Un Jeu de paramètres

```
C:\WINNT\system32\cmd.exe
C:\Documents and Settings\Administrator>ping
Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-w IOS]
          [-r count] [-s count] [[-j host-list] | [-k host-list]]
          [-v timeout] destination-list

Options:
-t          Ping the specified host until stopped.
            To see statistics and continue - type Control-Break;
            To stop - type Control-C.
-a          Resolve addresses to hostnames.
-n count   Number of echo requests to send.
-l size    Send buffer size.
-f          Set Don't Fragment flag in packet.
-i TTL     Time To Live.
-w IOS     Type Of Service.
-r count   Record route for count hops.
-s count   Timestamp for count hops.
-j host-list Loose source route along host-list.
-k host-list Strict source route along host-list.
-v timeout Timeout in milliseconds to wait for each reply.
```

## Un Script qui fait le job



## Une Plate-forme cible

WebSphere ®

Tomcat ®

Jboss ®

Apache ®

Mq Series ®

OS (Aix ®/Linux)

...

# 1. Le Jeu de paramètres

- Les présenter de façon structurée.
- Avec un intitulé par argument lisible par tous (human readable).
- Donner aux opérateurs des accès multiples par des interfaces simples.
- Les historiser (reproductibilité).

# En informatique le format de présentation structuré est le xml

```
<crtcluster type='action' name='rrefr1_remise_cls_01' enable_ha='false'/>
```

```
<crtserver type='action' bal='was.crtserver'>  
  <scope server='rrefr1_remise_jvm_02' node='WAS7RC1_Node4' cluster='rrefr1_remise_cls_01'/>  
</crtserver>
```

```
<instapp  
  type='action'  
  bal='was.instapp'  
  name='rrefr1_remise_app'  
  ear_path='/stor/rrefr1/was/7/livraison/installable/rrefr1_remise_app.ear'  
  ear_install_path='/stor/rrefr1/was/7/deploy'  
  backup_path='/stor/rrefr1/was/7/livraison/archives'  
  vhost='rrefr1_remise_vhost'  
>  
  <scopes>  
    <scope cluster='rrefr1_remise_cls_01'/>  
    <scope node='WASHDRC1_ihs1-node' server='rrefr1_remise_wbs_01' server_type='ws'/>  
    <scope node='WASHDRC1_ihs1-node' server='rrefr1_remise_wbs_03' server_type='ws'/>  
    <scope node='WASHDRC1_ihs2-node' server='rrefr1_remise_wbs_02' server_type='ws'/>  
    <scope node='WASHDRC1_ihs2-node' server='rrefr1_remise_wbs_04' server_type='ws'/>  
    <scope node='WASHDRC1_ihs2-node' server='rrefr1_remise_wbs_06' server_type='ws'/>  
  </scopes>  
</instapp>
```

Le xml est le format naturel du monde middleware gsa/etude ...

```
<mkdirs
  type='action'
  bal='sys.mkdir'
  allow_remove='true'
>
  <mustexist>
    <dir path ='/stor/rrefr1/was' as_fs='false'/>
    <dir path ='/stor/rrefr1/logs' as_fs='false'/>
    <group name = 'was7'/>
  </mustexist>

  <create>
    <group name = 'rrefr1'/>
    <user name = 'rrefr1' group = 'wasgrp' groups = 'rrefr1,was7' home = '/stor/rrefr1'/>

    <dir path = '/stor/rrefr1/logs/rrefr1'>
      <access user = 'rrefr1' group = 'wasgrp' mode = '752' recursive = 'false'/>
    </dir>

    <dir path = '/stor/rrefr1/was/7/deploy'/>
    <dir path = '/stor/rrefr1/was/7/livraison/installable'/>
    <dir path = '/stor/rrefr1/was/7/livraison/archives'/>
    <dir path = '/stor/rrefr1/was/7/sharedlib'/>
    <dir path = '/stor/rrefr1/was/7/properties'/>
    <dir path = '/stor/rrefr1/was/7/rrefr1_remise_jvm_02/temp'/>
    <dir path = '/stor/rrefr1/was/7/rrefr1_remise_jvm_04/temp'/>

    <link path = '/stor/rrefr1/was/7/rrefr1_remise_jvm_02/logs'
      to = '/stor/rrefr1/logs/rrefr1_remise_jvm_02'/>
    <link path = '/stor/rrefr1/was/7/rrefr1_remise_jvm_04/logs'
      to = '/stor/rrefr1/logs/rrefr1_remise_jvm_04'/>
  </create>

</mkdirs>
```

- La sémantique des xml est human readable, elle correspond à des clauses significatives dans la logique de l'opérateur (pas de paramètres barbares).
- Ces fichiers sont stockés, donc l'action n'est pas perdue.
- La structure de ces fichiers est contrôlée de façon stricte on ne peut donc pas entrer n'importe quoi.
- Des droits par rôle sont applicables sur les fichiers eux-même ainsi qu'à chaque tag/attribut.

## 2. Le Script

- Rendre le script lisible
- Dans le langage naturel de la plate-forme concernée (donc de l'opérateur sur cette plateforme)
- Bannissement des scripts bibles qui font toutes les tâches à la fois.
- Mais des scripts unitaires focalisent sur l'action à réaliser.

# Pour chacune des 3 actions précitées on observe la présence du tag : **bal** (Basic Action Locator)

bal : <plate-forme>.<action>

- Qui permet de façon immédiate de retrouver les ressources de l'Action :

ex :

bal=was.url	==>	<kikonf>/plugins/actions/ <b>was/crtcluster</b>
bal=was.instapp	==>	<kikonf>/plugins/actions/ <b>was/instapp</b>
bal=sys.mkdirs	==>	<kikonf>/plugins/actions/ <b>sys/mkdirs</b>

(descripteur)

<kikonf>/plugins/actions/**was/crtcluster/by/kikonf**/ACT\_INF/action.xml

(script)

<kikonf>/plugins/actions/**was/crtcluster/by/kikonf**/**<version>/crtcluster.py**

- Et ceci s'applique strictement à toutes les Actions.
- Un script unique isolé par Action facilite la lisibilité.
- Les scripts sont exprimés dans le langage admin naturel de la plate-forme cible.
- Un opérateur peu copier coller /by/kikonf en /by/luimeme : modifier le code et faire tourner directement son Action avec les mêmes prérogative que les autre Actions.



# Le script de l'action was.crtcluster

```
def inject(self):
    self_funct='inject'
    crtcluster_node = self.getTop()
    crtcluster_attrs = crtcluster_node.getAttrs()
    self.verbose(name=crtcluster_attrs.name)
    indent=self.getIndent() + 3* ' '

    self.rmvCluster(crtcluster_attrs.name, indent=indent) #-- remove first

    cmdvalues=[]
    cmdvalues.append(['name', crtcluster_attrs.name,])
    cmdvalues.append(['enableHA', crtcluster_attrs.enable_ha])
    cmdvalues.append(['nodeGroupName', crtcluster_attrs.node_group])

    if crtcluster_attrs.desc!=None:cmdvalues.append(['description', crtcluster_attrs.desc])
    else:cmdvalues.append(['description', DESIGN_NOTICE])

    verbose('ServerCluster:' + crtcluster_attrs.name + ' creating.', level=self.getVerbose(), ifLevel=4,
    indent=indent, logFile=self.getLogFile())
    AdminConfig.create('ServerCluster', AdminConfig.getid('/Cell:'+ self.getCell()), cmdvalues)
    verbose('ServerCluster:' + crtcluster_attrs.name + ' created.', level=self.getVerbose(), ifLevel=3,
    indent=indent, logFile=self.getLogFile())
```

# 3. La Plate-forme cible

- Les Actions doivent-êtré clairement associées à la plate-forme ciblée, de façon humainement lisible.
- Les scripts d'Actions doivent parler le langage humain des plate-formes ciblées.  
<=> le langage opérateur système et pas le langage natif de la plate-forme.(ex WebSphere ®/Weblogic ®:wsadmin/wlst et non Java ® qui est incompréhensible pour l'ingénieur système non développeur)
- Faciliter la prise en main/modification du code.

L'instruction du script :

```
AdminConfig.create('ServerCluster', AdminConfig.getid('/Cell:' + self.getCell()), cmdvalues)
```

- Correspond à la commande de langage naturelle passée par la console WAS ® que l'opérateur peut observer/reproduire à la console en mode commande assistance.
- Les facades de command WAS sont effectivement directement mise à disposition des scripts : AdminConfig, AdminTask, AdminApp, AdminControle. Elles permettent le pilotage à 100 % de WAS.
- Ce qui est vrai pour la plate-forme WAS est vrai pour les autres.

# La séquence des Actions 1

- On peut exécuter une Action de façon unitaire  
ex :



was.instapp

**Kikact was.instapp**

- Ou de façon groupée selon un ordre :



was.instapp



was.clsloader



was.plugin

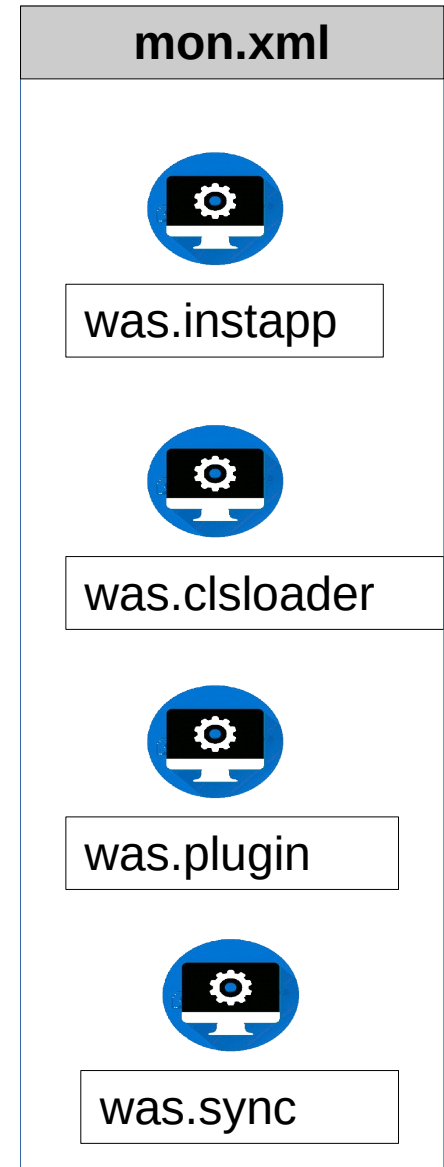


was.sync

**Kikact was.instapp,was.clsloader,was.plugin,was.sync**

# La séquence des Actions 2

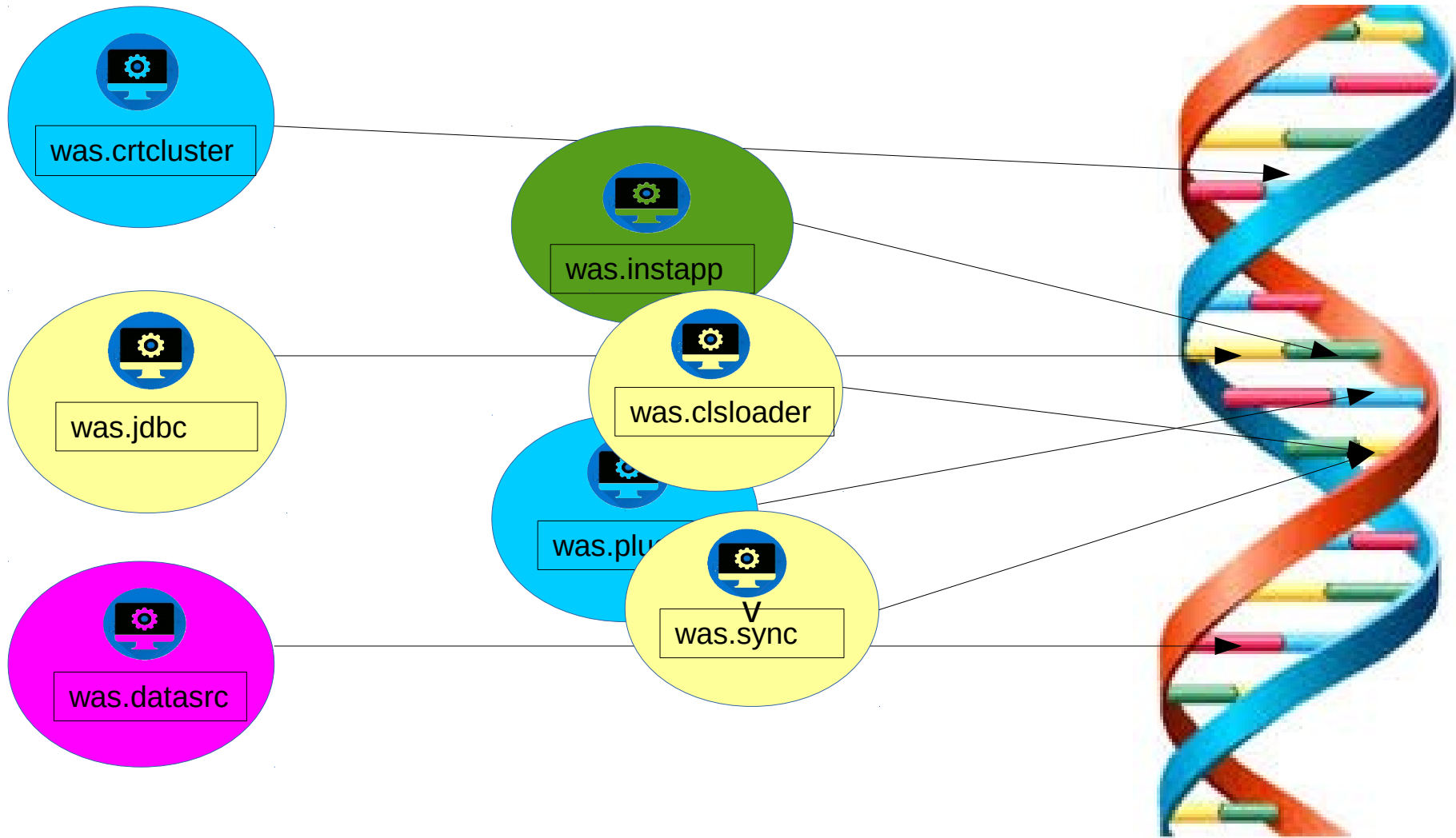
- On peut copier coller toutes les actions dans un seul fichier xml.



- Et les exécuter de façon groupée selon leur ordre d'apparition dans le xml :

**kikarc mon.xml**

# Penser ADN



- Une **Action** (tache unitaire) représente un **gène**.
- Un **jeu d'Actions** représente l'**ADN**.
- On peut faire un ADN pour une Application, un Qmanager, une Partition, ... le scope que l'on désire.

# Disponibilité aux Menus

- Toutes les actions unitaires sont disponibles à console (donc modifiable/exécutable par ce biais)

ex :

```
kikupd --kikact was.instapp
```

```
kikupd --kikact was.datasrc
```

- Le fait qu'elles auto-génèrent des Menus les rend incorporables dans tout autre Menu.

```
| ACTION: INSTAPP |
```

```
| _____ |
```

```
1/ instapp> [Install one Application.]  
name: rrefr1_remise_app  
ear_path: /stor/rrefr1/was/7/livraison/installable/rrefr1_remise_app.ear
```

```
ear_install_path: /stor/rrefr1/was/7/deploy  
backup_path: /stor/rrefr1/was/7/livraison/archives  
backup_remove_og: true  
vhost: rrefr1_remise_vhost  
sharedlibs: None  
post_script: None
```

```
2/ instapp/scopes/scope> [A scope refers to a Name Space where to create the resource]
```

```
cluster: None  
node: WASHDR1_ihs2-node  
server: rrefr1_remise_wbs_02  
server_type: ws
```

```
3/ instapp/scopes/scope> [A scope refers to a Name Space where to create the resource]
```

```
cluster: None  
node: WASHDR1_ihs2-node  
server: rrefr1_remise_wbs_04  
server_type: ws
```

```
4/ instapp/scopes/scope> [A scope refers to a Name Space where to create the resource]
```

```
cluster: None  
node: WASHDR1_ihs2-node  
server: rrefr1_remise_wbs_06  
server_type: ws
```

```
( Exit:0 Up:+)
```

```
Choice ?
```



[u203rhi35] DATASRC

```
|  
| ACTION: DATASRC |  
|
```

1/ **datasrc**> [Creates a new DataSource]

```
name: rrefr1_remise_ds_01  
jndi_name: /jdbc/mydb  
provider: rrefr1_remise_oracle_01  
dbtype: oracle  
statement_cache_size: 30  
prefix: None  
desc: None
```

2/ **datasrc/scope**> [A scope refers to a Name Space where

```
cell: false  
node: WAS7RC1_Node4  
server: rrefr1_remise_jvm_04  
cluster: None
```

3/ **datasrc/jaas**> [Specify an allowed credential to the

```
name: rrefr1_remise_jaas_04  
user: mysuser  
password: mypass  
create: true  
desc: None
```

4/ **datasrc/connection\_pool/connection\_timeout**>

value: 180

5/ **datasrc/connection\_pool/max\_connections**>

value: 10

( Exit:0 Up:+)

Choice ?

# Les customs xml sont aussi accessibles aux Menus

ex :

```
kikupd --kikarc myc.xml
```

[u203rhi35] MYTAG1

```
|_____|
| MYTAG1 |
|_____|
```

1/ mytag1>

2/ Action: /mytag1/chgports [Change all the JVM (TM) ports]  
starting\_port: 20030

3/ Action: /mytag1/jdbc [Creates one JDBC provider]  
name: rrefr1\_remise\_oracle\_01  
path: /WebSphere/was/7.0/wascore/driver/oracle/11.2.0.3.0  
prefix: None  
desc: None

4/ New mytag1/mytag2>

5/ mytag1/mytag2>

6) Show custom file

7) Print

8) Check All

9) Save

( Exit:0)

Choice ? █

# Accessibilité des Actions

Les Actions sont donc:

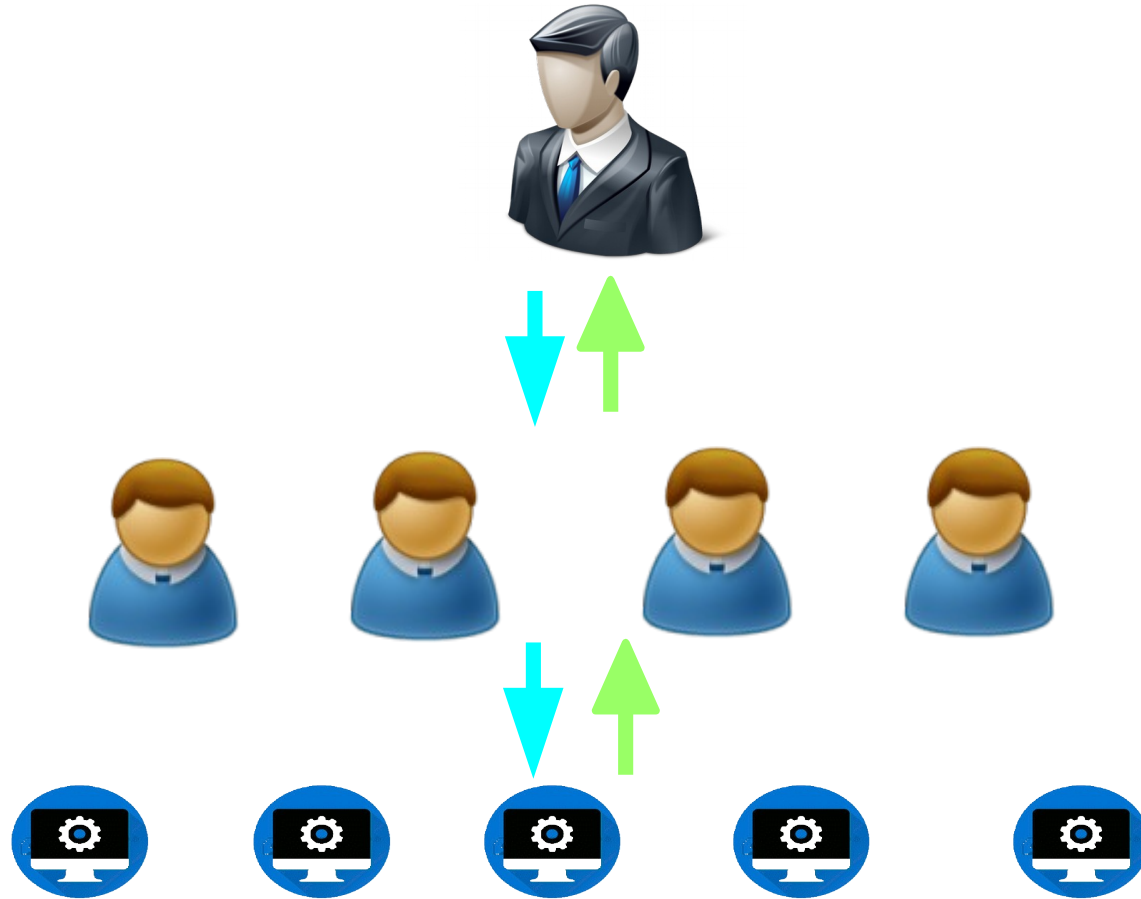
- 100% disponibles en mode commande
- 100% disponibles en mode Menu
- 100% disponibles en mode Web accès centralisé
- Nous proposons donc 3 couches d'interaction là où les autres en proposent qu'une

# On passe du bastion



De l'ingénieur qui détient un capital de commande.

# A la visibilité/reproductibilité pour tous



A un socle d'actions lisibles de tous

# TradeMarks

- "IBM", "AIX", "WebSphere", "WebSphere Application Server", "WAS", "IBM MQ Series", "MQ Series", "IBM WebSphere MQ", "WebSphere MQ", "WebSphere MQ Series", "IBM HTTP Server", "IHS" are registered trademarks of International Business Machines Corporation.
- "WebLogic Server" and "Java" are registered trademarks of Oracle and/or its affiliates.
- "Apache Http Server" and "Tomcat" are trademarks of the Apache Software Foundation.
- "Apache Tomcat" and "Tomcat" are trademarks of the Apache Software Foundation.
- "JBoss" and "JBoss AS" are trademarks or registered trademarks of Red Hat.