Présentation AUTOSYS

- 1. Architecture
- 2. Process, Services
- 3. Normes
- 4. Commandes Autosys
- 5. Arborescence, Users
- 6. Ordonnancement des traitements
- 7. Suivi des traitements

Architecture

Définition.

 Le produit a été renommé en CA Workload Automation Autosys Edition r11.3

AutoSys est un ordonnanceur d'exploitation multiplates-formes qui permet:

- D'ordonnancer et planifier les travaux de production.
- De surveiller le déroulement des tâches.
- De déclencher des actions.
- De produire des rapports d'exécution.
- De sécuriser l'exploitation

Fonctionnement.

- Mode événementiel.
- Balayage cyclique de la base d'événements.
- Déclenchement d'actions associées aux événements.
- Positionnement des statuts associés aux actions exécutées

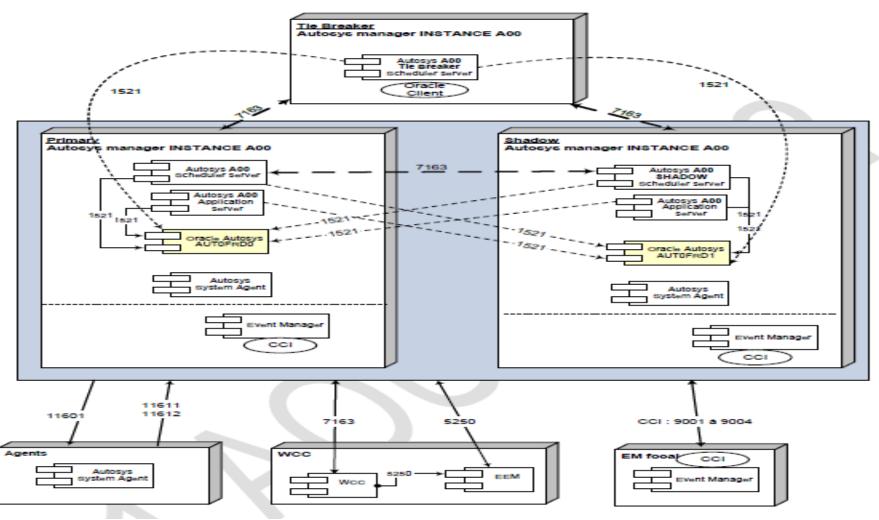
2

Architecture

- Qu'est ce qu'un ordonnanceur événementiel
- Les serveurs (Manager, WCC, Tie breacker)
- La base Oracle
- BCC: Base Connaissance Collaborative

http://formicaio.itp.echonet/wiki/index.php/Accueil

Schémas d'architecture



Les Composants

Le Scheduler .

- Interprète et prend en charge tous les événements AutoSys lus dans la base de données.
- Ordonne à l'agent distant la prise en charge des traitements.
- Processus event_demon.

L'EventServer (AEDB=base d'événements).

- ORACLE(9,10,11),SYBASE(12.5,15),Microsoft SQLSERVEUR (2005,2008).
- Contient l'ensemble des définitions AutoSys, les statuts et l'historique.

L'applicationServer.

- Il fait l'interface avec la base de données pour les agents, les commandes et les requêtes des GUI.
- Toujours connecté à la AEDB.
- Processus as_server

Le client.

Utilitaires ligne de commande.

L'agent distant (Remote Agent).

- Exécute les traitements et retourne le statut des Jobs Autosys dont il a la charge.
- Processus auto_remote (11.0) ou cybAgent (11.3)

Les Composants

WCC: Workload Command Center

- Interface Web pour la conception et le suivi des chaînes d'ordonnancement.
- Principal accès aux instances Autosys d'un ou n Serveurs Autosys (et CA-7)

Business Object (11.0)

- Permet grace à des collectors spécifiques de ramener les historiques d'éxécution des jobs et de faire ainsi des rapports détaillés sur les events des jobs et les statistiques afférentes.
- Associé à une Base de données type Sql (Datawarehouse) et Mysql, il peut résider sur la même machine que WCC ou sur un Reporting Server à part.

CCS: Common communication service

Composant contenant cci,l'Event console, les messages, les messages actions.

• SSA: Secure Socket Adapter

Service de communication entre Scheduler, Agent, Application server et WCC.
 Port7163. Gère le SSL. Le processus est csampmux.

Command Sponsor

 Bibliothèque du serveur Autosys sollicitée par l'interface WCC pour exécuter des commandes en ligne. Géré par le service http igateway (port http5250).

Les Composants

CCI: Common communication interface

Protocole CA, permet de communiquer avec d'autres scheduler d'autres OS(Ca-7) ou d'autres composants (Event Console, Messages, Actions).

Jaws: Job Autosys Workload Service Manager

- Client Java, Permet de faire un suivi temps réel sur le déroulement des jobs autosys, des analyses en se basant sur les historiques d'éxecution, Alerte prédictive, chemin critique, Alerte si un job ne démarre pas....
- II accède en READ ONLY sur Autosys.

Eem : Embedded Entitlement Management

eTrust Embedded Management (equivalent IAM=Identity Access Management).
 Permet de sécuriser les accès des fonctions autosys au travers de droits sur des users et des roles. Géré par le service http igateway (port http 5250).

Universal Agent

- AS/400 (autosys 11.0), VMS, Linux z/Os.
- Remarque : l'agent autosys System 11.3 existe sur AS/400.

Adapters ERP.

SAP, Peoplesoft, Oracle application.

L'instance AutoSys (variable \$AUTOSERV)

OBLIGATOIRE:

- Un Scheduler.
- Un Event Server (Database).
- Un Application Server.
- Un agent (local ou distant)
- On peut avoir plusieurs instances Autosys sur une même machine.
- Sur Unix, il est préférable que chaque instance soit géré par un compte
 Ex : « P01 », Scheduler et Application server lancé par « autop01 »
- Le processus agent est lancé par root. Il est unique.

OPTIONNEL ET INDEPENDANT:

- Une base secondaire (Dual)
- Un Scheduler & AS secondaires (H.A)
- D'autres agents (distants)

La variable '\$AUTOSERV' ou %AUTOSERV% nomme une instance

Exemple:

rgtprd1@parvl7119481:PRD:/home/rgtprd1# echo \$AUTOSERV PF3

Les variables Autosys

La variable AUTOSYS contient le produit (Procédures d'installation, bibliothèques d'API et d'exemples, les exécutables Autosys, ...) :

root@parvl7119481:PRD:/root# echo \$AUTOSYS /apps/waae/11.3/autosys

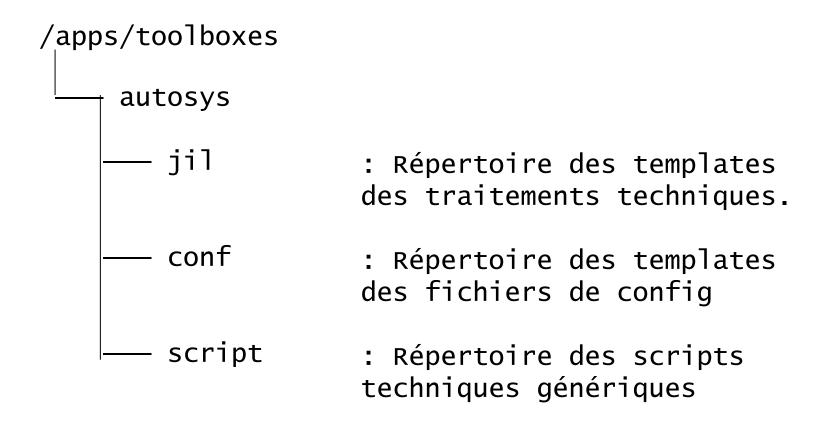
La variable AUTOUSER contient les fichiers de configuration et les fichiers logs : root@parvl7119481:PRD:/root# echo \$AUTOUSER /apps/waae/11.3/autouser.PF3

La variable AUTOSERV contient le nom de l'instance Autosys : root@parvl7119481:PRD:/root# echo \$AUTOSERV PF3

La variable ORACLE_HOME contient les fichiers base de données nécessaires à la connexion des processus servcers :

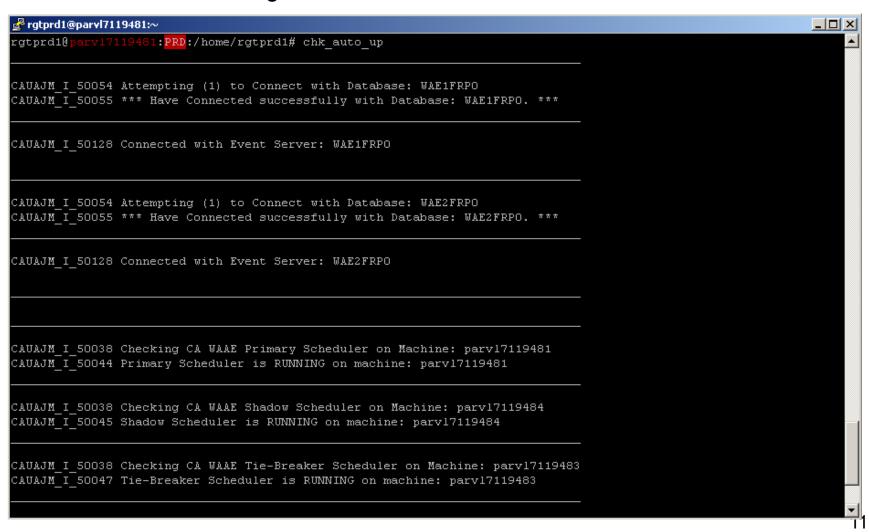
root@parvl7119481:PRD:/root# echo \$ORACLE_HOME /apps/oracle/11.2.0.1

L'arborescence toolbox (manager):



Vérification de l'environnement

Vérification du Manager à l'aide des commandes suivantes :



Vérification de l'environnement

CA Services Status Report Component Name Pid Status CA-CCI Server 8406 running CA-CCI Remote Server 8409 running CA-CCI Clean Up 8407 running CA-CCI Legacy Proxy 8408 running WAAE Application Server (PF3) 13939 running WAAE Scheduler (PF3) 14026 running
CA-CCI Server 8406 running CA-CCI Remote Server 8409 running CA-CCI Clean Up 8407 running CA-CCI Legacy Proxy 8408 running WAAE Application Server (PF3) 13939 running
CA-CCI Server 8406 running CA-CCI Remote Server 8409 running CA-CCI Clean Up 8407 running CA-CCI Legacy Proxy 8408 running WAAE Application Server (PF3) 13939 running
CA-CCI Clean Up 8407 running CA-CCI Legacy Proxy 8408 running WAAE Application Server (PF3) 13939 running
CA-CCI Legacy Proxy 8408 running WAAE Application Server (PF3) 13939 running
WAAE Application Server (PF3) 13939 running
WAAE Scheduler (PF3) 14026 running
·····-
WAAE Agent (parv17119481) 13681 running
CA-Message Queuing Service 5563 running
Systems Performance LiteAgent - not active
Atech Service Control Mgr 8300 running
Atech Object Request Broker 8434 running
Atech SNMP Administrator 8476 running
Atech scriptAgt 8523 running
Atech caiLogA2 8540 running
Atech caiUxsA2 8505 running
Postgres 10235 running
CA-Doc Help 11754 running
CA-Calendar Server 11814 running
CA-EM Routing Server 12896 running
CA-Event Manager 26040 running
CA-Event SAF Client 26070 running
CA-Event Syslog Reader 26072 running
CA-Event Total Action process(es) - 1 running
Caiopr Action process(es) - 1 running
CA-Star Server 13492 running
CA-SNMP Trap Multiplexer 13626 running
CA-SNMP Trap Manager 13636 running root@parv17119481:PRD:/root# [

Vérification de l'environnement

Vérifier que l'agent est déclaré sur le Manager :

root@parvl7119481:PRD:/root# autoping -m parva7117720

CAUAJM_I_50023 AutoPinging Machine [parva7117720] CAUAJM_I_50025 AutoPing WAS SUCCESSFUL.

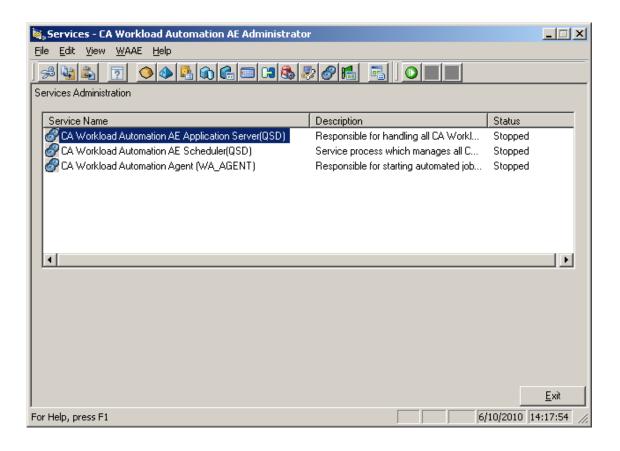
Ou

Process & Services

Sur l'agent, vérifier si Autosys est installé :

- Le Process cybAgent sur un serveur Unix ou Linux
 - Ps –ef | grep cybAgent
- Les services Windows
 - CA Workload Automation Agent
- Le répertoire Autosys
 - /apps/waae
 - C:\apps\waae

Les services Windows



CA iTechnology iGateway 4.5	Allows iSpo Started	Automatic	Local System
CA License Client	CA License	Manual	Local System
CA Workload Automation AE Application Server (QSD)	Allow progr	Manual	Local System
CA Workload Automation AE Scheduler (QSD)	Interprets	Manual	Local System
CA Workload Automation Agent (WA_AGENT)	Executes r	Manual	Local System
Central Management Server	Provides sc	Manual	Local System
ClipBook 🖏	Enables Cli	Disabled	Local System
COM+ Event System	Supports S Started	Automatic	Local System

Toolbox AUTOSYS

Accès à la toolbox sur le Manager (/apps/toolboxes/autosys/script)

```
root@parvl7119481:/apps/toolboxes/autosys/script
-rwxr-xr-x 1 1190505 autosys 13105 Apr 15 13:39 autosys check norme.ksh
        v17119481:PRD:/apps/toolboxes/autosys/script# autosys.ksh
###################
  Toolbox AUTOSYS #
###################
Répertoire applicatif :
autosys rep appli.ksh :
                                          Création des répertoires d'une application
Modification du prefixe application dans un fichier de définition jil :
autosys jil prefixe.ksh :
                                          Job et box
Export des définitions d'une application dans le répertoire jil/save :
autosys exportdef variable.ksh :
                                          Variable
autosys exportdef cal.ksh :
                                          Calandrier
autosys exportdef job.ksh :
                                          Job
autosys exportdef machine.ksh :
                                          Machine
autosys exportdef ressource.ksh :
                                          Ressource
AEL :
autosys ael.ksh :
                                          Export CSV pour le fichier excel qui est dans la BCC
Check :
autosys forecast.ksh :
                                          Planification
autosys calendrier.ksh :
                                          Expiration des calendriers
autosys check norme.ksh :
                                          Vérification des normes
Reporting :
autosys reporting plan.ksh :
                                          Format csv
root@parv17119481:PRD:/apps/toolboxes/autosys/script#
```

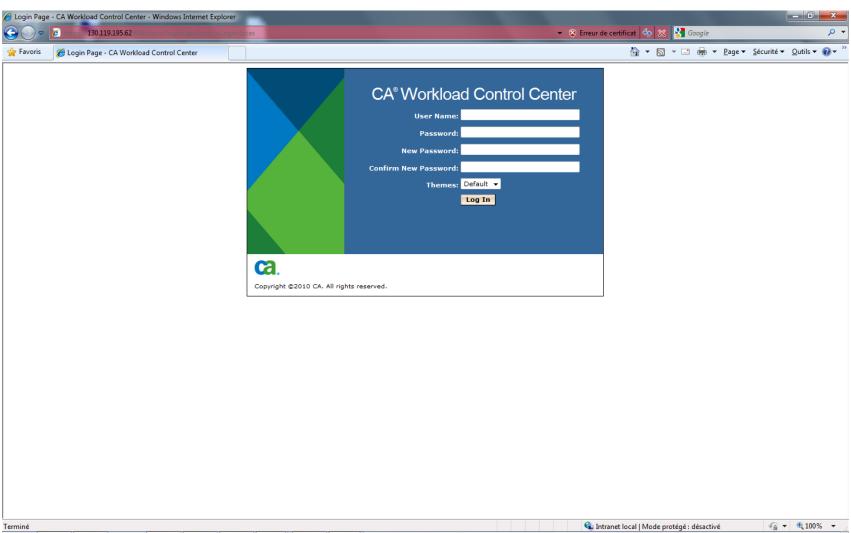
Toolbox AUTOSYS

JOB	DESCRIPTION
autosys_calendrier.ksh	Display de l'utilisation des calendriers d'une application
autosys_check_dependencies.ksh	Display Missing dependencies d'une application
autosys_check_late.ksh	Affiche les jobs qui sont en retard
autosys_check_norme.ksh	Vérification du nommage des jobs applicatif
autosys_exportdef_cal.ksh	Export de la définition des calendriers d'une application
autosys_exportdef_job.ksh	Export de la définition des jobs d'une application
autosys_exportdef_machine.ksh	Export de la définition des machines d'une application
autosys_exportdef_ressource.ksh	Export de la définition des ressources d'une application
autosys_exportdef_variable.ksh	Export de la définition des variables d'une application
autosys_forecast.ksh	Display des traitements d'une application qui sont planifiés à une date
autosys_jil_prefixe.ksh	Modification du prefix des objets applicatif dans un fichier JIL
autosys_liste_job_appli.ksh	Display de la liste des jobs d'une application
autosys_reporting_plan.ksh	Display du reporting des traitements d'une application entre J 15:00 et J+1 15:00

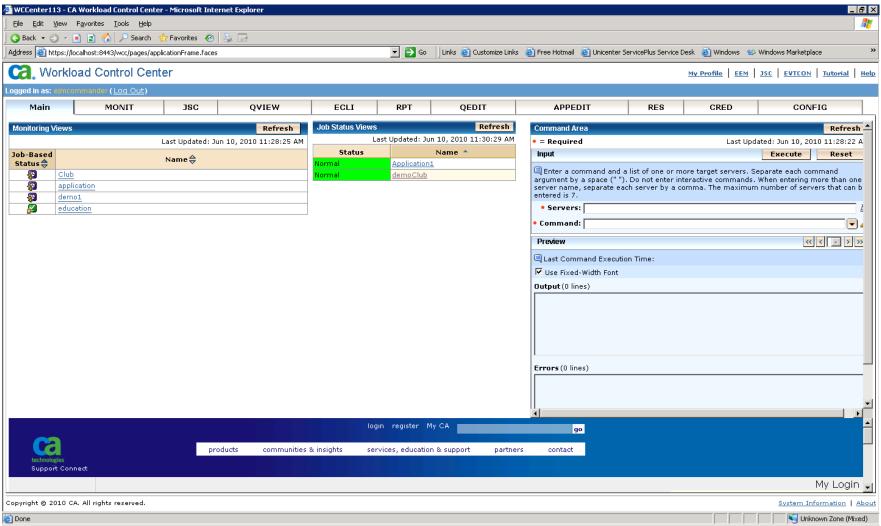
Normes et Standards

- Définition des Boxes (Niveau 0 et 1)
- Définition des Jobs (Cycliques ou non)
- Les calendriers (standard, extended)
- Les alias machines
- Les variables globales
- Règles de planification

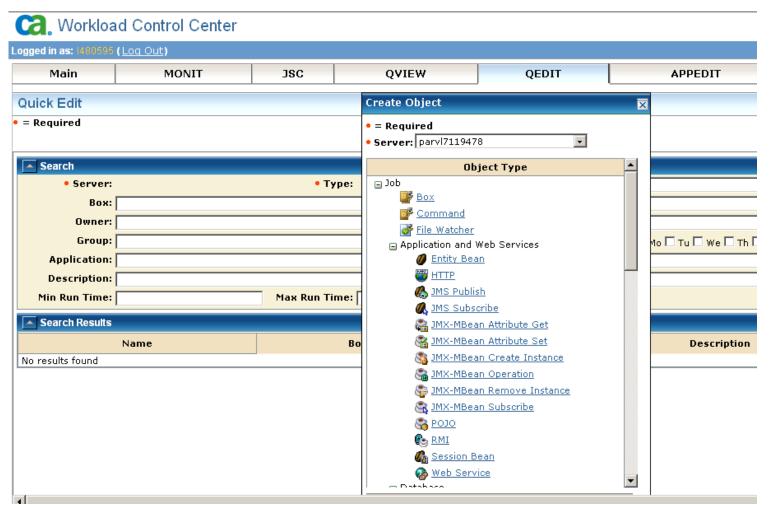
WCC: http://server:9090/wcc



WCC: http://server:9090/wcc



Définition des jobs par la WCC



21

Properties Com			
Expand All Collapse All			
Name	Value		
□ Primary			
• Name			
Description			
Send to machine	localhost		
Condition			
Resources			
Owner			
Вох			
Group			
Application			

Name	JIL	nom du job	
Description	description	champ de saisie libre	
Send to Machine	machine	machine cible pour le job	
Condition	condition	dépendances, c'est-à-dire le ou les jobs ou boxes prédécesseurs	
Resources	resources	les ressources logiques en dépendance	
Owner	owner	user de soumission. Sur Windows, noter user@domain	
Box	box_name	box d'appartenance	
Group	group	groupe logique pour organisation et construction des vues	
Application	application	groupe logique pour organisation et construction des vues	

□ Command	
• Command	
Shell	
Interactive	
Heartbeat interval	
Standard input file	
Blob input	
Standard output file	
Standard error file	
Agent job class	

Command	command	ligne de commande et paramètres	
Shell	shell	spécifie un Shell, comme /bin/ksh.	
Interactive	interactive	y ou n . Uniquement Windows.	
Heartbeat	heartbeat_interval	gestion de l'alarme MISSING_HEARTBEAT. Le job doit alors	
interval		donner des 'signes de vie' toutes les n minutes. L'adapter SAP	
		le fait de base. Sur Unix, le hearbeat est le signal SIGUSR2.	
Standard Input file	std_input_file	Fichier de réponses pour les scripts interactifs.	
Blob input	blob_input	Binary Large Object. Peut contenir de l'ascii ou du binaire. Utile pour les jobs de type Web Services pour stocker le fichier XML en input.	
Standard output file	std_out_file	redirection de l'affichage de la commande vers un log géré par l'agent.	
Standard error file	std_err_file	redirection des erreurs de la commande vers un log géré par l'agent.	
Agent job class	job_class	pour SAP	

Environment Environment		
Environment variables		
Profile		
Ulimit		
File system check		

Environment variables	envvars	ajoute des variables pour l'exécution du job. PATH=, PARAM1=3
Profile	profile	spécifie un profile à charger avant de lancer la commande. Exemple /home/oracle/.profile
Ulimit	ulimit	id=soft_value,hard_value, Exemple c="100,200", s="250,300", t="unlimited,4000", m="3332, unlimited" c means the core file size in kilobytes d means the data segment size in kilobytes f means the maximum file size in kilobytes m means the process virtual size in kilobytes n means the number of files s means the stack size in kilobytes t means the CPU time in seconds
File system check	chk_files	vérifie l'espace libre avant le lancement et si le file système existe. Exemple : /data 20000 /log 1000 Génère une alarme RESOURCE.

=	Termination	
	Terminate containing box on job failure	
	Terminate on failure of containing box	
	Minutes to wait before terminating	
	Hours after successful completion to delete job	
	Max exit code for SUCCESS	
	Success exit codes	
	Failure exit codes	

Terminate containing box	box_terminator	si le job est FAILURE, la box est alors stoppée. Il en découle que les jobs restant ACTIVATED deviennent INACTIVE.
Terminate on failure of	job_terminator	si la box est stoppée (KILLJOB), le job reçoit un KILLJOB.
Minutes to wait before terminating	term_run_time	n minutes après le démarrage, le job reçoit un KILLJOB par le scheduler. Utile pour les File triggers
Hours after successful completion	auto_delete	auto suppression de la base n heures après le SUCCESS.
Max exit code for success	max_exit_success	seuil de code retour au-dessus duquel le job est FAILURE
Ou		
Success exit codes	success_codes	liste et/ou plages de code retour rendant le job SUCCESS. Exemple 3,6,9,20-30
Failure exit codes	fail_codes	liste et/ou plages de code retour rendant le job FAILURE. Exemple 3,6,9,20-30

□ Schedule	
Times to restart job after failure	
Auto hold job when box starts	
Average run time	
Priority	
Job load	
Date/Time conditions	
Time zone	
Run window	

Times to restart job after	n_retryrs	nombre de resoumissions automatiques après un FAILURE
Auto Hold	auto_hold	valable pour les jobs ou une box contenue dans une box. Le job passe ON_HOLD dès que la box passe RUNNING.
Average run time	avg_runtime	pour information
Priority		pour le load balancing
Job Load	job_load	pour le load balancing. Spécifie un poids.
Date/Time conditions	date_condutions	active la planification.
Timezone	timezone	
Run Window	run_window	fenêtre horaire de démarrage possible Syntaxe: "08:00 – 18:00"

□ Days	Run calendar 🔻
Run calendar	
Exclude calendar	
☐ Time	Times of day
Times of day	
Must start times	
Must complete times	

Run days	days_of_week	run days permet de cocher un ou plusieurs jours de la semaine
		
Run calendar	run_calendar	nom du calendrier
Exclude calendar	exclude_calendar	annule la planification du run calendar si un jour
		est commun
Times of days ou Minutes		
afer hours.		
Times of day	start_times	heure(s) de démarrage. Exemple
		08:00,12:00,20:00
Minutes after each hours	start_mins	00,10,20,30,40,50
Must start times, Must	must_start_times	émission d'alarmes
Complete times	must comp times	Par exemple pour ouverture TP
		start_times : " 06:30"
		must_start_times: "08:00"
		Pour un cyclique
		start_times: " 08:30, 10:00,16:00"
		must_start_times: "08:40,10:10,16:10"
		must_complete_times: "09:00, 10:30,16:30"
		En relatif
		start_times : " 08:30, 10:00,16:00"
		must start times: "+10"
		must_complete_times: "+30"
		Sur plus de 24 heures
		start_times: "20:00"
		must_complete_times:"70:00" (2
		iours plus tard à 22h)

27

□ Notification		
□ Alarms		
Minimum run time (minutes)		
Maximum run time (minutes)		
Send alarm on failure	V	
☐ Notification Services		
Notification	Do not notify	
ID		
Message		
nessage		

Minimum run time	min_run_alarm	alarme si le job n'est pas RUNNING assez Iongtemps
Maximum run time	max_run_alarm	alarme si le job est RUNNING depuis trop longtemps
Send alarm on failure	alarm_if_fail	en cas de FAILURE, créer une alarme.
Notification		intégration avec Notification de CA NSM r11.x
Service Desk		intégration avec CA Service Desk

28

Définition d'une box, champs spécifiques

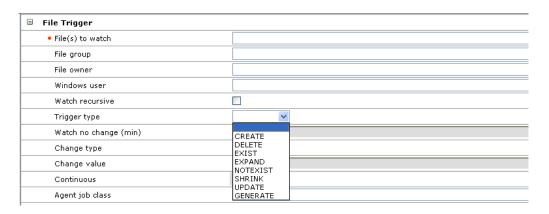


Ces 2 champs servent à changer le comportement par défaut d'une box:

- -Success: conditions qui rendent SUCCESS une box. Par exemple le SUCCESS d'un job parmi 3. Le code retour 5 d'un job.
- -Failure: Souvent les conditions opposées
- -En jil : box_success ou box_failure
- -Le champs owner n'a pas d'utilité pour la soumission des jobs. Il n'y a pas d'héritage vers les jobs.

<u>Définition d'un File Trigger, champs spécifiques</u>

job_type: FT



File group	watch_file_groupname	groupe Unix	
File owner	watch_file_owner	propriétaire du fichier	
Minimum file size		taille minimale	
Watch recursive	watch_file_recursive	détecte le fichier dans les sous-répertoires.	
Trigger type		CREATE: équivalent à un file watcher. Attend la création.	
		UPDATE: attend l'update du fichier. Par défaut le fichier doit déjà	
		exister.	
		EXIST: ok si le fichier existe	
		EXPAND: ok si le fichier croit de la bonne valeur en pourcentage,	
		delta ou taille.	
		SHRINK: réduction de taille.	
		NOTEXIST: ok si le fichier n'existe pas	
		DELETE: attend la suppression d'un fichier	
Watch no change	watch_no_change	60 secondes par défaut. Le fichier doit rester inchangé pendant n	
		secondes.	
Change type	watch_file_change_type	DELTA, PERCENT ou SIZE	
Change value	watch_file_change_value	valeur pour DELTA, PERCENT ou SIZE	
Continuous	continuous	le file trigger reste RUNNING et envoie un message à chaque	
		update.	

File Trigger:

- Type CREATE :
 - Lorsque le FT démarre :
 - Si le fichier existe déjà, le FT pass SUCCESS
 - Si le fichier n'existe pas, il attend sa création, à moins qu'un « Term_run_time » soit positionné
- Type UPDATE :
 - Lorsque le FT démarre :
 - Si le fichier existe déjà, le FT attend l'update (timestamp)
 - Si le fichier n'existe pas, il passe à failure
- Type GENERATE :
 - Lorsque le FT démarre :
 - Si le fichier existe déjà, le FT attend l'update
 - Si le fichier n'existe pas, le FT attend sa création

<u>File Trigger:</u>

- Attribut continuous : y
 - Le FT reste en running. A chaque bon critère, un message est envoyé à la console NSM Event Manager avec le nom du fichier
- Attribute recursive : Y
 - Le FT scanne aussi les sous-répertoires
 - Les wildcards sont autorisés dans le chemin
 - Exemple :
 - Watch_file: /cft/data*/fic????.txt
 - Le FT passe RUNNING
 - Un programme externe crée le répertoire « /cft/data0104213 » et le fichier « fic1234.txt » dans ce répertoire
 - → Le FT passe SUCCESS

Relations Box/Job

- Un job ne démarre que si la box est Running
- A chaque éxecution de la box, un job ne s'éxecute qu'une fois.
- Une Box restera en status Running, jusqu'à ce que tous ces jobs soient finis.
- Le statut d'une box est Success, si tous ses jobs ont le status Success.
- Le statut d'une box est Failure, si tous ses jobs se sont exécutés et un ou n Jobs ont le status Failure.
- Lorsqu'une box est arrêtée, ses jobs en attente sont désactivés, ceux en Running continuent.
- Une box peut être cyclique, rendant donc cyclique ses jobs

Les statuts d'une box en exécution

-RUNNING →SUCCESS/FAILURE

Les statuts d'un job seul

-STARTING →RUNNING →SUCCESS/FAILAURE

Les statuts d'un job contenu dans une box

-ACTIVATED →STARTING →RUNNING →SUCCESS/FAILURE

Les conditions de dépendance

Définition.

- Les conditions de dépendance permettent de définir l'ordre d'exécution des jobs.
- L'exécution d'un traitement est dépendante de conditions de type :
 - Prédécesseur/successeur.
 - Temporel (date/heure).
 - File d'attente.
 - Ressource virtuelle.

Les conditions de type événementiel.

- Utilisation de :
 - Statut du ou des jobs prédécesseurs.
 - code retour (0 à 255) du ou des jobs prédécesseurs.
 - Valeurs de variables globales.

Les conditions de dépendance

- Les conditions de type événementiel (suite).
 - Dépendances sur le statut

Syntaxe : statut(nom du job)

Liste des statuts possibles :

• **S**UCCESS : Exécuté avec succès.

FAILURE : Exécuté avec erreur.

• **D**ONE : Exécuté (success/failure/terminated)

TERMINATED : Terminé volontairement(kill)

NOTRUNNING: Tout sauf RUNNING

– Exemple:

SUCCESS(Backup) ou S(Backup) ou s(Backup)

Les conditions de dépendance

Exemples:

Le job START_ORA démarre si JOB1 et JOB2 sont finis et l'un des deux est réussi et la variable TP est égale à OUVERT

→ DONE(JOB1) AND DONE(JOB2) AND (SUCCESS(JOB1) OR SUCCESS(JOB2)) AND VALUE(TP)=OUVERT

Ou

→ d(JOB1)&d(JOB2)&(s(JOB1)|s(JOB2))&v(TP)=OUVERT

Règle de définition

- Pas de job cyclique dans une box non cyclique.
- La fréquence de planification est associé à un calendrier extended.
- Pas de lien sur les boxes de niveau 0 (sauf sur une box cyclique qui à un lien pour bloquer le rerun de la box en cas d'erreur).
- Pas de lien sur des jobs qui ne sont pas dans la même box.
- Pas de planification différente dans une box.

Règle de définition

Les conditions de fin

Pour qu'une box se termine même si des jobs sont encore en statut ACTIVATED, il est possible d'utiliser les paramètres : run_window, term_run_times, box_terminator, job_terminator, box_success.

Pour	rappel:
	□ run_window : dès que l'heure de démarrage est hors de la fenêtre, le job/box passe en statut INĀCTIVE ce qui peut permettre la fin de la box parente.
	□ term_run_time : Kill le job/box au bout de n minutes. Va donc permettre la replanification automatique d'une box et le départ des successeurs sur DONE ou FAILURE pour un chemin dégradé par exemple. Utile pour un File Trigger: attente d'un fichier pendant n minutes.
	box_terminator: sur un job ou sous-box. Dès qu'il est FAILURE, la box s'arrête. Les jobs qui étaient ACTIVATED passent INACTIVE. Va donc permettre la replanification automatique d'une box et le départ des successeurs sur DONE ou FAILURE pour un chemin dégradé par exemple.
	□ Job_terminator : si on KILL la box, Autosys va faire un KILLJOB en cascade des jobs de la box sur lesquels on aura positionné. job_termintor
	□ Box_success, box_failure: condition de SUCCESS ou FAILURE d'une box. Change le comportement par défaut d'une box pour forcer le SUCCESS ou FAILURE selon ses propres critères.

Règles de nommage applicatif

- Le nommage des objets autosys (box, job, calendrier, ...) se trouve dans le document des normes AUTOSYS
- http://formicaio.itp.echonet/wiki/index.php/Alerting/Scheduling
- Tous les objets sont préfixés par :

⅓ IAAAECC

Chaîne	Description							
I	I pour les jobs ordonnancés par IPS							
	ou							
	B pour les jobs ordonnancés par BP2I							
AAA	Code application							
	La liste normalisée des codes application se trouve dans le libellé							
	court de							
	REFI.							
	https://refitp.group.echonet/refitp-authentification-web/index.jsp							
E	Code environnement sur 1 caractère (ex: P pour production ,R pour							
	recette,)							
CC	Numéro de version de l'application sur 2 chiffres							
	Ce numéro est utilisé pour avoir différentes image d'une application							
	dans un même environnement.							

Règles de nommage applicatif

Exemple:

↑ IAYSR01-E-ALLDAYS

```
⅓ Box:
↑ IAYSR01-Q1BOX-AUTOSYS-EXPORT
⅓ Job:

↓ IAYSR01-Q2EXP01-AUTOSYS-EXPORT-JOB

↓ IAYSR01-Q2EXP01-AUTOSYS-EXPORT-VARIABLE

    ∀ Variable :

¾ IAYSR01-ACTIVE
↑ Machine:
∛ IAYSR01-AGENT

    ★ Calendrier :
```

Règles de nommage applicatif

Box niveau 0:

insert_job: IAAAECC-F0BOX-VAR0 job_type: BOX

owner: user

date_conditions: 1

run_calendar: IAAAECC-T-VAR

start_times: "15:00"

condition: v(IAAAECC -ACTIVE) = "YES"& v(IAAAECC -LIVRAISON) = "NO"

Box niveau 1:

insert_job: IAAAECC-F1BOX-VAR0-VAR1 job_type: BOX

box name: IAAAECC-F0BOX-VAR0

owner: user

condition: v(IAAAECC-ACTIVE) = "YES"& v(IAAAECC-LIVRAISON) = "NO"

<u>Job :</u>

insert job: IAAAECC-F2TTT-VAR0-VAR1-VAR2 job type: CMD

box name: IAAAECC-F1BOX-VAR0-VAR1

command: "/apps/exploit/autosys/AAAECC/script/ IAAAECC-F2TTT-VAR0-VAR1-VAR2.ksh"

profile: "/apps/exploit/autosys/AAAECC/conf/WAAE_profile"

machine: IAAAECC-VAR

owner: user

condition: v(IAAAECC-ACTIVE) = "YES"

std_out_file: "/apps/exploit/autosys/AAAECC/log/\${AUTO_JOB_NAME}_\${AUTORUN}.log" std_err_file: "/apps/exploit/autosys/AAAECC/log/\${AUTO_JOB_NAME}_\${AUTORUN}.err"

Les variables

- IAAAECC-ACTIVE : Variable qui permet de bloquer une application. Cette variable doit être définie dans les jobs et les boxes.
- Cette variable est initialisé avec la valeur YES.
- IAAAECC-LIVRAISON : Variable qui permet de bloquer une box lors d'une livraison. Cette variable doit être définie dans les boxes.
- Cette variable est initialisée avec la valeur NO.
- Exemple:
- - Variables globales
- \$ autorep -G IWAED01%

• Global Name	Value	Last Changed	
•			
• IWAED01-ACT	TIVE YES	2012/06/11 13:08	:39
• IWAED01-LIV	RAISON NO	2012/06/11 13:08	:46

Les variables

- Box de niveau 0

Name	IWAED01-Q1BOX-BATCH-START
Description	.::
-Box de niveau 1	V(IWAED01-ACTIVE) = "YES" & V(IWAED01-LIVRAISON) = "NO" .::
BOX de III veau 1	
Name	IWAED01-Q0BOX-BATCH
Description	
Condition	v(IWAED01-ACTIVE) = "YES" & v(IWAED01-LIVRAISON) = "NO" ∴
-Job	
Name	IWAED01-Q2WAE01-BATCH-START-BATCH
Description	.::
 Send to machine 	IWAED01-MANAGER
Condition	v(IWAED01-ACTIVE) = "YES"

L'arborescence et les Users

- Les users :
 - Création du user (Manager et Agent)
 - Fichier .profile (Manager et Agent)
- L'arborescence
 - Sur le Manager
 - Sur l'agent

Création du user UNIX

root@:/# mkgroup -'A' id='430' aaaeee1

root@:/# mkuser id='430' gecos='User APPLICATION aaa' pgrp='aaaeee1' home='/home/ aaaeee1 ' shell='/bin/ksh' aaaeee1

- Faire une demande de justification de compte du user créé (Envoyer un mail à PARIS ITP ITPS MEOA ACCES SERVEUR).
- Détailler dans la BCC :

http://formicaio.itp.echonet/wiki/index.php/Alerting/Scheduling

Création du user Windows

- Après création du user Windows, il faut que celui-ci soit déclaré sur le Manager Autosys
- Lancer la commande autosys secure :

rgtprd1@parvl7119481:PRD:/home/rgtprd1# autosys_secure

CA WAAE Security Utility

Please select from the following options:

- [1] Revert to NATIVE instance security.
- [2] Manage CA EEM server settings.
- [3] Change database password.
- [4] Change remote authentication method.
- [5] Manage user@host users.
- [6] Get encrypted password.
- [0] Exit CA WAAE Security Utility.

>

Création du user Windows

> 5

Manage user@host users

Please select from the following options:

- [1] Create user@host or Domain password.
- [2] Change user@host or Domain password.
- [3] Delete user@host or Domain password.
- [4] Show all user@host users.
- [9] Exit from "Manage user@host users" menu.
- [0] Exit CA WAAE Security Utility.

> 1

CAUAJM_I_60207 Create an user@host user: Input the user name (or hit enter to cancel): admregent@PARS07118479

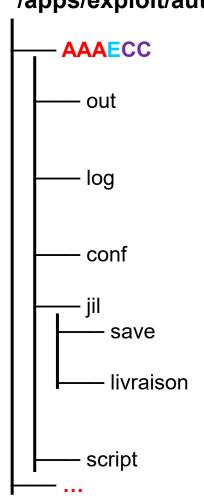
Enter new password:

L'arborescence applicative (manager et agent) :

/apps/exploit/autosys

Ou

/apps/exploit/autosys/ressource_cluster_name



- : Nom de l'application dans AUTOSYS
- : Répertoire des fichiers générés par la toolbox AUTOSYS
- : Répertoire des fichiers log des jobs autosys
- \${AUTO_JOB_NAME}_\${AUTORUN}.log
 \${AUTO_JOB_NAME}_\${AUTORUN}.err
- : Répertoire des fichiers de config de
- l'application
- : Répertoire de backup des définitions des objets autosys
- : Répertoire de livraison des définitions des objets autosys
- : Répertoire des scripts qui sont ordonnancés dans les jobs autosys. Le nom des fichiers est au format JORNAME, ksh

L'arborescence applicative

Création de l'arborescence applicative.

Exemple: application AAAQ01

- Copier le script du manager autosys
 (/apps/toolboxes/autosys/script/autosys_rep_appli.ksh)
 sur le serveur agent.
- Lancer le script : autosys_rep_appli.ksh -a AAAQ01 -t a
 - -a = application au format AAAECC
 - -t = A pour agent ou M pour manager
 - -i = Instance autosys

L'arborescence applicative (Exemple)

```
root@:/#./autosys rep appli.ksh -a AAAQ01 -t a
# AAAQ01 #
###########
[INFO] : Vérification code application : OK
[INFO]: Vérification format nom instance autosys: OK
[INFO]: Option manager [M] ou agent autosys [A]: OK
[INFO]: Vérification FS /apps/exploit/autosys: OK
[INFO]: Création du répertoire /apps/exploit/autosys/AAAQ01
[INFO] : Création répertoire : OK
[INFO] : Création des sous-répertoires :
[INFO]: Création répertoire /apps/exploit/autosys/AAAQ01/log
[INFO] : Création répertoire : OK
[INFO]: Création répertoire /apps/exploit/autosys/AAAQ01/out
[INFO] : Création répertoire : OK
[INFO]: Création répertoire /apps/exploit/autosys/AAAQ01/conf
[INFO] : Création répertoire : OK
[INFO]: Création répertoire /apps/exploit/autosys/AAAQ01/jil/save
[INFO] : Création répertoire : OK
[INFO]: Création répertoire /apps/exploit/autosys/AAAQ01/jil/livraison
[INFO] : Création répertoire : OK
[INFO]: Création répertoire /apps/exploit/autosys/AAAQ01/script
[INFO] : Création répertoire : OK
[INFO]: Création répertoire /apps/exploit/autosys/AAAQ01/backup
[INFO] : Création répertoire : OK
INFO]: Création du fichier profile /apps/exploit/autosys/AAAQ01/conf/WAAE profile
[INFO]: Création fichier profile: OK
[INFO]: Ajout des droits 755 sur le réperoire /apps/exploit/autosys/AAAQ01
[INFO] : Ajout des droits sur le répertoire : OK
```

50

Ordonnancement des traitements

- 1. Création des alias machine
- 2. Création des variables globales
- 3. Création des calendriers
- 4. Création des boxes et jobs

1 - Création des alias machine

Vérifier si le serveur est déclaré sur le manager

root@parvl7119472:REC-QUA:/root# autoping -m PARS07118480

CAUAJM I 50023 AutoPinging Machine [PARS07118480] CAUAJM I 50025 AutoPing WAS SUCCESSFUL.

ou

root@parvl7119481:PRD:/apps/exploit/autosys/RGTP01/jil/livraison# autorep -M ALL | grep PARS07118476

PARS07118476

1.00 Sys Agent Online

Créer le fichier jil

root@parvl7119481:PRD:/apps/exploit/autosys/RGTP01/jil/livraison# vi creat machine.jil

insert machine: IRGTP01-APPLI

type: v

machine: PARS07118476

Exécuter le jil root@parvl7119481:PRD:/apps/exploit/autosys/RGTP01/jil/livraison# jil < creat_machine.jil

2 - Ajout des variables globales

Ajouter les variables globales.

Création:

sendevent -E SET_GLOBAL -G IAAAQ01-ACTIVE=YES

sendevent -E SET_GLOBAL -G IAAAQ01-LIVRAISON=NO

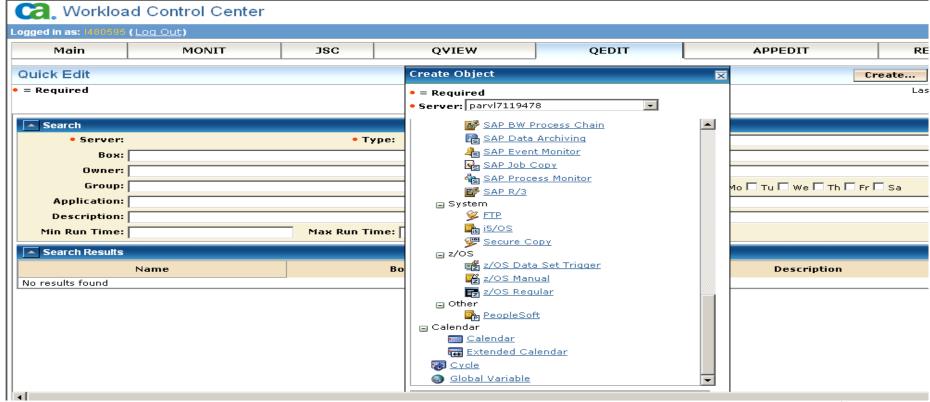
Suppression:

sendevent -E SET_GLOBAL -G IAAAQ01-ACTIVE=DELETE

3 - Création des calendriers

- Il y a 2 manières de procéder :
 - On associe les jours de la semaine directement dans le job.
 - On crée un calendrier.

Il y a les calendriers standards et les calendriers étendus, lesquels sont recalculés automatiquement chaque année. On peut y mettre des conditions, des cycles, des exceptions....



Classification: Internal

Calendrier Standard

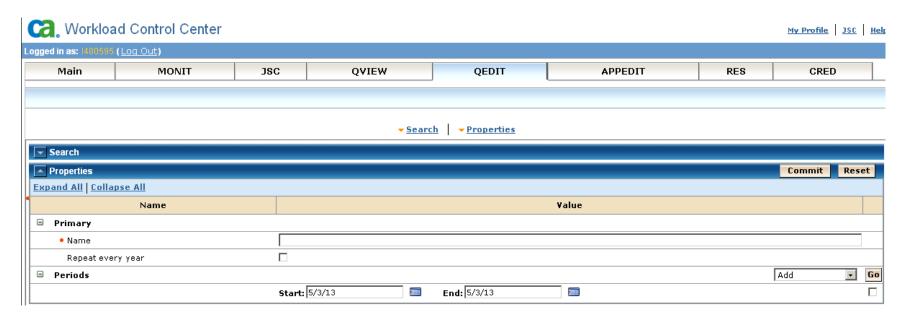
- On utilise les calendriers standards pour les jours fériés ou pour des tâches à la demande. Même s'il y a un « Wizard » d'aide à la sélection de date, les règles ne sont pas sauvegardées.
- Donc pour une planification régulière, on préférera un calendrier étendu.

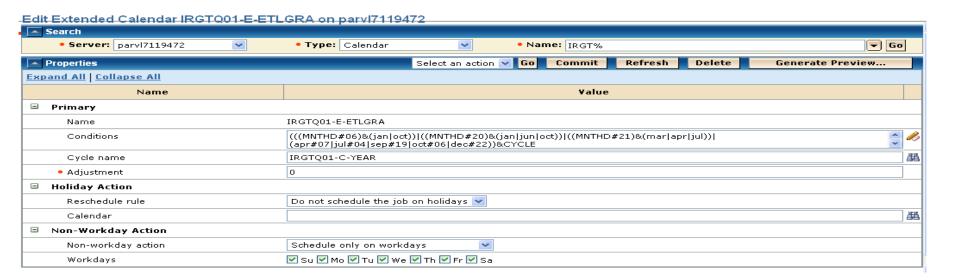


- ·Sélection au coup par coup.
- Navigation dans le temps sur10 ans
- •Gestion des conflits avec l'outil d'aides mais attention, il n'y a pas de sauvegarde de règle et donc de recalcul.

Periodes, Cycles, Calendrier etendu:

- Une période est un intervalle de temps délimité par une date de début et une date de fin.
- Un Cycle est un ensemble de périodes(1 à n).
- Un calendrier étendu est un calendrier généré à partir de conditions (mots clés), de périodes-cycles et d'ajustements,...
- Il y a des règles spécifiques qui s'appliquent aux cycles.





En ligne de commande

extended calendar: IRGTP01-E-ETLGRA

workday: all

non workday: N

holiday: holcal:

cyccal: IRGTP01-C-YEAR

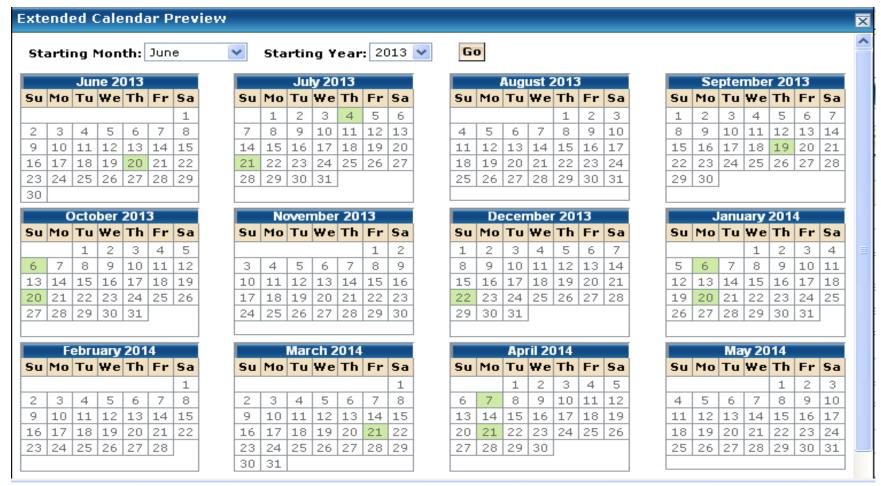
adjust: 0 condition:

(((MNTHD#06)&(jan|oct))|((MNTHD#20)&(jan|jun|oct))|((MNTHD#21)&(mar|apr|jul))|(apr#07|jul#04|sep#19|oct#06|dec#22))&CYCLE

charger le fichier avec autocal_asc -I fichier

Création des calendriers

Bouton Generate Preview ...



Autocal_asc interactif pour les calendriers étendus:

non_workday

Indicates the action to take when the selected day falls on a non-workday. Valid values are:

blank—Schedule always.

O—Schedule only on a non-workday.

N—Schedule on the next day, which could be a non-workday

W—Schedule on the next workday.

P—Schedule on the previous workday.

holiday

Indicates the action to take if the selected day is a holiday. Valid values are:

blank—Do not schedule.

O—Schedule only on a holiday.

S—Schedule always.

N—Schedule on the next day, which could be a holiday.

W—Schedule on the next workday.

P—Schedule on the previous workday

Holcal

Defines the calendar name to use for holiday dates.

Cyccal

Defines the cycle name to use and is only valid when the cycle keywords appear in the Extended calendar conditions.

adjust

Defines the number of days before (negative number) or after (positive number) the other specified criteria to run an associated job. Use adjustments if the other rules do not define the necessary days. For example, to run a job the day after every holiday, you could enter "holidays" in the Extended Calendar conditions field and "+1" in the Adjustments field.

59

Constitution des calendriers-Etendus.

Les règles: voir le RéférenceGuide, dont voici un extrait:

- Xddd#n—Include all occurrences of day ddd (mon sun) in a month except the nth one. Replace n with a number in the range 1 to 5 or L for the last occurrence in the month.
- dddMn—Include the nth occurrence of day ddd (mon sun) from the end of the month. Replace n with a number in the range 1 to 5.
- XdddMn—Include all occurrences of day ddd (mon sun) in a month except the nth one from the end of the month. Replace n with a number in the range 1 to 5.
- WORKDAYS—Include every day defined in the workday parameter but not in the holiday calendar specified in the holcal parameter.

 Dates in the holiday calendar are never considered workdays, irrespective of the action you specify in the holiday parameter.
- WORKD#nn—Include the nnth workday of the month. Replace nn with a number in the range 1 to 31 or L for the last day of the month.
- WORKDMnn—Include the nnth workday from the end of the month. Replace nn with a number in the range 1 to 31.
- **WORKDX***nn*—Include all workdays of the month except the *nn*th one. Replace *nn* with a number in the range 1 to 31 or L for the last day of the month.
- **FOMWORK**—Include the first workday of the month.
- **EOMWORK**—Include the last workday of the month.
- XFOMWORK—Include all workdays of the month except the first one.
- **XEOMWORK**—Include all workdays of the month except the last one.
- **WEEKDAYS**—Include Monday through Friday except for holidays. Dates in the holiday calendar are never considered weekdays, irrespective of the action you specify in the holiday parameter.
- **WEEKD#***n*—Include the *n*th weekday. Replace *n* with a number in the range 1 to 5 or L for the last weekday.
- **WEEKDM***n*—Include the *n*th weekday from the end of the week. Replace *n* with a number in the range 1 to 5.
- **WEEKDX***n*—Include all weekdays except the *n*th one. Replace *n* with a number in the range 1 to 5 or L for the last weekday.
- **FOMWEEK**—Include the first weekday of the month.
- **EOMWEEK**—Include the last weekday of the month.
- XFOMWEEK—Include all weekdays of the month except the first
- XEOMWEEK—Include all weekdays of the month except the last one.
- **FOM**—Include the first day of the month.
- **EOM**—Include the last day of the month.
- XFOM—Include all days of the month except the first one.
- XEOM—Include all days of the month except the last one.
- MNTHD#nn—Include the nnth day of the month. Replace nn with a number in the range 1 to 31 or L for the last day of the month.
- MNTHDMnn—Include the *nn*th day from the end of the month. Replace *nn* with a number in the range 1 to 31.
- MNTHDXnn—Include all days of the month except the *nn*th one. Replace *nn* with a number in the range 1 to 31 or L for the last day of the month.

60

Classification: Internal

Aide CYCLE.

You can use the following cycle keywords:

- CYCLE—Include any day within the date ranges, or periods defined in the cycle.
- CYCL#nnn—Include a specific day of each period in the cycle. Replace nnn with a number in the range 1 to 365.
- CYCL#L—Include the last day of each period in the cycle.
- CYCLMnnn—Include a specific day from the end of each period in the cycle. Replace nnn with a number in the range 1 to 365.
- CYCLXnnn—Include all days except day nnn of each period in the cycle. Replace nnn with a number in the range 1 to 365.
- CYCP#nn—Include a specific period in the cycle. Replace nn with a number in the range 1-36.
- CWORKD—Include all workdays in the cycle date range.
- **CWEEK#**nn—Include a specific week of each period in the cycle. The week begins on the first day of the period. The last week of a period may be a partial week. Replace nn with a number in the range 1 to 52.
- CWEEK#L—Include the last week of each period in the cycle. The last week of a period may be a partial week.
- **CWEEKM**nn—Include a specific week from the end of each period in the cycle. Replace nn with a number in the range 1 to 52.
- CWEEKXnn—Include all weeks except week nn of each period in the cycle. Replace nn with a number in the range 1 to 52.
- CWEEKD—Include all weekdays in the cycle date range.
- **CWRK**#nnn—Include a specific workday of each period in the cycle. Replace nnn with a number in the range 1 to 300.
- CWRK#L—Include the last workday of each period in the cycle.
- **CWRKM***nnn*—Include a specific workday from the end of each period in the cycle. Replace *nnn* with a number in the range 1 to 300.
- **CWRKX***nnn*—Include all workdays except day *nnn* of each period in the cycle. Replace *nnn* with a number in the range 1 to 300.
- CWRKXL—Include every workday except the last workday of each period in the cycle.
- **CWEK#***n*—Include a specific weekday of each period in the cycle. Replace *n* with a number in the range 1 to 5.
- CWEK#L—Include the last weekday of each period in the cycle.
- **CWEKM***n*—Include a specific weekday from the end of the week of each period in the cycle. Replace *n* with a number in the range 1 to 5.

61

Création des calendriers

Exemple de calendrier :

extended_calendar: IRGTP01-E-TRIM05

workday: all

non_workday: N

holiday:

holcal:

cyccal: IRGTP01-C-YEAR

adjust: 0

condition:

((((MNTHD#21|MNTHD#22|MNTHD#23|MNTHD#24|MNTHD#25|MNTHD#26|MNTHD#27|MNTHD#28|MNTHD#29|MNTHD#30)&(mar|jun|sep|dec))|(MNTHD#31&(mar|dec))|((MNTHD#01|MNTHD#02|MNTHD#03|MNTHD#04|MNTHD#05|MNTHD#06|MNTHD#07)&(apr|jul|oct|jan)))&CYCLE

Création des calendriers

Définition du cycle :

mktgua1@parvl7119472:/apps/exploit/autosys/MKTQ01/jil/livraison\$ vi CREATE CYCLE.txt

cycle: IMKTQ01-C-YEAR

start_date: 10/31/2012 end_date: 12/31/2012

Définition du calendrier :

rgtprd1@parvl7119481:PRD:/apps/exploit/autosys/RGTP01/jil/livraison# Vi IRGTP01-E-TLJ.txt

extended_calendar: IRGTP01-E-TLJ

workday: all

non workday: N

holiday: holcal:

cyccal: IRGTP01-C-YEAR

adjust: 0

condition: CYCLE

Chargement du cycle et du calendrier :

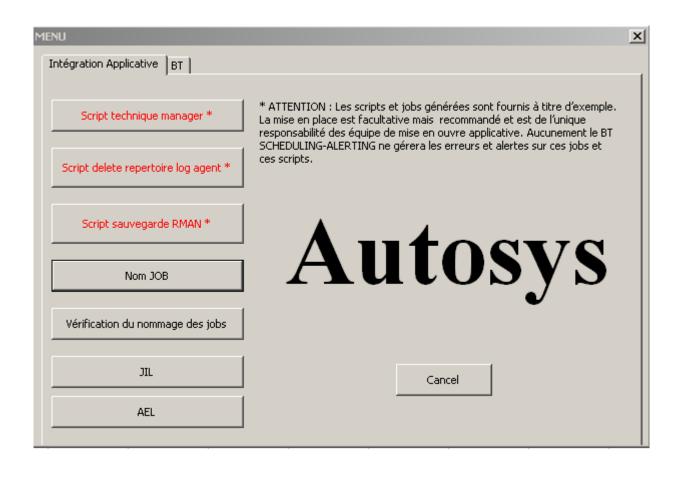
rgtprd1@parvl7119481:PRD:/apps/exploit/autosys/RGTP01/jil/livraison# autocal_asc -I CREATE_CYCLE.txt

Utilisation du fichier Excel :

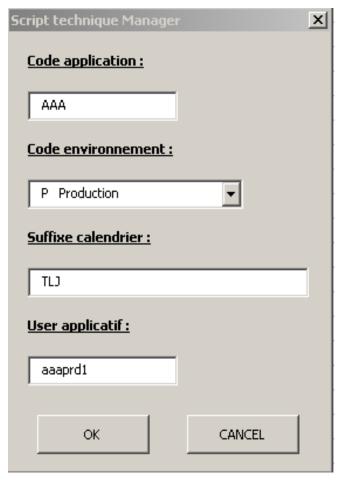
http://formicaio.itp.echonet/images/f/f8/AUTOSYS_JIL_AEL.zip

Sub contrated iii	code and licalic	on lake	reduence Code	Rivery 2	par ou lob	Destributed Box ni	Destribut Box	gestul. Description tob	/
IRIGP01-Q0BOX-APPLICATION	RIG	Q	BOX	0		APPLICATION			
IRIGP01-Q1BOX-APPLICATION-DEFINITION	RIG	Q	BOX	1		APPLICATION	DEFINITION		
IRIGP01-Q2EXP01-APPLICATION-DEFINITION-CALENDAR	RIG	Q	EXP	2	01	APPLICATION	DEFINITION	CALENDAR	NO
IRIGP01-Q2EXP01-APPLICATION-DEFINITION-JOB	RIG	Q	EXP	2	01	APPLICATION	DEFINITION	JOB	NO
IRIGP01-Q2EXP01-APPLICATION-DEFINITION-MACHINE	RIG	Q	EXP	2	01	APPLICATION	DEFINITION	MACHINE	NO
IRIGP01-Q2EXP01-APPLICATION-DEFINITION-RESSOURCE	RIG	Q	EXP	2	01	APPLICATION	DEFINITION	RESSOURCE	NO
IRIGP01-Q2EXP01-APPLICATION-DEFINITION-VARIABLE	RIG	Q	EXP	2	01	APPLICATION	DEFINITION	VARIABLE	
GP01-Q1BOX-APPLICATION-DELETE	RIG	Q	BOX	1		APPLICATION	DELETE		
IRIGP01-Q2DEL01-APPLICATION-DELETE-LOG	RIG	Q	DEL	2	01	APPLICATION	DELETE	LOG	NO
IRIGP01-Q2DEL01-APPLICATION-DELETE-OUT	RIG	Q	DEL	2	01	APPLICATION	DELETE	OUT	NO
IRIGP01-Q2DEL01-APPLICATION-DELETE-SAVE	RIG	Q	DEL	2	01	APPLICATION	DELETE	SAVE	
IRIGP01-Q1BOX-APPLICATION-CHECK	RIG	Q	BOX	1		APPLICATION	CHECK		
IRIGP01-Q2CHK01-APPLICATION-CHECK-CALENDAR	RIG	Q	СНК	2	01	APPLICATION	CHECK	CALENDAR	

Après activation des macros du fichier Excel, nous avons accès au menu



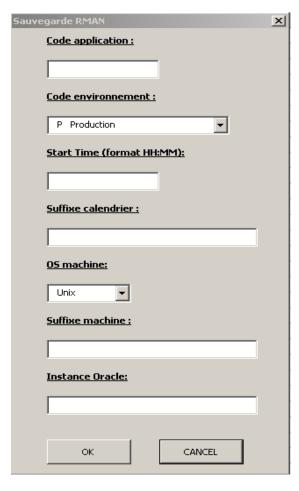
Menu Script technique



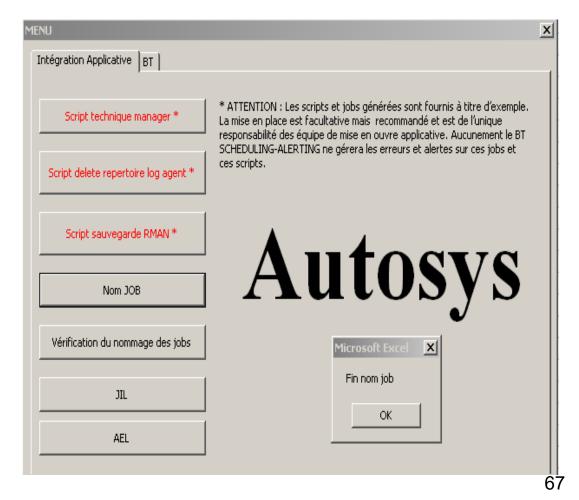
Menu script delete répertoire log agent

Delete log application agent	×
Code application :	
Code environnement :	
P Production	
<u>User applicatif :</u>	
Start Time (format HH:MM):	
<u>Suffixe calendrier :</u>	
<u>05 machine:</u>	
Unix	
<u>Suffixe machine :</u>	
OK CANCEL	

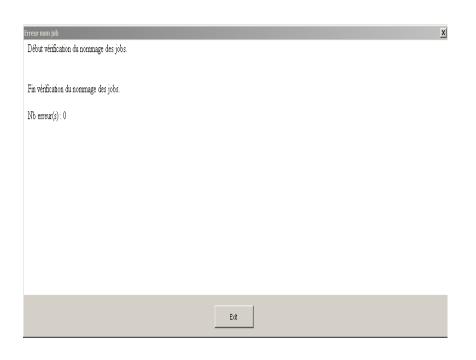
Menu **script** sauvegarde RMAN

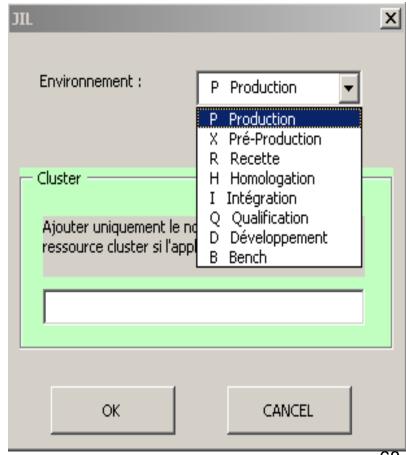


Vérification du nommage des jobs

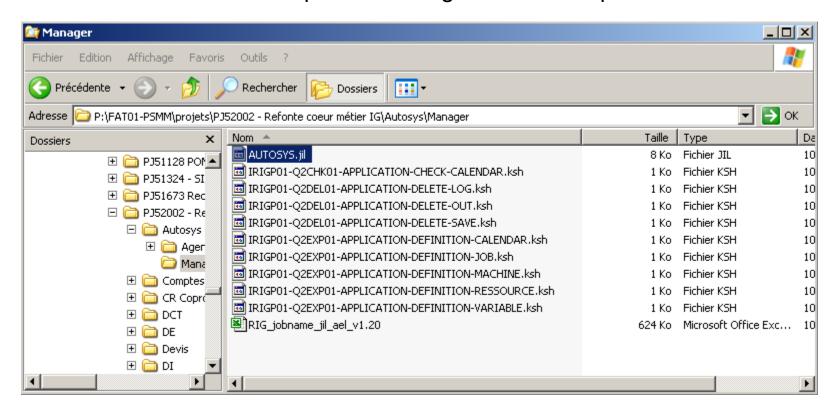


Bouton **JIL** : Création du fichier JIL et des scripts d'appels UNIX et Windows





Le fichier JIL et les scripts sont chargés dans un répertoire au choix



Transférer le fichier jil sur le manager dans le répertoire /apps/exploit/autosys/AAAE01/jil/livraison, puis le charger dans Autosys avec la commande : jil < fichier.jil

Exemple de fichier JIL créé :

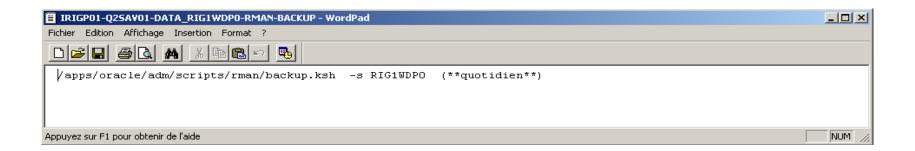
insert job: IRIGP01-Q0BOX-APPLICATION job type: BOX owner: rigprd1 permission: date conditions: 1 run calendar: IRIGP01-E-ALLDAYS start times: "15:00" condition: v(IRIGP01-ACTIVE) = "YES"& v(IRIGP01-LIVRAISON) = "YES" description: "" alarm if fail: 0 insert job: IRIGP01-Q1BOX-APPLICATION-DEFINITION job_type: BOX box name: IRIGP01-Q0BOX-APPLICATION owner: rigprd1 permission: date conditions: 1 run calendar: IRIGP01-E-ALLDAYS start times: "16:00" condition: v(IRIGP01-ACTIVE) = "YES"& v(IRIGP01-LIVRAISON) = "YES" description: "" alarm if fail: 0 insert_job: IRIGP01-Q2EXP01-APPLICATION-DEFINITION-CALENDAR job_type: CMD box name: IRIGP01-Q1BOX-APPLICATION-DEFINITION command: "/apps/exploit/autosys/RIGP01/script/IRIGP01-Q2EXP01-APPLICATION-DEFINITION-CALENDAR.ksh" profile: "/apps/exploit/autosys/RIGP01/conf/WAAE profile" machine: localhost owner: rigprd1 permission: date_conditions: 0 condition: v(IRIGP01-ACTIVE) = "YES" description: "Sauvegarde définition calendrier" std out file: "/apps/exploit/autosys/RIGP01/log/\${AUTO JOB NAME} \${AUTORUN}.log" std err file: "/apps/exploit/autosys/RIGP01/log/\${AUTO JOB NAME} \${AUTORUN}.err" alarm_if_fail: 1

70

Exemple de script généré :

```
IRIGP01-Q2CHK01-APPLICATION-CHECK-CALENDAR - WordPad
                                                                                                                  Fichier Edition Affichage Insertion Format ?
#!/bin/ksh
 APPLI=RIGPO1
 mois=2
 autosys calendrier.ksh -a I$(APPLI) -m $(mois)|sed 's/\f//g;s/\s\s*;/;/g;s/;\s\s*/;/g;/^$/d'|sed '/^;;*$/d'
 if [ $? -ne 0 ]; then
      echo""
  echo "SQL ko"
  echo""
  exit 4
 NB LIGNE=\{(autosys calendrier.ksh -a I\{\{APPLI\} -m \{\{mois\}\}sed 's/\{f//q;s/\s\s*;/;/q;s/;\s\s*/;/q;/^$/d'|sed '/^;;*$/d']
 if [[ "$NB LIGNE" != "2" ]] ; then
  echo "Il y a un calendrier qui n a pas de date de planification"
  echo""
  exit 8
 NB LIGNE=$(autosys calendrier.ksh -a I$(APPLI) -m $(mois)|sed 's/\f//q;s/\s\s*;/;/q;s/;\s\s*/;/q;/^$/d'|sed '/^;;*$/d
 if [[ "$NB LIGNE" != "0" ]] ; then
  echo""
  echo "Il y a un calendrier qui n est pas utilisé par un job"
  echo""
  exit 16
 fi
```

Exemple de script Unix



Exemple de script Windows



- ☐ Job Information Language, interface ligne de commandes.
 - ➤ Langage propre à AutoSys.
 - > Permet la définition des:
 - -Jobs
 - -Machines
 - -Monbro
 - -Cross-instances
 - -Ressources
- ☐ Utile pour sauvegarderles définitions
 - Autorep –J ALL –q >JOBS.jil
 - Autorep –M ALL –q >MACHINES.jil
- ☐ Import/Export d'une instance à une autre
 - ➤ Autorep –J jt% -q –D P01 | jil –S P02
- ☐ Chargement/ Modification en masse
- Mode INTERACTIF/BATCH.

Les règles syntaxiques.

- □ Règle 1
 - Chaque paragraphe débute par le type de commande (sub_command).
 - > sub command: nomdujob
- ☐ Règle 2
 - Chaque type de commande précède un ou plusieurs attributs. La définition des attributs doit respecter la syntaxe:
 - attribute keyword: valeur de l'attribut
- □ Règle 3
 - Plusieurs attributs peuvent être définis sur la même ligne s'ils sont séparés par au moins 1 espace.
- □ Règle 4
 - Une Box doit être définie avant les jobs (et les box) qui la composent.
- ☐ Règle 5
 - Les machines doivent être définies avant les jobs.

Remarques:

Il n'y a pas d'ordre pour le chargement des jobs.

Les calendriers peuvent être définis après.

□ Règle 6

- Chaque valeur contenant le caractère ":" doit être entre guillemets ou précédée de backslash.
- Exemple, pour préciser l'heure de démarrage, soit :
 - "10:00" on met 10\:00
 - C:\tmp on met c\:\tmp ou "c:\tmp"

□ Règle 7

- Des lignes de commentaires peuvent être ajoutées:
 - # lignede commentaire
 - /* ceci est un commentaire */

<u>Définition des objets en JIL:</u>

- □ Les types de commande pour job/box:
 - insert_job
 - update_job
 - delete_job
 - > override_job
 - delete_box
 - → Supprime la box et ses jobs
- □ Les types de commande pour les machines:
 - > insert machine
 - > delete machine
 - update_machine
- □ Les types de commande pour les ressources:
 - > insert resource
 - > delete resource
 - update_resource

Les modes d'utilisation.

```
Mode batch:
 Création d'un fichier jil : jil file
 $iil < jil file : import dans la base de données (en Unix)</p>
 Le code retour est égal au nombre d'erreurs.
Mode interactif (Unix ou Windows):
 $jil
 jil>>1>insert job: jobA
 jil>>2>command: sleep20
 iil>>3>machine: ferrari
 jil>>4>exit ou ctrl^D (import dans la base d'événements)
Exemple de fichier jil (création de Job):
 /*----*/
 insert job: jobA job type:c
 command: sleep20
 machine: ferrari
 owner: autosys@ferrari
 permission: gx,wx
 alarm if fail:1
```

□ Update de job

Pour mettre à jour un champ, 2 lignes sont nécessaires. La ligne update_job et la ligne modifiée

Exemple:

update_job: job1

owner: root

Pour supprimer un champ, le mettre à blanc.

update_job: job1 description:

Pour supprimer la planification

update_job: job1 date_conditions: 0

Suivi des Traitements

- Les différentes commandes
- La console graphique (WCC)
- Gestion des alertes sous Autosys
- Les actions Autosys

Toutes les commandes suivantes sont dans \$AUTOSYS/bin

<u>Définition des objets :</u>

- 1. JIL (Job Information language)
 - Permet la déclaration des objets (box,job) en mode interactif ou en mode batch.
 - Syntaxe : jil [option] <fichier.jil
 - Les options :
 - -v batch : vérifie les dépendances manquantes après l'insertion
 - -v syntax : vérifie la syntaxe jil sans insérer.

USAGE: jil [-q] [-SAUTOSERV] [-V none | job | batch | syntax] [-x] [-?]

WHERE: -q Runs the command in quiet mode

-S AUTOSERV Specifies the CA WAAE instance name

-V none Does not verify dependencies
 -V job Verifies job dependencies
 -V batch Verifies job dependencies after

JIL file is entirely processed

-V syntax Verifies the validity of the job definition only

-x Returns version information

-? Help

autocal_asc: gestion des calendriers

Surveillance de l'état du système:

1. autoping

- Vérification de la communication avec les agents
- Vérification de la connection au 'SE' (opt. -D)
- Émission d'alarme: -A

Syntaxe : Autoping -m <agent | ALL>

2. autosyslog

consultation des logs :

Scheduler: Autosyslog –e
Application server: Autosyslog –s

Job: Autosyslog –j <job> -t [-r]

-t

A=job agent log file

P=job agent environment profile log file

O=job standard output file

E=job standard error file

Pattern matching: -p

-r <n° run précédent>

3. Chase

Vérifie que les jobs qui sont marqués en «STARTING» et en «RUNNING» dans la database, sont effectivement dans cet état sur les agents, sur lesquels ils s'exécutent.

-A: émission d'alarme

-E : passe les jobs en FAILURE

4. chk_auto_up

Vérification de l'état du processeur d'événements, du serveur d'événements et du mode
 HA. Le code retour représente l'état de la configuration.

Reporting, dépendances et conditions.

1. job_depends

- Rapport sur les dépendances des jobs et leurs conditions de démarrage.
- En fonction de l'option spécifiée :
 - état courant des conditions de dépendance (-c).
 - Liste des prédécesseurs (-d).
 - Dépendance temporelle (-t) : forecast

Exemple:

job_depends –d –J box : montre les dépendances intra-box. Job_depends –c –J box/job

Job Name	Status	Date Cond?	Start Cond?	Dependent Jobs?
JOB1	INACTIVE	Met	Yes	No
Condition: s(JOB_ftp2)				
Atomic Condition		Current Status	T/F	
s(TEST ftp2)		INACTIVE	F	

Monitoring et reporting sur les statuts des jobs.

autorep

- c'est la commande AutoSys la plus utilisée.
- Produit un rapport sur le statut des jobs courants.
- Produit un rapport sur le statut d'une exécution passée.(*)
- Produit un rapport sur l'état des machines.
- Produit un rapport sur l'état des variables AutoSys.
- Donne la définition JIL des jobs et des machines.(*)

(*)Exemple:

```
« autorep –J jobname –r -2 » (Rapport du run -2)
« autorep –J ALL –q » (définition JIL, texte, de tous les Jobs)
```

> Quelque soptions:

-J : les jobs -B : groupes

-I : Applications

-o : override -V : Ressource

-G : variables globales

-M: machines

-r –x : historiquedesjobs -q : affiche le script JIL -Lx : niveau d'imbrication

Autorep –J ALL –L0 : affiche toutes les boxes de1er niveau et les jobs seuls

Autorep –J jt% -r -1 : affiche un rapport du run moins un des jobs jt*

Modification de la planification d'un job

```
@parvl7119481:/home/admbtscheduling$ autorep -j IBIVP01-Q2STO01-TLJ-ARRET-SURV -d
Job Name
                              Last Start
                                         Last End
                                                     ST Run/Ntry Pri/Xit
IBIVP01-Q2STO01-TLJ-ARRET-SURV
                                        2013/01/29 15:02:23 2013/01/29 15:02:48 SU 29165/1 0
                     Ntry ES ProcessTime
Status/[Event] Time
                                         Machine
STARTING
           RUNNING
           <Executing at PARS07118477>
SUCCESS
           [STARTJOB] 2013/01/30 02:00:00 0 PD 2013/01/30 02:00:00
 <Event was scheduled based on job definition.>
@parvl7119481:/home/admbtscheduling$ autorep -j IBIVP01-Q2STO01-TLJ-ARRET-SURV -q -l0 | sed 's/insert job: /update job: /g' | jil
CAUAJM I 50323 Inserting/Updating job: IBIVP01-Q2STO01-TLJ-ARRET-SURV
CAUAJM I 10122 Job 'IBIVP01-Q2STO01-TLJ-ARRET-SURV' scheduled: 2013/01/31 02:00:00
CAUAJM I 50205 Database Change WAS Successful!
CAUAJM I 52301 Exit Code = 0
@parvl7119481:/home/admbtscheduling$ autorep -j IBIVP01-Q2STO01-TLJ-ARRET-SURV -d
                              Last Start
                                         Last End
                                                     ST Run/Ntry Pri/Xit
Job Name
IBIVP01-Q2STO01-TLJ-ARRET-SURV
                                        2013/01/29 15:02:23 2013/01/29 15:02:48 SU 29165/1 0
Status/[Event] Time
                     Ntry ES ProcessTime
                                         Machine
STARTING
           RUNNING
           <Executing at PARS07118477>
SUCCESS
           [STARTJOB] 2013/01/30 02:00:00 0 PD 2013/01/30 02:00:00
 <Event was scheduled based on job definition.>
                                                                                                85
          2013/01/31 02:00:00 0 UP
[STARTJOB]
 <Event was scheduled based on job definition.>
```

Classification : Internal

Autostatus

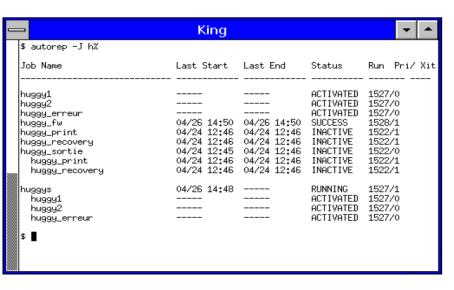
➤ donne l'état courant d'un job ou d'une variable AutoSys.

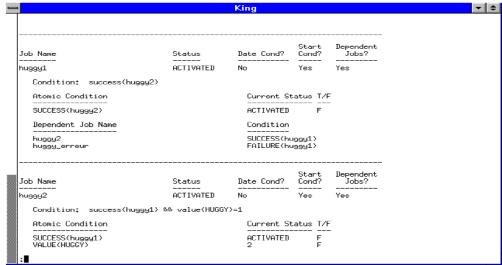
Ex : autostatus -J IBIVQ01-Q2SAV01-TLJ-SAVEORAC-BACKUP

ACTIVATED

➤ Autorep –J h%

job_depends -c -J h%.





Le forecast :

job dépends –t

```
job_depends –J h% -t –F 'MM/dd/YYYY HH:MM '' –T ''MM/dd/YYYY HH:MM'' L'option –e affiche toutes les exécutions à venir d'un job cyclique.
```

> forecast

Comme job_depends avec la possibilité de spécifier la machine : -M Et d'afficher les successeurs : -l

Les actions.

> sendevent

-E: Event

-A: Alarm

-J: JobName

-B: GroupName

-I: ApplicationName

-s : Status

-S : AUTOSERV CA WAAE instance name
-P : Event Priority Assigns a priority to the event

-M : Max Send Trys Number of attempts to be made to send event in case of failure

-q : Job Queue Priority-G : Global=ValueAssigns new queue priority to the jobSets value to the global variable

-C : Comment Sends a comment (Can be alphanumeric and should not exceed 255 characters)

-U: (Un-SENDEVENT) Cancels the last event specified by -E event

-T : Time of Event Processes the event on specified date and time in "MM/dd/[yy]yyhh:mm" format

-K : Signal(s) Sends a signal to a running Job

-r: Response Sends a response to a running Interactive Job

-o : Step Name Sends a step name to a failed Microfocusor ZOS Job -N : Machine Real Machine name to send MACH OFFLINE and

-F: FileName File name containing the sendevent commands for batch processing MACH_ONLINE

events

> Les événements:

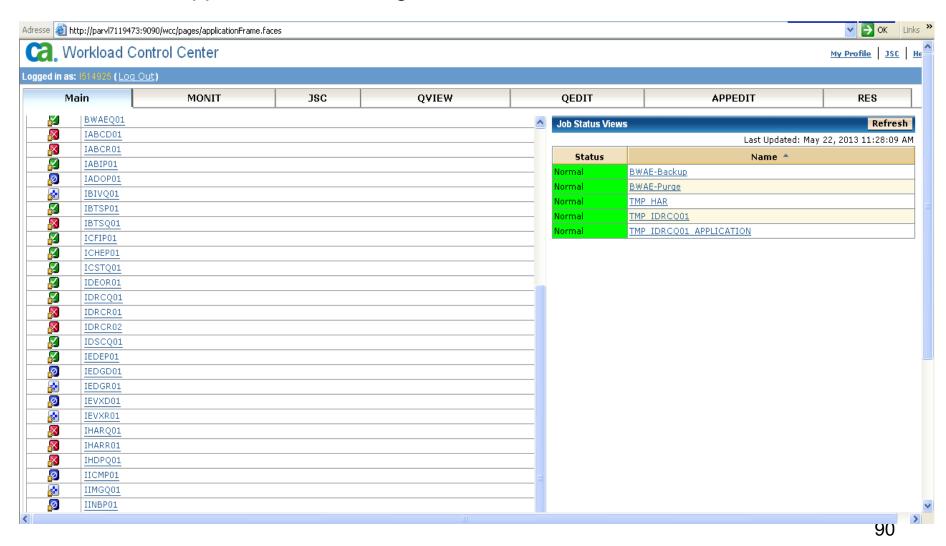
STARTJOB, FORCE_STARTJOB, KILLJOB, RESTARTJOB, STOP_DEMON, JOB_ON_HOLD, JOB_OFF_HOLD, JOB_ON_ICE, JOB_OFF_ICE, CHANGE_STATUS, RELEASE_RESOURCE, REPLY_RESPONSE, MACH_ONLINE, MACH_OFFLINE, DELETEJOB, SET_GLOBAL, 88 CHANGE PRIORITY

Classification: Internal

☐ Les onglets :

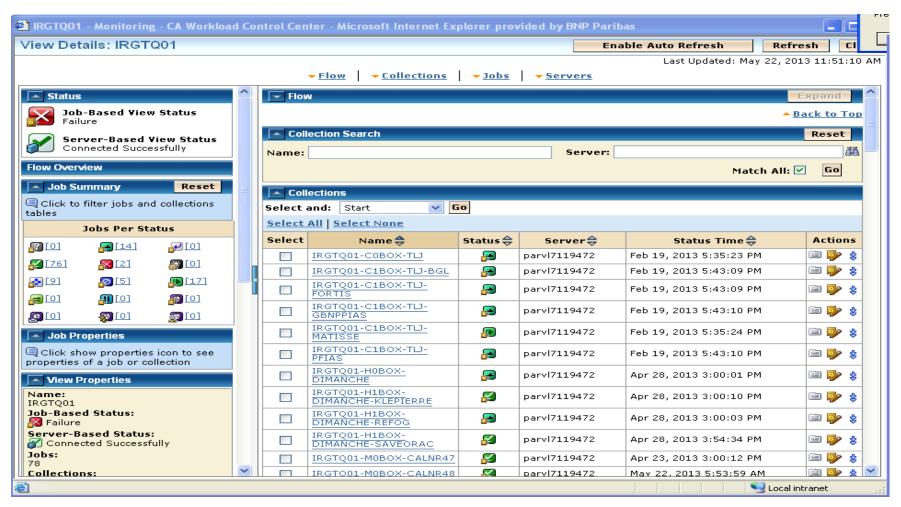
- ➤ **MONIT**: pour Monitoring Views. Sert au pilotage des chaines sous forme de diagramme de Pert.
- > **JSC**: job status Console. Pilotage des jobs sous forme tabulaire. Accès aux logs du server, aux alarmes et Dashboard.
- QVIEW: Quick View. Accès à l'ensemble des informations sur un job: définition, dépendances, historiques, logs. Le Quick View est accessible contextuellement dans MONIT et JSC.
- > ECLI: éditeur de commandes en ligne
- > RPT: reporting, forecast, diagrammes de Gantt.
- > QEDIT: Edition jobs, calendriers, cycles et variables globales
- > APPEDIT: Design des chaînes d'ordonnancement.
- > **RES**: gestion des ressources
- > CRED: les credentials users (mapping des comptes Web).
- > CONFIG: Configuration de WCC

Sélection de l'application dans l'onglet Main

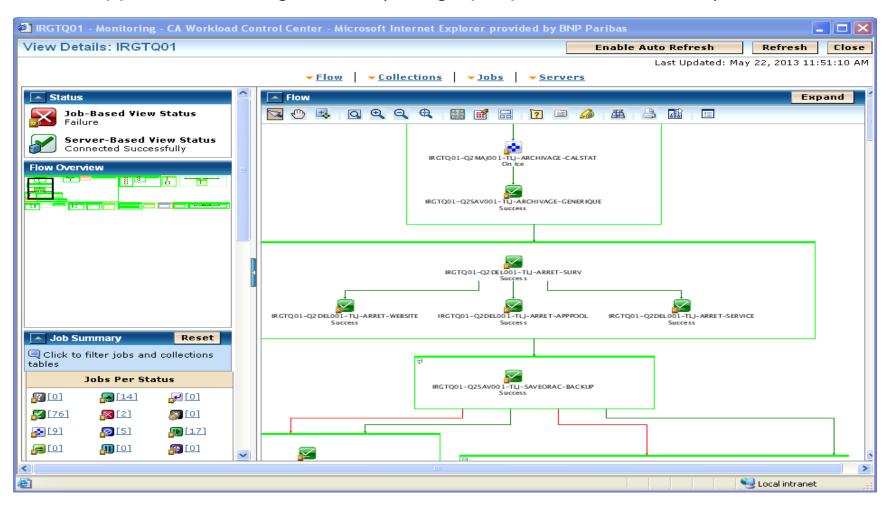


Classification: Internal

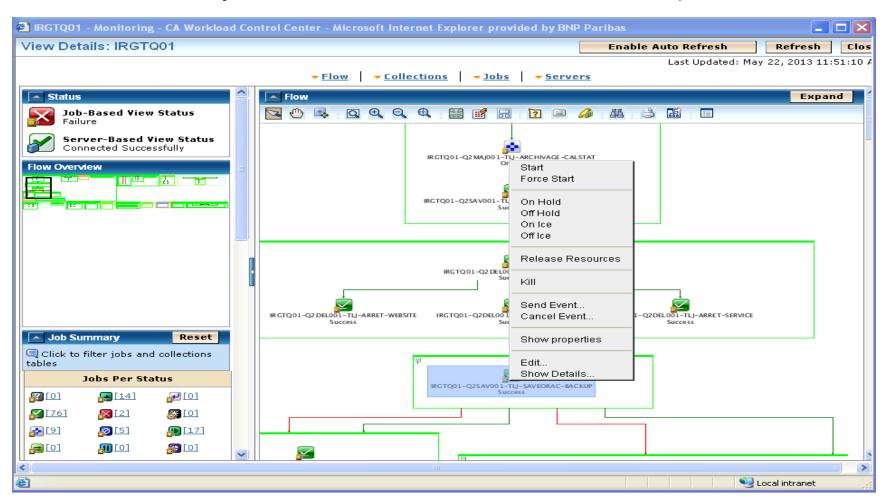
Vue des Boxes et jobs



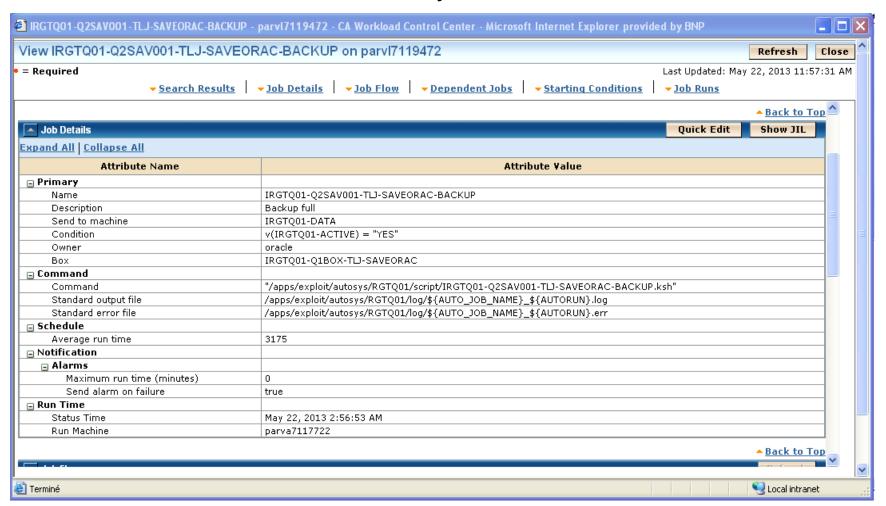
Développement de l'onglet Flow (Vue graphique des traitements)



Le clic droit sur un job ou une boxe donne la liste des actions possibles



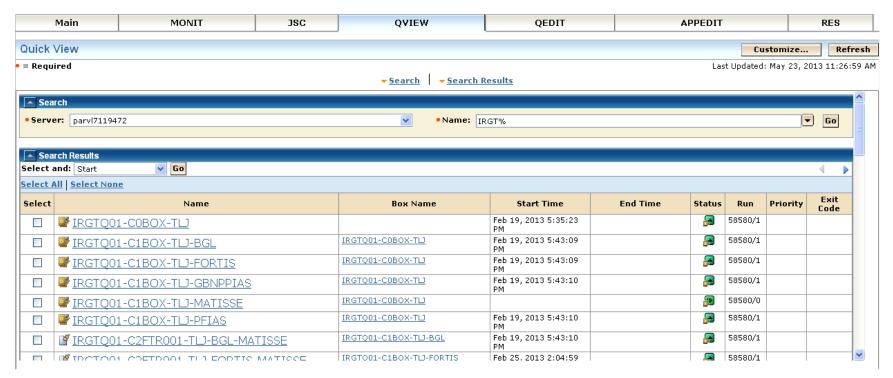
Vue Show Detail : Détail de l'état du job



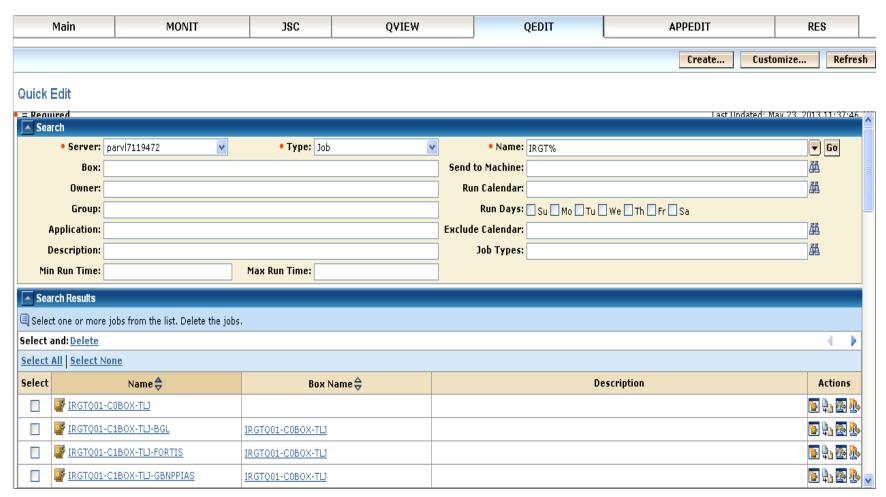
Possibilité de voir les logs d'éxécution

Start time May 22, 2013 2:04:08 AM May 21, 2013 2:04:17 AM May 20, 2013 2:04:07 AM May 19, 2013 2:04:02 AM May 18, 2013 2:04:18 AM May 17, 2013 2:04:20 AM May 16, 2013 2:04:09 AM 32m 2.1.9 14/08/2012 : SunOS - AIX5L/6	End Time May 22, 2013 2:56:53 AM May 21, 2013 2:57:12 AM May 20, 2013 2:57:06 AM May 19, 2013 2:57:23 AM May 18, 2013 2:57:02 AM May 17, 2013 2:57:30 AM May 16, 2013 2:56:44 AM × - Linux x86 64 / serveur : parva7117722 @[0	Status SUCCESS SUCCESS SUCCESS SUCCESS SUCCESS SUCCESS SUCCESS SUCCESS	Run 110561/1 109829/1 109201/1 108766/1 108199/1 107466/1 106620/1	Exit Code 0 0 0 0 0 0 0 0	Machine parva7117722 parva7117722 parva7117722 parva7117722 parva7117722 parva7117722 parva7117722
May 22, 2013 2:04:08 AM May 21, 2013 2:04:17 AM May 20, 2013 2:04:07 AM May 19, 2013 2:04:02 AM May 18, 2013 2:04:18 AM May 17, 2013 2:04:20 AM May 16, 2013 2:04:09 AM	May 22, 2013 2:56:53 AM May 21, 2013 2:57:12 AM May 20, 2013 2:57:06 AM May 19, 2013 2:57:23 AM May 18, 2013 2:57:02 AM May 17, 2013 2:57:30 AM May 16, 2013 2:56:44 AM	SUCCESS SUCCESS SUCCESS SUCCESS SUCCESS SUCCESS	110561/1 109829/1 109201/1 108766/1 108199/1 107466/1	0 0 0 0 0	parva7117722 parva7117722 parva7117722 parva7117722 parva7117722 parva7117722
May 21, 2013 2:04:17 AM May 20, 2013 2:04:07 AM May 19, 2013 2:04:02 AM May 18, 2013 2:04:18 AM May 17, 2013 2:04:20 AM May 16, 2013 2:04:09 AM	May 21, 2013 2:57:12 AM May 20, 2013 2:57:06 AM May 19, 2013 2:57:23 AM May 18, 2013 2:57:02 AM May 17, 2013 2:57:30 AM May 16, 2013 2:56:44 AM	SUCCESS SUCCESS SUCCESS SUCCESS	109829/1 109201/1 108766/1 108199/1 107466/1	0 0 0 0 0 0	parva7117722 parva7117722 parva7117722 parva7117722 parva7117722
May 20, 2013 2:04:07 AM May 19, 2013 2:04:02 AM May 18, 2013 2:04:18 AM May 17, 2013 2:04:20 AM May 16, 2013 2:04:09 AM	May 20, 2013 2:57:06 AM May 19, 2013 2:57:23 AM May 18, 2013 2:57:02 AM May 17, 2013 2:57:30 AM May 16, 2013 2:56:44 AM	SUCCESS SUCCESS SUCCESS	109201/1 108766/1 108199/1 107466/1	0 0 0	parva7117722 parva7117722 parva7117722 parva7117722
May 19, 2013 2:04:02 AM May 18, 2013 2:04:18 AM May 17, 2013 2:04:20 AM May 16, 2013 2:04:09 AM	May 19, 2013 2:57:23 AM May 18, 2013 2:57:02 AM May 17, 2013 2:57:30 AM May 16, 2013 2:56:44 AM	SUCCESS SUCCESS	108766/1 108199/1 107466/1	0 0	parva7117722 parva7117722 parva7117722
May 18, 2013 2:04:18 AM May 17, 2013 2:04:20 AM May 16, 2013 2:04:09 AM	May 18, 2013 2:57:02 AM May 17, 2013 2:57:30 AM May 16, 2013 2:56:44 AM	SUCCESS	108199/1 107466/1	0	parva7117722
May 17, 2013 2:04:20 AM May 16, 2013 2:04:09 AM	May 17, 2013 2:57:30 AM May 16, 2013 2:56:44 AM	SUCCESS	107466/1	0	parva7117722
May 16, 2013 2:04:09 AM	May 16, 2013 2:56:44 AM				<u>'</u>
	· ·	SUCCESS	106620/1	0	parva7117722
32m 2.1.9 14/08/2012 : SunOS - AIX5L/6	x - Linux x86 64 / serveur : parva7117722 @[0				
le configuration : '/apps/oracle/adm/scrip og global : /apps/oracle/backup/log/; alogue RMAN : RMANCAT atalogue RMAN : parvl9011338 ni : RMN1FRP0 n : hot : 0 : FRA	ts/rman/rman_backup_parva7117722.conf				
os os de og alc alc an	sous DISK : 1 DAYS sous SBT : DAYS cous SBT : DAYS : /apps/oracle/adm/scripts/r configuration : /apps/oracle/backup/log/l global : /apps/oracle/backup/log/l gue RMAN : RMANCAT talogue RMAN : parvl9011338 : RMN1FRP0 : hot : 0 : FRA	sous DISK : 1 DAYS sous SBT : DAYS : /apps/oracle/adm/scripts/rman/rman_errors.keys configuration : /apps/oracle/adm/scripts/rman/rman_backup_parva7117722.conf global : /apps/oracle/backup/log/parva7117722_05.lst glue RMAN : RMANCAT talogue RMAN : parvl9011338 : RMN1FRP0 : hot : 0 : FRA : RGT3FRQ0	sous DISK : 1 DAYS sous SBT : DAYS : /apps/oracle/adm/scripts/rman/rman_errors.keys configuration : /apps/oracle/backup/log/parva7117722_conf global : /apps/oracle/backup/log/parva7117722_05.lst glue RMAN : RMANCAT talogue RMAN : parvl9011338 : RMN1FRP0 : hot : 0 : FRA : RGT3FRQ0	sous DISK : 1 DAYS sous SBT : DAYS : /apps/oracle/adm/scripts/rman/rman_errors.keys configuration : /apps/oracle/backup/log/parva7117722_conf global : /apps/oracle/backup/log/parva7117722_05.lst glue RMAN : RMANCAT talogue RMAN : parvl9011338 : RMN1FRP0 : hot : 0 : FRA : RGT3FRQ0	sous DISK : 1 DAYS sous SBT : DAYS : /apps/oracle/adm/scripts/rman/rman_errors.keys configuration : /apps/oracle/backup/log/parva7117722_conf global : /apps/oracle/backup/log/parva7117722_05.lst orgue RMAN : RMANCAT talogue RMAN : parvl9011338 : RMNIFRP0 : hot : 0 : FRA

Onglet QVIEW: Informations sur les traitements, permet aussi d'intervenir sur les traitements



QEDIT : Edition des jobs, boxes, calendriers, cycles et variables globales



Remontée des alertes

➤ Voici le format de message qui est remonté vers les serveurs ARS (les valeurs qui ne sont pas en gras sont les valeurs données par défaut pour respecter le format workload du message):

AAAAAAA BBBBBBB CC DDDD EEEEEEEEEE FFF GGGGGGGGGGGG EM2:

CAUAJM_I_40245_XXXXXXXXXX ZZZZZZZ AUTOSYSJOB 0000 0000 XXXXXXXXXXX at HH.HH.HH

code: 00000001 at manager EEEEEEEEEE WCC = YYYYYYYYY

AAAAAAA : Date de l'alerte

BBBBBBBBB : Heure de l'alerte

CC : Metier

DDDD : Environnement

EEEEEEEEE : Serveur

FFF : OS

GGGGGGG : Application

YYYYYYYY : Nom du serveur WCC

XXXXXXXXX : Type de l'alarme (JOBFAILURE, MUST_START_ALARM,

MUST_COMPLETE_ALARM, MAXRUNALARM)

AUTOSYSJOB : Nom du job donné par défaut

ZZZZZZZZZ : Nom du jobset (cela correspond en réalité au nom du job autosys)

Voici un exemple de message qui est remonté vers les serveurs ARS :

02/06/11 23:59:58 FG PROD parva0106038 AIX MUTUALISATION EM2: CAUAJM_I_40245_JOBFAILURE MONJOBA AUTOSYSJOB 0000 0000 JOBFAILURE at 19.00.29 code: 00000001 at manager parva0106038 WCC = parsl9001649

>ARS : Mécanisme de création des tickets

Ressource 1

Nom du Job Autosys (Exemple : PAUTOSYSP01-O1CHK-jobtest1)

Ressource 2

AUTOSYSJOB (Valeur par défaut car autosys ne remonte pas le nom du jobset/box)

Type Incident

Type d'alerte autosys. Ces nouveaux codes ont du être créés au niveau de la base ARS afin que les tickets ARS puissent être créés automatiquement lors de l'arrivée des messages autosys. A chacun de ces codes est associé un libellé (Champs « Libellé Type incident »). Voici les 4 cas codes alertes créés :

- o CAUAJM_I_40245_JOBFAILURE : Traitement terminé en erreur
- o CAUAJM_I_40245_MUST_START_ALARM: Traitement non démarré
- o CAUAJM I 40245 MUST COMPLETE ALARM: Traitement non terminé
- o CAUAJM I 40245 MAXRUNALARM: Traitement non terminé

Qualifier

0000 (Valeur par défaut, car autosys ne remonte pas cette information)

Code Erreur

0000001 (Valeur par défaut, car autosys ne remonte pas cette information)

OS

Os du serveur (Exemple : AIX)

Date Incident

Date du message (Exemple : 02/06/11 23:59:58)

Code METIER

Métier du message (Exemple : **FG**)

Application

Nom de l'Application associé au job autosys (Le nom de l'application est défini dans une table ARS. Cette table ARS est alimentée par le pilotage DP lorsque les MOE demandent à mettre en pilotage de nouveaux jobs.)

Libellé Type Incident

Description du type d'incident (voir Type Incident pour détails).

- Voici ci-dessous quelques exemples de code ael pour des alertes de job autosys :
 - CAUAJM_I_40245_JOBFAILURE_MONJOB1_AUTOSYSJOB
 - CAUAJM_I_40245_MUST_START_ALARM_MONJOB2_AUTOSYSJOB
 - CAUAJM_I_40245_ MUST_COMPLETE_ALARM_MONJOB3_AUTOSYSJOB
 - CAUAJM_I_40245_ MAXRUNALARM_MONJOB4_AUTOSYSJOB

Gestion des Alarmes Autosys

En version WAAE 11.3, vous trouverez ci-dessous le répertoire « /apps/waae/custom » contenant des scripts utiles :

- custom
- custom/waae_custom.ini (Contains hostname for WCCNODE)
- custom/opr (Uniquement disponible si la version Event Manager 11.2 est présente sur la machine. OPR uniquement disponible pour la remontée des alertes WAAE.)
- custom/opr/create msg.ksh
- custom/opr/list msg.ksh
- custom/opr/enable msg.ksh
- custom/opr/disable_msg.ksh
- custom/opr/delete_msg.ksh

custom/opr gestion des alertes

-Création de message pour une application XXXXX, c'est-à-dire que le nom des jobs commence par XXXXX

create_msg.ksh XXXXX*

crée ces messages

CAUAJM I 40245 EVENT: ALARM	ALARM: JOBFAILURE JOB: XXXXX*
CAUAJM_I_40245 EVENT: ALARM	ALARM: MUST_START_ALARM JOB: XXXXX*
CAUAJM_I_40245 EVENT: ALARM	ALARM: MUST_COMPLETE_ALARM JOB: XXXXX*
CAUAJM_I_40245 EVENT: ALARM	ALARM: MAXRUNALARM JOB: XXXXX*

Lister les messages: list_msg.ksh

```
        opra_msg_tkn |
        opra_msg_id
        | opra_msg_active

        37 | NSM_I_ACT EXPORT DSMNODE
        | Y

        38 | .CAOP_I_DAEMONINIT Daemon initialized*
        | Y

        39 | .CAOP_I_DBCACHED Database (re)loaded*
        | Y

        40 | NSM_I_ACT EXPORT WCCNODE
        | Y

        41 | NSM_I_ACT SET WCCNODE
        | Y

        42 | CAUAJM_I_ACT SET WCCNODE
        | Y

        43 | CAUAJM_I_40245 EVENT: ALARM
        ALARM: JOBFAILURE
        JOB: * | N

        43 | CAUAJM_I_40245 EVENT: ALARM
        ALARM: MUST_START_ALARM JOB: * | N
```

- supprimer un message: delete_msg.ksh

delete_msg.ksh 42

supprime le message "CAUAJM_I_40245 EVENT: ALARM ALARM: JOBFAILURE JOB: *"

- désactiver un message: disable_msg.ksh

disable_msg.ksh 42

désactive "CAUAJM_I_40245 EVENT: ALARM ALARM: JOBFAILURE JOB: *"

- Activer un message: enable_msg.ksh

enable_msg.ksh 42

Active "CAUAJM_I_40245 EVENT: ALARM ALARM: JOBFAILURE JOB: *"

Les statuts

- Job seul
 - > STARTING: état temporaire. Attend le retour de l'agent pour passer RUNNING
 - > RUNNING : commande en cours.
 - SUCCESS: bon code retour
 - > FAILURE: mauvais code retour
 - TERMINATED: commande « killée »
- Box
 - RUNNING: active tous les box/jobs de la box et réinitialise les statuts en les passant en ACTIVATED
 - SUCCESS/FAILURE: dépend des statuts de ses jobs.
- Job / Box dans une box
 - ACTIVATED: statut qui précède le STARTING/RUNNING. Signifie qu'il est en attente de dépendance de prédécesseur ou calendaire.
 - INACTIVE: désactivé par l'arrêt de la box.
- Job en attente:
 - ACTIVATED
 - PENDING_MACHINE: attend que la machine repasse ON_LINE
 - RESOURCE WAIT: attente de ressource
 - > WAIT REPLY: sur Windows, pour les jobs interactifs
 - QUEUE WAIT: en attente dans une file.
- Job « désactivés »
 - ON HOLD
 - ON_ICE

104

<u>Moyens</u>

- Utilisation de l'interface graphique.
- Utilisation de la commande sendevent
- Démarrage :
 - ➤ L'événement STARTJOB pour un job/box à la demande.
 - ➤ L'événement FORCE_STARTJOB pour outrepasser les planifications et dépendances et démarrer un job/box se trouvant dans une box non RUNNING ou qui s'est déjà exécuté dans cette box.
 - Sendevent –E FORCE_STARTJOB –J gr1_box0 –c « demande de M untel »
- ☐ KILL, ARRET
 - > KILLJOB
 - -Sur un job : kill la commande et passe le job en TERMINATED

- ☐ Changement des statuts, « Déblocage »
 - > Ressource

Sendevent –E RELEASE_RESOURCE –J job1

> Statuts:

Sendevent –E CHANGE_STATUS –J job1 –s SUCCESS

SUCCESS INACTIVE FAILURE TERMINATED

Sur une box RUNNING, un changement de statut rend INACTIVE ses jobs qui ne sont pas encore exécutés.

□ Désactivation, Hold

> ON_HOLD : dans un enchaînement, bloque le successeur Sur un cyclique, met en pause.

Sendevent –E JOB_ON_HOLD –J job

➤ ON_ICE : dans un enchaînement, saute le job Sur un cyclique, met en pause.

Sendevent -E JOB_ON_ICE -J job

Actions impossible sur les jobs en STARTING ou en RUNNING.

□ Réactivation

> OFF_HOLD : débloque le successeur

Sur un cyclique, réactive immédiatement la planification, démarrage immédiat si la dernière planification est passée.

➤ OFF_ICE : dans un enchaînement, réactive le job pour la prochaine fois. Sur un cyclique, replanifie le job

Autres

- □ SEND_SIGNAL
 - Envoi d'un signal Unix. Le signal souhaité doit être saisi.

Sendevent –E SEND SIGNAL –k 9 – J job1

- □ CHANGE_PRIORITY
 - Dans une file d'attente, priorise un job

Sendevent -E CHANGE_PRIORITY -q 10 -J job1

- □ SET_GLOBAL
 - Valorise une variable Globale

Sendevent -E SET GLOBAL -G var=valeur

- Suppression d'un job
 - > sendevent -E DELETEJOB -J job1

Communs à tous les sendevent

- -c « commentaires » : ajoute un commentaire à l'événement dans le log du Scheduler
- -P n : priorité de l'événement
- -T « MM/dd/YYYY » : planification de l'événement
- -U : annule la précédente planification

Référentiel

Referentiel Bp2i:

http://sia.bp2i.echonet/public/Scheduling/Forms/AllItems.aspx

BCC:

http://formicaio.itp.echonet/wiki/index.php/Alerting/Scheduling

Support Formation Autosys:

\\Fa3s00562569\CSBP SOGETI\FORFAIT SOGETI\FORMATION\2013\Formation Autosys

🖃 🧀 Formation Autosys

Document

Documentation CA et BNP

🗉 🧰 Mode Opi

Modes Opératoires

田 🛅 Planning

Planning des formations

Support de cours

Supports de cours

Démarrage cybagent =>
/apps/waae/11.3/SystemAgent/<SERV
EUR>./waae_agent start

 root@parsI7406731:/apps/waae/11.3/S ystemAgent/parsI7406731# ./cybAgent -a