Utilizzo e sviluppo dei demoni con itaDaemon

E' stata sviluppata in itaEngine un'infrastruttura che permette l'esecuzione di script in maniera continuativa simulando il comportamento di un demone Linux o di un servizio Windows.

Creazione di un demone

Per creare un nuovo demone è necessario creare una nuova classe che estende la classe itaBaseDaemon ed impostare di conseguenza il file di configurazione dei demoni posto sotto /daemon/config/autostart.ini

Creazione della classe *Daemon.class.php

Per creare un nuovo demone è necessario creare una classe che estende itaBaseDaemon. Tale classe deve avere come nome *"<nome_demone>Daemon.class.php"* ed essere inserita nel percorso *"/daemon/daemons/*Daemon.class.php"*. La classe deve implementare tre metodi:

- **executeStart($args)** Il metodo viene eseguito ad ogni ciclo di esecuzione del demone
- **executeStop($args)** Il metodo viene eseguito una sola volta, alla distruzione del demone, ovvero quando questo viene fermato o nel caso avvenga un'eccezione.
- **getSleepTime()** questa funzione dovrà semplicemente restituire un integer che rappresenta il tempo in secondi fra un'esecuzione di executeStart e la successiva.

E' inoltre possibile, ma non necessario, implementare i seguenti metodi:

- **checkStartPreconditions($args)** Il metodo viene eseguito una sola volta alla creazione del demone, in maniera simile ad un costruttore.
- **checkStopPreconditions($args)** Il metodo viene eseguito una volta prima di executeStop()

n.b. *Un crash completo del demone (come ad esempio quello provocato da un sigkill) non porta all'esecuzione di executeStop.*

Configurazione del file autostart.ini

Oltre alla creazione della classe è necessario registrare il nuovo demone all'interno del file di configurazione *"/daemon/config/autostart.ini"* Il file di configurazione è strutturato in questo modo:

- Ogni sezione rappresenta un demone, con il nome della sezione che deve assumere il nome del demone
- All'interno di ogni sezione sono presenti due parametri:

1. *autostart*: può assumere valore true o false ed indica se il demone parte automaticamente all'avvio del servizio di gestione
2. *parameters*: sono parametri aggiuntivi che vengono passati sia alla creazione che alla distruzione del demone. E' possibile passare più parametri dividendoli dal carattere di pipe (|)

Utilizzo del demone

E' possibile lanciare e controllare un demone fondamentalmente in tre modi differenti:

- istanziando un oggetto di tipo itaDaemonManager e usando i metodi forniti da quest'ultimo per gestire un demone.
- Lanciando il file /daemon/itaDaemonAutostart.php e poi controllando i demoni con lo script /daemon/itaDaemonController.php
- Wrappando lo script php /daemon/itaDaemonExecutor.php all'interno di un servizio Windows o di un demone Linux

Istanziare itaDaemonManager

Per istanziare la classe di controllo dei demoni itaDaemonManager è sufficiente includere il file e creare un nuovo oggetto:

```
require_once ITA_BASE_PATH . '/daemon/lib/itaDaemonManager.class.php';
$daemonManager = new itaDaemonManager();
```

Una volta creato itaDaemonManager è possibile usarlo per gestire i demoni con i seguenti metodi:

- **getDaemons()** Restituisce un array contenente tutti i demoni presenti sul file di configurazione autostart.ini. Ogni elemento è a sua volta un'array, si consiglia di leggere il commento al metodo per maggiori informazioni.
- **getDaemonStatus($daemon)** Restituisce un array contenente le informazioni sul demone richiesto. Si consiglia di leggere il commento al metodo per maggiori informazioni.

  ```
  $info = $daemonManager->getDaemonStatus('dummy');
  ```

- **refreshDaemons()** Fa un controllo dello stato dell'esecuzione di tutti i demoni, riavviando quelli che dovrebbero essere attivi ma risultano fermi da un tempo troppo elevato.
- **startDaemon($daemon,$args)** Avvia il demone passato con, opzionalmente, dei parametri addizionali. Tali parametri vanno passati sotto forma di array.

  ```
  $daemonManager->startDaemon('dummy',array('hello World'));
  ```

- **stopDaemon($daemon,$args)** Ferma il demone passato con, opzionalmente, dei parametri addizionali. Tali parametri vanno passati sotto forma di array.
- **pauseDaemon($daemon)** Mette in pausa l'esecuzione del demone passato. Questo continuerà a girare e controllare il proprio stato ma non manderà in esecuzione il metodo executeStart().
- **resumeDaemon($daemon)** Permette di far ripartire un demone precedentemente messo in pausa.

Ogni metodo di itaDaemonManager può causare una Exception, risulta dunque opportuno usare tali metodi in un blocco try-catch.

```
try{
    $daemonManager->startDaemon($daemon);
}
```

```
catch(ItaException $e){
    Out::msgStop("Errore", $e->getNativeErroreDesc());
}
catch(Exception $e){
    Out::msgStop("Errore", $e->getMessage());
}
```

La console di controllo dei demoni cwbDaemonConsole sfrutta tale metodo, se necessario implementare la gestione di un demone altrove si può usare tale implementazione come esempio.

Nota: I demoni sono gestiti in modo asincrono, l'invio di un comando ad un demone non è recepito in maniera istantanea ma può richiedere diversi secondi.

Avvio tramite itaDaemonAutostart.php

Risulta sufficiente impostare l'esecuzione dello script itaDaemonAutostart.php in esecuzione automatica all'avvio del sistema operativo o wrapparlo come servizio. Lo script al suo avvio si occuperà di censire tutti i demoni presenti nel file di configurazione autostart.ini, avviare i demoni che hanno la flag autostart=true e controllare in maniera ciclica lo stato dei demoni riavviando quelli che per qualche motivo vengono terminati in maniera errata (essenzialmente a seguito di un crash).

Il controllo dei demoni gestiti da itaDaemonAutostart può essere fatto tramite lo script itaDaemonController.php. In particolare lo script accetta questa sintassi: php itaDaemonController.php <comando> <parametri addizionali> dove comando può essere:

- start
- stop
- pause
- resume

Nota: C'è possibilità di interscambio dei metodi di controllo, risulterà dunque possibile controllare tramite la classe itaDaemonManager un demone lanciato tramite itaDaemonAutostart e viceversa. Questo vale anche per il wrap dello script php in un demone o un servizio descritti di seguito.

Wrap dello script itaDaemonExecutor.php in un demone (Linux)

E' possibile wrappare un demone usando itaDaemonExecutor.php all'interno di un demone linux: Centos7: creazione di un nuovo script (/etc/systemd/system/testphp.service):

```
After=network.target

[Service]
Type=forking
User=root
ExecStart=/bin/bash /vagrant/system/start-daemon.sh par1 par2
ExecStop=/bin/bash /vagrant/system/stop-daemon.sh

[Install]
```

```
WantedBy=multi-user.target
```

start-daemon.sh:

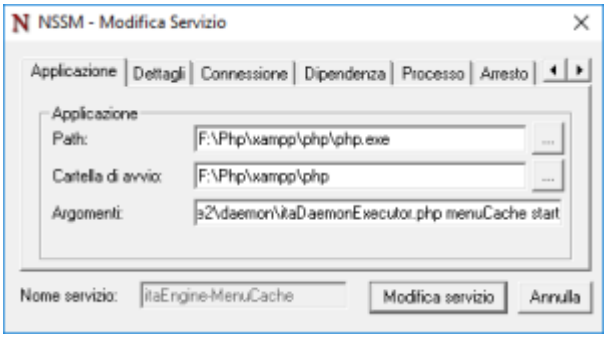
```
php /vagrant/itaEngine/daemon/itaDaemonExecutor.php dummy start $1 $2 &
```

stop-daemon.sh:

```
php /vagrant/itaEngine/daemon/itaDaemonExecutor.php dummy stop
```

Wrap dello script itaDaemonExecutor.php in un servizio (Windows)

In maniera analoga a quanto fatto per linux è possibile procedere alla registrazione di un servizio Windows usando NSSM per wrappare lo script itaDaemonExecutor.php <demone> start come nell'immagine d'esempio.



Nota: A differenza che per i metodi precedenti il wrap dello script php in un demone/servizio potrebbe causare delle incompatibilità con gli altri metodi di controllo e non è stato testato. Mentre a livello teorico il sistema dovrebbe continuare a funzionare è raccomandabile non usare l'ultimo metodo descritto per dei demoni che si desida poter gestire tramite interfaccia grafica di itaEngine.

Nota: Il modo più robusto e compatibile per lanciare e gestire i demoni di itaEngine è effettuare il wrap di itaDaemonAutostart.php in un demone/servizio e poi controllare i demoni tramite la classe itaDaemonManager o tramite la riga di comando con itaDaemonController