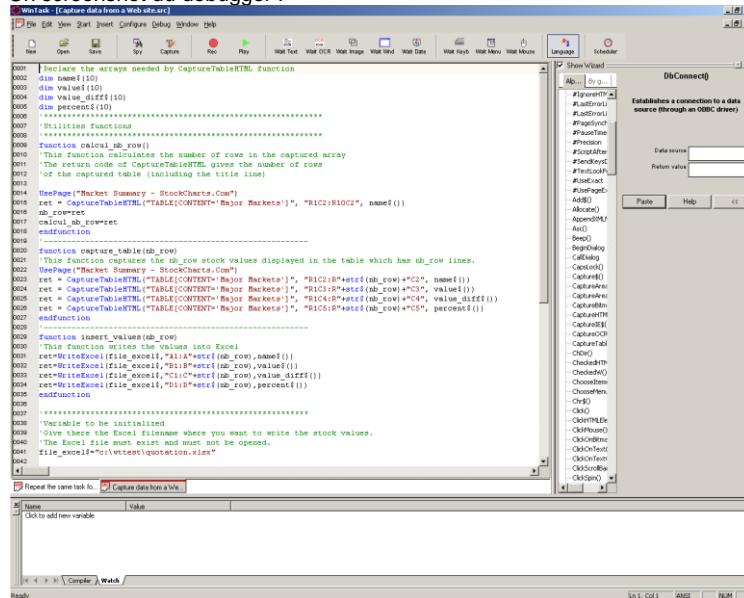
	<h2 style="margin: 0;">WinTask v3.9</h2>	
	<h3 style="margin: 0;">Windows automation et Web automation</h3>	
	<p>System Requirements: Operating system : Windows (XP/2003/2008/Vista/Seven) Browser : Internet Explorer, Mozilla Firefox, Google Chrome</p>	
	Développé par:	Taskware (France - Suresnes) www.wintask.com
Licence commerciale	Personne à contacter	jeanpierre.latour@smals.be

Fonctions

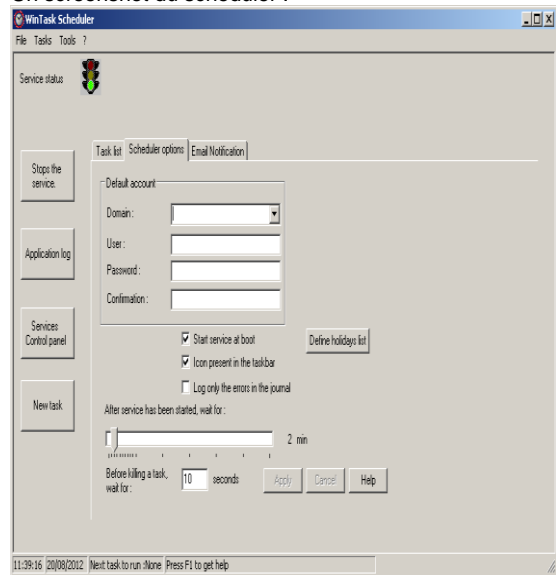
WinTask est un logiciel d'automatisation des tâches sous **Windows**. L'utilisation d'un tel outil vise à augmenter la fiabilité et la productivité des tâches répétitives, comme, par exemple :

- l'assistance à l'automatisation du **reporting** : lancer plusieurs programme de gestion, en extraire des données, agréger les données, préparer un rapport (en s'appuyant sur un outil dédié piloté par WinTask), expédier le rapport par mail (ou FTP) ;
- transférer des données entre programmes non intégrés (**data integration**) ou transférer des données d'une application Windows vers un site Web ou l'inverse (web data extract) ;
- effectuer du **mass data entry**, automatiser des **transferts FTP** ;
- effectuer des **tests de régression** d'application Windows ou d'applications Web ;
- vérifier la **disponibilité** et la **performance** d'applications Windows ou d'applications Web ;
- **installer** des logiciels (ou des périphériques tels des imprimantes) par voie de scripting ;
- créer des **macros** pour des logiciels non pourvus de cette fonctionnalité (*) ;
- planifier des **tâches d'administration** sur des serveurs ;
- ...

Un screenshot du debugger :



Un screenshot du scheduler :



(*) Dans ce cas il peut être intéressant d'utiliser l'outil gratuit Clavier+ pour lancer les scripts WinTask (qui sont au format binaire) à l'aide de séquences de touches clavier.

Fonctions (suite)

WinTask inclut un enregistreur de macros (**recorder**). Les macros enregistrées sont converties en un **langage de scripting de type objet** (style VBA). Le recorder fonctionne au niveau objet et non au niveau des coordonnées souris. Cette approche conduit à un langage de scripting concis, à la fois facilement lisible et modifiable via un éditeur intégrant un debugger. Les fonctions disponibles dans le langage de scripting sont nombreuses (Windows management, capture, user dialog, file management, system, Web, FTP, flow control, response time, real calculations, ...) et bien documentées. Chacune fait l'objet d'une **aide en ligne** (accessible par la touche F1) dans l'éditeur de scripts et d'un **wizard** lorsque la fonction s'y prête. Un **debugger** est disponible (exécutions pas à pas, point d'arrêt, visualisation du contenu des variables, ...).

Les scripts font l'objet d'une **compilation**, en vue d'en vérifier la syntaxe et d'assurer leur rapidité d'exécution. L'instruction **Include** permet de compiler plusieurs scripts (.scr) en un seul exécutable au format binaire (.rob). Ceci offre évidemment la possibilité d'élaborer des programmes WinTask de grande ampleur.

L'outil est doté d'un **module OCR**, très utile dans le cadre d'opérations de Web data extract (il peut fonctionner au choix avec le module OCR d'Office).

Les facilités d'interaction du langage de scripting avec Microsoft Excel est particulièrement appréciable. Il en est de même des possibilités liées aux fonctions **ODBC**.

Un autre point fort de l'outil concerne ses **possibilités de synchronisation** (par exemple attendre qu'une fenêtre soit ouverte avant d'y envoyer des inputs). La synchronisation est d'une importance cruciale dans le cadre d'une démarche d'automatisation.

Mentionnons encore les facilités de capture des ressources Windows en cours d'utilisation, en vue d'en récupérer les identifiants et propriétés dans l'éditeur de script.

Un **scheduler** (implémenté sous la forme d'un service Windows) permet de planifier les scripts, éventuellement de manière répétitive. Il permet des exclusions dans le calendrier. Du point de vue sécurité, il offre la possibilité de ne faire tourner un script qu'avec certains credentials Windows. Des **notifications** sont possibles via un serveur SMTP.

Conclusions / Recommandations

Dès lors que l'on veut bien attacher l'importance qu'elles méritent aux notions de productivité et de fiabilité des tâches répétitives dans le cadre Windows automation et/ou Web automation, un outil de ce type est sans aucun doute à considérer. Des alternatives existent, comme par exemple l'open source AutoHotKey, mais qui nous est apparu bien moins productif et agréable à utiliser (plus orienté programmation « système » et qui ne possède pas un véritable éditeur de script).

La question se pose de savoir si un outil tel que WinTask ne pourrait pas être utilisé avec intérêt pour les opérations de déploiement d'applications Web d'un environnement à l'autre (sous OS Windows).

Tests et Résultats

Nos tests ont démontré la facilité de prise en mains de WinTask, l'étendue de ses fonctionnalités, la qualité de sa documentation, ainsi que sa fiabilité. Un bémol : le module OCR natif nous est apparu peu efficace en regard de celui disponible dans Office.

Budget

WinTask existe en 3 déclinaisons : WinTask Lite, WinTask et WinTask64. Les différences sont données ici : <http://www.wintask.com/compare-versions.php>.

Une seule licence par poste. Les prix sont dégressifs en fonction des quantités.

WinTask Lite	\$99.00 <i>Software only</i>	WinTask	\$249.00 <i>Software only</i>	WinTask x64	\$399.00 <i>2 years upgrades*</i>
	\$119.00 <i>2 years upgrades*</i>		\$299.00 <i>2 years upgrades*</i>		