

DevTest Solutions

Installation

Version 8.0



La présente documentation, qui inclut des systèmes d'aide et du matériel distribués électroniquement (ci-après nommés "Documentation"), vous est uniquement fournie à titre informatif et peut être à tout moment modifiée ou retirée par CA.

La présente Documentation ne peut être copiée, transférée, reproduite, divulguée, modifiée ou dupliquée, en tout ou partie, sans autorisation préalable et écrite de CA. La présente Documentation est confidentielle et demeure la propriété exclusive de CA. Elle ne peut pas être utilisée ou divulguée, sauf si (i) un autre accord régissant l'utilisation du logiciel CA mentionné dans la Documentation passé entre vous et CA stipule le contraire ; ou (ii) si un autre accord de confidentialité entre vous et CA stipule le contraire.

Nonobstant ce qui précède, si vous êtes titulaire de la licence du ou des produits logiciels décrits dans la Documentation, vous pourrez imprimer ou mettre à disposition un nombre raisonnable de copies de la Documentation relative à ces logiciels pour une utilisation interne par vous-même et par vos employés, à condition que les mentions et légendes de copyright de CA figurent sur chaque copie.

Le droit de réaliser ou de mettre à disposition des copies de la Documentation est limité à la période pendant laquelle la licence applicable du logiciel demeure pleinement effective. Dans l'hypothèse où le contrat de licence prendrait fin, pour quelque raison que ce soit, vous devrez renvoyer à CA les copies effectuées ou certifier par écrit que toutes les copies partielles ou complètes de la Documentation ont été retournées à CA ou qu'elles ont bien été détruites.

DANS LES LIMITES PERMISES PAR LA LOI APPLICABLE, CA FOURNIT LA PRÉSENTE DOCUMENTATION "TELLE QUELLE", SANS AUCUNE GARANTIE, EXPRESSE OU TACITE, NOTAMMENT CONCERNANT LA QUALITÉ MARCHANDE, L'ADÉQUATION À UN USAGE PARTICULIER, OU DE NON-INFRACTION. EN AUCUN CAS, CA NE POURRA ÊTRE TENU POUR RESPONSABLE EN CAS DE PERTE OU DE DOMMAGE, DIRECT OU INDIRECT, SUBI PAR L'UTILISATEUR FINAL OU PAR UN TIERS, ET RÉSULTANT DE L'UTILISATION DE CETTE DOCUMENTATION, NOTAMMENT TOUTE PERTE DE PROFITS OU D'INVESTISSEMENTS, INTERRUPTION D'ACTIVITÉ, PERTE DE DONNÉES OU DE CLIENTS, ET CE MÊME DANS L'HYPOTHÈSE OÙ CA AURAIT ÉTÉ EXPRESSÉMENT INFORMÉ DE LA POSSIBILITÉ DE TELS DOMMAGES OU PERTES.

L'utilisation de tout produit logiciel mentionné dans la Documentation est régie par le contrat de licence applicable, ce dernier n'étant en aucun cas modifié par les termes de la présente.

CA est le fabricant de la présente Documentation.

Le présent Système étant édité par une société américaine, vous êtes tenu de vous conformer aux lois en vigueur du Gouvernement des Etats-Unis et de la République française sur le contrôle des exportations des biens à double usage et aux autres réglementations applicables et ne pouvez pas exporter ou réexporter la documentation en violation de ces lois ou de toute autre réglementation éventuellement applicable au sein de l'Union Européenne.

Copyright © 2014 CA. Tous droits réservés. Tous les noms et marques déposées, dénominations commerciales, ainsi que tous les logos référencés dans le présent document demeurent la propriété de leurs détenteurs respectifs.

Support technique

Pour une assistance technique en ligne et une liste complète des sites, horaires d'ouverture et numéros de téléphone, contactez le support technique à l'adresse <http://www.ca.com/worldwide>.

Table des matières

Chapitre 1: Préinstallation	9
Configuration système requise	9
Configuration requise pour le système d'exploitation	10
Spécification de votre propre machine virtuelle Java	13
Configuration système requise pour DevTest Server.....	15
Configuration système requise de DevTest Workstation pour CA Application Test	16
Configuration système requise de CA Service Virtualization	16
Configuration système requise pour CAI	16
Configuration requise pour la communication	17
Configuration système requise pour la base de données	18
Navigateurs pris en charge.....	20
Planification de votre système DevTest	21
Composants de DevTest.....	23
A propos des composants DevTest Server	25
Relations entre les processus DevTest.....	27
Architecture de DevTest Solutions.....	28
Architecture de CA Application Test	29
Architecture de CA Service Virtualization	32
Architecture de CAI	34
Architecture du tableau de bord Enterprise Dashboard.....	35
Composants de DevTest Server	36
Flux de données dans DevTest Server pour CA Application Test et CA Service Virtualization	39
Téléchargement des programmes d'installation de DevTest Solutions	41
Chapitre 2: Présentation de l'installation de DevTest	43
Options d'installation	44
Procédure d'installation et de configuration de DevTest Solutions	48
Fonctionnement de l'activation de la licence.....	50
Chapitre 3: Installation du DevTest Server et tâches de post-installation	51
Installation de DevTest Server sous Windows	51
Installation de DevTest Server sous UNIX	56
Installation du DevTest Server sous Mac	59
Activation des registres	63
Vérification de l'activation de registre	64
Post-installation	65

Configuration des registres existants.....	66
Utilisation d'un serveur proxy HTTP/S - DevTest Server	69
Exécution des composants sur différents systèmes	71
Calcul des instances de simulateur	72
Dimensionnement de serveur de charge et de performances	73
Utilisation de DevTest Workstation dans un environnement Java	74
Modification de l'emplacement d'accueil de projet par défaut	75
Structure des répertoires de projet	75
Désinstallation de DevTest Server.....	76
Chapitre 4: Installation de DevTest Workstation	79
Installation de DevTest Workstation sous Windows.....	80
Installation de DevTest Workstation sous UNIX.....	82
Installation de la station de travail DevTest Workstation sur un Mac	85
Utilisation d'un serveur proxy HTTP/S - DevTest Workstation	87
Paramètres d'environnement	89
Chapitre 5: Installation du serveur de démonstration	91
Chapitre 6: Vérification de l'installation de DevTest Solutions	93
Démarrage de DevTest et connexion aux interfaces utilisateur	93
Démarrage de processus ou de services DevTest	94
Connexion en tant que superutilisateur standard	98
Création d'un utilisateur avec un rôle de superutilisateur.....	100
Accès aux interfaces utilisateur de DevTest.....	101
Chapitre 7: Installation des outils d'intégration	105
Installation du moniteur de performances (PerfMon).....	106
Installation et configuration de SNMP	107
Installation de l'agent Microsoft SNMP	108
Configuration de l'agent Microsoft SNMP	109
Exécution de TCPMon	110
Utilisation de TCPMon comme intermédiaire explicite	111
Utilisation de TCPMon comme expéditeur de demande	111
Installation du module d'extension HP ALM - Quality Center.....	112
Installation d'IBM Rational Quality Manager	113
Implémentation	114
Installation de l'interface utilisateur de l'adaptateur	115
Exécution de la ligne de commande Adapter	116
Flux de travaux d'utilisation générale	116

Configuration de l'intégration à CA APM	117
Instrumentation des processus DevTest.....	118
Fichiers de l'agent Introscope	120
Mesures signalées par les outils de suivi	121
Dérivation de mesures DevTest	123
Configuration de l'outil de suivi	124
Installation de l'agent Introscope	126
Configuration de la journalisation de l'outil de suivi.....	127
Mesures standard signalées.....	127
Affichage des mesures	129
Reporting (Génération de rapports)	137
Configuration de SAP SLD (System Landscape Directory)	137
Enregistrement manuel de DevTest dans SLD	138
Importation du fichier XML SLD pour DevTest.....	139

Chapitre 8: Configuration de l'environnement de test d'application mobile 141

Configuration système requise pour le test d'applications mobiles	141
Systèmes d'exploitation pris en charge pour le test d'application mobile	141
Systèmes d'exploitation mobiles pris en charge	142
Configuration matérielle requise pour le test d'application mobile	142
Etapes de préinstallation pour le test d'application mobile	142
Etapes de préinstallation pour le test d'application mobile - Macintosh	143
Etapes de préinstallation pour les tests d'application mobile - Windows	145
Définition de la propriété ANDROID_HOME	146
Configuration du kit de développement logiciel Android	147
Utilisation de Genymotion	149
Gestion des unités Genymotion.....	150
Définition de la propriété VBOXMANAGE_CMD.....	151

Glossaire 153

Chapitre 1: Préinstallation

Ce chapitre traite des sujets suivants :

- [Configuration système requise \(page 9\)](#)
- [Planification de votre système DevTest \(page 21\)](#)
- [Architecture de DevTest Solutions \(page 28\)](#)
- [Téléchargement des programmes d'installation de DevTest Solutions \(page 41\)](#)

Configuration système requise

Cette section répertorie la configuration requise pour les différents composants de DevTest. Cette configuration générale peut varier, selon l'étendue de votre projet.

- [Configuration requise pour le système d'exploitation \(page 10\)](#)
- [Spécification de votre propre machine virtuelle Java \(page 13\)](#)
- [Configuration système requise pour DevTest Server \(page 15\)](#)
- [Configuration système requise pour la station de travail DevTest Workstation pour CA Application Test \(page 16\)](#)
- [Configuration système requise pour CA Service Virtualization \(page 16\)](#)
- [Configuration requise pour la communication \(page 17\)](#)
- [Configuration système requise pour la base de données \(page 18\)](#)
- [Navigateurs pris en charge \(page 20\)](#)

Configuration requise pour le système d'exploitation

Les systèmes d'exploitation suivants sont pris en charge :

- Microsoft Windows :
 - Windows Server 2012
 - Windows 7
 - Windows 8 (avec le dernier service pack et toutes les mises à jour critiques appliqués)
- Linux et UNIX
 - Fedora 19
 - Red Hat Enterprise Linux 6.3
 - SUSE Linux 10 SP2, 11.x
 - Ubuntu 11.04, 12.04, 13.x
 - Oracle Solaris 10 et 11
 - IBM AIX 6.1, 7.0
- Mac OS X 10.9 et 10.10

Il est recommandé d'utiliser un système d'exploitation 64 bits et un JRE, notamment pour le DevTest Server.

Configuration système requise pour la machine virtuelle Java du DevTest Server

Il est recommandé d'utiliser Java 7 64 bits pour DevTest Server lorsque des tailles de segment de mémoire supérieures à 2 G sont requises.

I>JDK Java

Les produits composant DevTest Solutions sont des applications Java. Un JRE Oracle est fourni avec chaque programme d'installation spécifique aux plates-formes (JRE 1.7.0_17 comprenant un fichier **tools.jar** du JDK), hormis le programme d'installation générique UNIX. JRE IBM version 7 ; la version 1 est également prise en charge si vous [fournissez votre propre machine virtuelle Java](#) (page 13).

La version Java minimum prise en charge pour DevTest Workstation, DevTest Server et VSE est la version 1.7 mise à jour 6.

Cette condition concerne DevTest uniquement. L'exécution de DevTest dans une machine virtuelle Java 1.7 peut être utilisée pour tester des applications sur des serveurs d'applications exécutés avec un JRE antérieur ou ultérieur.

Le tableau suivant répertorie le support fourni pour les différents JDK :

	DevTest Workstation	DevTest Server - coordinateur, simulateur, registre	DevTest Server - VSE	Agent Java pour DevTest (CAI)
JDK 1.5	Non pris en charge	Non pris en charge	Non pris en charge	Pris en charge
JDK 1.6	Non pris en charge	Non pris en charge	Non pris en charge	Pris en charge
JDK 1.7	Requis pour l'exécution	Requis pour l'exécution	Requis pour l'exécution	Pris en charge

Remarque : DevTest ne prend pas en charge le JRE IBM version 8, le JRE Oracle 1.8, ni OpenJDK.

L'authentification Kerberos n'est pas prise en charge sur les JRE IBM.

DevTest Workstation ne peut pas tester les services Web et les dorsales de messagerie sur différentes versions de Java.

Les informations suivantes fournissent des orientations générales sur la prise en charge du JDK dans l'agent Java pour DevTest (CAI) :

- CAI prend en charge des systèmes basés sur les kits JDK 1.5, 1.6 et 1.7.
- Seuls les JDK d'Oracle sont intégralement pris en charge. La prise en charge des JDK d'IBM est limitée.

Verrouillage de dossier par DevTest

Le dossier suivant disponible dans le répertoire LISA_HOME requiert des droits d'accès en lecture/écriture :

- locks (verrrous)

En général, vous pouvez limiter le reste du répertoire LISA_HOME à l'aide de droits en lecture seule :

Dossier lisatmp

Le dossier suivant dans le répertoire de base de l'utilisateur sur UNIX, Linux et OS X ou le répertoire Documents and Settings sur Windows requiert des droits en lecture et en écriture :

- lisatmp_x.x (s'il existe.)

Agent CICS pour DevTest

L'agent CICS pour DevTest prend en charge les versions 3.2, 4.1 et 4.2 de CICS.

Spécification de votre propre machine virtuelle Java

Le programme d'installation UNIX générique n'inclut aucun JRE. Si vous utilisez le programme d'installation générique UNIX, vous devez fournir votre propre machine virtuelle Java.

Vous pouvez également utiliser cette procédure avec un programme d'installation d'une autre plate-forme pour remplacer le JRE inclus. Reportez-vous à la section [Utilisation de DevTest Workstation avec votre environnement Java](#) (page 74).

Remarque : DevTest prend en charge le JRE IBM 7, Version 1. DevTest ne prend pas en charge le JRE IBM 8, le JRE Oracle 1.8, ni OpenJDK.

Procédez comme suit:

1. Téléchargez le package du [JDK SE Java \(kit de développement Java\) 7](#) pour votre plate-forme à partir du site Web d'Oracle.
2. Téléchargez les fichiers [Java Cryptography Extension \(JCE\) Unlimited Strength Jurisdiction Policy Files 7](#) à partir du site Web d'Oracle.
3. Installez le JDK 7 pour votre système d'exploitation sur l'ordinateur sur lequel vous prévoyez d'installer DevTest. S'il n'existe aucun répertoire **Java**, créez-en un (mkdir java).
Par exemple, installez le JDK 7 sous \usr\java, qui crée le répertoire \usr\java\jdk1.7.0_67 (JDK_HOME).
4. Une fois le fichier tar.gz jdk-7u67-platform copié, saisissez la commande suivante :
`tar zxvf jdk-7u67-platform-x64.tar.gz`
5. Définissez les variables d'environnement :
 - Définissez la variable d'environnement **JDK_HOME** sur le répertoire dans lequel vous avez installé le JDK 7.
 - Définissez la variable d'environnement **JAVA_HOME** sur le répertoire **JDK_HOME**.

Par exemple :

```
cd \usr\java\jdk1.7.0_67  
pwd  
export JAVA_HOME=$PWD  
export JDK_HOME=$PWD
```

6. Effectuez l'extraction du dossier UnlimitedJCEPolicy à partir du fichier UnlimitedJCEPolicyJDK7.zip. Déplacez les fichiers JAR suivants à partir de ce dossier vers le répertoire **JDK_HOME\jre\lib\security** :
 - local_policy.jar
 - US_export_policy.jar

Cette action remplace les fichiers JAR existants portant les mêmes noms.

7. Copiez le fichier **tools.jar** du répertoire **JDK_HOME\lib** vers le répertoire **JDK_HOME\jre\lib\ext**.

```
$JDK_HOME\jre\lib\ext cd  
cp $JDK_HOME\lib\tools.jar.
```

8. Si vous remplacez le JRE inclus, effectuez les étapes supplémentaires suivantes après l'installation de DevTest :

- a. Définissez la variable d'environnement **LISA_JAVA_HOME**. Par exemple :

```
cd \usr\java\jdk1.7.0_67  
pwd  
export LISA_JAVA_HOME=$PWD
```

- b. Remplacez le nom du répertoire **jre** sous le répertoire d'installation de DevTest par **jre_default** tel qu'indique à la section [Utilisation de DevTest Workstation \(Station de travail\) dans un environnement Java](#) (page 74).

Configuration système requise pour DevTest Server

CA Application Test, CA Service Virtualization et CA Continuous Application Insight requièrent le registre dans le DevTest Server.

La configuration minimum requise pour le DevTest Server est la suivante :

- **UC** : 2 GHz ou plus, 4 noyaux minimum
- **Mémoire RAM** : 4 Go
- **Espace disque** : 50 Go
- **Base de données** : reportez-vous à la rubrique [Configuration système requise pour la base de données](#) (page 18). La base de données peut résider sur un système différent et doit disposer de 200 Go minimum de stockage.

La configuration recommandée pour le DevTest Server est la suivante :

- **UC** : 2 GHz ou plus, 8 noyaux minimum
- **Mémoire RAM** : 8 Go
- **Espace disque** : 50 Go
- **Base de données** : reportez-vous à la rubrique [Configuration système requise pour la base de données](#) (page 18). La base de données peut résider sur un système différent et doit disposer de 500 Go minimum d'espace de stockage.

Pour les tests de charge et de performance, les ressources suivantes sont recommandées :

- 250 utilisateurs virtuels pour chaque simulateur
- 1 noyau de processeur et 2 Go de mémoire RAM pour chaque simulateur

Exemple pour 4 000 utilisateurs virtuels parallèles : 16 simulateurs, 16 noyaux de processeur ; 32 Go de mémoire RAM (pour DevTest)

Pour chaque centre de données :

- 1 registre de test et 1 coordinateur
- 1 noyau de processeur par processus = 2 noyaux de processeur
- 2 Go de mémoire RAM/chacun = 4 Go (pour DevTest)

La configuration de base pour DevTest Server est composée d'un tableau de bord d'entreprise, d'un registre et d'un portail, requis pour l'ensemble des produits. CA Application Test requiert un serveur de coordination et un serveur de simulation. CA Service Virtualization requiert un environnement de serveur virtuel. CA Continuous Application Insight requiert un intermédiaire. Un seul serveur de tableau de bord d'entreprise est requis pour chaque configuration spécifique.

Remarque : La configuration requise indiquée ici doit être utilisée comme référence. Dans le cas d'environnements de charge étendus, il est recommandé de contacter des services professionnels pour faciliter leur développement en fonction de vos besoins.

Configuration système requise de DevTest Workstation pour CA Application Test

La configuration minimum requise pour DevTest Workstation est la suivante :

- **UC** : 2 GHz ou plus, 2 noyaux minimum
- **Mémoire RAM** : 4 Go
- **Espace disque** : 2 Go d'espace disque disponible

Configuration système requise de CA Service Virtualization

La maintenance d'un environnement de virtualisation requiert le composant VSE de CA Service Virtualization. Le VSE est un service de niveau serveur qui peut coexister avec un registre disposant d'un coordinateur et d'un simulateur qui y sont connectés. Le simulateur et le coordinateur ne sont pas requis pour l'exécution du VSE.

La configuration requise suivante constitue la configuration requise de référence uniquement :

- **UC** : 2 GHz ou plus, 2 noyaux minimum
- **Mémoire RAM** : 2 Go pour le VSE, en plus de la mémoire RAM pour DevTest Workstation, DevTest Server, ou CAI
- **Espace disque** : espace disque disponible de 50 Go
- **Base de données** : reportez-vous à la rubrique [Configuration système requise pour la base de données](#) (page 18). La base de données peut résider sur un système différent et doit disposer de 10 Go minimum d'espace de stockage.

Configuration système requise pour CAI

Configuration système requise pour CA Continuous Application Insight :

- **UC** : 2 GHz ou plus, 8 noyaux minimum
- **Mémoire RAM** : 8 Go
- **Espace disque** : 50 Go d'espace de stockage sur disque local
- **Base de données** : reportez-vous à la rubrique [Configuration système requise pour la base de données](#) (page 18). La base de données peut résider sur un système différent et doit disposer de 500 Go minimum d'espace de stockage.

Configuration requise pour la communication

Veillez à ce que votre pare-feu permette permettent l'envoi et la réception des transmissions réseau par DevTest Solutions. La fonctionnalité fournie par DevTest Solutions requiert l'accès aux ressources réseau et ne fonctionnera pas correctement si elle est bloquée par un pare-feu. Autorisez les applications de DevTest Solutions.

Remarque : Pour implémenter la communication sécurisée, reportez-vous à la section Utilisation du protocole SSL pour sécuriser la communication et Utilisation de la communication HTTP/S avec la console DevTest de la rubrique *Administration*.

Les communications à partir des ports DevTest et vers ceux-ci doivent être ouvertes dans un pare-feu approprié. Reportez-vous aux sections de la rubrique *Administration*.

- Numéros de port par défaut de DevTest Server
- Numéros de port par défaut de DevTest Workstation
- Numéros de port par défaut du serveur de démonstration

Configuration système requise pour la base de données

Les composants suivants permettent de stocker des informations sur une base de données :

- **DevTest Server** : la base de données est utilisée pour obtenir des résultats de rapport, que vous pouvez exporter vers d'autres formats si nécessaire. La base de données est également utilisée pour le contrôle d'accès (liste de contrôle d'accès).
- **VSE** : la base de données est utilisée pour les nombres d'utilisation et les images de service virtuel héritées.
- **CAI** : la base de données est utilisée pour les chemins, y compris des données de demande et de réponse, des instructions SQL et des journaux d'application. La base de données est également utilisée pour les tickets.
- **Enterprise Dashboard (Tableau de bord d'entreprise)** : la base de données est utilisée pour les données de rapport d'audit de l'utilisation de DevTest Solutions, d'autres informations de registre, les journaux d'événements historiques et les mesures.

Important : Le tableau de bord Enterprise Dashboard requiert sa propre base de données unique et volumineuse. La base de données peut résider sur un système différent et doit disposer de 50 Go minimum d'espace de stockage. IBM DB2 n'est pas pris en charge comme base de données pour le tableau de bord Enterprise Dashboard.

Par défaut, ces composants utilisent une base de données Apache Derby fournie avec DevTest. Cette base de données, uniquement adéquate pour des déploiements de petite taille qui ne requièrent pas de test de charge et de performance, n'est pas prise en charge. Pour tous les autres scénarios, configurez DevTest pour utiliser une base de données externe.

Pour garantir des performances correctes lors de l'utilisation d'une base de données externe, le serveur de base de données et le serveur de DevTest doivent disposer d'une bande passante de réseau élevée et un niveau de latence faible.

Dans le cas d'une configuration distribuée, DevTest dépend en grande partie des composants de serveur disposant d'une connexion haut débit et à latence faible à une base de données professionnelle correctement maintenue.

Tous les composants de serveur de DevTest communiquent directement avec la base de données pour enregistrer leurs actions et toute restriction appliquée au flux de ces données aura un impact négatif.

Afin d'assurer le fonctionnement de votre DevTest, le temps d'aller-retour de tous les composants de serveur DevTest à l'hôte de base de données doit être inférieur à 20 ms. Si la latence de réseau dépasse cette valeur de 20 ms, vous êtes susceptible de rencontrer des problèmes de performances.

Important : Utilisez un schéma de base de données propre pour toute nouvelle installation. Avant l'installation, vous pouvez restaurer des données à partir de la même version de DevTest dans le schéma propre. N'utilisez pas de données d'autres versions.

Les bases de données externes suivantes sont prises en charge :

- **DB2 10.1 ou 10.5 d'IBM** (non pris en charge pour le tableau de bord Enterprise Dashboard) - la page de codes de la base de données doit être 1208. De plus, la taille de page doit être de 8 Ko minimum.
- **MySQL 5.5 ou 5.6** - la base de données MySQL doit fournir un classement et un ensemble de caractères en prenant en charge le codage UTF-8 ; les caractères codés à deux octets sont stockés dans la liste de contrôle d'accès et les tables de génération de rapports. La page de codes par défaut pour la base de données doit être codée en UTF-8 ; la définition seule de votre base de données en UTF-8 n'est pas suffisante.
- **11g version 2 ou 12c d'Oracle** - l'ensemble de caractères doit être codé en Unicode.
- **Microsoft SQL Server 2008 R2 ou 2012**

Le schéma est automatiquement créé dans la base de données externe lors du premier démarrage du registre. Avant la création du schéma, vérifiez que l'utilisateur de DevTest dispose de droits d'administration des bases de données (DBA). Une fois le schéma créé, vous pouvez supprimer les droits de DBA de l'utilisateur.

Si votre stratégie de sécurité ne permet pas cette approche, l'administrateur de base de données peut créer manuellement le schéma. Les fichiers DDL dans le répertoire **LISA_HOME\database** contiennent des instructions SQL pour créer les tables de génération de rapports et des index, ainsi que le schéma de base de données d'agent utilisé par CA Continuous Application Insight. Fournissez ces informations à l'administrateur de base de données.

Remarque : Pour plus d'informations sur la configuration d'une base de données externe, consultez la rubrique *Administration*. Pour plus d'informations sur l'obtention d'instructions SQL pour le schéma de base de données d'agent, reportez-vous à la section *Agents*.

Important : Chaque registre et tableau de bord Enterprise Dashboard doit disposer d'un schéma unique. Ne pointez pas plusieurs registres vers le même schéma de base de données.

Pour les tests de charge et de performances, adaptez la base de données externe de sorte à assurer la prise en charge de l'espace de stockage des données requis par DevTest.

Le registre, le coordinateur, les simulateurs et chaque environnement de service virtuel requièrent un accès hautes performances à la base de données. Les données de performances sont enregistrées directement dans la base de données par ces composants. CA recommande que la base de données soit présente dans le même centre de données. Les bases de données hébergées par une machine virtuelle ne sont pas recommandées pour l'utilisation de la disponibilité générale.

Navigateurs pris en charge

DevTest inclut les tableaux de bord, les portails et les consoles Web et suivants :

- Tableau de bord Enterprise Dashboard (<http://nom-hôte:1506/>)
- Portail DevTest (<http://nom-hôte:1507/devtest>)
- Console DevTest (<http://nom-hôte:1505/>)
 - Reporting Portal (Portail de rapports)
 - Continuous Validation Service (Service de validation en continu)
 - Server Console (Console de serveur)
 - VSEasy
 - Console CAI (redirige vers le portail DevTest)
- Demo Server (Serveur de démonstration) (<http://nom-hôte:8080/lisabank/>)

Le portail DevTest requiert l'application HTML 5. Par conséquent, la dernière technologie de navigateur Internet est requise pour utiliser la nouvelle interface utilisateur Web. Les versions de navigateur Internet suivantes prennent en charge l'application HTML 5 :

- Google Chrome 36
- Mozilla FireFox 30
- Apple Safari 7.0
- Microsoft Internet Explorer 11

Remarque : La prise en charge des navigateurs pour l'enregistrement des tests d'intégration de Selenium est limitée à Mozilla Firefox. Une fois ces tests importés dans DevTest, l'exécution des scénarios de test a été vérifiée sur Google Chrome, Mozilla FireFox 24, ou Internet Explorer 10 et 11.

Remarque : La prise en charge du navigateur Internet Explorer pour VSEasy est limitée à la version 10. Tous les autres navigateurs répertoriés sont pris en charge.

Planification de votre système DevTest

Pour installer DevTest Solutions, tenez préalablement compte des points de décision suivants :

- Quel système d'exploitation pris en charge envisagez-vous d'utiliser ?
- Effectuez-vous une installation partagée ou locale ?
- L'installation est-elle destinée à être utilisée par une station de travail ou des serveurs ?
- Quelle base de données d'entreprise prévoyez-vous d'utiliser ?
Dans le cas d'une installation permanente ou si vous prévoyez de l'utiliser dans un environnement généralement disponible, la base de données Derby fournie n'est pas prise en charge.
- Prévoyez-vous d'installer le serveur de démonstration ?
Le serveur de démonstration est fourni séparément dans le programme d'installation de produit.

Avant de démarrer une installation, le fichier de licence (devtestlic.xml) doit être disponible et présent sur votre système cible.

Remarques supplémentaires sur la disposition idéale des divers services et processus dans les systèmes cibles :

- Le tableau de bord Enterprise Dashboard doit être accessible par le registre, mais pas nécessairement de manière locale. Le tableau de bord Enterprise Dashboard requiert sa propre base de données unique.
- Le registre et le coordinateur doivent résider sur le même ordinateur, lorsque cela est possible.
- Les simulateurs doivent être sur le même segment de réseau que le coordinateur, si cela est possible.
- Le registre, le coordinateur, les simulateurs et chaque environnement de service virtuel requièrent un accès hautes performances à la base de données. Les données de performances sont enregistrées directement dans la base de données par ces composants. CA recommande que la base de données soit présente dans le même centre de données. Les bases de données hébergées par une machine virtuelle ne sont pas recommandées pour l'utilisation de la disponibilité générale.

Pour plus d'informations sur les configurations requises pour la base de données, consultez la rubrique [Configuration requise pour la base de données](#) (page 18).

Avant de commencer la planification de votre système DevTest, consultez les Notes de parution de la version que vous installez. Si vous effectuez une migration de système, consultez le *Manuel de migration*.

Une partie de la planification d'un système DevTest consiste à décider du nombre de DevTest Server à installer et des processus ou services à exécuter sur chaque ordinateur hôte contenant un DevTest Server.

Lorsque vous téléchargez les programmes d'installation DevTest Solutions, vous pouvez sélectionner l'un des trois produits, tous pointant vers les mêmes programmes d'installation. La section [Composants de DevTest](#) (page 23) décrivent de manière graphique ces trois produits avec des interfaces utilisateur et des processus spécifiques aux produits et l'interface utilisateur commune et les processus partagés par tous les produits. La section [A propos des composants du DevTest Server](#) (page 25) décrit l'objectif de chaque processus et les sections dans lesquelles d'autres détails sont disponibles.

La section [Exemples de relations dans un système distribué](#) (page 27) introduit des concepts majeurs sur les processus que vous pouvez exécuter sur chaque hôte dans un système distribué. La section [Post-installation](#) (page 65) traite les rubriques qui peuvent vous aider à adapter votre système distribué, notamment :

- [Exécution des composants sur différents systèmes](#) (page 71)
- [Calcul des instances de simulateur](#) (page 72)
- [Dimensionnement de serveur de charge et de performances](#) (page 73)

Un autre aspect de la planification consiste à concevoir des conventions d'attribution de nom pour la [structure de répertoires de votre projet](#) (page 75).

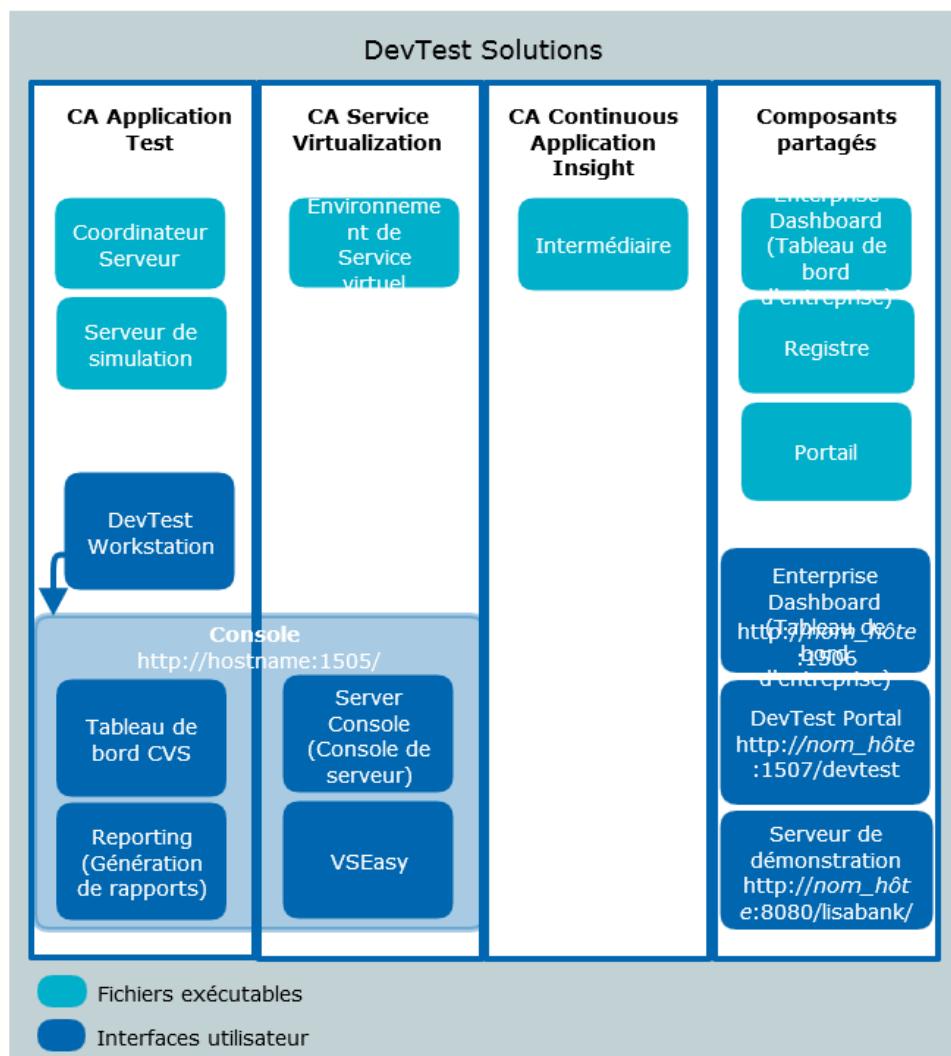
Vous pouvez utiliser le JRE fourni avec DevTest ou installer votre propre JRE. Pour plus d'informations, reportez-vous à la section [Utilisation de la station de travail DevTest Workstation dans un environnement Java](#) (page 74).

Composants de DevTest

Le programme d'installation de DevTest Solutions installe les produits suivants lors de l'installation d'un DevTest Server (Nommé *Server* dans l'assistant d'installation) :

- CA Application Test
- CA Service Virtualization
- CA Continuous Application Insight

Ces trois produits ont chacun leurs propres fichiers exécutables (et services correspondants). Il ne s'agit toutefois pas de produits indépendants, car ils requièrent tous le même fichier exécutables pour le registre. Auparavant, chaque produit disposait de sa propre interface utilisateur. Désormais, le portail DevTest devient l'interface utilisateur que partagent tous les produits. Certaines fonctionnalités sont accessibles uniquement à partir de la station de travail DevTest Workstation ou de la console DevTest.



[La section A propos des composants DevTest Server](#) (page 25) décrit chaque composant exécutable ainsi que les sections dans lesquelles vous trouverez plus d'informations.

[La section Démarrage des composants de serveur](#) (page 94) spécifie l'ordre dans lequel lancer les fichiers exécutables.

[La section Connexion aux interfaces utilisateur](#) (page 101) inclut les procédures permettant d'accéder à chaque interface utilisateur.

Pour obtenir des détails sur la configuration et la maintenance DevTest Solutions, reportez-vous à la rubrique *Administration*.

Pour des détails spécifiques aux produits, consultez les rubriques suivantes :

- *Utilisation de CA Application Test*
- *Utilisation de CA Service Virtualization*
- *Utilisation de CA Continuous Application Insight*

A propos des composants DevTest Server

Démarrez DevTest Server en lançant une série de processus ou de services. Les trois composants que vous démarrez en premier (Tableau de bord Enterprise Dashboard, registre et portail) prennent en charge DevTest Solutions. Les autres composants sont spécifiques au produit. Pour des détails sur le démarrage, reportez-vous à la rubrique [Démarrage des composants de serveur](#) (page 94).

nom	Requis par	Requis pour
Enterprise Dashboard (Tableau de bord d'entreprise)	Registres (Tous les produits de DevTest Solutions)	Démarrer DevTest Solutions avec la licence de produit Surveiller les activités de l'entreprise Générer le rapport d'audit d'utilisation
Registre	Tous les produits de DevTest Solutions	Enregistrer les composants de DevTest Server et DevTest Workstation. Le registre est le concentrateur central ou le moteur de tous les processus. Collecter des données d'utilisation pour le rapport d'audit d'utilisation
Portail	Tous les produits de DevTest Solutions	Lancer l'interface utilisateur du portail DevTest
Intermédiaire	CA Continuous Application Insight (CAI)	Coordonner les agents Java et les consoles CA Continuous Application Insight

nom	Requis par	Requis pour
Coordinateur	CA Application Test	Coordonner l'exécution des tests sur des simulateurs
Simulator (Simulateur)	CA Application Test	Exécuter des tests
VSE (Environnement de service virtuel)	CA Service Virtualization	Créer et déployer des services virtuels

Les rubriques référencées fournissent plus d'informations sur chaque processus prenant en charge le DevTest Server.

Processus	Référence
Enterprise Dashboard (Tableau de bord d'entreprise)	Fenêtre principale du tableau de bord Enterprise Dashboard dans la rubrique <i>Administration</i>
Registre	Registre dans la rubrique <i>Utilisation de CA Application Test</i>
Portail	Utilisation du portail DevTest dans la rubrique <i>Prise en main</i>
Coordinateur	Serveur de coordination dans la rubrique <i>Utilisation de CA Application Test</i>
Simulator (Simulateur)	Serveur de simulation dans la rubrique <i>Utilisation de CA Application Test</i>
Station de travail	DevTest Workstation dans la rubrique <i>Utilisation de CA Application Test</i>
Virtual Service Environment (Environnement de service virtuel)	Préparation de la virtualisation dans la rubrique <i>Utilisation de CA Service Virtualization</i>
Intermédiaire	Démarrage de l'intermédiaire dans la rubrique <i>Agents</i>

Relations entre les processus DevTest

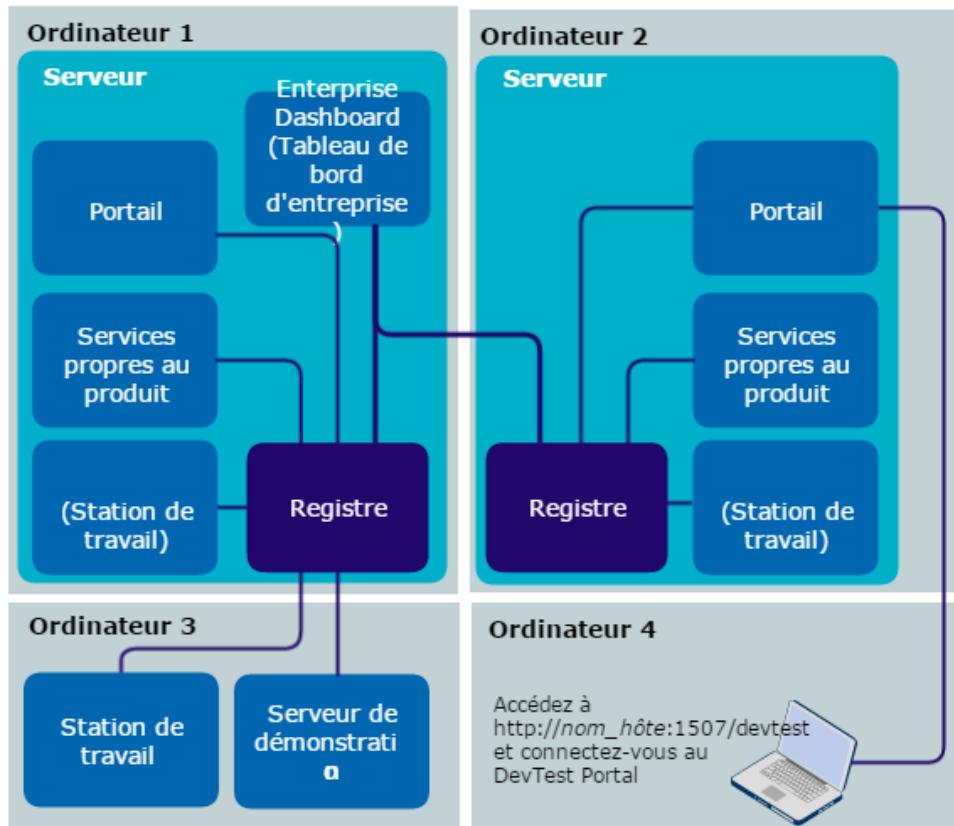
Le registre est au centre de tous les systèmes DevTest. En général, le programme d'installation est exécuté sur plusieurs ordinateurs avec différentes sélections. Prenons l'exemple suivant :

- Ordinateur 1 : composants de tableau de bord Enterprise Dashboard et de serveur et sur lequel la station de travail intégrée peut ne pas être utilisée
- Ordinateur 2 : Serveur sur lequel le composant de registre est lié au tableau de bord Enterprise Dashboard sur l'ordinateur 1. En général, un seul registre est nécessaire dans un système distribué. Ce deuxième registre est ajouté ici dans le but de démontrer que plusieurs registres sont pris en charge.
- Ordinateur 3 : station de travail et serveur de démonstration avec des liens au registre sur l'ordinateur 1. Cet ordinateur représente les ordinateurs d'utilisateur.
- Ordinateur 4, qui peut également représenter des ordinateurs d'utilisateur, notamment d'utilisateurs de CA Continuous Application Insight qui requièrent uniquement l'accès au portail.
- Vous pouvez également installer le serveur DevTest sur un ordinateur pour exécuter un seul service spécifique au produit, comme l'exécutable ou le service du simulateur.

Cet exemple de système partage les caractéristiques suivantes avec un système DevTest :

- Chaque système de DevTest Solutions possède un seul tableau de bord Enterprise Dashboard.
Remarque : Vous devez disposer d'un tableau de bord Enterprise Dashboard par réseau accessible. Si vous disposer de réseaux fermés (non accessibles entre eux), un tableau de bord Enterprise Dashboard est nécessaire pour ceux dans lesquels les registres sont exécutés.
- Chaque tableau de bord Enterprise Dashboard est connecté à un ou plusieurs registres. En général, un seul registre suffit.
- L'opération d'installation de chaque DevTest Server inclut un registre, le composant qui audite l'activité de tous les utilisateurs.
- L'opération d'installation de chaque DevTest Server inclut une station de travail locale. Cette station de travail intégrée est utilisée dans une installation autonome. Si le DevTest Server est installé dans un environnement distribué, l'utilisation de la station de travail locale n'est pas obligatoire.
- Chaque registre de DevTest Server dans un système distribué peut être connecté à une ou plusieurs stations de travail distantes.
- Un navigateur Web peut accéder au portail DevTest via un DevTest Server sur le port 1507 avec une URL terminant par devtest (en minuscule). Vous pouvez accéder au portail à partir d'un navigateur Web sans installer d'autre logiciel de DevTest Solutions sur l'ordinateur.

Le diagramme suivant présente les relations entre le tableau de bord Enterprise Dashboard, les registres, la station de travail et le serveur de démonstration dans un système distribué. Il montre également que les utilisateurs sans composant DevTest local peuvent accéder au portail DevTest.



Architecture de DevTest Solutions

Informations complémentaires :

[Composants de DevTest Server](#) (page 36)

[Flux de données dans DevTest Server pour CA Application Test et CA Service](#)

[Virtualization](#) (page 39)

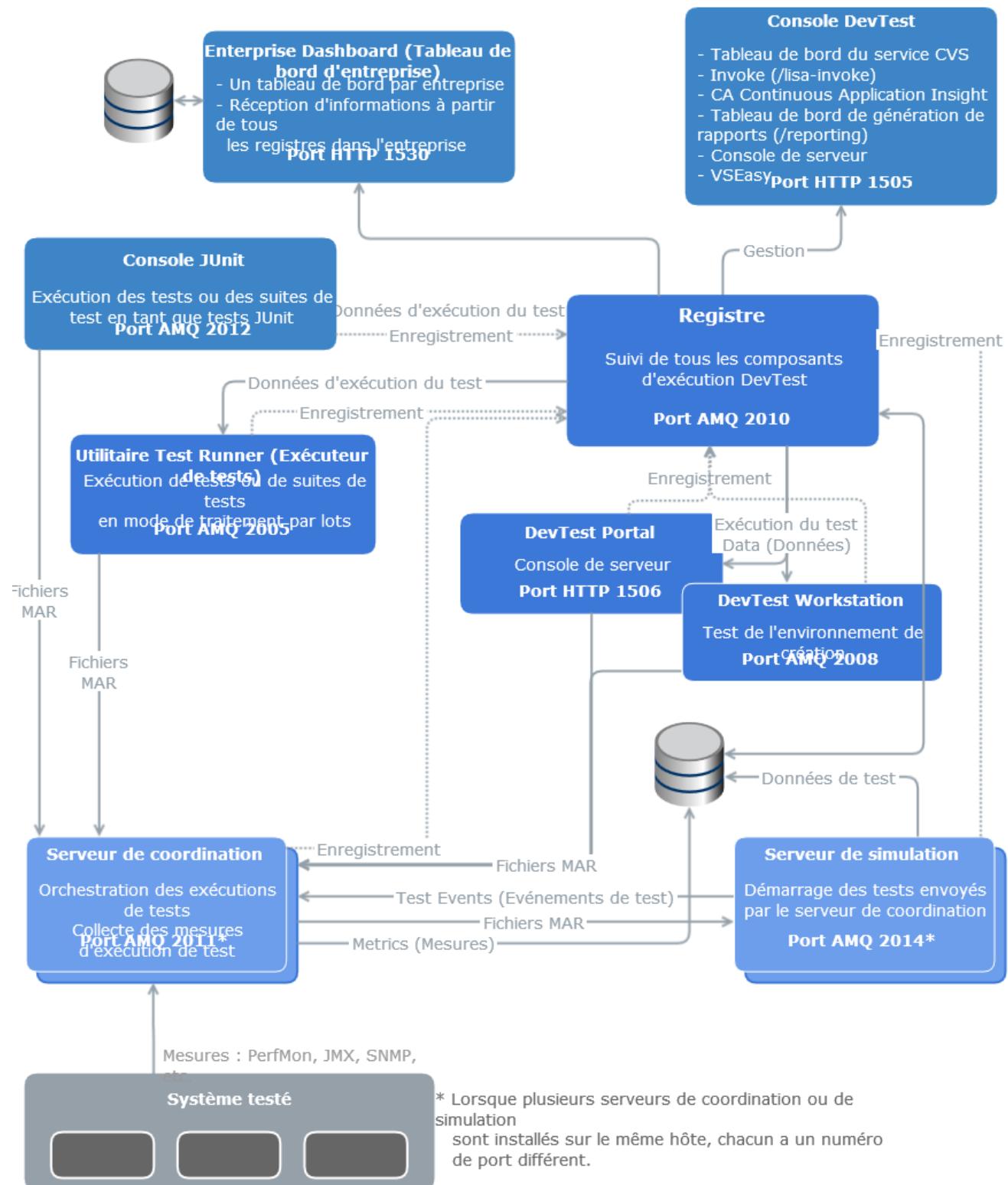
Architecture de CA Application Test

CA Application Test utilise les composants du DevTest Server suivants :

- Registre
- DevTest Workstation
- Portail DevTest
- Coordinator server (Serveur de coordination)
- Serveur de simulation

Le portail DevTest et la station de travail DevTest Workstation sont utilisés pour créer et surveiller les tests, mais les scénarios de test sont exécutés dans l'environnement du DevTest Server. Un serveur de coordination et un serveur de simulation sont intégrés à DevTest Workstation.

Le diagramme ci-après illustre l'architecture de CA Application Test :



Tous les tests sont exécutés par les utilisateurs virtuels (ou simulateurs) générés par le serveur de simulation sous la surveillance d'un coordinateur dans le serveur de coordination.

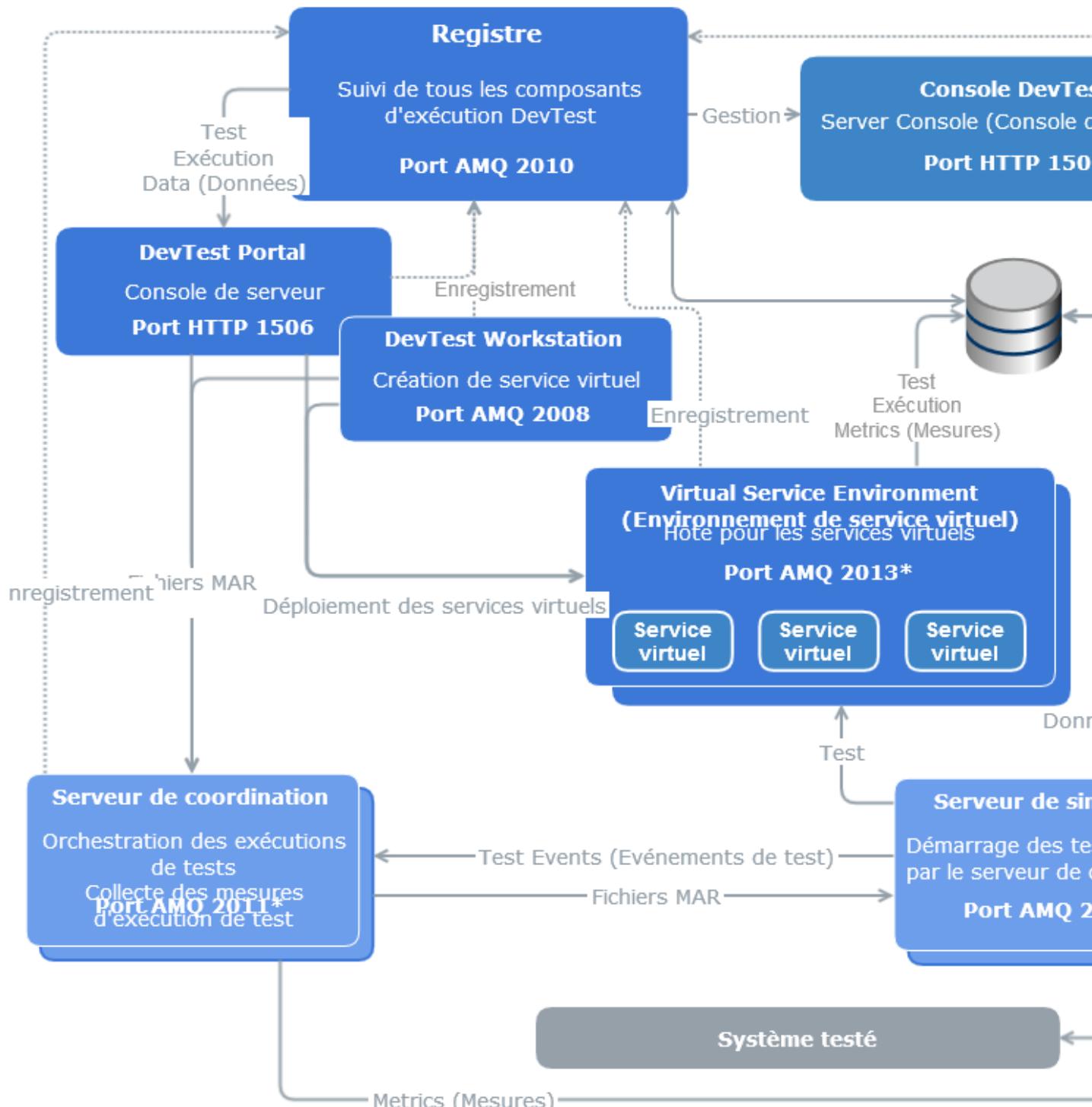
Chaque simulateur se connecte au système testé et appelle des actions. Un test de charge se produit lorsque les utilisateurs virtuels sont exécutés en mode simultané.

Une instance intégrée de la station de travail DevTest Workstation est exécutée sur le même ordinateur sur lequel le serveur DevTest Server est exécuté. Vous pouvez également configurer des stations de travail DevTest Workstation autonomes à exécuter sur différents ordinateurs dans un environnement distribué étendu. Les conditions préalables requises pour le test déterminent l'architecture de serveur à utiliser. Pour mettre en échelle DevTest Solutions dans des environnements de test étendus, répartissez les différents composants dans plusieurs systèmes matériels/d'exploitation.

Les produits de DevTest incluent CA Application Test, CA Service Virtualization et CA Continuous Application Insight ; chacun d'eux se connecte à un registre DevTest central. Les utilisateurs de CA Application Test utilisent une station de travail DevTest Workstation intégrée avec son serveur de coordination facultatif et ses serveurs de simulation et toute station de travail DevTest Workstation installée à distance. Les composants de tous les produits DevTest se connectent à un registre central.

Architecture de CA Service Virtualization

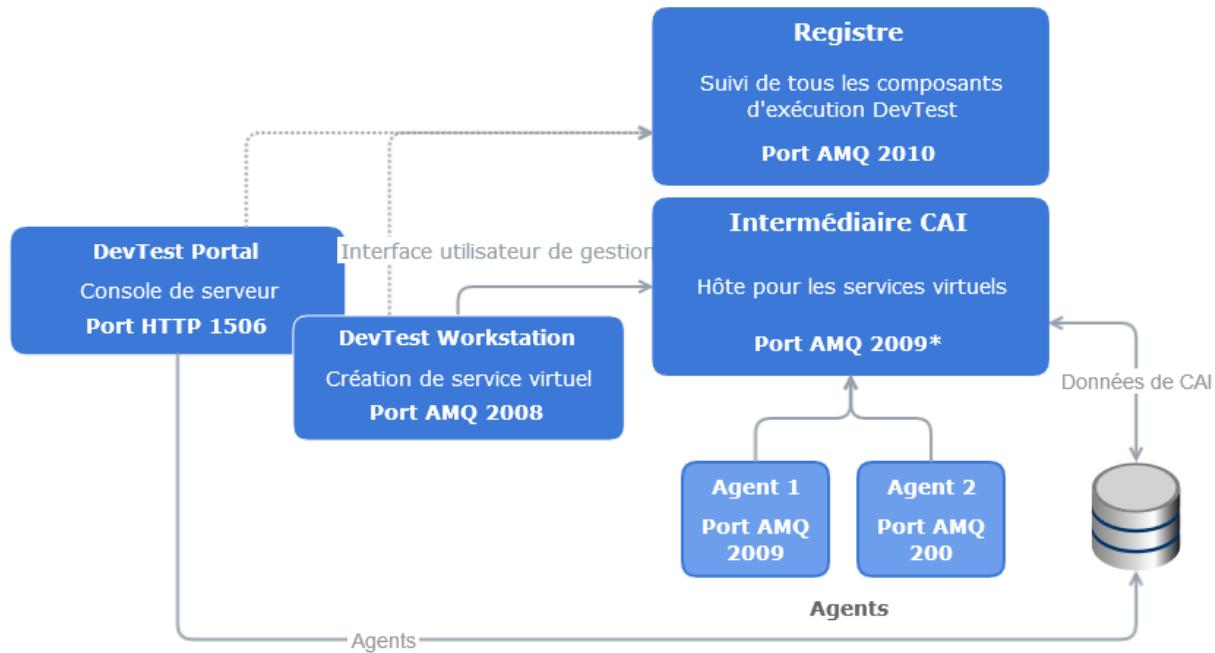
Le diagramme ci-après illustre l'architecture de CA Service Virtualization.



* Lorsque plusieurs coordinateurs, simulateurs et VSE sont installés sur un ordinateur identique, ils auront des numéros de port différents.

Architecture de CAI

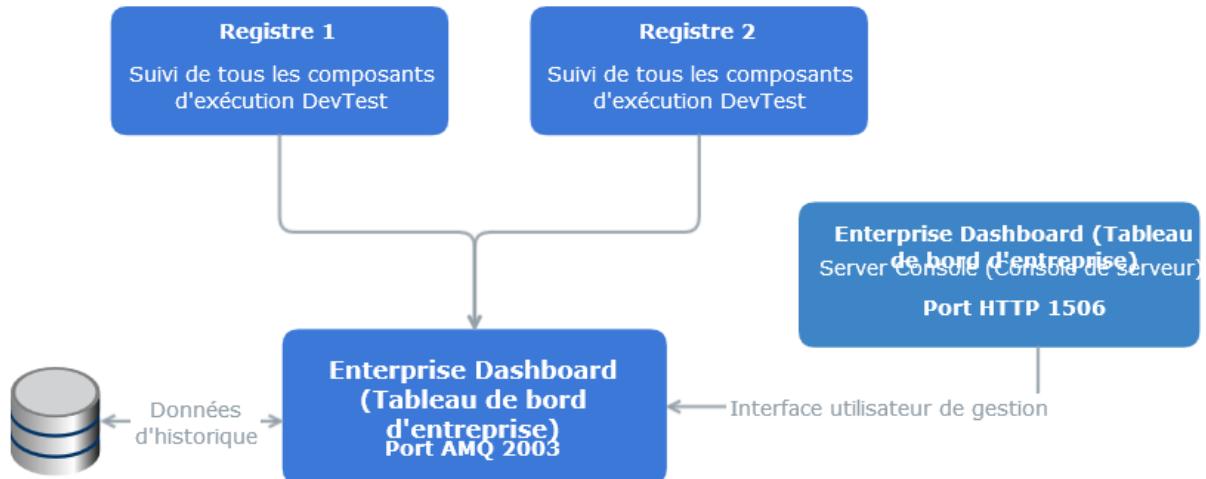
Le diagramme ci-après illustre l'architecture de CAI.



* Lorsque plusieurs VSE sont installés sur un ordinateur identique, ils auront des numéros de port différents.

Architecture du tableau de bord Enterprise Dashboard

Le diagramme ci-après illustre l'architecture du tableau de bord Enterprise Dashboard.

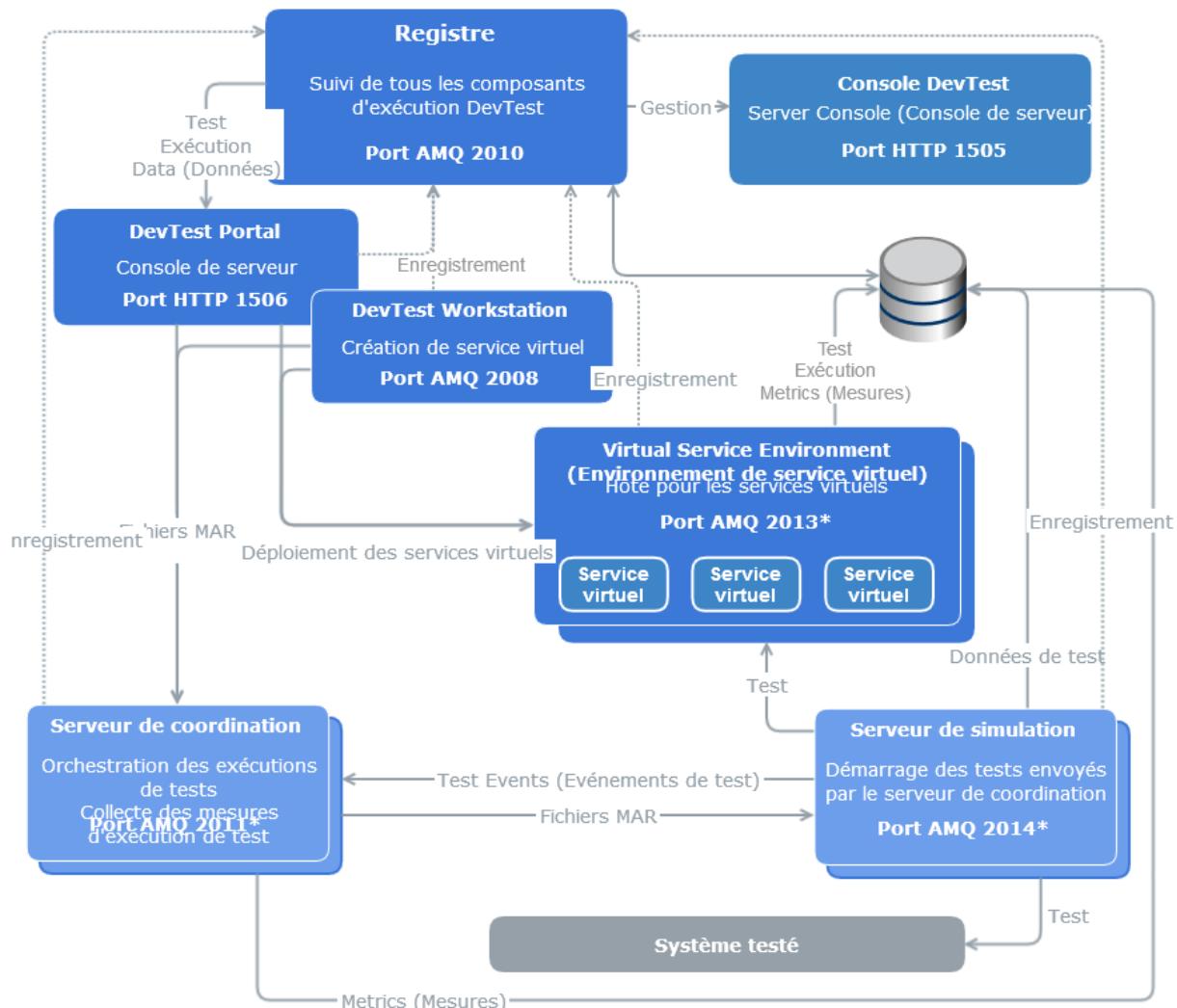


Composants de DevTest Server

Dans CA Application Test, les tests sont exécutés dans l'environnement de DevTest Server. DevTest Workstation se connecte à l'environnement DevTest Server pour déployer et surveiller des tests développés dans DevTest Workstation.

CA Application Test et CA Service Virtualization utilisent les composants du DevTest Server suivants :

- **DevTest Workstation** : un environnement de développement intégré (IDE) dans lequel des actifs de scénario de test et des modèles de service virtuel sont créés et modifiés. Vous pouvez exécuter des scénarios de test et des modèles localement dans la station de travail, ou les simuler pour une exécution distante. DevTest Workstation doit être installé sur des ordinateurs de bureau pour les utilisateurs qui créent des tests et des actifs de modèle virtuel. Plusieurs stations de travail peuvent être connectées au registre et partager l'environnement de serveur. Pour plus d'informations, consultez la section DevTest Workstation de la rubrique *Utilisation de CA Application Test*.
- **Registry (Registre)** : emplacement central pour l'enregistrement de tous les composants DevTest Server et DevTest Workstation. Pour plus d'informations, consultez la section Registre de la rubrique *Utilisation de CA Application Test*.
- **DevTest Console (Console DevTest)** : la console inclut des liens vers la console de serveur (y compris le VSE), le CVS Dashboard (Tableau de bord du service CVS), la génération de rapports et la console VSEasy.
- **Coordinator Server (Serveur de coordination)** : le coordinateur reçoit les informations de test sous forme de fichiers MAR et coordonne les tests exécutés sur un ou plusieurs serveurs de simulation. Pour plus d'informations, consultez la section Coordinator Server (Serveur de coordination) de la rubrique *Utilisation de CA Application Test*.
- **Simulator Server (Serveur de simulation)** : Le simulateur exécute les tests sous la surveillance du serveur de coordination. Pour plus d'informations, consultez la section Simulator Server (Serveur de simulation) de la rubrique *Utilisation de CA Application Test*.
- **VSE** : utilisé pour déployer et exécuter des modèles de service virtuel. Pour plus d'informations, reportez-vous à la rubrique *Utilisation de CA Service Virtualization*.



* Lorsque plusieurs coordinateurs, simulateurs et VSE sont installés sur un ordinateur identique, ils auront des numéros de port différents.

Le registre, le serveur de coordination, le serveur de simulation et le VSE sont des applications Java sans périphérique de contrôle, qui sont exécutés dans des machines virtuelles distinctes.

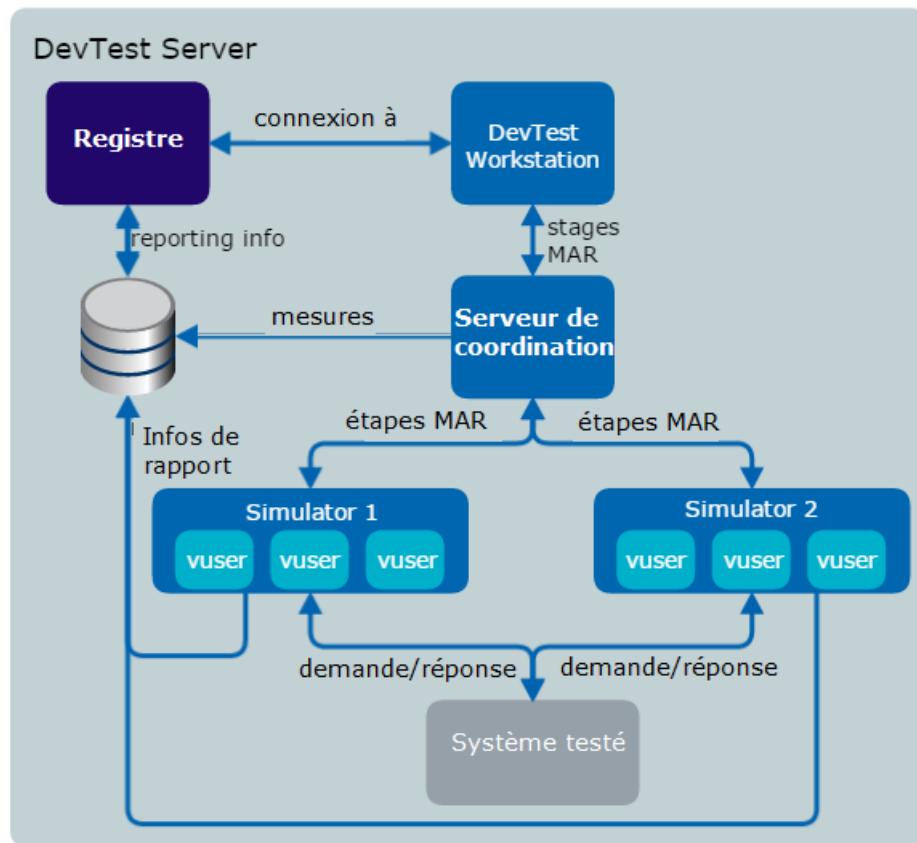
La configuration de DevTest Server pour CA Application Test et CA Service Virtualization doit au minimum inclure un exemplaire de chacun de ces composants. Autant d'instances de chaque type que nécessaire peuvent exister dans un environnement de test spécifique.

En général, une configuration de DevTest Server pour CA Application Test contient un registre, un serveur de coordination et plusieurs serveurs de simulation.

Le VSE est un service de niveau serveur. Le service peut coexister avec un registre disposant d'un coordinateur et d'un simulateur qui y sont connectés. Le simulateur et le coordinateur ne sont pas requis pour l'exécution du VSE.

Flux de données dans DevTest Server pour CA Application Test et CA Service Virtualization

Le graphique suivant illustre le flux de données dans le registre, DevTest Workstation, le serveur de coordination, les serveurs de simulation et la base de données.



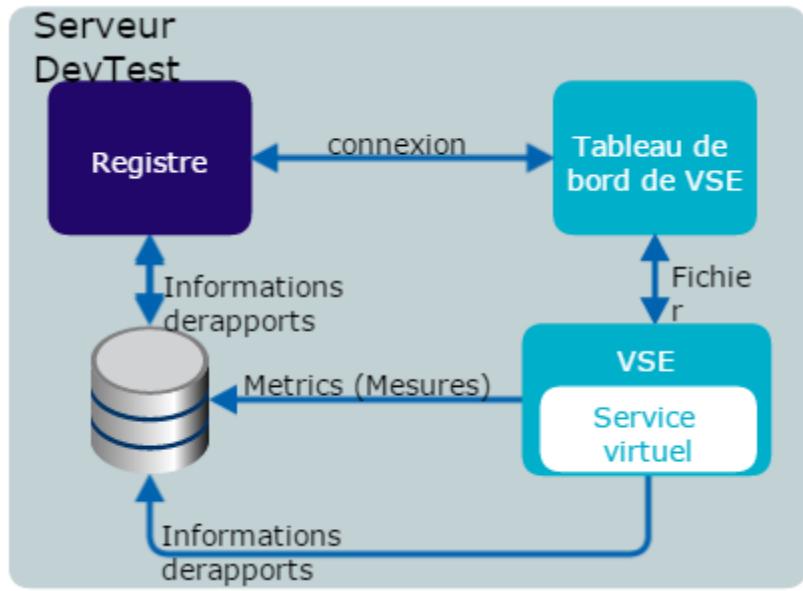
Le serveur de coordination envoie des scénarios de test à un ou à plusieurs serveurs de simulation sous forme d'archives de modèle (MAR).

Les serveurs de simulation interagissent avec le système testé. Les types de données échangées entre ces composants peuvent varier considérablement. Il peut s'agir de demandes HTTP simples avec des réponses HTML, appels de service Web, des appels de base de données, etc.

Les différents composants envoient des mesures et des informations de génération de rapports à la base de données.

Le registre héberge le portail de génération de rapports, de sorte à communiquer avec la base de données pour récupérer des données de génération de rapports.

Le graphique suivant illustre le flux de données dans le registre, VSE, le tableau de bord du VSE et la base de données.



Téléchargement des programmes d'installation de DevTest Solutions

Le Download Center (Centre de téléchargements) accessible à partir du site de support en ligne de CA permet de télécharger un ou plusieurs programmes d'installation spécifiques aux plates-formes et un fichier facultatif relatif à cette version de DevTest Solutions.

Téléchargez d'abord les programmes d'installation spécifiques aux plates-formes. A l'issue du téléchargement, répétez le processus pour télécharger le serveur de démonstration avec des exemples. Le serveur de démonstration vous permettra de pratiquer avec des exemples ou d'utiliser les didacticiels inclus dans la documentation de CA Application Test.

Procédez comme suit:

1. Accédez à www.support.ca.com.
2. Cliquez sur Login (Connexion) et connectez-vous avec votre nom d'utilisateur et votre mot de passe du site de support en ligne de CA.
3. Passez la souris sur Download Center (Centre de téléchargements) et sélectionnez Download Products (Télécharger des produits).
La page Download Center (Centre de téléchargements) s'ouvre, avec l'option All Products (Tous les produits) sélectionnée.
4. Saisissez l'un des noms de produits de DevTest suivants pour afficher la liste déroulante des options appropriées.
 - CA Service Virtualization for Performance – MULTI-PLATFORM
 - CA Service Virtualization Power User – MULTI-PLATFORM
 - CA Service Virtualization Runtime User – MULTI-PLATFORM
 - CA Service Virtualization Virtual User – MULTI-PLATFORM
 - CA Continuous Application Insight Power User – MULTI-PLATFORM
 - CA Continuous Application Insight Runtime User – MULTI-PLATFORM
 - CA Application Test – MULTI-PLATFORM
5. Sélectionnez le produit à télécharger. Chaque sélection permet de télécharger l'assistant d'installation des DevTest Solutions pour votre système d'exploitation. Ce programme d'installation vous permet d'installer le DevTest Server, qui inclut les trois produits ainsi que la station de travail DevTest Workstation.
6. Sélectionnez la version à télécharger.
7. Acceptez les valeurs par défaut pour tous les autres champs.
8. Cliquez sur Go (OK).

Une liste de composants de produit s'ouvre.

9. Sélectionnez la case à cocher dans la colonne Add to Cart (Ajouter au panier) pour chaque composant à télécharger.
 - (Obligatoire) Le programme d'installation pour chacun de vos systèmes d'exploitation
 - (Facultatif) Le serveur de démonstration DevTest
10. Cliquez sur My Download Cart (Mon panier de téléchargement) dans la partie supérieure de la fenêtre.
La page My Download Cart (Mon panier de téléchargement) s'ouvre.
11. Entrez votre adresse électronique dans la section Checkout (Validation) et cliquez sur Checkout.
La page Download Method (Méthode de téléchargement) s'affiche.
12. Cliquez sur Download (Télécharger) et enregistrez le fichier.

Chapitre 2: Présentation de l'installation de DevTest

Cette documentation couvre l'installation unique de l'ensemble DevTest Solutions. Les sections ne sont pas séparées par produit. CA Application Test, CA Service Virtualization et CA Continuous Application Insight sont installés comme ensemble de solutions.

Ce chapitre traite des sujets suivants :

[Options d'installation](#) (page 44)

[Procédure d'installation et de configuration de DevTest Solutions](#) (page 48)

[Fonctionnement de l'activation de la licence](#) (page 50)

Options d'installation

Les utilisateurs sans droits d'administrateur système peuvent installer le serveur (DevTest Server), la station de travail DevTest Workstation et le serveur de démonstration. Toutefois, seuls les administrateurs système peuvent installer les composants de serveur comme services Windows.

Options de composants à installer

L'assistant d'installation de DevTest Solutions présente trois boîtes de dialogue qui contrôlent les options installées sur l'hôte local. Vous pouvez sélectionner une seule option dans chaque boîte de dialogue. L'option par défaut apparaît en italique :

- **Sélection de composants**
 - *Server* (Serveur, avec station de travail) également appelé DevTest Server
 - *Workstation (only)* (Station de travail uniquement)
- **Enterprise Dashboard (Tableau de bord d'entreprise)**
 - *New Enterprise Dashboard* (*Nouveau tableau de bord d'entreprise*)
 - Existing Enterprise Dashboard Service (Service de tableau de bord d'entreprise existant)
Cette option requiert le fichier de licence devtestlic.xml pour être disponible.
- **Serveur de démonstration**
 - Installation
Cette option requiert le téléchargement et la présence du fichier d'installation du serveur de démonstration.
 - *Do not install* (*Ne pas installer*)

Le programme d'installation présente une boîte de dialogue dans laquelle vous devez sélectionner un type de configuration. Les deux types de configuration Shared (Partagé) et Local, la configuration partagée convient pour un système unique à plusieurs utilisateurs. La configuration locale convient à une installation accessible par les utilisateurs à partir d'autres systèmes ou dans laquelle les composants sont contrôlés et exécutés sur plusieurs systèmes (configuration distribuée).

Configurations standard

Autonome (Un seul ordinateur utilisé pour héberger les composants DevTest)

Si vous procédez à une nouvelle installation autonome DevTest Solutions sur un hôte, installez les trois composants :

- Server (Serveur, avec station de travail)
- New Enterprise Dashboard (Nouveau tableau de bord d'entreprise)
- Serveur de démonstration

Si plusieurs utilisateurs utiliseront la station de travail DevTest Workstation intégrée sur un seul ordinateur, sélectionnez l'option Shared pour le type d'installation. Pour plus de détails, reportez-vous à la section Type d'installation partagée.

Serveurs distribués (Plusieurs ordinateurs utilisés pour héberger les composants DevTest)

Si vous installez un système distribué de DevTest Solutions sur plusieurs hôtes, tenez compte de l'approche suivante :

1. Dans le cas de la *première installation*, à partir d'un serveur respectant la configuration système requise, sélectionnez les options suivantes :
 - Serveur
 - New Enterprise Dashboard (Nouveau tableau de bord d'entreprise), via lequel vous accédez au fichier de licence **devtestlic.xml**. Le programme d'installation valide le fichier de licence et son contenu concernant la signature, la version et la date d'expiration.
2. Dans le cas de *l'installation de plusieurs serveurs*, sélectionnez les options suivantes :
 - Serveur
 - Existing Enterprise Dashboard (Tableau de bord d'entreprise existant)

Remarque : Installez le serveur même si vous prévoyez d'utiliser un seul composant à partir de celui-ci. Par exemple, si vous collectez de nombreuses mesures ou demandez de nombreux rapports, vous pouvez exécuter un serveur de coordination sur un ordinateur sur lequel aucun autre service n'est exécuté.
3. Pour installer des composants sur des hôtes spécifiques pour des utilisateurs de CA Application Test et CA Service Virtualization, sélectionnez les options suivantes :
 - Station de travail

Lorsque les utilisateurs se connectent à la station de travail, ils se relient au registre installé avec l'un des serveurs.

 - Demo Server - Install (Installer le serveur de démonstration) ; pour de nouveaux utilisateurs qui peuvent tirer parti des didacticiels
4. Pour les utilisateurs de CA Continuous Application Insight, veillez à ce qu'un [navigateur pris en charge](#) (page 20) soit installé. Toutes les tâches sont effectuées à partir d'un portail Web.

Options d'installation de processus requis (Windows)

Jusqu'à sept processus doivent être démarrés pour utiliser DevTest Solutions. L'assistant d'installation vous permet de sélectionner le mode de réalisation de cette tâche :

- Create a Start menu folder (Créer un dossier dans le menu Démarrer)
- Install Services (Installer des services)

Remarque : Vous pouvez sélectionner l'une de ces options, les deux , ou aucune des deux. Si vous n'en sélectionnez aucune, vous pouvez démarrer les fichiers exécutables des processus à partir du dossier bin sous le répertoire d'installation du serveur DevTest (LISA_HOME\bin) ou à partir d'une invite de commande.

Procédure d'installation et de configuration de DevTest Solutions

Ce manuel d'installation inclut la configuration système requise, un arrière-plan conceptuel et des procédures requises uniquement dans certaines circonstances. Utilisez les directives suivantes pour la première installation DevTest Solutions :

1. [Téléchargez les programmes d'installation de DevTest Solutions](#) (page 41).
2. Si vous installez le tableau de bord Enterprise Dashboard, stockez votre fichier de licence **devtestlic.xml** sur le système cible.
Remarque : Enregistrez l'emplacement du fichier de licence pour une utilisation future par le service de support de produits CA.
3. Si vous utilisez le programme d'installation générique UNIX, [vous devez disposer de votre propre machine virtuelle Java](#) (page 13).
4. Installez e DevTest Server sur autant de systèmes que nécessaire. Pour la première installation du DevTest Server, sélectionnez New Enterprise Dashboard (Nouveau tableau de bord d'entreprise) et accédez au fichier de licence **devtestlic.xml**.
 - [Windows](#) (page 51)
 - [UNIX](#) (page 56)
 - [Mac](#) (page 59)

Etapes Post-Installation

1. Configurez des bases de données externes.
 - Configurez une base de données externe pour les registres. (DB2, MySQL, Oracle, ou SQL Server)
 - Configurez une base de données externe pour le tableau de bord Enterprise Dashboard. (MySQL, Oracle, ou SQL Server)
Remarque : Il n'existe aucun chemin de migration à partir de la base de données Derby d'Apache installée avec DevTest Solutions vers une base de données d'entreprise. Si vous utilisez Derby, vous devez ressaisir les données l'authentification d'utilisateur obligatoires (rôle) lorsque votre système devient généralement disponible, dans lequel une base de données d'entreprise est requise.
2. Activez le tableau de bord Enterprise Dashboard et tous les registres.
 - a. Si vous procédez à une mise à niveau, [configurez les registres existants](#) (page 66). La configuration des registres est automatisée pour les versions 8.0.0 et ultérieures.
 - b. [Activez les registres](#) (page 63).

Remarque : Ce processus active également le tableau de bord Enterprise Dashboard avec une clé de produit.

- c. [Vérifiez les activations de registre](#) (page 64). (Nouvelle installation ou mise à niveau)
3. Effectuez les tâches de [post-installation](#) (page 65).
4. (Facultatif) Installer d'autres DevTest Workstation avec le serveur de démonstration à utiliser avec chaque registre.
 - [Windows](#) (page 80)
 - [UNIX](#) (page 82)
 - [Mac](#) (page 85)
5. [Vérifiez l'installation de DevTest Solutions](#) (page 93).
6. Pour éviter une utilisation non autorisée, modifiez les mots de passe des utilisateurs standard.
7. [Installez les outils d'intégration](#) (page 105) dont vous avez besoin.
8. [Configurez l'environnement de test d'application mobile](#) (page 141).

A l'issue de l'installation, consultez la section Administration pour plus d'informations sur la configuration de sécurité au moyen d'une liste de contrôle d'accès et le respect de votre contrat de licence.

Fonctionnement de l'activation de la licence

Une licence de DevTest Solutions (**devtestlic.xml**) est émise pour chaque entreprise. Ce fichier déverrouille toutes les fonctionnalités de DevTest Solutions. Lors de la première installation du DevTest Server à l'aide de l'assistant d'installation DevTest Solutions, accédez au fichier de licence. Le processus d'installation inclut alors le tableau de bord Enterprise Dashboard et place ce fichier dans le répertoire d'installation du DevTest Server (LISA_HOME). Si vous installez plusieurs serveurs de registre DevTest, indiquez l'URL au serveur du tableau de bord Enterprise Dashboard.

Remarque : Pour changer l'emplacement de la base de données du tableau de bord Enterprise Dashboard, vous devez réexécuter le programme d'installation tout en conservant le schéma de base de données actuel. Avant de procéder à une nouvelle installation, sauvegardez les fichiers de propriétés personnalisés. Ceux-ci incluent les fichiers local.properties, site.properties et les plusieurs fichiers .vmoptions.

Important : En cas de modification de l'URL vers le tableau de bord Enterprise Dashboard, vous devez mettre à jour le fichier **local.properties** dans le répertoire LISA_HOME pour chaque serveur de registre qui envoie des rapports au tableau de bord Enterprise Dashboard, en définissant la propriété **lisa.enterprisedashboard.service.url** sur la nouvelle URL.

Lorsque vous démarrez le processus du tableau de bord Enterprise Dashboard et chaque processus de registre, le processus du tableau de bord Enterprise Dashboard lit les paramètres de registre et active les registres. Les registres activés s'affichent dans l'interface utilisateur du tableau de bord Enterprise Dashboard à des fins de vérification.

Important : En cas de modification du nom d'hôte ou du port d'un registre, vous devez le redémarrer de sorte que le tableau de bord Enterprise Dashboard puisse le réactiver.

Remarque : Reportez-vous à la section Réactivation d'un registre ou d'un tableau de bord Enterprise Dashboard dans la rubrique *Administration*.

Chapitre 3: Installation du DevTest Server et tâches de post-installation

Cette section décrit la procédure d'installation et de configuration de DevTest Server.

Ce chapitre traite des sujets suivants :

- [Installation de DevTest Server sous Windows](#) (page 51)
- [Installation de DevTest Server sous UNIX](#) (page 56)
- [Installation du DevTest Server sous Mac](#) (page 59)
- [Activation des registres](#) (page 63)
- [Vérification de l'activation de registre](#) (page 64)
- [Post-installation](#) (page 65)
- [Désinstallation de DevTest Server](#) (page 76)

Installation de DevTest Server sous Windows

Cette rubrique décrit la procédure d'installation des composants de serveur de DevTest Solutions dans un environnement Windows.

Pour prévisualiser les sélections d'installation, reportez-vous à la section [Composants pouvant être installés lors de l'exécution du programme d'installation](#) (page 44).

Avant de lancer la procédure d'installation, téléchargez les fichiers suivants à partir du [Download Center](#) (page 41) (Centre de téléchargement) :

- Le programme d'installation pour votre plate-forme
- (Facultatif) Le fichier .zip du serveur de démonstration

Remarque : En général, le serveur de démonstration est installé avec les composants de serveur uniquement dans un système autonome. Dans un système distribué, le serveur de démonstration est habituellement installé avec la station de travail DevTest Workstation pour des utilisateurs qui ne connaissent pas encore DevTest Solutions. Le serveur de démonstration est utilisé pour les didacticiels et de nombreux artefacts disponibles du projet Examples.

Procédez comme suit:

1. Exécutez le fichier du programme d'installation ; par exemple : devtest_win_x64.exe.

L'étape Welcome to the DevTest Solutions Setup Wizard (Bienvenue dans l'assistant d'installation de DevTest Solutions) s'ouvre.

2. Cliquez sur Next (Suivant).

Le contrat de licence d'utilisateur final CA s'affiche.

3. Lisez le contrat de licence, sélectionnez l'option I accept the terms of the License Agreement (J'accepte les termes de cette licence) à la fin de la page, puis cliquez sur Next (Suivant).

L'étape Select Destination Directory (Sélectionner le répertoire de destination) s'ouvre.

4. Saisissez le chemin et le nom du répertoire d'installation (LISA_HOME). Si vous voulez distinguer la version actuelle des versions antérieures, saisissez un nom qui inclut l'identificateur de version. Par exemple, entrez : C:\DevTestServer_8.0. Vous pouvez également accepter la valeur par défaut (C:\Program Files\CA\DevTestSolutions). Si vous accédez à un chemin et que vous saisissez le nom d'un nouveau dossier, l'assistant d'installation créera ce dossier.

5. Cliquez sur Next (Suivant).

La page Installation Type (Type d'installation) s'affiche.

6. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

Local

Tous les composants de DevTest Solutions sont installés dans un répertoire unique sur l'ordinateur local. Par défaut, toutes les données sont stockées dans ce répertoire et chaque utilisateur est associé à un répertoire temporaire personnel. Ce type d'installation est le plus commun et est utilisé dans la plupart des environnements.

Shared (Partagé)

Cette option est utilisée par les administrateurs pour installer tous les composants de DevTest Solutions à un emplacement partagé auquel plusieurs utilisateurs peuvent se connecter pour utiliser la station de travail DevTest Workstation. Toutes les données et les fichiers temporaires sont stockés dans des répertoires spécifiés par l'utilisateur. Chaque utilisateur dispose de ses données personnelles, mais tous partagent une installation de DevTest Solutions commune. Dans une installation partagée, les utilisateurs requièrent uniquement des droits d'accès en lecture au répertoire de programmes des DevTest Solutions. Ce type d'installation convient à une installation autonome.

L'étape Select Components (Sélectionner des composants) s'ouvre.

7. Vérifiez que la case à cocher Server (Serveur) est sélectionnée ; une station de travail intégrée est installée avec le serveur. Cliquez sur Next (Suivant).

L'étape Enterprise Dashboard (Tableau de bord d'entreprise) s'ouvre.

8. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

New Enterprise Dashboard (Nouveau tableau de bord d'entreprise) - spécifiez l'emplacement du fichier de licence.

S'il s'agit de l'installation d'un premier serveur, cliquez sur Browse (Parcourir), accédez à l'emplacement du fichier de licence, sélectionnez **devtestlic.xml** et cliquez sur Open (Ouvrir). Le programme d'installation copie le fichier **devtestlic.xml** dans le répertoire d'installation spécifié (LISA_HOME) sur l'hôte local. Le nouveau processus Enterprise Dashboard (Tableau de bord d'entreprise) est installé dans le répertoire LISA_HOME\bin. Cette option spécifie que tous les registres doivent se connecter au nouveau tableau de bord Enterprise Dashboard.

Existing Enterprise Dashboard Service (Service de tableau de bord d'entreprise existant)

S'il ne s'agit pas de la première installation de serveur dans ce réseau, saisissez l'URL du tableau de bord Enterprise Dashboard existant. Cette option spécifie que le registre installé avec ce serveur doit se connecter au tableau de bord Enterprise Dashboard existant. Remplacez *localhost* par le nom d'hôte approprié :

`tcp://localhost:2003/EnterpriseDashboard`

L'étape Demo Server (Serveur de démonstration) s'ouvre.

9. Si vous voulez que le programme d'installation décomprime le fichier DevTestDemoServer.zip dans le répertoire LISA_HOME, sélectionnez l'option Install demo server (Installer le serveur de démonstration). Puis, acceptez le chemin par défaut (Répertoire Downloads (Téléchargement)) ou spécifiez un autre chemin complet.

Remarque : Dans un système distribué dans lequel les stations de travail sont installées sur des ordinateurs d'utilisateur, le serveur de démonstration est généralement installé avec les stations de travail des nouveaux utilisateurs, non avec le serveur.

10. Cliquez sur Next (Suivant).
11. Si vous avez sélectionné le type d'installation locale, ignorez l'étape suivante, qui s'applique à un type d'installation partagée.
12. Spécifiez le répertoire de données et cliquez sur Next (Suivant) après chaque étape.

L'étape Select Start Menu Folder (Sélectionner le dossier dans le menu Démarrer) s'ouvre.

13. (Facultatif) Spécifiez si vous pouvez démarrer les processus de DevTest Solutions à partir du menu Démarrer. Vous pouvez lancer manuellement le fichier exécutable pour chaque processus à partir du répertoire bin sous le répertoire d'installation. Contrairement aux services associés, le lancement de fichiers exécutables permet de surveiller les messages affichés dans l'interface de ligne de commande.
 - Pour créer un dossier dans le menu Démarrer avec des raccourcis pour tous les utilisateurs, acceptez toutes les valeurs par défaut. Vous pouvez également saisir un nouveau nom de dossier.
 - Si vous ne souhaitez pas créer un dossier dans le menu Démarrer, désélectionnez la case à cocher Create a Start Menu folder (Créer un dossier dans le menu Démarrer).
 - Pour créer un dossier dans le menu Démarrer et limiter l'affichage des raccourcis à votre menu Démarrer :
 - Acceptez la sélection de Create a Start Menu folder (Créer un dossier dans le menu Démarrer).
 - Acceptez le nom par défaut ou saisissez en un autre.
 - Désélectionnez la case à cocher Create shortcuts for all users (Créer des raccourcis pour tous les utilisateurs).
14. Cliquez sur Next (Suivant).
L'étape Desktop Icons (Icônes d'ordinateur de bureau) s'ouvre.
15. (Facultatif) Si vous ne voulez pas créer d'icône d'ordinateur de bureau pour les interfaces utilisateur du tableau de bord Enterprise Dashboard de DevTest, du portail DevTest et de la DevTest Workstation (Station de travail), désélectionnez la case à cocher Create a desktop icon (Créer une d'icône d'ordinateur de bureau). Cliquez sur Next (Suivant).
 - Si vous avez sélectionné un type d'installation local et vous êtes administrateur, l'étape Windows Services s'ouvre.
 - Si vous avez sélectionné un type d'installation Local, mais que vous n'êtes pas administrateur, ignorez l'étape suivante.
 - Si vous avez sélectionné le type d'installation Shared (Partagé), passez à l'étape Information.

16. (Facultatif) Sélectionnez Install Services (Installer des services) pour créer les services Windows suivants.

- Service d'intermédiaire DevTest (pour CAI)
- Service de coordinateur DevTest
- Service de tableau de bord d'entreprise DevTest
- Service de portail DevTest
- Service de registre DevTest
- Service de simulateur DevTest
- Service du VSE DevTest

Cette sélection permet d'ajouter les services à Administrative Tools (Outils d'administration), Component Services (Services de composants), Services.

Contrairement aux fichiers exécutables associés, le lancement des services permet de limiter le nombre d'icônes affichées dans la barre d'état système. Si vous voulez définir le type de démarrage sur Automatic de sorte que les services démarrent lors du redémarrage de l'ordinateur hôte, sélectionnez la case à cocher Start on bootup (Lancer au démarrage). Vous pouvez configurer le démarrage automatique dans les services, si vous ne sélectionnez pas cette case à cocher.

17. Cliquez sur Next (Suivant).

L'étape Select File Associations (Sélectionner des associations de fichiers) s'ouvre. Toutes les associations sont sélectionnées par défaut.

18. Laissez toutes les associations sélectionnées ou supprimez les extensions de fichier que vous ne voulez pas associer à DevTest Solutions. Les extensions de fichier que vous pouvez associer à DevTest Solutions sont les suivantes :

- *.tst -- sélectionnez cette extension pour créer des scénarios de test avec CA Application Test.
- *.vsm et *.vsi -- sélectionnez ces extensions pour créer des services virtuels avec CA Service Virtualization.
- *.ste -- sélectionnez cette extension pour exécuter une suite avec l'exécuteur de tests dans CA Application Test.
- *.stg -- sélectionnez cette extension pour exécuter un scénario de test comme document de simulation dans CA Application Test.

19. Cliquez sur Install (Installer) pour lancer l'installation.

L'étape **Installing (Installation)** s'ouvre. A l'issue de l'installation, l'étape **Information** s'ouvre.

20. Lisez les informations et cliquez sur **Next (Suivant)**.

L'étape de l'assistant **Completing the DevTest Solutions Setup (Finalisation de l'installation de DevTest Solutions)** s'ouvre.

21. Cliquez sur **Finish (Terminer)**.

Poursuivez les étapes décrites dans la section [Procédure d'installation de DevTest Solutions](#) (page 48).

Installation de DevTest Server sous UNIX

Cette rubrique décrit la procédure d'installation de DevTest Server dans un environnement UNIX ou Linux.

Avant de lancer la procédure d'installation, téléchargez les fichiers suivants à partir du [Download Center](#) (page 41) (Centre de téléchargement) :

- Le programme d'installation pour votre plate-forme
- (Facultatif) Le fichier .zip du serveur de démonstration

Remarque : En général, le serveur de démonstration est installé avec les composants de serveur uniquement dans un système autonome. Dans un système distribué, le serveur de démonstration est habituellement installé avec la station de travail DevTest Workstation pour des utilisateurs qui ne connaissent pas encore DevTest Solutions.

La procédure suivante correspond à la version graphique du programme d'installation. Pour utiliser la version de la ligne de commande du programme d'installation, ajoutez l'option **-c**. Par exemple :

```
./devtest_platform_x64.sh -c
```

Remarque : Si vous utilisez le programme d'installation UNIX générique, vérifiez qu'une [machine virtuelle Java](#) (page 13) est installée sur le même ordinateur. La version de la machine virtuelle Java doit être 1.7. Vous pouvez spécifier une machine virtuelle Java en définissant la variable d'environnement **JAVA_HOME**. Si le programme d'installation ne parvient pas à détecter une machine virtuelle Java, il affichera un message avant de se fermer.

Procédez comme suit:

1. Dans une fenêtre de terminal, accédez au répertoire dans lequel le fichier de programme d'installation est placé.

2. Vérifiez que le fichier de programme d'installation dispose de l'autorisation d'exécution.

```
chmod 777 devtest_platform_x64.sh
```

Cette commande octroie les autorisations rwxrwxrwx sur le fichier.

3. Exécutez le fichier d'installation. Par exemple :

```
./devtest_platform_x64.sh
```

L'assistant Welcome to the DevTest Solutions Setup Wizard (Bienvenue dans l'assistant d'installation de DevTest Solutions) s'ouvre.

4. Cliquez sur Next (Suivant).

Le contrat de licence d'utilisateur final CA s'affiche.

5. Lisez le contrat de licence, sélectionnez la case à cocher I accept the terms of the License Agreement (J'accepte les termes de cette licence), puis cliquez sur Next (Suivant).

L'étape Select Destination Directory (Sélectionner le répertoire de destination) s'ouvre.

6. Spécifiez le répertoire dans lequel vous voulez installer DevTest Solutions ; par exemple : /opt/CA/DevTest. N'utilisez pas un chemin dont le nom de répertoire contient des espaces.

7. Cliquez sur Next (Suivant).

La page Installation Type (Type d'installation) s'affiche.

8. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

Local

Tous les composants de DevTest Solutions sont installés dans un répertoire unique sur l'ordinateur local. Par défaut, toutes les données sont stockées dans ce répertoire et chaque utilisateur est associé à un répertoire temporaire personnel. Ce type d'installation est le plus commun et est utilisé dans la plupart des environnements.

Shared (Partagé)

Les administrateurs utilisent ce type d'installation pour installer tous les composants des DevTest Solutions dans un emplacement partagé auquel les utilisateurs peuvent accéder à partir de plusieurs ordinateurs. Toutes les données et les fichiers temporaires sont stockés dans des répertoires spécifiés par l'utilisateur. Chaque utilisateur dispose de ses données personnelles, mais tous partagent une installation de DevTest Solutions commune. Dans une installation partagée, les utilisateurs requièrent uniquement des droits d'accès en lecture au répertoire de programmes des DevTest Solutions.

L'étape Select Components (Sélectionner des composants) s'ouvre.

9. Vérifiez que la case à cocher Server (Serveur) est sélectionnée, puis cliquez sur Next (Suivant).

L'étape Enterprise Dashboard (Tableau de bord d'entreprise) s'ouvre.

10. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

New Enterprise Dashboard (Nouveau tableau de bord d'entreprise) - spécifiez l'emplacement du fichier de licence.

S'il s'agit de l'installation d'un premier DevTest Server, cliquez sur Browse (Parcourir), accédez à l'emplacement du fichier de licence, sélectionnez **devtestlic.xml** et cliquez sur Open (Ouvrir). Lorsque vous cliquez sur Next (Suivant), le programme d'installation copie le fichier devtestlic.xml dans le répertoire d'installation spécifié (LISA_HOME) sur l'hôte local. Le processus Enterprise Dashboard (Tableau de bord d'entreprise) est installé dans le répertoire LISA_HOME/bin.

Existing Enterprise Dashboard Service (Service de tableau de bord d'entreprise existant)

Si le registre de ce DevTest Server se connecte au tableau de bord Enterprise Dashboard installé avec le premier DevTest Server installé, saisissez l'URL du tableau de bord Enterprise Dashboard existant.

L'étape Specify Demo Server (Spécifier un serveur de démonstration) s'ouvre.

11. Si vous voulez que le programme d'installation décomprime le serveur de démonstration dans le même répertoire dans lequel vous installez le DevTest Server, sélectionnez la case à cocher Install demo server (Installer le serveur de démonstration) et recherchez le fichier DevTestDemoServer.zip.

12. Cliquez sur Next (Suivant).

Si vous sélectionnez le type d'installation partagé, vous êtes invité à spécifier le répertoire de données et le répertoire de fichiers temporaire.

13. Spécifiez les répertoires et cliquez sur Next (Suivant) après chaque étape.

L'étape Select Directory for Symlinks (Sélectionner le répertoire pour Symlinks) s'ouvre.

14. Cliquez sur Browse (Parcourir) et accédez au répertoire dans lequel DevTest crée des liens symboliques vers les fichiers exécutables. La valeur par défaut est /usr/local/bin. Vous devez disposer des autorisations requises pour écrire dans le répertoire. Si vous ne voulez pas créer de lien symbolique, désélectionnez la case à cocher Create symlinks (Créer des liens symboliques).

15. Cliquez sur Next (Suivant).

L'étape Desktop Icons (Icônes d'ordinateur de bureau) s'ouvre.

16. (Facultatif) Si vous ne voulez pas créer d'icône d'ordinateur de bureau pour le tableau de bord Enterprise Dashboard de DevTest, le portail DevTest et la DevTest Workstation (Station de travail), désélectionnez la case à cocher.
17. Cliquez sur Install (Installer) pour lancer l'installation.
A l'issue de l'installation, l'étape Information s'ouvre.
18. Lisez les informations et cliquez sur Next (Suivant).
L'étape de l'assistant Completing the DevTest Solutions Setup (Finalisation de l'installation de DevTest Solutions) s'ouvre.
19. Cliquez sur Finish (Terminer).

Poursuivez les étapes décrites dans la section [Procédure d'installation de DevTest Solutions](#) (page 48).

Installation du DevTest Server sous Mac

Cette rubrique décrit la procédure d'installation des composants de serveur de DevTest Solutions dans un environnement Mac.

Pour prévisualiser les sélections d'installation, reportez-vous à la section [Composants pouvant être installés lors de l'exécution du programme d'installation](#) (page 44).

Avant de lancer la procédure d'installation, téléchargez les fichiers suivants à partir du [Download Center](#) (page 41) (Centre de téléchargement) :

- Le programme d'installation pour votre plate-forme
- (Facultatif) Le fichier .zip du serveur de démonstration

Remarque : En général, le serveur de démonstration est installé avec les composants de serveur uniquement dans un système autonome. Dans un système distribué, le serveur de démonstration est habituellement installé avec la station de travail DevTest Workstation pour des utilisateurs qui ne connaissent pas encore DevTest Solutions. Le serveur de démonstration est utilisé avec des didacticiels.

Procédez comme suit:

1. Exécutez le fichier du programme d'installation ; par exemple : devtest_osx_x64.dmg.
L'étape Welcome to the DevTest Solutions Setup Wizard (Bienvenue dans l'assistant d'installation de DevTest Solutions) s'ouvre.
2. Cliquez sur Next (Suivant).
Le contrat de licence d'utilisateur final CA s'affiche.

3. Lisez le contrat de licence, sélectionnez l'option I accept the terms of the License Agreement (J'accepte les termes de cette licence) à la fin de la page, puis cliquez sur Next (Suivant).

L'étape Select Destination Directory (Sélectionner le répertoire de destination) s'ouvre.

4. Spécifiez le dossier dans lequel vous voulez installer un ou plusieurs composants de DevTest Solutions. Si vous spécifiez un dossier qui n'existe pas, l'assistant d'installation le créera.

5. Cliquez sur Next (Suivant).

La page Installation Type (Type d'installation) s'affiche.

6. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

Local

Tous les composants de DevTest Solutions sont installés dans un répertoire unique sur l'ordinateur local. Par défaut, toutes les données sont stockées dans ce répertoire et chaque utilisateur est associé à un répertoire temporaire personnel. Ce type d'installation est le plus commun et est utilisé dans la plupart des environnements.

Shared (Partagé)

Cette option est utilisée par les administrateurs pour installer tous les composants de DevTest Solutions à un emplacement partagé auquel plusieurs utilisateurs peuvent se connecter pour utiliser la station de travail DevTest Workstation. Toutes les données et les fichiers temporaires sont stockés dans des répertoires spécifiés par l'utilisateur. Chaque utilisateur dispose de ses données personnelles, mais tous partagent une installation de DevTest Solutions commune. Dans une installation partagée, les utilisateurs requièrent uniquement des droits d'accès en lecture au répertoire de programmes des DevTest Solutions. Ce type d'installation convient à une installation autonome.

L'étape Select Components (Sélectionner des composants) s'ouvre.

7. Vérifiez que la case à cocher Server (Serveur) est sélectionnée ; une station de travail intégrée est installée avec le serveur. Cliquez sur Next (Suivant).

L'étape Enterprise Dashboard (Tableau de bord d'entreprise) s'ouvre.

8. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

New Enterprise Dashboard (Nouveau tableau de bord d'entreprise) - spécifiez l'emplacement du fichier de licence.

S'il s'agit de l'installation d'un premier serveur, cliquez sur Browse (Parcourir), accédez à l'emplacement du fichier de licence, sélectionnez **devtestlic.xml** et cliquez sur Open (Ouvrir). Le programme d'installation copie le fichier **devtestlic.xml** dans le répertoire d'installation spécifié (LISA_HOME) sur l'hôte local. Le nouveau processus Enterprise Dashboard (Tableau de bord d'entreprise) est installé dans le répertoire LISA_HOME/bin. Cette option spécifie que le registre installé avec ce serveur doit se connecter au nouveau tableau de bord Enterprise Dashboard.

Existing Enterprise Dashboard Service (Service de tableau de bord d'entreprise existant)

S'il ne s'agit pas de la première installation de serveur dans ce réseau, saisissez l'URL du tableau de bord Enterprise Dashboard existant. Cette option spécifie que le registre installé avec ce serveur doit se connecter au tableau de bord Enterprise Dashboard existant. Remplacez *localhost* par le nom d'hôte approprié :

`tcp://localhost:2003/EnterpriseDashboard`

L'étape Demo Server (Serveur de démonstration) s'ouvre.

9. Si vous voulez que le programme d'installation décomprime le serveur de démonstration dans le répertoire LISA_HOME, sélectionnez l'option Install demo server (Installer le serveur de démonstration). Puis, acceptez le chemin par défaut (Répertoire Downloads (Téléchargement)) ou spécifiez un autre chemin complet.

Remarque : Dans un système distribué dans lequel les stations de travail sont installées sur des ordinateurs d'utilisateur, le serveur de démonstration est généralement installé avec les stations de travail des nouveaux utilisateurs, non avec le serveur.

10. Cliquez sur Next (Suivant).
11. Si vous avez sélectionné le type d'installation locale, ignorez l'étape suivante, qui s'applique à un type d'installation partagée.
12. Spécifiez le répertoire de données et cliquez sur Next (Suivant) après chaque étape.
13. Cliquez sur Next (Suivant).

L'étape Desktop Icons (Icônes d'ordinateur de bureau) s'ouvre.

14. (Facultatif) Si vous ne voulez pas créer d'icône d'ordinateur de bureau pour le tableau de bord Enterprise Dashboard de DevTest, le portail DevTest et la DevTest Workstation (Station de travail), désélectionnez la case à cocher. Cliquez sur Next (Suivant).
15. Cliquez sur Next (Suivant).

L'étape Select File Associations (Sélectionner des associations de fichiers) s'ouvre. Toutes les associations sont sélectionnées par défaut.

16. Laissez toutes les associations sélectionnées ou supprimez les extensions de fichier que vous ne voulez pas associer à DevTest Solutions. Les extensions de fichier que vous pouvez associer à DevTest Solutions sont les suivantes :

- *.tst -- sélectionnez cette extension pour créer des scénarios de test avec CA Application Test.
- *.vsm et *.vsi -- sélectionnez ces extensions pour créer des services virtuels avec CA Service Virtualization.
- *.ste -- sélectionnez cette extension pour exécuter une suite avec l'exécuteur de tests dans CA Application Test.
- *.stg -- sélectionnez cette extension pour exécuter un scénario de test comme document de simulation dans CA Application Test.

17. Cliquez sur Install (Installer) pour lancer l'installation.

L'étape Installing (Installation) s'ouvre. A l'issue de l'installation, l'étape Information s'ouvre.

18. Lisez les informations et cliquez sur Next (Suivant).

L'étape de l'assistant Completing the DevTest Solutions Setup (Finalisation de l'installation de DevTest Solutions) s'ouvre.

19. Cliquez sur Finish (Terminer).

Poursuivez les étapes décrites dans la section [Procédure d'installation de DevTest Solutions](#) (page 48).

Activation des registres

A l'issue de l'installation de toutes les instances de DevTest Server, démarrez le processus du tableau de bord Enterprise Dashboard sur le premier DevTest Server installé. Puis, démarrez le processus du registre sur ce DevTest Server et tous les autres DevTest Server. L'activation des registres ne se produit pas immédiatement. Par exemple, si vous redémarrez les registres entre 10 h 15 et 10 h 30, l'activation se produira à 11 h 00.

Vérifiez qu'un registre à partir de chaque DevTest Server nouveau et existant apparaît dans le tableau de bord Enterprise Dashboard.

Procédez comme suit:

1. Démarrez le serveur du tableau de bord Enterprise Dashboard.
 - a. Connectez-vous à l'hôte sur lequel le tableau de bord Enterprise Dashboard est installé.
 - b. Démarrez le serveur du tableau de bord Enterprise Dashboard.

Les utilisateurs Windows peuvent sélectionner Enterprise Dashboard Server (Serveur du tableau de bord d'entreprise) à partir de l'option Enterprise Dashboard dans le menu Démarrer. De même, accédez au répertoire LISA_HOME\bin et lancez EnterpriseDashboard.exe.

2. Si vous disposez déjà de registres, [configurez-les](#) (page 66).
 3. Démarrez chaque registre.
 - a. Connectez-vous à l'hôte sur lequel un DevTest Server est installé.
 - b. Démarrez le registre.
- Les utilisateurs Windows démarrent le registre à partir du menu Démarrer (sous DevTest Solutions) ou à partir du Registry.exe dans le répertoire bin sous le répertoire d'installation du DevTest Server.
- c. Répétez ces étapes pour chaque registre.

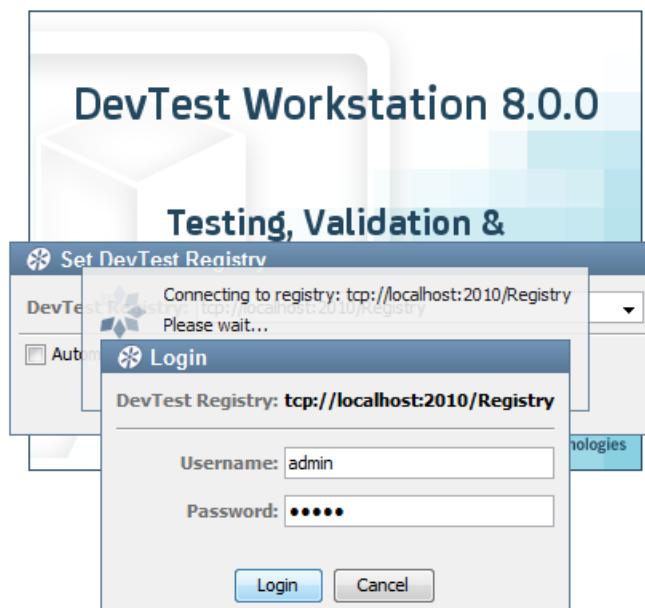
Remarque : Une fois que cette propriété est définie, le registre n'envoie pas de données au tableau de bord Enterprise Dashboard jusqu'à l'heure suivante. Par exemple, si vous définissez cette propriété à 13 h 35, les données de registre ne s'affichent pas avant 14 h 00.

Vérification de l'activation de registre

Après le démarrage de chaque registre, patientez jusqu'à la fin de l'heure suivante avant de confirmer leur activation.

Procédez comme suit:

1. Ouvrez l'interface utilisateur tableau de bord Enterprise Dashboard de l'une des manières suivantes :
 - Accédez au tableau de bord Enterprise Dashboard. En cas d'installation distante, spécifiez l'adresse IP ou le nom d'hôte, ou spécifiez le **localhost** s'il s'agit d'une installation locale.
`http://hostname:1506`
 - A partir de l'ordinateur sur lequel le tableau de bord Enterprise Dashboard est installé, les utilisateurs Windows sélectionnent l'option dans le menu Démarrer, Enterprise Dashboard (Tableau de bord d'entreprise), Enterprise Dashboard UI (Interface utilisateur du tableau de bord d'entreprise).
2. Connectez-vous.



Pour vous connecter, saisissez *admin* dans le champ Username (Nom d'utilisateur) et *admin* dans le champ Password (Mot de passe), puis cliquez sur Login (Connexion). Le tableau de bord Enterprise Dashboard s'ouvre.

3. Examinez les configurations de registre dans le tableau de bord Enterprise Dashboard.

Une liste de vos registres s'affiche au format similaire à l'exemple suivant :

Registry									
Display Name	Name	Status	Version	Coordinators	Simulators	VSEs	Workstations	Agents	Labs
Registry@FQDN:2010	tcp://hostname:2010/Registry	Running	8.0.0 (8.0.0.282)	0	0	1	1	0	1

4. Vérifiez qu'un registre à partir de chaque DevTest Server nouveau et existant apparaît dans le tableau de bord Enterprise Dashboard.

Remarque : Si les registres existants ne s'affichent pas, [configurez-les](#) (page 66).

Post-installation

Cette section comprend les rubriques suivantes :

- [Configuration des registres existants](#) (page 66)
- [Utilisation d'un serveur proxy HTTP/S - DevTest Server](#) (page 69)
- [Exécution des composants sur différents systèmes](#) (page 71)
- [Calcul des instances de simulateur](#) (page 72)
- [Dimensionnement de serveur de charge et de performances](#) (page 73)
- [Utilisation de DevTest Workstation dans un environnement Java](#) (page 74)
- [Modification de l'emplacement d'accueil de projet par défaut](#) (page 75)
- [Structure des répertoires de projet](#) (page 75)

Configuration des registres existants

Chaque DevTest Server que vous installez possède un registre. Les licences de produit requièrent que les registres soient configurés. Le processus de configuration varie en fonction de la version :

- **Version 8.0 et supérieure** : le processus d'installation configure automatiquement les nouveaux registres.
- **Versions 7.5.x** : suivez la procédure de configuration des registres (DevTest 7.5.x).
- **Versions antérieure à 7.5** : suivez la procédure d'ajout d'un registre à partir d'une version antérieure DevTest 7.5.

Pour configurer les registres (DevTest 7.5.x) :

1. Connectez-vous à un ordinateur sur lequel un DevTest Server est installé et accédez au répertoire d'installation.
2. Ouvrez le fichier local.properties et recherchez la section 1 - Enterprise Dashboard (Tableau de bord d'entreprise). Si le fichier local.properties n'existe pas, copiez _local.properties et remplacez le nom de la copie par local.properties.
3. Supprimez les commentaires de la ligne suivante et remplacez *somehost* par le nom de l'hôte sur lequel le tableau de bord Enterprise Dashboard est installé.
`lisa.enterprisedashboard.service.url=tcp://somehost:2003/EnterpriseDashboard`
4. Enregistrez le fichier local.properties, puis fermez-le.
5. (Facultatif) Si vous prévoyez d'utiliser une DevTest Workstation (Station de travail) à partir d'un ordinateur distant, vous pouvez spécifier une propriété qui vous permet de vous connecter automatiquement à un registre spécifique.
 - a. Connectez-vous à l'ordinateur sur lequel la Workstation (Station de travail) est installée.
 - b. Accédez à LISA_HOME.
 - c. Ouvrez le fichier local.properties et recherchez la section 2 - Autoconnection (Connexion automatique).
 - d. Supprimez les commentaires de la ligne suivante et remplacez *somehost* par le nom d'hôte de l'ordinateur auquel vous êtes connecté.
`lisaAutoConnect=tcp://somehost:2010/Registry`
 - e. Enregistrez le fichier local.properties, puis fermez-le.
6. Connectez-vous à l'ordinateur sur lequel le registre cible se trouve.
7. Démarrez ou redémarrez le registre.
 - a. Accédez au répertoire LISA_HOME\bin.
 - b. Exécutez registry.exe.
8. Répétez les étapes 1 à 7 pour chaque registre.

- Pour vérifier les configurations de registre, accédez au tableau de bord Enterprise Dashboard.

`http://hostname:1506`

Important : En cas de modification du nom d'hôte du système sur lequel vous installez DevTest Solutions, répétez cette procédure.

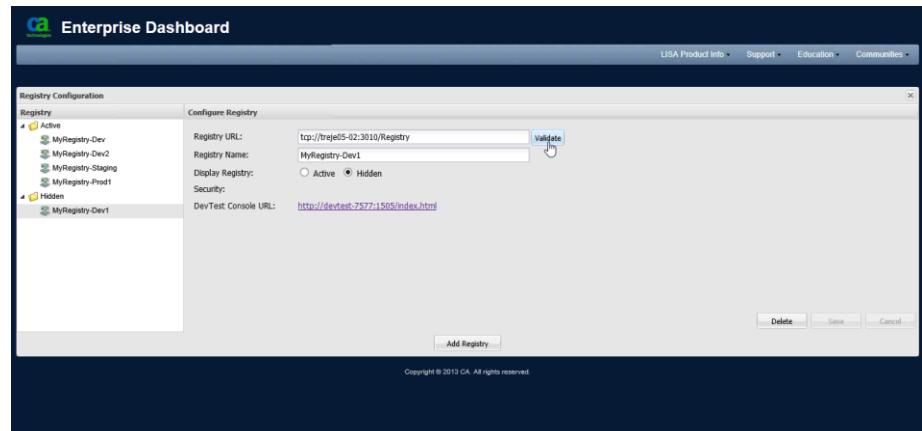
La fenêtre Registry Configuration (Configuration de registre) du tableau de bord Enterprise Dashboard permet d'ajouter et de configurer d'anciens registres, ainsi que de supprimer ceux devenus inutiles et qui apparaissent dans le tableau de bord.

Pour ajouter un registre à partir d'une version antérieure à DevTest 7.5 :

- Ouvrez le tableau de bord Enterprise Dashboard.
- Cliquez sur Options dans le coin supérieur droit de la fenêtre principale du tableau de bord Enterprise Dashboard.
- Sélectionnez Configure (Configurer).

La fenêtre Registry Configuration (Configuration de registre) s'ouvre et contient les champs de configuration de registre vides.

Remarque : Pour retourner à la fenêtre principale, cliquez sur Go Back to registry view (Revenir à la vue de registre) ou sur Overview (Présentation) dans les chemins de navigation du coin supérieur gauche de la page.



4. Cliquez sur Add Registry (Ajouter un registre) dans la fenêtre Registry Configuration (Configuration de registre). Cliquez sur Yes pour confirmer.

5. Renseignez les champs suivants :

Registry URL (URL de registre)

URL du registre. Ce champ est obligatoire.

Exemple :

`tcp://hostname:2010/Registry`

Pour valider l'URL, cliquez sur Validate (Valider).

Registry Name (Nom du registre)

Nom commun ou surnom de ce registre

Exemple :

`MyRegistry@hostname:2010`

Display Registry (Afficher le registre)

Sélectionnez Active pour afficher ce registre dans le panneau principal du tableau de bord, ou Hidden (Masqué) pour masquer ce registre. Ce paramètre option contrôle également le dossier sur le côté gauche qui contient ce registre.

La fenêtre Registry Configuration (Configuration de registre) inclut également les champs suivants :

Security (Sécurité)

Indique si la sécurité de liste de contrôle d'accès est activée ou désactivée.

DevTest Console URL (URL de la console DevTest)

URL de la console DevTest pour ce registre Exemple :

`http://hostname:1505/index.html`

6. Cliquez sur Save (Enregistrer) pour ajouter le registre au tableau de bord.

Pour valider un registre :

1. Sélectionnez Configure (Configurer) dans le menu Options.
2. Cliquez sur le registre que vous voulez valider dans la liste des registres de la colonne Registry (Registre).
3. Cliquez sur Validate (Valider).

Le serveur de tableau de bord interroge le registre sélectionné et affiche le statut ; par exemple : Is running (En cours d'exécution).

4. Cliquez sur OK.

Utilisation d'un serveur proxy HTTP/S - DevTest Server

Si vous utilisez un serveur proxy HTTP simple ou un serveur proxy HTTP sécurisé via SSL, définissez ce serveur proxy et tous les hôtes à exclure dans le fichier **local.properties** du répertoire LISA_HOME.

Procédez comme suit:

1. Connectez-vous à l'hôte sur lequel DevTest Server est installé.
2. Accédez à LISA_HOME.
3. Si le fichier local.properties n'existe pas, copiez le fichier **_local.properties** et enregistrez la copie comme local.properties (sans le trait de soulignement).
4. Ouvrez le fichier local.properties pour le modifier et recherchez l'en-tête de section pour le serveur proxy HTTP ou HTTPS, selon que votre serveur proxy utilise une connexion HTTP simple ou sécurisée via SSL.
5. Identifiez votre serveur proxy par son nom de domaine complet ou l'adresse IP et le port :
 - Dans le cas d'un serveur HTTP, utilisez les propriétés lisa.http.webProxy.host et lisa.http.webProxy.port.
 - Dans le cas d'un serveur HTTPS, utilisez les propriétés lisa.http.webProxy.ssl.host et lisa.http.webProxy.ssl.port.
6. Identifiez les hôtes à exclure de votre serveur proxy :
 - Dans le cas d'un serveur HTTP, utilisez la propriété lisa.http.webProxy.nonProxyHosts.
 - Dans le cas d'un serveur HTTPS, utilisez la propriété lisa.http.webProxy.ssl.nonProxyHosts.
7. Veillez à supprimer les symboles de commentaires qui figurent devant les propriétés.
8. Enregistrez le fichier et fermez la fenêtre.

Exemple :

Les deux premières lignes de l'exemple suivant spécifient que l'URL du serveur proxy HTTP est **http://192.168.24.242:49185**.

La troisième ligne spécifie que les hôtes qui ne passeront pas par ce proxy incluent l'adresse de bouclage de l'hôte local (127.0.0.1) et les adresses IP comprises dans la plage 192.168.32.0 à 192.168.32.255. Notez que la barre verticale (|) est utilisée comme délimiteur entre les adresses IP à exclure. Vous noterez également que le caractère générique (*) représente toute valeur valide, les valeurs valides d'un noeud d'adresse IP allant de zéro à 255. Vous pouvez également utiliser le caractère générique avec des noms de domaine complets et des noms d'hôte, lorsque les hôtes à exclure partagent une convention d'attribution de nom standard.

```
lisa.http.webProxy.host=192.168.24.242
lisa.http.webProxy.port=49185
lisa.http.webProxy.nonProxyHosts=127.0.0.1|192.168.32.*
```

Paramètres de serveur proxy HTTP/S dans le fichier local.properties

```
## =====
## HTTP Proxy Server
## =====
#lisa.http.webProxy.host=<machine name or ip>
##list of excluded machine names or ip addresses delimited by pipes, * wildcard
accepted <machine name or ip>[|<machine name or ip>]*
lisa.http.webProxy.nonProxyHosts=127.0.0.1
#lisa.http.webProxy.port=


## =====
## HTTPS Proxy Server
## =====
#lisa.http.webProxy.ssl.host=<machine name or ip>
##list of excluded machine names or ip addresses delimited by pipes, * wildcard
accepted <machine name or ip>[|<machine name or ip>]*
lisa.http.webProxy.ssl.nonProxyHosts=127.0.0.1
#lisa.http.webProxy.ssl.port=


## === Leave blank to use integrated NTLM authentication
#lisa.http.webProxy.host.domain= used for NTLM authentication
#lisa.http.webProxy.host.account=
#lisa.http.webProxy.host.credential=


## === Exclude simple host names from proxy use - default value is true
#lisa.http.webProxy.nonProxyHosts.excludeSimple=false


## === Preemptively send authorization information rather than waiting for a
challenge
## ===== valid values are basic or ntlm
#lisa.http.webProxy.preemptiveAuthenticationType=ntlm
```

Exécution des composants sur différents systèmes

Si les composants de serveur sont installés sur différents systèmes, veillez à utiliser les propriétés suivantes :

- Le registre utilise la propriété **lisa.registryName** pour s'attribuer un nom différent de la valeur par défaut.
- Les autres composants de serveur utilisent la propriété **lisa.registry.url** comme localisateur.

Dans le fichier **local.properties** des composants de serveur autres que le registre, spécifiez le registre à l'aide de la propriété **lisa.registry.url**.

```
lisa.registry.url=tcp://nom-hôte-ou-adresse-IP-registre:port/nom-registre
```

Par exemple :

```
lisa.registry.url=tcp://myserver.example.com:2010/Registry
```

N'utilisez pas la propriété **lisa.registryName** à cet effet.

Une autre option permettant de spécifier le registre consiste à transmettre l'argument **-m** lors du démarrage du composant de serveur autre que le registre.

```
./CoordinatorServer -m tcp://nom-hôte-ou-adresse-IP-registre:port/nom-registre
```

Par exemple :

```
./CoordinatorServer -m tcp://myserver.example.com:2010/Registry
```

Calcul des instances de simulateur

Pour calculer le nombre d'instances d'un simulateur spécifique, effectuez l'analyse suivante :

1. Démarrez DevTest Workstation, sélectionnez le registre et connectez-vous. Notez l'utilisation de la mémoire dans Help (Aide), DevTest Runtime Info (Informations d'exécution DevTest).
2. Exécutez la suite de tests au niveau local et notez l'utilisation de la mémoire dans Help (Aide), DevTest Runtime Info (Informations d'exécution DevTest).
3. Notez la différence entre l'utilisation de la mémoire à l'étape 2 et à l'étape 1.
4. Multipliez votre mémoire RAM disponible par 60 %.
5. Divisez la mémoire RAM disponible à l'étape 4 par l'utilisation de la mémoire à l'étape 3.

Le résultat de l'étape 5 est une estimation de départ convenable du nombre d'utilisateurs virtuels (instances) dont la configuration est requise dans votre serveur de simulation.

Si le serveur de coordination et le registre sont exécutés sur le même serveur que le serveur de simulation, multipliez la mémoire RAM disponible par 40 %. Utilisez 40 % au lieu de 60 %, car le serveur de coordination collecte tous les rapports et toutes les mesures ; par conséquent, il consomme la mémoire RAM.

Cette technique fournit un point de départ. Pour obtenir le nombre correct d'instances pour chaque simulateur, utilisez plusieurs itérations et d'autres méthodes intuitives.

Remarque : Vous pouvez définir le nombre d'instances simultanées pour un simulateur à l'aide d'une option de ligne de commande. Ouvrez une invite de commande, accédez au répertoire LISA_HOME\bin et saisissez **simulator --help** pour plus d'informations.

Dimensionnement de serveur de charge et de performances

Il est difficile de calculer le nombre de serveurs de simulation nécessaires pour un test de charge particulier. Le nombre de serveurs requis dépend de nombreux facteurs, notamment :

- Configuration d'hôte de serveur (nombre d'unités centrales, quantité de mémoire RAM)
- Empreinte de scénario de test (nombre d'étapes de test, type d'étapes de test)
- Autre configuration requise de test (nombre de rapports, taille d'ensembles de données)

Il est recommandé d'effectuer plusieurs exécutions de test de performances. Ces exécutions de test permettent de collecter des données qui peuvent être utiles pour déterminer la configuration de votre environnement de DevTest Server. La collecte de mesures et la surveillance de l'utilisation de la mémoire et de l'UC s'avèrent très utiles pour évaluer le nombre d'utilisateurs virtuels que vous pouvez utiliser sur un serveur de simulation spécifique.

Le registre est léger et requiert peu de ressources informatiques. Vous pouvez exécuter le registre à partir de pratiquement tous les ordinateurs de votre réseau.

Le serveur de coordination requiert des ressources. Bien que le serveur de coordination ne requière pas d'ordinateur propre, son installation sur un ordinateur distinct est une pratique commune. Suivez cette pratique si vous collectez de nombreuses mesures et/ou sollicitez de nombreux rapports.

Les serveurs de simulation sont utilisés pour simuler des milliers d'utilisateurs virtuels. Il est recommandé d'exécuter un serveur de simulation par serveur physique.

Techniquement, vous pouvez démarrer un serveur de simulation unique avec autant d'instances que vous le voulez. Toutefois, la taille de la mémoire et la vitesse du serveur limitent généralement le nombre d'instances de chaque simulateur. 250 utilisateurs virtuels constitue une limite supérieure correcte.

Vous pouvez utiliser la mise à l'échelle verticale ou horizontale pour le dimensionnement du serveur. Dans le cas de la mise à l'échelle verticale, vous augmentez la vitesse d'UC et la mémoire disponible, qui sont généralement limitées. Dans le cas de la mise à l'échelle horizontale, vous ajoutez d'autres serveurs. Pour augmenter le nombre d'utilisateurs virtuels, la mise à l'échelle horizontale est recommandée.

Le nombre d'instances par simulateur dépend de nombreux facteurs. Il n'existe pas de règle simple pour calculer le nombre d'instances maximum.

La latence du réseau affecte la charge et les performances. Il est recommandé d'héberger la base de données sur un serveur situé dans le même centre de données que les composants DevTest principaux.

Utilisation de DevTest Workstation dans un environnement Java

Vous pouvez remplacer le JRE par défaut utilisé par l'installation de DevTest par votre propre environnement Java. Pour cela, il est important de comprendre la procédure de sélection par DevTest Workstation du JRE à utiliser.

La priorité suivante est utilisée pour sélectionner la machine virtuelle Java à utiliser lors du démarrage de DevTest Workstation :

1. Le JRE installé avec DevTest Workstation dans le répertoire LISA_HOME\jre
2. Variable d'environnement LISA_JAVA_HOME
3. variable d'environnement JAVA_HOME
4. variable d'environnement JDK_HOME

Procédez comme suit:

1. Renommez le répertoire LISA_HOME\jre (par exemple, remplacez jre par jre_default).
2. Pointez la variable d'environnement LISA_JAVA_HOME vers votre répertoire d'installation Java.

Reportez-vous également à la section [Spécification de votre propre machine virtuelle Java](#) (page 13).

Modification de l'emplacement d'accueil de projet par défaut

Par défaut, les projets sont enregistrés dans le répertoire **LISA_HOME\Projects**.

Cette procédure décrit la procédure permettant de modifier comment l'emplacement par défaut du portail DevTest. Vous ne pouvez pas changer l'emplacement par défaut pour la station de travail DevTest Workstation.

Procédez comme suit:

1. Accédez au répertoire **LISA_HOME**.
2. Créez un fichier texte nommé **res-hub-config.properties**.
3. Ajoutez la propriété **resHub.projects.dir** au fichier. Veillez à utiliser des barres obliques dans le chemin d'accès au répertoire, même sous des plates-formes Windows. Par exemple :
`resHub.projects.dir=C:/MyNewProjectHome`
4. Enregistrez et fermez le fichier.
5. Si le composant de serveur du portail est exécuté, redémarrez-le.

Structure des répertoires de projet

Il est recommandé de mettre les actifs de test (par exemple, des projets) à la disposition des composants de serveur qui les utilisent.

Pour gérer l'accès aux actifs de test, la configuration requise se présente comme suit :

- Utilisez des normes relatives à l'attribution de noms. Plusieurs équipes peuvent utiliser le même environnement de serveur. Pour distinguer l'appartenance et l'objectif, ainsi que pour maintenir l'ordre, utilisez des normes relatives à l'attribution de noms.
- Les noms de projet doivent être uniques. Dans l'environnement de serveur, si deux projets déployés portent le même nom, un comportement inattendu peut se produire.

Désinstallation de DevTest Server

Le niveau supérieur du répertoire LISA_HOME inclut une application de désinstallation. Pour désinstaller les composants du DevTest Server, utilisez l'assistant de désinstallation de DevTest Solutions.

Procédez comme suit:

1. Arrêtez DevTest Solutions.
 - a. Vérifiez que tous les utilisateurs sont déconnectés des stations de travail de DevTest, du portail, de la console DevTest et du tableau de bord Enterprise Dashboard.
 - b. Vérifiez qu'aucun utilitaire de ligne de commande de DevTest n'est en cours d'exécution.
 - c. Fermez les fichiers exécutables ou arrêtez les services dans l'ordre suivant : simulateur, coordinateur, VSE, intermédiaire, portail, registre et tableau de bord Enterprise Dashboard.
2. (Facultatif) Supprimez le répertoire dans lequel les fichiers journaux principaux se trouvent.
 - a. Si les journaux sont stockés dans un répertoire personnalisé, ouvrez la station de travail DevTest Workstation et sélectionnez DevTest Runtime Info (Informations sur l'exécution de DevTest) dans le menu Help (Aide) et accédez à lisa.tmpdir. Le chemin est la valeur de lisa.tmpdir.
 - b. Accédez au répertoire lisatmp_*numéro-version*. L'emplacement par défaut de chaque répertoire de journaux spécifique aux versions (lisatmp_*numéro-version*) est le répertoire USER_HOME.
 - c. Cliquez avec le bouton droit de la souris sur le répertoire et sélectionnez Delete (Supprimer). Dans l'invite de confirmation, cliquez sur Yes.
3. Si vous désinstallez une installation de type partagé, supprimez manuellement le fichier lisa.user.properties. L'emplacement par défaut est le répertoire USE_HOME de l'utilisateur qui a installé DevTest.

Important : Si vous ne supprimez pas ce fichier, les installations de type local ultérieures ne pourront pas être effectuées.
4. Démarrez le processus de désinstallation.
 - (Windows) Vous pouvez démarrer cette application à partir de l'élément suivant :
 - Option du menu Démarrer de Windows
 - Programme de désinstallation de DevTest Solutions
 - Panneau de configuration, Programmes, Programmes et fonctionnalités, fenêtre Désinstaller ou modifier un programme
 - UNIX ou Linux : Ouvrez DevTest, cliquer sur Uninstall et sélectionnez Run.

L'étape DevTest Solutions Uninstall (Désinstaller DevTest Solutions) s'ouvre.

5. Cliquez sur Next (Suivant).
6. Pour supprimer des dossiers de la base de données, hotDeploy, lib/core et des préférences utilisateur associées, sélectionnez la case à cocher Delete all files (Supprimer tous les fichiers).
7. Cliquez sur Next (Suivant).
Si vous avez décidé de ne pas supprimer tous les fichiers, l'étape Results of Uninstaller (Résultats du programme de désinstallation) s'ouvre et contient la liste de fichiers qui n'ont pas été supprimés.
8. Cliquez sur Finish (Terminer).

Chapitre 4: Installation de DevTest Workstation

Vous pouvez installer DevTest Server avec une station de travail DevTest Workstation intégrée ou installer DevTest Workstation comme application autonome.

Cette section décrit la procédure d'installation et de configuration de DevTest Workstation comme application autonome.

A l'issue de l'installation et de la configuration de DevTest Workstation, vous pouvez vous connecter en suivant les étapes indiquées à la section Ouverture de DevTest Workstation de la rubrique *Utilisation de CA Application Test*. Si la station de travail DevTest Workstation est installée sur votre ordinateur sans DevTest Server, spécifiez un registre exécuté sur un ordinateur distant. Les installations de DevTest Workstation n'incluent pas de registre local.

Pour plus d'informations, reportez-vous à la section [Utilisation de DevTest Workstation avec votre environnement Java](#) (page 74).

Ce chapitre traite des sujets suivants :

[Installation de DevTest Workstation sous Windows](#) (page 80)

[Installation de DevTest Workstation sous UNIX](#) (page 82)

[Installation de la station de travail DevTest Workstation sur un Mac](#) (page 85)

[Utilisation d'un serveur proxy HTTP/S - DevTest Workstation](#) (page 87)

[Paramètres d'environnement](#) (page 89)

Installation de DevTest Workstation sous Windows

Cette rubrique décrit la procédure d'installation de la station de travail DevTest Workstation avec ou sans le serveur de démonstration dans un environnement Windows.

Avant de lancer la procédure d'installation, téléchargez les fichiers suivants à partir du [Download Center](#) (page 41) (Centre de téléchargement) :

- Le programme d'installation pour votre plate-forme
- (Facultatif) Le fichier .zip du serveur de démonstration

Procédez comme suit:

1. Exécutez le fichier du programme d'installation ; par exemple : devtest_win_x64.exe.
L'assistant Welcome to the DevTest Solutions Setup Wizard (Bienvenue dans l'assistant d'installation de DevTest Solutions) s'ouvre.
2. Cliquez sur Next (Suivant).
Le contrat de licence d'utilisateur final CA s'affiche.
3. Lisez le contrat de licence, sélectionnez l'option I accept the terms of the License Agreement (J'accepte les termes de cette licence), puis cliquez sur Next (Suivant).
L'étape Select Destination Directory (Sélectionner le répertoire de destination) s'ouvre.
4. Spécifiez le dossier dans lequel vous voulez installer un ou plusieurs composants de DevTest Solutions. Vous pouvez utiliser un répertoire qui contient des espaces ; par ex. : **C:\Program Files\CA\DevTestSolutions**. Si vous spécifiez un dossier qui n'existe pas, l'assistant d'installation le créera.
5. Cliquez sur Next (Suivant).
La page Installation Type (Type d'installation) s'affiche.
6. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

Local

Tous les composants de DevTest Solutions sont installés dans un répertoire unique sur l'ordinateur local. Par défaut, toutes les données sont stockées dans ce répertoire et chaque utilisateur est associé à un répertoire temporaire personnel. Ce type d'installation est le plus commun et est utilisé dans la plupart des environnements.

Shared (Partagé)

Les administrateurs utilisent ce type d'installation pour installer tous les composants des DevTest Solutions dans un emplacement partagé auquel les utilisateurs peuvent accéder à partir de plusieurs ordinateurs. Toutes les données et les fichiers temporaires sont stockés dans des répertoires spécifiés par l'utilisateur. Chaque utilisateur dispose de ses données personnelles, mais tous partagent une installation de DevTest Solutions commune. Dans une installation partagée, les utilisateurs requièrent uniquement des droits d'accès en lecture au répertoire de programmes des DevTest Solutions.

L'étape Select Components (Sélectionner des composants) s'ouvre.

7. Désélectionnez la case à cocher Server (Serveur), vérifiez que la case à cocher Workstation (Station de travail) est sélectionnée et cliquez sur Next (Suivant).
Si vous sélectionnez le type d'installation partagé, vous êtes invité à spécifier le répertoire de données et le répertoire de fichiers temporaire.
8. Spécifiez les répertoires et cliquez sur Next (Suivant) après chaque étape.
L'étape Demo Server (Serveur de démonstration) s'ouvre.
9. Si vous voulez que le programme d'installation décomprime le serveur de démonstration dans le répertoire LISA_HOME, sélectionnez l'option Install demo server (Installer le serveur de démonstration) et spécifiez le chemin complet du fichier .zip du serveur de démonstration.
10. Cliquez sur Next (Suivant).
L'étape Select Start Menu Folder (Sélectionner le dossier dans le menu Démarrer) s'ouvre.
11. Spécifiez si la station de travail DevTest Workstation doit être ajoutée à votre menu Démarrer et si cela s'applique également à d'autres utilisateurs.
 - Pour créer un dossier dans le menu Démarrer avec des raccourcis pour tous les utilisateurs, acceptez toutes les valeurs par défaut. Vous pouvez également saisir un nouveau nom de dossier.
 - Si vous ne souhaitez pas créer un dossier dans le menu Démarrer, désélectionnez la case à cocher Create a Start Menu folder (Créer un dossier dans le menu Démarrer).
 - Pour créer un dossier dans le menu Démarrer et limiter l'affichage des raccourcis à votre menu Démarrer :
 - Acceptez la sélection de Create a Start Menu folder (Créer un dossier dans le menu Démarrer).
 - Acceptez le nom par défaut ou saisissez en un autre.
 - Désélectionnez la case à cocher Create shortcuts for all users (Créer des raccourcis pour tous les utilisateurs).
12. Cliquez sur Next (Suivant).
L'étape Select File Associations (Sélectionner des associations de fichiers) s'ouvre.

13. Sélectionnez toutes les extensions à utiliser avec les didacticiels d'exemples de projet disponibles à la section *Utilisation de CA Application Test*. Les extensions de fichier que vous pouvez associer à DevTest Solutions sont les suivantes :
 - *.tst -- sélectionnez cette extension pour créer des scénarios de test avec CA Application Test.
 - *.vsm et *.vsi -- sélectionnez ces extensions pour créer des services virtuels avec CA Service Virtualization.
 - *.ste -- sélectionnez cette extension pour exécuter une suite avec l'exécuteur de tests dans CA Application Test.
 - *.stg -- sélectionnez cette extension pour exécuter un scénario de test comme document de simulation dans CA Application Test.
14. Cliquez sur **Install (Installer)** pour lancer l'installation.
A l'issue de l'installation, l'étape **Information** s'ouvre.
15. Lisez les informations et cliquez sur **Next (Suivant)**.
L'étape de l'assistant **Completing the DevTest Solutions Setup (Finalisation de l'installation de DevTest Solutions)** s'ouvre.
16. Cliquez sur **Finish (Terminer)**.

Installation de DevTest Workstation sous UNIX

Cette rubrique décrit la procédure d'installation de DevTest Workstation dans un environnement UNIX ou Linux.

Avant de lancer la procédure d'installation, téléchargez les fichiers suivants à partir du [Download Center](#) (page 41) (Centre de téléchargement) :

- Le programme d'installation pour votre plate-forme
- (Facultatif) Le fichier .zip du serveur de démonstration

La procédure suivante correspond à la version graphique du programme d'installation. Pour utiliser la version de la ligne de commande du programme d'installation, ajoutez l'option **-c**. Par exemple :

```
./devtest_linux_x64.sh -c
```

Remarque : Si vous utilisez le programme d'installation UNIX générique, vérifiez qu'une [machine virtuelle Java](#) (page 13) est installée sur le même ordinateur. La version de la machine virtuelle Java doit être 1.7. Vous pouvez spécifier une machine virtuelle Java en définissant la variable d'environnement **JAVA_HOME**. Si le programme d'installation ne parvient pas à détecter une machine virtuelle Java, il affichera un message avant de se fermer.

Procédez comme suit:

1. Dans une fenêtre de terminal, accédez au répertoire dans lequel le fichier de programme d'installation est placé.

2. Vérifiez que le fichier de programme d'installation dispose de l'autorisation d'exécution.

```
chmod 777 devtest_platform_x64.sh
```

Cette commande octroie les autorisations rwxrwxrwx sur le fichier.

3. Exécutez le fichier d'installation. Double-cliquez sur l'icône ou saisissez une commande similaire à la commande suivante à partir d'une fenêtre de terminal :

```
./devtest_platform_x64.sh
```

L'assistant DevTest Solutions Setup (Installation des DevTest Solutions) s'ouvre.

4. Cliquez sur Next (Suivant).

Le contrat de licence d'utilisateur final CA s'affiche.

5. Lisez le contrat de licence, sélectionnez la case à cocher I accept the terms of the License Agreement (J'accepte les termes de cette licence), puis cliquez sur Next (Suivant).

L'étape Select Destination Directory (Sélectionner le répertoire de destination) s'ouvre.

6. Spécifiez le répertoire dans lequel vous voulez installer la station de travail DevTest Workstation. N'utilisez pas un répertoire contenant des espaces. Le chemin par défaut est /opt/CA/DevTest.

7. Cliquez sur Next (Suivant).

La page Installation Type (Type d'installation) s'affiche.

8. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

Local

Tous les composants de DevTest Solutions sont installés dans un répertoire unique sur l'ordinateur local. Par défaut, toutes les données sont stockées dans ce répertoire et chaque utilisateur est associé à un répertoire temporaire personnel. Ce type d'installation est le plus commun et est utilisé dans la plupart des environnements.

Shared (Partagé)

Les administrateurs utilisent ce type d'installation pour installer tous les composants des DevTest Solutions dans un emplacement partagé auquel les utilisateurs peuvent accéder à partir de plusieurs ordinateurs. Toutes les données et les fichiers temporaires sont stockés dans des répertoires spécifiés par l'utilisateur. Chaque utilisateur dispose de ses données personnelles, mais tous partagent une installation de DevTest Solutions commune. Dans une installation partagée, les utilisateurs requièrent uniquement des droits d'accès en lecture au répertoire de programmes des DevTest Solutions.

L'étape Select Components (Sélectionner des composants) s'ouvre.

9. Désélectionnez la case à cocher Server (Serveur), vérifiez que la case à cocher Workstation (Station de travail) est sélectionnée et cliquez sur Next (Suivant).

Si vous sélectionnez le type d'installation partagé, vous êtes invité à spécifier le répertoire de données et le répertoire de fichiers temporaire.

10. Spécifiez les répertoires et cliquez sur Next (Suivant) après chaque étape.

L'étape Specify Demo Server (Spécifier un serveur de démonstration) s'ouvre.

11. Si vous voulez que le programme d'installation décomprime le serveur de démonstration dans le répertoire **LISA_HOME**, sélectionnez la case à cocher Install demo server (Installer le serveur de démonstration) et spécifiez le chemin complet du fichier .zip du serveur de démonstration.

12. Cliquez sur Next (Suivant).

L'étape Select Additional Tasks (Sélectionner des tâches supplémentaires) s'ouvre.

13. Si vous ne voulez pas créer d'icône de bureau pour DevTest, désélectionnez la case à cocher.

14. Cliquez sur Install (Installer).

A l'issue de l'installation, l'étape Information s'ouvre.

15. Lisez les informations et cliquez sur Next (Suivant).

L'étape de l'assistant Completing the DevTest Solutions Setup (Finalisation de l'installation de DevTest Solutions) s'ouvre.

16. Cliquez sur Finish (Terminer).

Installation de la station de travail DevTest Workstation sur un Mac

Cette rubrique décrit la procédure d'installation de la station de travail DevTest Workstation avec ou sans le serveur de démonstration dans un environnement Mac.

Avant de lancer la procédure d'installation, téléchargez les fichiers suivants à partir du [Download Center](#) (page 41) (Centre de téléchargement) :

- Le programme d'installation pour votre plate-forme
- (Facultatif) Le fichier .zip du serveur de démonstration

Procédez comme suit:

1. Exécutez le fichier du programme d'installation ; par exemple : devtest_osx_x64.dmg.

L'assistant Welcome to the DevTest Solutions Setup Wizard (Bienvenue dans l'assistant d'installation de DevTest Solutions) s'ouvre.

2. Cliquez sur Next (Suivant).

Le contrat de licence d'utilisateur final CA s'affiche.

3. Lisez le contrat de licence, sélectionnez l'option I accept the terms of the License Agreement (J'accepte les termes de cette licence), puis cliquez sur Next (Suivant).

L'étape Select Destination Directory (Sélectionner le répertoire de destination) s'ouvre.

4. Spécifiez le dossier dans lequel vous voulez installer un ou plusieurs composants de DevTest Solutions. Si vous spécifiez un dossier qui n'existe pas, l'assistant d'installation le créera.

5. Cliquez sur Next (Suivant).

La page Installation Type (Type d'installation) s'affiche.

6. Sélectionnez l'une des options suivantes, puis cliquez sur Next (Suivant).

Local

Tous les composants de DevTest Solutions sont installés dans un répertoire unique sur l'ordinateur local. Par défaut, toutes les données sont stockées dans ce répertoire et chaque utilisateur est associé à un répertoire temporaire personnel. Ce type d'installation est le plus commun et est utilisé dans la plupart des environnements.

Shared (Partagé)

Les administrateurs utilisent ce type d'installation pour installer tous les composants des DevTest Solutions dans un emplacement partagé auquel les utilisateurs peuvent accéder à partir de plusieurs ordinateurs. Toutes les données et les fichiers temporaires sont stockés dans des répertoires spécifiés par l'utilisateur. Chaque utilisateur dispose de ses données personnelles, mais tous partagent une installation de DevTest Solutions commune. Dans une installation partagée, les utilisateurs requièrent uniquement des droits d'accès en lecture au répertoire de programmes des DevTest Solutions.

L'étape Select Components (Sélectionner des composants) s'ouvre.

7. Désélectionnez la case à cocher Server (Serveur), vérifiez que la case à cocher Workstation (Station de travail) est sélectionnée et cliquez sur Next (Suivant).

L'étape Demo Server (Serveur de démonstration) s'ouvre.

8. Si vous voulez que le programme d'installation décomprime le serveur de démonstration, sélectionnez l'option Install demo server (Installer le serveur de démonstration) et spécifiez le chemin complet du fichier .zip du serveur de démonstration.
9. Cliquez sur Next (Suivant).

L'étape Select Additional Tasks (Sélectionner des tâches supplémentaires) s'ouvre.

10. (Facultatif) Si vous ne voulez pas créer d'icône de bureau pour DevTest Workstation, désélectionnez la case à cocher.
11. Cliquez sur Next (Suivant).

L'étape Select File Associations (Sélectionner des associations de fichiers) s'ouvre.

12. Sélectionnez toutes les extensions à utiliser avec les didacticiels d'exemples de projet disponibles à la section *Utilisation de CA Application Test*. Les extensions de fichier que vous pouvez associer à DevTest Solutions sont les suivantes :
 - *.*tst* -- sélectionnez cette extension pour créer des scénarios de test avec CA Application Test.
 - *.*vsm* et *.*vsi* -- sélectionnez ces extensions pour créer des services virtuels avec CA Service Virtualization.
 - *.*ste* -- sélectionnez cette extension pour exécuter une suite avec l'exécuteur de tests dans CA Application Test.
 - *.*stg* -- sélectionnez cette extension pour exécuter un scénario de test comme document de simulation dans CA Application Test.

13. Cliquez sur **Install (Installer)** pour lancer l'installation.

A l'issue de l'installation, l'étape Information s'ouvre.

14. Lisez les informations et cliquez sur Next (Suivant).

L'étape de l'assistant Completing the DevTest Solutions Setup (Finalisation de l'installation de DevTest Solutions) s'ouvre.

15. Cliquez sur **Finish (Terminer)**.

Utilisation d'un serveur proxy HTTP/S - DevTest Workstation

Si vous utilisez un serveur proxy HTTP simple ou un serveur proxy HTTP sécurisé via SSL, définissez ce serveur proxy et tous les hôtes à exclure dans le fichier **local.properties** du répertoire LISA_HOME.

Procédez comme suit:

1. Connectez-vous à l'hôte sur lequel DevTest Workstation est installé.
2. Accédez à LISA_HOME.
3. Si le fichier local.properties n'existe pas, copiez le fichier **_local.properties** et enregistrez la copie comme local.properties (sans le trait de soulignement).
4. Ouvrez le fichier local.properties pour le modifier et recherchez l'en-tête de section pour le serveur proxy HTTP ou HTTPS, selon que votre serveur proxy utilise une connexion HTTP simple ou sécurisée via SSL.
5. Identifiez votre serveur proxy par son nom de domaine complet ou l'adresse IP et le port :
 - Dans le cas d'un serveur HTTP, utilisez les propriétés lisa.http.webProxy.host et lisa.http.webProxy.port.
 - Dans le cas d'un serveur HTTPS, utilisez les propriétés lisa.http.webProxy.ssl.host et lisa.http.webProxy.ssl.port.
6. Identifiez les hôtes à exclure de votre serveur proxy :
 - Dans le cas d'un serveur HTTP, utilisez la propriété lisa.http.webProxy.nonProxyHosts.
 - Dans le cas d'un serveur HTTPS, utilisez la propriété lisa.http.webProxy.ssl.nonProxyHosts.
7. Veillez à supprimer les symboles de commentaires qui figurent devant les propriétés.
8. Enregistrez le fichier et fermez la fenêtre.

Exemple :

Les deux premières lignes de l'exemple suivant spécifient que l'URL du serveur proxy HTTP est **http://192.168.24.242:49185**.

La troisième ligne spécifie que les hôtes qui ne passeront pas par ce proxy incluent l'adresse de bouclage de l'hôte local (127.0.0.1) et les adresses IP comprises dans la plage 192.168.32.0 à 192.168.32.255. Notez que la barre verticale (|) est utilisée comme délimiteur entre les adresses IP à exclure. Vous noterez également que le caractère générique (*) représente toute valeur valide, les valeurs valides d'un noeud d'adresse IP allant de zéro à 255. Vous pouvez également utiliser le caractère générique avec des noms de domaine complets et des noms d'hôte, lorsque les hôtes à exclure partagent une convention d'attribution de nom standard.

```
lisa.http.webProxy.host=192.168.24.242
lisa.http.webProxy.port=49185
lisa.http.webProxy.nonProxyHosts=127.0.0.1|192.168.32.*
```

Paramètres de serveur proxy HTTP/S dans le fichier local.properties

```
## =====
## HTTP Proxy Server
## =====
#lisa.http.webProxy.host=<machine name or ip>
##list of excluded machine names or ip addresses delimited by pipes, * wildcard
accepted <machine name or ip>[|<machine name or ip>]*
lisa.http.webProxy.nonProxyHosts=127.0.0.1
#lisa.http.webProxy.port=


## =====
## HTTPS Proxy Server
## =====
#lisa.http.webProxy.ssl.host=<machine name or ip>
##list of excluded machine names or ip addresses delimited by pipes, * wildcard
accepted <machine name or ip>[|<machine name or ip>]*
lisa.http.webProxy.ssl.nonProxyHosts=127.0.0.1
#lisa.http.webProxy.ssl.port=


## == Leave blank to use integrated NTLM authentication
#lisa.http.webProxy.host.domain= used for NTLM authentication
#lisa.http.webProxy.host.account=
#lisa.http.webProxy.host.credential=


## == Exclude simple host names from proxy use - default value is true
#lisa.http.webProxy.nonProxyHosts.excludeSimple=false


## == Preemptively send authorization information rather than waiting for a
challenge
## == valid values are basic or ntlm
#lisa.http.webProxy.preemptiveAuthenticationType=ntlm
```

Paramètres d'environnement

La documentation de DevTest mentionne un jeton nommé **%LISA_HOME%** (Windows) ou **\$LISA_HOME** (OSX ou UNIX). Ce jeton indique l'emplacement d'installation de DevTest Solutions.

Sur tous les systèmes d'exploitation pris en charge, une variable d'environnement est automatiquement définie avec ce nom à partir des scripts ou des programmes de lancement.

Par exemple, si vous avez effectué l'installation de DevTest Server dans **C:\DevTest_numéro_version**, la valeur sera **%LISA_HOME%**. DevTest Workstation peut également accéder à la valeur de cette variable dans une propriété nommée **LISA_HOME**.

Pour placer d'autres fichiers JAR, ZIP, ou répertoires dans la variable classpath de DevTest, vous disposez de deux options :

- Définissez la variable d'environnement **LISA_POST_CLASSPATH** et les ressources que vous voulez placer dans cette variable.
- Placez-les dans le répertoire **%LISA_HOME%/hotDeploy**.

Remarque : Pour plus d'informations sur les paramètres d'environnement, reportez-vous à la section Propriétés communes et variables d'environnement de la rubrique *Utilisation de CA Application Test*.

Chapitre 5: Installation du serveur de démonstration

Le serveur de démonstration facultatif est un serveur d'applications JBoss 4.2.3 qui comporte plusieurs applications à des fins de démonstration des fonctionnalités de DevTest.

- Le projet **examples** contient des scénarios de test qui utilisent le serveur de démonstration.
- Certains des didacticiels de la rubrique *Utilisation de CA Application Test* utilisent le serveur de démonstration.

Remarque : Le serveur de démonstration utilise le port 1529. Ce port ne doit donc pas être utilisé par une autre application. Si ce port n'est pas disponible, le serveur de démonstration ne démarrera pas correctement.

Procédez comme suit:

1. Téléchargez le serveur de démonstration tel que décrit dans la section [Téléchargement des programmes d'installation de DevTest](#) (page 41).
2. Sélectionnez l'une des méthodes d'installation suivantes :
 - (Recommandé) Installez DevTest Solutions (serveur ou station de travail) et utilisez l'assistant d'installation pour décompresser le fichier dans le même répertoire d'installation (**LISA_HOME**). L'assistant d'installation crée également une icône de bureau pour le serveur de démonstration.
 - Décomprimez le fichier **DevTestDemoServer.zip** sur votre ordinateur. Sous Windows, ce fichier est téléchargé dans votre dossier des téléchargements. Accédez au dossier **lisa-demo-server** et suivez les instructions indiquées dans le fichier Readme. Le fichier Readme contient des instructions spécifiques à la plate-forme pour démarrer le serveur de démonstration.
3. Si vous choisissez la deuxième approche, tenez compte des informations suivantes :
 - Java 7 doit être installé dans un répertoire différent sur votre système.
 - Définissez la variable d'environnement **JAVA_HOME**. Cette variable est requise par JBoss pour compiler et exécuter des fichiers JSP.
 - Ne placez pas le répertoire de serveur JBoss sur votre ordinateur de bureau ni aucun chemin contenant des espaces. JBoss ne peut pas compiler les fichiers JSP si le chemin vers son répertoire contient des espaces.

Remarques :

- Pour démarrer le serveur de démonstration à partir de la ligne de commande, accédez au répertoire **LISA_HOME\DemoServer\lisa-dem-server** et démarrez le script de votre système d'exploitation :
 - (Windows) **start-windows.bat**
 - (UNIX ou Linux) **start-unix-linux.sh**
 - (OS/X) **start-osx.command**
- Pour exécuter le serveur de démonstration sur UNIX ou Linux, utilisez le shell **/bin/bash**.
- Le serveur de démonstration exécute l'agent Java pour DevTest par défaut, en signalant autant d'informations que possible à CA Continuous Application Insight. Pour désactiver la génération de rapports, utilisez l'indicateur **-noagent**. Pour désactiver uniquement les informations relatives au segment de mémoire ou à la pile, utilisez l'indicateur **-noheaps**.
- Si des fichiers binaires d'agent natifs sont présents sur les distributions UNIX et Linux largement utilisées, le serveur de démonstration lancera plutôt l'agent natif. Pour que le serveur de démonstration utilise l'agent Java pur dans ce cas, transférez le paramètre **--javaagent** à la commande de démarrage. Les fichiers binaires d'agent natifs sont uniquement destinés à l'utilisation avec Java 1.4 et version antérieure.
- La base de données du serveur de démonstration est créée lors du premier démarrage de celui-ci. Cette base de données se trouve dans le répertoire **LISA_HOME\DemoServer\lisa-demo-server\jboss\server\default\data\lisa-demo-server.db**.
- Après le démarrage du serveur de démonstration, vous pouvez accéder au serveur à l'aide d'un navigateur sur le port 8080.

Chapitre 6: Vérification de l'installation de DevTest Solutions

Pour vérifier l'installation de DevTest Solutions, démarrez le DevTest Server, ouvrez chaque interface utilisateur et connectez-vous.

Ce chapitre traite des sujets suivants :

- [Démarrage de DevTest et connexion aux interfaces utilisateur](#) (page 93)
- [Démarrage de processus ou de services DevTest](#) (page 94)
- [Connexion en tant que superutilisateur standard](#) (page 98)
- [Création d'un utilisateur avec un rôle de superutilisateur](#) (page 100)
- [Accès aux interfaces utilisateur de DevTest](#) (page 101)

Démarrage de DevTest et connexion aux interfaces utilisateur

Procédez comme suit:

1. [Démarrez les composants de serveur](#) (page 94).
2. Préparez la connexion aux interfaces utilisateur. Vous pouvez :
 - [Connectez-vous en tant que superutilisateur DevTest standard](#) (page 98).
 - [Créez un utilisateur DevTest avec le rôle de superutilisateur](#) (page 100) pour vous-même et connectez-vous avec vos propres informations d'identification.
3. [Accédez à chaque interface utilisateur](#) (page 101) et connectez-vous. Examinez les interfaces utilisateur.
4. (Facultatif) Consultez les répertoires du DevTest Server.

Démarrage de processus ou de services DevTest

Cette section décrit le processus de démarrage du DevTest Server avec tous les composants disponibles. Utilisez la séquence indiquée pour assurer le démarrage de tous les composants. Quelques méthodes permettant de démarrer des processus ou services DevTest sont décrites après la procédure de démarrage.

Remarque : La section [A propos des composants du DevTest Server](#) (page 25) décrit ces processus.

Séquence de l'ordre de démarrage

Démarrez les processus (ou services) du DevTest Server dans l'ordre suivant. Les raccourcis du menu Démarrer sont affichés.

Procédez comme suit:

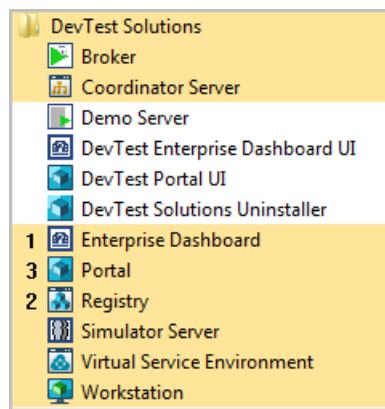
1. Démarrez le serveur du tableau de bord Enterprise Dashboard.
2. Démarrez chaque registre.
3. Démarrez le portail.
4. Démarrez les composants suivants dans l'ordre de votre choix :
 - Intermédiaire
 - Serveur de coordination
 - Le serveur de simulation associé à chaque coordinateur
 - Virtual Service Environment (Environnement de service virtuel)
5. En cas d'utilisation sur le DevTest Server :
 - Station de travail
 - Serveur de démonstration

Remarque : Pour arrêter les composants de serveur, appliquez l'ordre inverse.

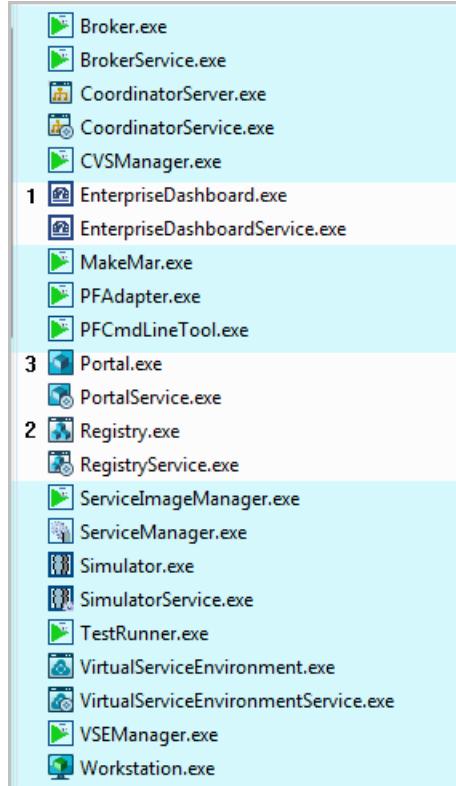
Méthodes de démarrage du DevTest Server

Accédez aux processus DevTest Server de l'une des manières suivantes :

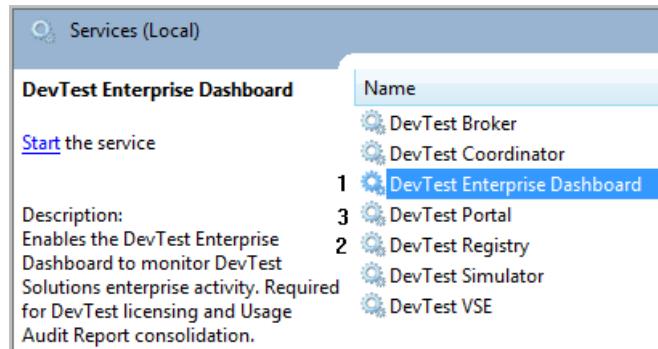
- (Windows) Cliquez sur le menu Démarrer et développez DevTest Solutions.
Démarrez les processus dans la séquence d'ordre de démarrage.



- Ouvrez le répertoire LISA_HOME\bin. Lancez les fichiers exécutables dans la séquence d'ordre de démarrage.

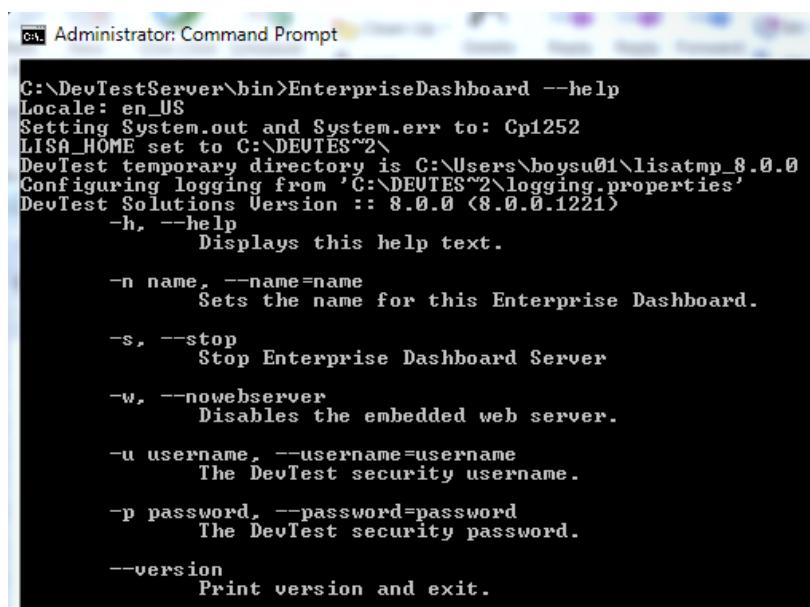


- (Windows) Sélectionnez Services et démarrez les services dans la séquence d'ordre de démarrage.



- Ouvrez une invite de commande en tant qu'administrateur ou ouvrez une fenêtre de terminal. Ouvrez LISA_HOME\bin, puis saisissez la commande pour démarrer chaque processus dans la séquence d'ordre de démarrage. Vous pouvez également démarrer les services respectifs ou lancer le fichier .vmoptions associé.

Vous pouvez obtenir de l'aide en saisissant le nom de la commande suivie de --help.
Par exemple :



```
G:\>DevTestServer\bin>EnterpriseDashboard --help
Locale: en_US
Setting System.out and System.err to: Cp1252
LISA_HOME set to C:\DEUTES^2\
DevTest temporary directory is C:\Users\boysu01\lisatmp_8.0.0
Configuring logging from 'C:\DEUTES^2\logging.properties'
DevTest Solutions Version :: 8.0.0 (8.0.0.1221)
-h, --help
    Displays this help text.

-n name, --name=name
    Sets the name for this Enterprise Dashboard.

-s, --stop
    Stop Enterprise Dashboard Server

-w, --nowebserver
    Disables the embedded web server.

-u username, --username=username
    The DevTest security username.

-p password, --password=password
    The DevTest security password.

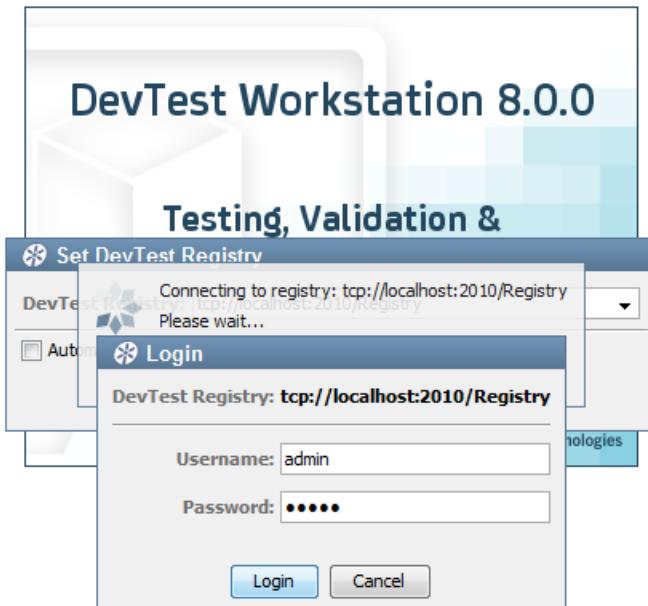
--version
    Print version and exit.
```

Connexion en tant que superutilisateur standard

Important : La liste de contrôle d'accès est obligatoire. Aucun utilisateur ne peut utiliser DevTest Solutions sans se connecter préalablement avec un nom et un mot de passe valides.

La configuration de la sécurité implique la création de comptes d'utilisateur basés sur les rôles et la spécification de la liste de contrôle d'accès avec l'authentification LDAP ou avec des informations d'identification que vous définissez. Si vous prévoyez d'utiliser l'authentification LDAP, n'ajoutez pas vos propres nom d'utilisateur et mot de passe, tels que décrit dans la rubrique [Création d'un utilisateur avec le rôle de superutilisateur](#) (page 100). Pour accéder aux interfaces utilisateur avant de configurer l'authentification LDAP, utilisez l'utilisateur standard défini avec le rôle de superutilisateur.

Lorsque vous accédez à la station de travail DevTest Workstation, une boîte de dialogue de connexion similaire au graphique suivant s'ouvre :



Pour vous connecter, saisissez *admin* dans le champ Username (Nom d'utilisateur) et *admin* dans le champ Password (Mot de passe), puis cliquez sur Login (Connexion). La station de travail DevTest Workstation s'ouvre.

Lorsque vous accédez au portail DevTest, la zone de connexion est similaire à l'exemple suivant.

The screenshot shows a web browser window for the 'DevTest Portal'. At the top, there's a dark header bar with the 'ca technologies' logo on the left and the text 'DevTest Portal' in the center. Below the header is a light-colored login form. The form has a heading 'Log In' in bold black text. It contains two text input fields: one labeled 'User Name' with the value 'admin' and another labeled 'Password' with the value '****'. At the bottom of the form is a large teal button with the text 'Log in' in white. A white hand cursor icon is positioned over the 'Log in' button, indicating it is the target of a click action.

Pour vous connecter, saisissez **admin** dans le champ User Name (Nom d'utilisateur) et **admin** dans le champ Password (Mot de passe), puis cliquez sur Login (Connexion).

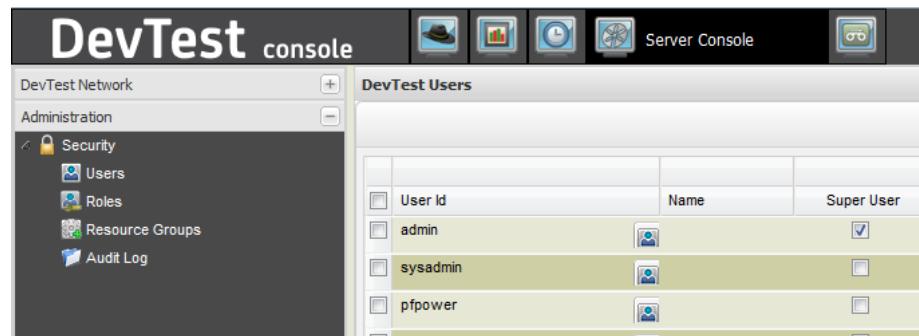
La console DevTest présente une boîte de dialogue de connexion similaire. Vous pouvez vous connecter à ce navigateur avec le même utilisateur standard.

Création d'un utilisateur avec un rôle de superutilisateur

Accordez-vous l'accès complet à DevTest Solutions.

Procédez comme suit:

1. Accédez à la console du serveur.
`http://localhost:1505`
2. Connectez-vous en tant qu'administrateur standard.
 - a. Entrez *admin* pour le nom d'utilisateur.
 - b. Saisissez *admin* pour le mot de passe.
3. Développez le volet Administration dans la barre de navigation gauche. Dans la zone Security, le superutilisateur fournit des informations d'identification pour authentifier les utilisateurs et accorde des autorisations d'accès à des fonctionnalités via des affectations de rôle.



The screenshot shows the DevTest console interface. The title bar says "DevTest console". The left sidebar has a tree view under "Administration" with "Security" expanded, showing "Users", "Roles", "Resource Groups", and "Audit Log". The main area is titled "DevTest Users" and contains a table with three columns: "User Id", "Name", and "Super User". There are four rows: "admin" (selected), "sysadmin", and "p1power" (both unselected). The "Super User" column for "admin" has a checked checkbox, while the others have empty checkboxes.

User Id	Name	Super User
admin		<input checked="" type="checkbox"/>
sysadmin		<input type="checkbox"/>
p1power		<input type="checkbox"/>

4. Créez un compte d'utilisateur pour vous avec un mot de passe que vous n'oublierez pas. Assignez un rôle de superutilisateur avec accès complet à DevTest Solutions.
 - a. Au bas du panneau de droite, cliquez sur Add User (Ajouter un utilisateur).
 - b. Dans le champ User ID (ID d'utilisateur), saisissez un ID unique composé d'une combinaison de caractères alphanumériques, trait d'union (-), trait de soulignement (_), point(.) et esperluette (@).
 - c. Dans le champ Password (Mot de passe), saisissez un mot de passe unique composé d'une combinaison de caractères alphanumériques, trait d'union (-), trait de soulignement (_), point(.) et esperluette (@).
 - d. Dans le champ Re-type Password (Confirmer le mot de passe), entrez à nouveau le mot de passe.
 - e. Dans le champ Name (Nom), entrez le nom d'utilisateur (caractères alphanumériques, trait d'union (-), trait de soulignement (_)) et d'espace).
 - f. Dans la zone Roles for the User (Rôles d'utilisateur), sélectionnez Super User (Superutilisateur).
 - g. Cliquez sur Add User (Ajouter un utilisateur).
 - h. Cliquez sur OK.

Vous pouvez désormais accéder à une interface utilisateur DevTest ou une interface de ligne de commande et vous connecter avec l'ID d'utilisateur et le mot de passe définis.

Accès aux interfaces utilisateur de DevTest

Le tableau de bord Enterprise Dashboard, le registre et le portail que vous utilisez doivent être exécutés pour toutes les interfaces utilisateur. Consultez la section [Démarrage des composants de serveur](#) (page 94).

Accédez aux interfaces utilisateur de la manière décrit dans cette section. Puis, [connectez-vous en tant que superutilisateur standard](#) (page 98) ou connectez-vous avec vos propres informations d'identification, en vous [définissant avec le rôle de superutilisateur](#) (page 100).

Serveur de démonstration

Pour en savoir plus sur DevTest Solutions, accédez au serveur de démonstration sur votre hôte local et visionnez les didacticiels de la rubrique *Utilisation de CA Application Test*.

- Accédez au serveur de démonstration et connectez-vous.
`http://localhost:8080/lisabank`
- (Windows) Dans le menu Démarrer, développez DevTest Solutions et sélectionnez Demo Server (Serveur de démonstration).

Enterprise Dashboard (Tableau de bord d'entreprise)

Pour afficher la liste de registres qui sont reliés au tableau de bord Enterprise Dashboard :

- Accédez à l'interface utilisateur du tableau de bord Enterprise Dashboard et connectez-vous.
`http://hostname:1506`
- (Windows) Connectez-vous au serveur sur lequel le tableau de bord Enterprise Dashboard est installé. Dans le menu Démarrer, développez DevTest Solutions et sélectionnez DevTest Enterprise Dashboard UI (Interface utilisateur du tableau de bord Enterprise Dashboard DevTest).

Portail

Avant d'utiliser CA Continuous Application Insight, installez l'agent Java pour DevTest. Accédez à l'URL suivante et connectez-vous au portail DevTest pour accéder à CAI. Spécifiez le nom d'hôte d'un ordinateur sur lequel un registre est exécuté. Consultez la documentation *Utilisation de CAI*.

L'exécution de CAI requiert le portail. L'exécution de l'interface utilisateur requiert l'intermédiaire. CA Application Test et CA Service Virtualization utilisent le portail pour accéder à certaines fonctionnalités. Consultez les rubriques *Utilisation de CA Application Test* et *Utilisation de CA Service Virtualization*.

- Accédez à l'interface utilisateur du portail et connectez-vous.
`http://nom-hôte:1507/devtest`
- (Windows) Dans le menu Démarrer, développez DevTest Solutions et sélectionnez DevTest Portal UI (Interface utilisateur du portail DevTest).

Station de travail

Démarrez DevTest Workstation sur votre hôte local. La station de travail est l'emplacement dans lequel vous créez, modifiez et simulez des tests. Il existe quelques méthodes alternatives, telles que TestRunner et tâche Ant junitlisa. Consultez la documentation *Utilisation de CA Application Test*. L'exécution de l'interface utilisateur requiert un serveur de coordination et un serveur de simulation.

- (Windows) Dans le menu Démarrer, développez DevTest Solutions et sélectionnez Workstation (Station de Travail), spécifiez le registre et connectez-vous.
- Accédez à LISA_HOME\bin et exécutez le fichier Workstation.exe.

Console DevTest

La console inclut des liens vers la console de serveur (y compris le VSE), le CVS Dashboard (Tableau de bord du service CVS), la génération de rapports et la console VSEEasy. Spécifiez le nom d'hôte d'un ordinateur sur lequel un registre est exécuté. Pour plus d'informations sur l'utilisation de l'onglet Administrative dans la console de serveur, reportez-vous à la rubrique *Administration de DevTest Solutions*. Pour plus d'informations, reportez-vous à la rubrique *Utilisation de CA Service Virtualization*. L'exécution de l'interface utilisateur de CA Service Virtualization requiert le VSE.

- Accédez à la console DevTest et connectez-vous.
`http://nom-hôte:1505`
- Ouvrez la Station de travail et cliquez sur le bouton de la barre d'outils de Server Console (Console de serveur).

Chapitre 7: Installation des outils d'intégration

Cette section décrit la procédure d'installation et de configuration d'outils tiers que vous pouvez utiliser avec DevTest.

Ce chapitre traite des sujets suivants :

- [Installation du moniteur de performances \(PerfMon\)](#) (page 106)
- [Installation et configuration de SNMP](#) (page 107)
- [Exécution de TCPMon](#) (page 110)
- [Installation du module d'extension HP ALM - Quality Center](#) (page 112)
- [Installation d'IBM Rational Quality Manager](#) (page 113)
- [Configuration de l'intégration à CA APM](#) (page 117)
- [Configuration de SAP SLD \(System Landscape Directory\)](#) (page 137)

Installation du moniteur de performances (PerfMon)

Le moniteur de performances (PerfMon) est un utilitaire qui illustre la surveillance des performances du système local ou distant. PerfMon démontre la procédure de surveillance des performances système à l'aide de compteurs de performances.

Pour utiliser le moniteur PerfMon pour surveiller les performances d'un système Windows :

- Votre système doit être compatible avec .NET Framework 2.0.
- Dans une invite de commande, exécutez le fichier **setup-wperfmon.bat** situé dans le répertoire **LISA_HOME\bin**.
- Sous Windows, vous devez exécuter l'invite de commande en tant qu'administrateur.

Remarque : Si vous exécutez Windows 2012, la compatibilité avec .NET Framework 2.0 peut être établie en installant .NET Framework 3.5.

De plus, vérifiez les points suivants :

- L'ID d'utilisateur est identique sur les deux ordinateurs.
- L'ID d'utilisateur dispose de droits d'administrateur sur les deux ordinateurs.
- Le partage de fichier et d'imprimante est activé.
- Le partage de fichier simple est désactivé.
- Le partage C\$ par défaut et/ou le partage ADMIN\$ sont activés.

Parfois, le pare-feu sur l'ordinateur à surveiller doit être arrêté.

Pour vérifier que la surveillance à distance fonctionne :

1. Sélectionnez Démarrer, Panneau de configuration, Outils d'administration, Performances.
2. Ajoutez un moniteur à l'ordinateur que vous voulez observer.

DevTest et Windows utilisent la même technologie pour effectuer la surveillance à distance. Si la surveillance de Windows fonctionne, en règle générale la surveillance de DevTest fonctionnera également.

Pour utiliser le moniteur PerfMon pour collecter des mesures dans DevTest, reportez-vous à la section Mesures PerfMon pour Windows de la rubrique *Utilisation de CA Application Test*.

Installation et configuration de SNMP

L'implémentation Microsoft Windows du protocole SNMP (protocole de gestion de réseau simple) est utilisée pour les tâches suivantes :

- Configuration des unités distantes
- Surveillance des performances de réseau
- Audit de l'utilisation du réseau
- Détection des pannes de réseau ou d'accès inadéquat

Prise en charge SNMP sous Windows

Windows 7 fournit un agent en mesure de répondre à des demandes SNMP et d'envoyer des interruptions.

- [Installation de l'agent Microsoft SNMP](#) (page 108)
- [Configuration de l'agent Microsoft SNMP](#) (page 109)

Prise en charge SNMP sur UNIX

La prise en charge SNMP est disponible auprès de votre fournisseur de système d'exploitation, ou vous pouvez essayer d'utiliser le package SNMP open source Net-SNMP. Pour obtenir des instructions d'installation et de configuration, consultez la documentation correspondante.

Installation de l'agent Microsoft SNMP

Installation de l'agent Microsoft SNMP sous Windows 7

Vous pouvez installer l'agent Microsoft SNMP sur un système d'exploitation Windows 7.

Procédez comme suit:

1. Dans le menu Démarrer, cliquez sur Panneau de configuration.
2. Cliquez sur Programmes.
3. Sous Programmes et fonctionnalités, sélectionnez Activer ou désactiver des fonctionnalités Windows.
La fenêtre Fonctionnalités de Windows s'ouvre.
4. Sélectionnez la case à cocher Protocole SNMP (Simple Network Management Protocol) et cliquez sur OK.
5. Patientez pendant l'installation du SNMP.

Installation de l'agent Microsoft SNMP sur des versions antérieures de Windows

Procédez comme suit:

1. Ouvrez le Panneau de configuration de Windows.
2. Double-cliquez sur l'icône Ajout/Suppression de programmes.
La fenêtre Ajout/Suppression de programmes s'affiche.
3. Cliquez sur Ajout/Suppression de programmes Windows sur le côté gauche de la fenêtre.
L'assistant Composants de Windows s'ouvre.
4. Sélectionnez Outils de gestion et d'analyse dans la liste Composants et cliquez sur Détails.
La fenêtre Outils de gestion et d'analyse s'ouvre.
5. Sélectionnez SNMP (Protocole simplifié de gestion de réseau) dans la liste Sous-composants d'outils de gestion et d'analyse et cliquez sur OK.
6. Cliquez sur Next (Suivant).
L'assistant Composants Windows installe l'agent Microsoft SNMP.
7. Une fois l'opération terminée, cliquez sur Terminer.

Configuration de l'agent Microsoft SNMP

Cette procédure décrit la configuration de l'agent SNMP.

Procédez comme suit:

1. Ouvrez le panneau de configuration de Windows.
2. Double-cliquez sur Outils d'administration.
La fenêtre Outils d'administration s'affiche.
3. Double-cliquez sur Services.
La fenêtre Services s'affiche.
4. Double-cliquez sur le service SNMP.
La boîte de dialogue SNMP Service Properties s'ouvre.
5. Définissez le type de démarrage sur **Automatique** dans l'onglet Général.
Cette action configure le service SNMP pour démarrer l'agent SNMP Microsoft lors du démarrage du système.
6. Cliquez sur l'onglet Interruptions.
7. Saisissez le nom de la communauté à laquelle votre ordinateur envoie des messages d'interruption dans le champ Nom de la communauté.
8. Cliquez sur Ajouter à la liste.
9. Cliquez sur Appliquer, puis sur OK.
10. Cliquez sur OK.

Pour utiliser Windows SNMP pour collecter des mesures, reportez-vous à la rubrique *Utilisation de CA Application Test*.

Exécution de TCPMon

TCPMon est un utilitaire qui permet de surveiller les messages transmis dans une conversation TCP.

L'utilitaire TCPMon comprend les éléments suivants :

- Windows : un fichier .jar, un fichier .bat
- UNIX : un script Shell

Pour exécuter TCPMon :

Double-cliquez sur le fichier .bat sur Windows ou exéutez le script Shell sur UNIX.

Un fichier **tcpmon.bat** est disponible dans le répertoire **LISA_HOME\bin**. Vous pouvez obtenir la dernière version de l'utilitaire TCPMon à l'adresse suivante :

<http://ws.apache.org/commons/tcpmon/>.

Remarque : Cette section décrit la version TCPMon d'Apache. Cette version de TCPMon contient l'onglet **Sender** qui n'est pas disponible dans la version de TCPMon fournie avec DevTest.

Informations complémentaires :

[Utilisation de TCPMon comme intermédiaire explicite](#) (page 111)

[Utilisation de TCPMon comme expéditeur de demande](#) (page 111)

Utilisation de TCPMon comme intermédiaire explicite

L'utilitaire TCPMon est plus communément utilisé comme intermédiaire. Cette utilisation est appelée explicite, car le client doit pointer vers l'intermédiaire, au lieu du terminal d'origine, pour surveiller les messages.



Pour démarrer l'utilitaire TCPMon dans cette configuration :

1. Dans l'onglet Admin, indiquez le port d'écoute, le nom d'hôte et le port pour le processus d'écoute pour TCPMonitor.
2. Pour ouvrir un nouvel onglet (Port 8000) permettant d'afficher les messages, cliquez sur Add.

Les demandes pointent désormais vers le port d'écoute du TCPMon au lieu du terminal d'origine.

Tous les messages transmis vers localhost:8080 et à partir de celui-ci sont surveillés.

- Nous définissons l'écouteur sur le port 8000 qui peut être un port inutilisé dans l'ordinateur local.
- Nous avons ajouté un processus d'écoute avec l'hôte **localhost** et le port 8080.
- Pointez le navigateur vers **localhost:8000** au lieu de localhost:8080.

Utilisation de TCPMon comme expéditeur de demande

Vous pouvez également utiliser TCPMon comme expéditeur de demande pour des services Web.

- Vous pouvez coller le message SOAP de demande dans la fenêtre Sender (Expéditeur), puis l'envoyer directement au serveur.
- Le terminal de service Web est entré dans la zone de texte connection Terminal.

Installation du module d'extension HP ALM - Quality Center

Le module d'extension HP ALM - Quality Center permet de charger et d'exécuter un scénario de test DevTest comme test Quality Center à partir d'une suite HP ALM - Quality Center. Vous pouvez importer et exécuter des tests DevTest à partir de Quality Center. Cette intégration vous permet de profiter de toutes les fonctionnalités de Quality Center associées à la puissance de test de DevTest. Le chargement d'un scénario de test DevTest dans Quality Center vous permet d'exécuter des tests DevTest en temps réel. Vous obtenez également une capture complète des résultats de test et des rappels DevTest renvoyés par les systèmes testés. Les tests DevTest peuvent être exécutés dans le flux de travaux de Quality Center et renvoyés les résultats afin de conserver le contexte et le statut du processus de test.

Les logiciels suivants doivent être installés :

- DevTest 7.0 ou version ultérieure
- Module d'extension HP ALM - Quality Center pour DevTest
- HP Quality Center 10 ou 11
- Composants HP Quality Center côté client
- .NET 2.0 Go Runtime

Lorsque vous installez le module d'extension sur un ordinateur sur lequel DevTest est installé, le fichier **LisaQCRunner.exe** est ajouté au répertoire **LISA_HOME\bin**. **LisaQCRunner.exe** est le serveur COM enregistré pendant le processus d'installation. DevTest est appelé à l'aide de l'interface Quality Center VAPI-XP. Le script VAPI-XP crée une instance de notre objet COM. Cet objet simule à son tour le test à exécuter et écoute les résultats. Enfin, l'objet COM prend les résultats et met à jour l'instance Quality Center avec les résultats de DevTest.

Les fichiers suivants sont également installés :

- Script_Template_js.txt
- Script_Template_vbs.txt

Ces fichiers contiennent des scripts utilisés lors de la configuration des tests DevTest dans Quality Center.

Remarque : Pour plus d'informations sur l'utilisation du module d'extension HP ALM - Quality Center, reportez-vous à la section *Utilisation de CA Application Test*.

Procédez comme suit:

1. Vérifiez que .NET Runtime est installé sur l'ordinateur client.
2. Installez les composants HP Quality Center côté client
3. Installez DevTest.
4. Accédez au répertoire **LISA_HOME\addons\qc** et exécutez le fichier **td_plugin.exe**.

5. Suivez la procédure de l'assistant.

Installation d'IBM Rational Quality Manager

IBM Rational Quality Manager est un environnement Web de gestion de tests centralisée pour la planification de tests, le contrôle de flux de travaux, le suivi et la génération de rapport sur les mesures. IBM Rational Quality Manager est extensible de différentes façons, notamment la possibilité d'utiliser des connecteurs ou des modules d'extension pour servir de passerelle entre des systèmes RQM et des systèmes externes. Le module d'extension RQM de DevTest permet de réutiliser ou de créer des scénarios de test DevTest et de les lier dans RQM. Les scénarios de test DevTest peuvent être ensuite exécutés à partir de l'interface de RQM. Les résultats des exécutions de test sont cumulés dans l'historique d'exécution et de génération de rapports sur les tests RQM.

Cette section comprend les rubriques suivantes :

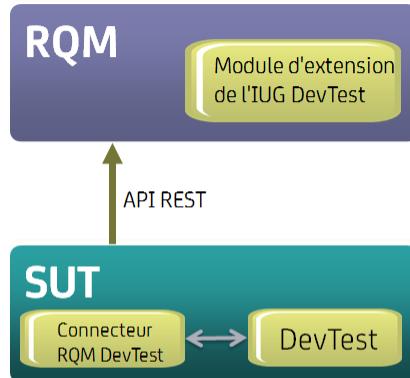
- [Implémentation](#) (page 114)
- [Installation de l'interface utilisateur de l'adaptateur DevTest](#) (page 115)
- [Exécution de l'outil de ligne de commande Adapter](#) (page 116)
- [Flux de travaux d'utilisation générale](#) (page 116)

Implémentation

La solution RQM de DevTest complète est implémentée sous forme de deux composants :

- Le connecteur RQM de DevTest
- L'extension d'interface utilisateur Web

L'interface utilisateur permet d'étendre l'interface utilisateur prédéfinie. Cette interface utilisateur permet également de transmettre les paramètres nécessaires et de les afficher entre DevTest et RQM. Le connecteur RQM pour DevTest est utilisé pour répondre aux tâches d'opération planifiées par RQM. Les informations sur les tâches collectées par l'interface utilisateur sont transférées au connecteur, puis celui-ci appelle l'utilitaire TestRunner avec les paramètres nécessaires.



Le connecteur utilise le commutateur `-html` de TestRunner pour générer la sortie HTML chargée ultérieurement vers RQM. A l'issue du test, les résultats sont chargés vers RQM et associés à l'exécution du test. L'utilisateur peut ensuite examiner les résultats dans l'interface RQM.

Installation de l'interface utilisateur de l'adaptateur

Procédez comme suit:

1. Accédez au répertoire **LISA_HOME\addons** et recherchez le fichier ZIP nommé **rqm-adapter.zip**.
2. Décompressez le fichier et recherchez le fichier **com.itko.lisa.integration.ibm.rqm.update.site.zip** dans le dossier **lisa-adapter-ui**.
3. Sur le serveur RQM, décompressez le fichier **com.itko.lisa.integration.ibm.rqm.update.site.zip**.
4. Ouvrez le fichier **com.itko.lisa.integration.ibm.rqm.adapter.web.update.ini** et mettez à jour la référence de sorte à pointer vers l'emplacement du site de mise à jour.
5. Copiez le fichier .ini modifié à l'emplacement suivant : **<Installation racine de RQM>\server\server\conf\jazz\provision_profiles**.
6. Mettez à jour la configuration de serveur à l'aide du service de réinitialisation du serveur Rational Quality Manager. L'utilitaire réinitialisé est disponible à l'URL suivante :
`[https://<nom-hôte>:<numéro-port>/jazz/admin?internal#action=com.ibm.team.repository.admin.serverReset|https://%3chostname%3e%3cportnumber%3e]`
Le service vous invite à vous connecter à la console d'administration d'IBM Rational Quality Manager. L'ID d'utilisateur doit disposer de droits d'administration.
7. Arrêtez le serveur RQM.
8. Redémarrez le serveur Rational Quality Manager.

Exécution de la ligne de commande Adapter

Procédez comme suit:

1. Accédez au répertoire **LISA_HOME\addons** et recherchez le fichier ZIP nommé **rqm-adapter.zip**.
2. Décompressez le fichier et recherchez le dossier enfant **lisa-adapter2.0** dans le dossier.
3. Dans une ligne de commande, démarrez l'adaptateur. Les paramètres suivants sont requis :
 - **-repository**
[`https://rqmserver:port/jazz|https://%3crqmserver%3e%3cport%3e`]
 - [`https://%3crqmserver%3e%3cport%3e`]-user *userid*
 - **-password** *password*
 - **-adapter** *adapter*(com.itko.lisa.integration.ibm.rqm)
 - **-adapterName** *adapterName* (LISA RQM Adapter)## -LISA_HOME *lisaHomePath* (d\:/lisa)
 - [-projectArea *project area* (Quality%20Manager)]
 - [-sleepTime *sleep time(5)*]

Flux de travaux d'utilisation générale

Créez un script de test dans RQM.

1. Associez un scénario de test DevTest, un document de simulation et un document de configuration facultatif avec le script de test.
2. Associez un ou plusieurs scripts de test à un scénario de test ; le scénario de test est alors prêt pour l'exécution.
3. Sélectionnez l'exécution du scénario de test, sélectionnez un adaptateur en cours d'exécution et cliquez sur OK.
4. Le serveur contacte le système testé via le module d'extension RQM Connector, qui appelle à son tour DevTest.
5. Patientez pendant l'exécution du test, puis vérifiez les résultats dans RQM.

Configuration de l'intégration à CA APM

DevTest Solutions est disponible en deux versions, station de travail et serveur. La version de station de travail est conçue pour créer et vérifier des scénarios de test DevTest et des simulations virtuelles de composants serveur. La station de travail est exécutée dans un processus Java unique (DevTest Workstation) et la taille des tests et des simulations est par conséquent limitée. La version serveur divise plusieurs fonctions en un ou plusieurs processus Java distincts que vous pouvez exécuter sur plusieurs systèmes, si nécessaire.

Vous pouvez instrumer tous les processus Java de DevTest avec des agents Introscope de CA Application Performance Management (APM). Cette instrumentation permet de signaler des mesures génériques qui décrivent l'état du processus au gestionnaire d'entreprise CA Wily Introscope. Ces mesures incluent :

- Utilisation de la mémoire et de l'UC
- Opérations de nettoyage de la mémoire
- Utilisation de sockets et de bases de données secondaires

De plus, l'outil de suivi TestEvent pour DevTest peut instrumer le coordinateur DevTest pour générer une série de mesures décrivant l'exécution de scénarios de test DevTest. Cette instrumentation est disponible indépendamment du fait que le coordinateur est exécuté comme processus distinct ou dans DevTest Workstation. DevTest Workstation est un processus interactif avec une interface utilisateur Windows. Hormis DevTest Workstation, vous pouvez exécuter tous les autres processus DevTest comme processus Windows standard ou comme services Windows. Vous pouvez instrumer les deux types de processus à l'aide d'agents Introscope.

Cette section comprend les rubriques suivantes :

- [Instrumentation des processus DevTest](#) (page 118)
- [Fichiers de l'agent Introscope](#) (page 120)
- [Mesures signalées par les outils de suivi](#) (page 121)
- [Dérivation de mesures DevTest](#) (page 123)
- [Configuration de l'outil de suivi](#) (page 124)
- [Installation de l'agent Introscope](#) (page 126)
- [Configuration de la journalisation de l'outil de suivi](#) (page 127)
- [Mesures standard signalées](#) (page 127)
- [Affichage des mesures](#) (page 129)
- [Reporting \(Génération de rapports\)](#) (page 137)

Instrumentation des processus DevTest

Le répertoire **LISA_HOME\bin** contient un fichier exécutable Windows pour chaque processus DevTest. Ce dossier inclut des versions distinctes pour les processus que vous pouvez exécuter comme processus Windows standard et comme services Windows. La liste des processus et leurs noms de fichier exécutable se présente comme suit :

Nom de processus DevTest	Type de processus	Nom d'exécutable
Coordinateur	Standard	CoordinatorServer.exe
Coordinateur	Service	CoordinatorService.exe
Enterprise Dashboard (Tableau de bord d'entreprise)	Standard	EnterpriseDashboard.exe
Enterprise Dashboard (Tableau de bord d'entreprise)	Service	EnterpriseDashboardService.exe
Portail	Standard	Portal.exe
Portail	Service	PortalService.exe
Registre	Standard	Registry.exe
Registre	Service	RegistryService.exe
Simulator (Simulateur)	Standard	Simulator.exe
Simulator (Simulateur)	Service	SimulatorService.exe
Virtual Service Environment (Environnement de service virtuel)	Standard	VirtualServiceEnvironment.exe
Virtual Service Environment (Environnement de service virtuel)	Service	VirtualServiceEnvironmentService.exe
Station de travail	Standard	Workstation.exe

Remarque : Cette liste exclut les fichiers exécutables dans le répertoire bin non pertinents pour l'instrumentation avec Introscope.

Pour instrumenter un processus DevTest avec un agent Introscope, créez un fichier portant le même nom que le fichier exécutable et l'extension **.vmoptions**. Par exemple, pour instrumenter la station de travail DevTest Workstation, créez un fichier **Workstation.vmoptions** dans le répertoire **bin** sous le répertoire d'installation du DevTest Server. Le fichier .vmoptions contient les deux lignes suivantes :

```
-javaagent:<AGENT_HOME>/Agent.jar  
-Dcom.wily.introscope.agentProfile=<AGENT_HOME>/core/config/IntroscopeAgent.profile
```

<**AGENT_HOME**> est le chemin d'accès à une installation d'agent Introscope propre à DevTest. Généralement, il s'agit d'un chemin absolu, mais il peut également s'agir d'un chemin relatif vers le répertoire actuel dans lequel le processus DevTest est exécuté.

Pour que l'agent Introscope génère un rapport sous un nom autre que celui par défaut DevTest Agent, ajoutez une ligne définissant la propriété système **Java com.wily.introscope.agent.agentName**. Par exemple, pour appeler l'agent DevTest Workstation 6.0.5.87, ajoutez la ligne suivante :

```
-Dcom.wily.introscope.agent.agentName=DevTest Workstation  
6.0.5.87 Agent
```

Ajoutez des lignes supplémentaires pour définir d'autres options de ligne de commande de la machine virtuelle Java ou des propriétés système requises. Chaque option doit être spécifiée dans une ligne distincte.

L'installation de l'agent Introscope que vous utilisez peut être spécifique à l'installation de DevTest, ou même au processus DevTest. Vous pouvez créer diverses configurations d'agent en variant le contenu de **IntroscopeAgent.profile** dans ces différentes installations d'agent. Vous pouvez également modifier la valeur de la propriété système **com.wily.introscope.agentProfile** définie dans chaque fichier **.vmoptions** pour créer plusieurs profils d'agent dans une seule installation d'agent. Cette approche permet d'utiliser différents niveaux de suivi ou de journalisation (ou même différents Introscope Enterprise Manager) pour surveiller différents processus DevTest.

Par exemple, un site peut inclure une utilisation de production de DevTest qui utilise un noeud de coordinateur exécuté comme service Windows. Ce site peut également inclure une utilisation de test/expérimentale qui utilise un coordinateur exécuté comme processus Windows standard. Ce coordinateur peut également être celui créé dans DevTest Workstation. Les mesures de production sont signalées à la production de gestionnaire d'entreprise, tandis que les mesures de test/expérimentales sont signalées au gestionnaire d'entreprise de test. Vous pouvez pour cela utiliser plusieurs profils et nommer le profil correct dans le fichier .vmoptions approprié. Vous pouvez également remplacer la valeur des propriétés **introscope.agent.enterprisemanager.connectionorder** et **introscope.agent.enterprisemanager.transport.*** définies dans le profil en définissant des propriétés système Java de ces noms dans le fichier .vmoptions. Pour obtenir la plage complète d'options de configuration de l'agent Introscope, consultez le manuel CA Wily Introscope Java Agent Guide.

Fichiers de l'agent Introscope

Les fichiers de l'agent Introscope propres à DevTest sont indiqués dans la table suivante. Pour plus de détails sur le contenu de ces fichiers et sur leur modification pour personnaliser l'instrumentation de DevTest, consultez la section [Configuration de l'outil de suivi](#) (page 124).

File (Fichier)	Emplacement relatif à <AGENT_HOME>	Contenu
Lisa.jar	core\ext	Outils de suivi propres à DevTest, formateurs de nom
IntroscopeAgent.profile	core\config	Profil d'agent spécifique à DevTest. Par défaut, référence lisa-typical.pbl pour définir un ensemble d'outils de suivi et instrumenter des processus DevTest.
lisa.pbd	core\config	Définition de points d'outils de suivi et de l'instrumentation
lisa-full.pbd	core\config	Liste de fichiers PBD à utiliser pour l'instrumentation complète de DevTest Référence lisa-toggles-full.pbd .
lisa-toggles-full.pbd	core\config	Liste de basculements utilisés pour activer des fonctionnalités d'instrumentation spécifiques lors de l'instrumentation en mode complet
lisa-typical.pbl	core\config	Liste de fichiers PBD à utiliser pour l'instrumentation classique de DevTest Référence lisa-toggles-typical.pbd .
lisa-toggles-typical.pbd	core\config	Liste de basculements utilisés pour activer des fonctionnalités d'instrumentation spécifiques lors de l'instrumentation en mode standard

Mesures signalées par les outils de suivi

Les outils de suivi de DevTest génèrent des mesures sur un ou plusieurs des quatre niveaux suivants :

- DevTest
- Scénario de test
- Simulator (Simulateur)
- Etape de test

Les mesures créées aux niveaux du simulateur et de l'étape de test sont également cumulées à un niveau maximum configurable par l'utilisateur (scénario de test, par défaut). Si un scénario de test est exécuté sur deux simulateurs et contient deux étapes de test, la mesure Responses Per Interval (Réponses par intervalle) est générée pour chaque étape de test sous chaque simulateur. Puis (selon la configuration), les valeurs sont cumulées pour générer les mesures Responses Per Interval (Réponses par intervalle) pour chaque simulateur. Pour finir, en fonction de la configuration, les valeurs sont à nouveau cumulées pour générer la mesure Responses Per Interval (Réponses par intervalle) pour le scénario de test. Le niveau le plus élevé auquel une mesure peut être cumulée est celui du scénario de test. Ainsi, seules les mesures générées au niveau de DevTest sont répertoriées dans la table suivante avec un niveau de DevTest.

Les données issues de tous les événements de type pertinent dans chaque intervalle de 15 secondes de génération de rapports de l'agent sont combinées comme indiqué.

Metric (Mesure)	Niveau	Source
Average Response Time (ms) (Temps moyen de réponse (en ms))	Etape de test	Moyenne de la valeur Long Description (Description longue) (convertie en nombre entier) à partir de tous les événements Step response time (Temps de réponse d'étape) (ID 18)
Responses Per Interval (Nombre de réponses par intervalle)	Etape de test	Nombre total d'événements Step response time (Temps de réponse d'étape) (ID 18)
Errors Per Interval (Nombre d'erreurs par intervalle)	Etape de test	Nombre total d'événements Step error (Erreur d'étape) (ID 20)
Failures Per Interval (Nombre d'échecs par intervalle)	Etape de test	Nombre total d'événements Abort (Interrompre) (ID 50) et Cycle failed (Echec du cycle) (ID 13)

Tests Running (Tests en cours d'exécution)	Scénario de test	Nombre de scénarios de test exécutés à la fin de l'intervalle de génération de rapports de l'agent. Le nombre augmente pour chaque événement Test started (Test démarré) (ID 4) et diminue pour chaque événement Test ended (Test arrêté) (ID 5).
Virtual Users Running (Utilisateurs virtuels en cours d'exécution)	Simulator (Simulateur)	Nombre d'utilisateurs virtuels exécutés à la fin de l'intervalle de génération de rapports de l'agent. Le nombre augmente pour chaque événement Cycle started (Cycle démarré) (ID 11) et diminue pour chaque événement Cycle ended (Cycle arrêté) (ID 24).
Test Runner Errors Per Interval (Nombre d'erreurs de l'exécuteur de tests par intervalle)	DevTest	Nombre total d'événements Cycle runtime error (Erreur au niveau de l'exécution du cycle) (ID 25)
Staging Errors Per Interval (Nombre d'erreurs de simulation par intervalle)	DevTest	Nombre total d'événements Model definition error (Erreur au niveau de la définition de modèle) (ID 23)

Les mesures Errors Per Interval (Nombre d'erreurs par intervalle) d'une étape de test, d'un simulateur, ou d'un scénario de test sont signalées lors de la première détection d'un événement d'erreur uniquement. Cela est également valable pour la mesure Failures Per Interval (Nombre d'échecs par intervalle).

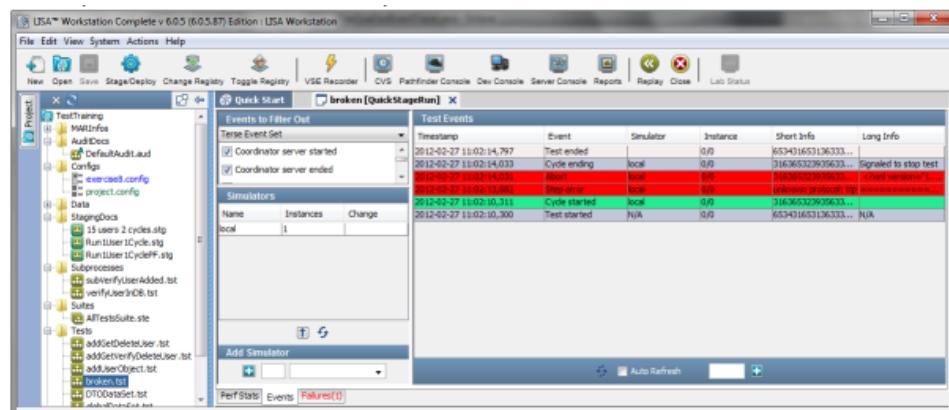
Si seuls quelques courts scénarios de test sont exécutés, les mesures Test running (Tests en cours d'exécution) et Virtual Users Running (Utilisateurs virtuels en cours d'exécution) peuvent continuer de signaler 0 pour chaque cycle de génération de rapports d'agent. Cela se produit lorsque des événements de démarrage et d'arrêt surviennent dans la même période de 15 secondes et que le nombre à l'issue de la période est 0.

Lorsque les mesures sont cumulées, vous pouvez créer la mesure de niveau inférieur dans une période de génération de rapports d'agent. Vous pouvez créer les mesures cumulées dans la période suivante. Il peut parfois exister des points dans lesquels la valeur des mesures de niveau inférieur et des mesures cumulées dans une période ne coïncident pas. Cette situation est inévitable.

Dérivation de mesures DevTest

L'outil de suivi d'événements de test DevTest est un outil de suivi de méthode. En d'autres termes, cet outil contient un code inséré immédiatement avant et après un appel de méthode spécifique. L'outil de suivi d'événements de test est conçu pour instrumenter la méthode du constructeur de la classe BaseCoordinator. A l'issue de l'opération du constructeur, l'outil de suivi crée un écouteur d'événements de test et enregistre cet écouteur dans le coordinateur. L'écouteur reçoit ensuite chaque objet TestEvent (Événements de test) des types affichés préalablement. L'écouteur extrait des informations pertinentes de ces objets (Événements de test) et les envoie au gestionnaire d'entreprise Introscope sous forme de valeurs de mesures Introscope.

La capture d'écran de DevTest Workstation suivante illustre les événements de test générés par une exécution de scénario de test nommé broken (interrompu), qui contient une erreur. Pour connaître les mesures Introscope créées, vérifiez chaque valeur dans la colonne Event (Événement), dans la section [Mesures signalées par les outils de suivi](#) (page 121).



Configuration de l'outil de suivi

Vous pouvez désactiver l'ensemble du suivi de DevTest en supprimant toutes les références aux fichiers .pbl propres à DevTest (**lisa-full.pbl** ou **lisa-typical.pbl**) dans le profil de l'agent (IntroscopeAgent.profile par défaut). Vous pouvez également supprimer le fichier **Lisa.jar**. Toutefois, si des références aux outils de suivi demeurent dans **lisa.pbd**, le démarrage de l'agent échouera. Cet échec est dû d'une tentative d'utilisation d'un outil de suivi introuvable dans les extensions d'agent.

Vous pouvez désactiver l'outil de suivi d'événements de test en commentant la directive TurnOn:LisaTestEventTracing dans le fichier de basculement de DevTest actif (**lisa-toggles-full.pbd** ou **lisa-toggles-typical.pbd**).

Vous pouvez utiliser de nombreux autres paramètres pour limiter le nombre de mesures signalées. Ces paramètres sont définis dans le fichier **lisa.pbd**. Vous pouvez contrôler les paramètres suivants :

- **minMetricLevel** : paramètre qui contrôle le niveau le plus bas de signalement des mesures. Le paramètre peut contenir l'une des valeurs suivantes :
 - Lisa
 - TestCase (Scénario de test ; valeur par défaut)
 - Simulator (Simulateur)
 - TestStep (Etape de test)
- **maxRollupLevel** : paramètre qui contrôle le niveau le plus élevé auquel des mesures peuvent être cumulées. Le paramètre peut contenir l'une des valeurs suivantes :
 - TestCase (Scénario de test ; valeur par défaut)
 - Simulator (Simulateur)
 - TestStep (Etape de test)

Les mesures générées sous le paramètre actuel **minMetricLevel** sont cumulées jusqu'à atteindre **minMetricLevel**. Ces mesures sont alors signalées à partir de ce niveau jusqu'au niveau du paramètre **maxRollupLevel**, sauf si **minMetricLevel** est supérieur à **maxRollupLevel**. Si **minMetricLevel** est supérieur à **maxRollupLevel**, les mesures ne seront pas signalées. Toute mesure générée à un niveau supérieur à **maxRollupLevel** est signalée au niveau généré uniquement, non au niveau cumulé.

De plus, six autres paramètres permettent à l'outil de suivi d'événements de test de limiter les mesures à des combinaisons sélectionnées de scénarios de test, simulateurs, ou étapes de test. Vous pouvez configurer l'outil de suivi pour inclure ou exclure en fonction du nom du scénario de test, du simulateur, ou de l'étape de test. Vous pouvez établir des correspondances du nom au niveau d'expressions régulières ou d'une paire d'expressions régulières. Tous les noms qui correspondent à l'expression régulière d'inclusion et non à l'expression régulière d'exclusion seront sélectionnés pour signaler des mesures. Si l'expression régulière d'inclusion n'est pas définie, tous les noms seront inclus. Si l'expression régulière d'exclusion n'est pas définie, aucun nom ne sera exclus.

Par défaut, tous les scénarios de test, simulateurs et étapes de test possibles sont inclus. Seules les étapes de test internes appelées abort (interrompre) et end (finaliser) sont exclues. Si l'expression régulière contient certains caractères spéciaux (par exemple : symbole | utilisé pour spécifier des modèles alternatifs), l'expression régulière complète doit être placée entre guillemets doubles. Sinon, les guillemets sont facultatifs. L'exemple suivant illustre les modèles d'inclusion et d'exclusion par défaut définis dans **lisa.pbd**.

```
SetTracerParameter: LisaCoordinatorTracer includeTestCasesRegExp
"
SetTracerParameter: LisaCoordinatorTracer excludeTestCasesRegExp
"
SetTracerParameter: LisaCoordinatorTracer
includeSimulatorsRegExp ""
SetTracerParameter: LisaCoordinatorTracer
excludeSimulatorsRegExp ""
SetTracerParameter: LisaCoordinatorTracer includeTestStepsRegExp
"
SetTracerParameter: LisaCoordinatorTracer excludeTestStepsRegExp
"abort|end"
```

Vous pouvez configurer le chemin de mesure sous lequel les mesures seront signalées. La partie de niveau DevTest du chemin de mesures est la dernière valeur spécifiée dans la directive **TraceOneMethodWithParametersIfFlagged** pour la méthode < init>. Elle est définie par défaut sur DevTest. Dans le cas des autres niveaux, les paramètres suivants définissent les parties du chemin de mesures pour les niveaux de scénario de test, de simulateur et d'étape de test :

- **pathComponentForTestCase**
- **pathComponentForSimulator**
- **pathComponentForTestStep**

Evitez d'utiliser les caractères suivants dans les valeurs de ces paramètres :

- Le signe deux-points (:) n'est pas valide pour les chemins de mesures.
- Les chemins de mesures ne doivent pas commencer ni finir par le séparateur de noeud du chemin de mesure (|).

- Les chemins de mesures ne doivent pas contenir deux séparateurs de noeud du chemin de mesures adjacents (|).

Un trait de soulignement (_) remplace chaque occurrence de deux-points ou de barre verticale. Les espaces de début et de fin sont supprimés de ces noms. La valeur Unknown (Inconnu) remplace un nom par une valeur nulle ou une chaîne vide.

Les valeurs par défaut de ces paramètres sont appliquées si le paramètre est commenté ou est une chaîne vide. Chaque valeur doit contenir l'espace réservé pertinent pour indiquer l'endroit au niveau duquel le nom du niveau pertinent est inséré dans le chemin de mesures. Voici un exemple de valeurs par défaut définies dans **lisa.pbd** :

```
SetTracerParameter: LisaCoordinatorTracer
pathComponentForTestCase "Test Case|{TestCase}"
SetTracerParameter: LisaCoordinatorTracer
pathComponentForSimulator "Simulator|{Simulator}"
SetTracerParameter: LisaCoordinatorTracer
pathComponentForTestStep "Step|{TestStep}"
```

Installation de l'agent Introscope

Pour installer l'agent Introscope pour DevTest, extrayez le contenu du fichier de distribution **CALISAIIntegrationNoInstaller9.1.1.0.zip** (Windows) ou **CALISAIIntegrationNoInstaller9.1.1.0.tar** (UNIX/Linux).

Pour éviter des conflits de nom de fichier de profil, extrayez ce fichier vers un répertoire qui ne contient aucune autre distribution d'agent Introscope Java. Puis, créez un ou plusieurs fichiers .vmoptions pour instrumenter l'ensemble de processus DevTest. Les seuls processus DevTest qui signalent des mesures propres à DevTest sont DevTest Workstation et le coordinateur. Vous pouvez instrumenter d'autres processus. Toutefois, les autres processus signalent un ensemble minimal de mesures. Ces mesures sont généralement limitées à l'utilisation de l'UC, l'utilisation de la mémoire, l'agent et l'identité de la machine virtuelle Java.

Configuration de la journalisation de l'outil de suivi

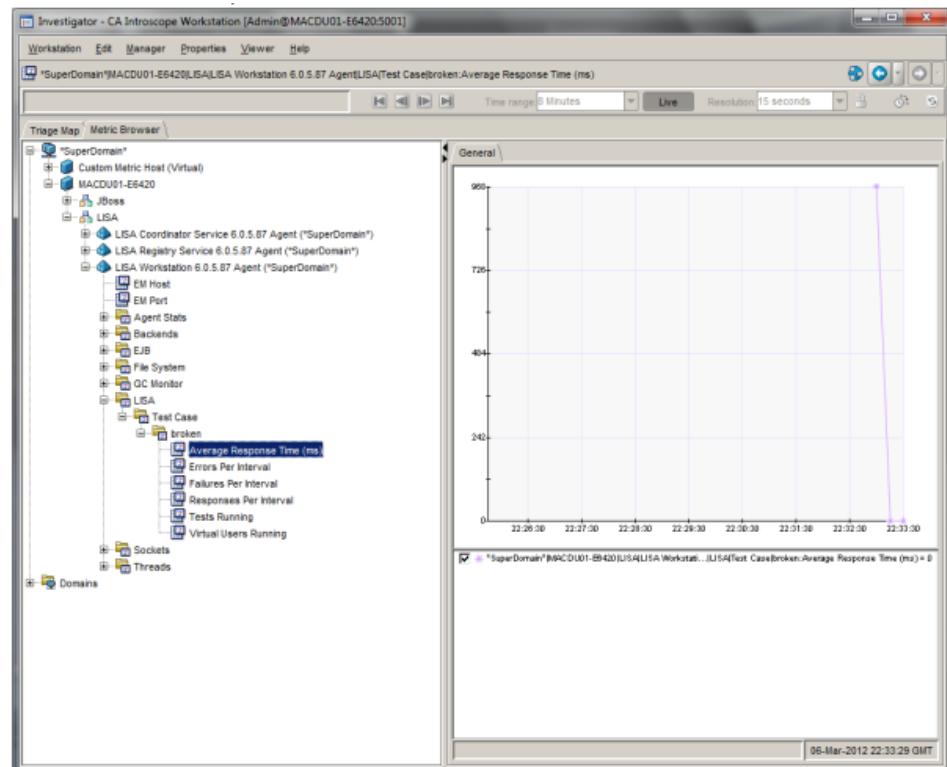
Pour écrire d'autres informations sur l'opération de l'outil de suivi DevTest dans le fichier journal de l'agent, ajoutez les lignes suivantes au profil de l'agent :

```
log4j.additivity.IntroscopeAgent.LisaCoordinatorTracer=false
log4j.logger.IntroscopeAgent.LisaCoordinatorTracer=DEBUG, logfile
```

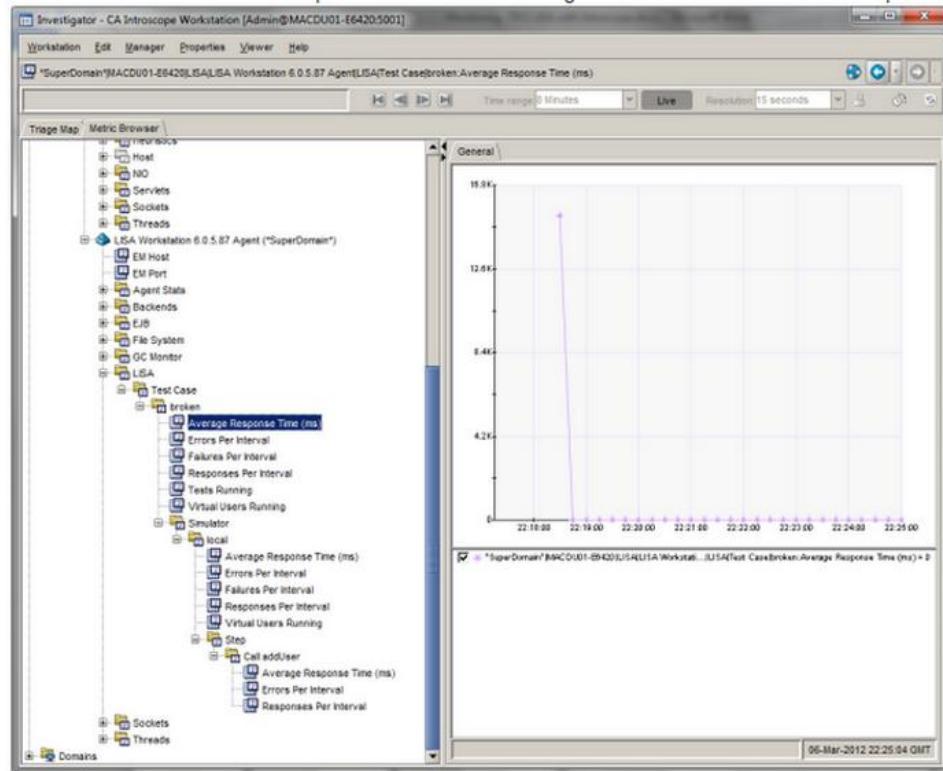
Si le niveau de journalisation de l'agent est défini sur DEBUG (Débogage), ces lignes ne sont pas requises. Toutefois, la journalisation à ce niveau écrit une quantité d'informations importante dans le journal de l'agent. Cette quantité d'informations peut compliquer la recherche des lignes à partir de l'outil de suivi DevTest.

Mesures standard signalées

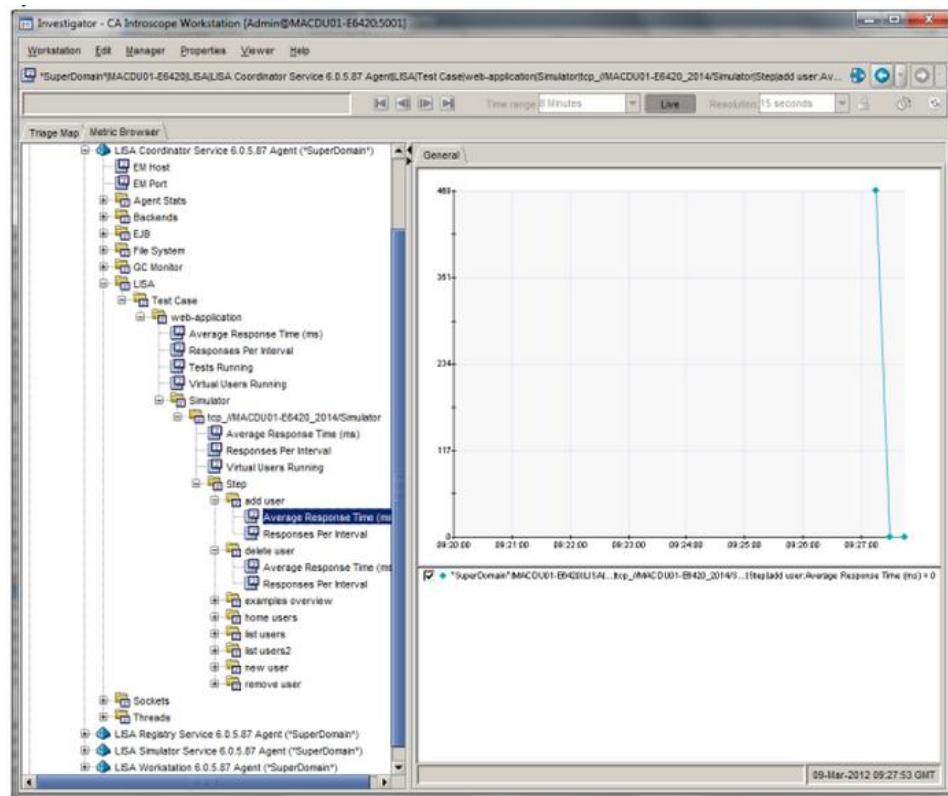
Le graphique suivant illustre des ensembles standard de mesures signalées par l'outil de suivi d'événements de test. Dans l'exemple suivant, le scénario de test simple nommé broken (rompu) a été exécuté. Ce scénario de test contient uniquement une étape de test qui contient une erreur de configuration. Cette erreur entraîne l'échec du scénario de test. Cet échec entraîne le signalement des mesures Errors Per Interval (Nombre d'erreurs par intervalle) et Failures Per Interval (Nombre d'échec par intervalle) comme indiqué.



Le graphique suivant illustre le même scénario de test exécuté après le remplacement de la valeur par défaut TestCase du paramètre **minMetricLevel** par TestStep. Les mesures sont générées pour les niveaux de simulateur et d'étape de test.



Le graphique suivant illustre un scénario de test plus complexe qui contient huit étapes de test et ne génère aucune erreur d'exécution. De plus, le test a été simulé à l'aide de processus de coordinateur et de simulateur distincts au lieu des processus créés dans la station de travail DevTest Workstation. Cette fenêtre indique les noeuds d'étape de test supplémentaires, mais aucune mesure Errors Per Interval (Nombre d'erreurs par intervalle) et Failures Per Interval (Nombre d'échecs par intervalle). Les mesures sont signalées sous un noeud Coordinator Service Agent (Agent du service de coordinateur), dont le nom a été personnalisé pour inclure le numéro de version de DevTest. Notez également qu'en raison de la non-validité des deux-points dans les chemins d'accès aux mesures, ceux présents dans le nom du simulateur ont été remplacés par des traits de soulignement.



Affichage des mesures

Le gestionnaire d'entreprise Introscope est fourni avec un module de gestion DevTest qui contient des groupements de mesures, des alertes, des tableaux de bord et des rapports utiles pour la surveillance de l'environnement DevTest. Deux tableaux de bord sont fournis :

- Tableau de bord [Overview](#) (page 130) (Présentation)
- [Tableau de bord System Under Test Overview \(Présentation du système testé\)](#) (page 136)

Tableau de bord Overview (Présentation)

Le tableau de bord Overview (Présentation) offre une vue d'ensemble de tous les processus instrumentés et des tests exécutés dans l'installation de DevTest.

La section Tests du tableau de bord de présentation inclut quatre icônes d'alerte simples :

Failures (Echecs)

Définie sur caution (attention) si un test a signalé un échec dans le dernier intervalle de génération de rapports. Définie sur danger si un test a signalé plusieurs échecs dans le dernier intervalle de génération de rapports.

Errors (Erreurs)

Définie sur caution (attention) si un test a signalé une erreur dans le dernier intervalle de génération de rapports. Définie sur danger si un test a signalé plusieurs erreurs dans le dernier intervalle de génération de rapports.

Runner Errors (Erreurs de l'exécuteur)

Définie sur caution (attention) si la mesure Test Runner Errors Per Interval (Nombre d'erreurs de l'exécuteur de tests par intervalle) a une valeur de 1 dans un intervalle de génération de rapports. Définie sur danger si la valeur est supérieure ou égale à 2 dans un intervalle de génération de rapports.

Staging Errors (Erreurs de stockage intermédiaire)

Définie sur caution (attention) si la mesure Staging Errors Per Interval (Nombre d'erreurs de simulation par intervalle) a une valeur de 1 pour un intervalle de génération de rapports. Définie sur danger si la valeur est supérieure ou égale à 2 dans un intervalle de génération de rapports.

Cette section inclut également les éléments suivants :

Alerte Overall (globale)

Alerte récapitulative définie sur caution (attention) si l'une des quatre alertes précédentes est définie sur caution (attention). Cette alerte est définie sur danger si l'une des quatre alertes précédentes est définie sur danger.

Graphique Test Average Response Time (ms) (Temps moyen de réponse lors du test, en millisecondes)

Affiche jusqu'à dix graphiques de mesures Average Response Time (ms) (Temps moyen de réponse, en millisecondes) pour un scénario de test, un simulateur et des étapes de test. Dans les dix graphiques sélectionnés, les dix premières valeurs de mesures dans la période sont affichées.

Graphique Test Responses Per Interval (Test de nombre de réponses par intervalle)

Affiche jusqu'à dix graphiques de mesures Responses Per Interval (Nombre de réponses par intervalle) pour un scénario de test, un simulateur et des étapes de test. Dans les dix graphiques sélectionnés, les dix premières valeurs de mesures dans la période sont affichées.

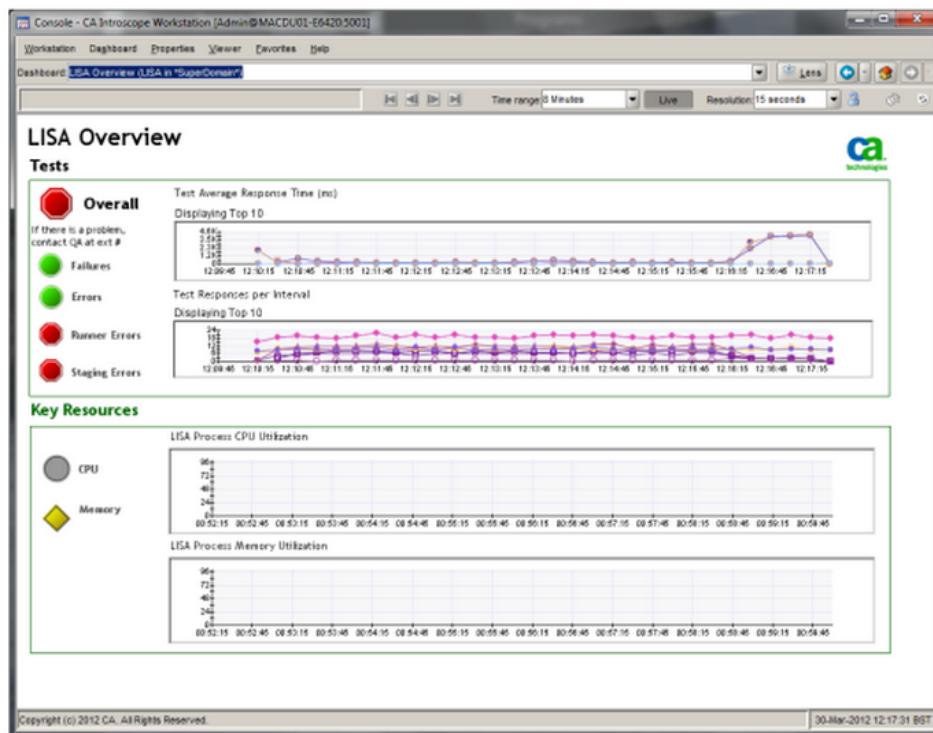
Graphique LISA Process CPU Utilization (Utilisation de l'UC par les processus LISA)

Affiche un graphique des mesures **CPU:Utilization Percent** (process) (UC : pourcentage d'utilisation, processus) pour tous les processus instrumentés.

Graphique LISA Process Memory Utilization (Utilisation de la mémoire par les processus LISA)

Affiche un graphique des mesures **GC Heap:Bytes In Use** (Segment de mémoire GC : octets utilisés) pour tous les processus instrumentés.

Le graphique suivant illustre le tableau de bord de présentation LISA.



Une autre version du tableau de bord de présentation remplace les graphiques dans le panneau en haut à gauche par des liens vers trois nouveaux tableaux de bord. Ces tableaux de bord contiennent des informations sur les mesures suivantes :

- Mesures de scénario de test
- Mesures de simulateur
- Mesures d'étape de test

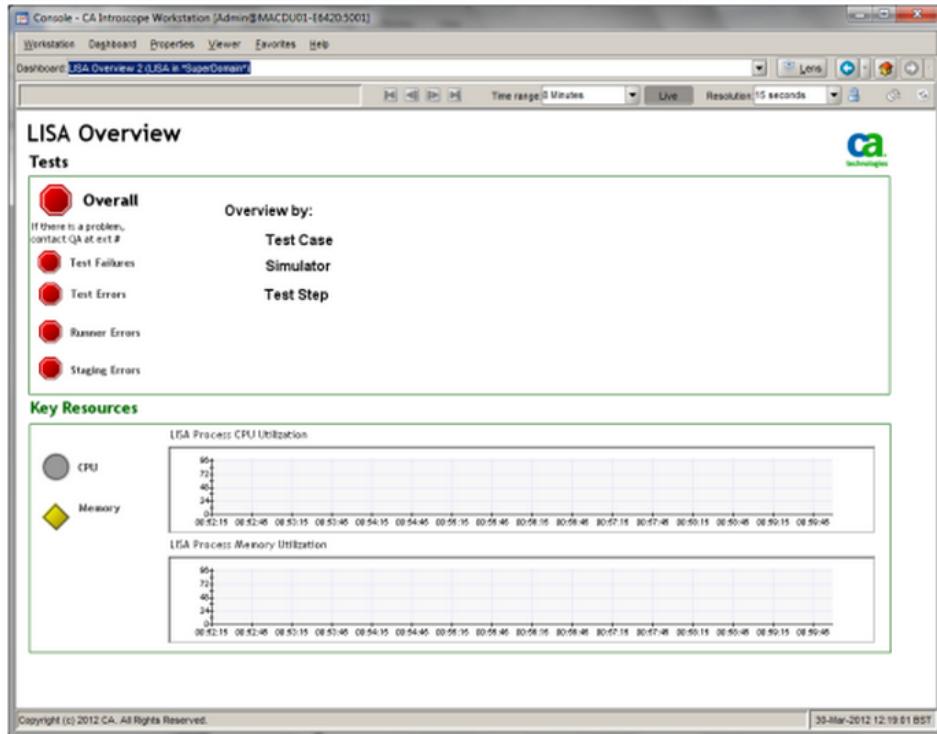


Tableau de bord Test Case Overview (Présentation de scénario de test)

Le tableau de bord Test Case Overview (Présentation de scénario de test) contient des graphiques pour les mesures suivantes :

- Average Response Time (ms) (Temps moyen de réponse (en ms))
- Responses Per Interval (Nombre de réponses par intervalle)
- Tests Running (Tests en cours d'exécution)
- Virtual Users Running (Utilisateurs virtuels en cours d'exécution)
- Failures Per Interval (Nombre d'échecs par intervalle)
- Errors Per Interval (Nombre d'erreurs par intervalle)

Dans chaque cas, seuls les dix premières mesures sont affichées.

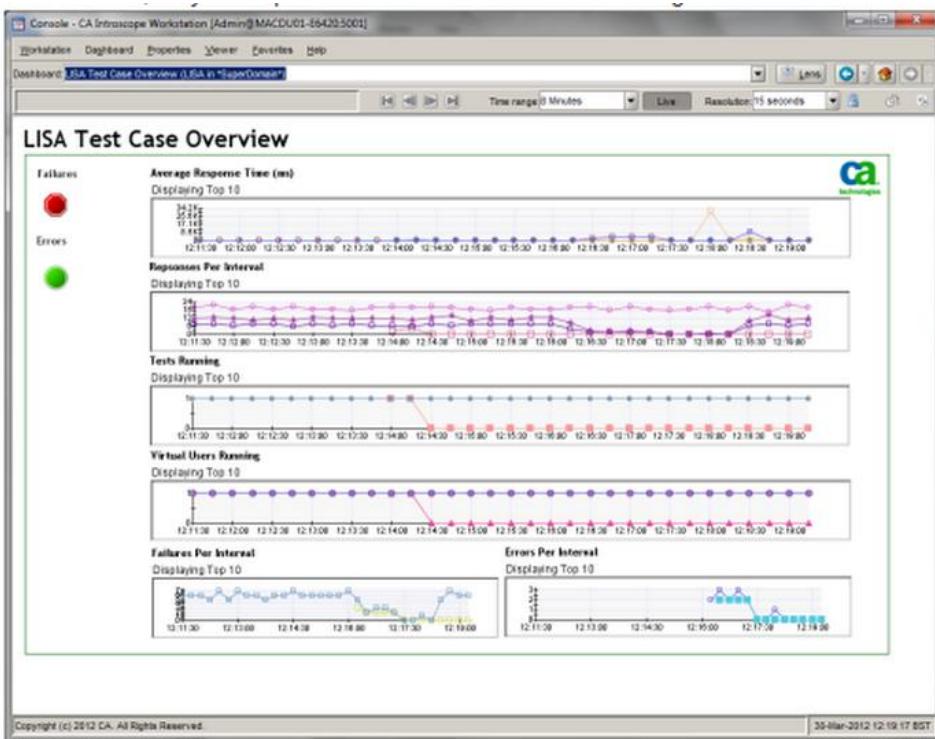


Tableau de bord Simulator Overview (Présentation du simulateur)

Le tableau de bord Simulator Overview (Présentation du simulateur) contient des graphiques pour les mesures suivantes :

- Average Response Time (ms) (Temps moyen de réponse (en ms))
- Responses Per Interval (Nombre de réponses par intervalle)
- Virtual Users Running (Utilisateurs virtuels en cours d'exécution)
- Failures Per Interval (Nombre d'échecs par intervalle)
- Errors Per Interval (Nombre d'erreurs par intervalle)

Dans chaque cas, seuls les dix premières mesures sont affichées.

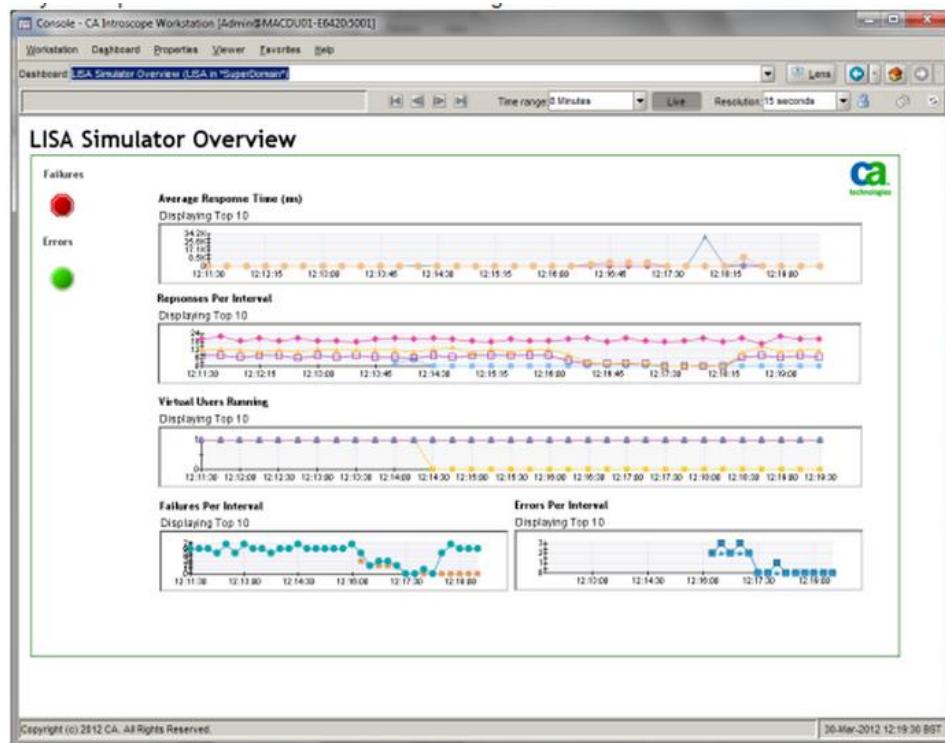


Tableau de bord Test Step Overview (Présentation des étapes de test)

Le tableau de bord Test Step Overview (Présentation des étapes de test) contient des graphiques pour les mesures suivantes :

- Average Response Time (ms) (Temps moyen de réponse (en ms))
- Responses Per Interval (Nombre de réponses par intervalle)
- Errors Per Interval (Nombre d'erreurs par intervalle)

Dans chaque cas, seuls les dix premières mesures sont affichées.

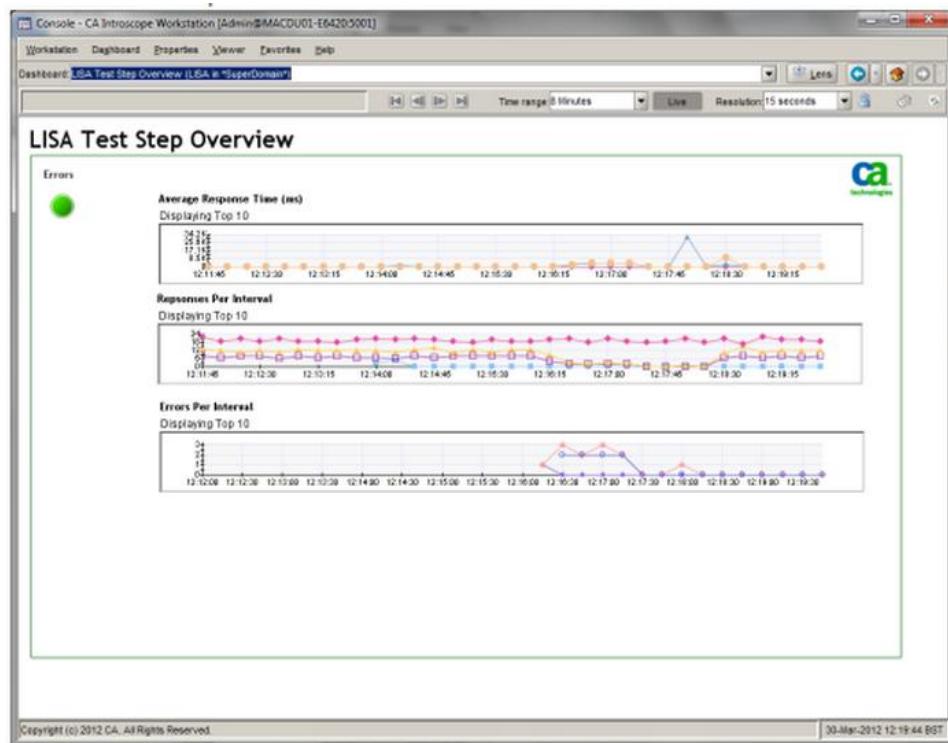


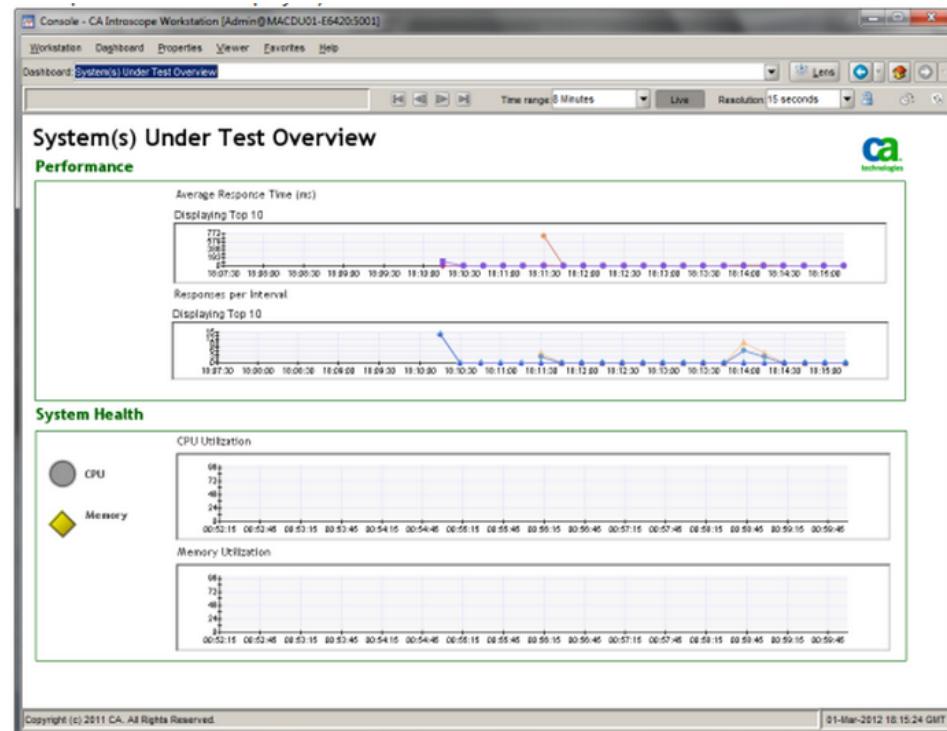
Tableau de bord SUT (Systèmes testés)

Le tableau de bord System(s) Under Test Overview (Présentation des systèmes testés) offre une vue d'ensemble des systèmes exécutant des transactions envoyées à partir de tests DevTest. Il est difficile d'identifier les systèmes testés par DevTest. Vous pouvez modifier les groupements de mesures pour définir les systèmes instrumentés par Introscope qui sont également testés par DevTest. Par défaut, le groupement contient le serveur de démonstration.

Le tableau de bord System(s) Under Test (Systèmes testés) affiche des mesures issues de systèmes testés par DevTest. Pour définir les agents qui instrumentent les systèmes testés, remplacez l'expression de l'agent de groupement de mesures pour tous les groupes de mesures définis dans le module Management (Gestion) par des noms commençant par Systems Under Test (Systèmes testés). Cette expression est définie par défaut sur **(.)|(.)|JBoss LISA Demo Server(.*)**, qui sélectionne le serveur de démonstration.

Le tableau de bord est divisé en deux sections. La section supérieure affiche les dix premières mesures frontales Average Response Time (ms) (Temps moyen de réponse, en ms) et Responses Per Interval (Nombre de réponses par intervalle) à partir de tous les serveurs d'applications testés.

La section inférieure indique l'utilisation de l'UC et de la mémoire de tous les serveurs d'applications testés. Cette section contient également deux icônes d'alerte qui indiquent les alertes relatives à l'utilisation de l'UC et de la mémoire.



Reporting (Génération de rapports)

Le module Management (Gestion) contient un rapport nommé DevTest Report. Ce rapport contient une série de graphiques décrivant l'état des systèmes testés par DevTest. Le rapport inclut également des mesures de test que DevTest collecte à partir de l'exécution de ces tests. Vous pouvez dupliquer et adapter le rapport à des tests spécifiques. Pour examiner des tests spécifiques, modifiez les expressions de regroupement de mesures dupliquées pour sélectionner uniquement les mesures pertinentes pour le rapport spécifique. Vous pouvez également omettre des graphiques spécifiques à partir du rapport dupliqué.

Configuration de SAP SLD (System Landscape Directory)

Cette rubrique décrit la procédure de configuration de SAP SLD (System Landscape Directory) pour CA Service Virtualization.

Procédez comme suit:

1. Obtenez les informations suivantes pour le serveur SLD :
 - Host (Hôte)
 - Port
 - User (Utilisateur)
 - Password (Mot de passe)
2. Connectez-vous à System Landscape Directory.
Remarque : vous devez disposer d'un rôle LcrInstanceWriterLD pour effectuer cette tâche.
3. Effectuez l'une des opérations suivantes :
 - [Enregistrement manuel de DevTest dans SLD](#) (page 138)
 - [Importation du fichier XML SLD pour DevTest](#) (page 139)

Enregistrement manuel de DevTest dans SLD

Cette rubrique décrit la procédure d'enregistrement manuel de DevTest auprès de SAP SLD (System Landscape Directory). Ce processus permet de créer un nom/une version de produit et un nom/une version de composant logiciel dans le catalogue de logiciels de SLD.

Procédez comme suit:

1. Dans le catalogue de logiciels de SLD, sélectionnez Products, New Product Version.
2. Dans le champ Product Name, entrez **CA LISA SV**.
3. Dans le champ Vendor Name, saisissez **CA Technologies**.
4. Dans le champ Product Version, saisissez le numéro de version ; par exemple : **V7.5**.
5. Cliquez sur Create (Créer).
6. Dans le champ Technical Name, entrez **CA LISA SV**.
7. Dans le champ Software Component Name, entrez **CA LISA**.
8. Dans le champ Software Component Version, saisissez le numéro de version ; par exemple : **V7.5**.
9. Dans le champ Production State, entrez **released**.
10. Cliquez sur Create (Créer).

Importation du fichier XML SLD pour DevTest

Cette rubrique décrit la procédure de configuration de SAP SLD (System Landscape Directory) en effectuant une importation du fichier XML de SLD pour DevTest.

Procédez comme suit:

1. Dans le catalogue de logiciels de SLD, sélectionnez Administration, Content, Import.
2. Dans le champ Selected File, saisissez le chemin d'accès au fichier .zip de données SLD fourni.
Ce fichier se trouve dans le répertoire **LISA_HOME\addons\sap\CALISA.xml**.
3. Cliquez sur Import Selected File.

Après l'installation de DevTest, consultez les chemins suivants :

- Chemin d'installation par défaut : **C:\lisa\bin**
- Chemin du fichier journal par défaut : **C:\Users\<ID d'utilisateur>\lisatmp_<numéro de version>**

Pour plus d'informations sur les fichiers journaux et leur signification, consultez la section Présentation des fichiers journaux dans la rubrique *Administration*.

Tous les fichiers journaux sont lisibles.

- Chemin d'accès au fichier de configuration par défaut : **C:\LISA_HOME**
- Fichiers de configuration par défaut : **lisa.properties** et **local.properties**

Pour plus d'informations sur l'ordre des fichiers de propriétés et la définition de valeurs dans les fichiers journaux, consultez la section Fichiers de propriété dans la rubrique *Utilisation de CA Application Test*.

Tous les fichiers de propriété sont des entrées Java lisibles, au format nom = valeur.

Chapitre 8: Configuration de l'environnement de test d'application mobile

Ce chapitre traite des sujets suivants :

- [Configuration système requise pour le test d'applications mobiles \(page 141\)](#)
- [Etapes de préinstallation pour le test d'application mobile \(page 142\)](#)
- [Utilisation de Genymotion \(page 149\)](#)

Configuration système requise pour le test d'applications mobiles

Cette section décrit la configuration système requise pour tester les applications mobiles dans DevTest Solutions.

Systèmes d'exploitation pris en charge pour le test d'application mobile

Les systèmes d'exploitation suivants sont pris en charge :

Mac OSX 10.8 - Mountain Lion

Les images de machine virtuelle sont prises en charge.

Mac OSX 10.9 - Mavericks

Les images de machine virtuelle sont prises en charge.

Windows 7 ou version ultérieure

Seuls les tests iOS distants sont pris en charge. Les tests iOS locaux ne le sont pas.

Systèmes d'exploitation mobiles pris en charge

Les systèmes d'exploitation mobiles suivants sont pris en charge :

- iOS 6.1 et version ultérieure
- Android 4.2 et version ultérieure
- Natif, hybride et Web

Remarque : La prise en charge d'Android Saucelabs est limitée à la version 4.2.

Configuration matérielle requise pour le test d'application mobile

Le matériel suivant est requis pour permettre la prise en charge du test d'application mobile :

UC

- Intel Core I5 2,0 GHZ ou équivalent AMD

Mémoire

- Tests d'applications mobiles cloud ou distants : 4 Go
- Simulateurs locaux : 8 Go

Etapes de préinstallation pour le test d'application mobile

Avant d'installer DevTest Solutions pour les tests d'application mobile, effectuez les tâches suivantes.

Etapes de préinstallation pour le test d'application mobile - Macintosh

Procédez comme suit:

1. Installez [Xcode 6.1](#) (page 143).

XCode est disponible gratuitement à partir de l'App Store de Mac.

2. Téléchargez le dernier kit de développement logiciel Android (ADT Bundle) à partir de :

<http://developer.android.com/sdk/index.html#download>

Remarque : Le test d'application mobile prend en charge les SDK uniquement à partir de la version SDK Platform 19 rev 3. Le groupement ADT Bundle contient des outils de création et le test d'application mobile requiert au minimum la version 19.0.1, 19.1.0, ou 20.0.0. Si ces versions ne sont pas installées avec votre groupement ADT Bundle, et qu'un message d'erreur s'affiche dans la station de travail DevTest Workstation sur zipalign ou aapt, reportez-vous à la section [Outils de création du SDK Android](#) (page 148).

3. [Définissez la propriété ANDROID_HOME](#) (page 146).
4. [Configurez votre kit de développement logiciel Android](#) (page 147).

Remarque : Les simulateurs d'appareil mobile disponibles dépendent de la version de Xcode ou du SDK installée.

Installez Xcode version 6.1.

Le test d'application mobile requiert Xcode 6.1 version ; les versions ultérieures ne sont pas prises en charge.

Téléchargement d'une nouvelle version

Procédez comme suit:

1. Accédez à <https://developer.apple.com/support/xcode>.
2. Téléchargez et installez Xcode 6.1.

Remarque : L'exécution de tests d'application iOS 7 à partir de Xcode 6 n'est pas prise en charge. Xcode 5 doit être installé pour l'exécution de tests d'application iOS 7.

Signature IPA

Pour tester des applications iOS sur des appareils, vous devez obtenir des certificats SSL et provisionner votre appareil à des fins de test. Vous pouvez effectuer cette action à partir du [f Apple Developer Member Center](#).

Vous pouvez ensuite provisionner votre appareil avec le fichier d'approvisionnement mobile pour identifier les applications signées à l'aide d'un certificat. L'ajout du fichier d'approvisionnement mobile à un fichier .ipa (via la signature IPA) permet d'assurer que les signatures sur le certificat correspondent aux signatures des utilisateurs pouvant exécuter les applications sur leur appareil.

Le test d'application mobile peut signer des fichiers iOS .ipa avant de les déployer vers votre appareil de test. Quelques conditions sont requises pour un fonctionnement correct.

Procédez comme suit:

1. Connectez-vous au Apple Developer [Member Center](#).
2. Cliquez sur Certificates, Identifiers, & Profiles.
3. Connectez votre appareil iOS à l'aide d'un câble USB.
4. Sous iOS Apps (à gauche), cliquez sur Devices.
5. Pour ajouter votre appareil, cliquez sur l'icône +.
Remarque : Vous pouvez également ajouter votre appareil via iTunes.
6. Confirmez que votre appareil est inclus dans la liste des appareils iOS.
7. Sous Certificates, cliquez sur Development.
8. Cliquez sur le certificat dans la liste.
9. Cliquez sur Download et enregistrez-le sur votre disque dur.
10. Installez ce certificat dans Keychain Access.
11. Sous Provisioning Profiles, cliquez sur Development.
12. Sélectionnez le profil de provisionnement iOS Team dans la liste.
13. Téléchargez ce fichier et enregistrez-le sur votre disque dur.
Le nom du fichier est similaire à l'exemple suivant :
iOS_Team_Provisioning_Profile_.mobileprovision
14. Ouvrez DevTest.
15. Dans le fichier de propriété *lisa.properties*, définissez les propriétés de projet suivantes :
*MOBILE_PROVISION = chemin d'accès à
iOS_Team_Provisioning_Profile_.mobileprovision*

`IOS_CERTIFICATE = nom du certificat tel qu'affiché dans Keychain Access` (non le nom de fichier de certificat même)

Remarque : Vous ne devez pas saisir la chaîne exacte, car l'utilitaire codesign peut y correspondre. Vous pouvez utiliser un nom tel que iPhone Developer, qui constitue un nom suffisamment unique. Toutefois, si vous avez d'autres certificats iOS, il est recommandé d'utiliser le nom de certificat complet.

Activation de UIAutomation

UIAutomation doit être activé sur des appareils iOS 8. L'activation de ce mode permet à Appium de détecter des instruments.

L'option UIAutomation est disponible sous Settings, Developer, Enable UI Automation.

Etapes de préinstallation pour les tests d'application mobile - Windows

Procédez comme suit:

1. Téléchargez et installez le dernier SDK Android.

<http://developer.android.com/sdk/index.html#download>

Remarque : DevTest prend en charge les SDK uniquement à partir de la version SDK Platform 19 rev 3. Le groupement ADT Bundle contient des outils de création et le test d'application mobile requiert au minimum la version 19.0.1, 19.1.0, ou 20.0.0. Si ces versions ne sont pas installées avec votre groupement ADT Bundle, et qu'un message d'erreur s'affiche dans la station de travail DevTest Workstation sur zipalign ou aapt, reportez-vous à la section [Outils de création du SDK Android](#) (page 148).

2. [Définissez la propriété ANDROID_HOME](#) (page 146).
3. [Configurez votre kit de développement logiciel Android](#) (page 147).

Définition de la propriété ANDROID_HOME

Pour que vos appareils virtuelle Android fonctionnent correctement, vous devez définir une propriété ANDROID_HOME.

Procédez comme suit:

1. Dans DevTest Workstation, cliquez sur System (Système), Edit Properties (Modifier les propriétés).
Les propriétés système s'ouvrent.
2. Insérez la propriété suivante :
`ANDROID_HOME= <Chemin complet vers votre groupement ADT Bundle>`
Par exemple :
`ANDROID_HOME= C:\ADT
Bundle\adt-bundle-windows-x86_64-20131030\sdk`
3. Cliquez sur Enregistrer.
4. Fermez la fenêtre des propriétés système.

Configuration du kit de développement logiciel Android

Le programme d'installation de DevTest Solutions inclut le groupement ADT Bundle (Android Developer Tools) contenant les outils utilisés pour développer des applications Android. Le groupement ADT Bundle inclut une version de l'IDE Eclipse avec ADT intégré. Utilisez le groupement ADT Bundle pour créer une ou plusieurs unités virtuelles Android.

Procédez comme suit:

1. Téléchargez le dernier kit de développement logiciel Android (ADT Bundle) :
<http://developer.android.com/sdk/index.html#download>
2. Décompressez le groupement ADT Bundle.
3. Exécutez **eclipse.exe** disponible dans le dossier **eclipse** du groupement ADT Bundle.
Le lanceur d'espaces de travail s'ouvre.
4. Sélectionnez l'espace de travail destiné au stockage de vos projets et cliquez sur OK.
La fenêtre Java ADT s'affiche.
5. Cliquez sur Window, Android Virtual Device Manager.
Le gestionnaire d'unités virtuelles Android s'ouvre.
6. Cliquez sur New.
La page Create New Android Virtual Device (AVD) (Créer une unité virtuelle Android) s'ouvre.
7. Renseignez les champs suivants :

AVD Name (Nom de RMI)

Nom de l'unité virtuelle Android à créer. Utilisez ce nom dans le champ **Android AVD** lors de la création d'un actif de session d'émulateur.

Unité

Sélectionnez le type d'appareil à tester.

Target (Cible)

Sélectionnez la cible de votre appareil virtuel Android.

Options de mémoire

Définissez la valeur de la mémoire RAM sur 512 et la valeur du segment de mémoire des machine virtuelle sur 32.

8. Acceptez les valeurs par défaut pour les champs restants ou personnalisez-les pour votre unité virtuelle Android.
9. Cliquez sur OK.
Votre nouvel appareil est ajouté à la liste des appareils dans le gestionnaire d'appareils virtuels Android.

Outils de création de kit de SDK pour Android

Le groupement SDK pour Android contient un composant appelé Build-tools (Outils de création). Le test d'application mobile requiert au minimum la version 19.0.1, 19.1.0, ou 20.0.0.

Si ces versions ne sont pas installées avec votre groupement ADT, un message d'erreur risque de s'afficher lors de la création d'un actif mobile dans la station de travail DevTest Workstation :

Cannot find the aapt command. Please install Android SDK Build-tools Rev 20. (Commande aapt introuvable. Installez les outils de création de SDK pour Android Rev 20).

Or

Cannot find the zipalign command. Please install Android SDK Build-tools Rev 20. (Commande zipalign introuvable. Installez les outils de création de SDK pour Android Rev 20).

Si vous rencontrez ce message d'erreur, mettez à jour votre version des outils de création du SDK pour Android.

Procédez comme suit:

1. Exécutez **android sdk** à partir du dossier qui contient le groupement ADT décompressé.
Le gestionnaire Android SDK Manager s'ouvre.
2. Sous le dossier Tools (Outils), vérifiez que la version 19.0.1, 19.1.0, ou 20.0.0 est installée et sélectionnée. La version 20.0.0 est la version optimale à utiliser.

Une fois la version correct des outils de création installée, vous pouvez poursuivre la création de votre actif mobile.

Activation du débogage USB

Vous pouvez activer le mode de débogage USB dans Android après avoir connecté votre unité directement à un ordinateur avec un câble USB. Activer ce mode vous permet d'établir une connexion entre une unité Android et le kit de développement logiciel Android.

Le mode de débogage USB se trouve dans les options développeur de votre unité, sous le menu Settings.

Utilisation de Genymotion

Genymotion est un émulateur Android basé sur VirtualBox. Genymotion a l'avantage d'être plus rapide que l'émulateur du Kit de développement Android et fournit de meilleures performances générales. Cet outil requiert une configuration supplémentaire.

Procédez comme suit:

1. Téléchargez et installez VirtualBox et Genymotion à partir de www.genymotion.com.

Remarque : L'installation de Genymotion requiert la création d'un compte d'utilisateur Genymotion, mais le téléchargement de base est libre. La plate-forme Windows dispose d'un programme d'installation Virtualbox/Genymotion combiné.

2. Ajoutez les unités virtuelles.

- a. Démarrez Genymotion.

La première fois que vous démarrez Genymotion, vous êtes invité à ajouter une unité virtuelle.

- b. Cliquez sur Yes (Oui).

La fenêtre Create a New Virtual Device s'affiche.

- c. Cliquez sur Connect et saisissez vos ID d'utilisateur et mot de passe Genymotion pour afficher les appareils disponibles.

Une liste d'unités virtuelles disponibles s'affiche. Vous pouvez filtrer la liste des unités par modèle d'unité ou par version Android en haut de la page.

- d. Sélectionnez l'appareil à ajouter, puis cliquez sur Next.

- e. Vérifiez les détails de l'unité sélectionnée.

Remarque: Le nom de l'appareil correspond au nom que vous entrez dans la boîte de dialogue d'actifs de DevTest Solutions. Vous pouvez modifier le nom Genymotion par défaut pour définir un nom plus simple.

- f. Cliquez sur Suivant.

Le téléchargement de votre unité virtuelle commence. La taille de ces images de téléchargement avoisine les 200 Mo.

- g. A l'issue du téléchargement, cliquez sur Finish.

- h. Pour ajouter un autre appareil virtuel, cliquez sur Add dans la fenêtre principale Genymotion et répétez les étapes précédentes.

3. Créez un fichier de configuration et un actif dans DevTest pour votre nouvel appareil virtuel.

- a. Créez un actif de session de simulateur standard pour un appareil virtuel Android. Pour plus d'informations, consultez la section Actifs mobiles de la rubrique *Utilisation de CA Application Test*.

Pour plus d'informations, consultez la section Didacticiel mobile 2 - Enregistrement d'un scénario de test Android dans la rubrique *Prise en main*.

- b. Dans le champ Android AVD (Unité virtuelle Android), entrez le nom de votre unité Genymotion, précédée par **genymotion:**. Par exemple :

`genymotion:Galaxy Nexus - 4.3 - API 18 - 720x1280`

Gestion des unités Genymotion

Vous pouvez gérer les unités Genymotion à l'aide de l'outil VBoxManage installé avec VirtualBox.

Voici une liste de lignes de commande utiles :

vboxmanage list avd

Répertorie toutes les unités virtuelles disponibles. Voici un exemple du résultat obtenu via cette commande :

```
Nexus 7 - 4.3 - API 18 - 1280x800"  
{144161dd-750e-4fff-8d46-0da8bc0c226b}  
GalaxyNexus4.2.2-API17" {d740a4fa-df15-4768-aee1-ffebfb883dc1}
```

Chaque ligne est une paire d'identificateurs. Le premier identificateur correspond au nom que vous avez donné à l'unité lors de sa création. Le deuxième identificateur correspond à l'UID. Vous pouvez utiliser n'importe quel identificateur lorsque vous définissez les unités en tant que cibles dans la ligne de commande. L'UID ne requiert pas besoin d'accolades lorsque vous l'utilisez dans la ligne de commande.

vboxmanage list runningvms

Répertorie les unités virtuelles en cours d'exécution.

Vous pouvez effectuer la même fonction avec la commande ADK, **\$ANDROID_HOME/platform-tools/adb devices**. La commande ADK répertorie les unités Genymotion et ADK émulées.

Voici un exemple du résultat obtenu via cette commande :

```
emulator-5554 device  
192.168.0.56:5555 device
```

Le préfixe emulator est ajouté aux unités ADK, tandis que les unités Genymotion commencent par l'adresse IP. Le nombre qui suit est le port utilisé par l'unité.

player --vm-name <UID ou nom>

Accédez au chemin d'installation de l'unité et utilisez cette commande pour la démarrer. Vous pouvez également démarrer une unité à partir de l'interface Genymotion.

Définition de la propriété VBOXMANAGE_CMD

Si vous installez VirtualBox dans un emplacement autre que par défaut, définissez la propriété VBOXMANAGE_CMD. Cette propriété permet à DevTest de rechercher l'outil vboxmanage, de sorte que vos simulateurs Genymotion fonctionnent de manière efficace.

Vous pouvez définir cette propriété de trois manières :

- Définissez une variable d'environnement dans Windows.
- Modifiez le fichier lisa.properties (pour appliquer à tous les projets).
- Ajoutez une propriété au fichier de propriété de projet dans l'éditeur de propriétés (pour appliquer à un seul projet).

Définition d'une variable d'environnement dans Windows

Dans les propriétés système de votre ordinateur, ouvrez la boîte de dialogue Environment Variables (Variables d'environnement) et ajoutez la commande suivante :

Windows

```
VBOXMANAGE_CMD=c:\program  
files\oracle\virtualbox\vboxmanage.exe
```

Mac

```
export VBOXMANAGE_CMD=/usr/bin/vboxmanage
```

Modification du fichier de propriété LISA

Modifiez le fichier lisa.properties affecte tous les projets définis dans la station de travail DevTest Workstation.

Procédez comme suit:

1. Dans le menu principal, sélectionnez System, Edit LISA Properties (Système, Modifier les propriétés LISA).
2. Pour définir VBOXMANAGE_CMD dans le fichier système, ajoutez la commande suivant :

```
VBOXMANAGE_CMD=c:\program  
files\oracle\virtualbox\vboxmanage.exe
```

3. Enregistrez le fichier.

Ajout d'une propriété au fichier de propriétés de projet

L'ajout d'une propriété au fichier de propriétés dans l'éditeur de propriétés affecte uniquement ce projet.

Procédez comme suit:

1. Dans le panneau Project (Projet), double-cliquez sur votre **projet** dans le dossier Configs.

L'éditeur de propriétés s'ouvre.

2. Pour ajouter une ligne, cliquez sur Add (Ajouter)  au bas de l'éditeur de propriétés.

3. Dans le champ Key (Clé), saisissez :

VBOXMANAGE_CMD

4. Dans le champ Value (Valeur), saisissez :

c:\program files\oracle\virtualbox\vboxmanage.exe



5. Dans la barre d'outils principale, cliquez sur Save (Enregistrer) .

Glossaire

Archive de modèle (MAR)

Une *archive de modèle (MAR)* est le principal artefact de déploiement dans DevTest. Les fichiers MAR contiennent un actif principal, tous les fichiers secondaires qui sont requis pour exécuter l'actif principal, un fichier d'informations et un fichier d'audit. Pour plus d'informations, consultez la section Utilisation des archives de modèle (MAR) de la rubrique *Utilisation de CA Application Test*.

assertion

Une *assertion* est un élément qui s'exécute après l'exécution d'une étape et de tous ses filtres. Les assertions vérifient que les résultats de l'étape sont conformes aux prévisions. Une assertion est généralement utilisée pour modifier le flux d'un scénario de test ou le modèle de service virtuel. Les assertions globales s'appliquent à chaque étape d'un scénario de test ou d'un modèle de service virtuel. Pour plus d'informations, consultez la section Assertions de la rubrique *Utilisation de CA Application Test*.

Audit document (Document d'audit)

Un *document d'audit* permet de définir des critères de réussite d'un test, ou d'un ensemble de tests dans une suite. Pour plus d'informations, consultez la section Génération de documents d'audit dans la rubrique *Utilisation de CA Application Test*.

companion (compagnon)

Un *Companion (Compagnon)* est un élément exécuté avant et après chaque exécution de scénario de test. Les compagnons sont comparables à des filtres applicables à l'ensemble du scénario de test, par opposition à des étapes de test spécifiques. Les compagnons sont utilisés pour configurer le comportement global dans le scénario de test. Pour plus d'informations, consultez la section Compagnons de la rubrique *Utilisation de CA Application Test*.

Configuration

Une *configuration* est une collection nommée de propriétés qui spécifient en général des valeurs propres à un environnement pour le système testé. La suppression de données d'environnement codées de manière irréversible permet d'exécuter un scénario de test ou un modèle de service virtuel au niveau d'environnements différents en modifiant simplement des configurations. La configuration par défaut dans un projet est appelée `project.config`. Un projet peut contenir de nombreuses configurations, mais une seule configuration peut être active à la fois. Pour plus d'informations, consultez la section Configurations de la rubrique *Utilisation de CA Application Test*.

Conversation tree (Arborescence des conversations)

Une *arborescence des conversations* est un ensemble de noeuds liés qui représentent des chemins de conversation pour les transactions avec état dans une image de service virtuel. Chaque noeud porte une étiquette de nom d'opération ; par exemple : withdrawMoney. Exemple de chemin de conversation pour un système d'opérations bancaires : getToken, getAccount, withdrawMoney, deleteToken. Pour plus d'informations, reportez-vous à la rubrique *Utilisation de CA Service Virtualization*.

coordinator (coordinateur)

Le *coordinateur* reçoit les informations d'exécution de test sous forme de document et coordonne les tests exécutés sur un ou plusieurs serveurs de simulation. Pour plus d'informations, consultez la section Coordinator Server (Serveur de coordination) de la rubrique *Utilisation de CA Application Test*.

data protocol (protocole de données)

Un *protocole de données* est également appelé gestionnaire de données. Dans CA Service Virtualization, un protocole de données est responsable de la gestion de l'analyse des demandes. Certains protocoles de transport autorisent (ou requièrent) un protocole de données auquel le job de création de demandes est délégué. C'est pourquoi le protocole doit connaître la charge utile de la demande. Pour plus d'informations, consultez la section Utilisation de protocoles de données dans la rubrique *Utilisation de CA Service Virtualization*.

Data set (Ensemble de données)

Un *Data set (Ensemble de données)* est une collection de valeurs que vous pouvez utiliser pour définir des propriétés dans un scénario de test ou un modèle de service virtuel lors de l'exécution. Les ensembles de données fournissent un mécanisme permettant d'introduire des données de test externes dans un scénario de test ou un modèle de service virtuel. Vous pouvez créer des ensembles de données internes à DevTest, ou de manière externe ; par exemple, dans un fichier ou une table de base de données. Pour plus d'informations, consultez la section Ensembles de données de la rubrique *Utilisation de CA Application Test*.

desensitize (désensibiliser)

La *désensibilisation* consiste à convertir des données sensibles par des valeurs de substitution définies par l'utilisateur. Par exemple, les numéros de carte de crédit et les numéros de sécurité sociale sont des données sensibles. Pour plus d'informations, consultez la section Désensibilisation de données de la rubrique *Utilisation de CA Service Virtualization*.

Event (Événement)

Un *Event (Événement)* est un message sur une action qui s'est produite. Vous pouvez configurer des événements au niveau d'un scénario de test ou d'un modèle de service virtuel. Pour plus d'informations, consultez la section Introduction aux événements de la rubrique *Utilisation de CA Application Test*.

Filter (Filtre)

Un *filtre* est un élément exécuté avant et après une étape. Les filtres permettent de traiter les données des résultats ou de stocker des valeurs dans des propriétés. Les filtres globaux s'appliquent à chaque étape d'un scénario de test ou d'un modèle de service virtuel. Pour plus d'informations, consultez la section Filtres de la rubrique *Utilisation de CA Application Test*.

Groupe

Un *groupe* ou un *groupe de services virtuels* est une collection de services virtuels portant la même balise de groupe. Cela permet de les surveiller ensemble dans la console VSE.

Interactive Test Run (ITR) (Exécuter un test interactif)

L'utilitaire *Interactive Test Run (ITR)* (Exécuter un test interactif) permet d'exécuter un scénario de test ou un modèle de service virtuel étape par étape. Vous pouvez changer le scénario de test ou le modèle de service virtuel lors de l'exécution et le réexécuter pour vérifier les résultats. Pour plus d'informations, consultez la section Utilisation de l'utilitaire Interactive Test Run (ITR) (Exécuter un test interactif) de la rubrique *Utilisation de CA Application Test*.

Lab (Laboratoire)

Un *laboratoire* est un conteneur logique pour un ou plusieurs membres de laboratoire. Pour plus d'informations, consultez la section Laboratoires et membres de Laboratoire dans la rubrique *Utilisation de CA Application Test*.

magic date (date magique)

Pendant un enregistrement, un analyseur de dates analyse les demandes et les réponses. Une valeur correspondant à une définition étendue de formats de la date est convertie en *date magique*. Les dates magiques permettent de vérifier que le modèle de service virtuel fournit des valeurs de date explicites dans des réponses. Exemple de date magique : {{=doDateDeltaFromCurrent("yyyy-MM-dd","10");/*2012-08-14*/}. Pour plus d'informations, consultez la section Chaînes et dates magiques de la rubrique *Utilisation de CA Service Virtualization*.

Magic string (Chaîne magique)

Une *chaîne magique* est une chaîne générée pendant la création d'une image de service. Une chaîne magique est utilisée pour vérifier que les réponses fournies par le modèle de service virtuel contiennent des valeurs de chaîne explicites. Exemple de chaîne magique : {{=request_fname;/chris/}}. Pour plus d'informations, consultez la section Chaînes et dates magiques de la rubrique *Utilisation de CA Service Virtualization*.

Match tolerance (Tolérance de correspondance)

La *tolérance de correspondance* est un paramètre qui permet de contrôler la méthode utilisée par CA Service Virtualization pour comparer une demande entrante avec les demandes dans une image de service. Les options disponibles sont EXACT, SIGNATURE et OPERATION. Pour plus d'informations, consultez la section Tolérance de correspondance de la rubrique *Utilisation de CA Service Virtualization*.

Metrics (Mesures)

Les mesures permettent d'appliquer des méthodes et des mesures quantitatives aux performances et aux aspects fonctionnels de vos tests, ainsi qu'au système testé. Pour plus d'informations, consultez la section Génération de mesures de la rubrique *Utilisation de CA Application Test*.

Model Archive (MAR) Info (Fichier d'informations d'archive de modèle (MAR))

Un *Model Archive (MAR) Info (Fichier d'informations d'archive de modèle (MAR))* est un fichier qui contient des informations requises pour la création d'une archive de modèle MAR. Pour plus d'informations, consultez la section Utilisation des archives de modèle (MAR) de la rubrique *Utilisation de CA Application Test*.

navigation tolerance (tolérance de navigation)

La tolérance de navigation est un paramètre qui permet de contrôler le méthode utilisée par CA Service Virtualization pour rechercher la transaction suivante dans une arborescence des conversations. Les options disponibles sont CLOSE, WIDE et LOOSE. Pour plus d'informations, consultez la section Tolérance de navigation de la rubrique *Utilisation de CA Service Virtualization*.

Network graph (Graphique du réseau)

Le graphique de réseau est une zone de la console de serveur qui contient une représentation graphique du composant DevTest Cloud Manager et des laboratoires associés. Pour plus d'informations, consultez la section Démarrage d'un laboratoire de la rubrique *Utilisation de CA Application Test*.

Node (Noeud)

Une étape de test interne à DevTest peut également être appelée *node* (noeud), ce qui explique l'intégration du terme node dans l'ID de certains événements.

Path (Chemin)

Un *chemin* contient des informations sur une transaction capturée par l'agent Java. Pour plus d'informations, reportez-vous à la rubrique *Utilisation de CA Continuous Application Insight*.

Path graph (Graphique de chemin)

Un *graphique de chemin* contient une représentation graphique d'un chemin et ses trames. Pour plus d'informations, consultez la section Graphique de chemin de la rubrique *Utilisation de CA Continuous Application Insight*.

Project (Projet)

Un *projet* est une collection de fichiers DevTest liés. Les fichiers peuvent inclure des scénarios de test, des suites, des modèles de service virtuel, des images de service, des configurations, des documents d'audit, des documents de simulation, des ensembles de données, des moniteurs et des fichiers d'informations MAR. Pour plus d'informations, consultez la section Panneau Project (Projet) de la rubrique *Utilisation de CA Application Test*.

Property (Propriété)

Une *propriété* est une paire clé-valeur que vous pouvez utiliser comme variable d'exécution. Les propriétés peuvent stocker plusieurs types de données différents. Exemple de propriétés communes : LISA_HOME, LISA_PROJ_ROOT et LISA_PROJ_NAME. Une configuration est une collection nommée de propriétés. Pour plus d'informations, consultez la section Propriétés de la rubrique *Utilisation de CA Application Test*.

quick test (test rapide)

La fonctionnalité *Quick test (Test rapide)* permet d'exécuter un scénario de test avec une installation minimale. Pour plus d'informations, consultez la section Simulation d'un test rapide du *Utilisation de CA Application Test*.

Registry (Registre)

Le *registre* fournit un emplacement central pour l'enregistrement de tous les composants de DevTest Server et de DevTest Workstation. Pour plus d'informations, consultez la section Registre de la rubrique *Utilisation de CA Application Test*.

ressource

Un *actif* est un ensemble de propriétés de configuration groupées dans une unité logique. Pour plus d'informations, consultez la section Actifs de la rubrique *Utilisation de CA Application Test*.

service image (image de service)

Une *image de service* est une version normalisée de transactions enregistrées dans CA Service Virtualization. Chaque transaction peut être avec état (conversationnel) ou sans état. Une image de service peut être créée à l'aide de l'enregistreur d'image de service virtuel. Les images de service sont stockées dans un projet. Une image de service est également appelée *virtual service image (image de service virtuel, VSI)*. Pour plus d'informations, consultez la section Images de service de la rubrique *Utilisation de CA Service Virtualization*.

Simulator (Simulateur)

Un *simulateur* exécute les tests sous la surveillance du serveur de coordination. Pour plus d'informations, consultez la section Simulator Server (Serveur de simulation) de la rubrique *Utilisation de CA Application Test*.

Staging document (Document de simulation)

Un *document de simulation* contient les informations sur l'exécution d'un scénario de test. Pour plus d'informations, consultez la section Génération de documents de simulation dans la rubrique *Utilisation de CA Application Test*.

Subprocess (Sous-processus)

Un *sous-processus* est un scénario de test appelé par un autre scénario de test. Pour plus d'informations, consultez la section Génération de sous-processus de la rubrique *Utilisation de CA Application Test*.

Tableau de bord Continuous Service Validation (CVS) (Service de validation en continu)

Le *Continuous Service Validation (CVS) Dashboard* (*Tableau de bord du service de validation en continu*) permet de planifier l'exécution régulière de scénarios de test et des suites de test, sur une période étendue. Pour plus d'informations, consultez la section Service de validation en continu dans la rubrique *Utilisation de CA Application Test*.

test case (scénario de test)

Un *scénario de test* est une spécification de la procédure de test d'un composant métier dans le système testé. Chaque scénario de test contient une ou plusieurs étapes de test. Pour plus d'informations, consultez la section Génération de scénarios de test dans la rubrique *Utilisation de CA Application Test*.

test step (étape de test)

Une *étape de test* est un élément du flux de travaux de scénario de test qui représente une action de test unique à réaliser. Exemples d'étapes : Services Web, JavaBeans, JDBC et messagerie JMS. Une étape de test peut contenir des éléments de DevTest, tels que des filtres, des assertions et des ensembles de données liés. Pour plus d'informations, consultez la section Génération d'étapes de test de la rubrique *Utilisation de CA Application Test*.

Test suite (Suite de tests)

Une *suite de tests* est un groupe de scénarios de test, d'autres suites de tests, ou les deux, planifiés pour être exécutés l'un après l'autre. Un document de suite de tests spécifie le contenu de la suite, les rapports à générer et les mesures à collecter. Pour plus d'informations, consultez la section Génération de suites de tests de la rubrique *Utilisation de CA Application Test*.

Think time (Délai de réflexion)

Le *temps de réflexion* est la durée d'attente d'un scénario de test patiente avant d'exécuter une étape de test. Pour plus d'informations, consultez les sections Exemple d'ajout d'étape de test et Editeur de documents de simulation - Onglet Base de la rubrique *Utilisation de CA Application Test*.

transaction frame (Trame de transaction)

Une *trame de transaction* contient des données sur un appel de méthode intercepté par l'agent Java de DevTest ou un agent Light de CAI. Pour plus d'informations, consultez la section Transactions organisationnelles et trames de transaction de la rubrique *Utilisation de CA Continuous Application Insight*.

Virtual Service Environment (Environnement de service virtuel, VSE)

Le *Virtual Service Environment* (Environnement de service virtuel, VSE) est une application de DevTest Server qui permet de déployer et d'exécuter des modèles de service virtuel. VSE est également appelé CA Service Virtualization. Pour plus d'informations, reportez-vous à la rubrique *Utilisation de CA Service Virtualization*.

virtual service model (VSM, modèle de service virtuel)

Un *modèle de service virtuel* reçoit des demandes de service auxquelles il répond en l'absence du fournisseur de services réel. Pour plus d'informations, consultez la section Modèle de service virtuel de la rubrique *Utilisation de CA Service Virtualization*.