

VISHNU D1.0 - Spécifications techniques des besoins



COLLABORATORS

	<i>TITLE :</i> VISHNU D1.0 - Spécifications techniques des besoins		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Benjamin Isnard, Daouda Traoré, and Eugène Pamba Capo-Chichi	January 7, 2011	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME
01	07/12/2010	Exemple pour validation du format	B.Isnard
02	13/12/2010	Pre-delivrable	B.Isnard

Contents

1	Présentation du document	1
1.1	Objectifs	1
1.2	Structure du document	1
1.3	Références	1
1.4	Glossaire	1
2	Sécurité du système et des données	2
2.1	Tableau des Exigences	2
2.2	Glossaire	4
3	Performances, fiabilité et robustesse	5
3.1	Tableau des Exigences	5
3.2	Glossaire	6
4	Environnement matériel et logiciel	7
4.1	Tableau des Exigences	7
4.2	Glossaire	8
5	Installation et maintenance	9
5.1	Tableau des Exigences	9
5.2	Glossaire	9

Chapter 1

Présentation du document

1.1 Objectifs

Ce document présente les spécifications techniques des besoins pour le système Vishnu. Ces spécifications décrivent les caractéristiques techniques du système à réaliser du point de vue des utilisateurs d'une part et des administrateurs du système d'autre part. Elles sont basées sur les documents de Cahier des charges du système Vishnu (voir Références) et sur la proposition technique de la société Sysfera. Ces spécifications incluent également les contraintes d'implémentation du logiciel. L'objectif principal de ces spécifications est de valider l'adéquation de la solution proposée par rapport aux besoins des utilisateurs du système d'information et de ses administrateurs d'une part, et aux contraintes de l'environnement d'utilisation d'autre part.

Ce document pourra également contenir des prévisions sur les évolutions futures du logiciel et préciser quelles fonctions devraient être faciles à ajouter ou supprimer.

Ces spécifications techniques des besoins sont un prérequis pour les spécifications générales dans le processus de développement suivi pour le projet Vishnu.

1.2 Structure du document

Le document est composé de 4 parties correspondant à des domaines techniques différents:

- Besoins liés à la sécurité du système et des données
- Besoins liés à l'environnement matériel et logiciel
- Besoins en performance, fiabilité et robustesse
- Besoins pour l'installation et la maintenance du système

Chaque partie contient un tableau des exigences qui fait l'inventaire des tous les besoins techniques concernant le domaine spécifié. Ce tableau est suivi d'un dictionnaire des termes techniques employés afin de les préciser le cas échéant.

1.3 Références

1.4 Glossaire

- UMS ("User Management System"): nom du module Vishnu de gestion des sessions et des utilisateurs
 - TMS ("Tasks Management System"): nom du module Vishnu de gestion des tâches
 - FMS ("Files Management System"): nom du module Vishnu de gestion des transferts de fichiers
 - IMS ("Information Management System"): nom du module Vishnu de gestion des informations
-

Chapter 2

Sécurité du système et des données

2.1 Tableau des Exigences

ID	Name	Text
1	Sessions et authentification	
1.1	Types d'utilisateurs	Les utilisateurs de l'intergiciel réparti VISHNU seront de deux types : utilisateurs ou administrateurs. Les administrateurs sont des utilisateurs avec des droits supplémentaires. L'identification dans le système d'un utilisateur se fera à l'aide d'un login unique et d'un mot de passe qui sera crypté.
1.2	Format des identifiants et mots de passe	Le login et le mot de passe respecteront respectivement les expressions régulières suivantes: ^[A-Za-z0-9_]{LOGIN_MIN_SIZE,LOGIN_MAX_SIZE}\$ et ^[^\s]{PASSWORD_MIN_SIZE,PASSWORD_MAX_SIZE}\$
1.3	Caractéristique d'une authentification	Une authentification se fait à l'aide d'un login et d'un mot de passe qui doivent être au préalable enregistrés dans une base de données Postgresql version 8.4 pour vérification. Le login est unique et le mot de passe enregistré est crypté en utilisant la fonction crypt de la librairie libcrypt de Linux. La procédure d'authentification est donc sécurisée en ce sens que le login est unique dans la base de données et que le mot de passe est crypté par un algorithme efficace SHA-512. Avant tout ajout d'un nouvel utilisateur via vishnu dans la base de données le système vérifie que le login n'est pas déjà utilisé.
1.4	Authentification unique	Une fois authentifié dans l'intergiciel, l'utilisateur n'aura plus à s'authentifier pour utiliser les services de l'intergiciel avec son propre compte. Un identifiant de session sera créé par le système et retourné à l'utilisateur. Cet identifiant est ensuite enregistré dans une variable VISHNU_SESSION_ID pour éviter d'avoir à fournir cette information pour chaque requête (dans le cadre du client shell Unix). La complexité de cet identifiant garantit son unicité mais surtout un niveau de sécurité pour une utilisation du système sans authentification systématique.
1.5	Fermeture de session	
1.5.1	Fermeture manuelle de session	L'utilisateur peut fermer manuellement la session dans laquelle il se trouve. Avant la fermeture, le système vérifie la présence de commandes en cours d'exécution dans la session. Si il existe une ou plusieurs commandes en cours d'exécution le système rend impossible la fermeture de session sinon la session est fermée. Cette fermeture est possible en utilisant l'identifiant de session défini dans la variable VISHNU_SESSION_ID.

ID	Name	Text
1.5.2	Choix de l'option de fermeture automatique de session	L'utilisateur peut choisir entre deux options de fermeture automatique de session: soit l'option de fermeture à la déconnexion du terminal, soit l'option de fermeture après expiration du délai d'inactivité (période sans aucun appel aux services de l'intergiciel). Le choix est enregistré dans la configuration permanente de l'utilisateur et peut également être modifié pour une session donnée. Le choix de fermeture à la déconnexion du terminal n'est possible que dans le cas d'une session de type "shell unix". Dans les deux cas, la présence de commandes en cours d'exécution rend impossible la fermeture de session. Par contre, pour l'option de fermeture après expiration du délai d'inactivité, il y a une réinitialisation du délai d'inactivité lorsqu'il y a des commandes en cours d'exécution.
1.5.3	Vérification des commandes en cours avant fermeture de session	Le système vérifie que toutes les commandes lancées pendant la session sont terminées avant de fermer la session. Dans le cas où l'utilisateur s'est déconnecté de son terminal et que la session doit être fermée automatiquement, la session ne sera pas fermée avant la fin de l'exécution de toutes les commandes en cours, et elle sera fermée après expiration du délai d'inactivité.
1.6	Reconnection à une session	Un utilisateur peut se connecter à une session déjà ouverte sur la même machine cliente en fournissant un identifiant unique de session ainsi que son login et mot de passe. S'il se trouve déjà dans une session ouverte, le système lui propose de fermer cette session ou bien de la conserver ouverte avec une déconnexion automatique au bout du temps maximum d'inactivité.
1.7	Accès aux informations sur les sessions	Pour un utilisateur normal, l'affichage des sessions (actives ou fermées) n'est possible qu'après une authentification à partir du login et du mot de passe de l'utilisateur. Un administrateur peut par contre accéder à toutes les informations de sessions de tous les utilisateurs.
1.8	Accès aux machines	
1.8.1	Création et modification d'accès	L'accès à une machine au travers de l'intergiciel Vishnu se fera à l'aide d'un couple clé SSH privée/publique spécifique à chaque couple (machine, utilisateur). L'utilisateur du système Vishnu devra ajouter la clef publique aux clefs autorisées sur son compte Unix.
1.8.2	Révocation d'un accès au compte unix par un utilisateur	L'utilisateur peut supprimer l'accès au travers de l'intergiciel Vishnu à son compte Unix sur une machine donnée en supprimant simplement la clef publique de la liste des clefs autorisées sur son compte.
1.8.3	Respect des autorisations d'accès	L'accès au compte de l'utilisateur sur une machine et aux informations y étant stockées se faisant via SSH avec le compte de l'utilisateur, tous les droits d'accès système seront respectés.
1.8.4	Utilisation de comptes individuels	Un utilisateur Vishnu ne peut enregistrer un compte Unix sur une machine donnée que si ce compte n'est pas déjà utilisé par un autre utilisateur Vishnu. Si c'est le cas, le système Vishnu doit refuser la création (ou modification) de l'accès à la machine.
1.9	Changement de mot de passe	Un utilisateur peut modifier son mot de passe lorsqu'il a déjà ouvert une session. Le nouveau mot de passe est actif immédiatement c'est-à-dire que toute nouvelle authentification se fera avec le nouveau mot de passe.
2	Sécurité des communications	
2.1	Données des requêtes au système	Les requêtes au système vishnu sont authentifiées mais non cryptées.
2.2	Communications à travers un firewall	Les communications entre machines du système Vishnu pourront être configurées pour traverser un firewall permettant les communications cryptées.

ID	Name	Text
2.3	Fichiers des utilisateurs	Les fichiers des utilisateurs sont transférés en utilisant le cryptage SSH avec les clefs de l'utilisateur. Pour qu'un transfert puisse avoir lieu entre deux machines l'utilisateur doit avoir configuré ses clefs SSH (indépendamment de Vishnu) pour permettre une communication entre ces deux machines.
2.4	Informations stockées par le système	Le système Vishnu stockera toutes les informations relatives aux utilisateurs, aux tâches, transferts de fichiers, mesures de performance et trace des requêtes dans une base de données PostgreSQL sans être cryptées. L'accès du système Vishnu à la base de données sera authentifié par login/mot de passe dont l'envoi au travers de la connexion sera réalisé après hachage MD5. Les communications entre le système Vishnu et la base de données pourront être cryptées si nécessaires par le protocole SSL en configurant le système Vishnu et la base de données.
3	Sécurité des fonctions administrateurs	
3.1	Sécurité de la configuration du système	
3.1.1	Consultation et modification de la configuration UMS	Un administrateur peut consulter et modifier l'ensemble de la configuration des utilisateurs.
3.1.2	Sauvegarde et restauration de la configuration UMS	L'ensemble de la configuration des utilisateurs peut être sauvegardée dans un fichier et restaurée. La nouvelle configuration vient remplacer la configuration existante. La restauration a lieu sans redémarrage de l'intergiciel.
3.1.3	Prise en compte des modifications	La prise en compte des modifications de la configuration est immédiate sans redémarrage de l'intergiciel
3.2	Configuration du délai max. d'inactivité	Un administrateur peut définir la valeur maximum du délai d'inactivité pris en compte pour la déconnexion automatique.
3.3	Configuration de l'option par défaut de fin de session	Un administrateur peut définir quelle est l'option par défaut pour la fermeture automatique d'une session utilisateur: soit terminaison en cas de déconnexion de l'utilisateur du client, soit terminaison après expiration du délai d'inactivité
3.4	Utilisation d'un compte tiers	Un administrateur de l'intergiciel pourra ouvrir une session VISHNU avec le compte d'un utilisateur n'étant pas administrateur lui-même.

2.2 Glossaire

- **SSH**: à la fois programme exécutable et protocole de communication sécurisé utilisant un échange de clés de chiffrement en début de connexion.

Chapter 3

Performances, fiabilité et robustesse

3.1 Tableau des Exigences

ID	Name	Text
1	Fiabilité	
1.1	Réponse aux requêtes	Toute requête lancée à travers le système Vishnu conduit à une réponse. Cette réponse peut être soit le résultat prévu, soit un message d'erreur. Pour chaque message d'erreur, le système vishnu associe un code d'erreur (type entier) correspondant. Un code égal à zero correspondra à un résultat correct. Les interfaces de programmation de Vishnu fourniront un mécanisme d'exceptions permettant à l'utilisateur de traiter séparément les erreurs selon leur cause.
1.2	Enregistrement des requêtes	Le système Vishnu conserve la trace de chaque requête utilisateur dans une base de données. Cette trace comprend les identifiants de l'utilisateur et de la machine cliente, le nom de la commande ayant permis le lancement de la requête et les paramètres de la requête. Dans le cas d'une requête soumise via l'API Web Services, la machine cliente sera le serveur web lui-même.
1.3	Reproductibilité des requêtes	Le système Vishnu peut ré-exécuter une ou plusieurs requêtes à partir des traces enregistrées. Pour chaque requête, les paramètres de la requête seront les mêmes que lors de la soumission initiale à l'exception de l'utilisateur et de la machine cliente qui proviendront de la session courante (remarque: un administrateur peut ouvrir une session en tant qu'un autre utilisateur).
1.4	Erreurs dues à l'infrastructure	Lorsqu'une machine de l'infrastructure est arrêtée ou injoignable le système répond aux requêtes des utilisateurs concernant cette machine avec un message d'erreur indiquant que la machine n'est pas disponible.
1.5	Erreurs d'utilisation	Une requête lancée avec des paramètres incorrects sera rejetée par le système Vishnu, et pour chaque paramètre incorrect, le système Vishnu indiquera le motif du rejet. Le système Vishnu indiquera à l'utilisateur l'utilisation correcte de la requête avec les bons paramètres.
1.6	Erreurs dues à l'intergiciel	Lorsqu'un processus de l'intergiciel Vishnu crashe pendant l'exécution d'un service le système Vishnu informera l'utilisateur de l'arrêt de ce service.
1.7	Erreurs dues aux droits d'accès	L'accès à certaines informations nécessite des permissions en lecture ou écriture. Si un utilisateur n'a pas les droits nécessaires pour accéder à ces informations, le système Vishnu l'en informera par un message indiquant la nature de la permission manquante et l'objet concerné.
2	Capacité	

ID	Name	Text
2.1	Nombre de requêtes simultanées	Le système pourra supporter au minimum 100 requêtes simultanées.
2.2	Nombre d'utilisateurs	Le système pourra supporter au minimum 1000 utilisateurs enregistrés.
2.3	Nombre de machines serveur	Le système pourra supporter au minimum 10 machines serveurs (supercalculateurs ou frontales de cluster)
2.4	Nombre de machines clientes	Le système pourra supporter au minimum 1000 machines clientes (postes scientifiques)
3	Performance	
3.1	Latence	En fonctionnement normal (ressources des machines non saturées), avec l'interface d'utilisation en ligne de commande, la latence sera inférieure ou égale à celle obtenue par une commande ssh ayant le même effet en termes de soumission de job ou transfert de fichiers.
3.2	Consommation de ressources	
3.2.1	Consommation moyenne des processus Vishnu	TODO
3.2.2	Consommation ponctuelle pour l'exécution d'une requête utilisateur	L'utilisation de Vishnu pour exécuter des requêtes sur un gestionnaire de jobs (soumission de job ou interrogation) ne devra pas consommer en moyenne davantage de ressources sur le serveur que l'utilisation d'un client en ligne de commande (client "lourd" du gestionnaire de jobs).
4	Robustesse	
4.1	Absence de plantage intempestif	Les processus du système ne doivent pas s'interrompre en fonctionnement "de croisière" (c'est-à-dire hors de toute défaillance logicielle ou matérielle du système hôte).
4.2	Résistance aux défaillances	Le système doit rester disponible en cas de défaillance de l'infrastructure n'affectant pas la connection de l'utilisateur aux point(s) d'entrée du système. Plusieurs points d'entrée pourront être configurés mais le nombre de processus lancés par le système sur les machines sera alors augmenté.
4.3	Délestage	Le délestage permettra à un administrateur de réduire de manière immédiate la charge générée par l'intergiciel sur une machine de l'infrastructure. Deux niveaux de délestage seront possible.
4.3.1	Délestage "soft"	Au niveau "soft", le délestage interrompra les activités en cours sur la machine concernée (activités contrôlées par le système Vishnu uniquement). Ces activités pourront être relancées par le système lors de la commande de fin de délestage exécutée par l'administrateur.
4.3.2	Délestage "hard"	Au niveau "hard", le délestage interrompra tous les processus de l'intergiciel sur la machine concernée sans effectuer d'actions éventuellement nécessaires à leur redémarrage. Le redémarrage de Vishnu sur la machine utilisera la même procédure qu'un démarrage "à froid" standard.

3.2 Glossaire

- TODO

Chapter 4

Environnement matériel et logiciel

4.1 Tableau des Exigences

ID	Name	Text
1.	Plateformes supportées	
1.1	Machines clientes	
1.1.1	Système d'exploitation	Debian Linux v5 (Lenny) ou ultérieure
1.1.2	Architecture matérielle	Intel x86
1.1.3	Nombre de processeurs	Non limité
1.1.4	Mémoire vive	Minimum 1Go
1.1.5	Espace disque	Minimum 1Go disponible
1.1.6	Interfaces réseau	
1.2	Serveurs de Calcul	
1.2.1	Système d'exploitation	Debian Linux v5 (Lenny) ou ultérieure
1.2.2	Architecture matérielle	Intel x86 ou IBM BlueGene/P® (PPC 32-bits)
1.2.3	Mémoire vive	Minimum 1Go
1.2.4	Espace disque	Minimum 1Go disponible
1.2.5	Interfaces réseau	
1.2.6	Logiciels de gestion de jobs batch supportés	Torque / IBM LoadLeveler
1.3	Serveurs de stockage	
1.4	Serveur Web (pour API Web Services)	
1.4.1	Serveur d'applications supporté	L'API Web Services du système Vishnu est fournie en Java EJB, testée sur un serveur JBoss
1.4.2	Système d'exploitation	Debian Linux v5 (Lenny) ou ultérieure
1.4.3	Machine virtuelle Java	Java SDK v1.6.0 ou ultérieure
1.4.4	Serveur JBoss et module WS	JBoss AS 5.1.0 ou ultérieure
1.4.5	Module Web Services	Module JBossWS version Stack CXF Server 3.3.1.GA ou ultérieure
1.4.6	Web services standard	WSDL 1.1
1.5	Serveur de Base de données	
1.5.1	Base de donnée supportée	Le système Vishnu supporte nativement la BDD PostGreSQL version 8.0 ou ultérieure. La conception du système doit autoriser l'ajout de plusieurs interfaces vers d'autres BDD, par exemple Oracle SQL.
1.5.2	Réplication de la base de donnée	Afin de garantir un temps de réponse maximum aux requêtes des machines de la plateforme Vishnu, la base de donnée sera éventuellement répliquée selon une architecture maître-esclave (écriture sur un seul serveur, lecture sur plusieurs serveurs).
2.	Logiciels requis	

ID	Name	Text
2.1	SSH	serveur ssh et client ssh
2.2	CORBA	OmniORB v4.1.4 minimum
2.3	Librairie BOOST C++	v1.35 minimum
2.4	Librairies des Batch schedulers	Torque API / LoadLeveler API
2.5	Logiciels de transferts de fichiers	RSync v3.0 ou ultérieure
2.6	SysFera-DS	Une version de SysFera-DS supérieure ou égale à 2.6 et strictement inférieure à la version 3.0 doit être disponible
3.	Compilation	
3.1	CMake	CMake v2.6 minimum
3.2	GCC	GCC v3.4.3 minimum

4.2 Glossaire

- TODO

Chapter 5

Installation et maintenance

5.1 Tableau des Exigences

ID	Name	Text
1.	Packaging	
1.1	Sources	Récupérer les sources au format .tar.gz pour pouvoir compiler et installer VISHNU
1.2	Paquet debian	Récupérer le paquet debian correspondant à VISHNU
2.	Configuration	
2.1	omniORB	Avoir un fichier de configuration de omniORB
2.2	Port	Utiliser des ports libres sur la machine, notamment pour déployer les forwarders
2.3	VISHNU	Avoir des fichiers de configuration pour les agents, SeDs, forwarders(ils servent à passer les firewalls)
2.4	Log	Avoir le log service de DIET d'installé (module indépendant)
3.	Déploiement	VISHNU sera déployé manuellement en lançant un à un les éléments de l'architecture, en commençant depuis la racine vers les feuilles de l'arbre formé par la hiérarchie. Le module de log service doit être lancé avant VISHNU pour qu'il note bien chaque inscription et événement intervenant
3.1	Flags de compilation	VISHNU devra avoir les modules DAGDA, LOGSERVICE, DYNAMICS, CORI d'activés
3.2	Variable d'environnement	Le Système a besoin de variables d'environnement
3.2.1	Configuration VISHNU	VISHNU_CONF_DIR, contenant le fichier de configuration utilise par VISHNU
3.2.2	SSH_KEYPATH	Chez le client, la clé SSH du serveur (chemin complet sur la clé publique) afin d'identifier le client

5.2 Glossaire

- TODO