

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

Suivi des versions-révisions et des validations du document.			
<p>Ce document annule et remplace tout document diffusé de version-révision antérieure.</p> <p>Dès réception de ce document, les destinataires ont pour obligation de détruire les versions-révisions antérieures, toutes les copies, et de les remplacer par cette version.</p> <p>Si les versions-révisions antérieures sont conservées pour mémoire, les destinataires doivent s'assurer qu'elles ne peuvent être confondues avec cette présente version-révision dans leur usage courant.</p>			
Version.	Date.	Auteurs.	Création, modification ou validation.
A	22 oct. 2003.	JPD.	Création.

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

1 Tables

1.1 Table des matières

1	Tables	2
1.1	Table des matières	2
1.2	Table des illustrations	2
2	Références	3
2.1	Glossaire	3
2.2	Ressources	3
3	Introduction	4
3.1	Objet du document	4
3.2	Audience	4
3.3	Pré-requis	4
4	Cas de test	5
5	Composants matériels	6
5.1	Matériels	6
5.2	Logiciels de base	6
6	Composants logiciels	7
6.1	Modèle métier	7
6.2	Interfaces homme-machine	7
6.3	Editions	8
6.4	Appels de service	8
6.5	Echanges de données	8
6.6	Autres programmes batch	9

1.2 Table des illustrations

Tableau 1	Exemple de cas de test	5
Tableau 2	Exemple de test d'un matériel	6
Tableau 3	Exemple de test d'un logiciel de base	6
Tableau 4	Exemple de test d'une méthode d'une entité	7
Tableau 5	Exemple de test d'une action d'une interface homme-machine	7
Tableau 6	Exemple de test d'une règle de contrôle de validité d'une interface homme-machine	8
Tableau 7	Exemple de test d'exécution d'une édition	8
Tableau 8	Exemple de test d'un prototype d'un appel de service	8
Tableau 9	Exemple de test d'une exécution d'un échange de données	9
Tableau 10	Exemple de test d'exécution d'un programme batch	9

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

2 Références

2.1 Glossaire

Liste des définitions des termes employés.	
Ce tableau recense tous les termes, les concepts particuliers ainsi que les abréviations employés dans ce document.	
Terme, concept, abrégé.	Définition du terme, du concept ou de l'abréviation.
Cas de test	Voir page 5.

2.2 Ressources

Liste des documents applicables et en référence.		
<p>Un document est applicable à partir du moment où son contenu est validé et que l'activité ou le projet fait partie de son périmètre d'application. Il est obligatoire d'appliquer son contenu.</p> <p>Un document est en référence à partir du moment où son contenu n'est pas validé ou que l'activité ou le projet ne fait partie de son périmètre d'application. Il est recommandé d'appliquer son contenu mais cela n'est pas obligatoire.</p> <p>Un document applicable est indiqué par A1, A2, A3, etc. Un document en référence est indiqué par R1, R2, R3, etc.</p>		
Index.	Nom du document.	Commentaire.
A1	UpComp-Plan Qualité-000005	Méthode documentaire.
A2	UpComp-Plan Qualité-000006	Processus de management de projet.
A3	UpComp-Plan Qualité-000050	Modèle de spécification de test des composants.
R4	UpComp-Meilleure pratique-000019	Spécification de validité des composants.

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

3 Introduction

3.1 Objet du document

L'objet de ce document est de définir la méthode de spécification de test des composants du système pour vérifier la prise en compte des fonctionnalités et des contrôles de validité entrant dans leur périmètre technique. L'objectif est double :

- Harmoniser les tests des systèmes techniques.
- Faciliter la compréhension des tests techniques.

Le respect de ces règles fondamentales énoncées ci-après contribue à la réussite du projet où chaque intervenant trouvera satisfaction au regard de sa contribution.

Un test technique se définit par des cas de test déclinés pour les différents cas techniques énoncés dans la **Spécification technique des composants matériels**, les **Spécifications techniques d'un composant logiciel** ou les **Spécifications techniques d'interface**.

Ce document présente comment écrire ces cas de test à l'aide d'exemples. Le **Modèle de spécification de validité des composants** [A3] comporte des propriétés :

- **NomDuProjet** pour le nom du projet.
- **NomDuModule** pour le nom du module.

Un exemple de l'application de la méthode est la meilleure pratique **Spécification de validité des composants** [R4].

3.2 Audience

Ce document s'adresse aux :

- **Directeurs de projets et chefs de projets.**
Pour la compréhension du test technique d'un composant.
- **Ingénieurs de développement.**
Pour la compréhension du test technique précis.

Pour aider ces personnes à remplir le document **Spécification de validité des composants**, leur manager et la cellule de support projet se tiennent à leur disposition.

3.3 Pré-requis

Le pré-requis est la connaissance de la **Méthode documentaire** [A1] et le **Processus de management de projet** [A2].

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

4 Cas de test

Un **cas de test** est une simulation de fonctionnement d'une fonction d'un composant métier se déroulant dans un contexte particulier. Un cas de test comportant quelques étapes et un résultat attendu.

Voici un exemple de cas de test :

C4_2	Recherche et affichage d'une adresse.
Entrée.	Base de données de test contenant des adresses.
Etape.	Description.
Rechercher.	Rechercher une adresse par le nom de la rue.
Rechercher 2.	Rechercher une adresse par le code postal.
Rechercher 3.	Rechercher une adresse par le nom de la ville.
Sortie.	Les adresses sont trouvées.

Tableau 1 – Exemple de cas de test

Voici le contenu de chaque colonne de ce tableau :

- **Description du cas de test.**
Cette description présente l'objet du cas de test.
- **Entrée du cas de test.**
Cette description présente les pré-requis pour exécuter le cas de test.
- **Identifiant du cas de test.**
Ce terme unique est généralement composé d'une séquence pour la nature du cas et d'une seconde pour le cas dans une nature donnée.
- **Une liste d'étapes :**
Les étapes sont listées dans l'ordre d'exécution. Pour chaque étape, il y a :
 - **Description de l'étape.**
Cette description présente les actions précises à effectuer au cours de cette étape.
 - **Identifiant de l'étape.**
Ce terme unique est généralement composé d'un verbe à l'infinitif, éventuellement suivi d'une séquence si l'action est présente plusieurs fois dans la liste des étapes.
- **Sortie du cas de test.**
Cette description présente les conditions de validation du cas de test.

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

5 Composants matériels

5.1 Matériels

Pour chaque caractéristique critique ou majeure ne faisant pas partie de l'état de l'art de chaque matériel, un cas de test est écrit.

Voici un exemple pour la rapidité du **DAT** :

C2_1	La vitesse d'écriture du <i>DAT</i> pour réaliser la sauvegarde du serveur en moins d'un quart d'heure.
Entrée.	Base de données d'exploitation de taille nominale.
Etape.	Description.
Arrêter.	Arrêter la base.
Lancer.	Lancer une sauvegarde complète.
Sortie.	La durée de la sauvegarde doit durée moins de 15 minutes.

Tableau 2 – Exemple de test d'un matériel

5.2 Logiciels de base

Pour chaque caractéristique critique ou majeure ne faisant pas partie de l'état de l'art de chaque logiciel de base, un cas de test est écrit.

Voici un exemple pour la sauvegarde base ouverte de **MySql** :

C3_1	Sauvegarde de la base de données sans l'arrêter.
Entrée.	Base de données d'exploitation de taille nominale.
Etape.	Description.
Basculer.	Basculer les <i>tablespaces</i> de la base de données dans l'état <i>read-only</i> .
Sauver.	Faire une sauvegarde.
Consulter.	Consulter les fichiers journaux de la base de données et de la sauvegarde.
Sortie.	Il ne doit y avoir aucune erreur de signalée.

Tableau 3 – Exemple de test d'un logiciel de base

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

6 Composants logiciels

6.1 Modèle métier

Chaque méthode de chaque entité comporte un ou plusieurs cas de test.

Voici un exemple pour la méthode **Lire** de l'entité **Adresse** :

C4_2	Recherche et affichage d'une adresse.
Entrée.	Base de données de test contenant des adresses.
Etape.	Description.
Rechercher.	Rechercher une adresse par le nom de la rue.
Rechercher 2.	Rechercher une adresse par le code postal.
Rechercher 3.	Rechercher une adresse par le nom de la ville.
Sortie.	Les adresses sont trouvées.

Tableau 4 – Exemple de test d'une méthode d'une entité

6.2 Interfaces homme-machine

Chaque action de chaque interface homme-machine comporte un ou plusieurs cas de test.

Voici un exemple pour l'action **Modifier** de l'interface homme-machine **Adresse** :

C7_4	Modification d'une adresse.
Entrée.	Base de données de test contenant des adresses.
Etape.	Description.
Sélectionner.	Sélectionner une adresse dans la liste.
Modifier.	Modifier les champs de l'adresse.
Actionner.	Actionner le bouton Modifier .
Sortie.	L'adresse est modifiée.

Tableau 5 – Exemple de test d'une action d'une interface homme-machine

Chaque règle de contrôle de validité de chaque interface homme-machine comporte un ou plusieurs cas de test.

Voici un exemple pour l'action **Code postal obligatoire** de l'entité **Adresse** :

C8_1	Le code postal est obligatoire.
Entrée.	Base de données de test contenant des adresses.
Etape.	Description.
Sélectionner.	Sélectionner une adresse.
Supprimer.	Supprimer le code postal.
Actionner.	Actionner le bouton Modifier .

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

Sortie.	L'application refuse de modifier l'adresse.
----------------	--

Tableau 6 – Exemple de test d'une règle de contrôle de validité d'une interface homme-machine

6.3 Editions

Chaque mode d'exécution possible de chaque édition comporte un ou plusieurs cas de test.

Voici un exemple pour l'action **Sans paramètre** de l'édition **Liste de contacts** :

C12_1	Avoir la liste complète des contacts.
Entrée.	Base de données de test contenant des contacts.
Etape.	Description.
Lancer.	Lancer l'édition sans renseigner le paramètre IdentifiantContact .
Sortie.	La liste complète des contacts s'affiche.

Tableau 7 – Exemple de test d'exécution d'une édition

Chaque règle de contrôle de validité de chaque édition comporte un ou plusieurs cas de test.

6.4 Appels de service

Chaque prototype de chaque appel de service comporte un ou plusieurs cas de test.

Voici un exemple pour le prototype **ConsulterAdresse** pour **Corba** de l'appel de service **Consultation des adresses** :

C13_1	Consultation d'une adresse via Corba.
Entrée.	Base de données de test contenant des contacts.
Etape.	Description.
Ecrire.	Ecrire un client Corba utilisant le service IContacts.ChercherContact .
Sélectionner.	Sélectionner un contact.
Consulter.	Consulter son adresse via le client Corba .
Sortie.	L'adresse est correcte.

Tableau 8 – Exemple de test d'un prototype d'un appel de service

Chaque règle de contrôle de validité de chaque appel de service comporte un ou plusieurs cas de test.

6.5 Echanges de données

Chaque mode d'exécution de chaque échange de données comporte un ou plusieurs cas de test.

Voici un exemple pour l'exécution avec un mauvais paramètre de l'échange de données **Charger les contacts** :

C14_1	Exécution avec le paramètre Fichier mal renseigné.
Entrée.	Base de données de test contenant des contacts.
Etape.	Description.

	Méthode de spécification de validité des composants	Date rédaction : 17 novembre 2003.
	Diffusion restreinte	Date validation :
Référence : UpComp-Plan Qualité-000050-A Méthode de spécification de validité des composants.doc		

Lancer.	Lancer le chargement avec le paramètre <i>Fichier</i> mal renseigné..
Sortie.	Il y a un message signalant l'erreur dans le fichier journal.

Tableau 9 – Exemple de test d'une exécution d'un échange de données

Chaque règle de contrôle de validité de chaque échange de données comporte un ou plusieurs cas de test.

6.6 Autres programmes batch

Chaque mode d'exécution de chaque programme batch comporte un ou plusieurs cas de test.

Voici un exemple pour l'exécution *Exportation* du programme batch *Exportation / Importation* :

C16_1	Exportation de la base de données.
Entrée.	Base de données de test contenant des contacts et des sociétés.
Etape.	Description.
Exporter.	Exporter la base de données.
Sortie.	L'opération se déroule correctement.

Tableau 10 – Exemple de test d'exécution d'un programme batch

Chaque règle de contrôle de validité de chaque échange de données comporte un ou plusieurs cas de test.

Fin de document