Cobuto RM

(Release Management)

Manuel Utilisateur

Statut : Public

Auteur : Marte

Sommaire

[1 Introduction 5](#_Toc433357342)

[1.1 But du document 5](#_Toc433357343)

[1.2 Présentation générale 5](#_Toc433357344)

[1.3 Cadre technique 5](#_Toc433357345)

[2 Définitions des notions dans Cobuto RM 6](#_Toc433357346)

[2.1 Projet de release 6](#_Toc433357347)

[2.2 Palier de publication 6](#_Toc433357348)

[2.3 Release 6](#_Toc433357349)

[3 Introduction à l’admnistration des référentiels 7](#_Toc433357350)

[3.1 Modèle Conceptuel de données de la partie administration des référentiels 7](#_Toc433357351)

[3.2 Administration des référentiels 7](#_Toc433357352)

[3.2.1 Création d’une application. 7](#_Toc433357353)

[3.2.2 Création des composants d’une application. 8](#_Toc433357354)

[3.3 Administration de la liste des exigences et des changements 8](#_Toc433357355)

[4 Description de la gestion des utilisateurs et des profils 10](#_Toc433357356)

[4.1 Modèle Conceptuel de données de la partie Gestion des utilisateurs et des profils 10](#_Toc433357357)

[4.2 Administration de la partie Gestion des utilisateurs et des profils 10](#_Toc433357358)

[4.3 Gestion des droits et authentification 10](#_Toc433357359)

[4.4 Règles de gestion 10](#_Toc433357360)

[4.5 Les rôles mis en place 11](#_Toc433357361)

[4.6 Les profils à mettre en place 11](#_Toc433357362)

[5 Introduction à l’urbanisation des applications 12](#_Toc433357363)

[5.1 Modèle versionné des applications 12](#_Toc433357364)

[5.2 Préciser les versions déclaratives des applications, Unités de Packaging et Unités d’Installation 12](#_Toc433357365)

[5.3 Structure de versionnement hiérarchique de l’application et de ses composants 12](#_Toc433357366)

[5.4 Structure de versionnement de l’application et de ses composants dans le temps 12](#_Toc433357367)

[5.5 Contexte de versionnement de l’application et de ses composants 12](#_Toc433357368)

[5.6 Une urbanisation évolutive 13](#_Toc433357369)

[5.7 Cas pratique d’utilisation 13](#_Toc433357370)

[6 Généralités sur les projets de release et les releases 14](#_Toc433357371)

[6.1 Création initiale du Projet de Release 14](#_Toc433357372)

[6.2 Création de la Release 14](#_Toc433357373)

[6.3 Prédécesseurs et successeurs 14](#_Toc433357374)

[6.4 Liens de la Release avec les packages et les environnements 14](#_Toc433357375)

[6.5 Publication de la Release 14](#_Toc433357376)

[6.6 MCD  Projet de Release et Release (vue avec les changements et exigences) 15](#_Toc433357377)

[6.7 MCD  Projet de Release et Release (vue avec les packages et les environnements) 15](#_Toc433357378)

[7 Introduction au modèle livraison - packages 16](#_Toc433357379)

[7.1 Description de l’objet livraison 16](#_Toc433357380)

[7.2 Modèle Conceptuel de données de l’objet livraison 16](#_Toc433357381)

[7.3 Maille du Package en entrée de Cobuto RM 16](#_Toc433357382)

[7.4 Indicateurs de remplissage des packages 16](#_Toc433357383)

[7.4.1 Remplissage package « Complet » 16](#_Toc433357384)

[7.4.2 Remplissage package « Complet dans le projet de Release » 17](#_Toc433357385)

[7.4.3 Remplissage package « Partiel » 17](#_Toc433357386)

[8 Gestion des tableaux de bords 18](#_Toc433357387)

[8.1 Paramétrage des processus et activités de tests 18](#_Toc433357388)

[8.2 Création du palier de publication. Exemple de cas standard : VSI 18](#_Toc433357389)

[8.3 Accès au tableau de bords des paliers de publication 18](#_Toc433357390)

[8.4 Visualisation du palier de publication. Exemple de cas standard : VSI 19](#_Toc433357391)

[8.5 Accès aux tableaux de bords du palier de publication 20](#_Toc433357392)

[8.6 Visualisation du palier de publication (vue avec packages) 20](#_Toc433357393)

[8.7 MCD des processus et activités (et lien avec les projets de releases) 21](#_Toc433357394)

[9 Visualisation du projet de release 22](#_Toc433357395)

[9.1 Vue générale du projet de release 22](#_Toc433357396)

[9.2 Onglet Périmètre du projet de release 22](#_Toc433357397)

[9.3 Onglet Exigences du projet de release 22](#_Toc433357398)

[9.4 Onglet Livraisons du projet de release 23](#_Toc433357399)

[9.5 Onglet Releases du projet de release 25](#_Toc433357400)

[10 Visualisation des releases publiées d’une application 26](#_Toc433357401)

[11 Use case de création du projet de release 27](#_Toc433357402)

[11.1 Première étape : saisie des informations préalables 27](#_Toc433357403)

[11.2 Deuxième étape : saisie des exigences 28](#_Toc433357404)

[11.3 Troisième étape : cocher les Unités d’Installation modifiées dans votre projet de release 28](#_Toc433357405)

[11.4 Quatrième étape (optionnelle) : Sélectionner le palier de publication 28](#_Toc433357406)

[11.5 Récapitulatif des opérations 29](#_Toc433357407)

[12 Use case de livraison 30](#_Toc433357408)

[12.1 Accès à la fonction de livraison 30](#_Toc433357409)

[12.2 Etape 1 : Saisie des exigences 30](#_Toc433357410)

[12.3 Etape 2 : Saisie des commentaires 31](#_Toc433357411)

[12.4 Etape 3 : Saisie des Unités d’Installation 31](#_Toc433357412)

[12.5 Etape 4 : Saisie des packages 31](#_Toc433357413)

[12.6 Etape 5 : Récapitulatif de la livraison 32](#_Toc433357414)

[13 Use Case de création d’un package fusionné 33](#_Toc433357415)

[13.1 Accéder à l’assistant de création du package fusionné 33](#_Toc433357416)

[13.2 Assistant de création d’un package fusionné 34](#_Toc433357417)

[13.3 Constater le package fusionné dans l’onglet Livraisons du projet de Release 35](#_Toc433357418)

[13.4 Gestion des cas d’erreurs 35](#_Toc433357419)

[14 Use Case de création de la Release 37](#_Toc433357420)

[14.1 Accéder à l’assistant de création de la release 37](#_Toc433357421)

[14.2 Assistant de création de la Release (une seule et unique étape) 37](#_Toc433357422)

[15 Use Case de demande d’installation sur les environnements 39](#_Toc433357423)

[15.1 Accès à la fonction de demande d’installation 39](#_Toc433357424)

[15.2 Première étape : constater les informations de la release 40](#_Toc433357425)

[15.3 Deuxième étape : saisie de l’environnement 40](#_Toc433357426)

[15.4 Troisième étape : constatation des packages à installer 40](#_Toc433357427)

[15.5 Quatrième étape : récapitulatif 42](#_Toc433357428)

[15.6 Gestion des cas d’erreurs 42](#_Toc433357429)

[16 Use Case d’aquittement d’une installation sur un environnement 44](#_Toc433357430)

[16.1 Accès aux installations en cours 44](#_Toc433357431)

[16.2 Constatation des installations en cours à réaliser 44](#_Toc433357432)

[16.3 Acquittement d’une installation 45](#_Toc433357433)

[16.4 Mise à jour de l’état des packages si erreur 46](#_Toc433357434)

[17 Production des bons d’installation 47](#_Toc433357435)

[17.1 MCD contenant le lien vers l’objet Environnement 47](#_Toc433357436)

[17.2 Indicateurs de suivi des packages de livraison sur les environnements 47](#_Toc433357437)

[17.3 Production du bon d’installation VSI 47](#_Toc433357438)

# Introduction

## But du document

L’objet de ce document est de définir le manuel utilisateur détaillé de l’application Cobuto RM.

Ce document a pour but de décrire précisément :

* L’ensemble des fonctionnalités de l’application.
* Les objets manipulés, leurs buts et leurs principes de fonctionnement.
* Les écrans utilisateurs mettant en œuvre les fonctionnalités de l’application.

## Présentation générale

L’application Cobuto RM a pour but de répondre aux besoins de maîtrise des développements, des livraisons et des déploiements des versions successives du SI au sein de l’entreprise. En interaction avec l’outil Subversion, Cobuto RM répond à plusieurs besoins majeurs :

* Le suivi et la synthèse des activités de livraison,
* L’identification des livrables d’une version du SI,
* La mise au point de la dernière bonne version des livraisons successives par agrégation,
* La mise à disposition de l’ensemble des éléments nécessaires au déploiement,
* Le suivi et le pilotage de la version du SI et des déploiements sur les environnements,
* Le suivi de la gestion des changements.
* Le suivi de la gestion des exigences.

## Cadre technique

L’application Cobuto RM repose sur une architecture client-serveur en JAVA.

Le serveur applicatif répond aux spécificités suivantes :

* Système Unix / Linux
* JAVA 1.6
* Serveur Apache / Tomcat
* PERL version 5.10
* Client Subversion
* MySQL 5.5

# Définitions des notions dans Cobuto RM

## Projet de release

Objectif de mise en production à une date cible pour une application d’un ensemble d’exigences (évolutives et correctives).

La release applicative qui ira en production est une des dernières d’une série de tentatives nécessaires à la mise au point du projet de release.

Le projet de release, associé à un ensemble d’exigences, impacte un ensemble de composants de l’application.

Les composants applicatifs sont livrés sous forme de package.

Le projet de release est clôturé au plus tard au moment du déploiement de sa release éligible ou à tout moment en cas d’abandon.

## Palier de publication

Le palier de publication regroupe plusieurs projets de release qui partagent la même date cible de mise en production, le même cycle de vie et le même calendrier d’activité.

Chaque palier utilise des environnements qui lui sont alloués pour les périodes définies dans son calendrier.

Ces environnements permettent d’exécuter des activités, par exemple d’intégration, d’homologation, etc…

Le palier est créé au plus tard lors de la réservation des environnements.

Le palier est clôturé après le déploiement et la VSR de l’ensemble des releases éligibles ou à tout moment en cas d’abandon.

Le palier est associé à un ou plusieurs changements.

## Release

Ensemble des packages implémentant une version d’application satisfaisant un ensemble d’exigences. Les releases sont produites par les équipes de développement.

Une release peut être complète ou partielle.

Une release complète ou release est nécessaire et suffisante pour installer une version complète de l’application, comme par exemple les « service pack » de Microsoft.

Une release partielle ou patch permet d’installer une ou plusieurs modifications sur une application, comme par exemple les « corrective patch ». Une release partielle s’applique sur une release complète qui représente un prérequis à son installation. D’autres releases partielles peuvent également être nécessaires comme prérequis d’installation, en plus de la release complète.

Une release (complète) contient tous les packages nécessaires et suffisants pour installer l’application.

Un patch contient des packages éventuellement partiels dont l’installation permet de modifier une release complète déjà installée.

La liste des packages constituant une release ou un patch n’est pas modifiable. Elle est figée à la création de la release.

L’application est déployée par release et par patch.

Une version d’application est identifiée par une release et un ensemble de patchs.

# Introduction à l’admnistration des référentiels

## Modèle Conceptuel de données de la partie administration des référentiels



## Administration des référentiels

### Création d’une application.

Depuis l’outil Cobuto RM, il est possible d’administrer la création d’une nouvelle application :

* + - Créer une nouvelle application à partir d’une référence interne à l’Entreprise.
      * Création de l’objet Application à partir des attributs donnés par l’entreprise.

Note : Le code interne de l’application est important à respecter car il permet de versionner les objets.

* + - * L’action de créer une nouvelle application se fait à partir de l’interface prévue à cette effet.
      * L’application est initialisée ensuite dans Cobuto RM avec une Release initiale par défaut.
    - Modifier le détail d’une application.
    - Supprimer une application : cette opération supprime définitivement l’historique de cette application.

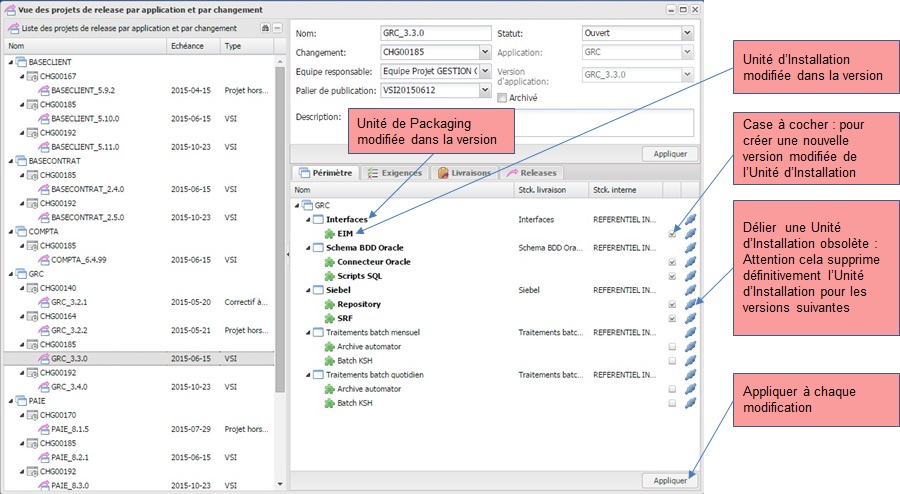
*☞ Reportez-vous au manuel d’administration pour plus de détails sur la création d’une nouvelle application.*

### Création des composants d’une application.

Toutes les applications doivent avoir une structure à trois niveaux :

* + - Application (nom en rapport avec le référentiel des applications)
    - Unité de Packaging (regroupement d’Unités d’Installation)
    - Unité d’Installation

Depuis l’outil Cobuto RM, il est possible d’administrer la création de nouvelles Unité de Packaging et de nouvelles Unités d’Installations en GCL. Ces actions ne sont possibles que dans le cadre d’un Projet de Release. Elles sont réalisables dans l’onglet « Périmètre » du Projet de Release :



Remarque cas particulier : lors de la création d’une nouvelle application un Projet de Release initial est automatiquement créé. L’initialisation des Unités de Packaging et des Unités d’Installation doit alors être réalisée sur ce Projet de Release initial.

*☞ Reportez-vous au manuel d’administration pour plus de détails sur la création de composants.*

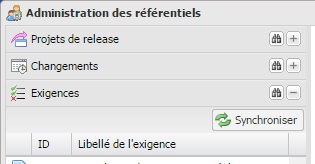
## Administration de la liste des exigences et des changements

Il existe deux manières non-exclusives de traiter la liste des exigences et des changements dans Cobuto RM :

* + - Utiliser l’outil Cobuto RM comme référentiel maître des exigences et/ou des changements dans votre Entreprise. Dans ce cas, les exigences et/ou changements peuvent être créés directement dans la partie administration du référentiel.
    - Choisir d’importer les exigences et/ou changements dans votre référentiel Cobuto RM. Dans ce cas il existe plusieurs types d’import :
      * Import depuis la consultation d’une base tiers (SQL Server, MySQL, Oracle) avec paramétrage du mapping des champs,
      * Import CSV avec paramétrage du mapping des champs des exigences et/ou changements.
      * Déclenchement d’un WebService REST HTTP.

Contacter [support@releasemanagement.fr](mailto:support@releasemanagement.fr) pour plus de détails sur cette fonctionnalité.

Le support Cobuto RM pourra ainsi vous mettre à disposition l’interface de votre choix. Une implémentation possible est d’utiliser le bouton synchroniser pour importer les exigences au format CSV comme indiqué sur l’impression d’écran ci-dessous :



# Description de la gestion des utilisateurs et des profils

## Modèle Conceptuel de données de la partie Gestion des utilisateurs et des profils



## Administration de la partie Gestion des utilisateurs et des profils

L’administration de Cobuto RM permet de gérer les utilisateurs de l’application et leurs habilitations. Il est possible de créer de nouveaux utilisateurs et de leur assigner un ou plusieurs profils. Un profil correspond à un niveau d’habilitation de navigation et d’interaction avec l’application.

Il existe un profil spécifique : « Administrateur » qui est le seul à permettre d’accéder à ces fonctionnalités.

Il est de plus possible de gérer les profils, permettant ainsi la création de nouveaux profils, l’application étant toutefois livrée avec un certain nombre de profils préétablis.

## Gestion des droits et authentification

L’application gère une authentification sécurisée basée sur le LDAP Unix. L’application traite alors en interne les droits des utilisateurs, soit en fonction des groupes LDAP de l’utilisateur, soit en fonction du compte LDAP ou local d’un utilisateur déclaré dans l’application.

Les droits sont définis par des rôles regroupés en profils pour simplifier leur assignation, et permettent d’accéder aux différentes fonctions de l’application ou à différentes données. Il est ainsi possible, par exemple, de restreindre l’accès aux informations à certains rôles et en fonction des fonctionnalités dont les utilisateurs ont un droit accès.

Remarque : il n’est pas obligatoire d’utiliser les groupes LDAP. L’authentification se fait alors avec le compte LDAP d’un utilisateur déclaré dans l’application. Il est donc nécessaire de déclarer tous les utilisateurs de l’application.

## Règles de gestion

Cette partie dispose des règles de gestion suivantes :

* + - Un utilisateur est authentifié soit par le LDAP soit par une authentification locale.
    - Un utilisateur local « guest » est fourni dans l’application et aura un accès au bon d’installation.
    - Un compte avec le profil « Administrateur » peut modifier, créer ou supprimer les utilisateurs et les profils. Un utilisateur Administrateur est fourni en standard dans l’application.
    - Un utilisateur ne peut se supprimer lui-même.
    - Un utilisateur identifié héritera automatiquement des profils Cobuto RM correspondant à ses profils.
    - Il est possible d’assigner à un utilisateur spécifique plusieurs profils.

## Les rôles mis en place

Les rôles mis en place sont précisés dans un fichier Excel annexe.

## Les profils à mettre en place

Les profils mis en place sont précisés dans le document Administration.

# Introduction à l’urbanisation des applications

## Modèle versionné des applications

**Cobuto RM**

Niveau 1 : Application

Niveau 2 : Unité de Packaging

Niveau 3 : Unité d’Installation

**Information importante** : l’Unité de Packaging est la maille de versionnement des packages.

## Préciser les versions déclaratives des applications, Unités de Packaging et Unités d’Installation

Lorsqu’un Projet de Release est créé, Cobuto RM donne la possibilité de définir le périmètre de la version d’application qui sera modifiée.

Définir le périmètre consiste à préciser à l’avance l’ensemble des versions de composants impactés de l’application.

Une fois cette opération réalisée, une nouvelle version de composants est créée pour chaque Unité de Packaging et chaque Unité d’Installation étant dans le périmètre. Cette opération est un pré-requis pour accueillir les livraisons. Les livraisons (et relivraisons) restent possibles tant que le Projet de Release n’est pas clôturé par l’Administrateur ou par la personne habilitée à le faire.

## Structure de versionnement hiérarchique de l’application et de ses composants

Les applications, les Unités de Packaging et les Unités d’Installation évoluent en version. Ces versions suivent une logique de versionnement hiérarchique dont le plus haut niveau est l’application et le plus bas niveau est l’Unité d’Installation.

L’Unité de Packaging est la maille de packaging. C’est à ce niveau qu’est versionné le package de livraison. Il regroupe plusieurs versions d’Unités d’Installation qui doivent correspondre généralement à la technologie utilisée pour déployer. La modélisation des Unités d’Installation reste cependant libre.

Une version de l’application correspond donc à un ensemble de versions d’Unités de Packaging qui correspondent eux-mêmes à un ensemble de versions d’Unités d’Installation.

## Structure de versionnement de l’application et de ses composants dans le temps

Les versions ont des successeurs et des prédécesseurs. Une version d’application peut être créée à partir d’une version précédente.

Lorsqu’une version d’application est créée à partir d’une version précédente, cela a pour effet de reporter les dernières versions déclaratives des composants sur la nouvelle version d’application. A ce stade, il n’est pas encore possible d’effectuer des livraisons sur les composants. En effet, il est nécessaire de créer des nouvelles versions de composants dans la nouvelle version d’application avant de livrer.

Ces versions déclaratives seront créées à la volée lorsque, dans l’assistant de création du Projet de Release, l’ensemble des composants du périmètre est coché. Le point de départ de versionnement est toujours la version déclarative de l’application.

Au même titre que la version d’application, les nouvelles versions déclaratives des composants créées sont successeurs des versions déclaratives des composants de la version d’application précédente.

## Contexte de versionnement de l’application et de ses composants

Le contexte de versionnement est toujours le Projet de Release. Il existe deux cas particuliers de versionnement :

* Création d’une nouvelle application. Dans ce cas un Projet de Release initial est automatiquement créée par le système. L’utilisateur doit alors se positionner sur le Projet de Release initial pour initialiser la trame de livraison de l’application (et donc créer les Unités de Packaging et les Unités d’Installation).
* Création d’un nouveau Projet de Release.
  + Un assistant de « création d’un projet de release » aide l’utilisateur dans la création du Projet de Release et dont l’une des étapes est la sélection des composants du périmètre.
  + Si l’urbanisation de l’application a évolué, l’utilisateur doit se positionner à postériori sur le Projet de Release pour ajouter, supprimer ou déplacer des composants.

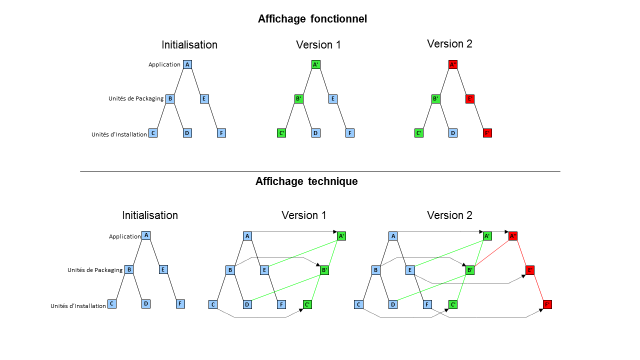
## Une urbanisation évolutive

Ce versionnement va permettre à l’urbanisation d’une application de pouvoir être amenée à évoluer dans le temps. Les projets de release se suivant dans le temps, la structure de découpage de l’application est héritée par défaut de projet en projet. Les caractéristiques de paramétrage de l’application sont également héritées.

A partir du moment où, pour un projet de release particulier, ces propriétés ont été modifiées, les projets de releases qui suivent le projet modifié vont automatiquement hériter des nouvelles propriétés. Les projets ayant précédé le projet de release restent par contre dans leur état d’origine sans qu’ils soient altérés par les modifications.

Les propriétés peuvent aussi bien concerner la structure d’Urbanisation (ajout ou suppression d’une nouvelle Unité de packaging ou d’installation) que le paramétrage (référencement sous Subversion d’une Unité de Packaging, Comportement par défaut du package en entrée partiel, complet, …).

## Cas pratique d’utilisation



# Généralités sur les projets de release et les releases

## Création initiale du Projet de Release

Un nouveau projet de Release se crée à partir :

* D’un projet de Release de référence
* D’une Release de référence au choix étant dans le projet de Release de référence (par défaut proposer celle qui est passée en production sinon proposer la dernière Release disponible).

Un projet de Release est toujours cloisonné par application.

A ce stade il n’existe pas encore de Release mais cette opération permet de créer le projet de release qui servira de cadre pour accueillir les packages de livraison et les futures releases. Ce projet de release est associé à une nouvelle structure versionnée de l’application.

A ce stade, il n’existe pas encore de packages de livraison associés aux composants versionnés.

## Création de la Release

Une fois qu’une ou plusieurs nouvelles livraisons ont eu lieu sur le Projet de Release, il devient possible de créer des Releases.

Créer une Release consiste à sélectionner l’ensemble des packages éligibles satisfaisant les exigences ayant donné lieu au projet et donc candidats pour une Mise en Production.

L’assistant de création de la Release doit pouvoir détecter les nouveaux packages livrés dans le projet de Release pour les proposer d’inclure dans la nouvelle release.

Il est possible désélectionner les packages que l’on ne souhaite plus inclure dans la distribution devant faire l’objet d’une Mise en Production.

Une nouvelle Release se crée obligatoirement :

* Après la création d’un projet de release.
* Après que le projet de Release ait été alimenté par des livraisons.

Le processus de création de la release est itératif la plupart du temps. Généralement plusieurs releases successives sont nécessaires en phase d’intégration et de recette avant que l’on puisse finaliser une ou plusieurs releases candidates à une publication et donc une Mise en Production.

## Prédécesseurs et successeurs

Une Release peut avoir plusieurs successeurs et plusieurs prédécesseurs.

* Si elle a plusieurs prédécesseurs cela signifie qu’elle est le résultat de la fusion de plusieurs releases.
* Si elle a plusieurs successeurs cela signifie qu’elle sert de référence à au moins deux releases en parallèle.

## Liens de la Release avec les packages et les environnements

Une Release concerne toujours une application.

Une Release s’installe sur un environnement.

Elle contient un ou plusieurs Packages qui concourent à sa distribution.

## Publication de la Release

Au cours de la phase de recette, plusieurs itérations de création de releases donnent lieu, en fin de recette, à une Release stable et représentative de ce qui sera installé en production le jour J. Il est encore possible à ce stade que le périmètre technique ou fonctionnel change légèrement, et donc qu’il soit nécessaire de créer de nouvelles releases. Cependant cette Release est jugée suffisamment stable pour être reportée sur les autres environnements de tests (en avance de phase par exemple).

Le Release Manager du projet doit alors publier la release lorsque les tests donnent satisfaction. Les releases contenant des packages KO ou Nfp (Not For Production) ne peuvent pas être publiées.

Remarque : uniquement les releases publiées peuvent donner lieu à une demande d’installation en production.

## MCD  Projet de Release et Release (vue avec les changements et exigences)



## MCD  Projet de Release et Release (vue avec les packages et les environnements)



# Introduction au modèle livraison - packages

## Description de l’objet livraison

Le Livreur projet doit pouvoir effectuer sa livraison dans le cadre d’un Projet de Release. Il doit de plus avoir son périmètre précisé contenant les versions modifiées des Unités de Packaging et des Unités d’Installation qui nécessiteront une livraison.

Le jour de la livraison, le Livreur projet sélectionne les exigences et les composants impactés par sa livraison et précise le ou les n° de sa livraison (si plusieurs). Il déclenche ensuite l’opération de « livraison » dans Cobuto RM.

## Modèle Conceptuel de données de l’objet livraison



## Maille du Package en entrée de Cobuto RM

La maille d’entrée du package est toujours l’Unité de Packaging.

Toutefois si cette maille change au cours du temps, il est possible de réorganiser la structure de versionnement des composants. Ainsi, en cas de réorganisation, les nouvelles livraisons se feront à partir de la nouvelle structure versionnée de l’application, alors que les anciennes livraisons resteront inchangées dans leur mode fonctionnement.

## Indicateurs de remplissage des packages

### Remplissage package « Complet »

Le package est « Complet » y compris en production. Il remplace totalement l’Unité de Packaging lorsqu’il s’installe en production. Par ailleurs, il se suffit à lui-même pour remettre à niveau l’Unité de Packaging dans un environnement. Il se caractérise par les comportements suivants :

* Manière dont il est proposé dans la release : il est pré-coché par défaut dans l’assistant de création de la release s’il n’existe pas de package complet plus récents,
* Manière dont il s’installe sur l’environnement : tous les précédents liens packages – environnement pour la même unité de Packaging sont supprimés.

Exemple de technologies ayant ce type de remplissage : Java (EAR), archive Automator (ordonnanceur), exécutable Windev.

### Remplissage package « Complet dans le projet de Release »

Le package est « Complet dans le projet de release » si le package correspondant à une Unité de Packaging est relivré en annule et remplace lors de chaque livraison pour un projet de Release, mais qu’il reste un delta du point de vue de l’Unité de Packaging modifié en production. Ce type de package n’est pas suffisant pour remettre à niveau un environnement mais se suffit à lui-même dans un projet de release.

* Manière dont il est proposé dans la release : il est pré-coché par défaut dans l’assistant de création de la release s’il n’existe pas de package complet dans le projet de release plus récent,
* Manière dont il s’installe sur l’environnement : tous les liens packages – environnement pour la même unité de Packaging et pour le même projet de release sont supprimés. Les liens packages - environnement des autres projets de releases et pour la même Unité de Packaging ne sont pas modifiés.

### Remplissage package « Partiel »

Le package est « Partiel » si le package correspondant à une Unité de Packaging est relivré partiellement lors de chaque livraison pour un projet de Release

* Manière dont il est proposé dans la release : il est toujours pré-coché par défaut dans l’assistant de création de la release s’il n’existe pas de package complet dans le projet de release plus récents. Il peut venir en complément d’un package complet (ou complet dans le projet de release) plus ancien.
* Manière dont il s’installe sur l’environnement : le lien package – environnement est simplement rajouté sans toucher aux liens packages - environnement déjà existants.

Exemple de technologies ayant ce type de remplissage : script SQL, Fichier batch.

# Gestion des tableaux de bords

## Paramétrage des processus et activités de tests

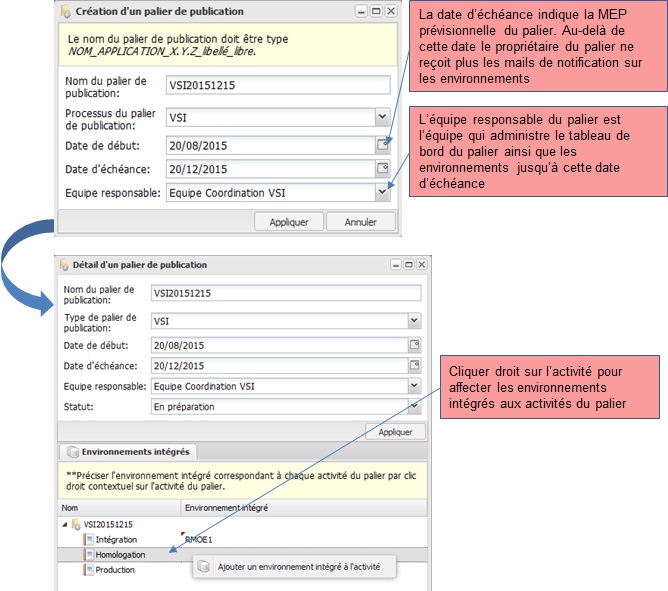
Cobuto RM donne la possibilité de paramétrer les processus logiciels en vigueur dans votre entreprise. Ces processus sont définis par des activités de tests qui s’enchaînent dans le temps (exemple : Intégration, Homologation, Production). Lors de la création d’un palier de publication, le processus correspondant sera sélectionné (exemple : Projet, Evolution, correctif à chaud, …). Cobuto RM donne alors la possibilité de suivre, sur un palier de publication, le déploiement de différentes releases sur les environnements associés aux différentes activités.

Lorsque plusieurs projets de release sont mutualisés au sein d’un même palier de publication, il est possible de suivre simultanément l’état d’avancement de ces différents projets.

*☞ Reportez-vous au manuel d’administration pour plus de détails sur le paramétrage des processus et activités.*

## Création du palier de publication. Exemple de cas standard : VSI

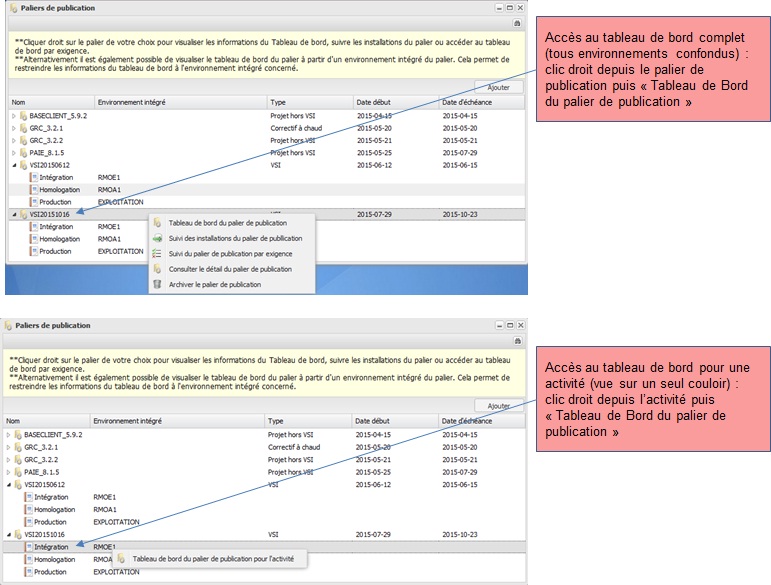
Dans l’exemple de l’impression d’écran ci-dessous, un palier de publication supportant le processus de VSI est créé dans un premier temps. Ce processus est défini par trois activités que l’on souhaite suivre : Intégration, Homologation et Production. Dans un deuxième temps, on définit les environnements intégrés que l’on souhaite associer à chaque activité du palier :



## Accès au tableau de bords des paliers de publication

Les accès se font depuis la Gestion des paliers.

Il est possible d’accéder au tableau de bord complet du palier de publication ou bien uniquement depuis un environnement intégré comme dans l’impression d’écran ci-dessous :



## Visualisation du palier de publication. Exemple de cas standard : VSI

Le tableau bord du palier de publication permet de visualiser toutes les applications de la VSI qui ont un projet de release faisant partie du palier.

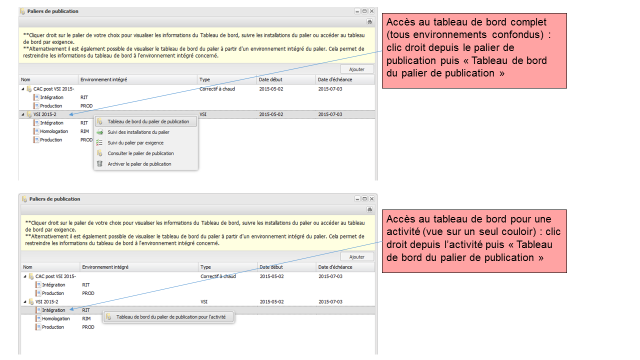
Chaque projet de release est présenté avec :

* + - Tous les projets de release du palier de publication
    - Pour chaque projet de release :
      * La dernière release disponible.
      * La release actuellement en demande d’installation sur l’environnement.
      * La release actuellement installée sur l’environnement.

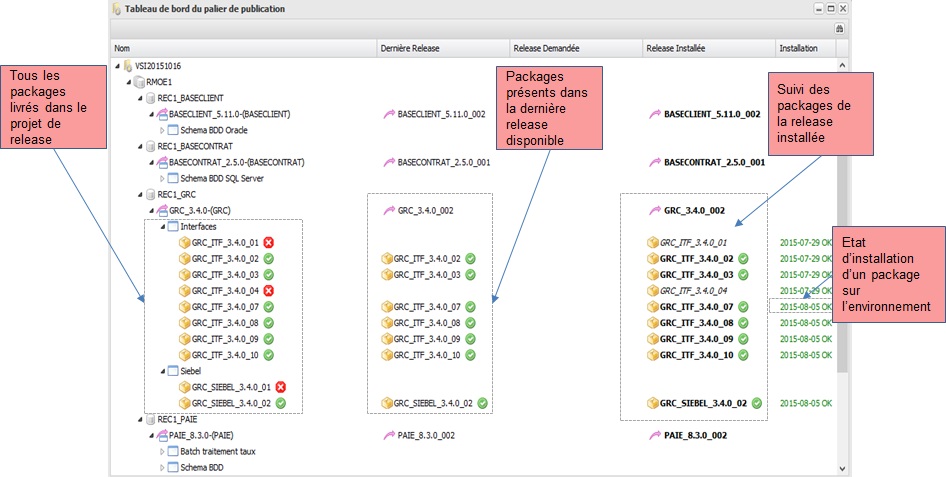
Pour chaque projet de release on visualise l’état des packages :

* + - Tous les packages du projet de release.
    - Les packages n’étant dans aucune release (reconnaissable grâce à une icône « New »).
    - Les packages étant dans la dernière release
    - Les packages étant dans la release en demande d’installation
    - Les packages étant dans la release installée avec leur état d’installation sur l’environnement.
      * Remarque : les packages installés sur l’environnement mais n’étant pas dans la release installée sont affichés en italique.

## Accès aux tableaux de bords du palier de publication



## Visualisation du palier de publication (vue avec packages)



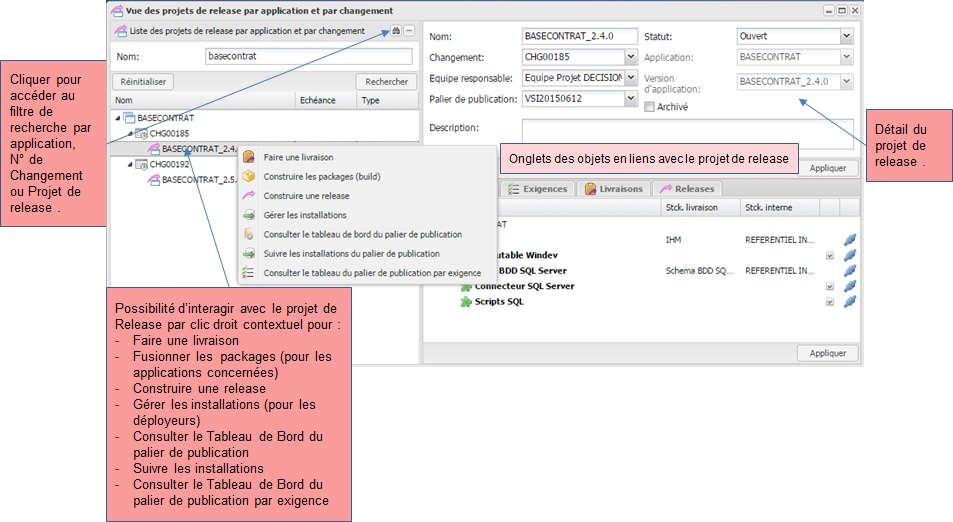
## MCD des processus et activités (et lien avec les projets de releases)



# Visualisation du projet de release

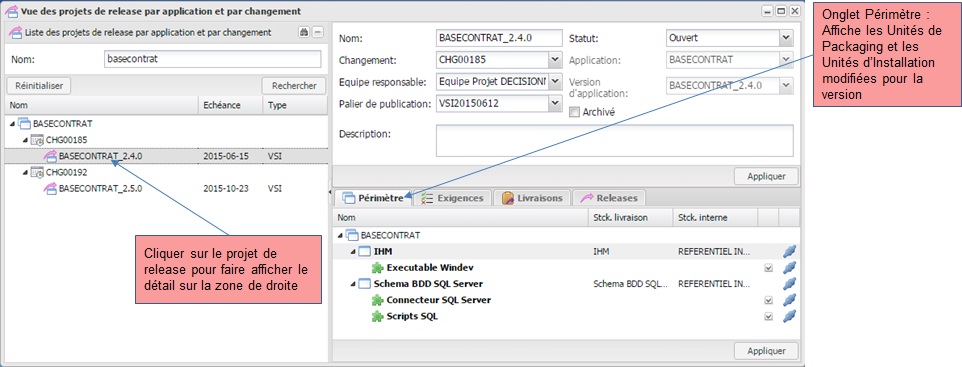
## Vue générale du projet de release

L’accès se fait depuis le menu **Démarrer | Gestion des projets de release**.



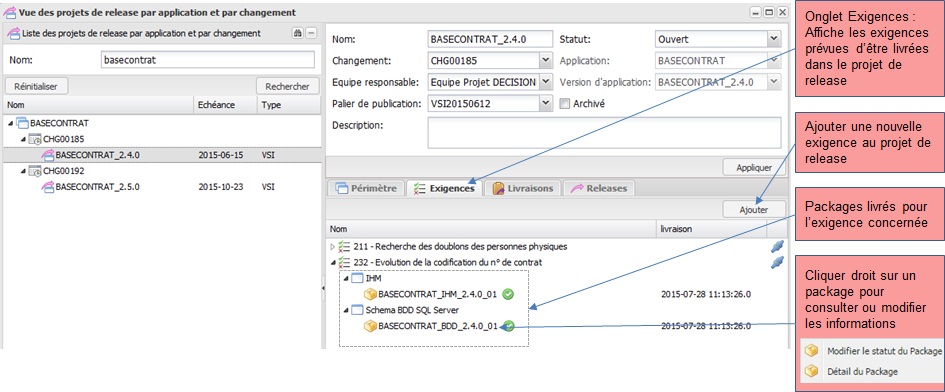
## Onglet Périmètre du projet de release

Le périmètre indique les Unités de Packaging ainsi que les Unités d’Installation modifiées pour la version.



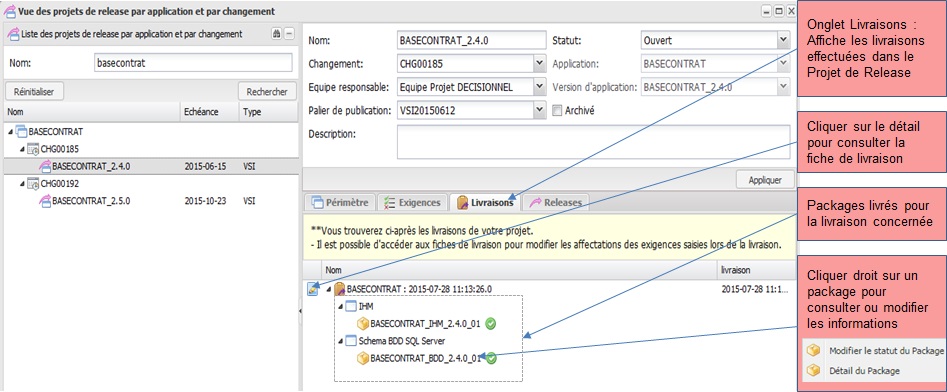
## Onglet Exigences du projet de release

Les exigences affichées sont celles rattachées au projet de release.

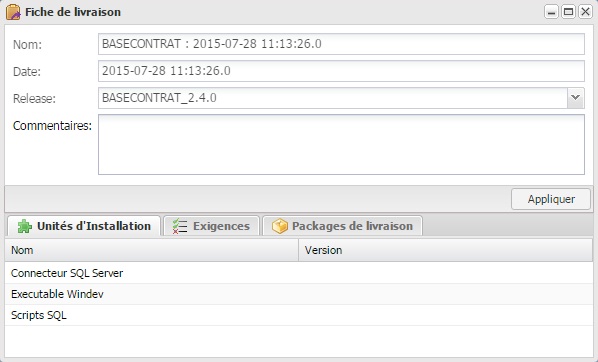


## Onglet Livraisons du projet de release

Les livraisons affichées présente les packages livrés pour ce projet de release.



En cliquant sur détail d’une livraison on obtient la fiche de livraison électronique qui se présente comme suit :



La fiche de livraison comprend :

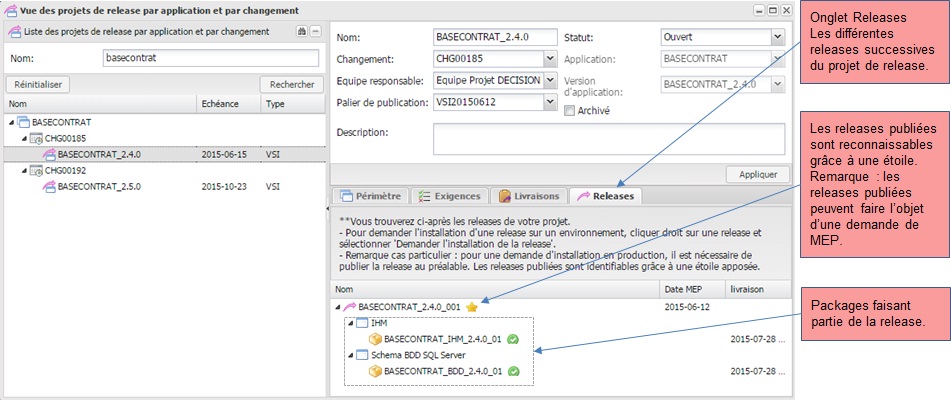
* + - Un commentaire (optionnel) saisi par le livreur
    - Elle comprend également 3 onglets :
      * Unités d’Installation : les Unités d’Installations implémentées par la livraison
      * Exigences : les Exigences implémentées par la livraison
      * Packages de livraison : les n° de packages et les composants modifiés par chaque package de livraison

Il est possible de revenir à tout moment sur cette fiche pour la faire évoluer :

* + - Supprimer des exigences sélectionnées par erreur
    - Ajouter des exigences éventuellement oubliées par le livreur.

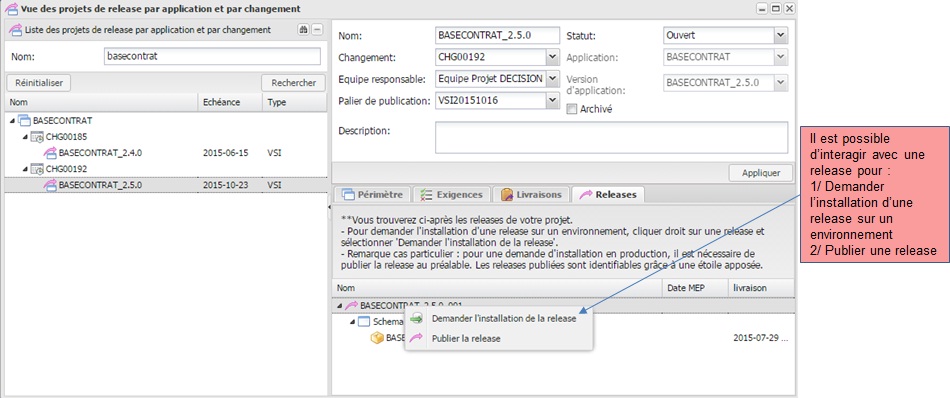
## Onglet Releases du projet de release

Cet onglet affiche les releases successives du projet de release. Les releases publiées sont identifiables grâce à une étoile apposée sur leur nom.



Il est possible d’interagir avec les releases par clic droit contextuel pour :

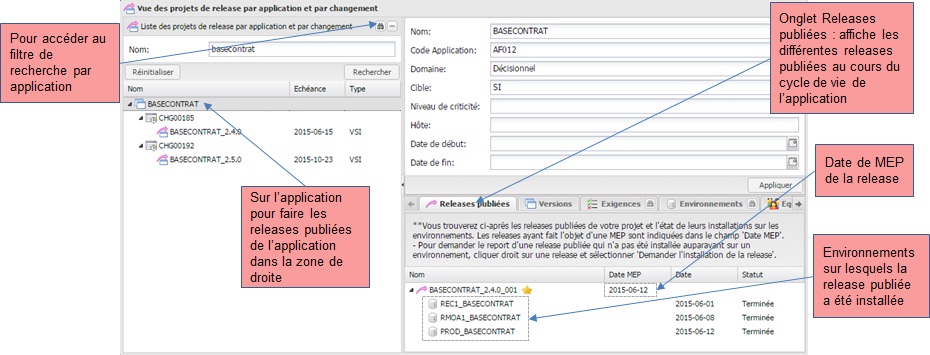
* + - Demander l’installation d’une release sur un environnement.
    - Publier une release (en vue d’en demander l’installation sur un environnement de production).



# Visualisation des releases publiées d’une application

L’accès se fait depuis le menu **Démarrer | Gestion des projets de release**.

En se plaçant sur une application, il est possible de visualiser toutes les releases publiées d’une application au cours de son cycle de vie. Les informations comme sa date de MEP ainsi que son installation sur les différents environnements de Recette et de Production sont affichées :

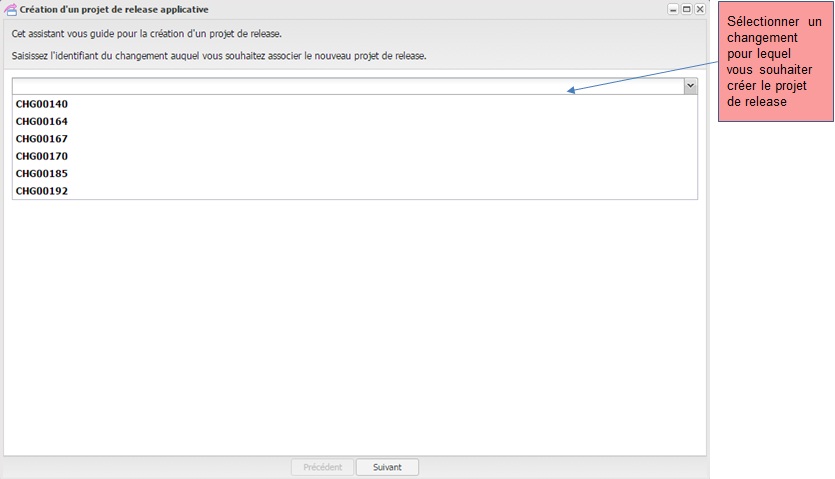


# Use case de création du projet de release

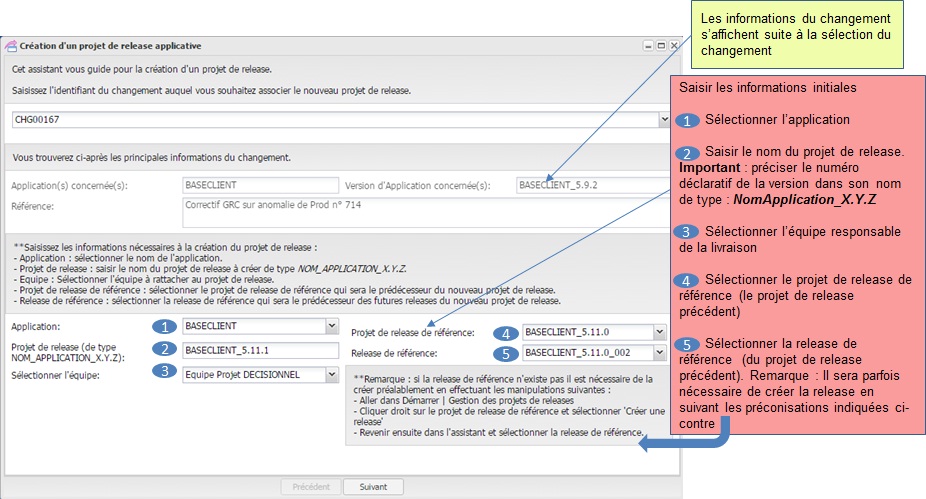
L’accès se fait depuis le menu **Démarrer | Création d’un projet de release**.

## Première étape : saisie des informations préalables

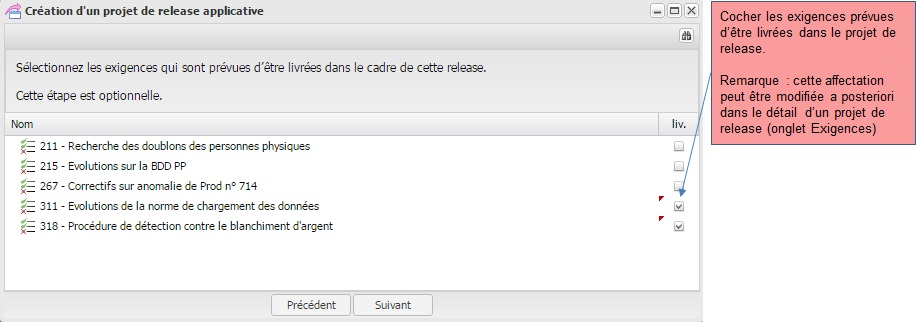
**Sélection du numéro de changement :**



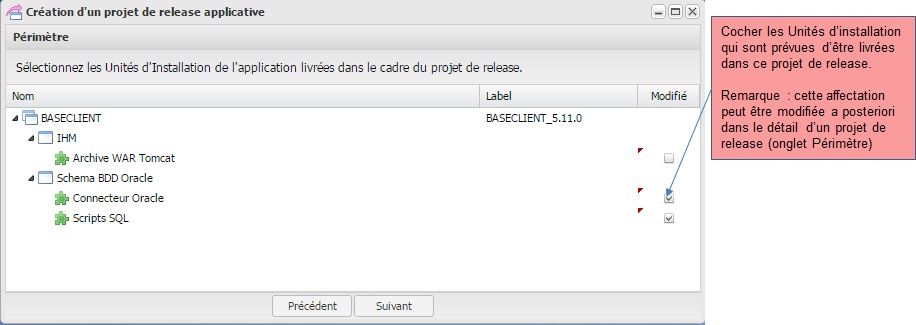
**Saisie des informations préalables :**



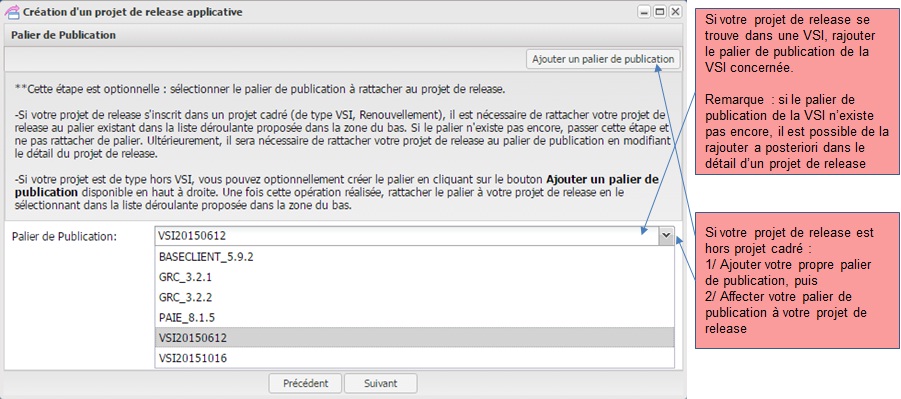
## Deuxième étape : saisie des exigences



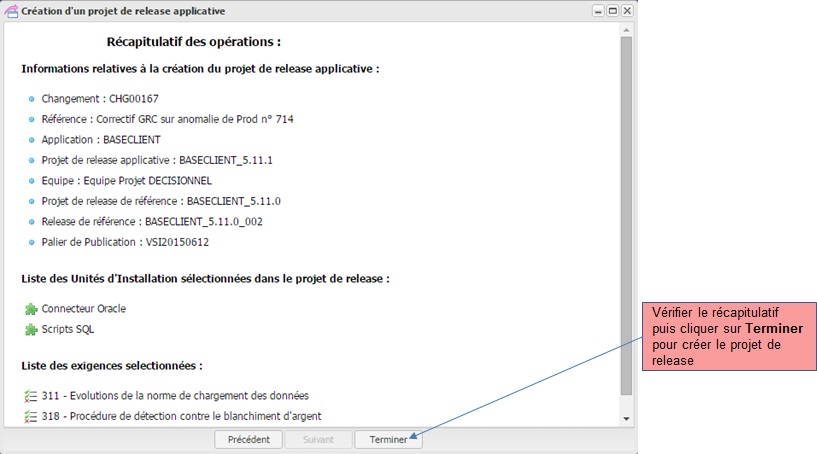
## Troisième étape : cocher les Unités d’Installation modifiées dans votre projet de release



## Quatrième étape (optionnelle) : Sélectionner le palier de publication



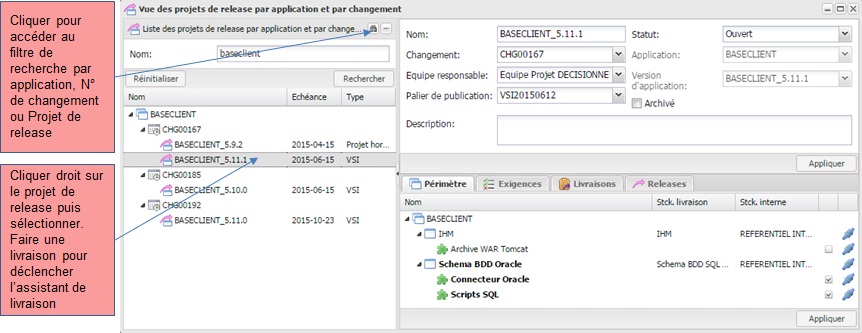
## Récapitulatif des opérations



# Use case de livraison

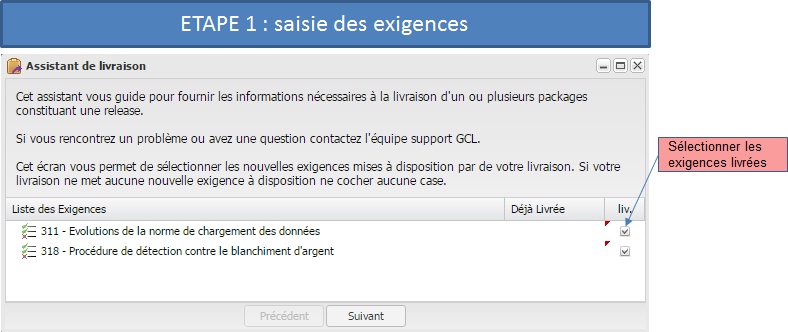
## Accès à la fonction de livraison

L’accès se fait depuis le menu **Démarrer | Création d’un projet de release**. Se positionner alors sur le projet de release concerné et déclencher la livraison par clic droit contextuel.



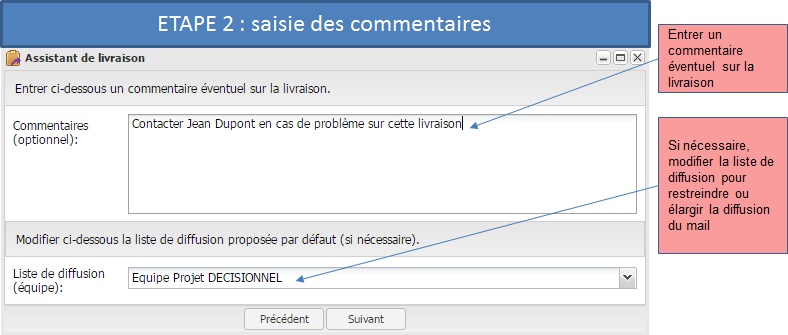
## Etape 1 : Saisie des exigences

Cette étape est optionnelle : cocher les exigences livrées.



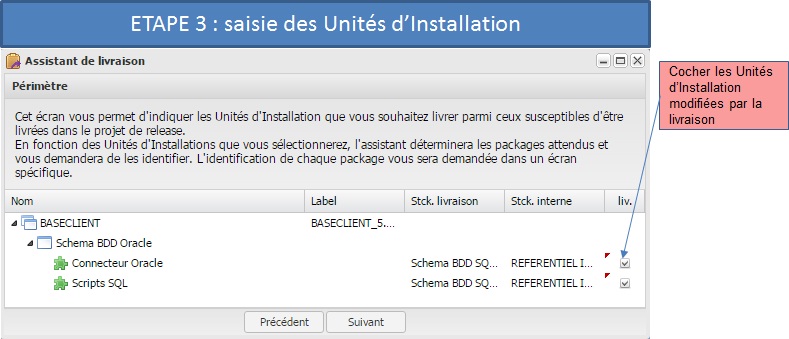
## Etape 2 : Saisie des commentaires

Cette étape est optionnelle : saisir les commentaires livrés.



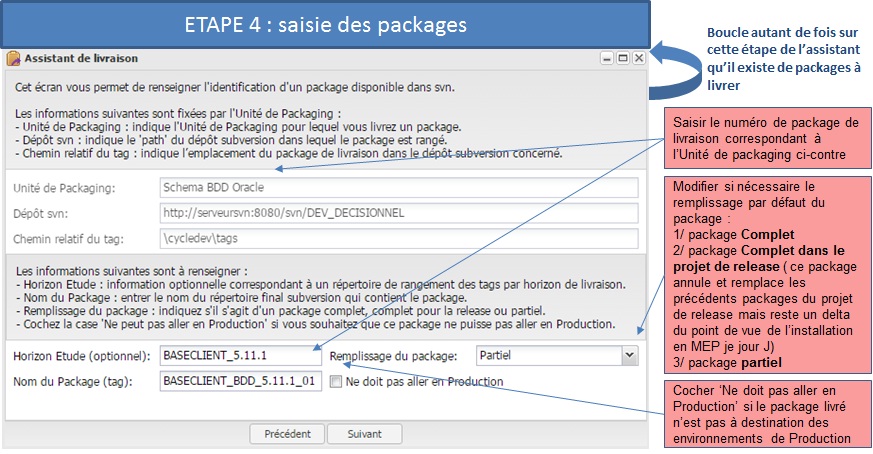
## Etape 3 : Saisie des Unités d’Installation

Cette étape est obligatoire : cocher les Unités d’Installation livrées.



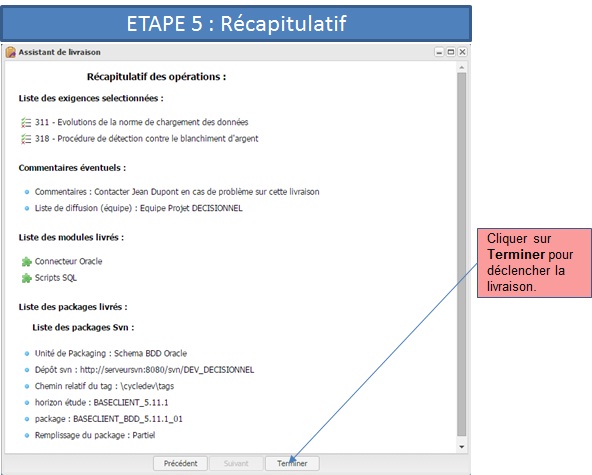
## Etape 4 : Saisie des packages

Cette étape est obligatoire : saisir les packages livrés. Remarque cette étape boucle autant de fois qu’il y a de packages à livrer (en fonction des Unités d’Installation cochées au cours de l’étape 3).



## Etape 5 : Récapitulatif de la livraison

Consulter et vérifier le récapitulatif puis cliquer sur **Terminer** pour déclencher l’opération de livraison.



# Use Case de création d’un package fusionné

## Accéder à l’assistant de création du package fusionné

Une option de construction de packages est mise en place dans Cobuto RM pour les Unités de Packaging pouvant être fusionnées.

Il est ainsi possible de construire un package d’installation résultant de la fusion de plusieurs packages de livraison. La construction de ce package se fait à partir des packages livrés dans le Projet de Release pour construire le package final.

Le package fusionné est techniquement construit en empilant l’ensemble des packages de la Release dans leur ordre d’arrivée.

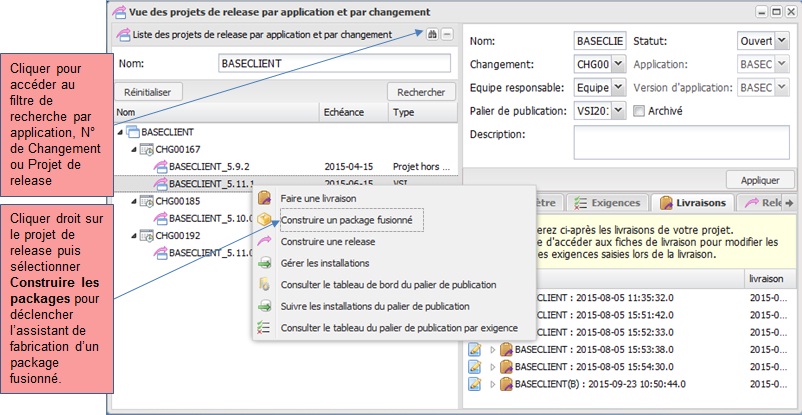
Afin de prévenir des écarts sur les dossiers et répertoires, il est fortement préconisé que les contrôles obligatoires soient généralisés pour les unités de packaging pour lesquels on souhaite utiliser cette fonctionnalité.

Remarque : si un package est construit par fusion, dans ce cas, c’est ce package qui sera en sortie de Cobuto RM et sera exploité pour :

* Concourir à une Release
* Etre exploité pour installation sur les environnements de tests ou de production.

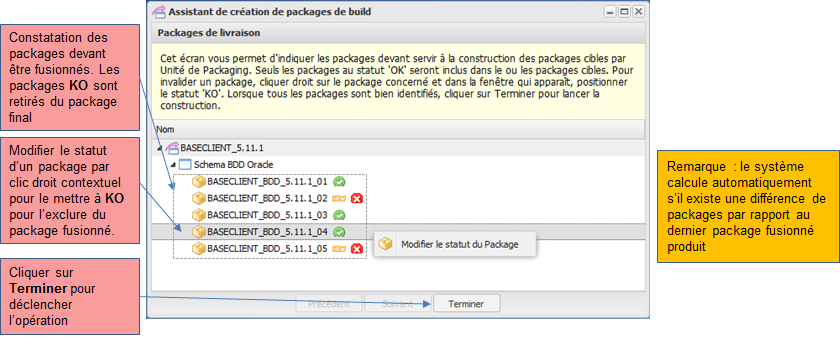
Dans le cas contraire le package de livraison est un package de sortie (pas de transformation).

L’accès à l’assistant de création d’un package fusionné se fait depuis le menu **Démarrer | Création d’un projet de release**. Se positionner alors sur le projet de release concerné et déclencher la fonctionnalité **Construire les packages** par clic droit contextuel comme indiqué sur l’impression d’écran ci-dessous :

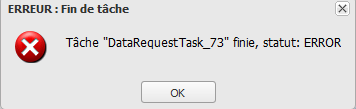


## Assistant de création d’un package fusionné

Les packages sont proposés pour les Unités de packaging comportant cette fonctionnalité. Il est possible de créer, dans une même opération autant de package à fusionner qu’il existe d’unité de packaging paramétrées. Les packages à fusionner sont détectés automatiquement par le système en comparant avec le dernier package fusionné produit. Si de nouveaux package ont été livrés ou si un package a été mis à KO par rapport à la construction précédente l’opération sera déclenchée sinon elle sera ignorée.



Patienter jusqu’à libération de l’assistant et vérifier que le message d’erreur suivant ne s’affiche pas :

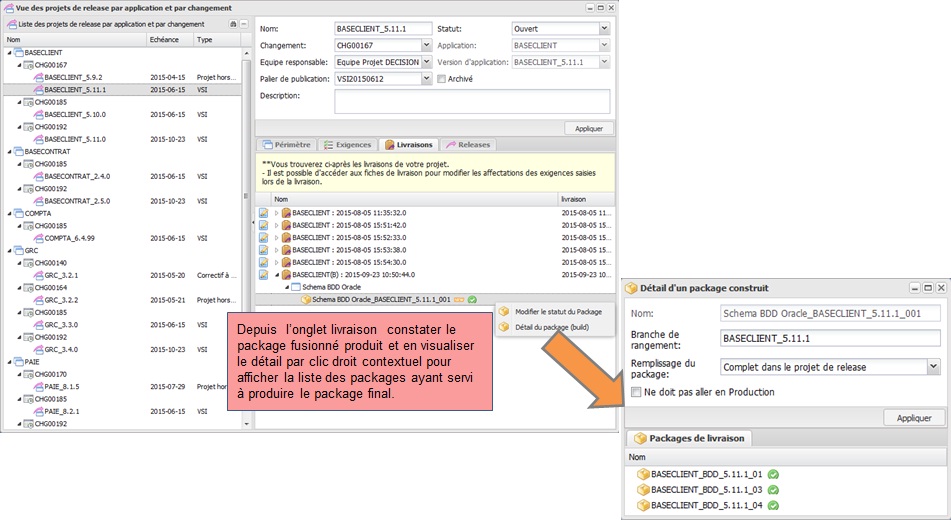


Si l’erreur s’affiche consulter la gestion relative à cette erreur en se reportant à la section ci-dessous : 13.4 Gestion des cas d’erreurs.

## Constater le package fusionné dans l’onglet Livraisons du projet de Release

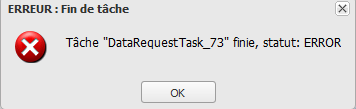
Vérifier la présence du package fusionné dans l’onglet « Livraisons » du projet de release. Il est possible d’y revenir à tout moment pour consulter la liste des packages ayant servi à fusionner le package.

Remarque : le package fusionné contient également la liste des exigences et des correctifs livrés dans chaque package de livraison.



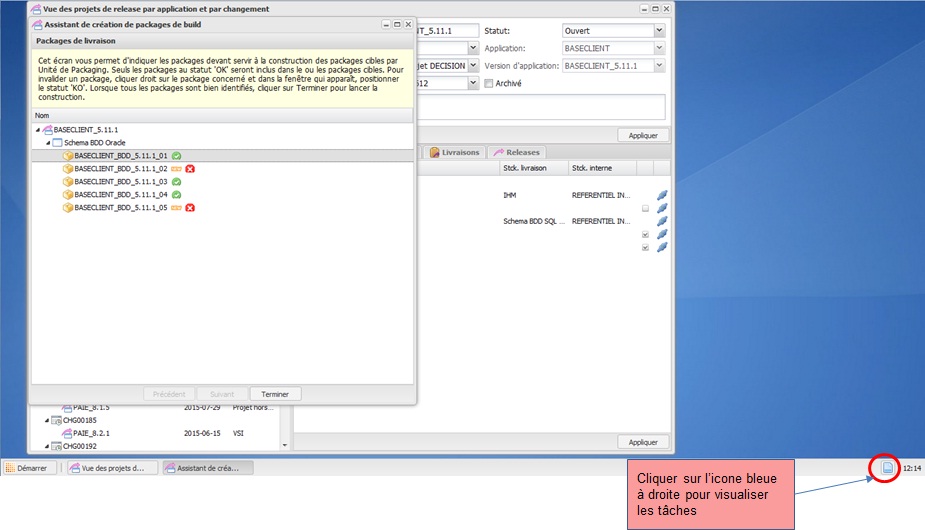
## Gestion des cas d’erreurs

Si des erreurs se produisent, le message d’erreur suivant apparaît :

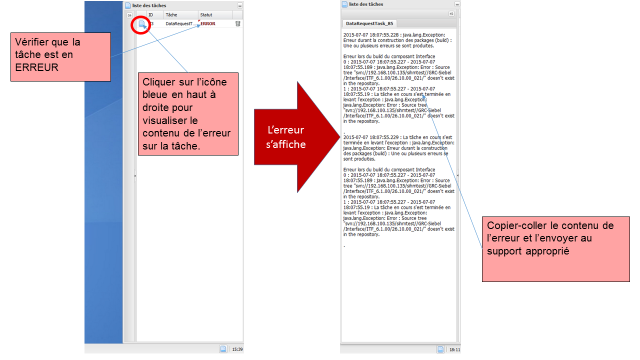


Dans ce cas effectuer, les manipulations suivantes :

1. Cliquer sur l’icône des tâches en bas à droite comme indiqué sur l’impression d’écran ci-dessous :



1. Constater le libellé d’erreur sur la tâche en haut à droite. Dans ce cas, cliquer sur l’icône en haut à droite pour faire apparaître l’erreur :



1. Copier-coller le contenu de l’erreur et l’envoyer au support approprié.

# Use Case de création de la Release

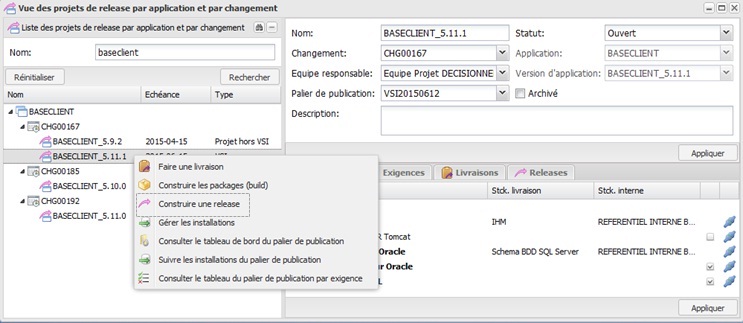
## Accéder à l’assistant de création de la release

La création de la Release se fait :

* A la suite d’une ou de plusieurs nouvelles livraisons sur le projet de Release.
* A partir du projet de Release.
* Elle précède généralement la demande d’installation sur un environnement.

L’accès se fait :

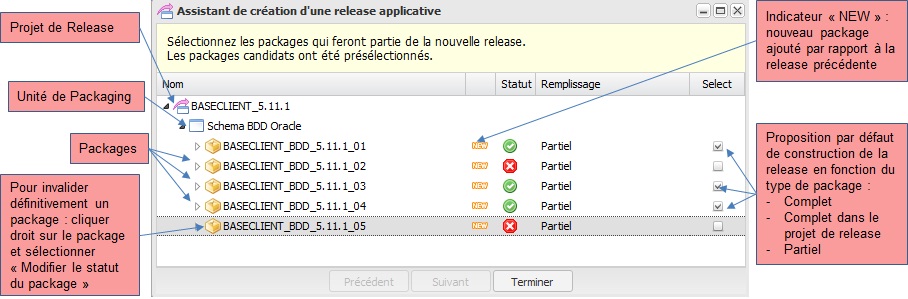
* Depuis la vue des projets de release (**Démarrer | Gestion des projets de release**).
* Par clic-droit contextuel depuis le projet de release puis sélectionner **Construire une release** :



## Assistant de création de la Release (une seule et unique étape)

Dans l’exemple ci-dessous, le projet de Release « GESPA Appli 1.13.0 » a été livré avec cinq unités de packaging (« Applicatif GESPA », « Documentaires », « Graphe GESPA Décisionnel », « Graphe GESPA Quotidien », « Schéma base de données »). Chaque package possède plusieurs informations :

* Complet : le package est complet y compris en Production,
* Complet dans le projet de release : le package est complet dans le projet de release, mais reste un delta par rapport à la production,
* Partiel : le package est partiel.



L’assistant propose une fonction intelligente de création de la release en fonction de ces trois types de packages. Les packages complets / complets dans le projet de release viennent remplacer les précédents alors que les packages partiels s’ajoutent simplement au reste. L’utilisateur peut, s’il le souhaite, ne cocher que les packages qu’il souhaite voir dans sa release. Les packages KO sont ignorés.

Remarque : la release doit être créée à chaque fois dans une perspective de MEP. Elle doit comporter tous les packages devant faire l’objet d’une MEP le jour J y compris les packages ayant déjà été livrés dans une précédente release.

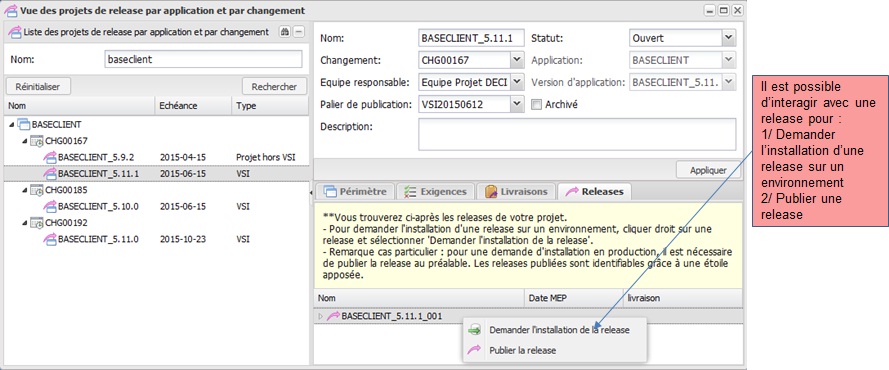
# Use Case de demande d’installation sur les environnements

## Accès à la fonction de demande d’installation

La demande d’installation sur un environnement se fait :

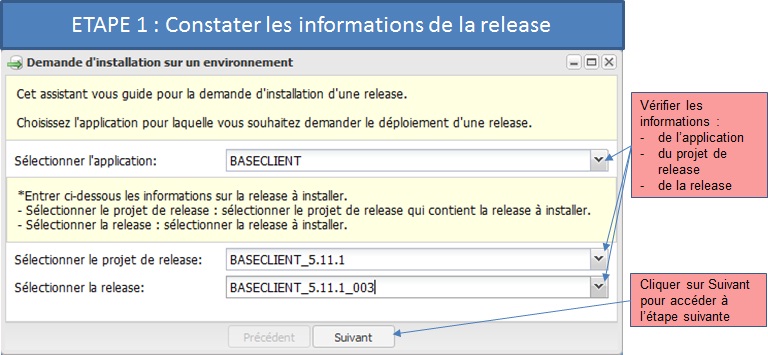
* A partir du projet de Release.
* Elle succède à la création de la Release et précède la demande de déploiement sur un environnement.
* Remarque : les releases faisant l’objet d’une demande d’installation en production doivent être publiées.

L’accès se fait depuis le menu **Démarrer | Création d’un projet de release**. Se positionner alors sur le projet de release concerné et cliquer sur l’onglet « Releases ». Effectuer un clic droit sur la release concernée par la demande d’installation et sélectionner **Demander l’installation de la release**.



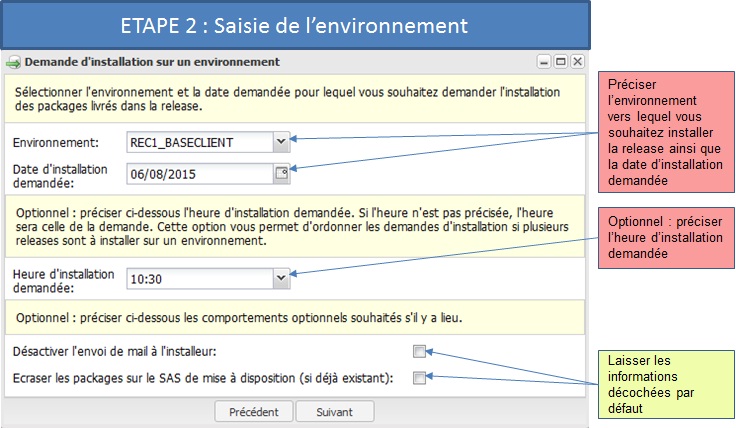
## Première étape : constater les informations de la release

Constater les informations de la release à installer :



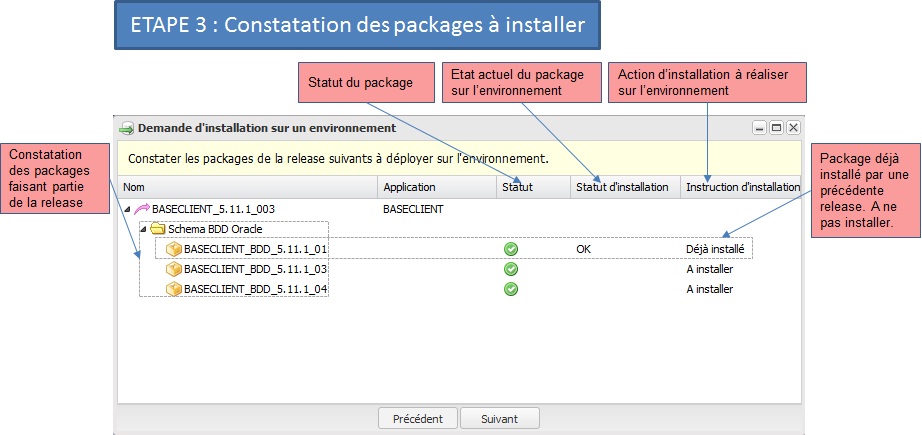
## Deuxième étape : saisie de l’environnement

Saisir l’environnement demandé pour installation.



## Troisième étape : constatation des packages à installer

Constater les packages à installer.

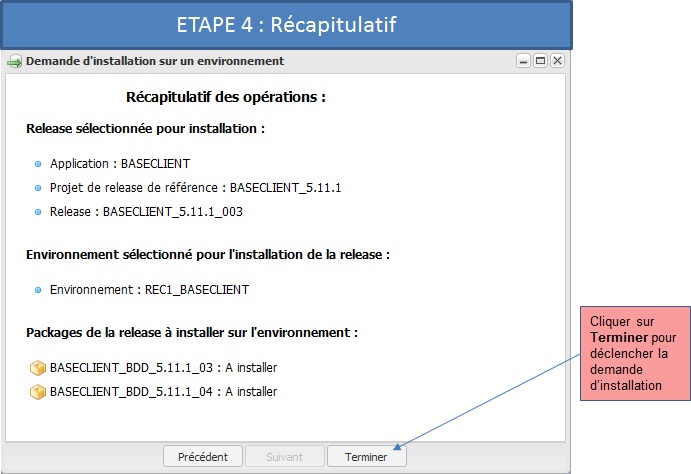


Remarque : cette étape compare les indicateurs des packages avec ce qui est installé sur l’environnement :

* Si le statut du package est OK et que le package n’est pas installé sur l’environnement, le système propose le package pour installation.
* Si le statut du package est OK et que le package est déjà installé sur l’environnement, le système ne le propose pas pour installation.
* Si le statut du package est KO et que le package est déjà installé sur l’environnement, le système propose de le désinstaller : toutefois les désinstallations de packages n’étant pas toujours gérées sur les environnements la demande de désinstallation sera désactivée vers l’installeur dans le mail livré (sauf demande contraire clientèle).
* Si le statut du package est KO et le package est non installé sur l’environnement, le système ignore la demande d’installation du package.

## Quatrième étape : récapitulatif

Vérifier les informations du récapitulatif et déclencher l’opération de mise à disposition.



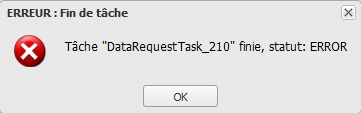
**Traitements de mise à disposition.**

En cliquant sur **Terminer**, l’utilisateur qui fait la demande d’installation déclenche les opérations de mise à disposition :

* Transfert physique des packages dans l’espace de stockage de mise à disposition correspondant à l’environnement.
* Pour chaque équipe d’installation dont au moins une Unité de Packaging comporte des packages à installer, envoi d’un mail récapitulatif avec les instructions d’installation à réaliser.

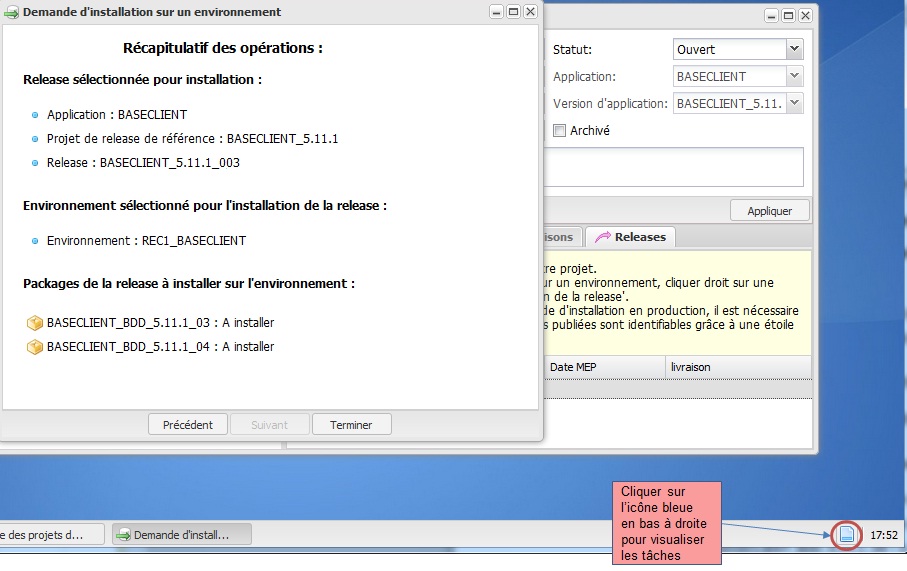
## Gestion des cas d’erreurs

Si des erreurs se produisent, le message d’erreur suivant apparaît :

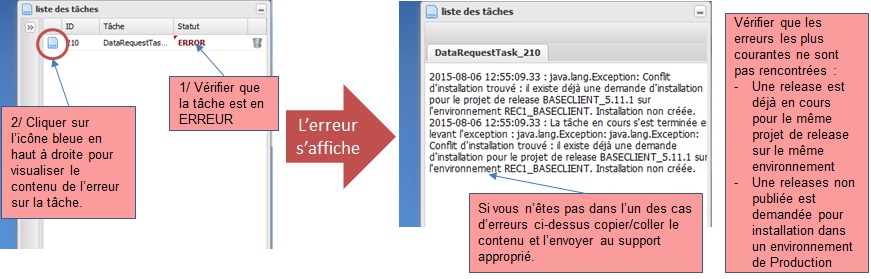


Dans ce cas effectuer, les manipulations suivantes :

1. Cliquer sur l’icône des tâches en bas à droite comme indiqué sur l’impression d’écran ci-dessous :



1. Constater le libellé d’erreur sur la tâche en haut à droite. Dans ce cas, cliquer sur l’icône en haut à droite pour faire apparaître l’erreur :



Copier-coller le contenu de l’erreur et l’envoyer au support approprié.

# Use Case d’aquittement d’une installation sur un environnement

## Accès aux installations en cours

La validation d’une installation sur un environnement se fait à la suite d’une demande d’installation sur un environnement.

L’accès se fait depuis le menu **Démarrer | Suivi des installations en cours**.

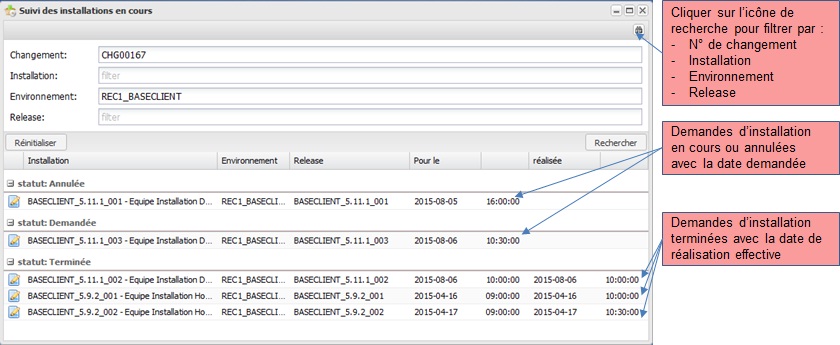
## Constatation des installations en cours à réaliser

Le suivi des installations en cours présente les demandes d’installation :

* Demandées,
* En cours,
* Terminée.

Les demandes d’installation s’affichent avec la date et l’heure demandée de leur installation.

Pour restreindre l’affichage, il est possible d’utiliser le filtre de recherche comme indiqué sur l’impression d’écran ci-dessous.

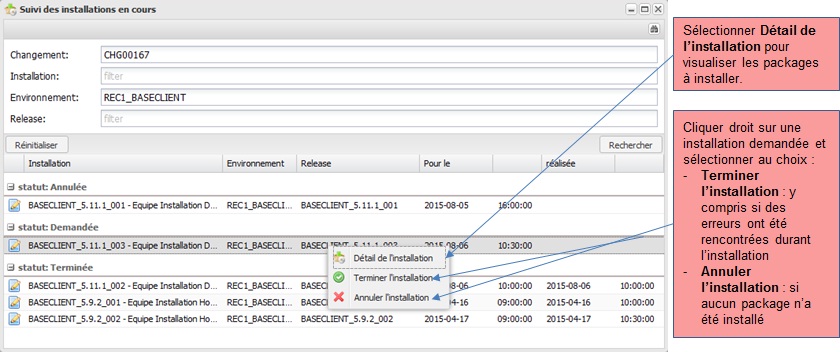


## Acquittement d’une installation

L’acquittement d’une installation se fait par clic droit contextuel, puis sélectionner **Terminer l’installation**.

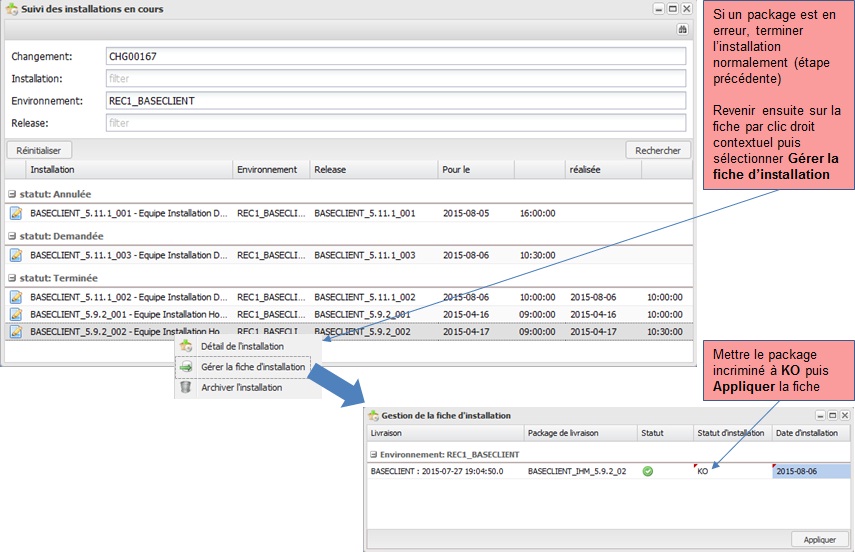
Remarque : il est important de valider les installations dans l’ordre à laquelle les installations ont été procédées, en particulier si plusieurs installations ont été effectuées dans la même journée pour la même application et le même environnement. En effet, un système de calcul des packages complet et complets dans le projet de release supprime au fur et à mesure les packages sur l’environnement s’ils sont remplacés par des packages du même type.

Ce système de calcul ne s’applique pas aux packages de type partiels.



## Mise à jour de l’état des packages si erreur

Si un ou plusieurs packages sont en erreur **Terminer l’installation** puis modifier la fiche a posteriori comme indiquer sur l’impression d’écran ci-dessous :



# Production des bons d’installation

## MCD contenant le lien vers l’objet Environnement

Dans la phase de « Réception » des livraisons, les objets « packages » d’installation sont exploitables et Cobuto RM permet les transformations en sortie.

Pour assurer le suivi sur les environnements, un lien est positionné sur l’objet package d’installation.



## Indicateurs de suivi des packages de livraison sur les environnements

Les packages de livraison ont un statut :

* + - OK : par défaut à la livraison
    - KO : les packages KO ne peuvent pas se trouver dans une release publiée (et donc faire l’objet d’une demande de MEP).

Le lien entre Package de livraison et Environnement a un statut :

* + - Demande de déploiement Annulée
    - Demande de déploiement Demandée
    - Déploiement OK
    - Déploiement KO

Le lien entre Package de livraison et Environnement est également complété par la date d’installation.

## Production du bon d’installation VSI

Pour la production du bon d’installation VSI un écran affiche l’état de l’environnement intégré :

* + - L’environnement intégré (ou « couloir »)
      * Le domaine 1 des applications concernées
        + L’application 1

La source de livraison 1

Les N° de packages de livraison étant au statut installés OK avec :

La date de livraison

Les exigences livrées

La source de livraison 2

Les N° de packages de livraison étant au statut installés OK avec :

La date de livraison

Les exigences livrées

Pour l’ensemble des sources

* + - * + L’application 2

…

* + - * Le domaine 2 des applications concernées

…

Date de début et date de fin :

Par défaut, lors de la création d’un environnement intégré, la date est initialisée au 1er janvier 1900. Par la suite, il est possible de mettre à jour cette date en prenant pour départ la date de rafraîchissement de l’environnement intégré. Cette action a pour effet :

* + - De lister tous les packages installés sur la plage de date (entre la date de début et la date de fin) pour une source donnée.
    - Si aucun package n’est trouvé sur la plage de date pour une source donnée, le bon d’installation affiche le dernier package installé ainsi que sa date.

Accès au bon d’installation :

L’utilisateur peut accéder au bon d’installation depuis l’application Cobuto RM.

Pour les MOAs et les utilisateurs invités, une URL spécifique est mise en place pour donner la possibilité d’un accès direct sans authentification.