**Manuel d'utilisation**

**Description technique cadriciel   
Trace Mémoire**

**Date : 28/09/15**

|  |  |
| --- | --- |
| **Identification du document** |  |
| Emetteur | Olivier Véhier |
| Référence |  |
| Date de création | 28/09/15 |
| Date de validation | 28/09/15 |

|  |  |
| --- | --- |
| **Responsable technique** | Olivier Véhier |

|  |  |  |
| --- | --- | --- |
| **Suivi des versions** | | |
| **Version** | **Date** | **Description** |
| 1 | 28/09/15 | Création du document |
|  |  |  |

# Sommaire

[1. Lexique 4](#_Toc431908322)

[2. Spécification technique 5](#_Toc431908323)

[2.1. Besoin 5](#_Toc431908324)

[2.2. Objets techniques 5](#_Toc431908325)

[2.3. Classe de message dédié aux Tâches // - ZTASK 6](#_Toc431908326)

[2.4. Principe de fonctionnement 6](#_Toc431908327)

[2.5. Utilisation 6](#_Toc431908328)

[2.6. Description technique 8](#_Toc431908329)

[2.6.1. Classe Tâche - ZCL\_TASK 8](#_Toc431908330)

[2.6.1.1. Approche de conception 8](#_Toc431908331)

[2.6.1.2. Détails techniques 8](#_Toc431908332)

[2.6.1.2.1. Constructeur 8](#_Toc431908333)

[2.6.1.2.2. Lancement du traitement - TASK\_START 8](#_Toc431908334)

[2.6.1.2.3. Handler fin traitement - AT\_END\_OF\_TASK 9](#_Toc431908335)

[2.6.1.2.4. Conversion paramètres statiques en dynamiques - CONVERT\_STATIC\_PARAMETERS 9](#_Toc431908336)

[2.6.1.2.5. Conversion paramètres dynamiques en statiques- CONVERT\_DYNAMIC\_PARAMETERS 9](#_Toc431908337)

[2.6.2. Classe Manager de Tâche - ZCL\_TASK\_MANAGER 10](#_Toc431908338)

[2.6.2.1. Approche de conception 10](#_Toc431908339)

[2.6.2.2. Détails techniques 10](#_Toc431908340)

[2.6.2.2.1. Constructeur - CONSTRUCTOR 10](#_Toc431908341)

[2.6.2.2.2. Lancement nouvelle Tâche - TASK\_START\_NEW 10](#_Toc431908342)

[2.6.2.2.3. Attente fin de toutes les Tâches - TASK\_FINISH\_RUNNING 11](#_Toc431908343)

[2.6.2.2.4. Handler fin d'une Tâche - HANDLER\_END\_OF\_TASK 11](#_Toc431908344)

[2.6.3. Classe Handler Manager de Tâche 12](#_Toc431908345)

[2.6.3.1. Approche de conception 12](#_Toc431908346)

[2.6.3.2. Détails techniques 12](#_Toc431908347)

[2.6.3.2.1. Constructeur - CONSTRUCTOR 12](#_Toc431908348)

[2.6.3.2.2. Handler Fin d'une Tâche (Manager) - HANDLER\_END\_OF\_TASK 12](#_Toc431908349)

[2.6.3.2.3. Création Handler - HANDLER\_FACTORY 13](#_Toc431908350)

[2.7. Perfectibilité 14](#_Toc431908351)

[2.7.1. Problème connu 14](#_Toc431908352)

[2.7.2. Axe d'amélioration 14](#_Toc431908353)

[2.7.3. Reste à faire 14](#_Toc431908354)

1. Lexique
2. Spécification technique
   1. Besoin

Fournir au développeur un cadriciel pour tracer l'utilisation mémoire d'un traitement

* 1. Objets techniques

Objets spécifiques utilisés :

|  |  |  |
| --- | --- | --- |
| Nom de l’objet | Type de l’objet | Commentaire |
| ZCL\_RUNTIME\_AND\_MEMORY\_UTIL | Classe | Classe Trace mémoire |
| ZCL\_RUNTIME\_AND\_MEMORY\_UTIL\_ST | Classe | Classe Singleton Trace mémoire (version alpha) |
| ZTEC\_TRACE\_CONF | Table | Table Configuration des Traces |
| ZTEC\_T\_TRACE | Table | Historique des Traces |

* 1. Table de Configuration - ZTEC\_TRACE\_CONF

Cette table permet d'activer - ou de désactiver - les Traces d'un "programme".

|  |  |  |
| --- | --- | --- |
| Champ | Élément donné | Description |
| MANDT | MANDT | Mandant |
| PROGRAMM | PROGRAMM | Nom du programme ABAP |
| TRACE\_RUNTIME | ZTEC\_TRACE\_RUNTIME | Trace durée d'exécution ? |
| TRACE\_MEMORY | ZTEC\_TRACE\_MEMORY | Trace l'Empreinte mémoire ? |
| TAKE\_MEMORY\_SNAPSHOT | ZTEC\_TAKE\_MEMORY\_SNAPSHOT | Capture empreinte mémoire |

Légende : Clefs, données

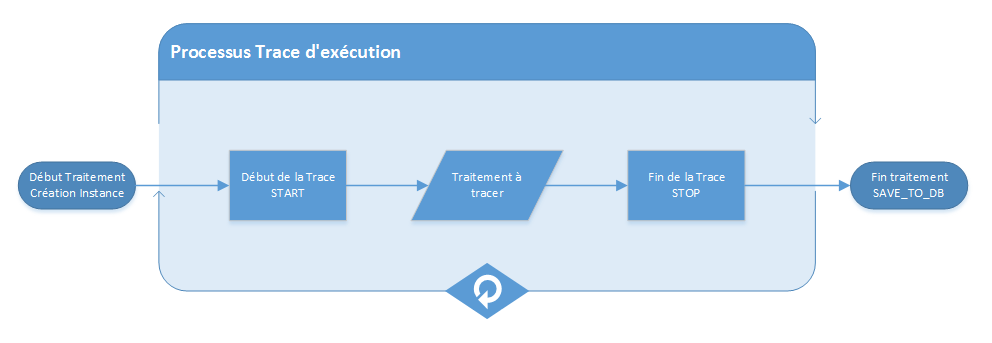
* 1. Table d'historique - ZTEC\_T\_TRACE

Cette table contient les différentes Traces enregistrées.

|  |  |  |
| --- | --- | --- |
| Champ | Élément donné | Description |
| MANDT | MANDT | Mandant |
| GUID\_RUNTIME | ZTEC\_GUID\_RUNTIME | ID Unique trace |
| PROGRAMM | PROGRAMM | Nom du programme ABAP |
| TRACE\_NO | ZTEC\_TRACE\_NO | Numéro de Trace |
| ID | ZTEC\_ID | ID Séquence |
| SEQNR | ZTEC\_SEQNR | Numéro de séquence |
| TIME\_START | TIMESTAMP | Horodatage UTC en abrégé (AAAAMMJJhhmmss) |
| TIME\_STOP | TIMESTAMP | Horodatage UTC en abrégé (AAAAMMJJhhmmss) |
| RTIME\_START |  | Début d'exécution |
| RTIME\_END |  | Fin d'exécution |
| RTIME\_DURATION |  | Durée d'exécution |
| MEM\_BEFORE | ABAP\_MSIZE | Type DEC correct pour nombre entier 64 octets |
| MEM\_AFTER | ABAP\_MSIZE | Type DEC correct pour nombre entier 64 octets |
| MEM\_ALLOCATED | ABAP\_MSIZE | Type DEC correct pour nombre entier 64 octets |
| OBJECT\_ID | ZTEC\_OBJECT\_ID | Identifiant Objet |
| DESCRIPTION | ZTEC\_DESCRIPTION | Description de la trace |
| EXEC\_BY | UNAME | Nom d'utilisateur |
| EXEC\_DATE | DATUM | Date |
| EXEC\_TIME | UZEIT | Heure |

Légende : Clefs, données

* 1. Principe de fonctionnement



* 1. Utilisation

Voici le principe d'utilisation :

* Deux modes de fonctionnement possible :
  + Création d'une instance de la classe "**ZCL\_RUNTIME\_AND\_MEMORY\_UTIL**" avec les paramètres :
    - Activation trace Durée d'exécution
    - Activation trace Empreinte mémoire

Légende : Facultatif

* + Appel de la méthode "**CREATE\_RUNTIME\_UTIL**" avec :
    - Nom du programme - Fait référence à l'identifiant "**PROGRAMM**" de la table "**ZTEC\_TRACE\_CONF**".

Légende : Obligatoire

* + - Une instance sera créée uniquement si au moins un des indicateurs d'activation de Trace est coché dans la table de configuration.
* Trace l'exécution du ou des traitements :
  + Appel de la méthode "**START**" afin de démarrer une trace précise :
    - **ID de la trace** : Identifiant permettant de retrouver la trace dans la table historique
      * Conseil : Mettre un identifiant texte
        + Exemple : "TRACE\_ALL\_PROCESS" pour indiquer que la trace portera sur l'ensemble du traitement.
    - **Numéro de séquence** : Utile si on veut plusieurs entrée différentes pour un même ID de Trace
    - **ID Objet** : Permet de fournir une clef "métier" d'un objet
      * Exemple : Dans le cas d'un traitement sur une Livraison Sortante, il est intéressant de renseigner ce paramètre avec le numéro de la Livraison.
    - **Description** : Texte libre

Légende : Obligatoire, Facultatif

* + Exécution du traitement à tracer.
  + Appel de la méthode "**STOP**" afin d'arrêter la trace :
    - **ID de la trace** : Indiquer l'ID de la trace à arrêter
    - **Numéro de séquence** : Indiquer le numéro de séquence de la trace à arrêter

Légende : Obligatoire, Facultatif

* Appel de la méthode "**SAVE\_TO\_DB**" afin de sauvegarder l'ensemble des Traces collectées durant le traitement. Les traces ainsi sauvegarder seront présentes dans la table "**ZTEC\_T\_TRACE**".
  1. Détails techniques
     1. Constructeur - CONSTRUCTOR

Cette méthode est appelée - par le système - lorsque le traitement asynchrone du MF "**Z\_TASK\_CALL\_PROCESS**" est terminé.

* Génération d'un ID unique de Tâche
* Initialisation attributs pour Mode Dédié
  + Initialisation attributs
  + Récupération paramètres d'appel du MF
  1. Perfectibilité
     1. Problème connu
* Aucun problème connu.
  + 1. Axe d'amélioration
* Finir la gestion de la classe singleton "**ZCL\_RUNTIME\_AND\_MEMORY\_UTIL\_ST**"
  + 1. Reste à faire
  1. Exemple d'utilisation

Le programme "**ZDEMO\_RUNTIME\_AND\_MEMORY**" sert d'exemple d'utilisation.