# OCR PDF

Commande pour OCRiser un fichier pdf image :

```
IMG2OCRPDFCS <fichier source> <fichier destination> <ocrmode> <jobqueueid>
<fichier texte> <fichier config>
```

- <fichier source> : le fichier à OCRiser.
- <fichier desination > : le nom du fichier après OCRisation.
- <ocrmode> : le numéro de la tentative.
- <fichier config > : le fichier de configuration opcsdk.conf

Exemple:

```
C:\usr\local\nchp\opcsdk\IMG2OCRPDFCS.exe monfichier.pdf monfichier.ocr.pdf
0 0 monfichier.txt C:\nchp\etc\nchp\ezged\opcsdk.conf
```

# OCR PDF + CB

Commande pour OCRiser avec lecture de code à barres:

```
IMG2OCRPDFCBCS <ficher source> <fichier destination> <ocrmode> <jobqueueid>
<fichier texte> <fichier config> <page_1_seule> <type cb> <code page> <max
bc> <resolution> <orientation> <max inclinaison> <zone>
```

# OCR Paramètres

Paramètres modifiable du fichier opcsdk.conf

par défaut l'ocr est limité en taille d'image à 8400 pixels en largeur et 8400 pixels en hauteur **pour autoriser l'OCR sur les fichiers plus gros** vous pouvez ajouter ces paramètres au fichier opcsdk.conf

```
Kernel.Img.Max.Pix.X = 32000
Kernel.Img.Max.Pix.Y = 32000
```

ATTENTION, les performances peuvent être dégradées et nous ne garantissons pas la sortie correcte du fichier Ocrisé, en effet nous avons constatés des bugs de rendu d'image sur certains plans noir et blanc dépassant le A0. Nous vous conseillons donc de garder les formats d'origine dans ces cas. 32000 est le maximum, ne le dépassez pas.

**pour définir le format de sortie en PDF/A** Modifier ce paramètre qui par défaut est en PDF1.4

```
Converters.Text.PDFImageOnText.Compatibility = PDF-A
```

**pour définir le format par défaut de compression MRC couleur** 0 = No compression 1 = Haute compression qualité minimale 2 = Moyenne compression qualité bonne 3 = Faible compression qualité sans perte

```
Converters.Text.PDFImageOnText.UseMRC = 2
```

**pour définir le qualité des couleurs** quand MRC n'est pas utilisé (noir et blanc par exemple) 1 = Haute compression qualité minimale 2 = Moyenne compression qualité bonne 3 = Faible compression qualité sans perte

```
Converters.Text.PDFImageOnText.ColorQuality = 2
```

# Conversion vers Office

Voici comment tester la conversion en ligne de commande :

```
IMG2OCRWORDCS <fichier source> <fichier destination> <ocrmode>  <jobqueueid>
<fichier texte> <fichier config> <format>
```

- <fichier source> : le fichier à convertir.
- <fichier desination > : le nom du fichier une fois qu'il sera converti.
- <ocrmode> : le numéro de la tentative (pour tester via un terminal mettre 0)
- <jobqueueid> : numéro de travail (pour tester via un terminal mettre 0)
- <fichier texte> : fichier vers lequel sera extrait le texte
- <fichier config > : le fichier de configuration opcsdk.conf
- <format> : format vers lequel la conversion sera effectuée

/!\ Si vous souhaitez convertir vers le format **docx** vérifiez que le **.Net Framework 3.5** est bien installé sur votre serveur.

Sinon en testant la conversion en ligne de commande vous aurez un joli message d'erreur :

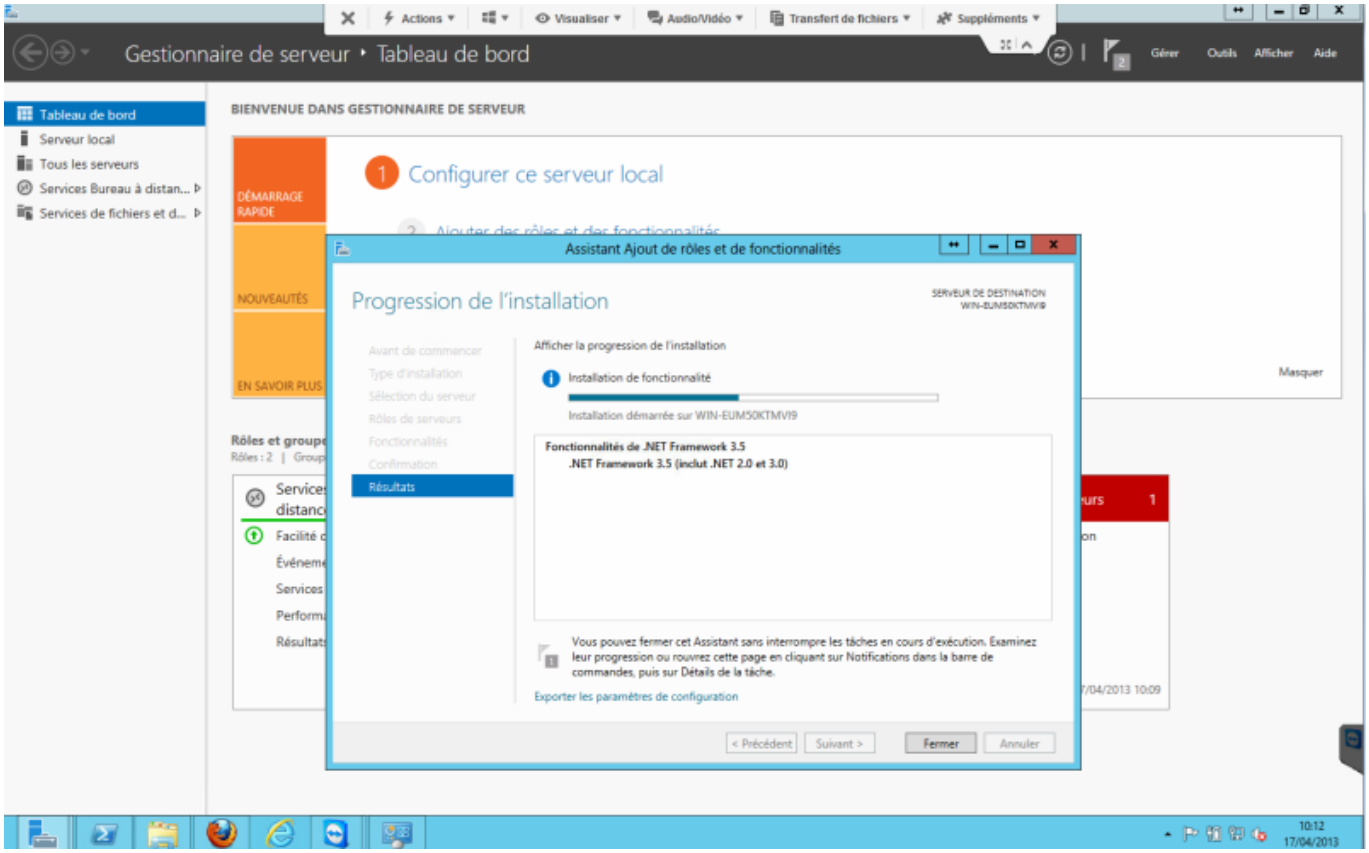
```
Erreur 18 :API_ERROR_ERR
```

# Problèmes d'OCR et résolutions

## Pas d'OCR sur certains documents

Symptôme : Des documents types .docx, .msg voire certains pdf, ne sont pas OCRisés.

Résolution: Vérifiez que le .NET Framework 3.5 est bien installé.




From:
http://wiki.ezdev.fr/ - **EzGED Wiki**

Permanent link:
**http://wiki.ezdev.fr/doku.php?id=ocr&rev=1504006504**

Last update: **2023/03/17 09:56**