



# Créer des Dashboards dynamiques avec PowerShell

Partie 2

Dans nos métiers de l'informatique, il est souvent utile d'avoir accès à des dashboards (tableaux de bord), permettant de résumer graphiquement des actions effectuées, des états spécifiques de machines, serveurs, services... Divers outils permettent de réaliser ces Dashboards. Dans cet article, je vais vous présenter l'outil Universal Dashboard, qui permet de réaliser facilement des Dashboards graphiques (responsive) avec un design moderne, à l'aide de PowerShell.

## Création d'un Grid

Le Grid permet de lister des données avec plusieurs paramètres, par exemple nom de machine, statuts, date...

Conservons le chart Donuts, et ajoutons un Grid, comme dans l'exemple [Fig. 7].

Pour créer le Grid ajoutez la partie suivante :

```
New-UdGrid -Title "Déploiements récents" -Headers @("Date", "Ordinateur", "Pourcentage", "Étape en cours", "Status") -Properties @("Date", "Computer Name", "Percent Complete", "Step Name", "Deployment Status") -AutoRefresh -RefreshInterval 60 -Endpoint {$Alldatas | Out-UDGridData }
```

Ajoutons maintenant quelques éléments permettant d'afficher rapidement le nombre d'éléments en cours d'installation, en échec... durant la dernière journée [Fig. 8].

Pour cela, nous allons utiliser le composant counter suivant : New-UDCounter

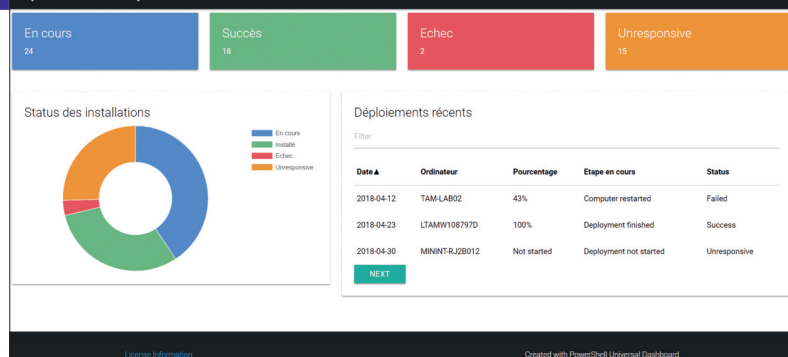
Ces « counter » seront inclus sur une ligne divisée en 4 colonnes de taille 3 chacune.

```
New-UDCounter -Title "Installations en cours" -BackgroundColor "#5290E9" -TextAlignment Left -Endpoint {$RunningInstall} -FontColor "White"
```

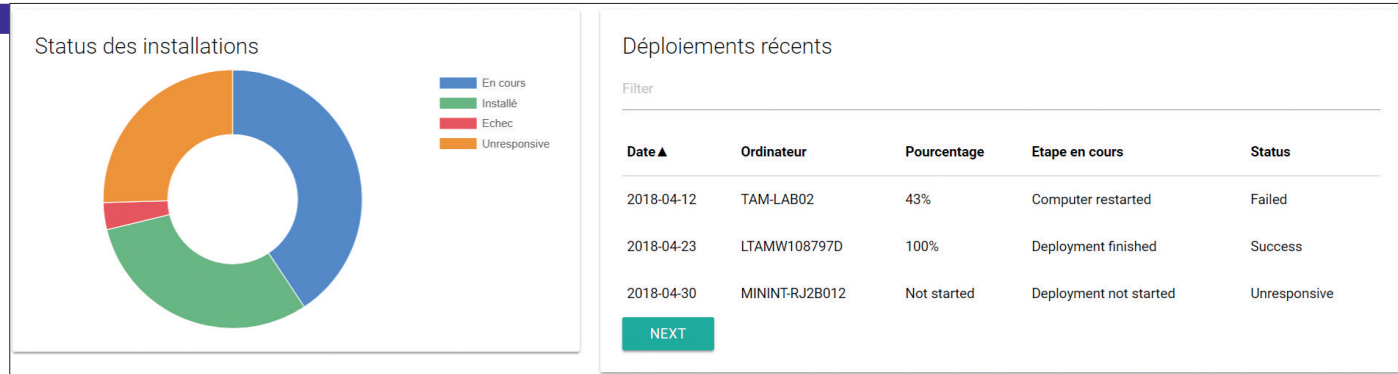
Le code utilisé pour cet exemple est le suivant :

```
New-UDRow -Columns {
  New-UDColumn -Size 3 {
    New-UDCounter -Title "En cours" -BackgroundColor "#5290E9" -Endpoint {
      $NB_Running
    } -FontColor "White"
  }
  New-UDColumn -Size 3 {
    New-UDCounter -Title "Succès" -BackgroundColor "#71B37C" -Endpoint {
      $NB_Success
    } -FontColor "White"
  }
  New-UDColumn -Size 3 {
    New-UDCounter -Title "Echec" -BackgroundColor "#E14D57" -Endpoint {
      $NB_Failed
    } -FontColor "White"
  }
  New-UDColumn -Size 3 {
    New-UDCounter -Title "Unresponsive" -BackgroundColor "#EC932F" -Endpoint {
      $NB_Unresponsive
    }
  }
}
```

## 8 Déploiement des postes de travail



## 7



```

    } -FontColor "White"
  }
}

New-UDRow -Columns {
  New-UDColumn -Size 5 {
    New-UDChart -Title "Status des installations" -Type "Doughnut" -Endpoint {
      $Data | Out-UDChartData -LabelProperty "Install" -Dataset @(
        New-UDChartDataset -Label "Install" -DataProperty Count -Background
Color @("#5290E9","#71B37C","#E14D57","#EC932F") -BorderColor @("white", "white",
"white", "white") -BorderWidth 1
      )
    } -Options $Options
  }

  New-UDColumn -Size 7 {
    New-UdGrid -Title "Déploiements récents" -Headers @("Date", "Ordinateur",
"Pourcentage", "Étape en cours", "Status") -Properties @("Date", "Computer Name",
"Percent Complete", "Step Name", "Deployment Status") -AutoRefresh -RefreshInterval 60
-Endpoint {
      $Alldatas | Out-UDGridData
    } -PageSize 3
  }
}

```

Un autre cas concret pouvant faire l'objet d'un Dashboard est l'audit de l'Active Directory (AD). Le dashboard ci-dessous basé sur le même format que le précédent permet de lister l'état de mots de passe et l'état des comptes AD.

### Monitoring d'un élément

Une autre commande, **New-UDmonitor**, permet de monitorer en temps réel un élément particulier, tel que l'analyse de l'utilisation du processeur ou de la RAM d'une machine.

Le design se présente comme ceci [Fig. 9].

Le code pour cet exemple est le suivant :

```

New-UDRow -Columns {
  New-UDColumn -Size 6 {
    New-UdMonitor -Title "CPU Monitoring" -Type Line -AutoRefresh -RefreshInterval 3 `
-ChartBackgroundColor "#80FF6B63" -ChartBorderColor "#FFF6B63" -Endpoint {
      Get-Counter "\Processor(_Total)\% Processor Time" -ErrorAction SilentlyContinue |
      Select-Object -ExpandProperty CounterSamples |
      Select-Object -ExpandProperty CookedValue | Out-UDMonitorData
    }
  }

  New-UDColumn -Size 6 {
    New-UdMonitor -Title "RAM Monitoring" -Type Line -AutoRefresh -RefreshInterval 3 `
-ChartBackgroundColor "#8028E842" -ChartBorderColor "#FF28E842" -Endpoint {
      Get-Counter "\memory\% committed bytes in use" -ErrorAction SilentlyContinue |
      Select-Object -ExpandProperty CounterSamples |
      Select-Object -ExpandProperty CookedValue | Out-UDMonitorData
    }
  }
}

```

```

    }
  }
}

```

Ajoutons maintenant quelques compteurs afin d'afficher des informations telles que l'espace disque disponible, la version de l'OS, le pourcentage du CPU utilisé et le nombre d'erreurs système rencontrées durant le dernier jour [Fig. 10]. Les deux dernières informations sont automatiquement rafraîchies.

### Des graphiques, mais pas que...

Universal Dashboard permet de créer des graphiques, mais pas que... On peut également créer différents contrôles (Controls) à intégrer à des pages Web, tels que :

- Un Dashboard multipage à New UDPage ;
- Une page d'authentification ;
- Une partie de préchargement à New-UDPreloader ;
- Des zones de saisie : simple texte, mot de passe ;
- Contrôles de sélection à Combobox, CheckBox, RadioButton.

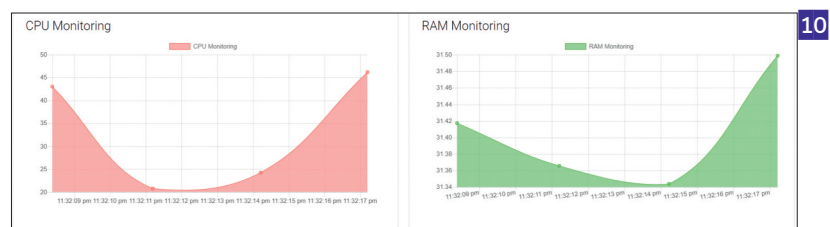
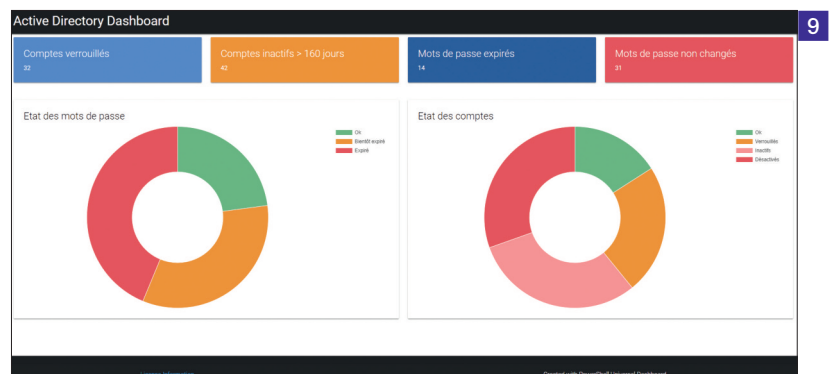
La création d'un Dashboard multipage se réalise en utilisant le code ci-dessous :

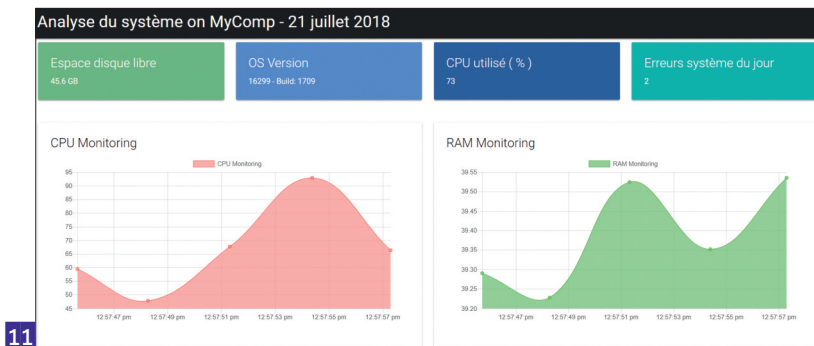
```

$Page1 = New-UDPage -Name "Analyse de déploiements" -Icon laptop -Content {Page1
_Content}
$Page2 = New-UDPage -Name "Analyse du système" -Icon area_chart -Content {}
$Dashboard = New-UDDashboard -Title "Mon Dashboard - $CompName - $Date" -Color
"#FF252525" -Pages @($Page1, $Page2)

```

- Insérez le contenu de votre 1<sup>re</sup> page dans la zone Page1\_Content
- Insérez le contenu de votre 2<sup>de</sup> page dans la zone Page2\_Content.





11

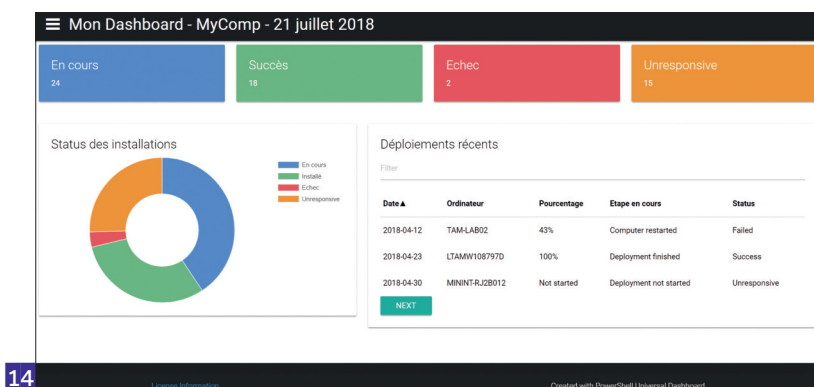
☰ Mon Dashboard - MyComp - 21 juillet 2018

12

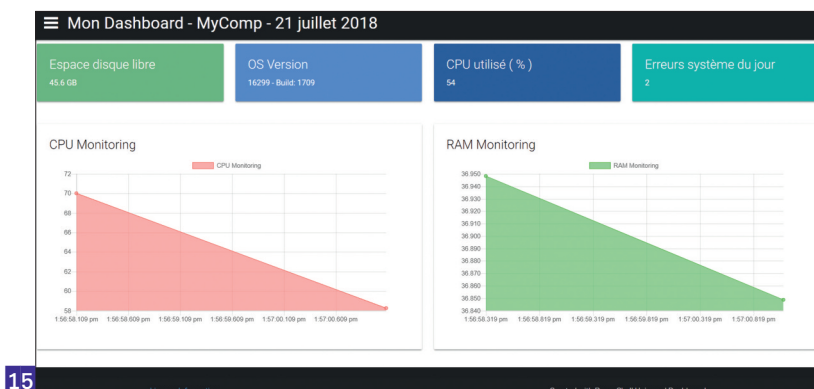
📄 Analyse de déploiements - MyComp - 21 juillet 2018

📊 Analyse du système

13



14



15

Mon formulaire

Tapez votre texte

Hello

Votre mot de passe

.....

Mon CheckBox

Choix 1

RadioButton1  
 RadioButton2  
 RadioButton3

SUBMIT

16

Le Dashboard s'affiche de la façon suivante [Fig. 11]. Cliquez sur le bouton pour afficher les pages disponibles [Fig. 12].

Créons à présent un Dashboard multipage qui contiendra nos deux précédents exemples

- Analyse des déploiements [Fig. 13]
- Analyse du système [Fig. 14]

L'exemple [Fig. 15] montre certains contrôles (Controls) disponibles dans Universal Dashboard permettant de créer vos propres formulaires.

On y retrouve :

- Une zone de saisie de texte normal ;
- Une zone de saisie de mot de passe ;
- Un Combobox permettant de lister différents choix ;
- Des CheckBox ;
- Des RadioButton ;

[Fig. 16]

Le code de cet exemple est le suivant :

```

New-UDRow -Columns {
    New-UDColumn -Size 6 {
        New-UDInput -Title "Mon formulaire" -Id "Form" -Content {
            New-UDTextField -Type 'textbox' -Name 'Text' -Placeholder 'Tapez votre texte'
            New-UDPasswordField -Type 'password' -Name 'password' -Placeholder 'Votre mot de passe'
            New-UDCheckBox -Type 'checkbox' -Name 'CheckBox' -Placeholder 'Ma CheckBox'
            New-UDSelect -Type 'select' -Name 'Combo' -Placeholder "" -Values @("Choix 1", "Choix 2", "Choix 3")
            New-UDRadioButton -Type 'radioButtons' -Name 'RadioButton' -Placeholder @("RadioButton1", "RadioButton2", "RadioButton3") -Values @("RadioButton1", "RadioButton1", "RadioButton1")
        } -Endpoint {
            param($Text, $CheckBox, $Combo, $RadioButton, $password)
        }
    }
}
    
```

Vous l'aurez compris, il existe beaucoup de possibilités avec ce produit qui s'utilise assez simplement.

Différents exemples de ce qu'il est possible de faire avec Charts.js et Material Design sont disponibles ici (6) :

- (6) <https://adamdriscoll.gitbooks.io/powershell-universal-dashboard/content/components/>
- (7) <http://demo.geekslabs.com/materialize/v2.1/layout03/index.html>