

leterrier-developpeurs - Feature - Fonctionnalité #4800

[Qt5.6] Scaling non fonctionnel

13/04/2016 14:42 - Icham Sirat

Statut: Assigned - En cours	Début: 15/04/2016
Priorité: Normale	Echéance:
Assigné à: Icham Sirat	% réalisé: 100%
Catégorie:	Temps estimé: 0.00 heure
Version cible:	
Description Fenêtre trop grande en maximise, trop petite en normale.... La chasse aux tailles gravées dans le code :)	
Sous-tâches: Feature - Fonctionnalité # 4806: MenuFeuille contextualisation des boutons Quitter/Redu... Fixed - Corrigé - Implé Bug # 4809: [TAB] Bizzarerie graphique Fixed - Corrigé - Implé Bug # 4811: [Qt5.6] Scaling SSOBox sans Parent Fixed - Corrigé - Implé Bug # 4815: resizeEvent ... Fixed - Corrigé - Implé Bug # 4812: Scaling MessageBox Fixed - Corrigé - Implé	

Historique

#1 - 14/04/2016 17:33 - Icham Sirat

- % réalisé changé de 0 à 50

Trouvé après toutes ces recherches : c'est le showMaximised() appelé dans l'AbulEduMenuFeuilleV1 svp...
Au passage j'ai corrigé le bug qui faisait que parfois il maximisait 2 fois avant de réduire.

J'effectue quelques tests croisés.

#2 - 15/04/2016 10:18 - Icham Sirat

- Tracker changé de Bug à Feature - Fonctionnalité

- Projet changé de Raconte-Moi à leterrier-developpeurs

#3 - 15/04/2016 10:22 - Icham Sirat

J'ai la solution : il faut fixer la taille maximale de la fenêtre par rapport à l'écran.
Seulement, il se pose la question du multi-écrans et de l'affichage déporté (videoprojecteur...).

Je viens de trouver comment gérer le changement d'écran :

```
connect(window()->windowHandle(), SIGNAL(screenChanged(QScreen*)), this, SLOT(screenChanged(QScreen*)));
```

Le problème, c'est que la QWindow retournée par windowHandle() est NULL tant que la fenêtre n'est pas complètement chargée. Le contournement consiste à créer un signal dans la méthode showEvent(), emit window_loaded() par exemple et de se connecter à ce signal pour ensuite récupérer la QWindow :)

C'est alambiquée mais c'est une solution viable =)

#4 - 15/04/2016 10:25 - Icham Sirat

```
#####
***.h
#####
private slots:
    void showEvent(QShowEvent *ev);
    void setMaxSize(QScreen *);
    void getWindow();

signals :
    void guiLoaded();

#####
***.cpp
#####
# dans le constructeur
    connect(this, SIGNAL(guiLoaded()), this, SLOT(getWindow()));

void ***::showEvent(QShowEvent *ev)
{
    QMainWindow::showEvent(ev);
    this->setMaximumSize(abeApp->primaryScreen()->size());
    emit guiLoaded();
}

void ***::getWindow()
{
    /* la gui est chargée, on peut récupérer la QWindow */
    connect(window()->windowHandle(), SIGNAL(screenChanged(QScreen*)), this, SLOT(setMaxSize(QScreen*)));
}

void ***::setMaxSize(QScreen *s)
{
    qDebug() << "QScreen changed : : " << s->size();
    this->setMaximumSize(s->size());
}
}
```

#5 - 15/04/2016 12:11 - Icham Sirat

Bon je ne comprends rien sur raconte-moi : 2 heures que je bataille à enlever les tailles fixes, enlever le menufeuille (pour tests), faire des setFullScreen (mais on ne voit plus la barre des taches), à builder sur jenkins et à copier sur tablette... Et il est toujours en tout petit à la première apparition !

#6 - 15/04/2016 12:24 - Icham Sirat

AHHHH trouvé : si la QMainWindow est frameless sous Windows, le scaling n'est pas bon.

C'est un bug rejeté de chez Qt (au moins au niveau Windows OS <https://bugreports.qt.io/browse/QTBUG-8361>)

Car, du point de vue gestionnaire desktop Windows, une fenêtre frameless est obligatoirement fullscreen.

Il faut contourner le problème en faisant :

```
Please consider workaround this windows issue calling setGeometry(QApplication::desktop()->availableGeometry()) in showMaximized() for frameless window.
```