

13

Étude d'une première application avec Qt Quick

L'objectif de ce chapitre est de développer une calculatrice avec Qt Quick, relativement basique et courte (deux cents lignes de code). Nous en présenterons le développement étape par étape, afin de montrer un cheminement possible pour construire une application, passant d'une simple maquette à une application de plus en plus évoluée. Cette démarche est surtout prévue pour être généralisable à d'autres applications : commencer avec peu de choses, puis ajouter petit à petit ; trouver un problème, le résoudre ; dénicher une nouveauté de Qt Quick et voir comment l'intégrer.

La calculatrice a été choisie parce que le sujet est relativement simple en soi mais permet un grand nombre de variations. L'objet émulé est bien connu de tous, vous ne perdrez pas de temps à en comprendre les tenants et aboutissants. Ainsi nous pourrons nous focaliser sur les aspects plus techniques de la réalisation avec Qt Quick.

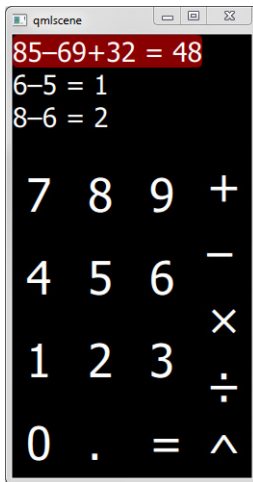
Le résultat final sera visuellement constitué de deux parties distinctes : d'une part l'historique des opérations, dans lequel l'utilisateur peut naviguer (par les touches directionnelles du clavier), d'autre part le pavé, comprenant les chiffres et les opérateurs. Ce clavier sera principalement constitué comme une grille, mais les chiffres et les opérateurs seront entièrement découplés (s'il faut afficher plus d'opérateurs par colonne qu'il n'y a de chiffres, nous pourrons ainsi décaler les deux parties l'une par rapport à l'autre, au lieu de déporter certains opérateurs en dehors de la zone d'affichage). L'utilisateur pourra interagir avec ce pavé aussi bien en cliquant sur les boutons qu'à l'aide du clavier.

Qt Quick est une technologie prévue pour faciliter le prototypage, c'est donc logiquement par là que nous allons commencer notre périple. Il ne nécessite aucune compilation et permet même d'éditer les fichiers sources à l'exécution, les changements étant apportés à l'interface en temps réel. Ensuite, nous introduirons de l'interactivité : l'utilisateur doit pouvoir savoir où il a cliqué et le résultat s'afficher. L'étape suivante consistera à mémoriser le résultat de toutes les opérations. Enfin, pour rendre la calculatrice plus ergonomique, nous permettrons la saisie des opérations au clavier ; là encore, l'utilisateur verra son action à l'écran.

Qt Quick 2.1 est arrivé pendant la rédaction de ce chapitre apportant son lot de nouveautés bien intéressantes, notamment pour adapter l'interface à la taille de la fenêtre : grâce aux Qt Quick Layouts, les éléments peuvent être répartis sur toute la surface, pas coincés dans un coin. La dernière étape consistera à ajouter de nouveaux opérateurs, chose pour laquelle la calculatrice n'était pas prévue au départ - le code s'y adaptera cependant très facilement.

Au fur et à mesure du développement, nous vous introduirons des outils de débogage pour vous donner les moyens de corriger à tout moment l'application.

Figure 13.1 : Résultat final



Attention > Ce chapitre se base sur les chapitres précédents du module Qt Quick, notamment l'[introduction au langage QML](#) et la [programmation déclarative](#), ainsi qu'à [JavaScript](#) et à la [programmation impérative](#). Très vite, l'[interactivité](#) sera requise, tant avec la souris que le clavier; l'affichage de l'historique utilise directement le patron de conception MVD et les [composants modèle-vue](#). Pour ceux qui préfèrent commencer par la lecture de l'exemple, des liens vous renverront vers les sections où sont traitées plus précisément ces notions.