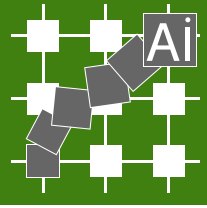
Eine Anwendung zur Darstellung verschiedener Fraktale mit Qt 5

Dimitri Tarnavski

18. Mai 2017



Was ist ein Fraktal

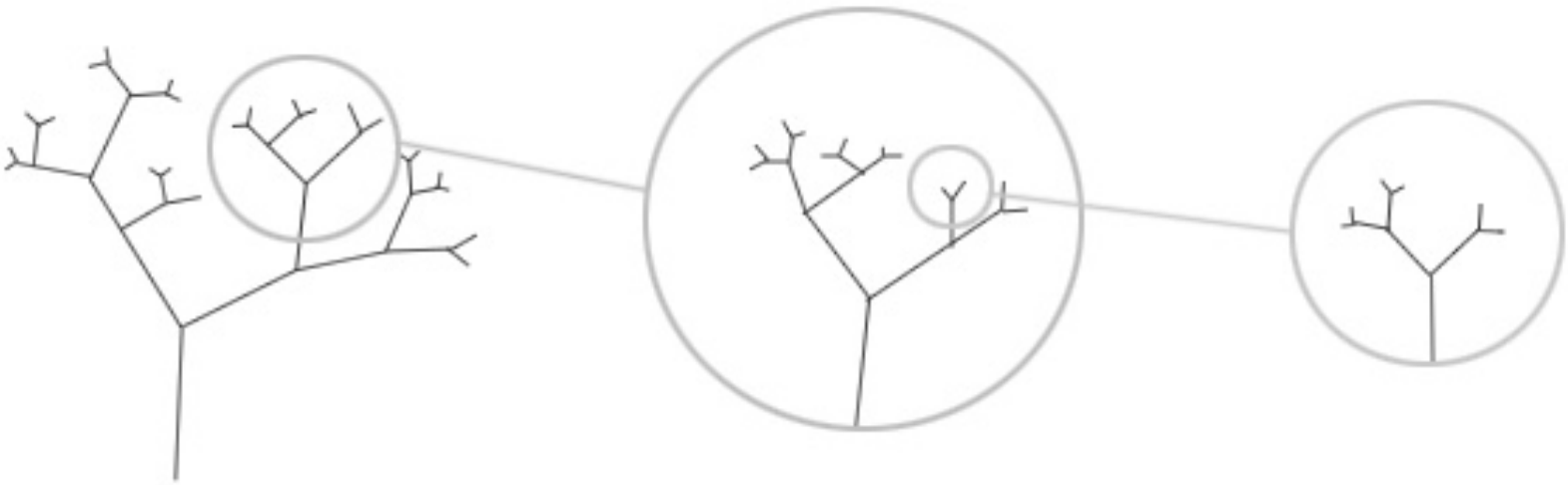
Drei konkrete Fraktale

Implementierung

Optimierungen

Was ist ein Fraktal?

- 1975 vom Mathematiker Benoît Mandelbrot geprägt
- Aus dem lat. *fractus* „gebrochen“, „in Teile gebrochen“
- Bezeichnet bestimmte *selbstähnliche* Muster/Strukturen



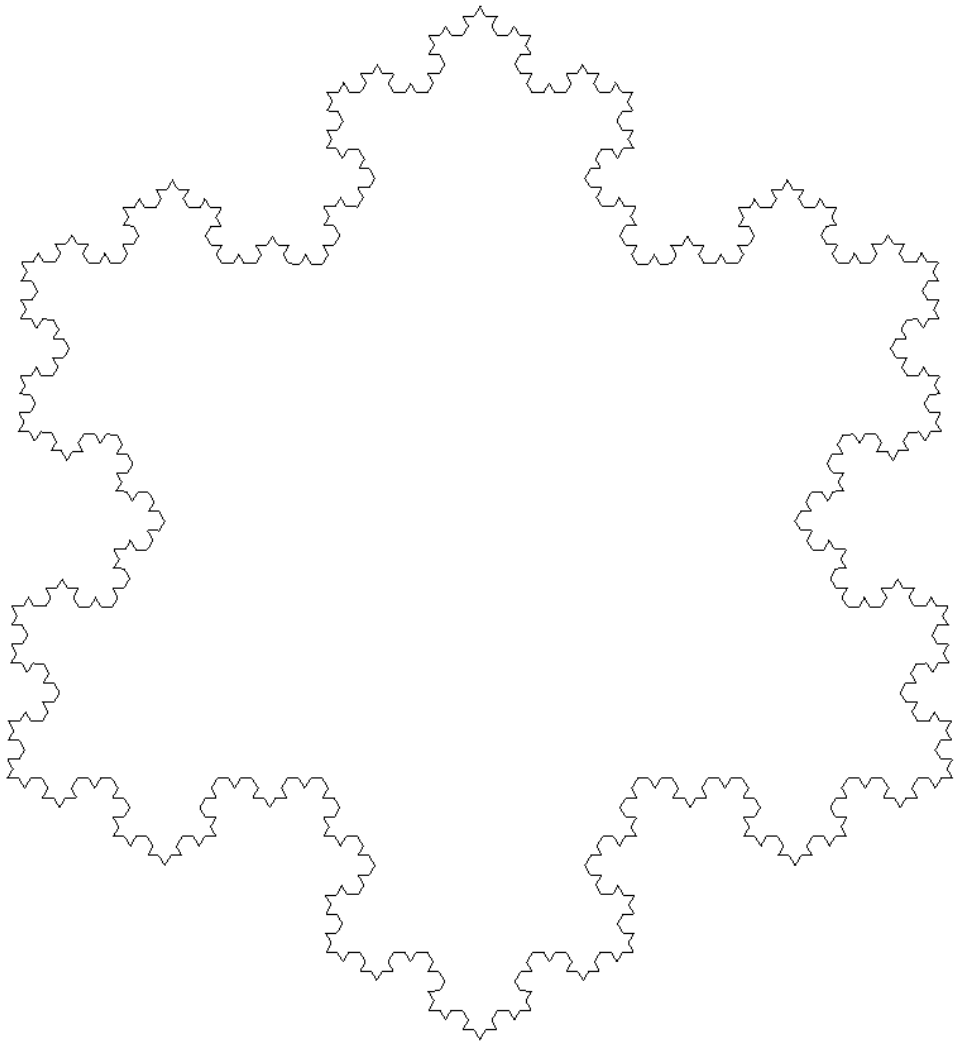
- Keine perfekte Kopie des Baums
- Eine Linie ist selbstähnlich, jedoch kein Fraktal

Fraktale Strukturen können oft in der Natur beobachtet werden:

- Bäume
- Blitze
- Berge und Küsten
- Blutgefäße

Eine fraktale Struktur ist immer mit rekursivem Verhalten verbunden.

Die Kochschneeflocke



Konstruktion der Kochlinie

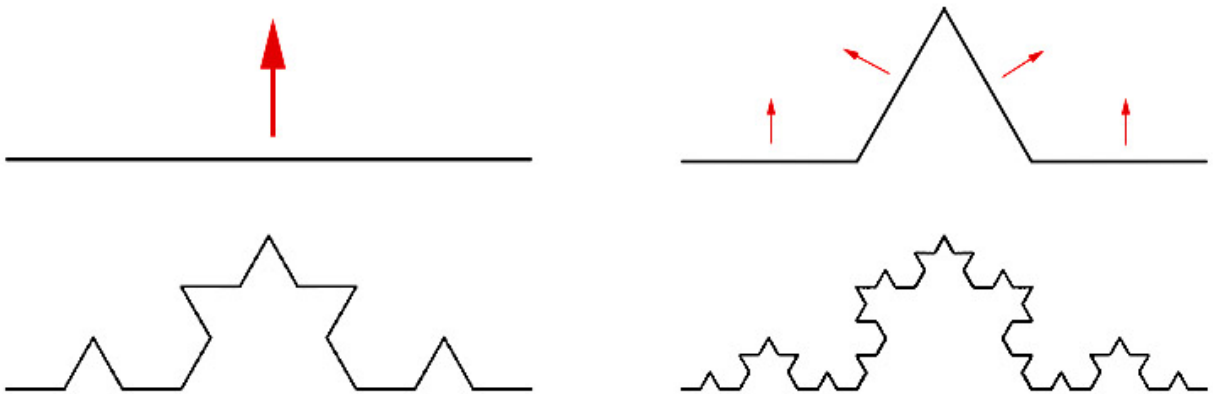
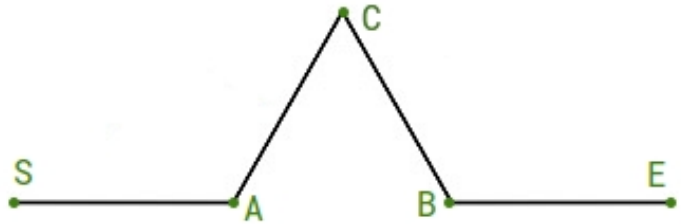


Abbildung: Konstruktion der Koch-Kurve, Quelle: Wikipedia

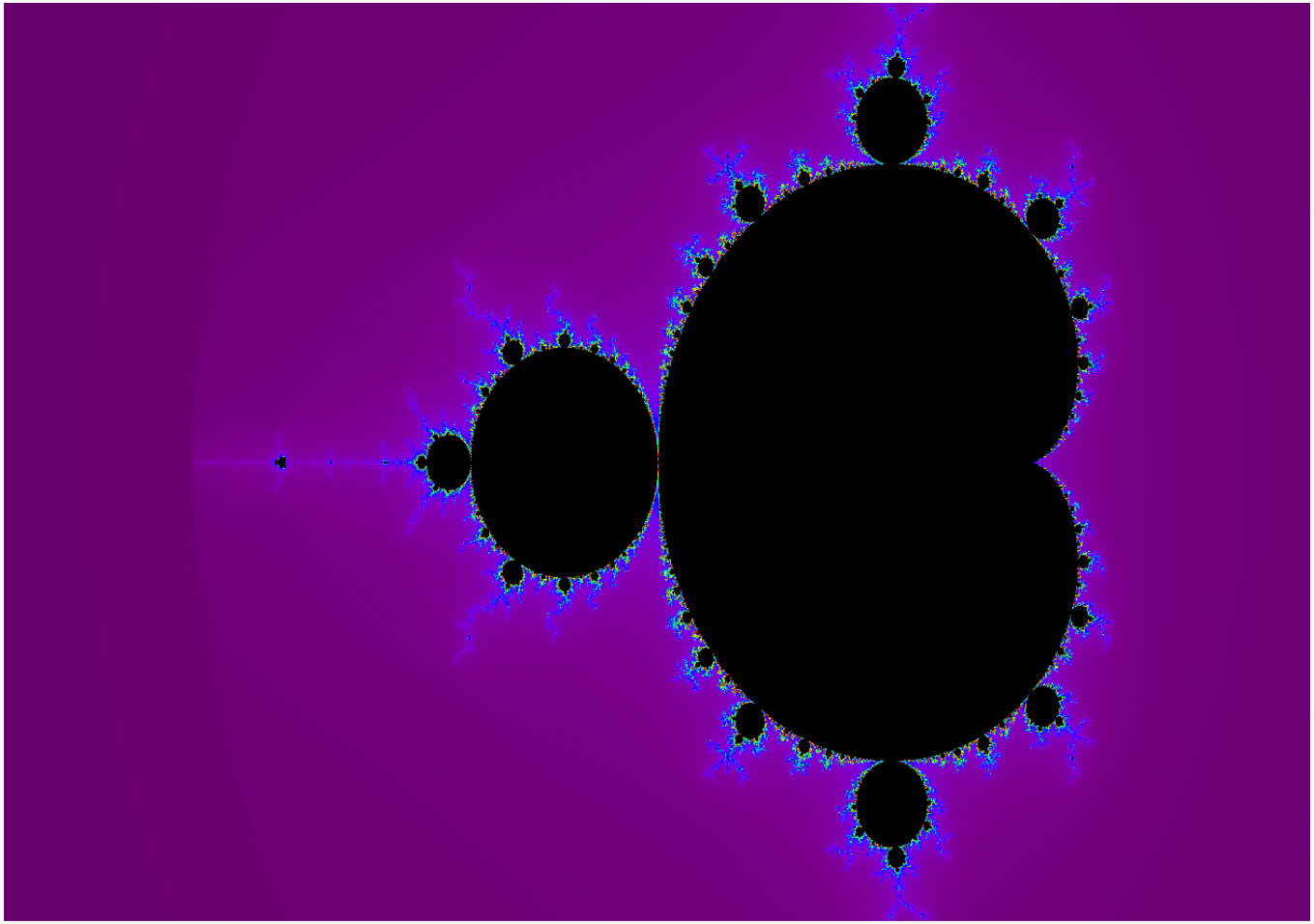
Berechnen der einzelnen Punkte



S = Anfangspunkt der Linie
E = Endpunkt der Linie
Diff = Differenz zwischen S & E

```
A.x = S.x() + Diff.x() / 3.0;
A.y = S.y() - Diff.y() / 3.0;

B.x = S.x() + 2.0 * Diff.x() / 3.0;
B.y = S.y() - 2.0 * Diff.y() / 3.0;

C.x = S.x() + Diff.x() / 2.0 - H * Diff.y() / 1.4;
C.y = S.y() - Diff.y() / 2.0 - H * Diff.x();
```

Die Mandelbrot-Menge



Definition der Menge

Die Mandelbrot-Menge wird definiert durch die rekursiv definierte Folge:

$$z_{n+1} = z_n^2 + c, z_0 = 0, c \in \mathbb{C}$$

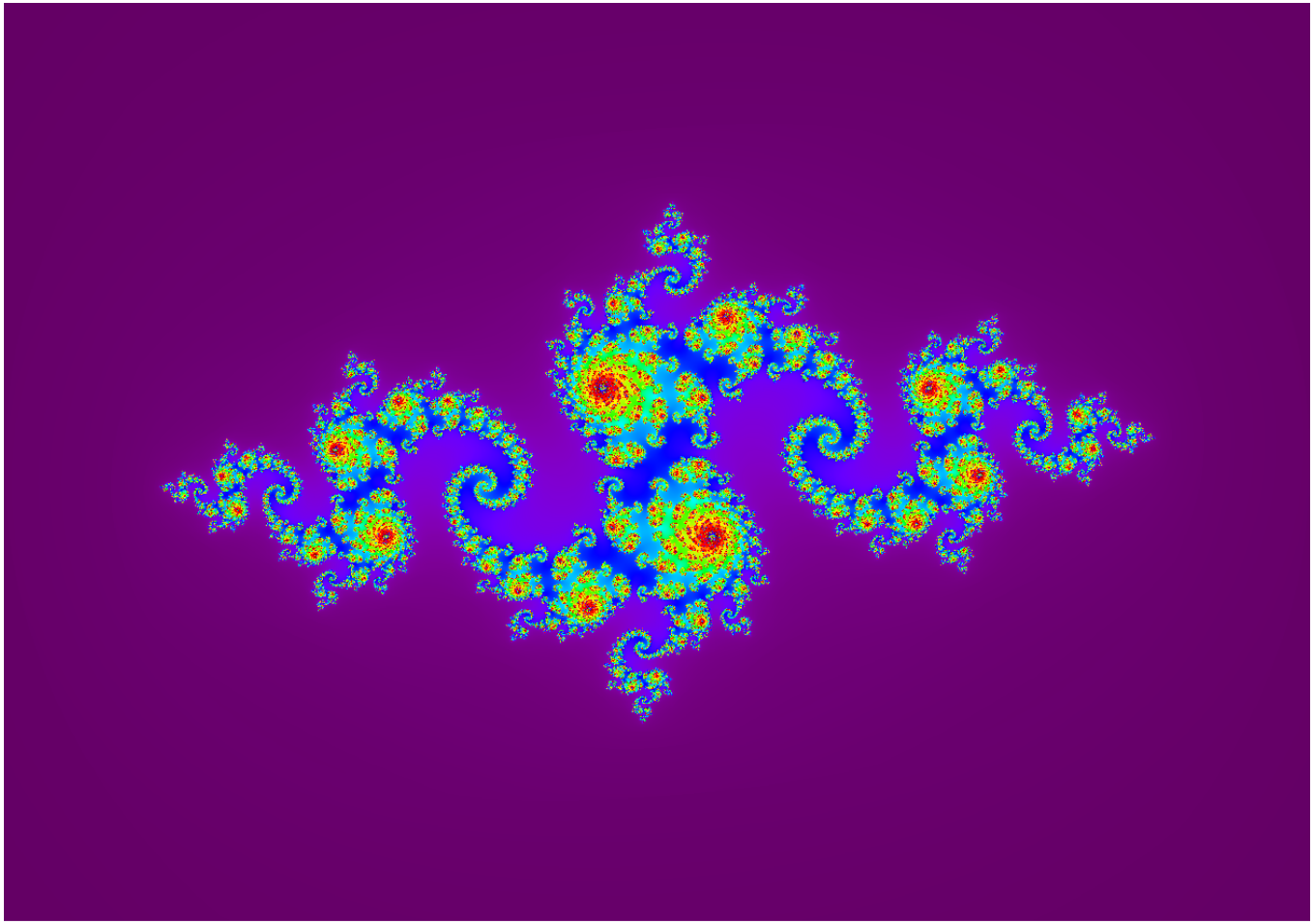
Zerlegt in Realteil und Imaginärteil:

$$x_{n+1} = x_n^2 - y_n^2 + c_x$$

$$y_{n+1} = 2 \cdot x_n \cdot y_n + c_y$$

```
for Alle Pixel (pX, pY):
  double x0 = pX abgebildet in [-2, 1]
  double y0 = pY abgebildet in [-1, 1]
  double x, y, iterations = 0;
  while(
    x * x + y * y < 4 && iterations < maxIterations
  ) {
    double xtemp = x * x - y * y + x0;
    y = 2 * x * y + y0;
    x = xtemp;
    iterations++;
  }
```

Die Julia-Menge



Definiert wie die Mandelbrot-Menge für ein festes c.

$$z_{n+1} = z_n^2 + c, z_0 = 0, c \in \mathbb{C} fixiert$$

- Es gibt viele Julia-Mengen.

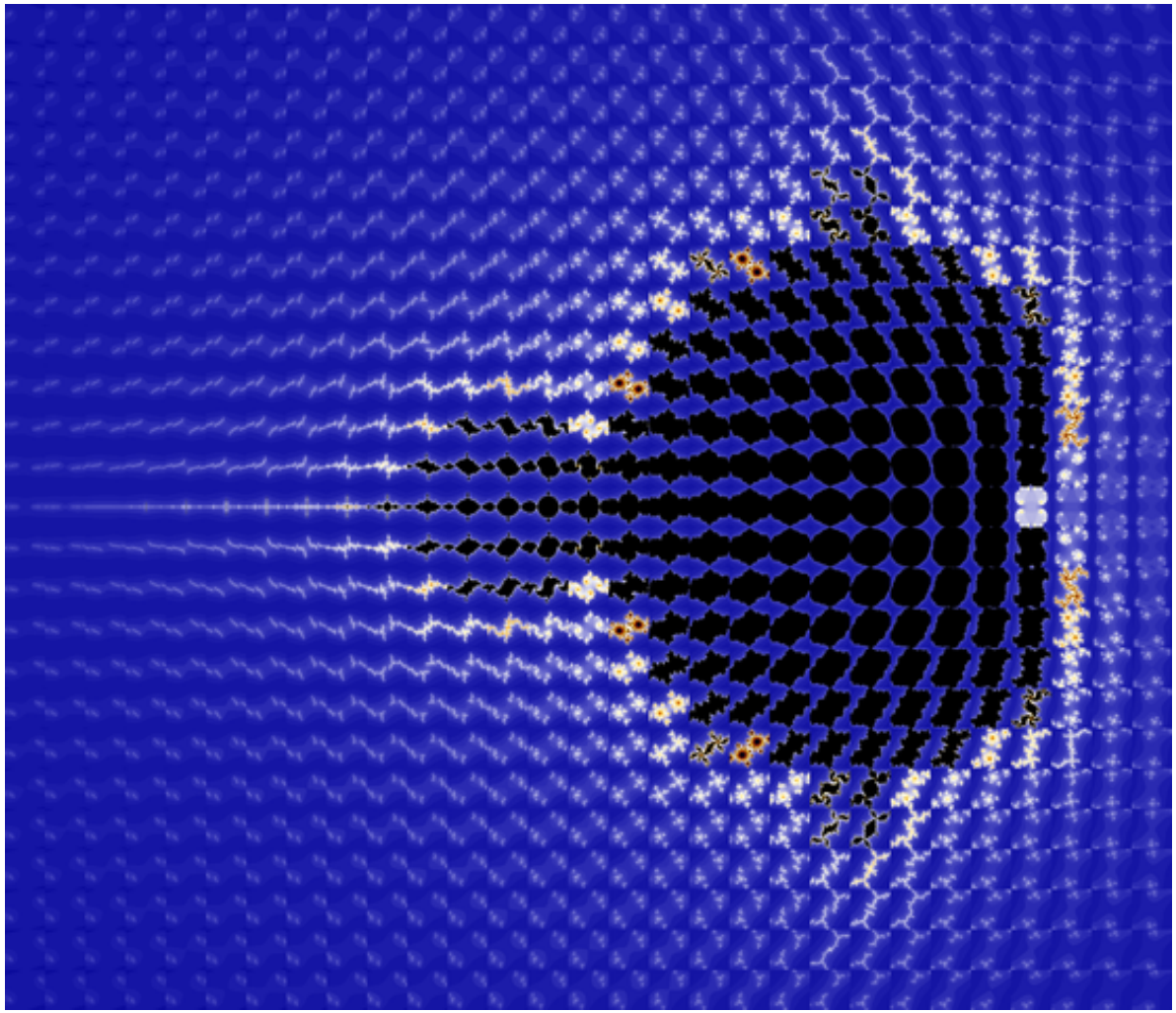
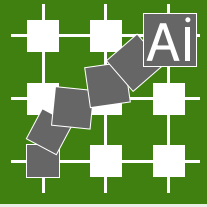


Abbildung: Julia-Mengen für verschiedene c, Quelle: Wikipedia



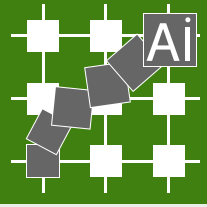
Das Klassendiagramm

Es gibt drei wichtige Klassen:

- Fractal - Elternklasse für alle Fraktale
- RenderTask - Elternklasse für Rendering-Aufgaben
- ColorMode - Elternklasse für Farbgebunden

Das Hauptfenster

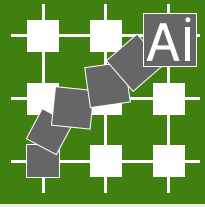
- Nimmt Benutzereingaben entgegen
- Ruft entsprechende Funktionen des aktuellen Fraktals auf
  - Z.B. Skalieren und Verschieben
- Stellt Funktionen zum Im-/Exportieren eines Fraktals



Hinzufügen eines Fraktals

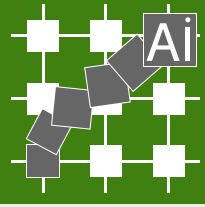
```
// Registriere die Fraktale, welche in der
// Applikation ausgewählt werden können.
registerFractal<EmptyFractal>(
  Fractals::ID::EMPTY_FRACTAL, "Keine Auswahl"
);
registerFractal<Mandelbrot>(
  Fractals::ID::MANDELBROT, "Mandelbrot-Menge"
);
...
```

```
std::map<
  Fractals::ID, std::function<Fractal::Ptr()>
> fractalFactory;
...
template<typename T>
void FractalWindow::registerFractal(
  Fractals::ID fractalID, QString label
) {
  // Fügt einen Eintrag zu der Combobox der Fraktale hinzu.
  fractalsCombo->addItem(label, QVariant::fromValue(fractalID));
  // Speichert eine Funktion zum Erzeugen eines neuen Fraktals,
  // welche bei Bedarf durch Ausführen ein neues Fraktal generiert.
  fractalFactory[fractalID] = [this] () -> Fractal::Ptr {
    return Fractal::Ptr(new T(
      canvas.size().width(), canvas.size().height()
    ));
  };
}
```



Reaktivität der Einstellungen

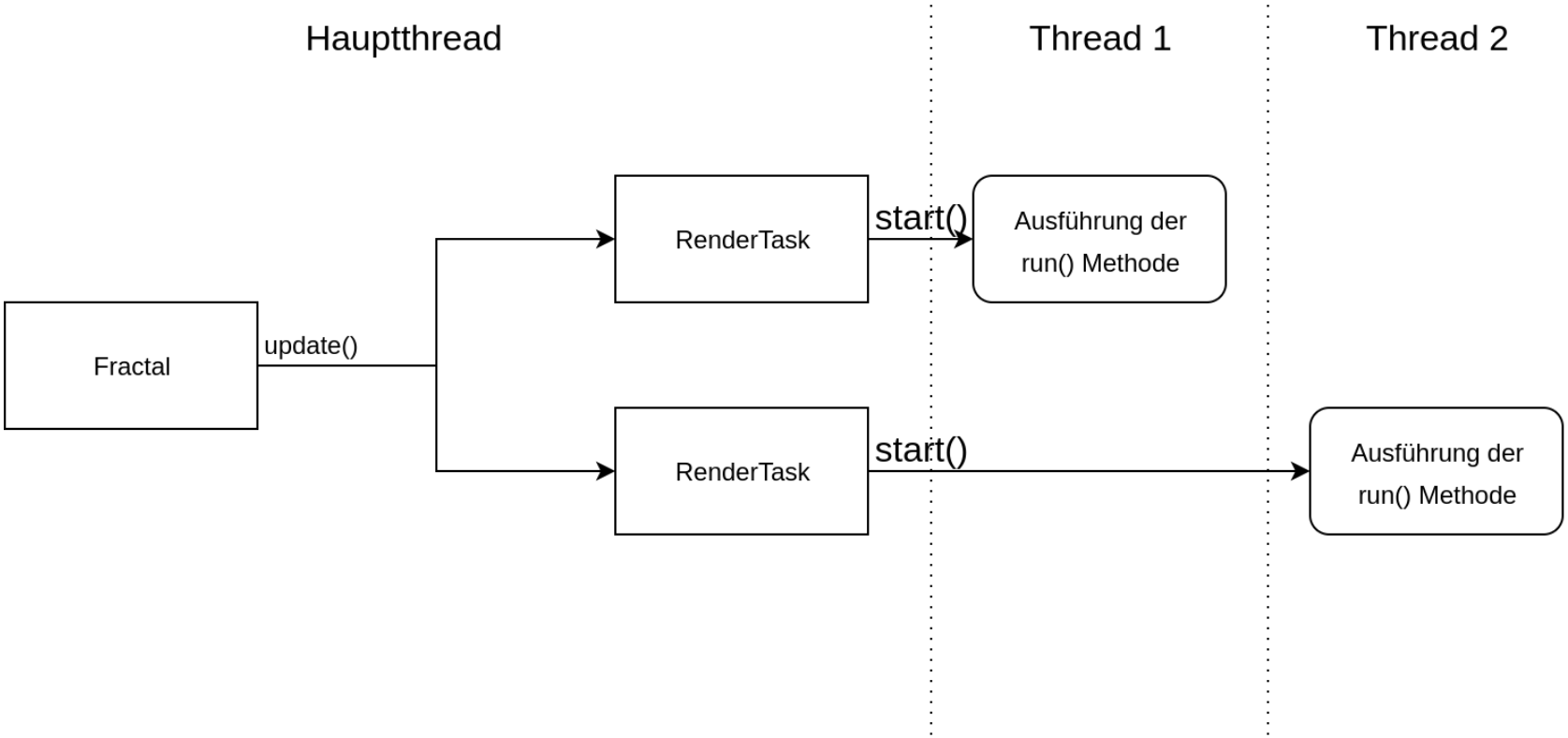
```
QSpinBox* iter = new QSpinBox;
...
connect(
  iter, static_cast<void(QSpinBox::*)(int)>(
    &QSpinBox::valueChanged
  ),
  this, &Fractal::setMaxIterations
);
connect(
  this, &Fractal::iterationsChanged,
  iter, &QSpinBox::setValue
);
```



Verschieben eines Fraktals

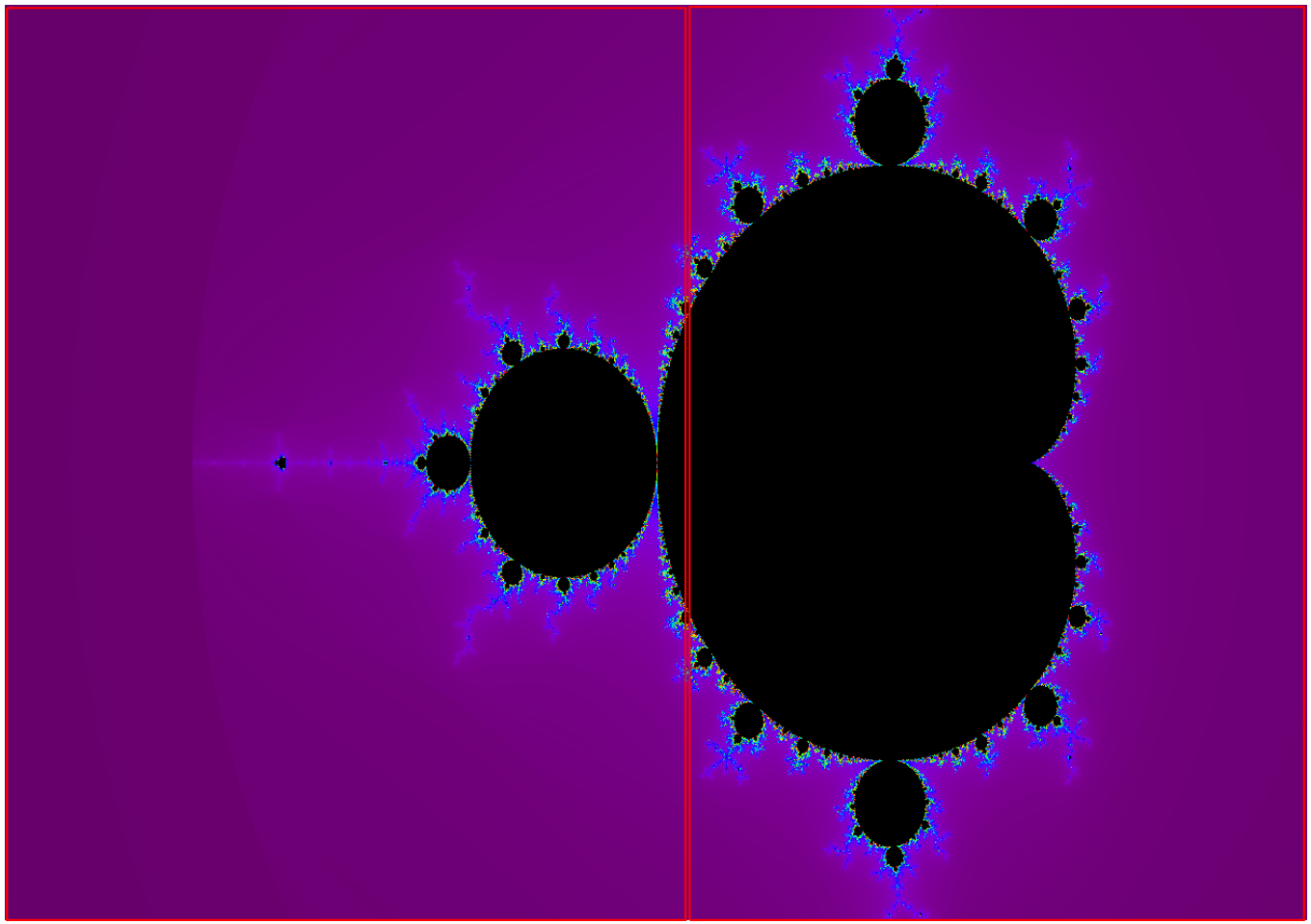
```
// Erstelle eine Kopie des Bildes
QImage traslated(image);
QPainter painter(&image);

// Fülle das Bild mit der Farbe Schwarz
image.fill(Qt::black);
// Zeiche das Bild versetzt
painter.drawImage(offset, traslated);
```

QThread



- QThread ist eine Verwaltungs-Klasse eines Threads
- Startet einen Thread mit der Funktion *QThread::start()*

QThread



Mandelbrot- & Julia-Menge

- Ignorieren der schwarzen Flächen
- Interpolation zwischen zwei Pixeln

Kochkurve

- Einführen von Clipping
- Abbrechen der Berechnung, wenn Linie kleiner als 1px