

# Пользовательские трансформы

Если пользователь еще не успел ничего трансформировать, то итоговый пользовательский трансформ **MP** равен:  $[1, 0, 0, 1, 0, 0]$ .

Теперь допустим пользователь сначала сместил, затем масштабировал и в конце повернул карту на 90° против часовой стрелки относительно центра координат, тогда итоговый пользовательский трансформ можно записать в виде произведения линейных операторов:

$$\mathbf{MP} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 30 \\ 0 & 1 & 55 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

избавимся от начальной единичной матрицы (умножение на нее не меняет результат):

$$\mathbf{MP} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 30 \\ 0 & 1 & 55 \\ 0 & 0 & 1 \end{bmatrix}$$

rotate ←

scale ←

shift

Обратите внимание на то, что порядок перемножения линейных операторов важен:

$$\mathbf{MP} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{MP} = \begin{bmatrix} 1 & 0 & 30 \\ 0 & 1 & 55 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{MP} \quad \text{сначала пришло смещение}$$

$$\mathbf{MP} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{MP} \quad \text{затем пришло масштабирование}$$

$$\mathbf{MP} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \mathbf{MP} \quad \text{и в конце пришел поворот}$$

Дело в том, что, чтобы получить координаты точки с учетом всех пользовательских трансформов мы умножаем матрицу итоговых пользовательских трансформов на точку, а не наоборот. То есть при умножении точка будет справа от матрицы:

$$\text{Point}_{\text{new}} = \mathbf{MP} \cdot \text{Point}$$

Что равносильно:

$$\text{Point}_{\text{new}} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 30 \\ 0 & 1 & 55 \\ 0 & 0 & 1 \end{bmatrix} \cdot \text{Point}$$

rotate ←

scale ←

shift

↙

То есть точка сначала сместится, затем масштабируется и в конце повернется.

А так как перемножение матриц некоммутативно, то любой порядок линейных операторов, отличный от указанного выше, приведет к тому, что точка окажется не в том месте, где ожидается.