What associativity is about

We all know what it means for a binary operator Δ to be associative:

$$(a \,\Delta\, b) \,\Delta\, c \;=\; a \,\Delta\, (b \,\Delta\, c) \quad .$$

Now, if you interview people, asking them what associativity is really about, one of the first answers you get is that it is allowed to omit the parentheses and write

$$a \,\Delta\, b \,\Delta\, c$$

without introducing ambiguity. And this is right!
Next you may get the answer that thanks to the associativity it does not matter whether the value of $a \,\Delta\, b \,\Delta\, c$ is computed from left to right or from right to left. And you may get several other answers, but hardly ever the one to be explained next.

*   *
      *

For people that calculate, the property of associativity may offer very strong heuristic guidance in proof construction : if, in a calculation, the expression

$$a \,\Delta\, b \,\Delta\, c$$

enters the picture as

$$(a \,\Delta\, b) \,\Delta\, c \quad ,$$

then the advice for the next step(s) is to focus on the subexpression $b \,\Delta\, c$ in

$$a \,\Delta\, (b \,\Delta\, c) \quad .$$

In our experience, this heuristic rule has worked well in a tremendous number of cases. And if you come to think of it: of course, what else could associativity be about?

*   *
*

The proof of the pudding is in the eating, and we shall now present the proof of a FANTASTIC theorem that our colleague Paul F. Hoogendijk challenged us to prove one day. The theorem is FANTASTIC to the extent that we have absolutely no idea what the mathematics involved "means".

In what follows, constant Γ and dummies x and y are of type Thing. Further, Λ and E map Things to Things, and, last but not least, binary operator $\circ$, which maps two Things on a Thing, is associative. In short, in the ensuing expressions there is no type conflict.

Now, given (0) and (1)

(0) $x = \Lambda.y \;\equiv\; \Gamma \circ x = y$ $\qquad (\forall x, y)$
(1) $\mathrm{E}.x \;=\; \Lambda.(x \circ \Gamma)$ $\qquad (\forall x)$

we have to prove

(2) $\mathrm{E}.x \circ \Lambda.y \;=\; \Lambda.(x \circ y)$ $\qquad (\forall x, y)$

Proof Property (0) is a so-called *Galois Connection* and the advice is that we then always write down the corresponding *cancellation rules* as well [?]. Here they are (instantiate (0) with $x := \Lambda.y$ and $y := \Gamma \circ x$, respectively, so as to make one side vacuously true) :

(3) $\Gamma \circ \Lambda.y \;=\; y$ $\qquad (\forall y)$
(4) $x \;=\; \Lambda.(\Gamma \circ x)$ $\qquad (\forall x)$

Now we tackle (2) and, for this occasion, we shall make the steps that appeal to the associativity of $\circ$ explicit.

$$
\begin{array}{ll}
 & \mathrm{E}.x \circ \Lambda.y \;=\; \Lambda.(x \circ y) \\
\equiv & \qquad \{\ (0)\text{ from left to right }\ \} \\
 & \Gamma \circ (\mathrm{E}.x \circ \Lambda.y) \;=\; x \circ y \\
\equiv & \qquad \{\ \circ \text{ is associative }\ \} \\
 & (\Gamma \circ \mathrm{E}.x) \circ \Lambda.y \;=\; x \circ y \\
\equiv & \qquad \{\ (1)\text{ to eliminate } \mathrm{E}\ \} \\
 & (\Gamma \circ \Lambda.(x \circ \Gamma)) \circ \Lambda.y \;=\; x \circ y \\
\equiv & \qquad \{\ (3)\text{ with } y := x \circ \Gamma\ \} \\
 & (x \circ \Gamma) \circ \Lambda.y \;=\; x \circ y \\
\equiv & \qquad \{\ \circ \text{ is associative }\ \} \\
 & x \circ (\Gamma \circ \Lambda.y) \;=\; x \circ y \\
\equiv & \qquad \{\ (3)\ \}
\end{array}
$$

$$x \circ y = x \circ y$$

$$\equiv \quad \{ \quad \}$$

$$true \quad .$$

End Proof .