

Implementierung einer Datenbank REDO

Grundsätzlich wäre mein Ansatz das Command Pattern einsetzen. Das Beispiel, welches ich in SWP-OP erarbeitet habe, war ein binärer Zahlen Rechner welcher die Rechenoperationen der Addition und der Subtraktion beherrscht hat. Ein sofort ersichtlicher Punkt ist, dass hier jeweils eine Operation die andere rückgängig macht indem man sich einfach die Funktions Parameter der Kommandos in historisch richtiger Reihenfolge merkt.

Wenn ich bei meiner Rechnung also gerade den Ausgangsstand 3 habe und den Operationsparameter 1 welchen man abzieht. Man erhält die Zahl 2.

$$3 - 1 = 2$$

Die Operation und die Werte wurden in gespeichert. Nun will ich diesen Schritt rückgängig machen.

Mein Ausgangsstand ist der Wert 2. Die Umkehrung von Subtraktion ist die Addition. Ich addiere also 2 mit dem Operationsparameter 1 und erhalte den Ursprünglichen und gleichzeitig neuen Ausgangsstand 3.

$$2 + 1 = 3$$

Dieselbe Methodik werde ich bei dem Datenbank Rado anwenden. Anstelle von simplen mathematischen Operationen werde ich gezielt zu jeder Methode, welche die Datenbank verändert eine dazu passende Umkehroperation implementieren. Auch des Ausgangsstand wird deutlich komplexer und der Umfang der Operationsparameter deutlich größer sein als im Beispiel davor aber das Prinzip bleibt gleich.

BSP:

Wenn ich eine Funktion implementiere welche einen Eintrag in eine Tabelle „user_items“ erzeugt gibt, dann wird es die Funktion geben, in der ich diesen Eintrag wieder lösche. Aber dafür notwendigen Parameter gehen niemals verloren. (Außer Reboot etc.)