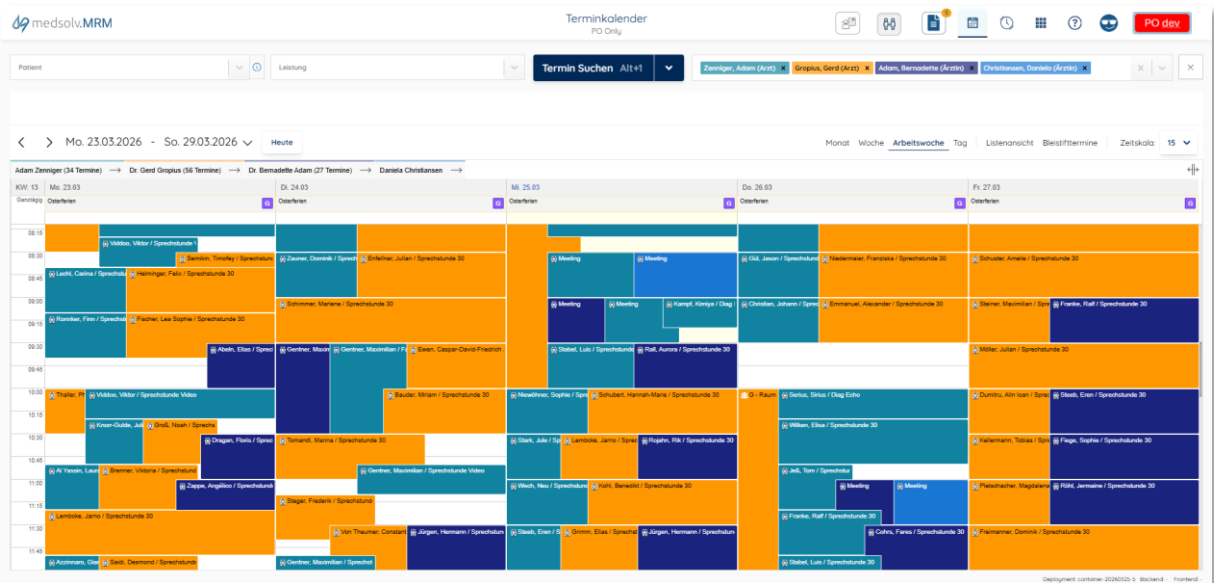


Die Darstellung von Terminen bei überlappenden Kalendern

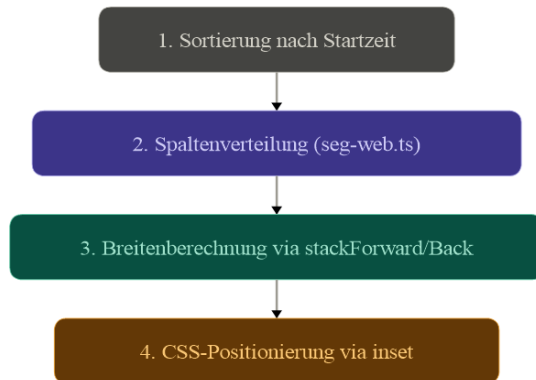


Im Folgenden wird das Prinzip erklärt. Zusätzlich wurde die Darstellung in einigen Details weiter angepasst.

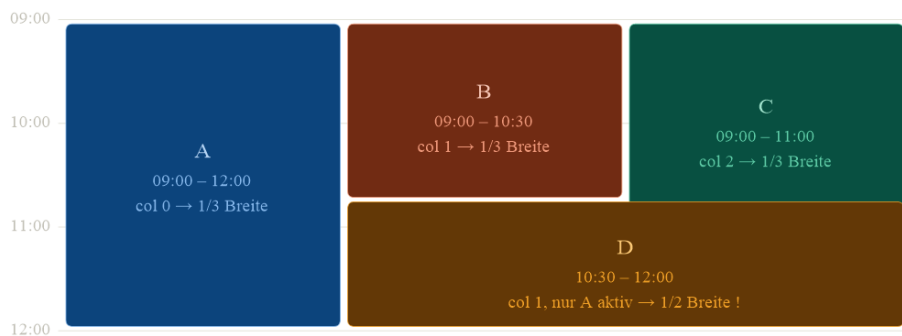
Segment = Termin Platzierung

Die zugrunde liegende Berechnung erfolgt durch eine Library „FullCalendar“.

FullCalendar (TimeGrid-Ansicht) verwendet für überlappende Termine einen mehrstufigen **Segment-Placement-Algorithmus**, der nicht simpel "gleiche Breite für alle" rechnet, sondern eine [graphenbasierte Logik](#) anwendet.



Beispiel: Spaltenverteilung mit 3 Terminen



$$\text{Breite} = 1 / (\text{stackForward} + 1 + \text{stackBackward})$$

stackForward = Anzahl Termine, die rechts davon beginnen

stackBackward = Anzahl Termine, die links davon laufen

Das ist der Kern der Logik – hier die Details im Einzelnen:

Details

Schritt 1: Sortierung Alle Termine werden nach Startzeit sortiert (bei Gleichstand nach dem eventOrder-Parameter).

Schritt 2: Spaltenverteilung (seg-web.ts) FullCalendar weist jedem Termin eine virtuelle "Spalte" zu. Das Prinzip: Ein Termin kommt in die erste freie Spalte, in der er zeitlich nicht mit einem bereits platzierten Termin kollidiert. Das entspricht dem klassischen Graphen-Färbeprobem (Interval Graph Coloring).

Schritt 3: Breitenberechnung über stackForward / stackBackward Jeder platzierte Termin bekommt ein rect-Objekt mit den Werten stackForward (wie viele Termine rechts von ihm starten) und stackBackward (wie viele Termine links von ihm noch laufen). Die Breite ergibt sich dann vereinfacht als:

$$\text{Breite} = 1 / (\text{stackBackward} + 1 + \text{stackForward})$$

Das erklärt das scheinbar "ungerechte" Verhalten: Wenn Termin D erst beginnt, wenn Termin C schon vorbei ist, hat D weniger Nachbarn als A, B, C – und bekommt daher eine **größere Breite**, obwohl D zeitlich mit A überlappt.

Schritt 4: CSS-Positionierung via inset Die berechneten Breiten und Offsets werden als CSS inset-Property (früher left/right) gesetzt. Ein Termin bekommt die Klasse fc-timegrid-event-harness-inset, wenn sein stackForward > 0 ist –wenn er also "versetzt" dargestellt wird.

Das Schlüsselprinzip

FullCalendar berechnet die Breite **nicht global** für eine Clique (alle sich überschneidenden Termine), sondern **lokal pro Termin** basierend auf seinen direkten Nachbarn zum Zeitpunkt seiner Platzierung. Daher können zwei Termine mit identischer Startzeit trotzdem unterschiedlich breit erscheinen, wenn ihre zeitlichen Kontexte verschieden sind.