An alignment shall determine the set of parameters P. It uses a set of constraint equations, C and data from tracks, D. The data taking may cover a long period of time and a subset of the parameters may change over the course of that time. The problem can be split in N time intervals $\tau_i = [t_i, t_{i+1})$ of arbitrary length $t_{i+1} - t_1$, where $i = 0, 1, 2 \ldots, N$. The set of parameters may be sliced into two subsets of parameters: P_{fr} is the subset of parameters hat are constant over the full range of the time and P_i the subset of parameters that may change but are assumed to be constant within a time interval τ_i. A similar split may be required for the constraint equations C. For any given interval τ_i, the union $P_{\text{fr}} \cup P_i$ describes the detector completely and the union of constraint equation, $C_{\text{fr}} \cup C_i$ is a complete set of constraints at that time.

The data will be partitioned at the same time boundaries and the following holds:

$$D = \bigcup_{i=1}^{N} D_i$$

Now, the full set of parameters to be determined for the whole time range will be

$$P_{\text{total}} = P_{\text{fr}} \cup \left(\bigcup_{i=1}^{N} P_i \right)$$

and likewise the set of constraint equations is

$$C_{\text{total}} = C_{\text{fr}} \cup \left(\bigcup_{i=1}^{N} C_i \right)$$

With this setup, the alignment algorithm can determine all the parameters and the positions for all parts at any time.

This setup has the following advantages: Assuming the detector consists of a hierarchy, like modules mounted on some rigid structure that make up an even bigger structure. This is the case for all subdetectors in CMS, e.g. the pixel barrel is split into two separate half barrels, made of three layers, made of ladders, and on each ladder, 8 modules are mounted in a row. The number of hits collected in a certain time-interval τ_i changes with the acceptance. The bigger structures collect more than the smaller ones, e.g. one half barrel sees 50% of the hits of the full barrel, but one module sees a fraction of about 1/768 hits of the full barrel.

The partitioning made above helps in this case. We know that the most likely change over time are movements of the big structures, e.g. the two half-barrels of the pixel. Within the half-barrels, the positions of the modules are fairly constant. It is therefore favourable to use a hierarchy constraint and determine the positions of the half barrels plus the modules. But the module positions are determined w.r.t. the half barrel they are mounted on. As $|D_i| < |D|$ if $|D_i| > 0 \, \forall i$, the number of tracks collected by one module in one interval can be fairly small, but the half-barrel still sees about 400 times more hits. Therefore it is a natural choice to have the module parameters $\in P_{\text{fr}}$ and the half-barrel parameters in every P_i. The constraint equations C_i will make sure that any time-evolution in the half-barrel movements will only be in all P_i and not in the module positions. The module positions will receive the full hit statistics collected over the full time range and hence will be determined with maximum precision.

In the Millepede implementation in CMS, the framework takes care to make the required split of parameter sets and the sets of constraint equations. After the alignment constants have been determined, the software framework will generate full geometry descriptions for each time interval. With this, the reconstruction algorithm can use the best geometry information available at any time interval an alignment has been determined.