

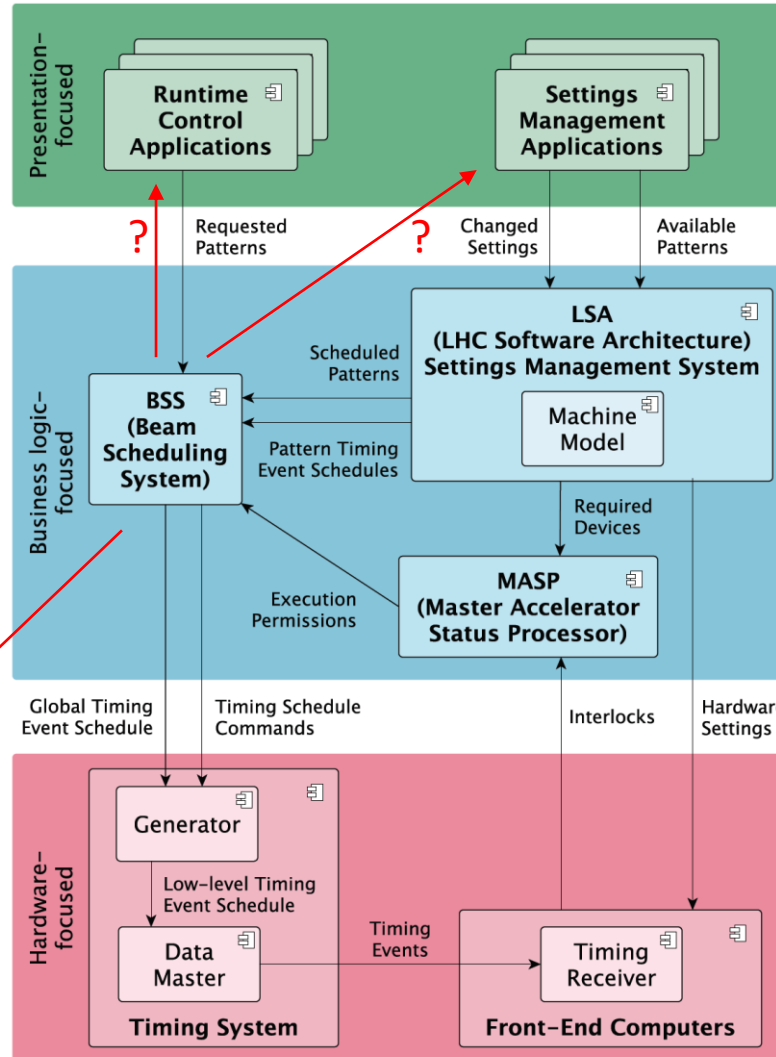
UNILAC Timing Overview

BSS GUI App:
UNILAC timing values -> to BSS

FPGA:
UNILAC timing values
SIS18 pattern(groups)
-> UNILAC schedule

???:
chain timing graph(s)
UNILAC schedule
-> timing graph

UNILAC Pulszentrale



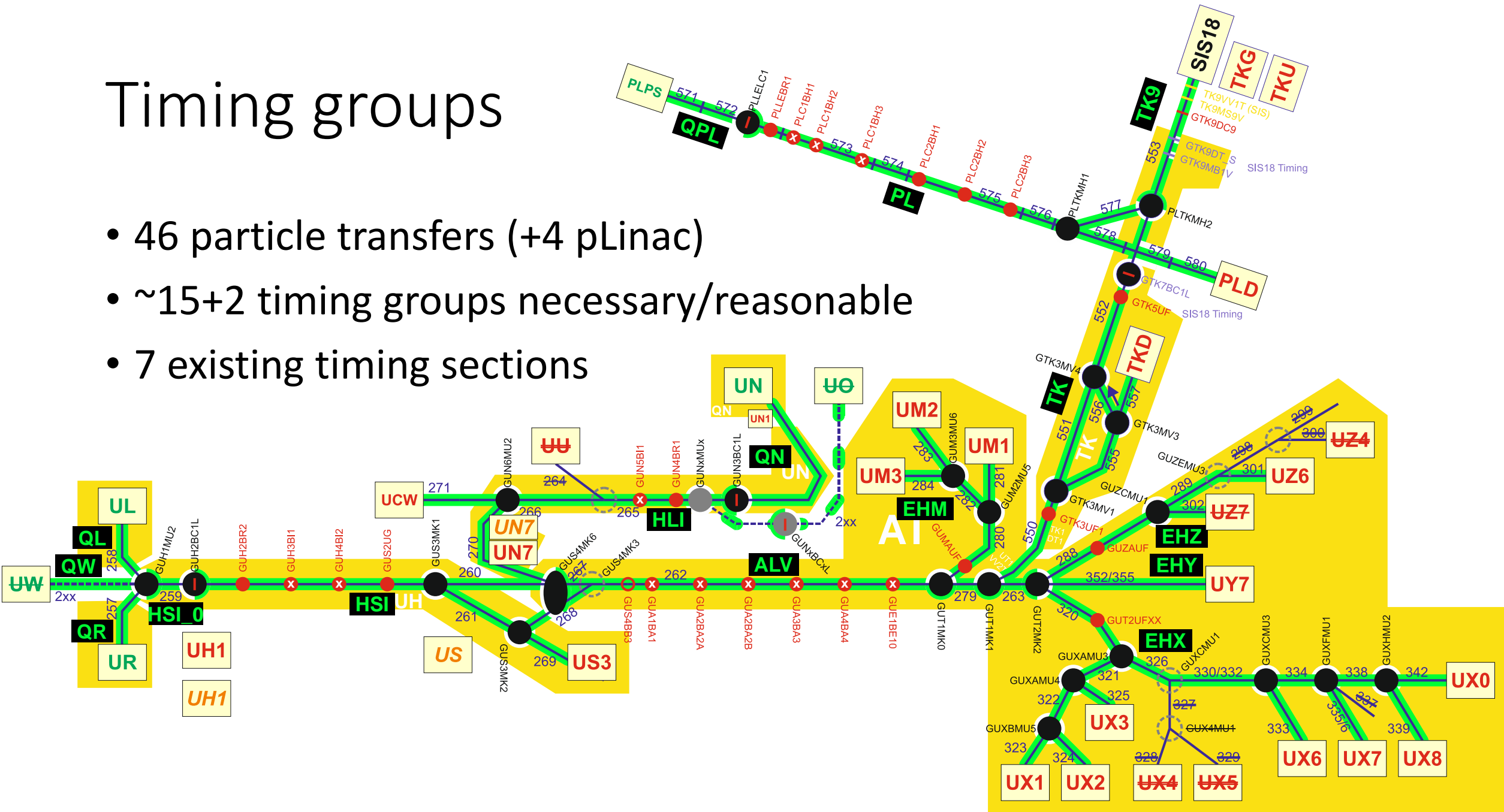
Scheduling App:
initial values, ... -> chain

MakeRule:
chain -> chain timing graph(s)

MASP:
Interlocks
Execution states (PG-Schutz, dry exec)
-> to BSS, to Data Master?

Timing groups

- 46 particle transfers (+4 pLinac)
- ~15+2 timing groups necessary/reasonable
- 7 existing timing sections



UNILAC timing parameters for chains & sources

Chains (accelerators):

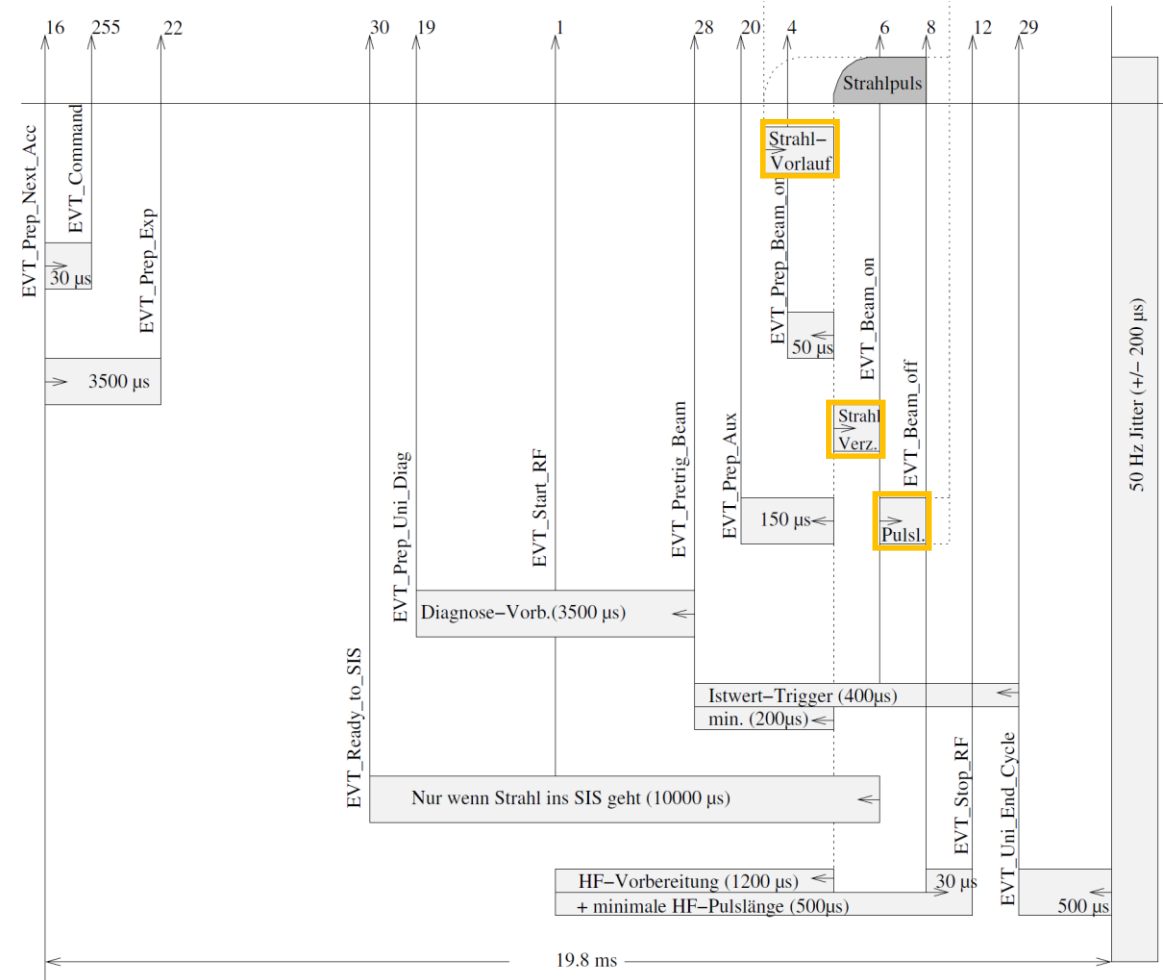
- beam pulse length (Pulslänge)
- (good) beam delay (Strahlverzögerung)
- prepulse / lead beam (Strahlvorlauf)

Sources:

- source pulse length (Quellenpulslänge)
- source lead (Vorlauf?)

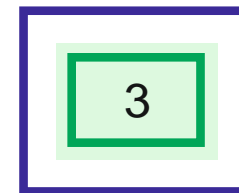
Comments:

- timing of chain and source are interrelated
- timing parameters are independent of other settings and schedule (and vice versa)



Generation of UNILAC chain timing graph

- Scheduling App: Initiate generation of chain (source), assign default values for timing parameters
 - Beam production chains need source
- LSA: MakeRule generates chain (source) timing graphs based on parameters and event dependencies
 - with beam (source)
 - without beam
 - with short beam pulse (PG-Schutz)
 - or all in one graph with conditional branching
- Accordingly for sources and beam out-processes
 - RF stabilization and conditioning
 - TK preparation

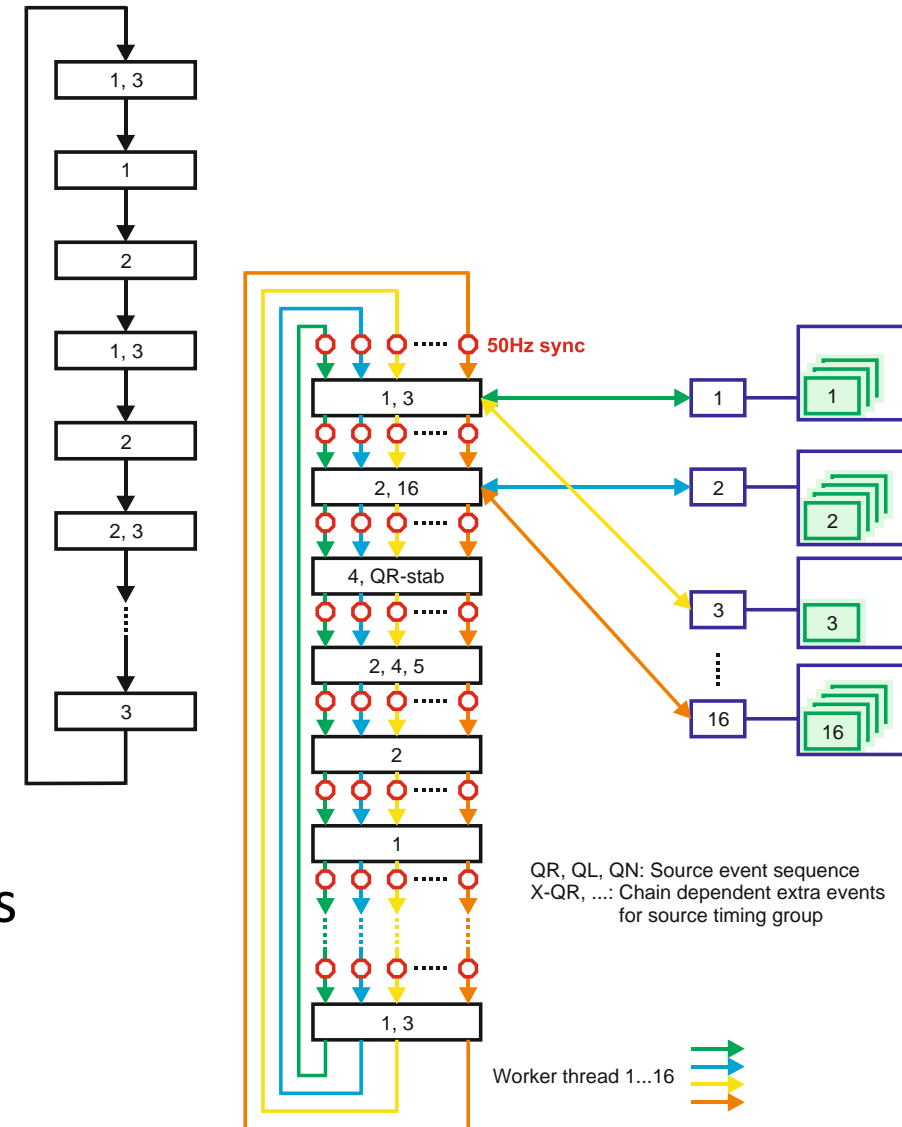


Timing operation 1

- Chain(s) introduced to BSS, timing graphs sent to BSS
- Change of parameters (pulse length) via Paramodi -> LSA trim -> new graph -> ...

Generation of UNILAC schedule & timing graph

- App/GUI?: Set schedule parameters
 - Select defined chains to be executed
 - Provide parameter values for execution:
 - Repetition count (could be defined by other quantity)
 - optional: special flags (w/o beam, strictly periodic, ...)
 - Initiate schedule generation
 - manually
 - on demand (change of SIS pattern(groups))
- FPGA:
 - takes UNILAC execution parameters and injections from SIS pattern(groups)
 - mimics Pulszentrale to produce a periodic schedule
- Combine Schedule from FPGA and chain timing graphs into comprehensive timing graph?



Chain and source timing

