Data Supply:
LSA and Applications

GSI Operator Training Course, 12.04.2018

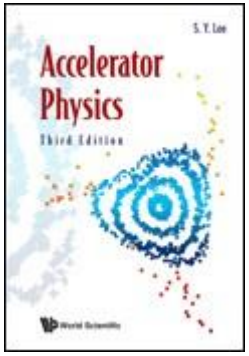
D. Ondreka, FAIR-DV

Outline

- Data Supply
- LSA
  - Overview
  - Basic structures
- Machine model
- Applications
  - SchedulingApp
  - ParamModi

Data Supply

- Operation of a complex accelerator requires consistent set values for many devices
  - Intricate relations between physics and device parameters (e.g. tune <-> quad currents, bucket size <-> RF voltage)
  - Set values need to be executed synchronously for the accelerator to work properly
  - Modifications of set values to adjust the accelerator based on observations are necessary
- Scope of Data Supply
  - Calculation of consistent set values for all devices of the accelerator based on a **machine model** providing an abstraction of the accelerator in terms of **physics parameters**
  - Provision of **tuning parameters** for adjusting the accelerator to compensate unavoidable deviations from the ideal machine behaviour
  - Provision of **applications** to initialize and manipulate the set values of the accelerator



Machine Model

Tasks:

- Representation of the accelerator by a set of **physics parameters**, hiding hardware details of individual devices
  - Beam parameters (E, A, Q) instead of dipole currents or frequencies
  - Tunes (Q_h, Q_v) instead of quadrupole currents
  - Angles instead of steerer currents
- **Algorithms** for conversion of physics values to hardware values
  - Integration of accelerator physics knowledge
  - Integration of device properties
  - Checking of hardware limitations
  - Determination of a consistent time evolution
- **Tuning parameters** for deviations from the ideal machine
  - Anticipation of possible deviations
  - Development of correction schemes
  - Interface for beam based corrections

Prerequisites:

- Device related data
  - Hardware limits (min/max values)
  - Calibration curves
  - Conversion factors
- Machine physics data
  - Accelerator geometry
    - Circumference
    - Distances between devices
  - Ion optical data
    - Optical strength of magnets
    - Twiss parameters
    - Global optics parameters (tune, chromaticity)

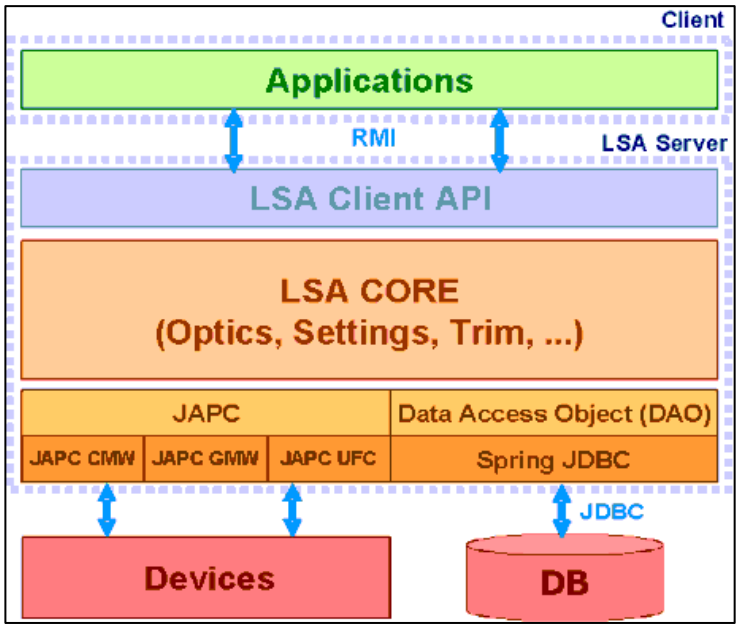
How can such a machine model be implemented?

LSA: Overview

- Framework for accelerator modeling
  - Developed by CERN and now maintained and extended by CERN/GSI collaboration
  - Java based 3-tier architecture
  - Generic structures for representing accelerators
  - Management of device and layout data in DB

- Advantages of LSA
  - Modern, maintainable architecture
  - Separation of physics and software technique
  - Access to all data via standard Java interfaces
  - Business logic separated from applications
  - All set values in DB including history

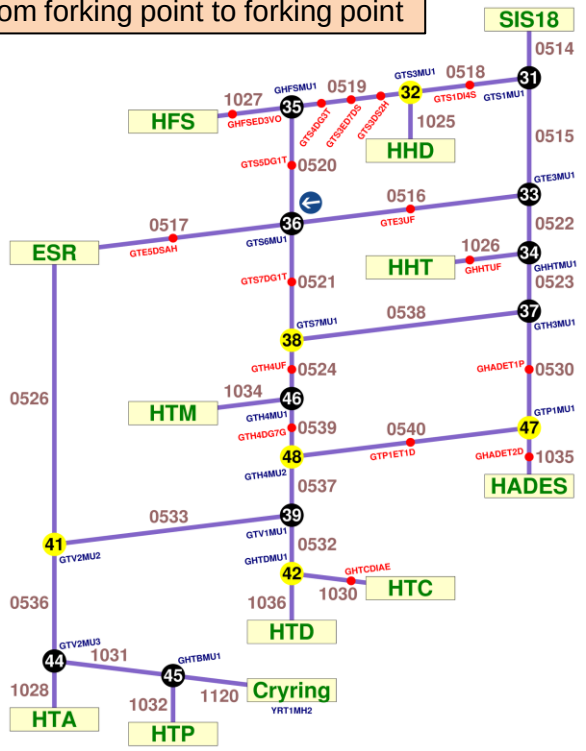
LSA Software Architecture



LSA: Basic Structures (I)

- Particle Transfer
  - Section of accelerator with identical timing
  - Same as timing group for timing system
- Beam Process
  - Temporal section within a particle transfer with specific purpose
    (e.g. Injection, Ramp, Extraction, Transfer, ...)
  - Atomic *Context* in LSA
  - Distinction between beam processes with (BEAM_IN) and without beam (BEAM_OUT)
- Parameter
  - Physics or hardware quantity
  - Associated with a device
  - Name typically <DEVICE>/<QUANTITY>
- Setting
  - Value of a parameter in a beam process
  - Sum of *TARGET* und *CORRECTION*
  - Physics parameters don‘t have values in BEAM_OUT, while hardware parameters do
  - Internally all values in (adapted) SI units

Particle transfers in GSI_HEBT:
from forking point to forking point

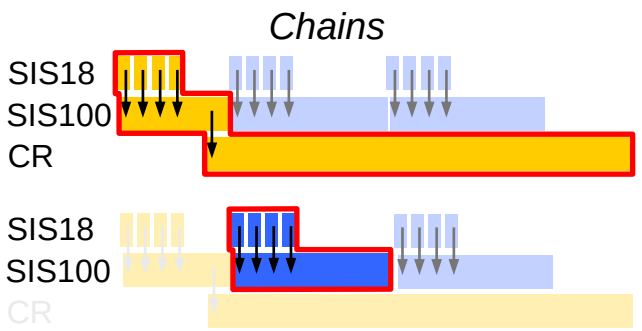


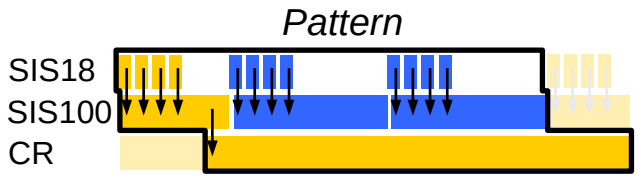
Setting of energy in SIS18	
Parameter	SIS18BEAM/E
Beam Process	P.C1.SIS18_RING.RING_INJECTION.1
TARGET	11400000
CORRECTION	0.0

eV/u !

LSA: Basic Structures (II)

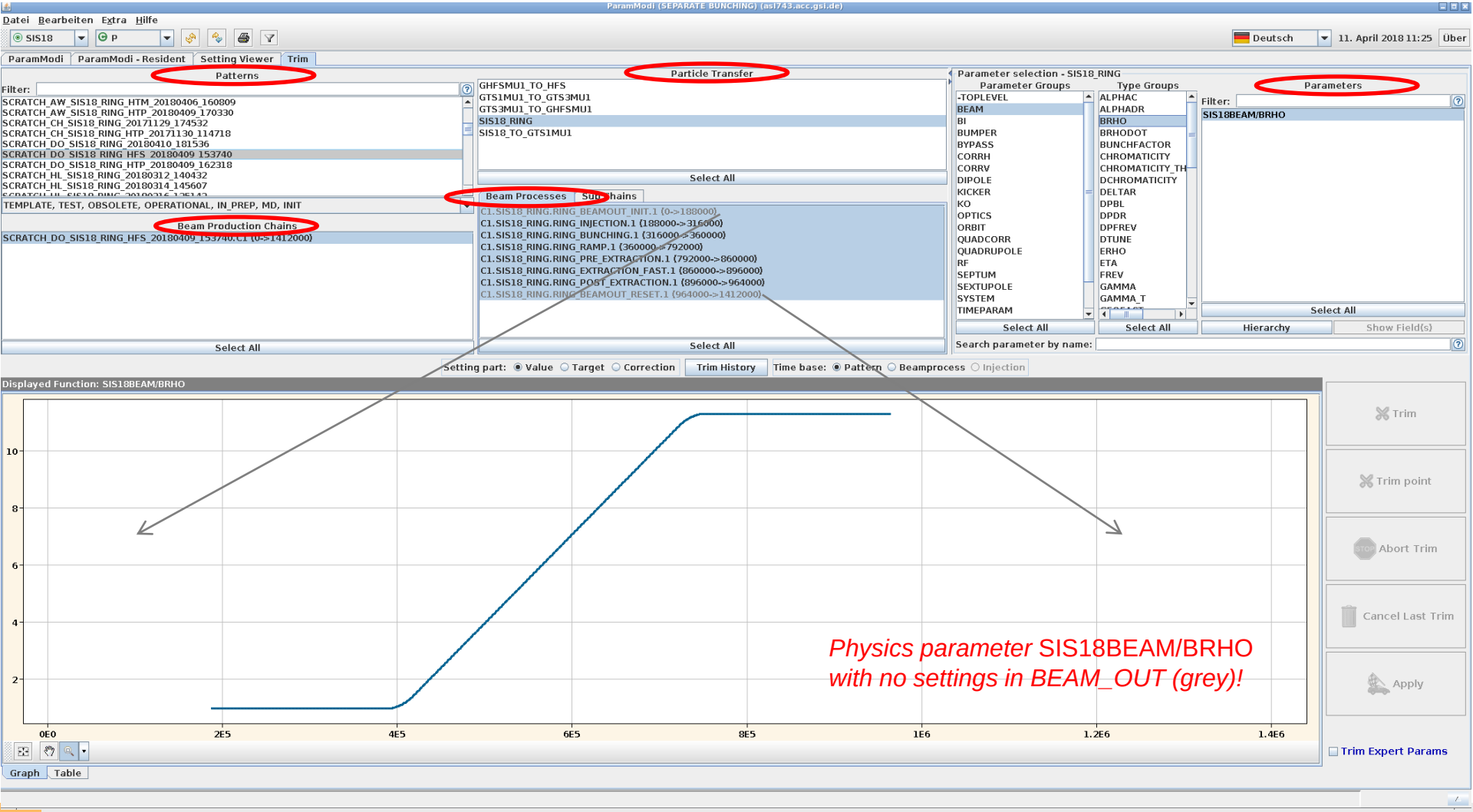
- Beam Production Chain
  - Set and temporal relation of all beam processes required for the production of a certain beam
  - Represents acceleration cycle including beam transport over transfer lines
  - Not executable on its own
  - *Context* for aggregation of beam processes
- Pattern
  - Grouping of chains
  - Intended to represent the periodic production of several beams in parallel
  - Executable unit for timing system
  - *Context* for aggregation of chains
- Present limitations
  - Only one chain per pattern
  - Such a pattern is equivalent to the former VirtAcc





LSA: Basic Structures at SIS18 (I)

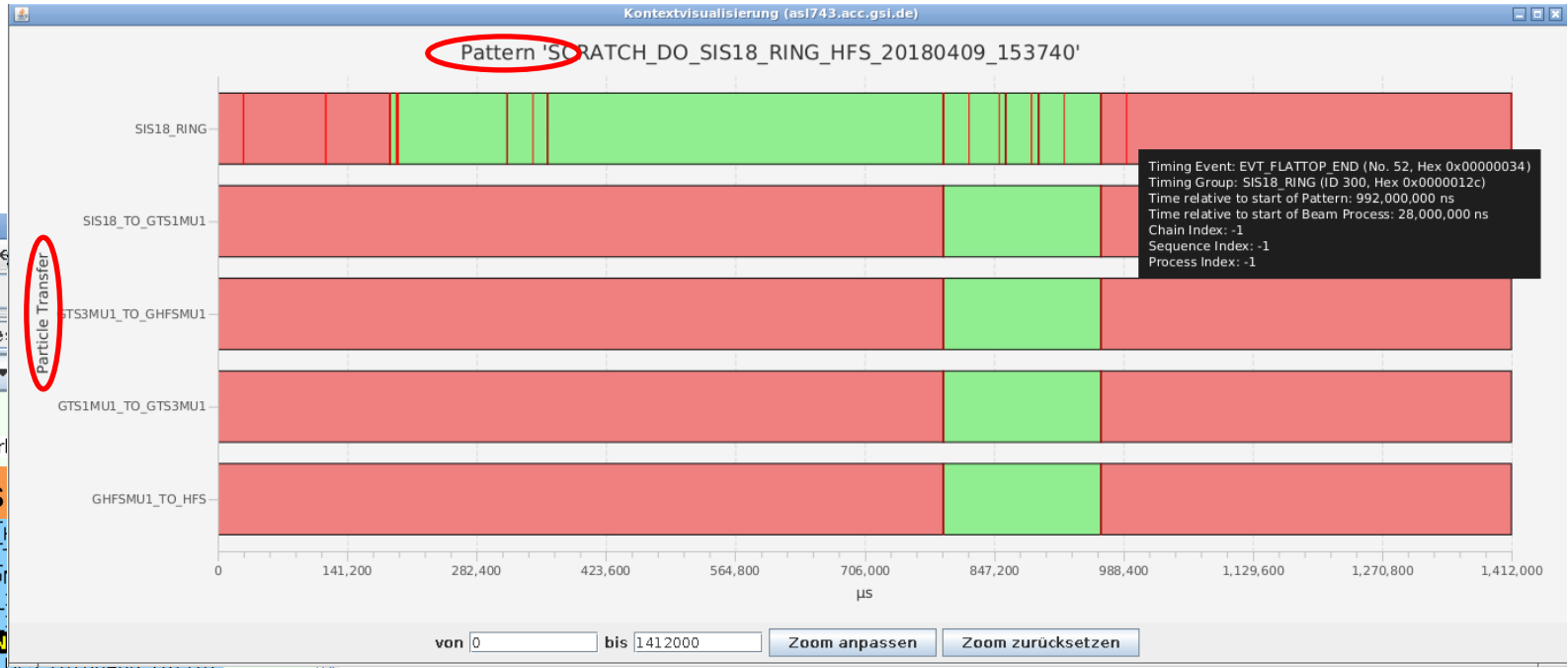
LSA basic structures as seen on ParamModi Trim view

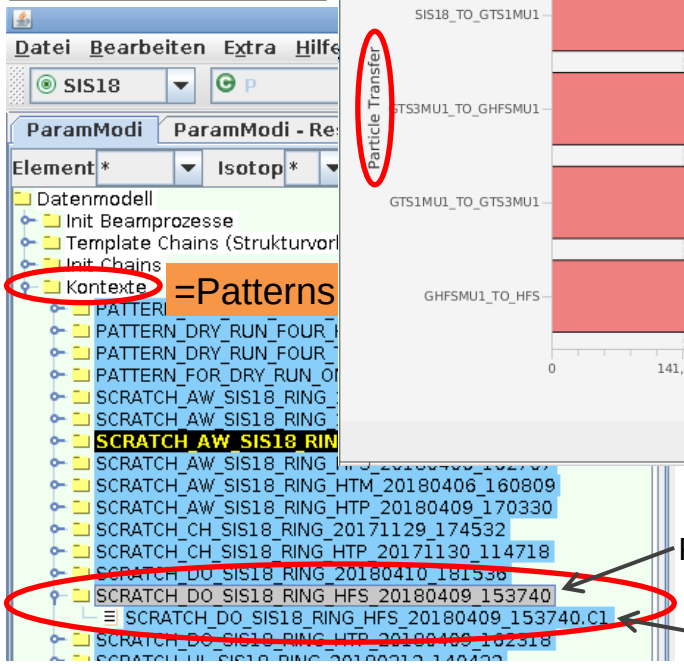


LSA: Basic Structures at SIS18 (II)

SIS18 pattern to HFS as seen on pattern display in ParamModi main tab

SIS18 patterns in ParamModi context selection panel





BP boundary

Timing event

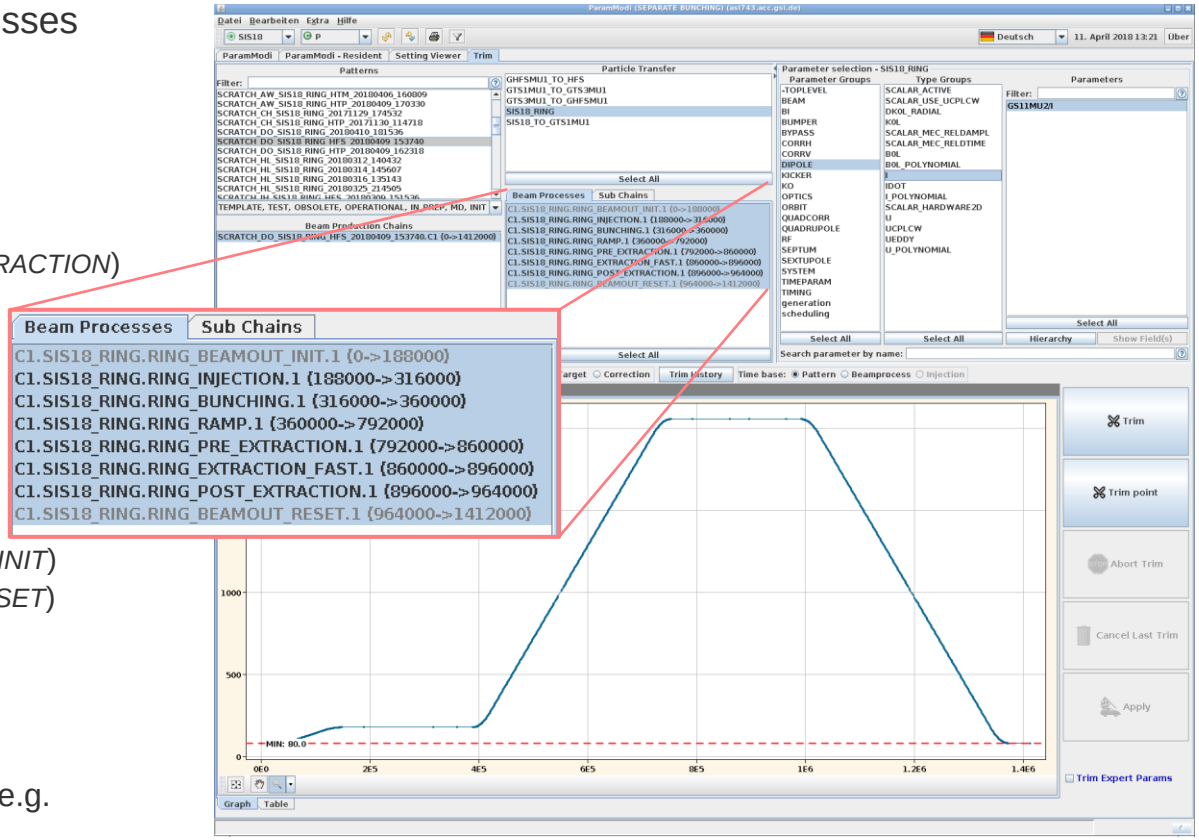
BEAM_IN

BEAM_OUT

LSA: Basic Structures at SIS18 (III)

- Division of SIS18 chain into beam processes according to *purpose* of the section
  - Beam processes with beam
    - Injection (*RING_INJECTION*)
    - RF capture (*RING_BUNCHING*)
    - Acceleration (*RING_RAMP*)
    - Preparation of extraction (*RING_PRE_EXTRACTION*)
    - Extraction (several purposes)
      - *RING_EXTRACTION_FAST*
      - *RING_EXTRACTION_SLOW*
      - *RING_EXTRACTION_KO*
    - Finalization of extraction (*RING_POST_EXTRACTION*)
  - Beam processes without beam
    - Preparation of injection (*RING_BEAMOUT_INIT*)
    - Return to inital state (*RING_BEAMOUT_RESET*)
  - Further purposes for special manipulations
- Standard chains for operation
  - Definition by modelers
  - Special sequences require special chains, e.g.
    - two ramps in one cycle
    - RF manipulations like bunch merging
  - Selection in SchedulingApp (see later)

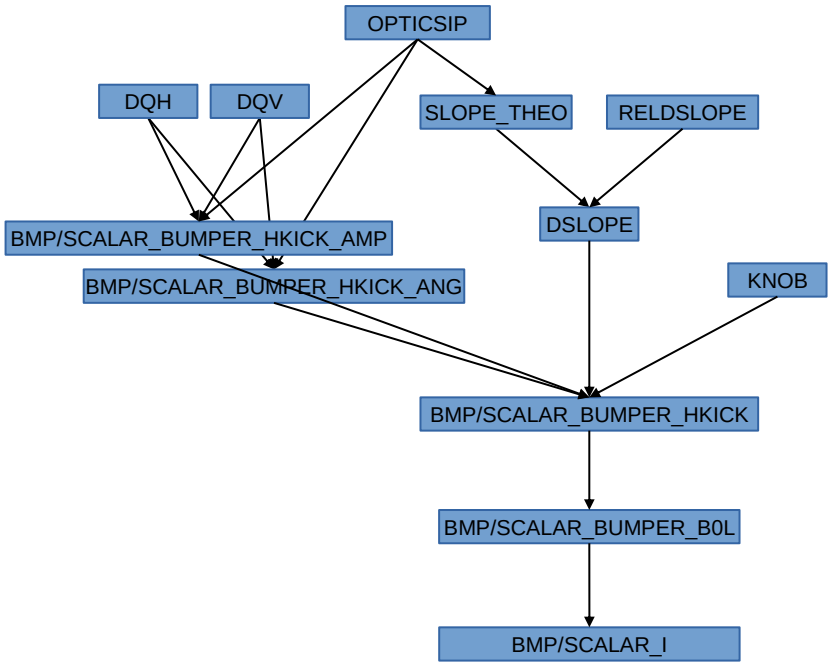
ParamModi/Trim: SIS18 pattern with fast extraction



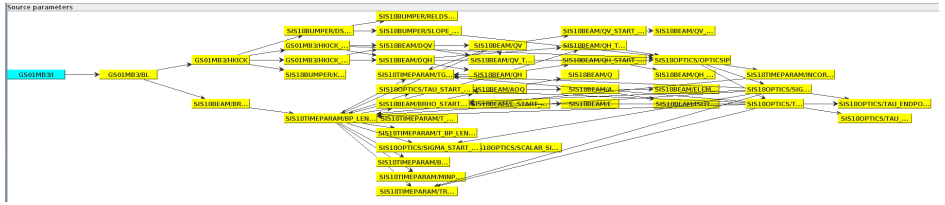
Machine Model: Parameter Hierarchy

- Explicit representation of the relation between parameters for the calculation of values
- Calculation from physics to hardware
  - Control of accelerator via physics parameters
  - Algorithms implemented as Java classes
  - Usage of data from DB (optics, calibrations, etc.)
  - Machine model uses *TARGET* only
    - ParamModi sets *TARGET* values
    - Calculations return *TARGET* values
  - Additive *CORRECTION* possible at each level
    - Supposed to be used for
      - inaccuracies of model
      - erroneous behavior of devices
    - No reverse calculation of explicit corrections
    - *CORRECTION* trims only accessible via Trim
- Hierarchy can be displayed in ParamModi
  - Hierachy button on Trim view
  - But beware: may catch huge number of params!

Example hierarchy: SIS18 injection bump (sketch)

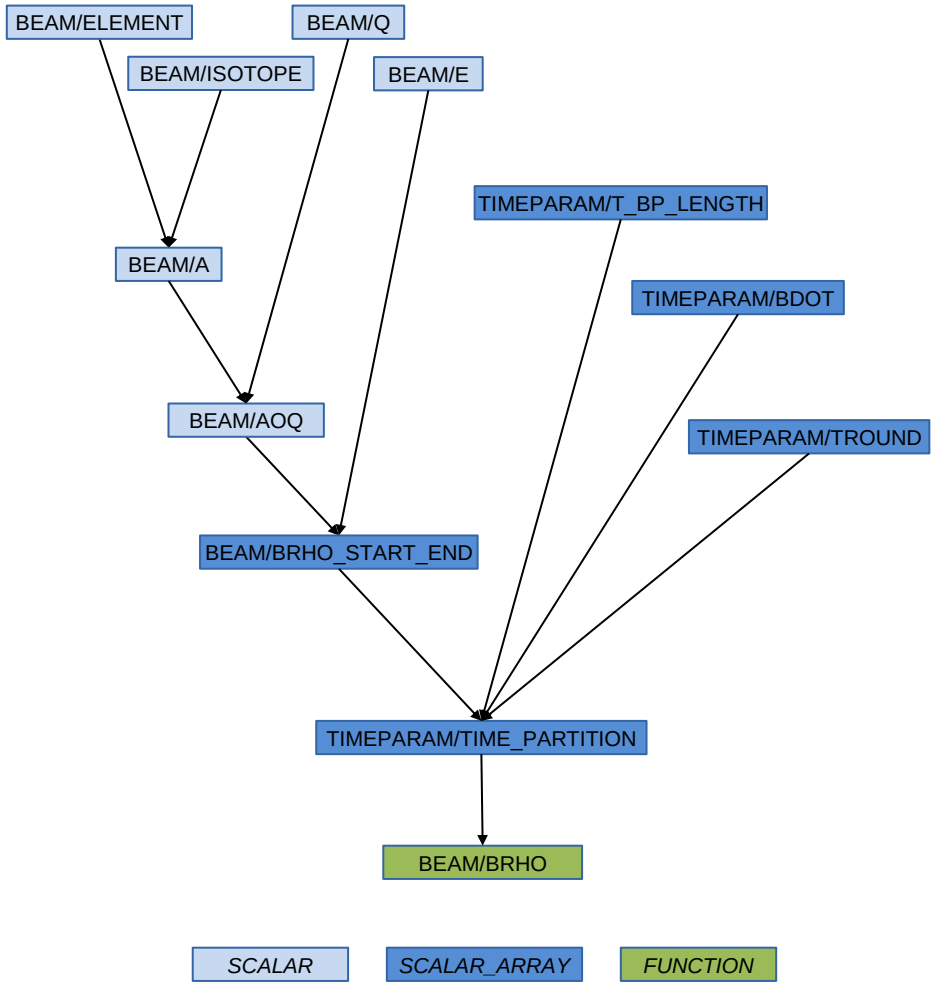


Full hierarchy of GS01MB3 as seen in hierarchy view



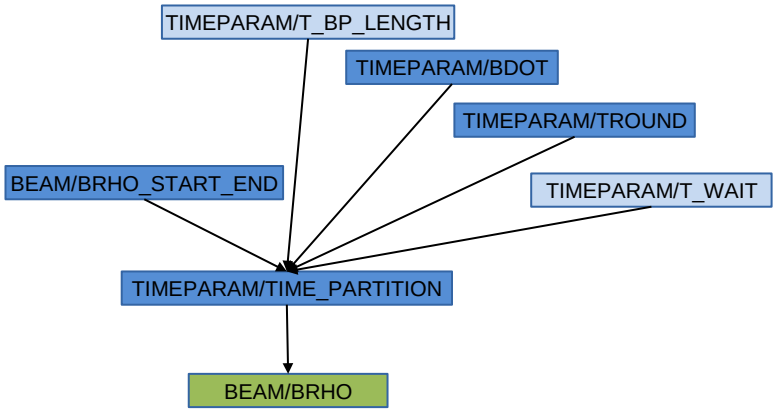
Machine Model: Top-Level Scalars

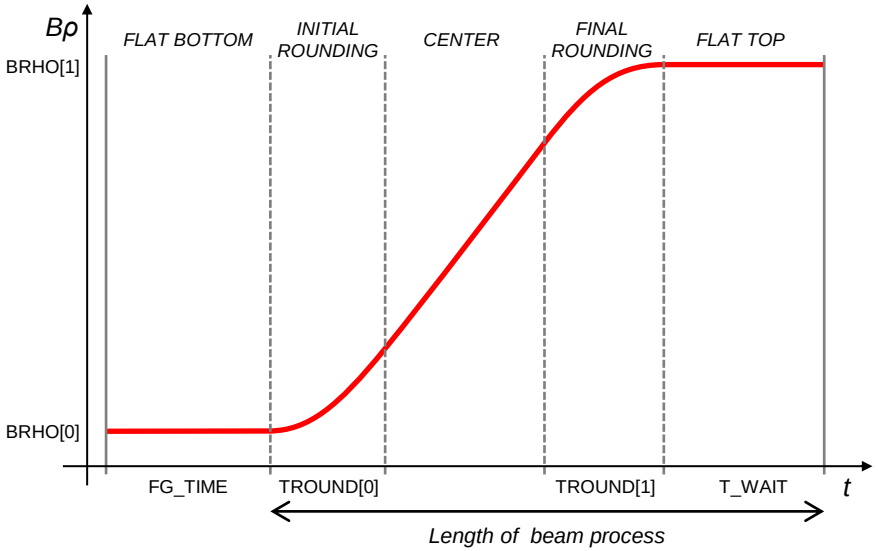
- Operation of rings requires time dependent set values
- Cycle skeleton needs to be created from scalar inputs
  - Beam parameters
    - Ion type, mass, charge
    - Injection and extraction energy
    - Particle number
  - Machine parameters
    - Tune and chromaticity
    - RF: Harmonic, bucket size
  - Time parameters
    - Bunching time, extraction time
    - Ramping speed
    - Rounding time



Machine Model: Transition to Functions (I)

- Each beam process partitioned into up to five segments
- Dipole field ramps
  - E.g. for acceleration/deceleration
  - Field ramp calculated based on
    - Ramping speed
    - Time parameters
  - Length of ramp is calculated accordingly
- Constant dipole field
  - Other parameters might change
    - Orbit bumps, tune, chroma
    - RF bucket size, RF manipulations

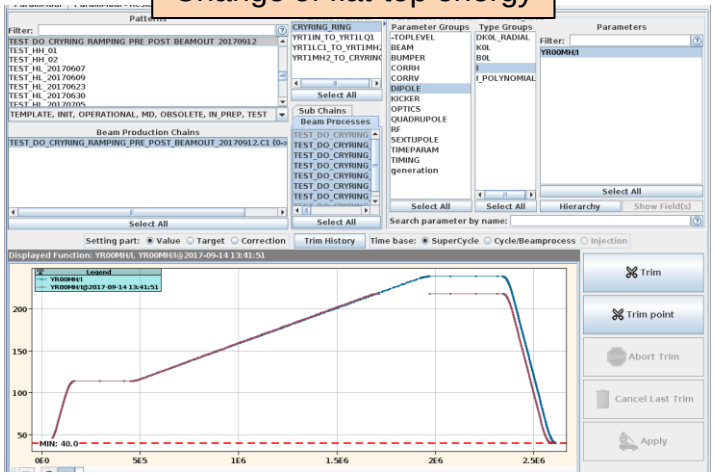




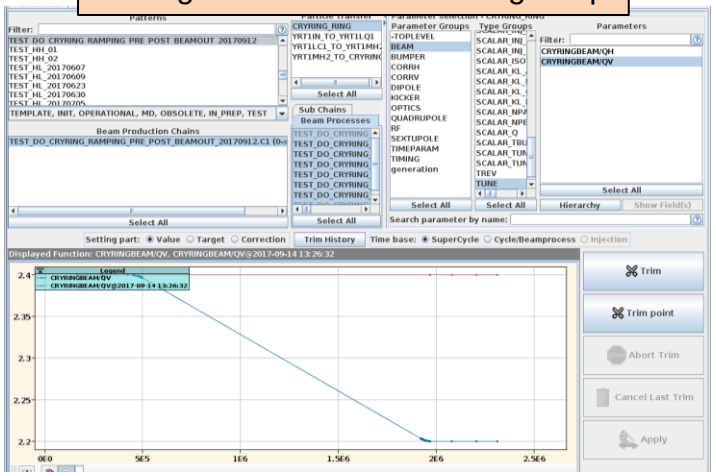
Machine Model: Transition to Functions (II)

- Top-level scalars typically define values in certain beam processes
  - Beam parameters (energy, tune, chroma, ...): injection and end of ramp
  - RF parameters (bucket fill factor, ...): injection, bunching, ramp, storage
  - Time parameters (ramping speed, ...): ramp, init, reset
- Transition to time functions through rules
  - Beam energy
    - Typically change of all time functions through make rules (link rules in beam-out)
  - Time parameters (e.g. ramping speed)
    - Typically change of all time functions through make rules, but only in affected BP
  - Optics and device parameters
    - Changes due to incorporation rules
    - Configuration per BP purpose
      - Changes e.g. during *RING_RAMP*, *RING_PRE_EXTRACTION*
      - Constant during *RING_INJECTION, RING_BUNCHING, RING_EXTRACTON_**

Change of flat-top energy



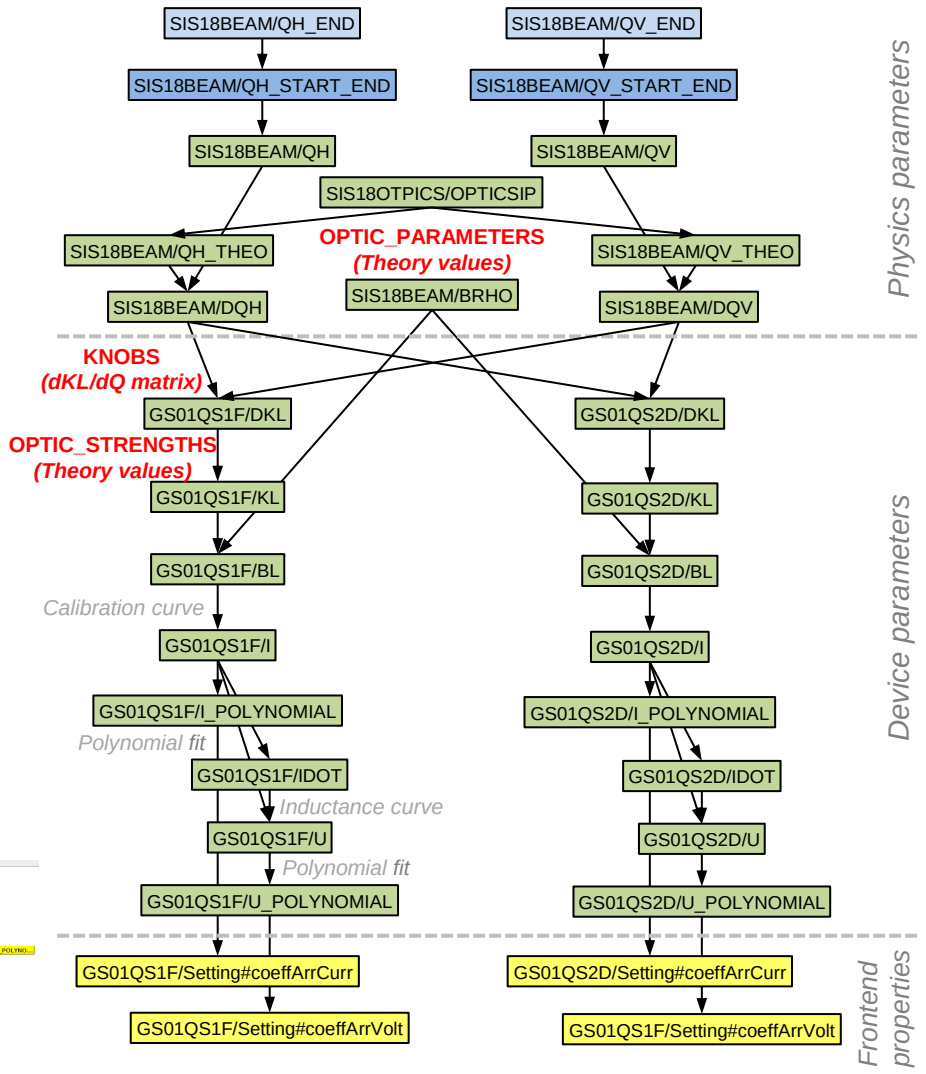
Change of vertical tune during ramp



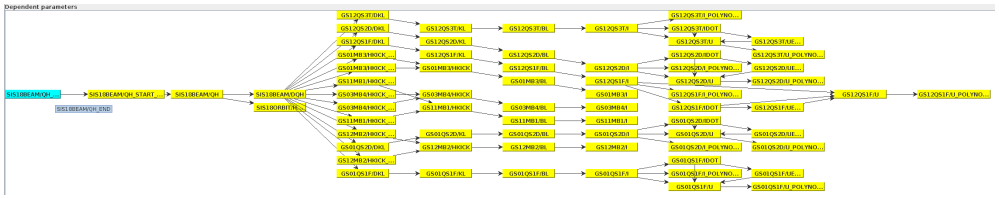
Machine Model: Tune Hierarchy

- Linear tune model
  - Theory strengths KL_0 and tunes $Q_{(h,v)0}$
  - Tune changes through matrix:
    $\Delta KL = dKL/dQ_h \cdot \Delta Q_h + dKL/dQ_v \cdot \Delta Q_v$
  - All data obtained from off-line simulations
  - One-way hierarchy: tunes → strengths
- Optics tables used for tune hierarchy
  - OPTIC_STRENGTHS: theory strengths
  - OPTIC_PARAMETERS: theory tunes
  - KNOBS: tune matrix coefficients
- Calculation in make rules
  - Optics data retrieved from LSA DB
  - Device data retrieved from LSA DB
    - Calibration curves, limits, precision

SIS18: Hierarchy of tunes and quadrupoles (sketch)



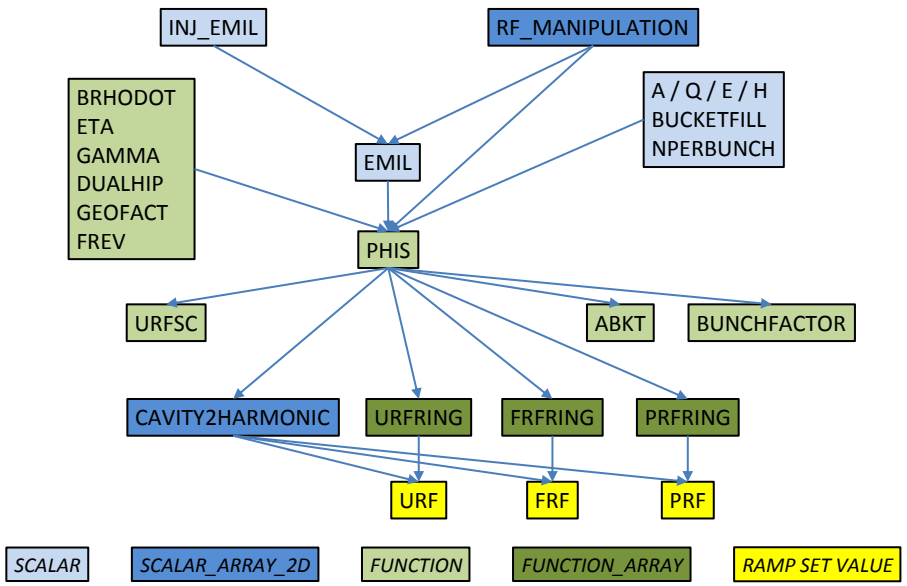
Hierarchy display: Hierarchy below SIS18BEAM/QH_END



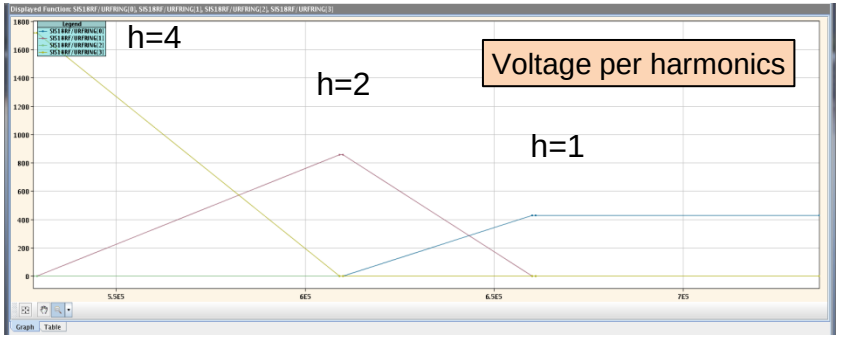
Machine Model: RF Hierarchy

- Calculations for RF among the most complex in the ring hierarchy
  - Multi-harmonic operation
  - RF gymnastics (merging, compression)
- Model driven by physics parameters
  - Input parameters
    - Beam: A, Q, E, emittance, #bunches, #particles
    - RF: h, bucket size, fill pattern, RF manipulations
    - Ramp: dB/dt, gamma, phase slip
  - Algorithms
    - Calculation from first principles
    - Lookup table for 2-h bucket calculations
    - Inclusion of space charge corrections
  - Output parameters
    - Ramp set values for cavities and master
    - Grouping of cavities according to harmonics
    - Additional physics quantities (SC voltage, bunching factor)

Generic parameter hierarchy for RF systems



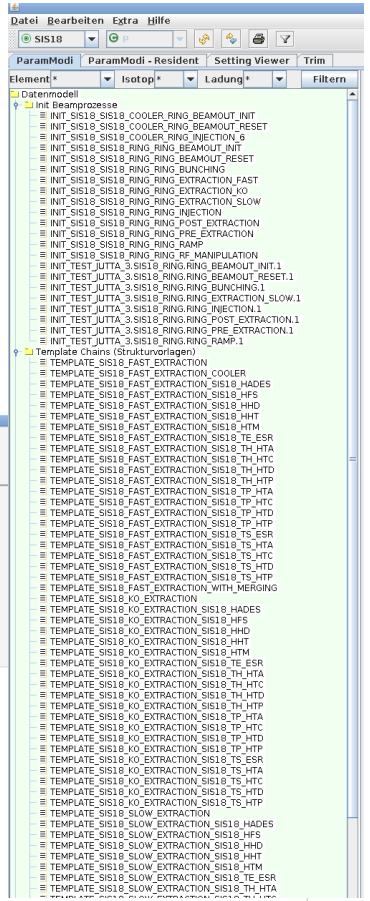
Bunch merging in SIS18: RF amplitude ramps



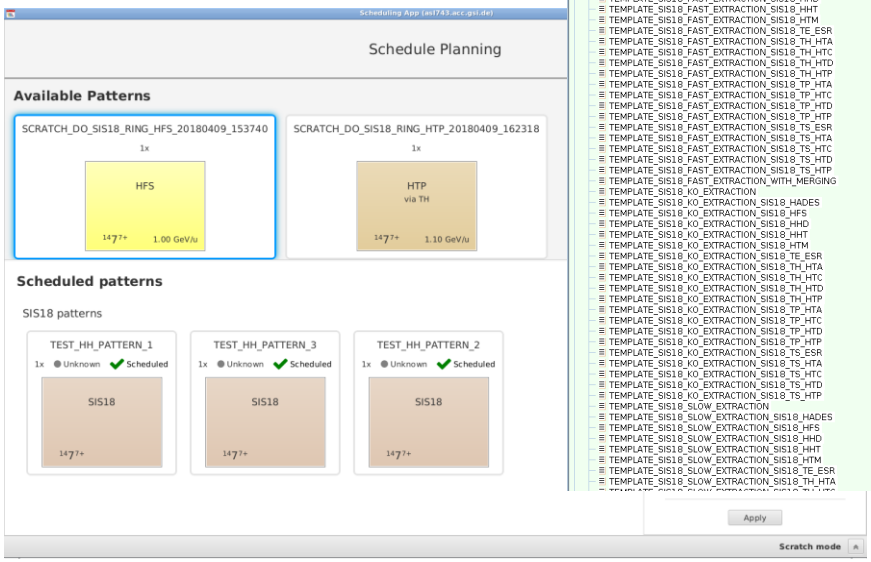
Creation of Patterns: Strategy

- Theory values stored in **Init Beam Processes**
  - Contain values for all top-level parameters which need to have a setting
  - Beam parameters are excluded, have to be filled in during pattern creation in **SchedulingApp**
  - Values in INIT BPs maintained by modelers
- Beam process structure for chain defined via **Template Chains**
  - Contains only beam processes and the definition of their temporal relations
  - Predefined by modelers for all operational cases
  - Depend on structural selection criteria
    - SIS18 extraction type (fast, slow, KO)
    - Electron cooler usage
    - Bunch merging for single bunch creation
    - Transfer line to target
- **SchedulingApp**
  - General planning tool for patterns
  - Used in particular for
    - creation of new patterns
    - scheduling of patterns for execution

Init BPs and Template Chains

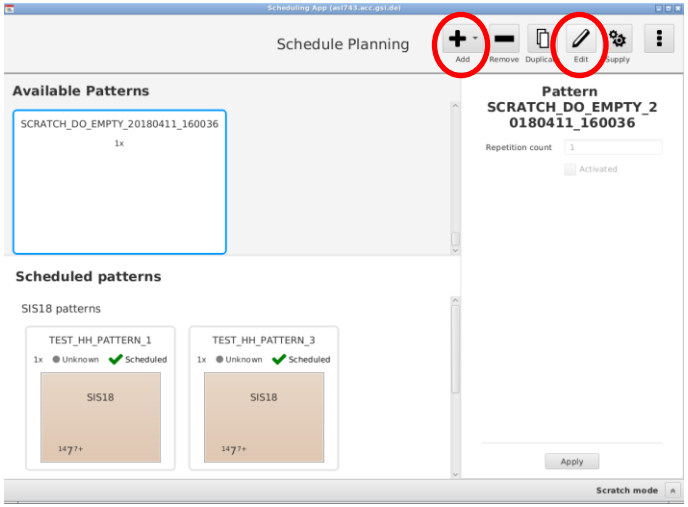


SchedulingApp

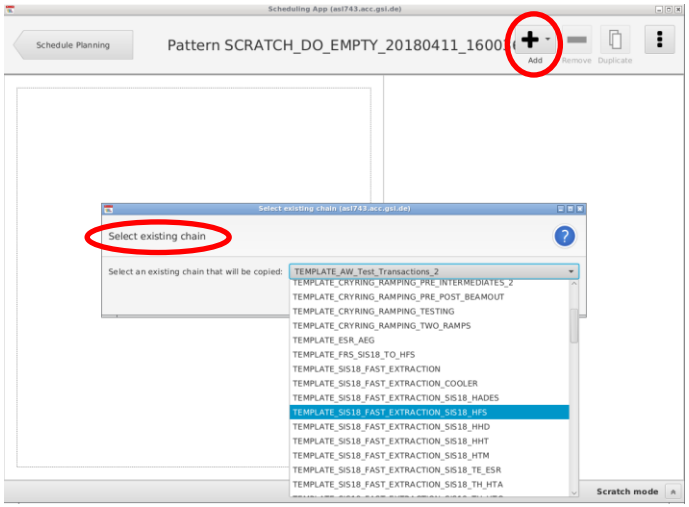


SchedulingApp: Pattern Creation (I)

- Procedure to create new pattern (as of today)
  - Create empty pattern
  - Go to pattern editor
  - Create chain
    - Select appropriate template chain
      - Drop-down menu
      - Based on naming convention
    - Enter missing values
      - Unilac Virtacc
      - Ion type, beam energy
      - Extraction time
      - Optics for transfer line
    - Change predefined values if desired
  - Add chain to pattern
- SchedulingApp is work in progress
  - Expect improved workflow in the future
    - Selection of template chain will be simplified
    - Selection of ion species from drop-down
    - Selection of Unilac Virtacc based on info in IPD
    - Omission of some intermediate steps

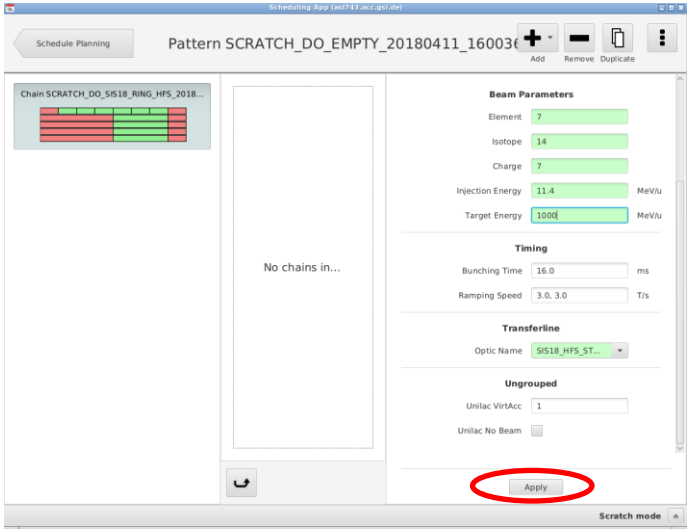


1. Planning view: Creation of new pattern

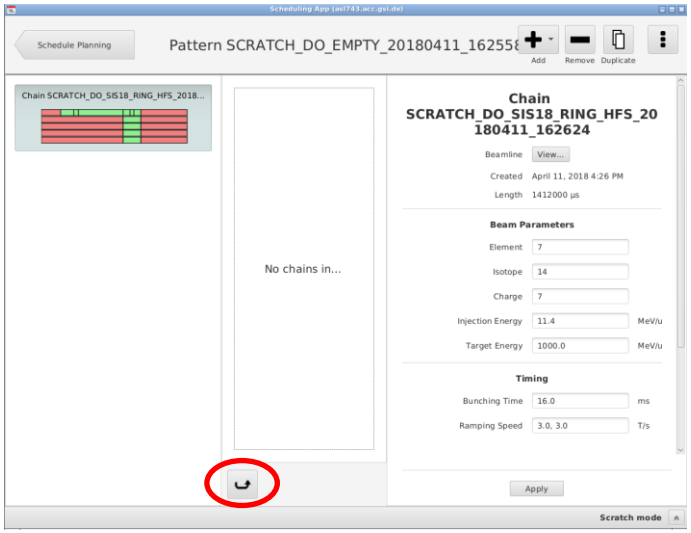


2. Edit view: Selection of template chain

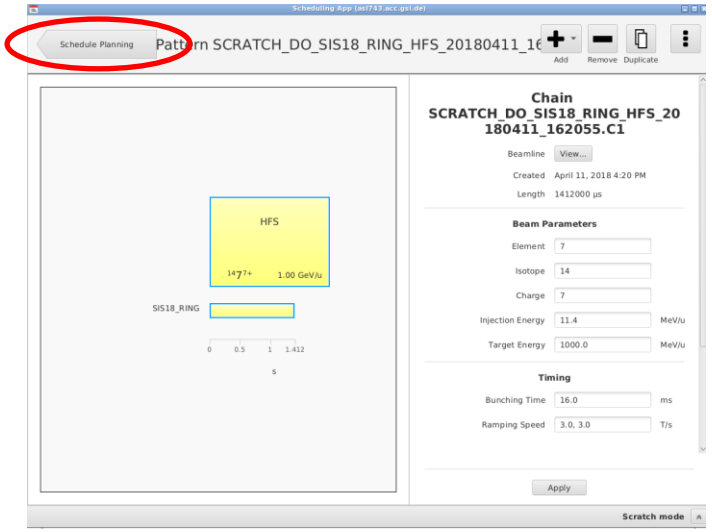
SchedulingApp: Pattern Creation (I)



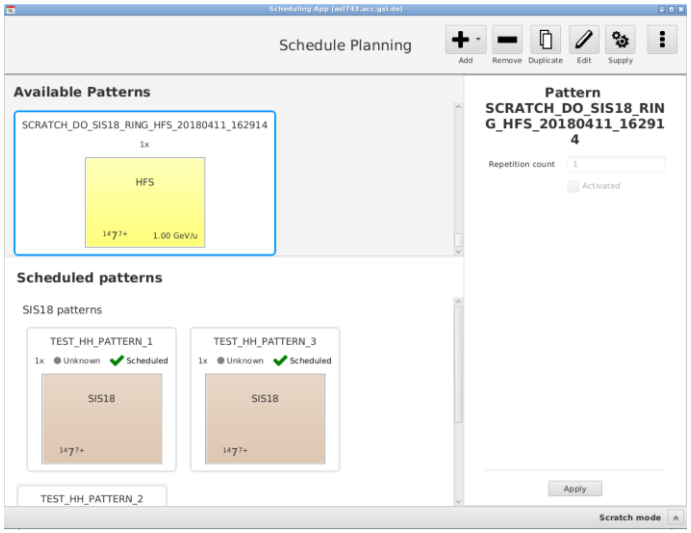
3. Edit view: Definition of input value



4. Edit view: Chain added to pattern



5. Edit view: Return to planning view

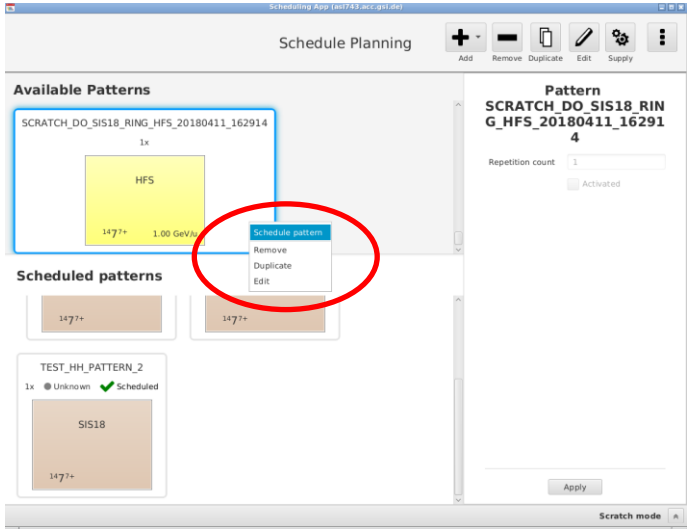


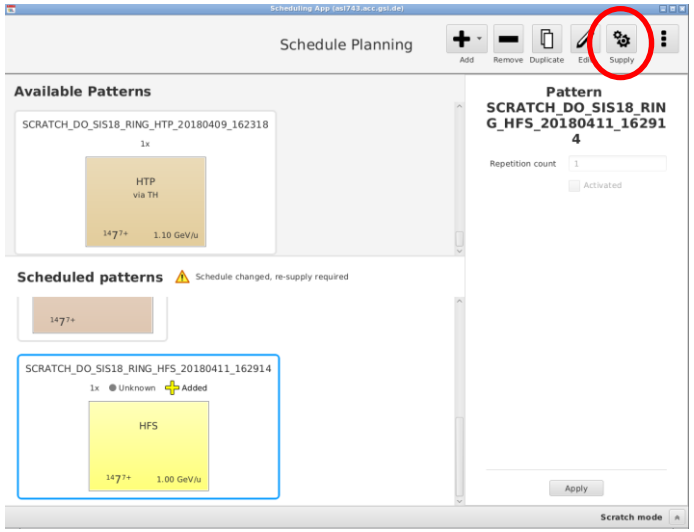
6. Planning view: Pattern ready for scheduling

CAVEAT:
Settings only in LSA DB at this stage, not yet sent to devices!

SchedulingApp: Modifying Super-Cycle

- Procedure to add pattern for execution
  - Select pattern
  - Context menu “Schedule Pattern”
  - Button “Supply” to finish
- Similar for removing patterns
- What happens during supply?
  - Timing system stopped
  - All set values sent from DB to devices (only when adding pattern)
  - New timing program sent to timing system
  - Timing system activated again
- Limitations of scope of SchedulingApp
  - Only definition of desired status of accelerator execution
  - No real-time information on actual execution
  - Intentional decision
  - Other tools needed for displaying actual status

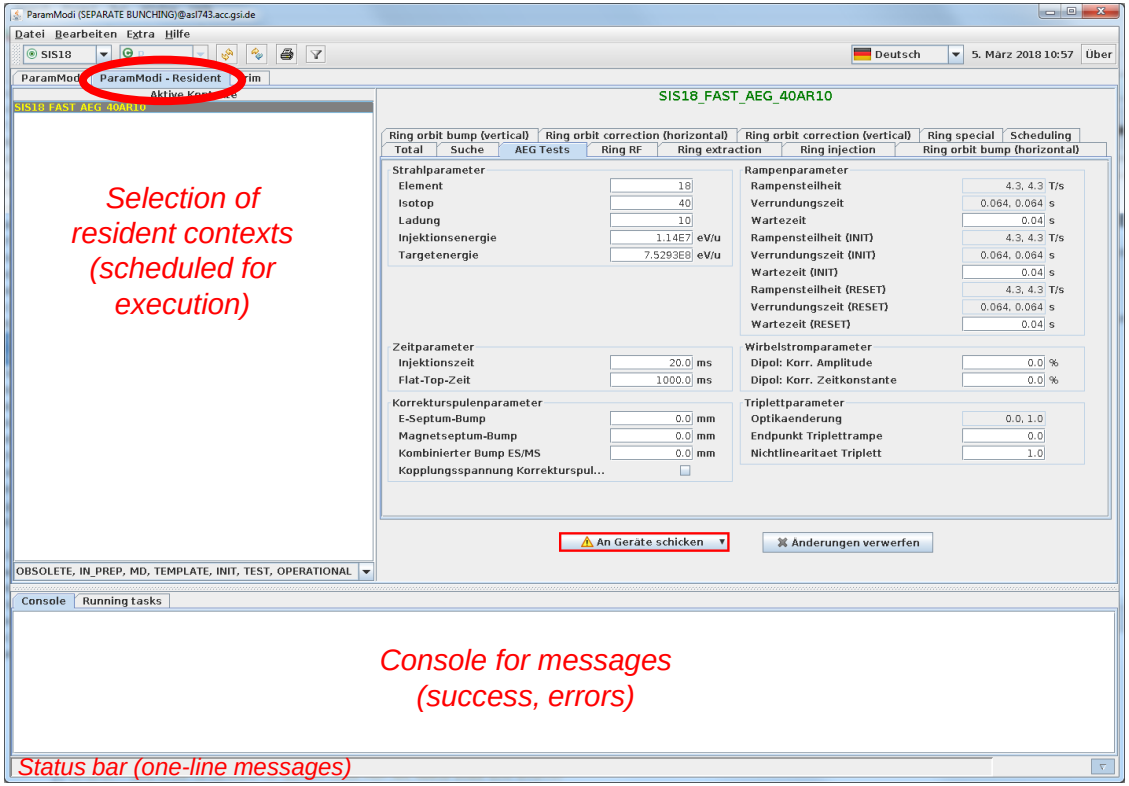




ParamModi: Operator Interface to Model

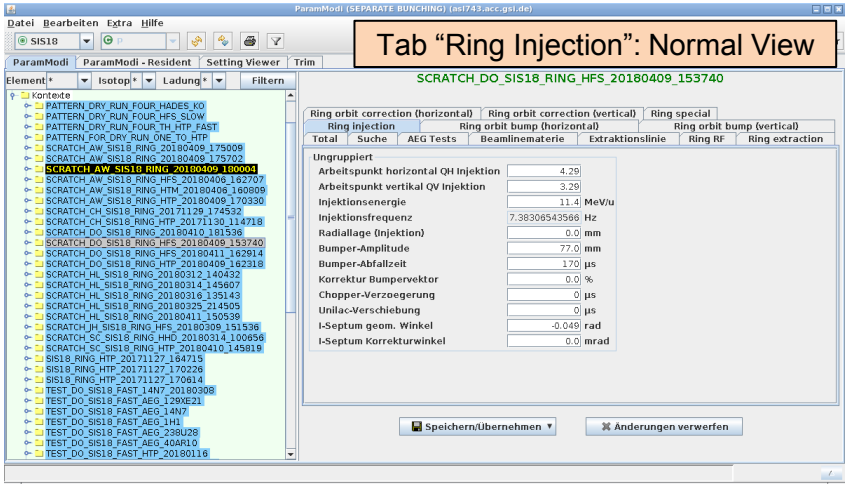
- Abstraction layer above parameters of the model
  - Input fields for top-level scalars
  - Combination of parameter and BP
  - Identification via display name
    - E.g. “Injektionsenergie” <-> SIS18BEAM/E in injection BP
- Grouping of input fields
  - Tabs for different functionalities (e.g. injection, extraction, RF)
  - Sub-grouping on tabs possible
  - Configurations defined by modeler
  - Generic standard configurations
- Views for offline and resident patterns
  - Resident view for operation
  - Offline view mainly for modelers
- Configuration is work in progress
  - Tabs for standard operation will be defined until DR8

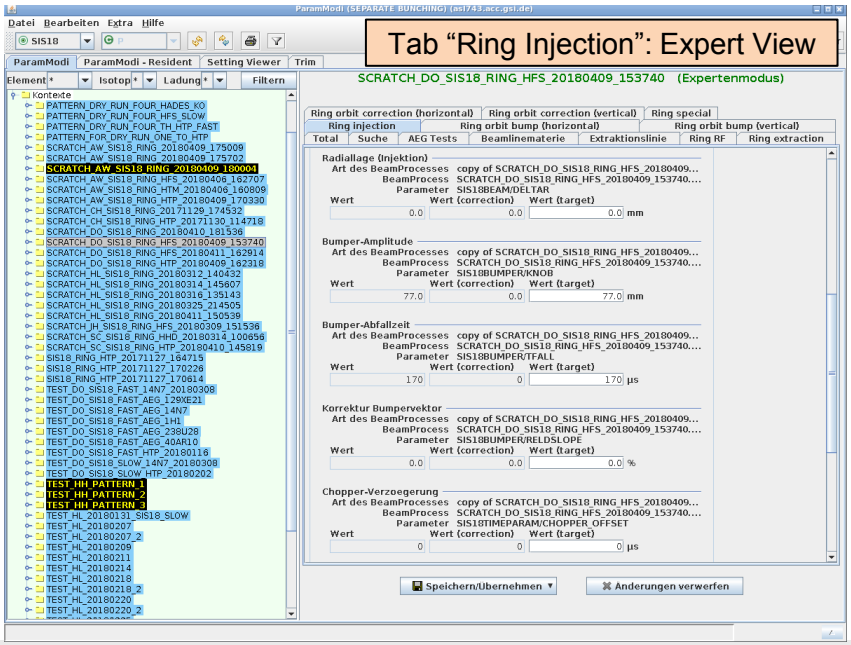
ParamModi Resident View: Parameters for SIS18 AEG-Tests



ParamModi: Input Fields and Parameters

- Disadvantage of abstraction
  - Relation to parameter and BP not obvious
  - Can only be seen on expert view
- Expert view
  - Available via Menu “Extra/Expertenmodus”
  - Displays additional information on input field
    - Display name (same as on normal view)
    - Beam process
    - Parameter
    - TARGET and CORRECTION value (CORRECTION should always be zero, though)
- Searching for input fields/parameters
  - Search button
  - Based on display name and parameter name
- Work in progress
  - Additional functionality for easier access to underlying parameters planned

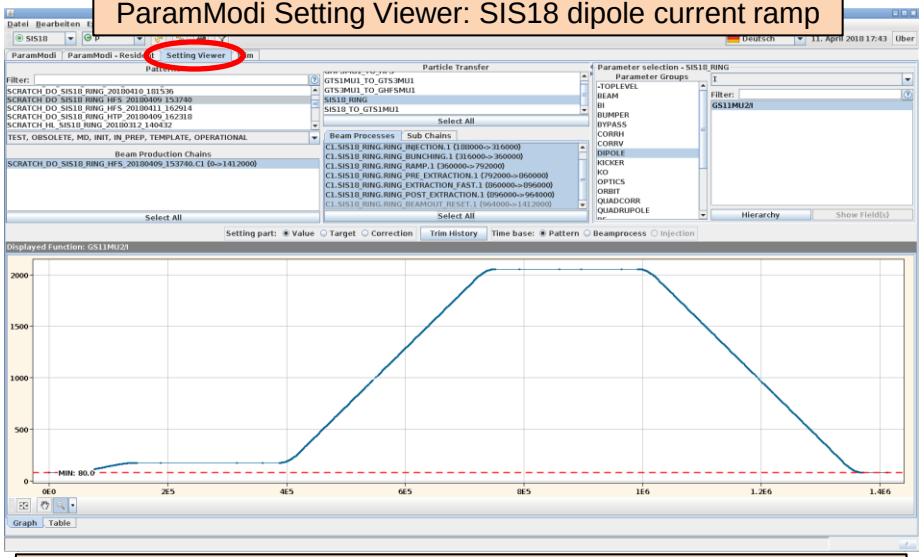




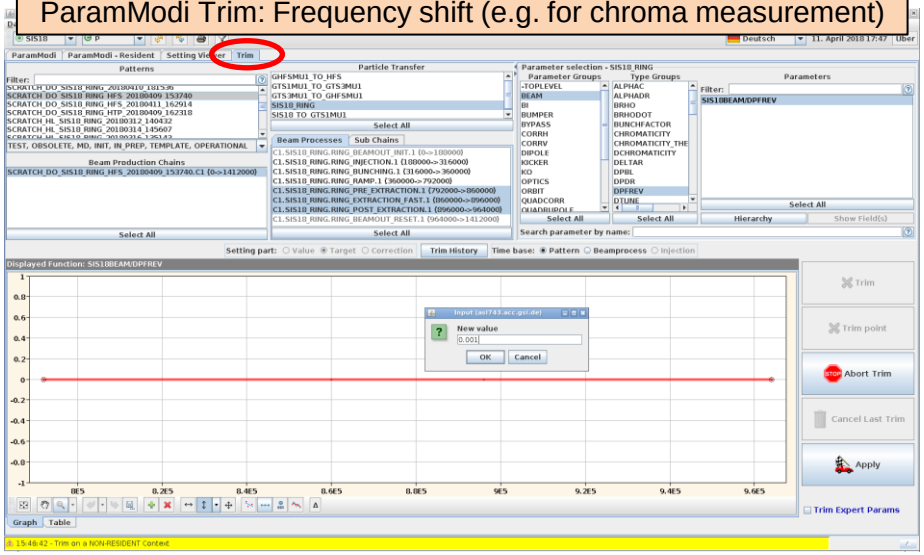
ParamModi: Setting Viewer and Trim

- Direct access to settings per parameter
  - Parameters grouped according to
    - Parameter Groups:
      Categories of parameters for devices with similar functionality (e.g. dipoles, quadrupoles, RF, etc.)
    - Parameter Type:
      Specifying the kind of quantity the parameter represents (e.g. strength, current, frequency, etc.)
  - Selection of subsets of beam processes possible
- Setting Viewer
  - Mainly used for inspecting function settings
  - No modifications possible
  - Includes buttons to activate
    - hierarchy view
    - trim history view
- Trim
  - Modifications of parameters (even functions)
  - Expert tool, not intended for standard operation
  - Changes not visible on normal view!
  - Use with care...

ParamModi Setting Viewer: SIS18 dipole current ramp



ParamModi Trim: Frequency shift (e.g. for chroma measurement)

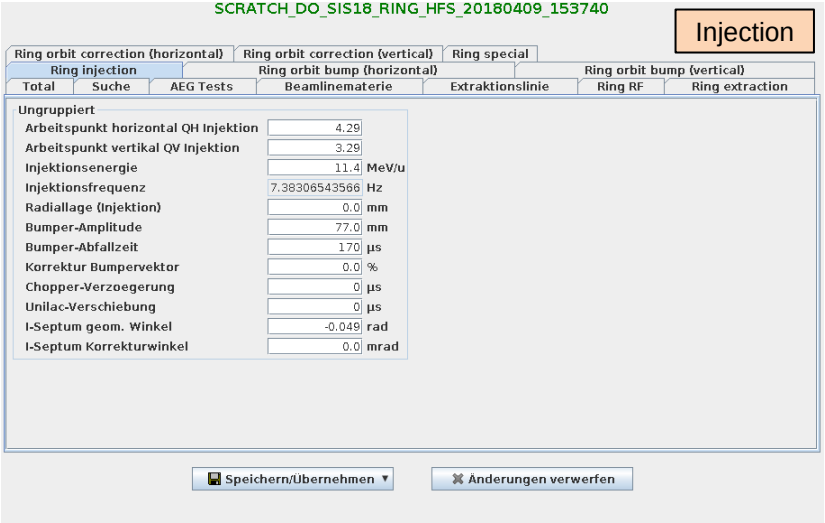


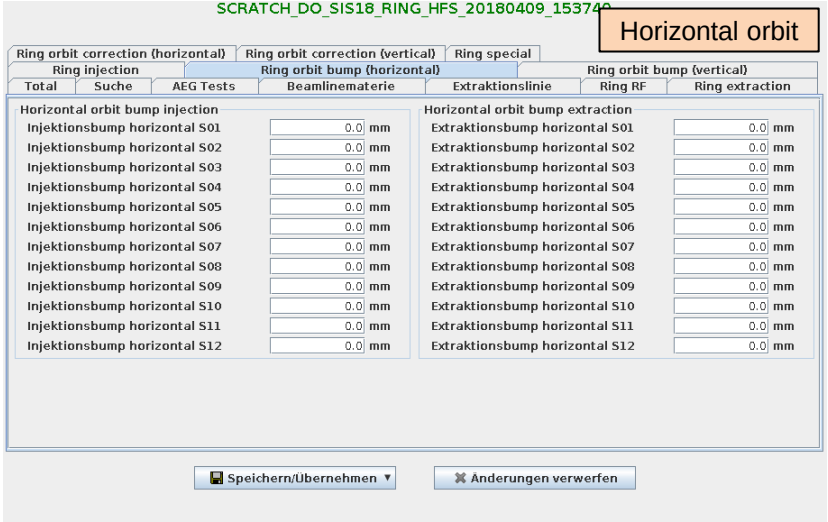
ParamModi: Parameters for SIS18

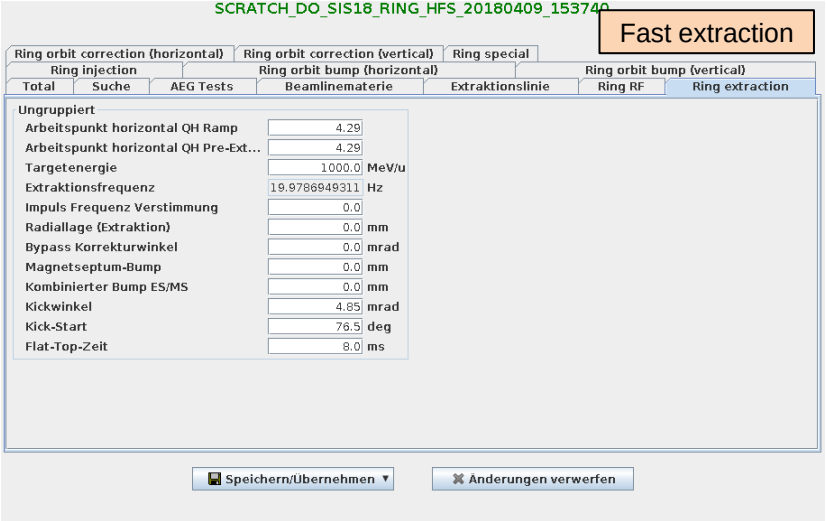
- Beam parameters
  - Energie
    - Injection, end of ramp
    - Independent of Unilac energy setting
  - Tunes (Q_h, Q_v) and chromaticity (C_h, C_v)
    - Injection, end of ramp, end of pre-extraction
  - Radial displacement (const. energy)
    - Injection, end of ramp
  - Momentum spread at injection (coasting beam)
    - Defines long. emittance, which is used as unit of RF bucket size
- RF parameters
  - Relative bucket size (in units of long. emittance)
    - Specification of boundary values in BPs
    - Ramping not in all BPs possible (e.g. injection)
  - *Note: RF voltage not an input parameter*
    - *Calculated by model from*
      - *long. emittance and rel. bucket size*
      - *ramp rate (during acceleration)*
- Timing parameters
  - Ramp rate
    - Only in BPs with **changing field**
    - Specification of rate dB/dt at start and end of ramp
  - Multi-turn injection
    - Injection pulse length and position
    - Bumper ramp-down time
  - Bunching time
  - Extraction time for slow extraction
- Slow extraction parameters
  - Sextupole amplitude and phase
  - Tune variation or KO parameters
  - Extraction bumps
- Fast extraction parameters
  - Kicker angle and phase
  - Extraction bumps
- Orbit correction
  - Local bumps at injection and flat-top

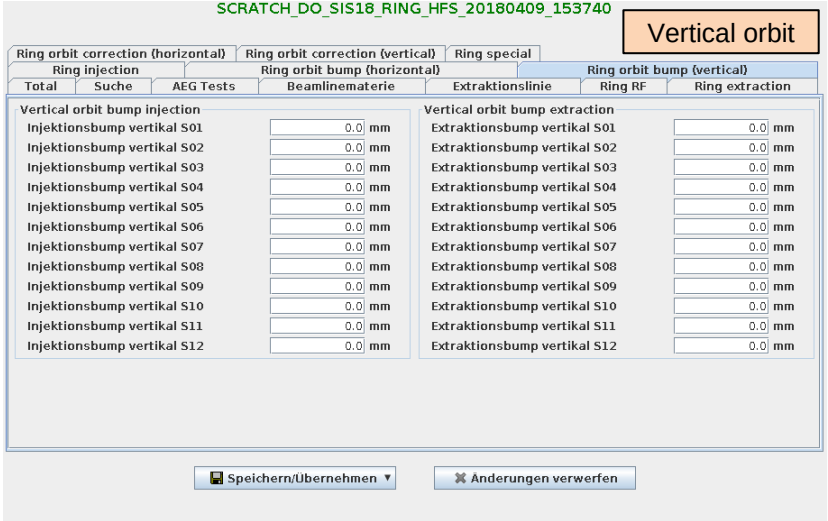
To be found on appropriate tabs in ParamModi starting from DR8...

ParamModi: SIS18 Tabs (DRAFT)



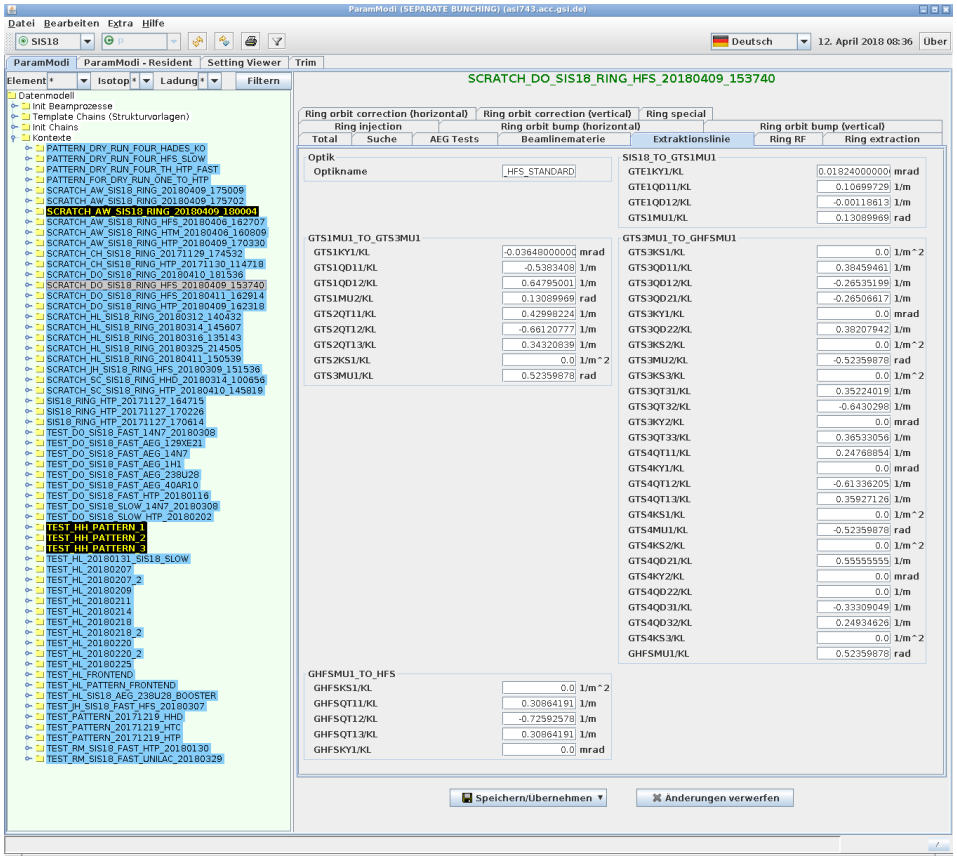






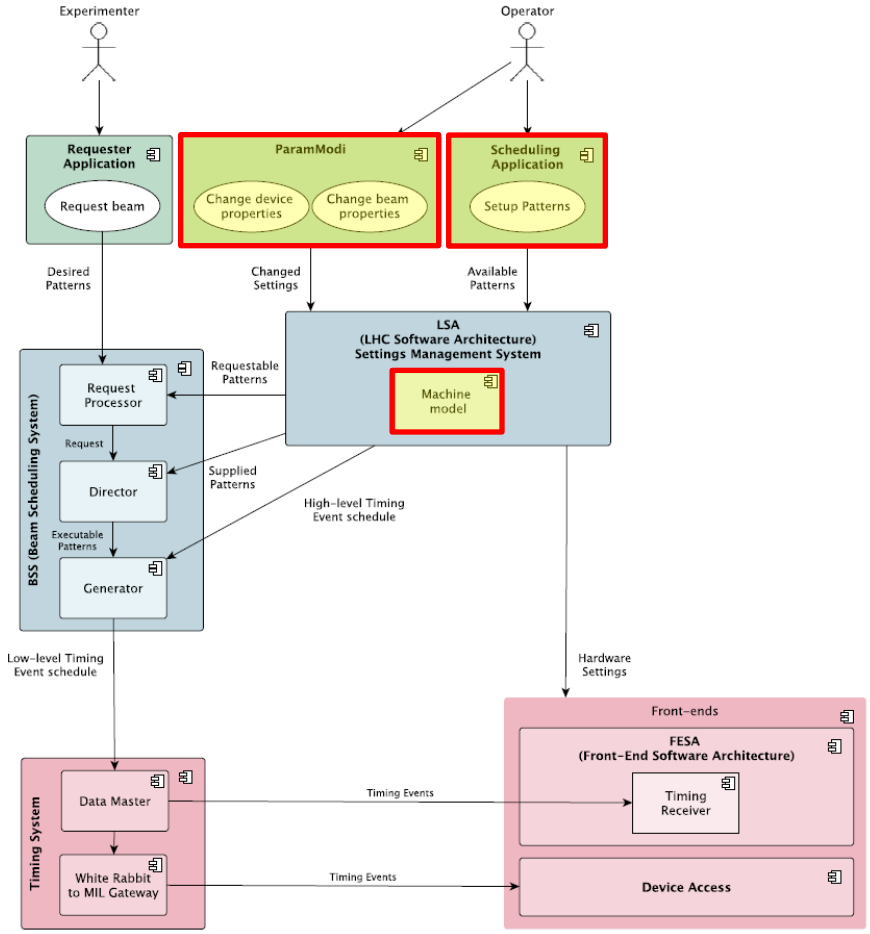
ParamModi: Transfer Lines

- Separate Tab for transfer line parameters
  - Sorted according to sequence in transfer line
  - Grouped into particle transfers
- Control of magnets via strength parameters
  - Dipoles and steerers: angle
  - Quadrupoles: focusing strength
  - Sextupoles: normalized amplitude
- Initialization from theory optic in LSA DB
  - Optic can be changed
  - Will overwrite all values with theory!
- ParamModi is work in progress
  - Geometric angle of dipoles hidden soon
    - Steering parameter instead
  - No way of creating/setting reference yet
    - Resort to export/import feature
  - More convenient input method planned
    - Arrows with selectable increment (as replacement for “poti board”)



Data Supply: The Overall Picture

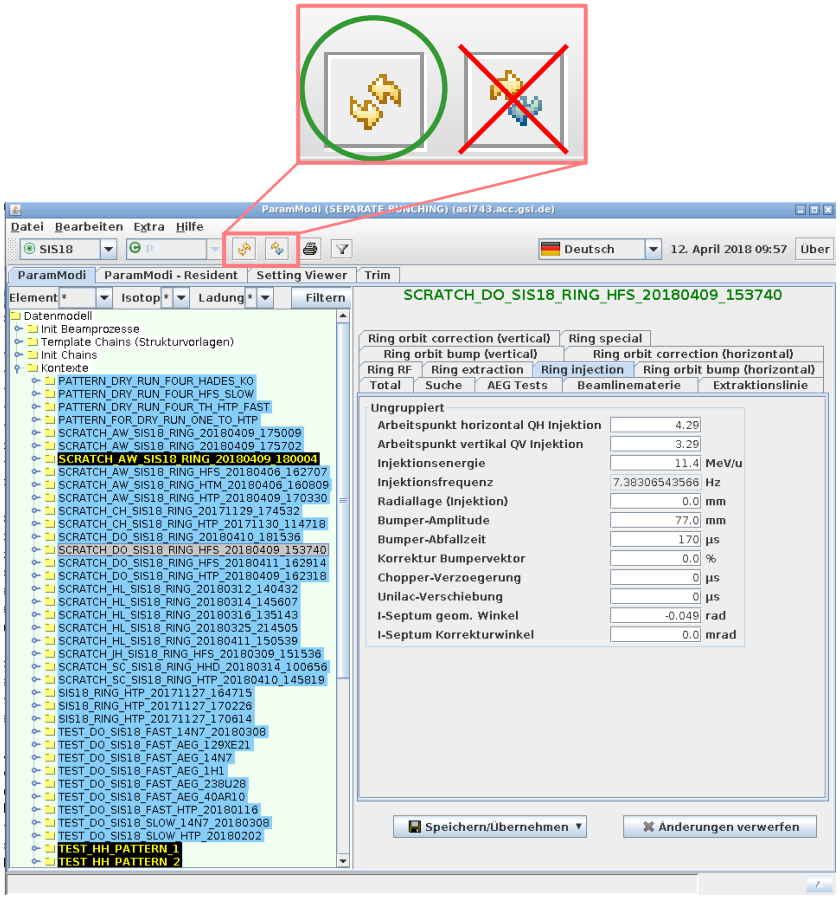
- Data supply embedded in control system
- LSA plays central role
  - Machine model part of LSA server
    - All calculations done on server
    - All data stored in LSA DB
  - LSA server provides data
    - to applications
    - to devices (including timing system)
  - No machine model logic in applications!
- Applications are operator's interface to model
  - SchedulingApp
    - Creation of initial patterns
    - Definition of patterns for execution
  - ParamModi
    - Modification of settings
    - Display of settings
  - Others applications in the future
    - Parameter scan application
    - Beam based feedback



H. Huether, Controls Architecture, presented in this course

Data Supply: Traps

- Separation of application and server
  - PRO: Several applications can modify same data
  - CON: Several applications can modify same data
- No messaging to inform about changes yet
  - Manual refreshing in ParamModi necessary if
    - new pattern has been created (otherwise new pattern not visible)
    - patterns have been added or removed (otherwise resident view is not correct)
    - settings have been changed by another instance of ParamModi (or other program)
  - Messaging to be implemented soon
- Errors during calculation
  - Reported as TrimException
  - Not yet easily interpretable by operators
  - On-call duty for data supply available



Summary

- Data supply: scope and task defined
- Machine model: task defined
- LSA framework
  - Overview over architecture
  - Basic structures and examples
- Machine model
  - Parameter hierarchy with examples
  - Creation and modification of functions
- SchedulingApp
  - Creation of new patterns
  - Adding and removing patterns for execution
- ParamModi
  - Different views: Offline, Resident, SettingViewer, Trim
  - Parameters for control of SIS18 and transfer lines
- Overall picture