FAIR Sequencer

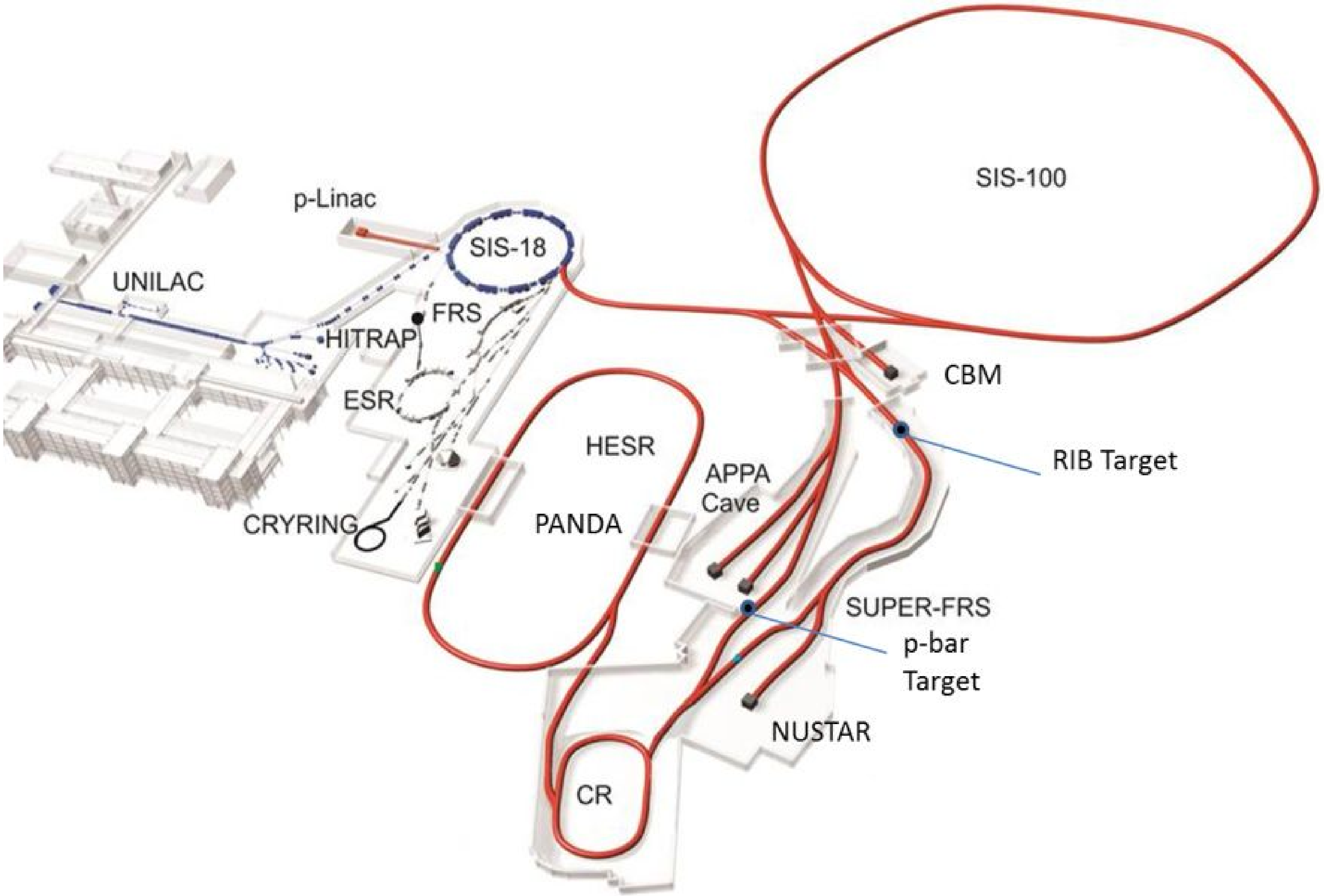
– computerized system validation –

Preliminary Concepts and first Prototype

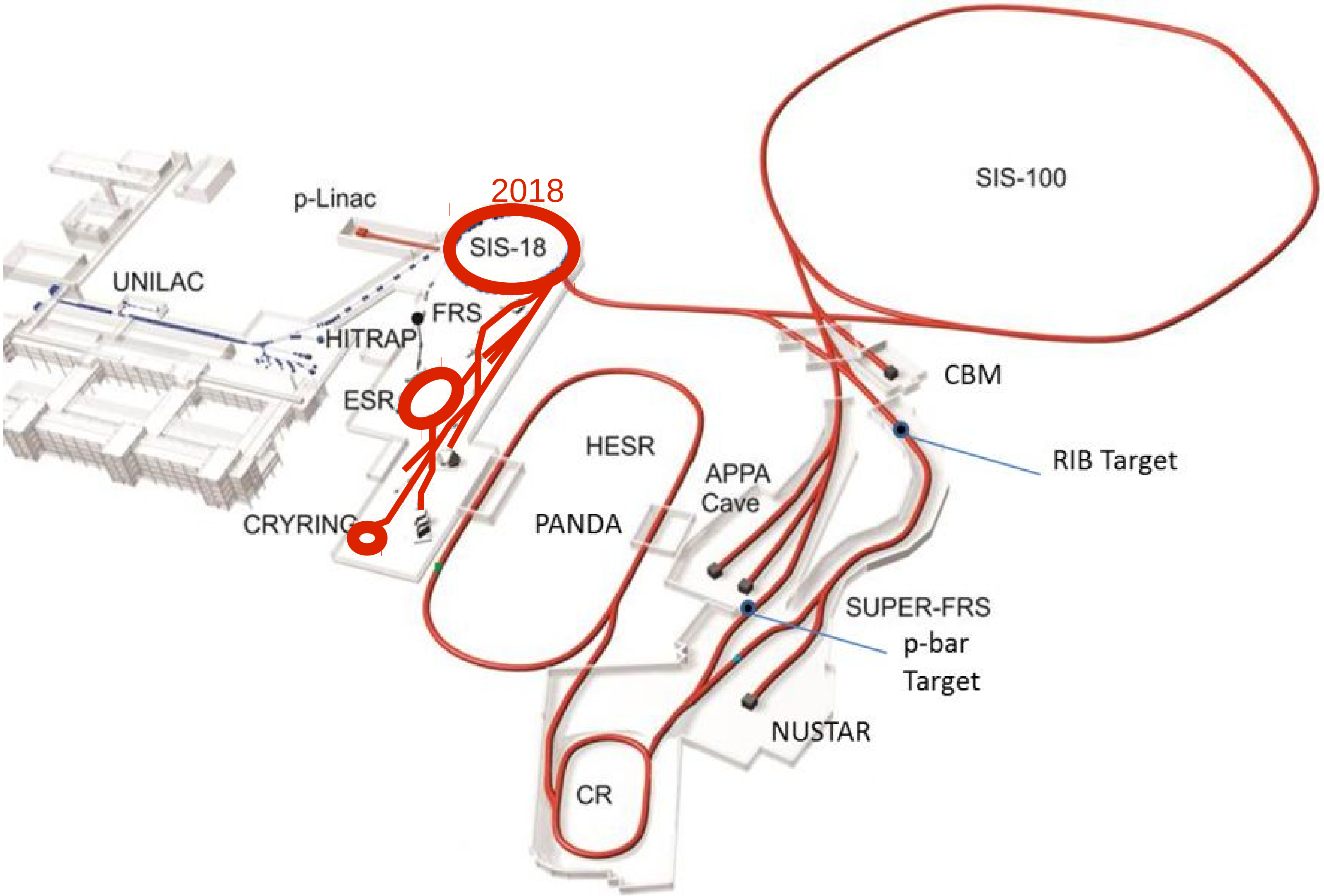
Ralph J. Steinhagen, R. Mueller

* based on 2015/16 FC2WG presentations & meeting minutes

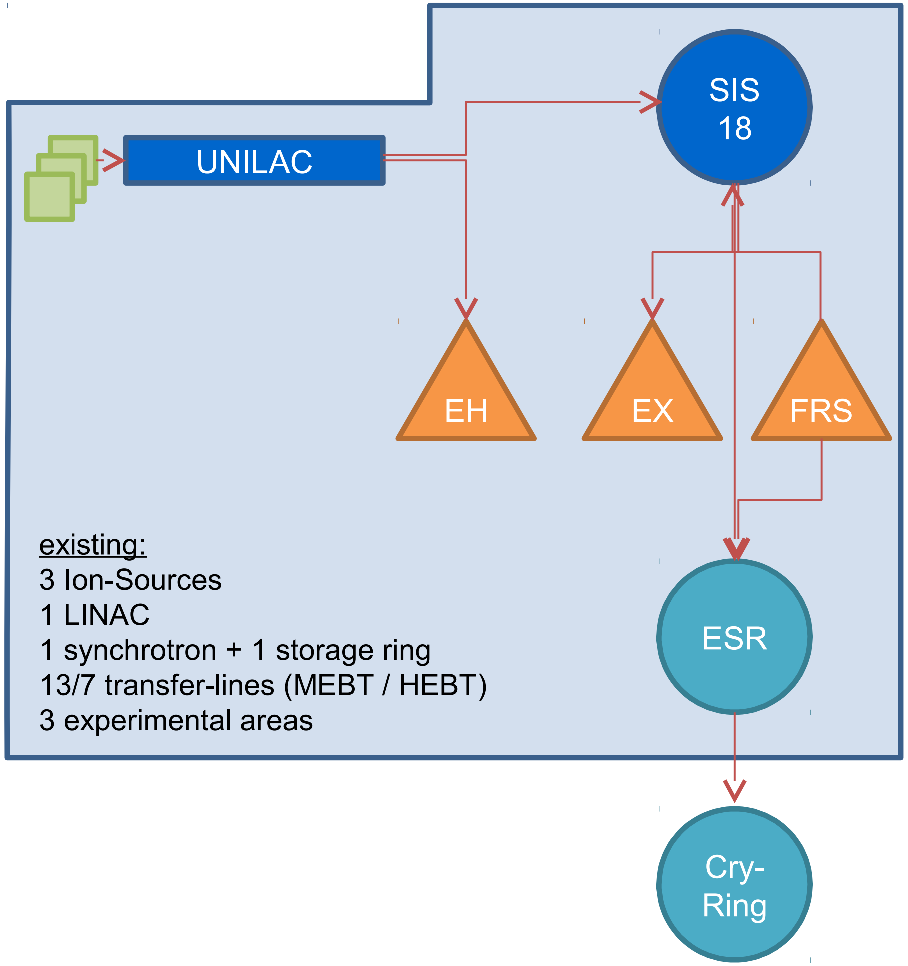
Accelerator Controls Retrofitting GSI → FAIR Transition in 2018



Accelerator Controls Retrofitting
GSI → FAIR Transition in 2018



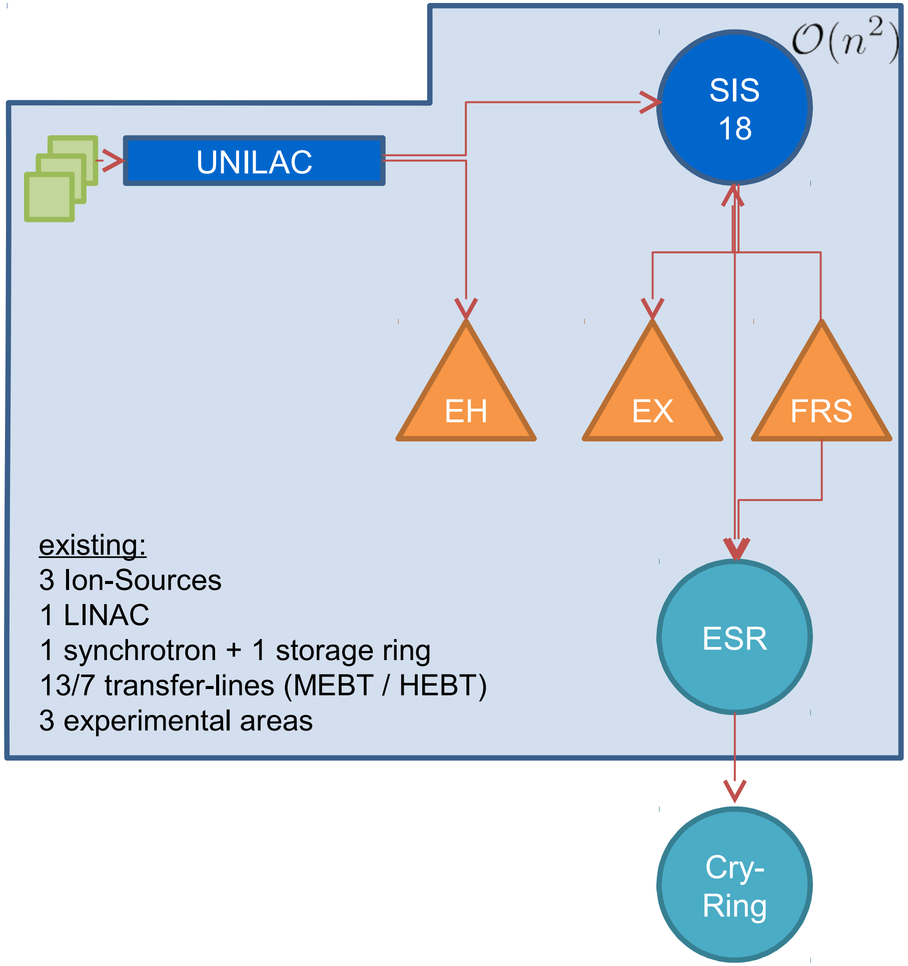
Migration Strategy
GSI 3 → FAIR 8½ (11+) accelerator(-like) machines



existing:
3 Ion-Sources
1 LINAC
1 synchrotron + 1 storage ring
13/7 transfer-lines (MEBT / HEBT)
3 experimental areas

Migration Strategy

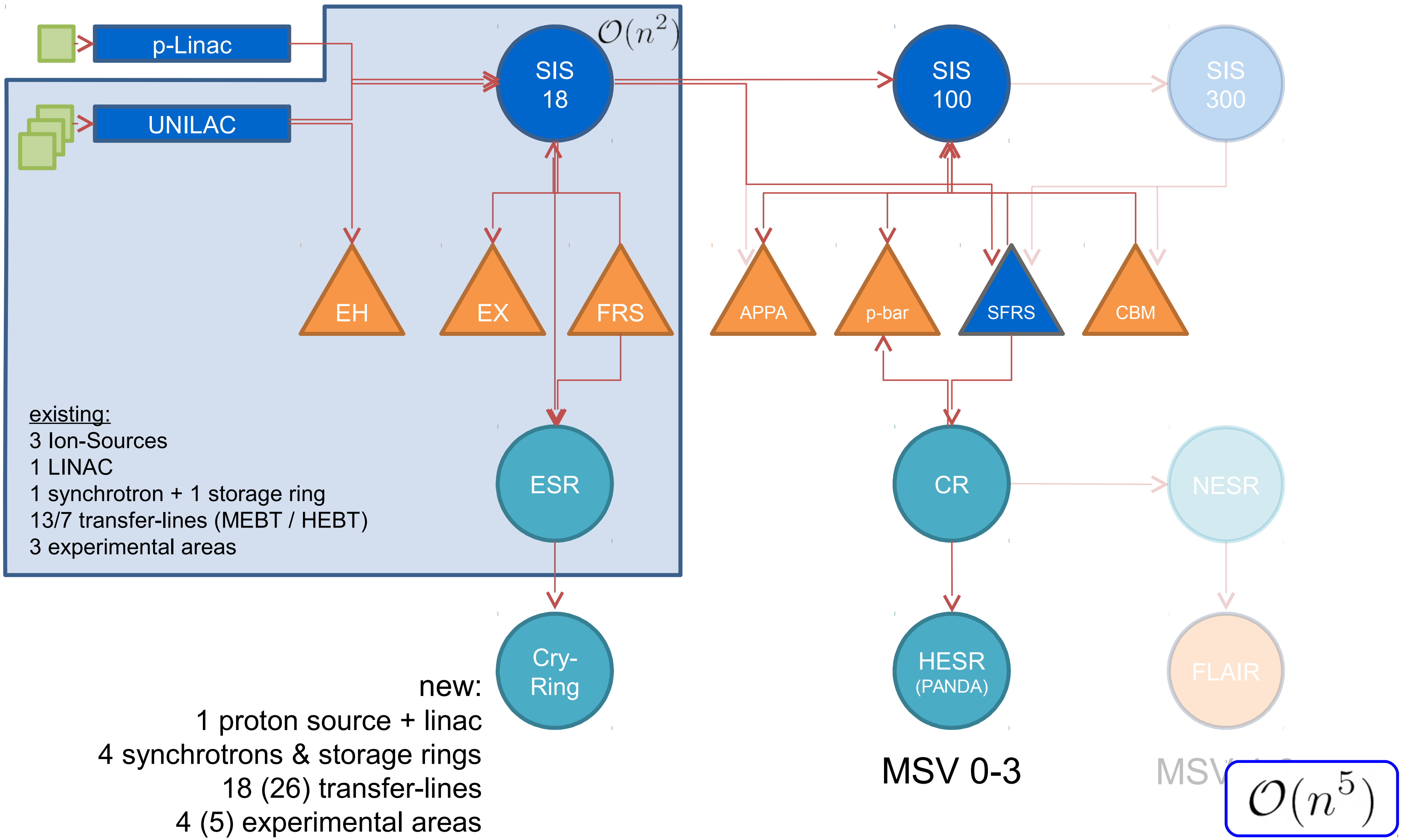
GSI 3 → FAIR 8½ (11+) accelerator(-like) machines



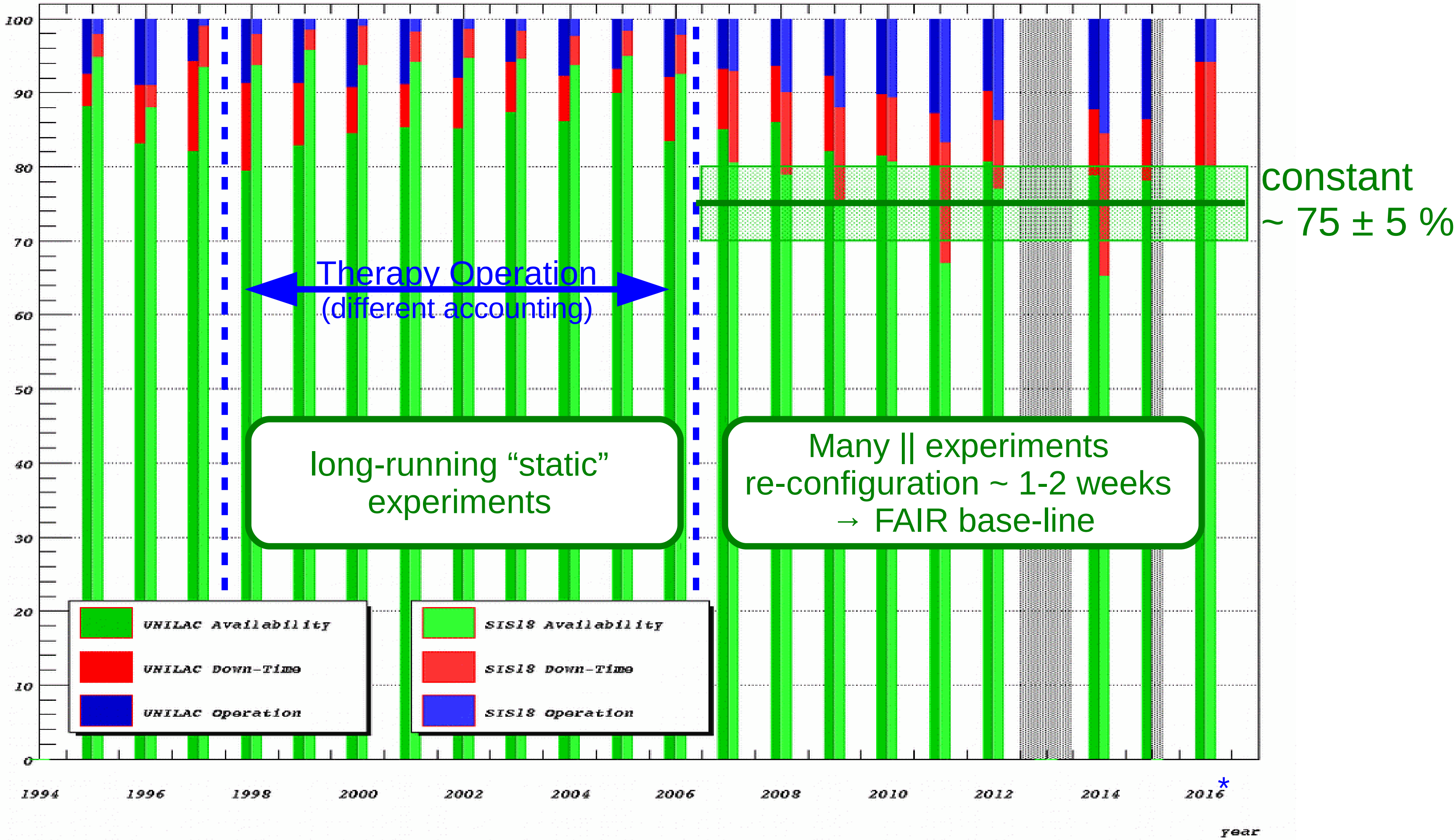
<u>existing:</u>
3 Ion-Sources
1 LINAC
1 synchrotron + 1 storage ring
13/7 transfer-lines (MEBT / HEBT)
3 experimental areas

Migration Strategy

GSI 3 → FAIR 8½ (11+) accelerator(-like) machines



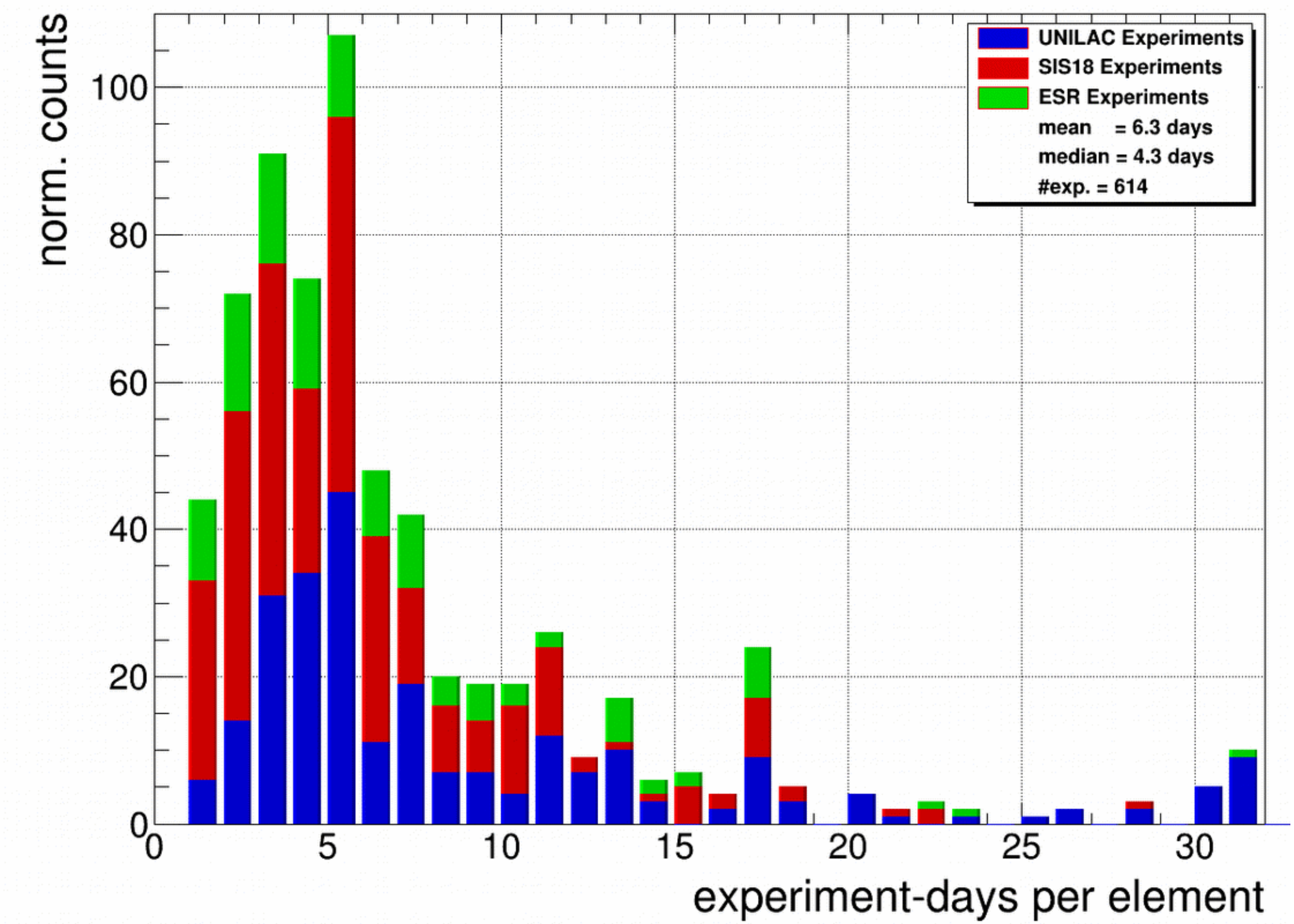
Accelerator Experience & Efficiency
1995-2016: U. Scheeler, S. Reimann, P. Schütt et al.



Based on: U. Scheeler, S. Reimann, P. Schütt et al., "Accelerator Operation Report", GSI Annual Scientific Reports 1992 – 2015 + 2016 (D. Severin)
https://www.gsi.de/en/work/research/library_documentation/gsi_scientific_reports.htm
N.B. ion source exchanges are factored out from UNILAC & SIS18 data (~ constant overhead)
Availability: *experiments + detector tests + machine development + beam to down-stream accelerators;*
Down-time: *unscheduled down-time + standby;* Operation: *accelerator setup + re-tuning*

* 2018 operation limitations:
- only ½ UNILAC (w/o A3 & A4)
- only 1 element in SIS18

Accelerator Experience & Efficiency
a closer look on Exp. Statistics 2006-2016*



*see GSI annual reports
2015/16 data courtesy D. Severin

- Much larger facility, cannot reliably extrapolate from present 'UNILAC→SIS18→ESR' operation to requirements for FAIR (9+ resp. 13 accelerators, higher/unsafe intensities, more users)

FAIR Commissiong &Controls Recap:
Global Requirements & Constraints

- Much larger facility, cannot reliably extrapolate from present 'UNILAC→SIS18→ESR' operation to requirements for FAIR (9+ resp. 13 accelerators, higher/unsafe intensities, more users)
- Will be in a constant flux of frequent adaptations to new cycles/beam parameters, etc. present estimate:
  - avg. experiment run: ~~~ 1-2 weeks~~ → 5-6 days many new storage rings and transfer lines with high(er) complexity → machine setup time-scale
  - high-intensity operation requires more and better fine-tuning
    - dynamic vacuum, activation & machine protection (mainly septa, instrumentation, etc.)
  - limited operator resources: 4-5 (beam operation) + 1-2 (infrastructure, cryo)

FAIR Commissiong &Controls Recap:
Global Requirements & Constraints

- Much larger facility, cannot reliably extrapolate from present 'UNILAC→SIS18→ESR' operation to requirements for FAIR (9+ resp. 13 accelerators, higher/unsafe intensities, more users)
- Will be in a constant flux of frequent adaptations to new cycles/beam parameters, etc. present estimate:
  - avg. experiment run: ~~~ 1-2 weeks~~ → 5-6 days many new storage rings and transfer lines with high(er) complexity → machine setup time-scale
  - high-intensity operation requires more and better fine-tuning
    - dynamic vacuum, activation & machine protection (mainly septa, instrumentation, etc.)
  - limited operator resources: 4-5 (beam operation) + 1-2 (infrastructure, cryo)

→ need to be smart and develop an efficient commissioning procedure, training and tools to facilitate fast turn-around and maintain (or improve) present operational efficiency

Fundamental FC²-Principles: Reworking, re-optimisation is inefficient and costly

Proposal to follow a long-term strategy and 'lean principles':

- Continuous improvement
  - Right processes to produce right results and for getting it right the first time
    - *commissioning procedures as evolving operation standard*
    - system integration: definition of what, how and when (prioritisation) is needed
  - Prevention of inefficiencies, inconsistencies & waste by design
    - 'poka-yoke' or 'error proofing' principle – culture of stopping and fixing
      1. early, when and where they occur (at the source)
      2. with low-intensity beam rather than with high-intensity beam
      3. addressing first basic parameters before complex higher-order effects
    - Examples:
      - first fix injection, trajectory, orbit, Q/Q' before addressing space-charge or slow-extraction problems
      - important losses for low-intensity beam have larger impact for high-intensity beam (↔ activation)
        - pilot-beam concept: always verify machine safety with low-intensities before moving on to high-intensity beams

- Respect for people – "develop people, then build products"
  - optimise operation ↔ smart tools & procedures, e.g. beam-based feedbacks, sequencer, …
    - automate routine task so that operator talents are utilised and focused on more important tasks
  - training, investment in and development of people – minimise overburden/strain of personnel
  - FAIR is a large facility and needs wider support: communicate concepts and strategy to broader base → FC²WG

Fundamental FC²-Principles: Reworking, re-optimisation is inefficient and costly

Proposal to follow a long-term strategy and 'lean principles':

- Continuous improvement
  - Right processes to produce right results and for getting it right the first time
    - *commissioning procedures as evolving operation standard*
    - system integration: definition of what, how and when (prioritisation) is needed
  - Prevention of inefficiencies, inconsistencies & waste by design
    - 'poka-yoke' or 'error proofing' principle – culture of stopping and fixing
      1. early, when and where they occur (at the source)
      2. with low-intensity beam rather than with high-intensity beam
      3. addressing first basic parameters before complex higher-order effects
    - Examples:
      - first fix injection, trajectory, orbit, Q/Q' before addressing space-charge or slow-extraction problems
      - important losses for low-intensity beam have larger impact for high-intensity beam (↔ activation)
        - pilot-beam concept: always verify machine safety with low-intensities before moving on to high-intensity beams

- Respect for people – “develop people, then build products”
  - optimise operation ↔ smart tools & procedures, e.g. beam-based feedbacks, sequencer, …
    - automate routine task so that operator talents are utilised and focused on more important tasks
  - training, investment in and development of people – minimise overburden/strain of personnel
  - FAIR is a large facility and needs wider support: communicate concepts and strategy to broader base → FC²WG

Fundamental FC²-Principles: Reworking, re-optimisation is inefficient and costly

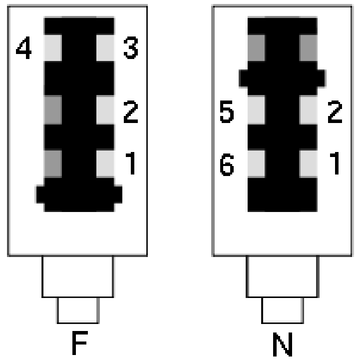
Proposal to follow a long-term strategy and 'lean principles':

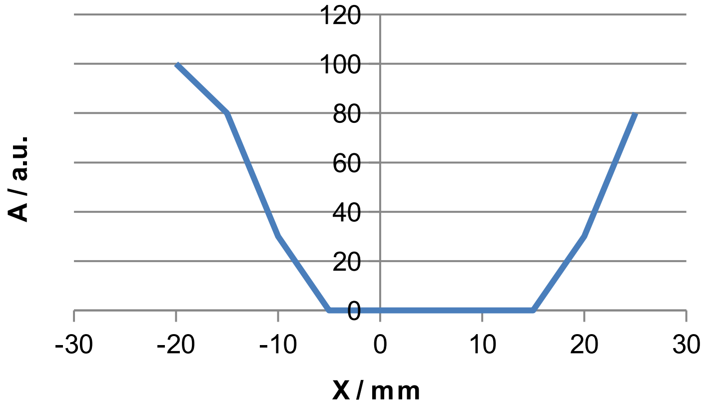
- Continuous improvement
  - Right processes to produce right results and for getting it right the first time
    - *commissioning procedures as evolving operation standard*
    - system integration: definition of what, how and when (prioritisation) is needed
  - Prevention of inefficiencies, inconsistencies & waste by design
    - 'poka-yoke' or 'error proofing' principle – culture of stopping and fixing
      1. early, when and where they occur (at the source)
      2. with low-intensity beam rather than with high-intensity beam
      3. addressing first basic parameters before complex higher-order effects
    - Examples:
      - first fix injection, trajectory, orbit, Q/Q' before addressing space-charge or slow-extraction problems
      - important losses for low-intensity beam have larger impact for high-intensity beam (↔ activation)
        - pilot-beam concept: always verify machine safety with low-intensities before moving on to high-intensity beams

- Respect for people – “develop people, then build products”
  - optimise operation ↔ smart tools & procedures, e.g. beam-based feedbacks, sequencer, …
    - automate routine task so that operator talents are utilised and focused on more important tasks
  - training, investment in and development of people – minimise overburden/strain of personnel
  - FAIR is a large facility and needs wider support: communicate concepts and strategy to broader base → FC²WG

Poka-Yoke (ポカヨケ) – 'Mistake-Proofing'

- Origin:
  - to avoid (yokeru) inadvertent errors (poka)
  - industrial processes designed to prevent human errors
    - Concept by Shigeo Shingo: 'Toyota Production System' (TPS, aka. 'lean' systems)
  - minimise common mistakes, procedural errors, etc. affecting machine performance and protection

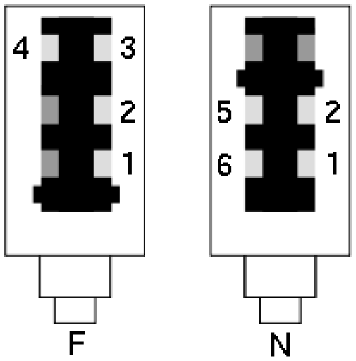


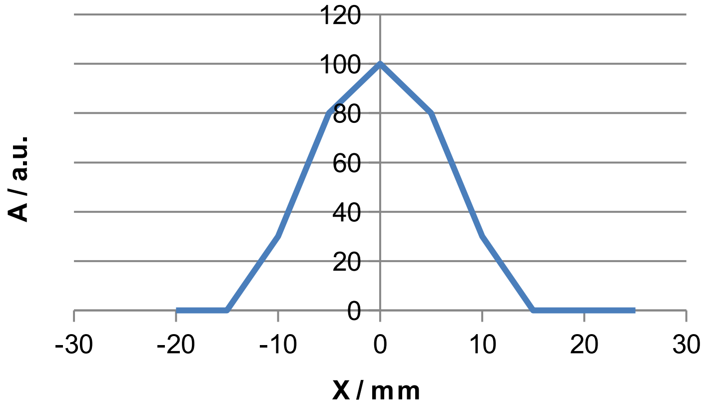


- Real-World Examples:
  - Polarity protection of electrical plugs (e.g. phone, Ethernet cable)
    - SIS18 profile grid connectors
  - Procedures: e.g. ATM machine: need to retrieve card before money is released (↔ prevents missing card)

Poka-Yoke (ポカヨケ) – 'Mistake-Proofing'

- Origin:
  - to avoid (yokeru) inadvertent errors (poka)
  - industrial processes designed to prevent human errors
    - Concept by Shigeo Shingo: 'Toyota Production System' (TPS, aka. 'lean' systems)
  - minimise common mistakes, procedural errors, etc. affecting machine performance and protection



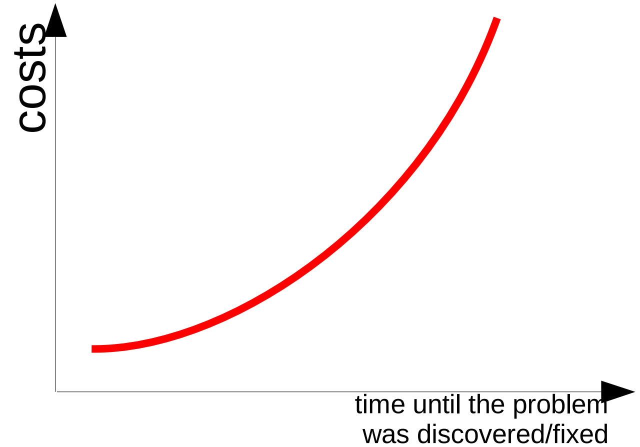


- Real-World Examples:
  - Polarity protection of electrical plugs (e.g. phone, Ethernet cable)
    - SIS18 profile grid connectors
  - Procedures: e.g. ATM machine: need to retrieve card before money is released (↔ prevents missing card)

Poka-Yoke (ポカヨケ) – 'Mistake-Proofing'
Reaction-Time and Cost → “fix” errors early

Fix problems early, when and where they occur

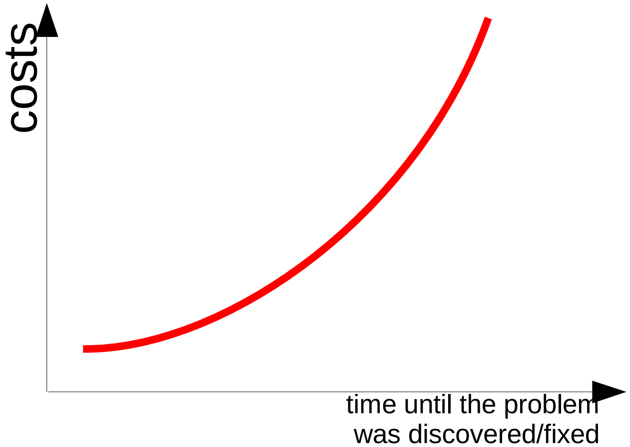
- Minimises procrastination of errors: “Safety starts with safe habits”!
  - big losses with big intensities → bad (activation)
  - large losses with small intensities → probably OK? … No!
    - requires paradigm change!
  - Interdependence between beam parameter & systems



Poka-Yoke (ポカヨケ) – 'Mistake-Proofing'
Reaction-Time and Cost → “fix” errors early

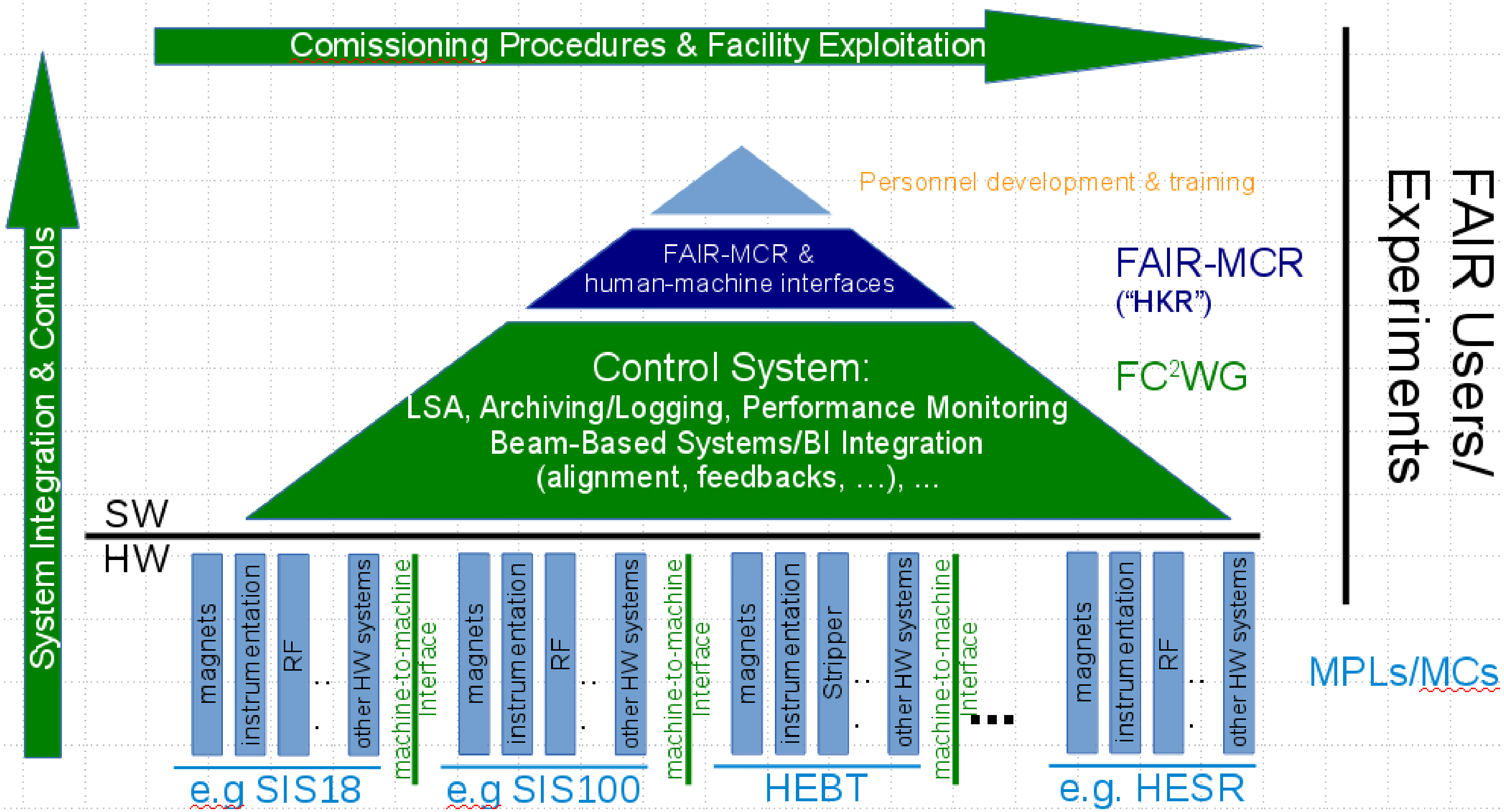
Fix problems early, when and where they occur

- Minimises procrastination of errors: “Safety starts with safe habits”!
  - big losses with big intensities → bad (activation)
  - large losses with small intensities → probably OK? … No!
    - requires paradigm change!
  - Interdependence between beam parameter & systems



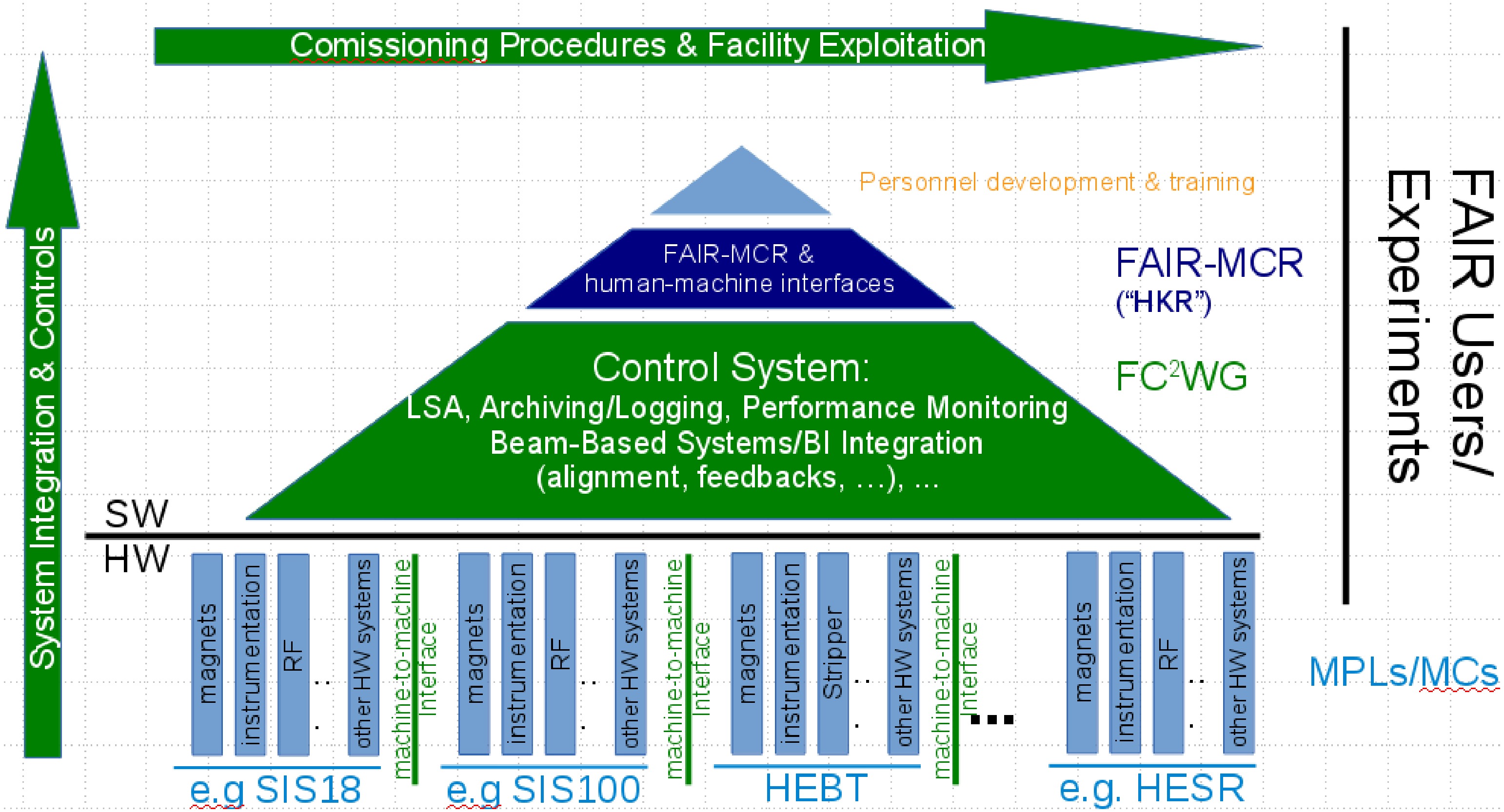
- Early indication of developing/not-yet-critical faults:
  - Post-Mortem analysis ('as good as new' SIL assurance)
  - Preventative maintenance → Sequencer
  - fix “domino effect” problems at the source not its symptoms
    - e.g. fix problems with low-intensity beam rather than with high-intensity beam (avoids revalidation of loss patterns, MPS setup, …)
    - e.g. fix basic accelerator parameters before moving on to higher-order effect (e.g. extraction/injection energy/trajectory → orbit → tune → chromaticity → optic → … → driving term s

An accelerator is more than the sum of its parts:



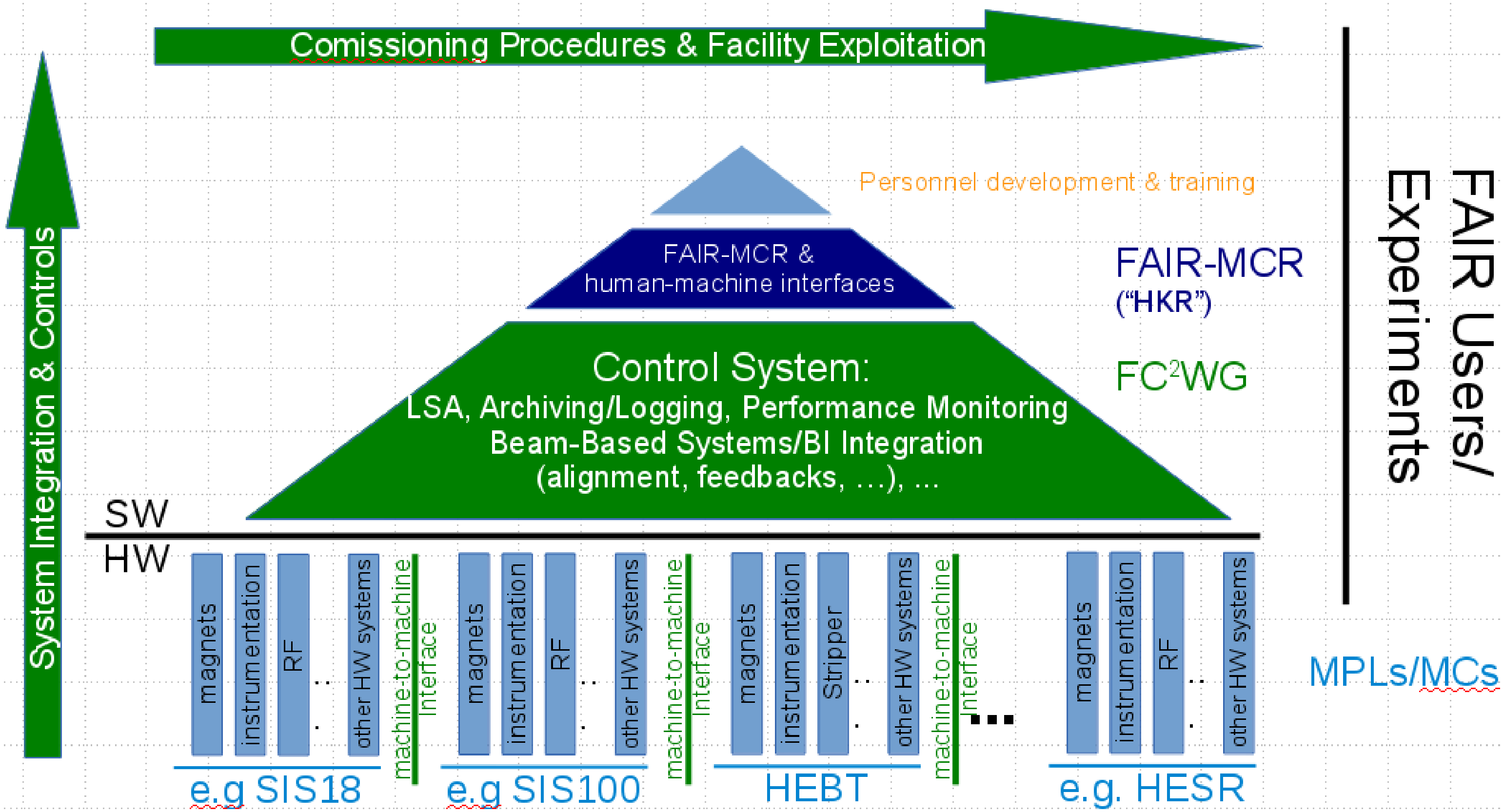
- FAIR Commissioning & Control Working Group
  - platform to discuss, coordinate and work-out FAIR commissioning and operation
  - open to all who can participate and contribute to this subject!

An accelerator is more than the sum of its parts:

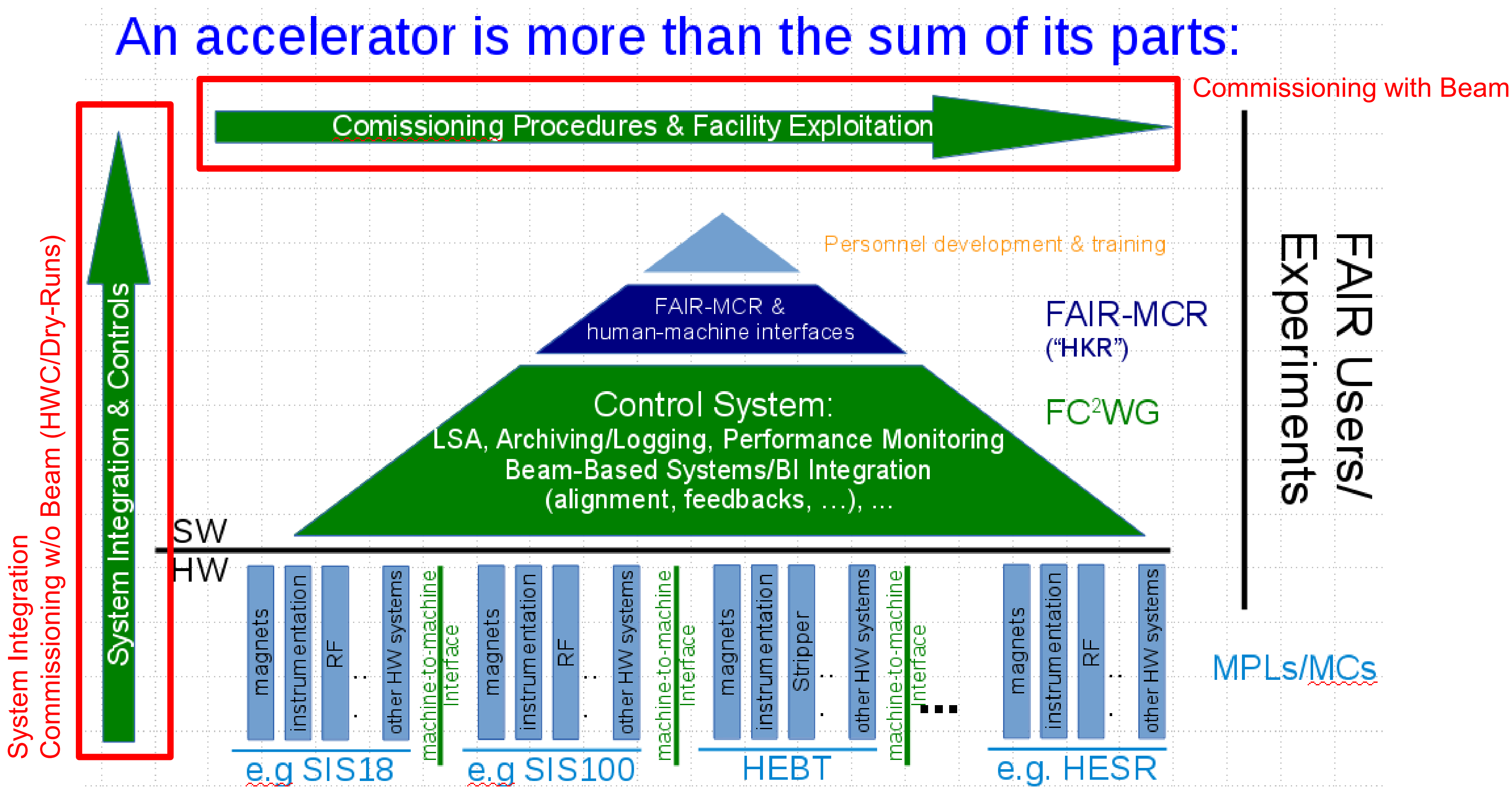


- FAIR Commissioning & Control Working Group
  - platform to discuss, coordinate and work-out FAIR commissioning and operation
  - open to all who can participate and contribute to this subject!

An accelerator is more than the sum of its parts:



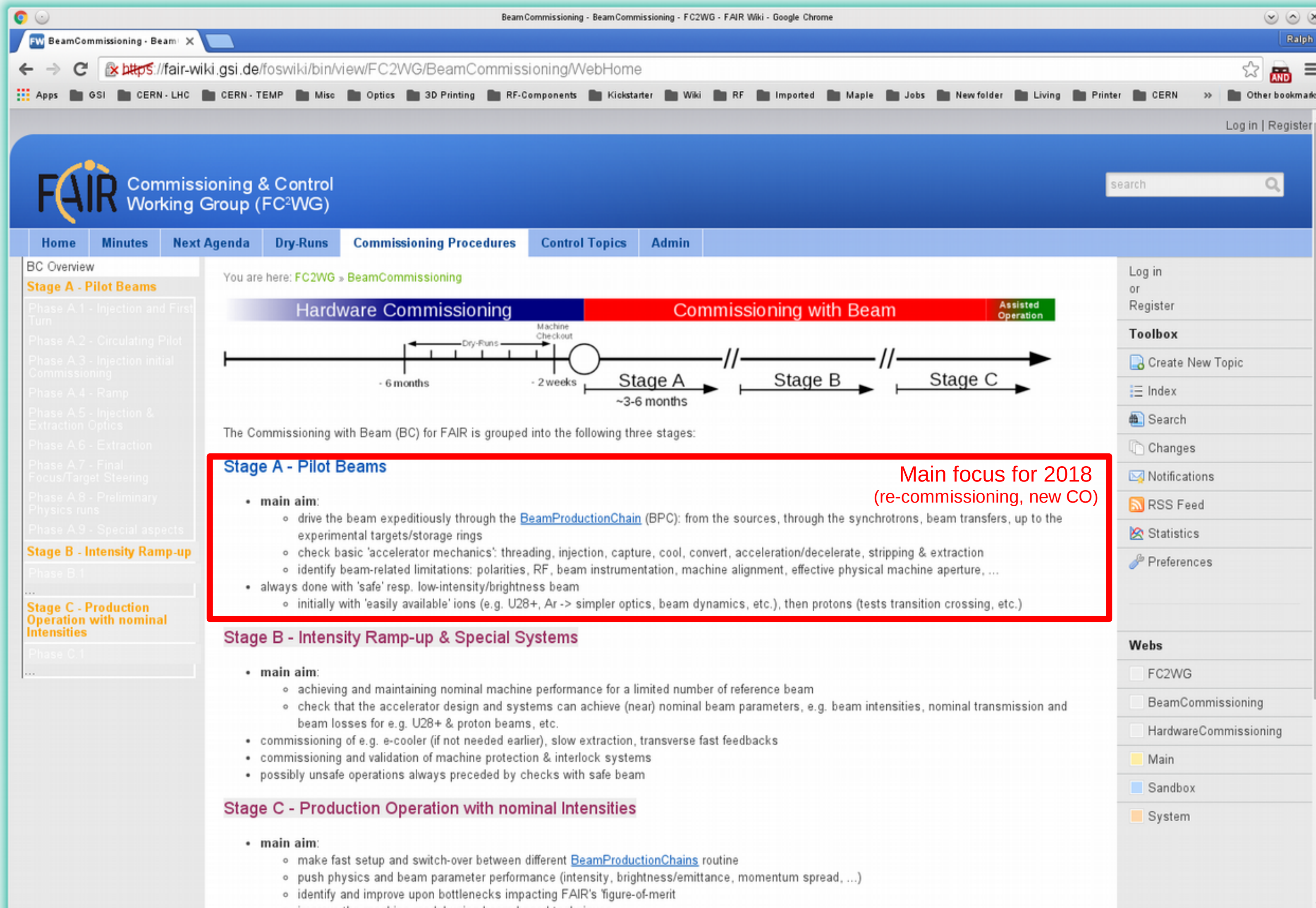
- FAIR Commissioning & Control Working Group
  - platform to discuss, coordinate and work-out FAIR commissioning and operation
  - open to all who can participate and contribute to this subject!



- FAIR Commissioning & Control Working Group
  - platform to discuss, coordinate and work-out FAIR commissioning and operation
  - open to all who can participate and contribute to this subject!

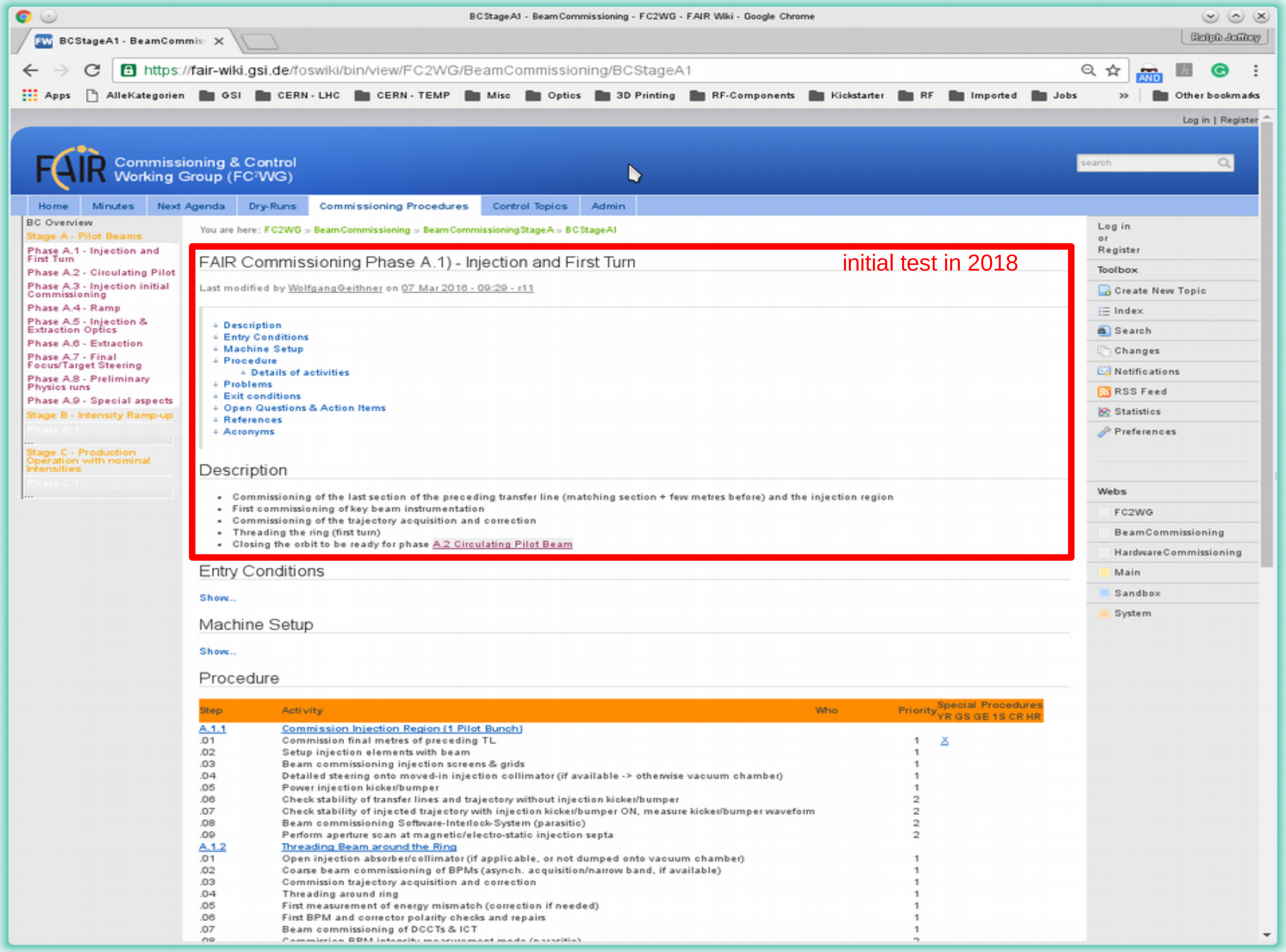
Example: FAIR Commissioning Procedures

https://fair-wiki.gsi.de/FC2WG/BeamCommissioning



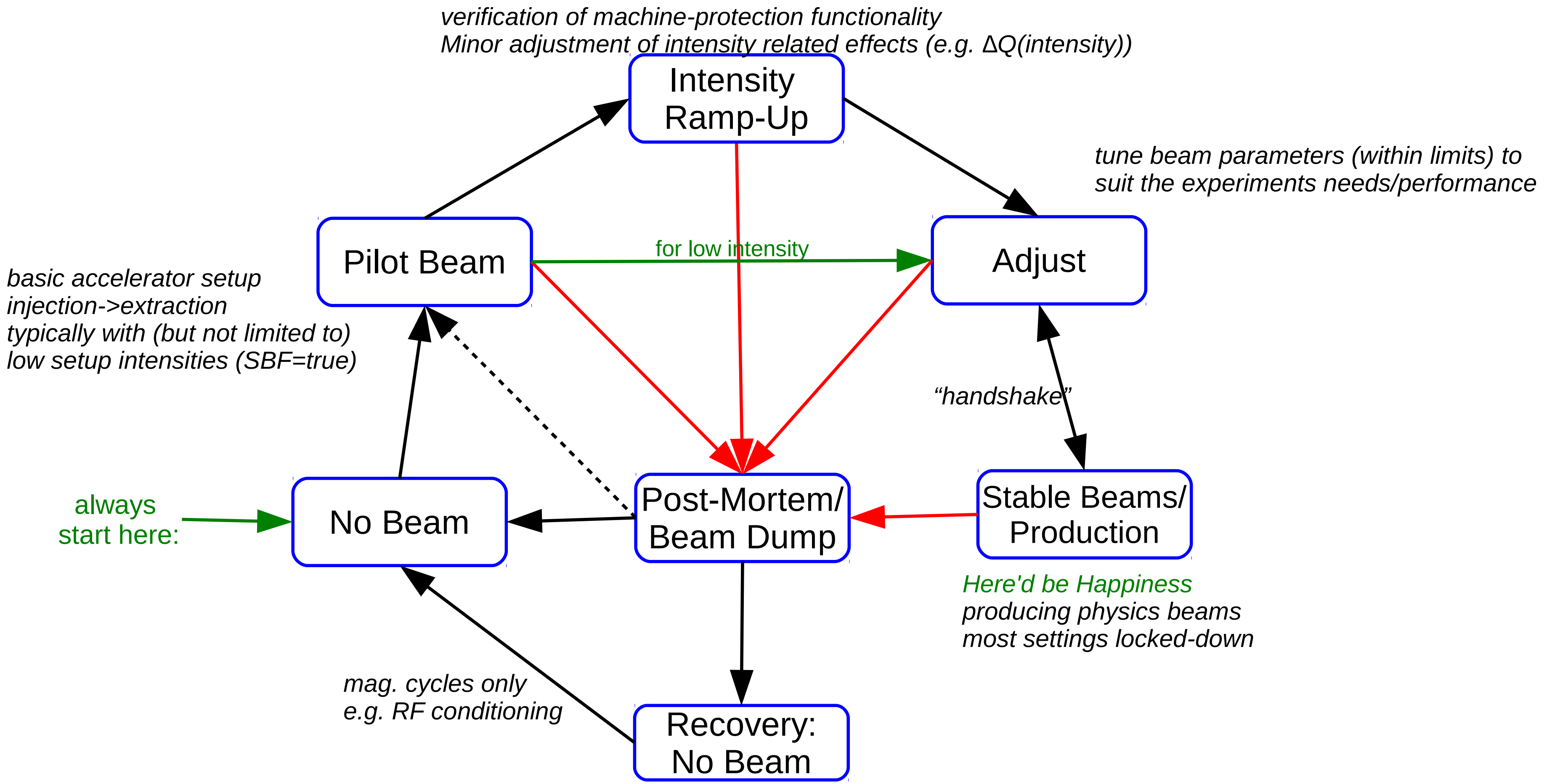
Example: FAIR Commissioning Procedures

https://fair-wiki.gsi.de/FC2WG/BeamCommissioning



FAIR Beam Modes – State Diagram

N.B. ‘mode’ := intended/targeted state of operation



normal operational path
error/fault case
low-intensity

N.B.:
1) omitted arrows to 'No Beam'/'Pilot Beam' for better visibility (always possible)
2) modes follow existing normal setup routine, initial transition acknowledged by operator, subsequent driven automatically by sequencer

Settings Protection & add. Actual States

- Management of Critical Settings – lock-down of critical machine settings depending on OP/MP scenario
  - tolerance bands depending on 'Accelerator' & 'Beam Modes': e.g. 'Pilot': fully open → 'Intensity Ramp-up' (limited 'safe range'. e.g. ΔQ < 0.01) → 'Adjust' (more stringent limits, e.g. only exp. target parameter) → 'Stable Beams' (only agreed settings, e.g. “beam-on-target position on 100 um level”)

Settings Protection & add. Actual States

- Management of Critical Settings – lock-down of critical machine settings depending on OP/MP scenario
  - tolerance bands depending on 'Accelerator' & 'Beam Modes': e.g. 'Pilot': fully open → 'Intensity Ramp-up' (limited 'safe range'. e.g. ΔQ < 0.01) → 'Adjust' (more stringent limits, e.g. only exp. target parameter) → 'Stable Beams' (only agreed settings, e.g. “beam-on-target position on 100 um level”)

- Beam Presence Flag (BPF) – indicates that cycle/settings have been validated with Pilot- or Physics-Beam in the recent past (< days, tbd.)
  - main usage: prevent high-intensity injections into an 'empty' machine with new untested magnetic settings or modified machine conditions
  - defined per accelerator or transfer-line segment (where necessary)

Settings Protection & add. Actual States

- Management of Critical Settings – lock-down of critical machine settings depending on OP/MP scenario
  - tolerance bands depending on 'Accelerator' & 'Beam Modes': e.g. 'Pilot': fully open → 'Intensity Ramp-up' (limited 'safe range'. e.g. ΔQ < 0.01) → 'Adjust' (more stringent limits, e.g. only exp. target parameter) → 'Stable Beams' (only agreed settings, e.g. "beam-on-target position on 100 um level")
- Beam Presence Flag (BPF) – indicates that cycle/settings have been validated with Pilot- or Physics-Beam in the recent past (< days, tbd.)
  - main usage: prevent high-intensity injections into an 'empty' machine with new untested magnetic settings or modified machine conditions
  - defined per accelerator or transfer-line segment (where necessary)
- Setup Beam Flag (SBF) – indicates beam used to setup the beam production chain (typically low-intensity)
  - defined per accelerator or transfer-line segment (where necessary)
  - SBF provides flexibility of masking interlocks during setup (e.g. MWPC/screen protection)
  - Used to enforce interlocks with high-intensity (primary) beam (↔ prevents the 'forgotten interlock syndrome')

Settings Protection & add. Actual States

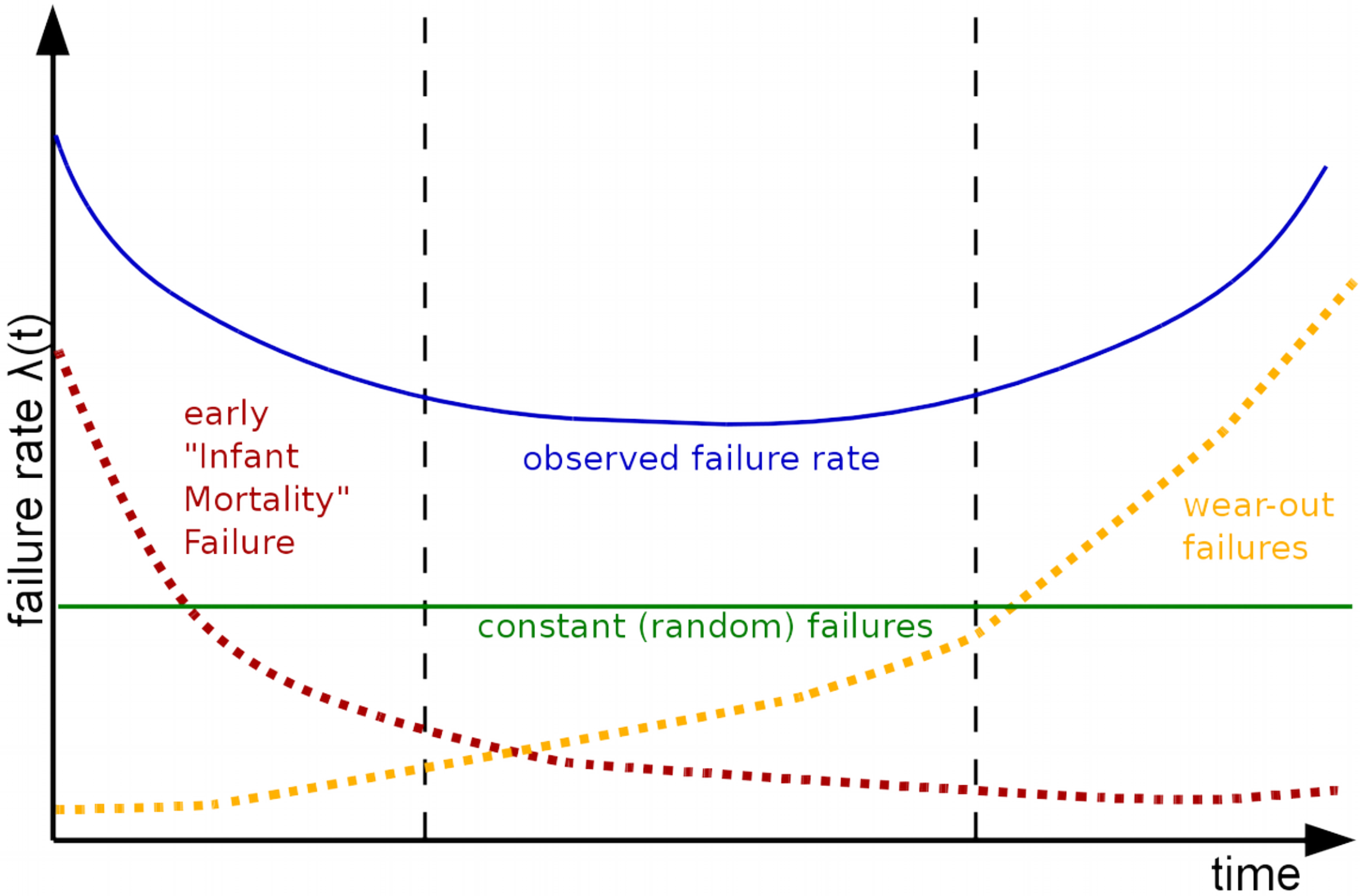
- Management of Critical Settings – lock-down of critical machine settings depending on OP/MP scenario
  - tolerance bands depending on 'Accelerator' & 'Beam Modes': e.g. 'Pilot': fully open → 'Intensity Ramp-up' (limited 'safe range'. e.g. ΔQ < 0.01) → 'Adjust' (more stringent limits, e.g. only exp. target parameter) → 'Stable Beams' (only agreed settings, e.g. “beam-on-target position on 100 um level”)

- Beam Presence Flag (BPF) – indicates that cycle/settings have been validated with Pilot- or Physics-Beam in the recent past (< days, tbd.)
  - main usage: prevent high-intensity injections into an 'empty' machine with new untested magnetic settings or modified machine conditions
  - defined per accelerator or transfer-line segment (where necessary)

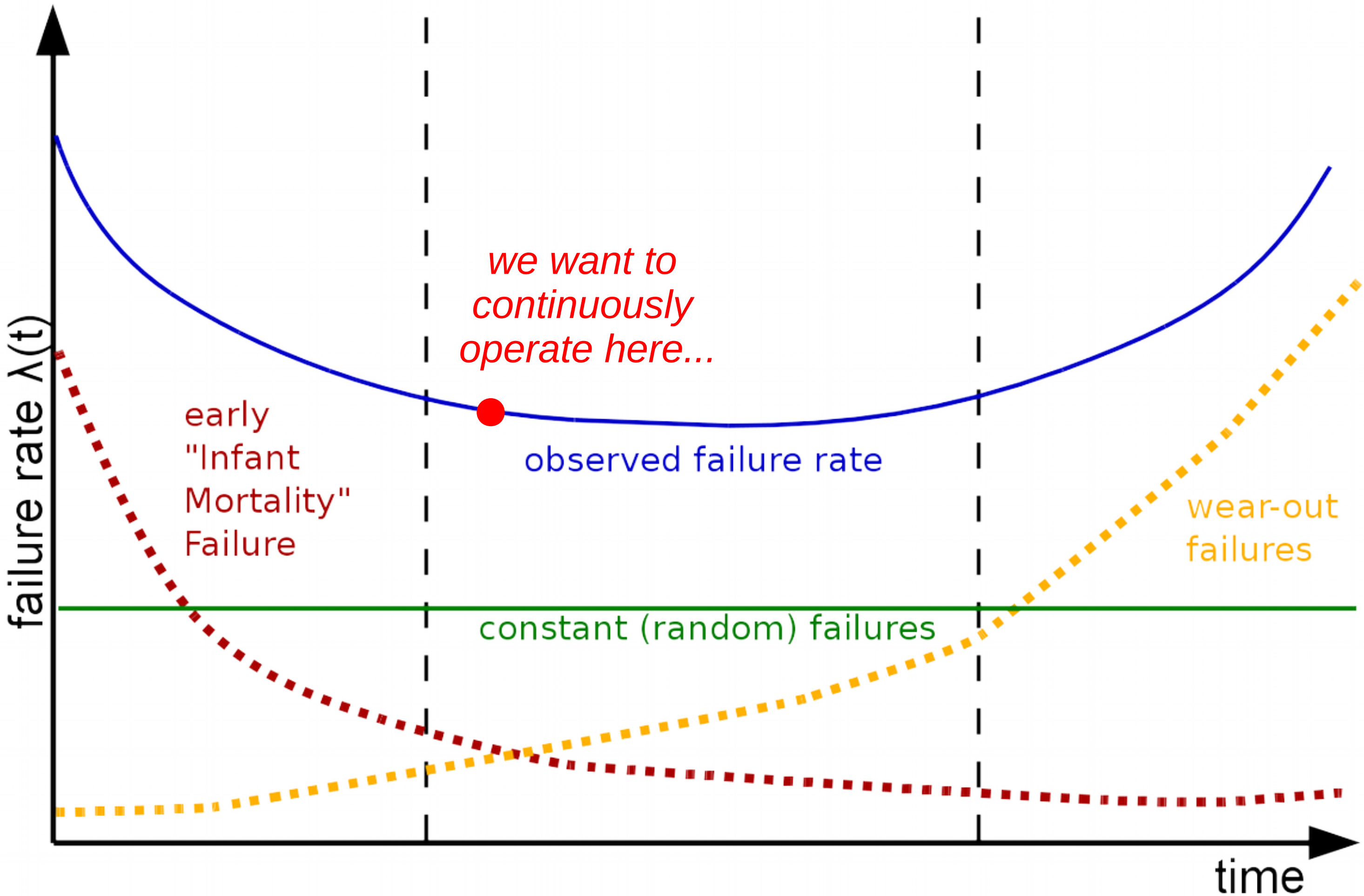
- Setup Beam Flag (SBF) – indicates beam used to setup the beam production chain (typically low-intensity)
  - defined per accelerator or transfer-line segment (where necessary)
  - SBF provides flexibility of masking interlocks during setup (e.g. MWPC/screen protection)
  - Used to enforce interlocks with high-intensity (primary) beam (↔ prevents the 'forgotten interlock syndrome')

- Injection/Extraction Permit – indicates if subsequent accelerator chain is ready (safe) to receive beam (→ fast beam aborts, discussed later)

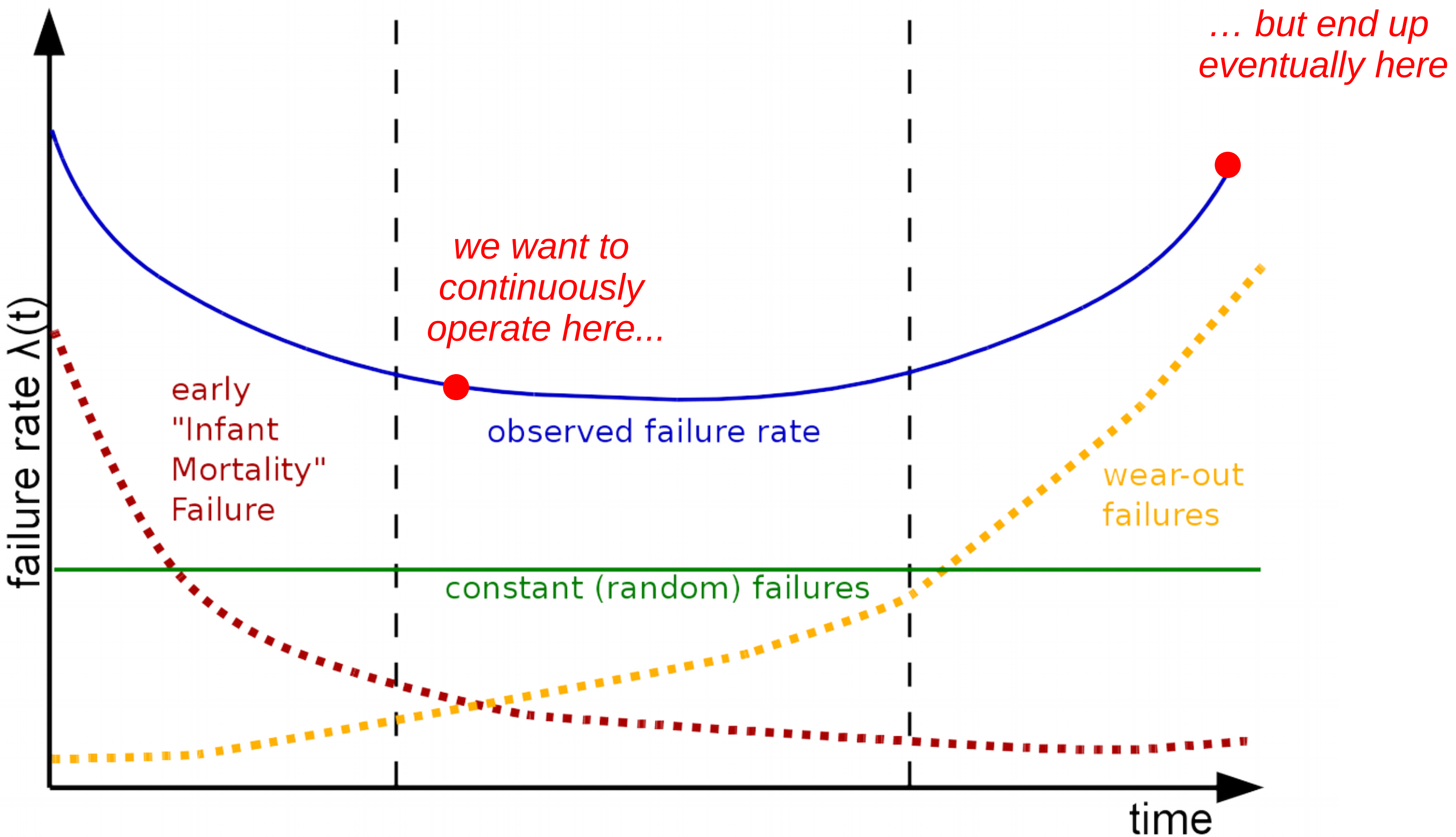
Problem definition: classical bath-tub curve – in an ideal/naïve world:



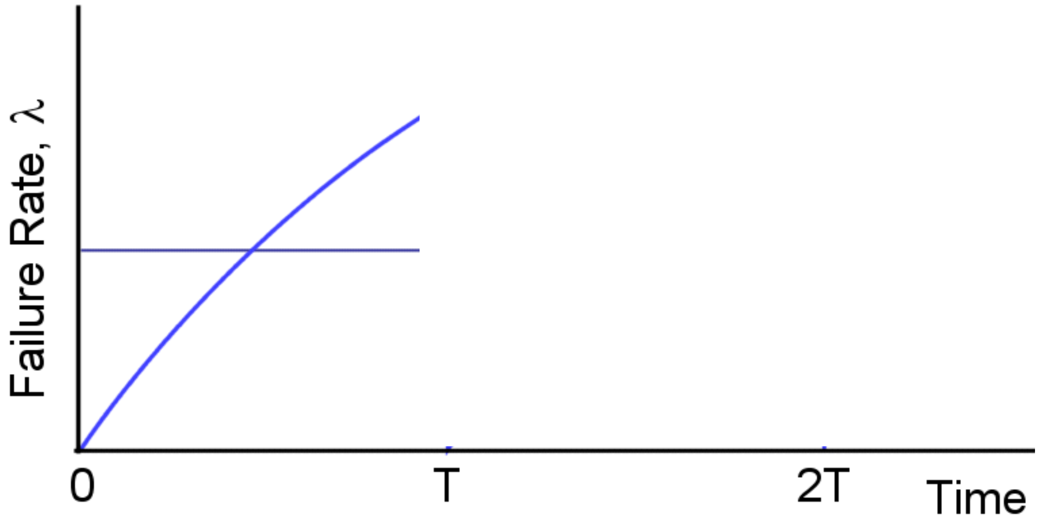
Problem definition: classical bath-tub curve – in an ideal/naïve world:



Problem definition: classical bath-tub curve – in an ideal/naïve world:



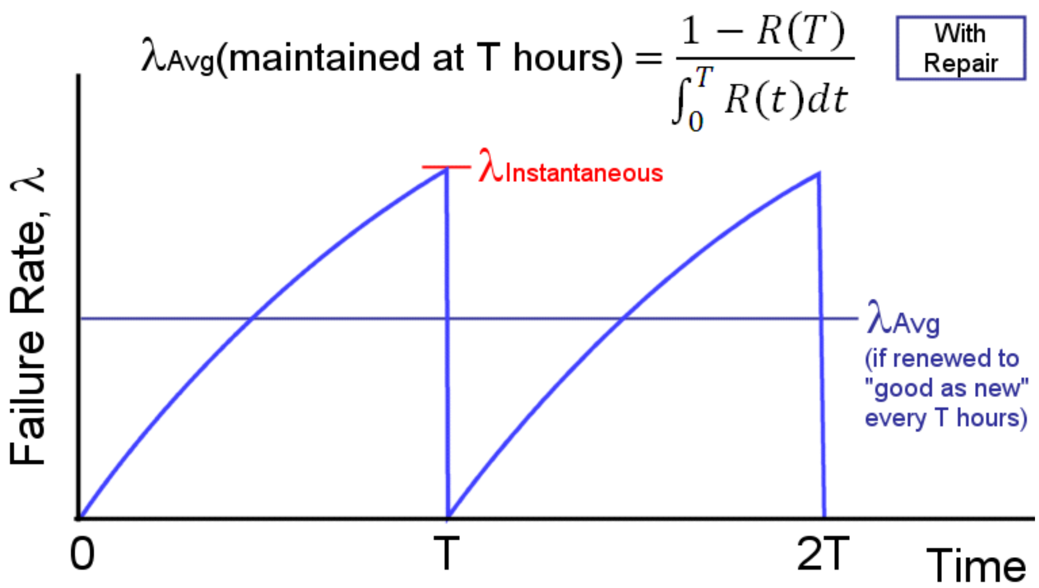
- Common improvement options: choice of materials, safety margin in material/parameter properties & adding redundancy



Reliability Engineering II/III

predictive maintenance or 'as good as new' checks

- Common improvement options: choice of materials, safety margin in material/parameter properties & adding redundancy
- However: redundancy provides only limited reliability gain
  → key to high reliability: performance surveillance + checks
  → 'as good as new' system validation

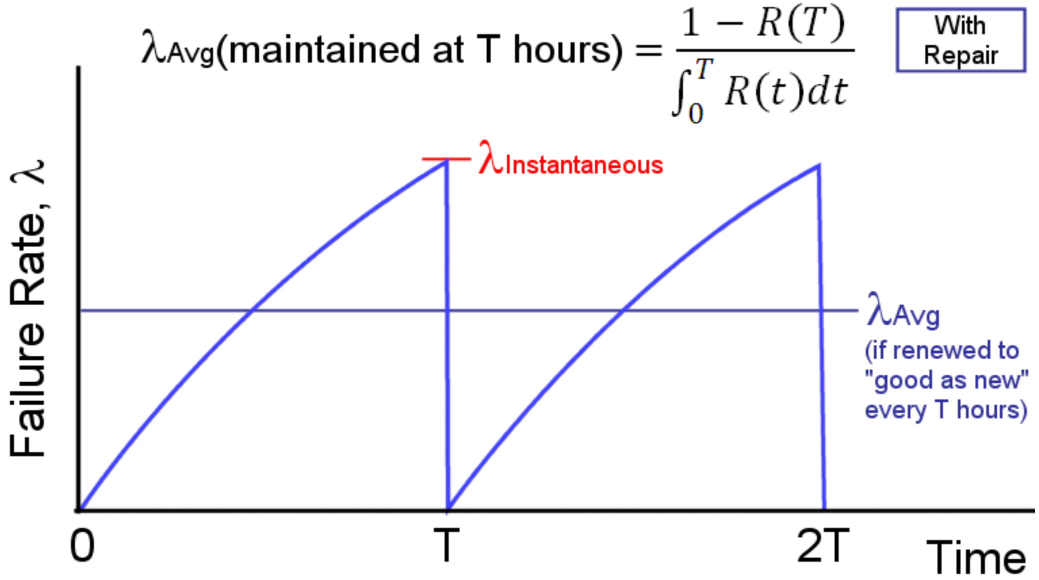


Reliability Engineering II/III

predictive maintenance or ‘as good as new’ checks

- Common improvement options: choice of materials, safety margin in material/parameter properties & adding redundancy
- However: redundancy provides only limited reliability gain
  → key to high reliability: performance surveillance + checks
  → ‘as good as new’ system validation
  - technical implementations at FAIR: Sequencer & Post-Mortem System

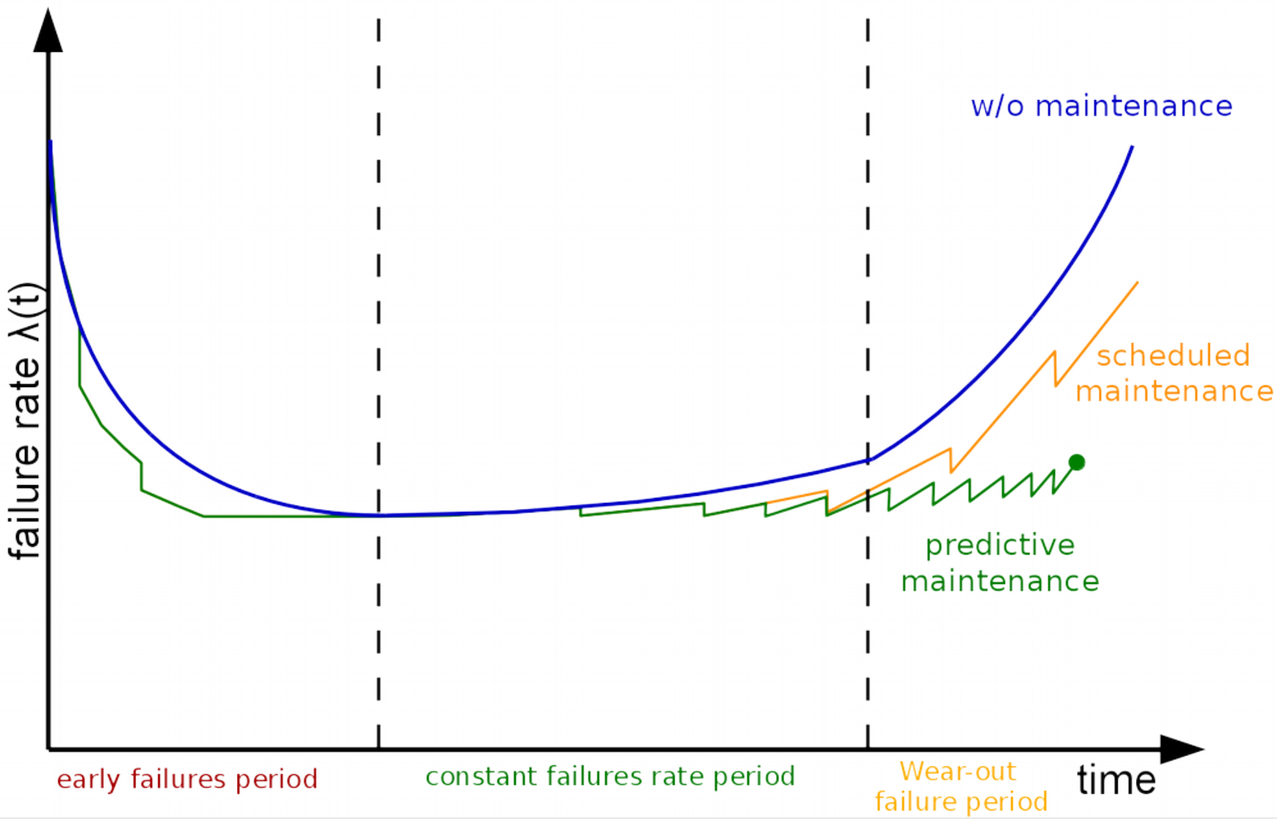
$$\lambda_{Avg}(\text{maintained at T hours}) = \frac{1 - R(T)}{\int_0^T R(t)dt}$$



Reliability Engineering III/III

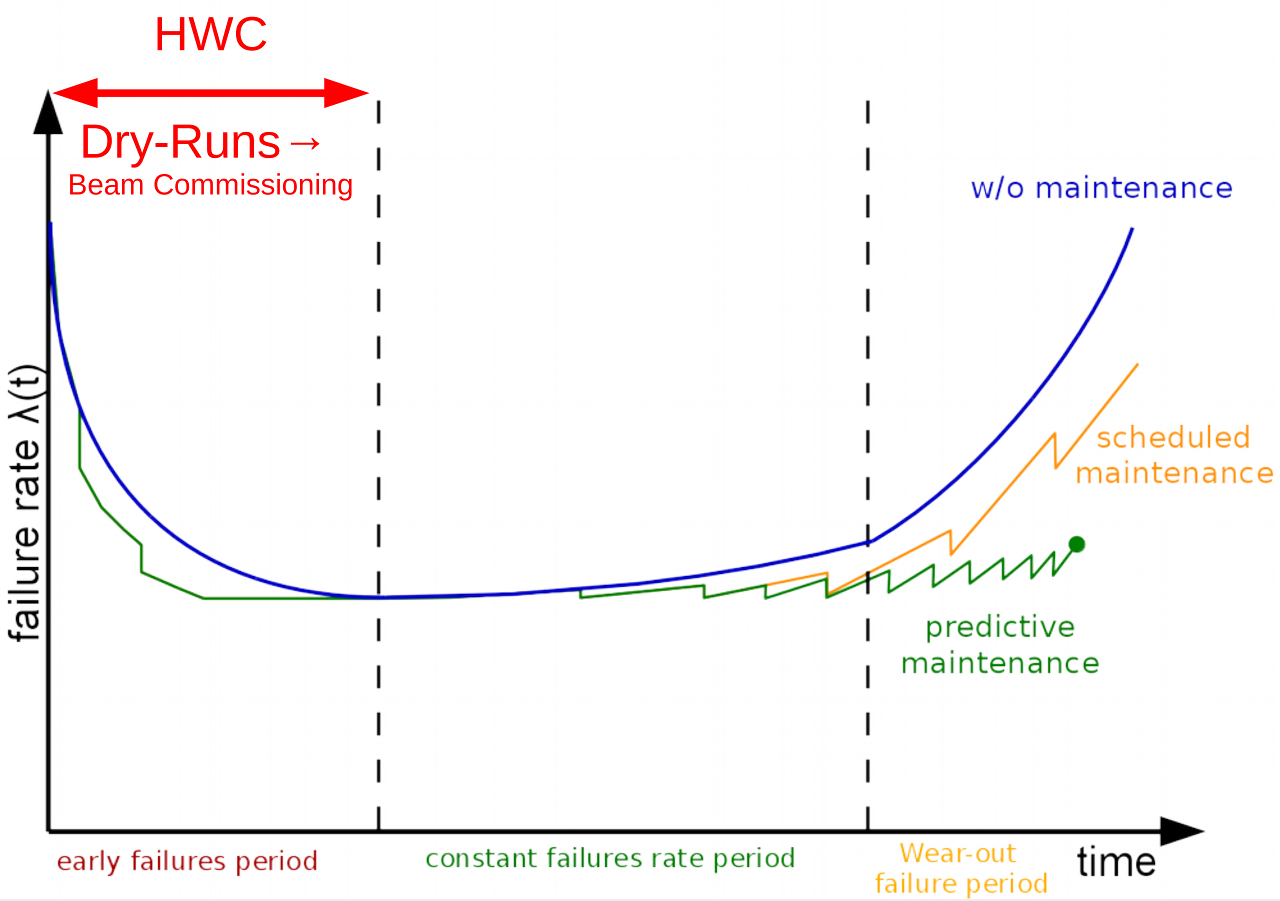
predictive maintenance or ‘as good as new’ checks

- Sequencer (OP triggered) and Post-Mortem (MP triggered) checks



Reliability Engineering III/III
predictive maintenance or ‘as good as new’ checks

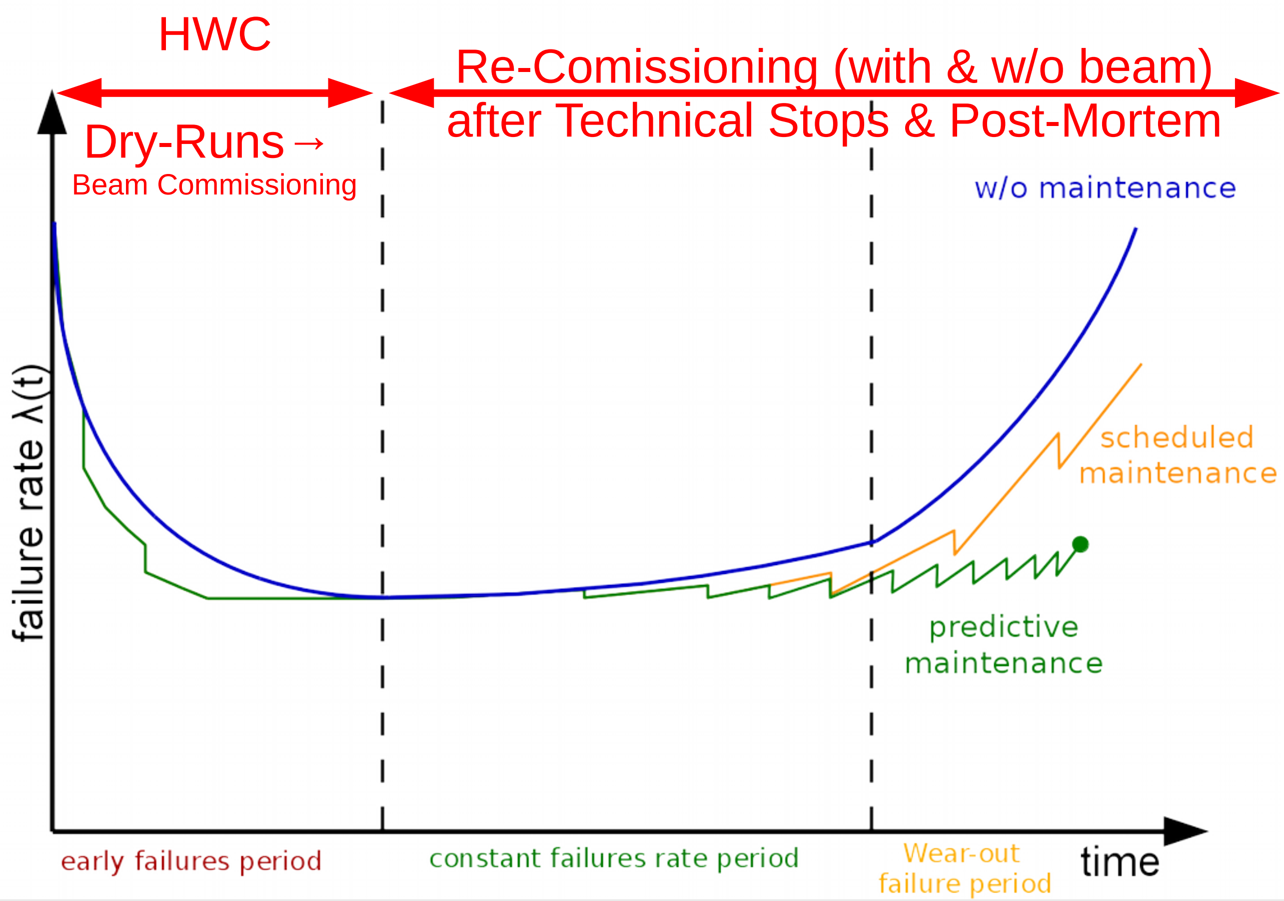
- Sequencer (OP triggered) and Post-Mortem (MP triggered) checks

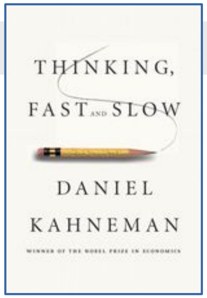


Reliability Engineering III/III

predictive maintenance or ‘as good as new’ checks

- Sequencer (OP triggered) and Post-Mortem (MP triggered) checks

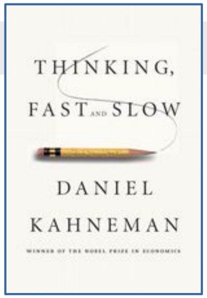




- Kahneman studies[1] famously described the two different ways of our brain works and forms thoughts

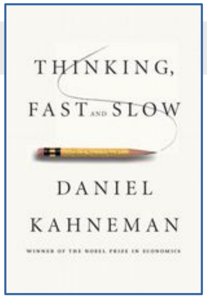
  (N.B. → warded the 2002 nobel prize):

[1]Daniel Kahneman, “Thinking, Fast and Slow”, Farrar, Straus and Giroux, 2011



- Kahneman studies[1] famously described the two different ways of our brain works and forms thoughts (N.B. → warded the 2002 nobel prize):
- System 1: “Fast”, automatic, frequent, emotional, stereotypic, subconscious.
  - role: assess the situation, deliver updates
  - based on past experience, intuition and learned experience
  - prone to cognitive bias, logic faults
  - Saves “mental energy”
    → usually preferred

[1]Daniel Kahneman, “Thinking, Fast and Slow”, Farrar, Straus and Giroux, 2011



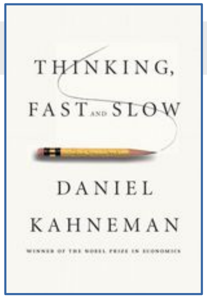
- Kahneman studies[1] famously described the two different ways of our brain works and forms thoughts
(N.B. → warded the 2002 nobel prize):

- System 1: “Fast”, automatic, frequent, emotional, stereotypic, subconscious.
  - role: assess the situation, deliver updates
  - based on past experience, intuition and learned experience
  - prone to cognitive bias, logic faults
  - Saves “mental energy” → usually preferred

- System 2: “Slow”, effortful, infrequent, logical, calculating, conscious.
  - role: seeks new/missing information, makes decisions
  - Can keep only up to five aspects in active memory
  - Requires (sometimes significant) ‘mental energy’ → unfavoured

[1]Daniel Kahneman, “Thinking, Fast and Slow”, Farrar, Straus and Giroux, 2011

Facility Complexity & the Human Factor
Thinking Fast and Slow & “Human Multitasking”



- Kahneman studies[1] famously described the two different ways of our brain works and forms thoughts
  (N.B. → warded the 2002 nobel prize):

- System 1: “Fast”, automatic, frequent, emotional, stereotypic, subconscious.
  - role: assess the situation, deliver updates
  - based on past experience, intuition and learned experience
  - prone to cognitive bias, logic faults
  - Saves “mental energy” → usually preferred

- System 2: “Slow”, effortful, infrequent, logical, calculating, conscious.
  - role: seeks new/missing information, makes decisions
  - Can keep only up to five aspects in active memory
  - Requires (sometimes significant) ‘mental energy’ → unfavoured

… performing multiple complex, high-risk tasks is a actually very bad idea
→ unnecessary strain on operators, machine experts and operational risk

[1] Daniel Kahneman, “Thinking, Fast and Slow”, Farrar, Straus and Giroux, 2011

- Wikipedia: *“... is the documented process of assuring that a computerized system does exactly what it is designed to do in a consistent and reproducible manner. The validation process begins with the system proposal/requirements definition and continues until system retirement and retention of the e-records based on regulatory rules”*

FAIR Sequencer & Post-Mortem
or: 'Computerized System Validation'

- Wikipedia: *"... is the documented process of assuring that a computerized system does exactly what it is designed to do in a consistent and reproducible manner. The validation process begins with the system proposal/requirements definition and continues until system retirement and retention of the e-records based on regulatory rules"*
  - or for FAIR equipment/machine experts: Java-based automatisation of the system integration, Site Acceptance Tests (SATs) and/or Beam Commissioning (BC) procedures

FAIR Sequencer & Post-Mortem

or: ‘Computerized System Validation’

- Wikipedia: “*... is the documented process of assuring that a computerized system does exactly what it is designed to do in a consistent and reproducible manner. The validation process begins with the system proposal/requirements definition and continues until system retirement and retention of the e-records based on regulatory rules*”
  - or for FAIR equipment/machine experts: Java-based automatisation of the system integration, Site Acceptance Tests (SATs) and/or Beam Commissioning (BC) procedures
  - or for FAIR software developer: JUnit tests for hardware-based and other complex systems

FAIR Sequencer & Post-Mortem
or: 'Computerized System Validation'

- Wikipedia: *"... is the documented process of assuring that a computerized system does exactly what it is designed to do in a consistent and reproducible manner. The validation process begins with the system proposal/requirements definition and continues until system retirement and retention of the e-records based on regulatory rules"*
  - or for FAIR equipment/machine experts: Java-based automatisation of the system integration, Site Acceptance Tests (SATs) and/or Beam Commissioning (BC) procedures
  - or for FAIR software developer: JUnit tests for hardware-based and other complex systems

- Main aspects:
  - test automation → reproducibility, consistency, true parallelism and multi-tasking
  - identification & localisation of faults
  - follow-up/handling of tests that can last over several hours → days
  - Machine protection (post-mortem): online validation of safety integrity level
  - Machine availability tracking and optimisation:
    - Continuous improvement of sequencer/commissioning procedures as evolving standard:
      - False-positive test procedure → modify/fix test sequence
      - False-negative tests → add missing test procedure
    - Proper heuristics → identify and provide a quantitative basis for facility upgrade decisisions

Sequences & Tasks I/III

- ‘Task’ = device class specific atomic test, e.g.
  - connectivity test, power ‘on’, power ‘off’, ...
  - actual vs. reference comparison, ...

Sequences & Tasks I/III

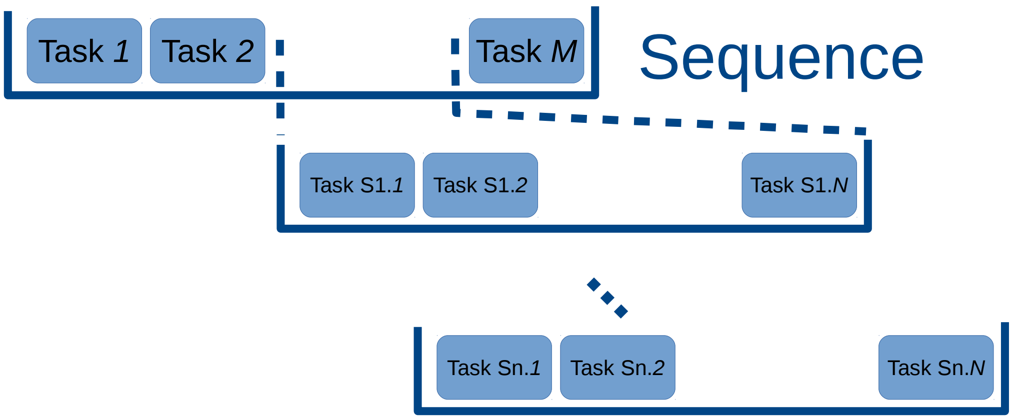
- ‘Task’ = device class specific atomic test, e.g.
  - connectivity test, power ‘on’, power ‘off’, ...
  - actual vs. reference comparison, ...
- ‘Tasks’ can be assembled to ‘Sequences’ …

...which may also contain further sub-sequences:

Sequences & Tasks I/III

- ‘Task’ = device class specific atomic test, e.g.
  - connectivity test, power ‘on’, power ‘off’, ...
  - actual vs. reference comparison, ...
- ‘Tasks’ can be assembled to ‘Sequences’ …

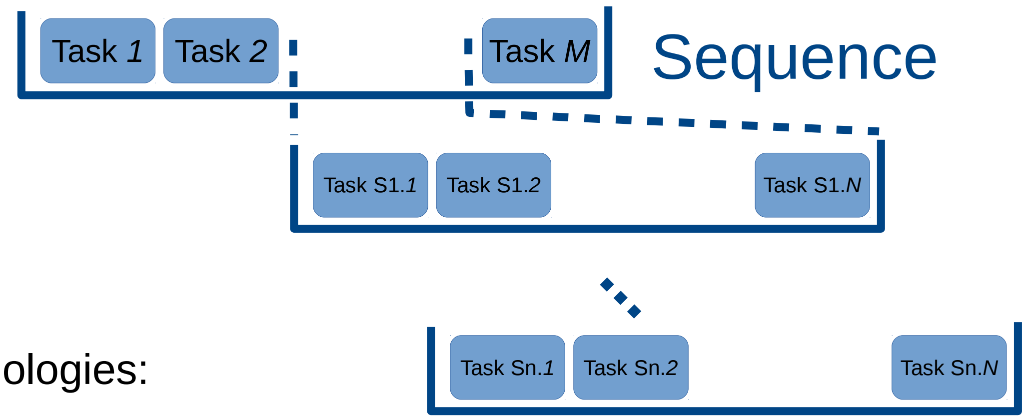
...which may also contain further sub-sequences:



Sequences & Tasks I/III

- 'Task' = device class specific atomic test, e.g.
  - connectivity test, power 'on', power 'off', ...
  - actual vs. reference comparison, ...
- 'Tasks' can be assembled to 'Sequences' …

  ...which may also contain further sub-sequences:



- CO backbone technologies:
  - FAIR Archiving Systems → Documentation
  - LSA-based Settings Management → Reference & Data Supply
  - System- and Site-wide Digitisation of Analog Signals → 'actual vs. reference' process monitoring

Sequences & Tasks II/III

- What is provided by the sequencer frame-work:

```
abstract class GenericHwcSequence {
    void exec() {
        initialize(); // communication to Archiving System, LSA, etc.
        specificPart();
        bookKeeping();
    }
}
```

- Level 1 & 2 tests (provided by the CO/equip. Group/machine experts):

```
class HwcSequence extends GenericHwcSequence {
    void initializeDeviceConnections();
    void specificPart() {
        super.specificPart();
        connectivityTest = initializeDeviceConnections(deviceName);
        if (connectivityTest.isHostReachable()) {  // example: basic connectivity tests
            connectivityTest.testNameserver();
            connectivityTest.testCMW3get();
            connectivityTest.testJAPCget();
            connectivityTest.testCMW3Subscribe();
            connectivityTest.testJAPCSubscribe();
        } else {
            // error reporting, etc.
        }
    }
}
```

Sequences & Tasks III/III

- what the user needs to implement

```
class HwcTest1 extends HwcSequence {
      void specificPart() {
          super.specificPart()
          task1();  // user/device-specific atomic test operation 1
          task2();  // user/device-specific atomic test operation 2
          [..]
      }

     void task1() {
          // test SAT-A sub-procedure x.1, see specification... item ...
          // [..]
     }

     void task2() {
          // test SAT-A sub-procedure x.2, see specification... item ...
          // [..]
     }
}
```

Sequences & Tasks III/III

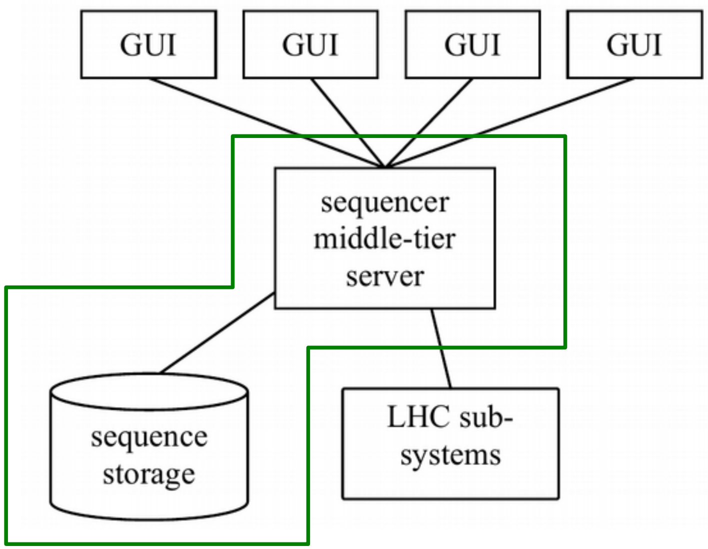
- what the user needs to implement

```
class HwcTest1 extends HwcSequence {
      void specificPart() {
          super.specificPart()
          task1();  // user/device-specific atomic test operation 1
          task2();  // user/device-specific atomic test operation 2
          [..]
      }

     void task1() {
          // test SAT-A sub-procedure x.1, see specification... item ...
          // [..]
     }

     void task2() {
          // test SAT-A sub-procedure x.2, see specification... item ...
          // [..]
     }
}
```

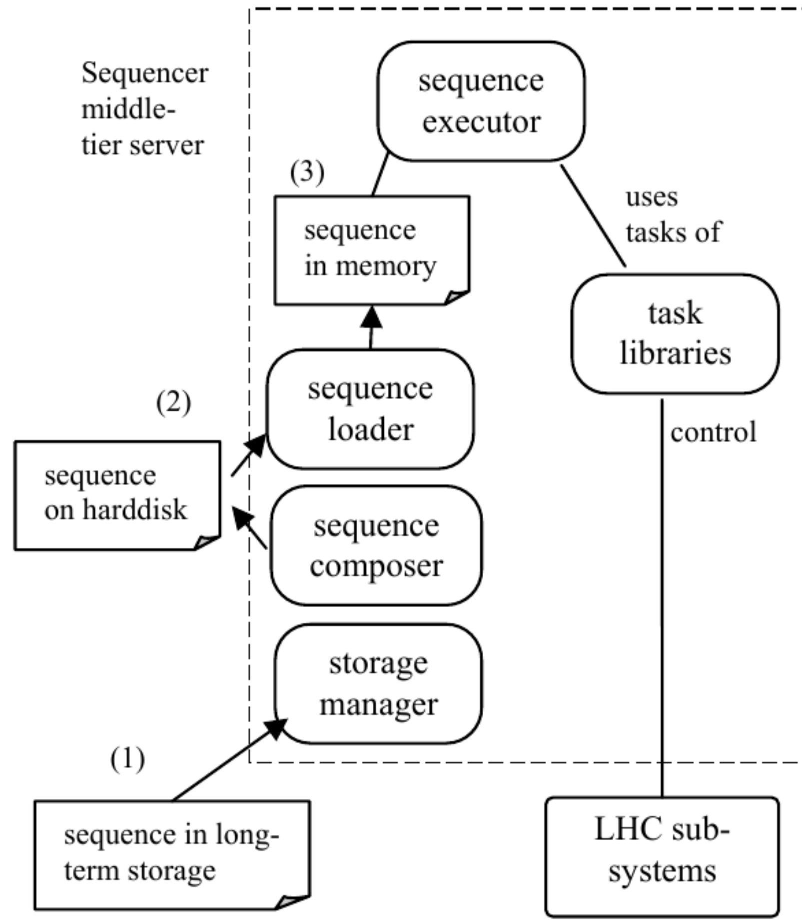
Some examples to get a flavour of the targeted code style and flavour:
https://www-acc.gsi.de/svn/applications/app-codesnippets/

LHC Sequencer Architecture

re-use for FAIR/re-commissioning in 2018



	HWC	BC & OP
Execution	Run, stop, break, skip	Run, stop, break, skip, jump
Error Handling	Fail and stop on error	Ignore, stop, run recovery sequence
State	int. variables	No variables
Control Statements	Loops, if/else, try/catch	
Typical parallelism	Sequencences in ||	Tasks in ||
Typical mode	run-through automatically	“debug” and run-through



courtesy Vito Baggiolini

Some Preliminary Test Hierarchy

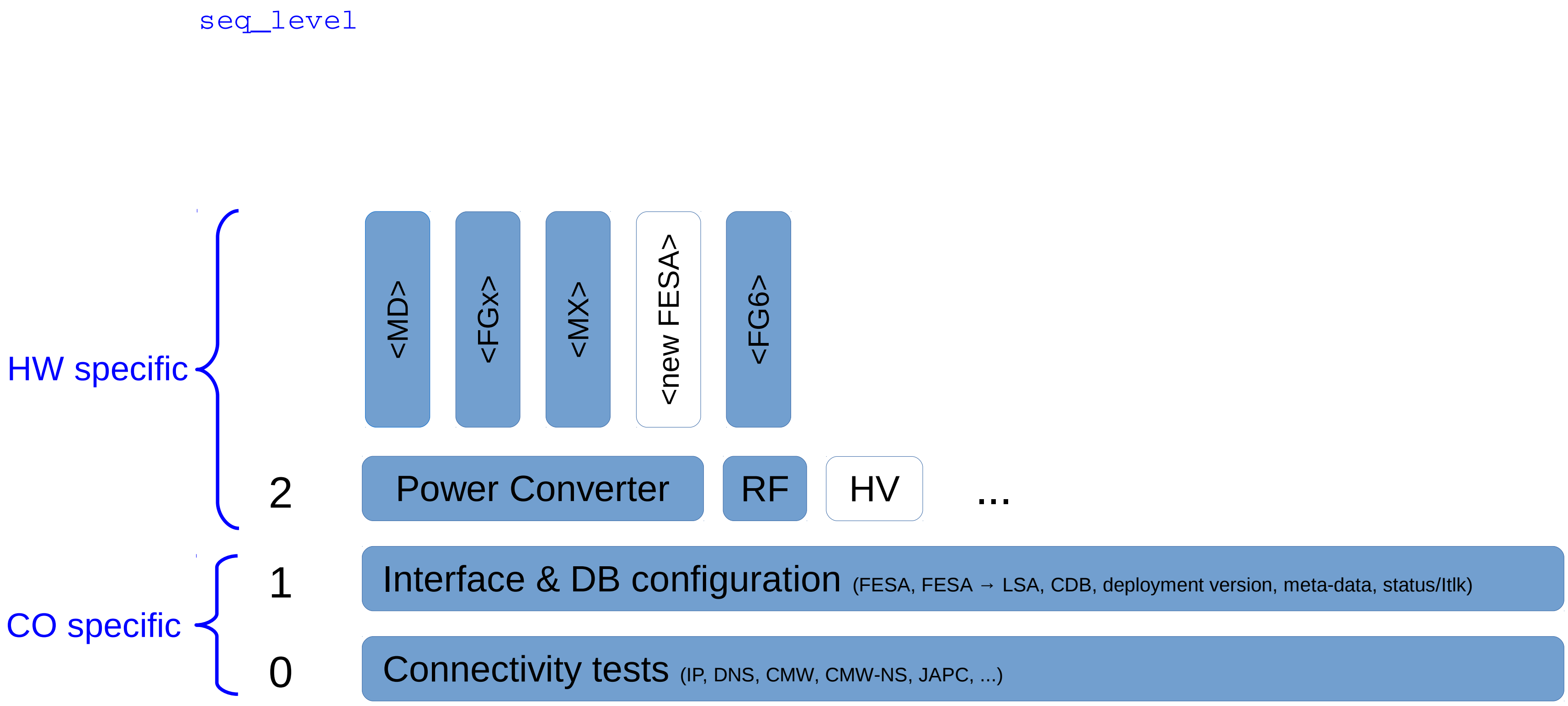
seq_level

CO specific { 1 Interface & DB configuration (FESA, FESA → LSA, CDB, deployment version, meta-data, status/Itlk)

0 Connectivity tests (IP, DNS, CMW, CMW-NS, JAPC, ...)

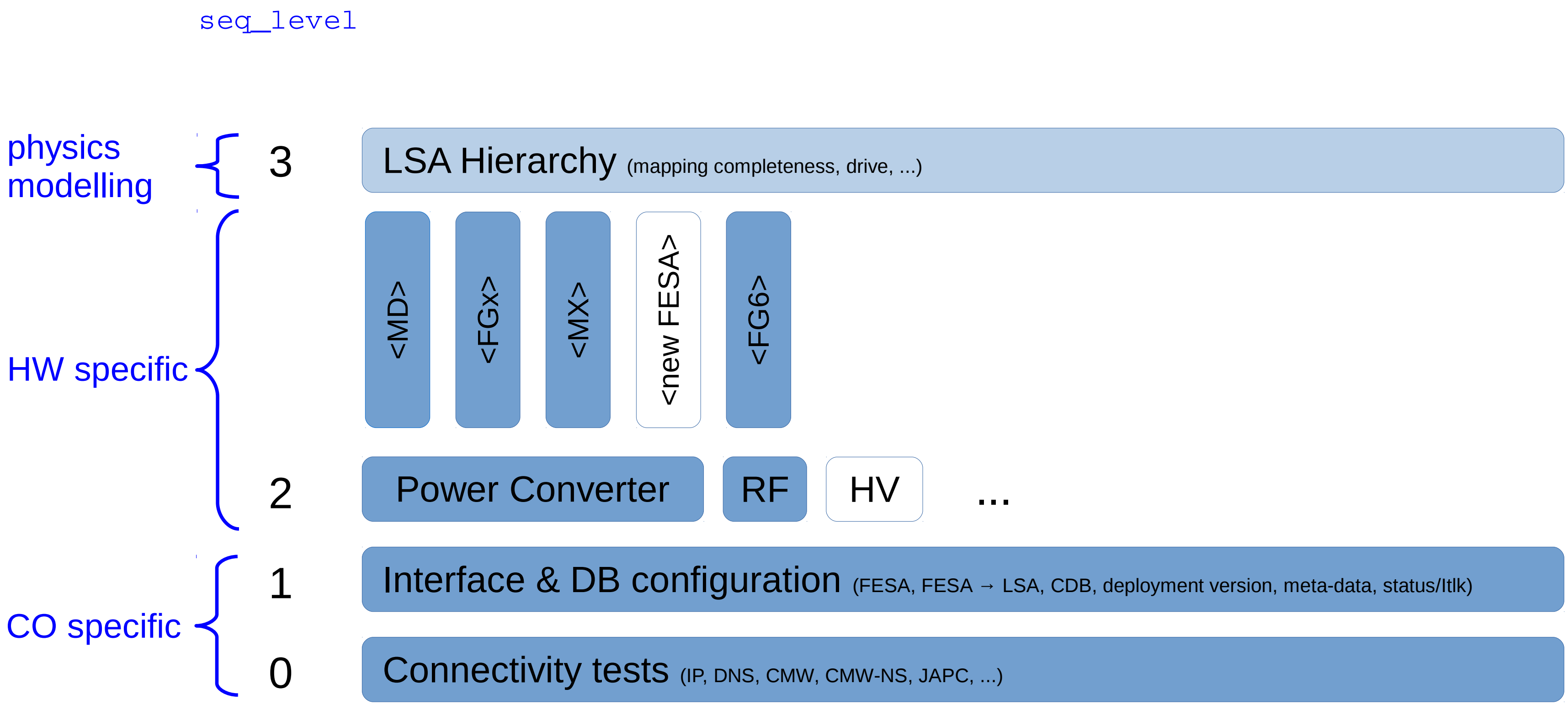
- Some logstash meta-data keys (see: https://logstash.acc.gsi.de/):
  - Existing tags: program: 'sequencer', user_name, pid, ...
  - seq_device: e.g. device name, LSA property name, global function
  - seq_level: <0 … 4>, seq_task: <task/class name>, seq_sequence: <collection of tasks> (???)
  - seq_testID: unique identifier for given sequencer run (↔ multi-user, parallelism)
  - seq_test_start: <time-stamp>

Some Preliminary Test Hierarchy



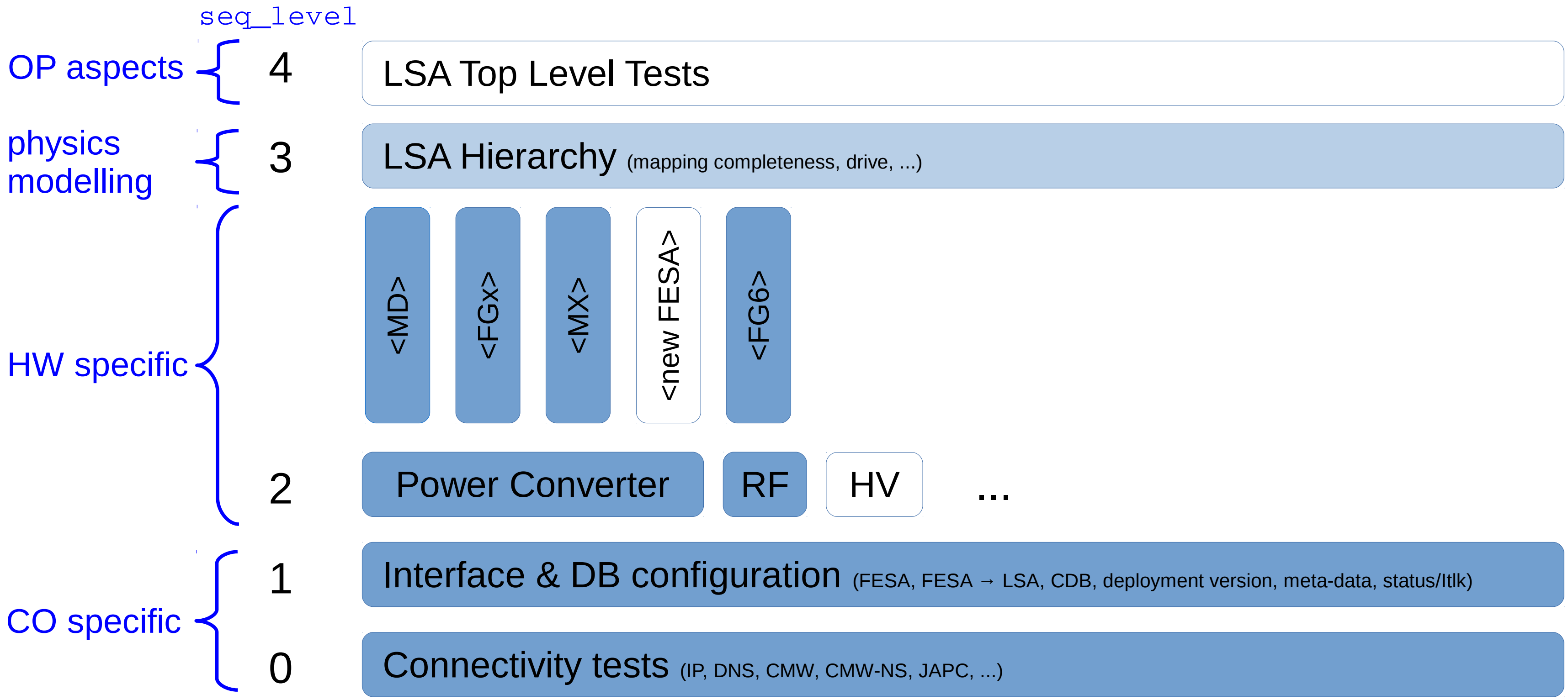
- Some logstash meta-data keys (see: https://logstash.acc.gsi.de/):
  - Existing tags: `program`: ‘sequencer’, `user_name`, `pid`, ...
  - `seq_device`: e.g. device name, LSA property name, global function
  - `seq_level: <0 … 4>, seq_task`: <task/class name>, `seq_sequence`: <collection of tasks> (???)
  - `seq_testID`: unique identifier for given sequencer run (↔ multi-user, parallelism)
  - seq_test_start: <time-stamp>

Some Preliminary Test Hierarchy



- Some logstash meta-data keys (see: https://logstash.acc.gsi.de/):
  - Existing tags: `program`: 'sequencer', `user_name`, `pid`, ...
  - `seq_device`: e.g. device name, LSA property name, global function
  - `seq_level: <0 … 4>`, `seq_task`: <task/class name>, `seq_sequence`: <collection of tasks> (???)
  - `seq_testID`: unique identifier for given sequencer run (↔ multi-user, parallelism)
  - seq_test_start: <time-stamp>

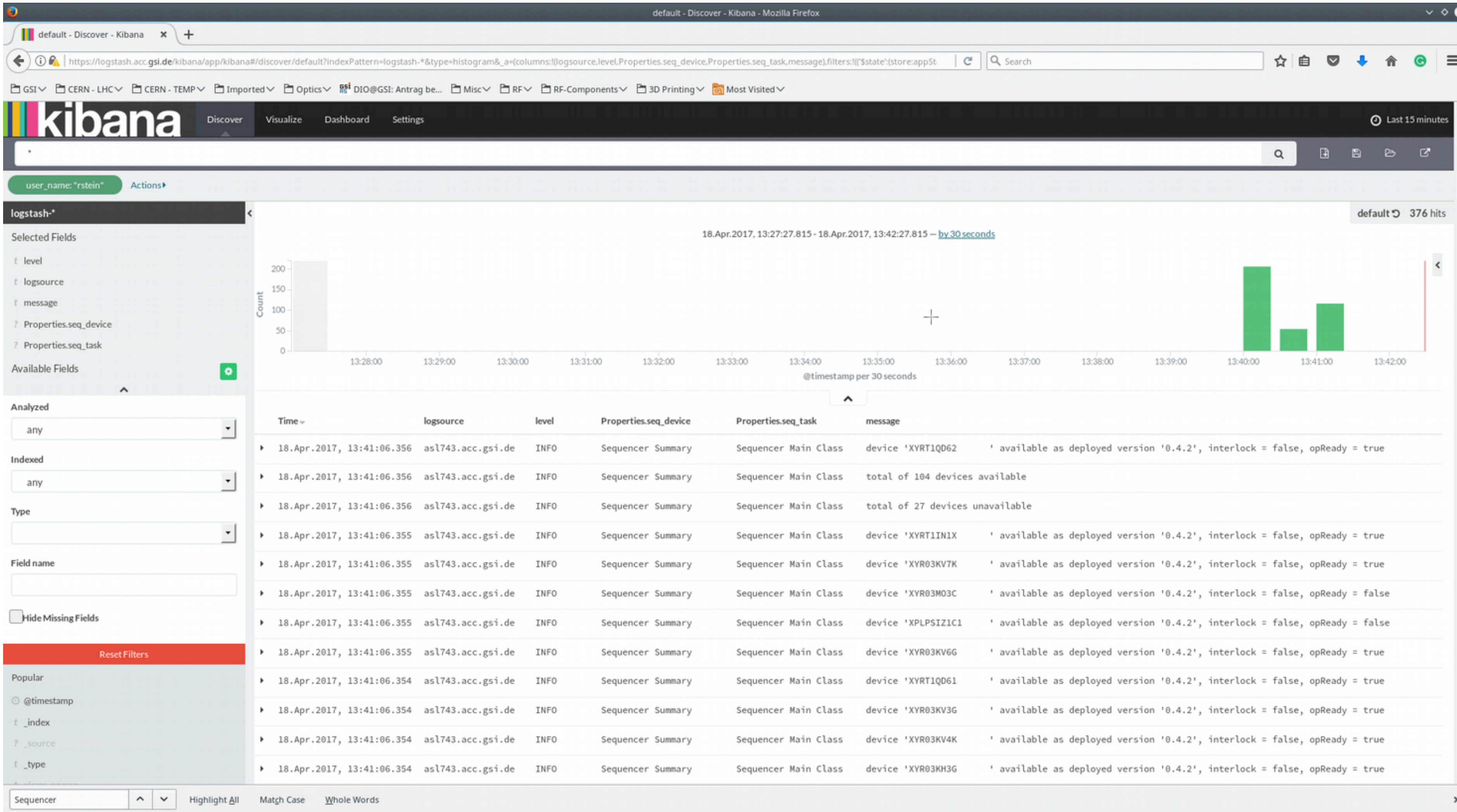
Some Preliminary Test Hierarchy



- Some logstash meta-data keys (see: https://logstash.acc.gsi.de/):
  - Existing tags: `program`: 'sequencer', `user_name`, `pid`, ...
  - `seq_device`: e.g. device name, LSA property name, global function
  - `seq_level: <0 … 4>, seq_task`: <task/class name>, `seq_sequence`: <collection of tasks> (???)
  - `seq_testID`: unique identifier for given sequencer run (↔ multi-user, parallelism)
  - seq_test_start: <time-stamp>

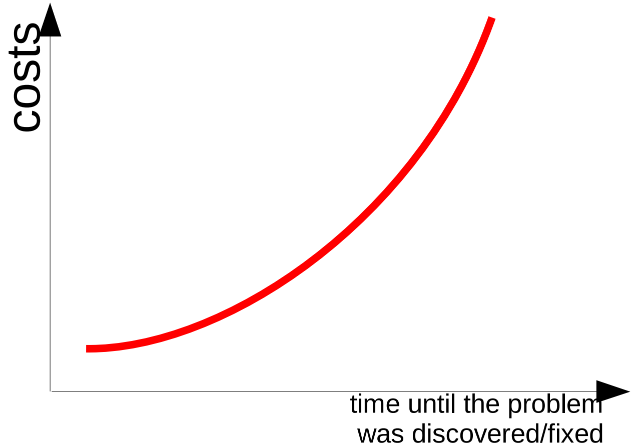
Sequencer Protocolling Example

https://logstash.acc.gsi.de/



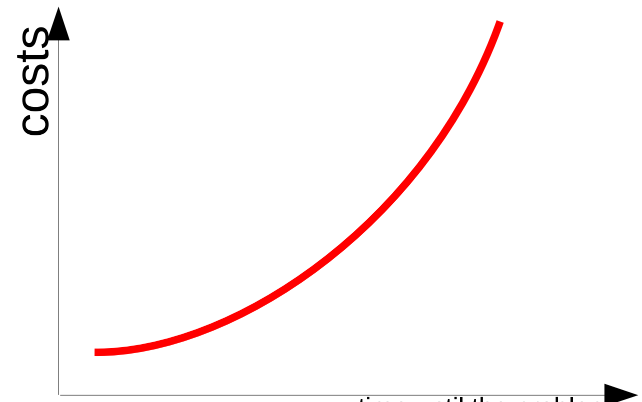
Sequencer Key Aspects in a Nut-Shell

1. Test automation → reproducibility, consistency, true parallelism and multi-tasking
2. Diagnostics: identification & localisation of faults
3. follow-up/handling of tests that can last over several hours → days (↔ SATs, UHV/RF/HV conditioning)
4. Machine protection (post-mortem): online validation of safety integrity level (SIL)
5. Machine availability tracking and optimisation:
   - Continuous improvement of sequencer & commissioning procedures as evolving standard:
     - False-positive test procedure → modify/fix test sequence
     - False-negative tests → add missing test procedure
   - Proper heuristics → identify and provide a quantitative basis for facility upgrade decisions



Sequencer Key Aspects in a Nut-Shell

1. Test automation → reproducibility, consistency, true parallelism and multi-tasking
2. Diagnostics: identification & localisation of faults
3. follow-up/handling of tests that can last over several hours → days (↔ SATs, UHV/RF/HV conditioning)
4. Machine protection (post-mortem): online validation of safety integrity level (SIL)
5. Machine availability tracking and optimisation:
   - Continuous improvement of sequencer & commissioning procedures as evolving standard:
     - False-positive test procedure → modify/fix test sequence
     - False-negative tests → add missing test procedure
   - Proper heuristics → identify and provide a quantitative basis for facility upgrade decisions



The sequence(r) is only as good as the procedures it implements
→ responsibility of every equipment group/owner and machine expert!

Main Aims for FAIR
Prototyping at SIS18 Recommissioning in 2018

- Sequencer – range of functionality:
  a) protocolling of executed tests → aim at 100% test coverage
  b) (semi-)automated test sequences ('JUnit-style' HW Tests)
  c) user-driven execution and configuration of test sequences (by non-Java equipment experts)

- next steps:
  - review/collect additional functional requirements
  - support test procedures together with equipment groups
    - main responsibility remains with equipment experts (EPC, CO, BI, …)
    - priorities: 1. EPC, 2. HV (Septa & Kicker), 3. Ring-HF (rational: large quantity, (fairly) low complexity).
    - other equipment test-procedures (besides connectivity tests) require additional man-power (CO, vacuum, BI, ...).
  - support/drive Sequencer development
    - initial proof-of-concept for Dry-Run #1 covering:
      - 'a)' protocolling: initially file-/logstash-based → Archiving System
      - 'b)' using simple Java based sequences executed via Eclipse (Java-expert only)
    - extend to covering also 'c)' requirements by Q1-2017 (on a 'best effort' basis)
      - initial aim: simple non-configurable GUI that can execute pre-defined test-sequences by non-Java/Eclipse-affine equipment experts
  - Follow-up of system- and machine commissioning procedures (with & w/o beams)
    → prerequisite for any sustainable system integration and accelerator facility operation

Thanks: a word from our sponsor

"Ask not what FAIR can do for you, ask which Commissioning Procedure you can help prepare for FAIR!"

Yes, we/you can!

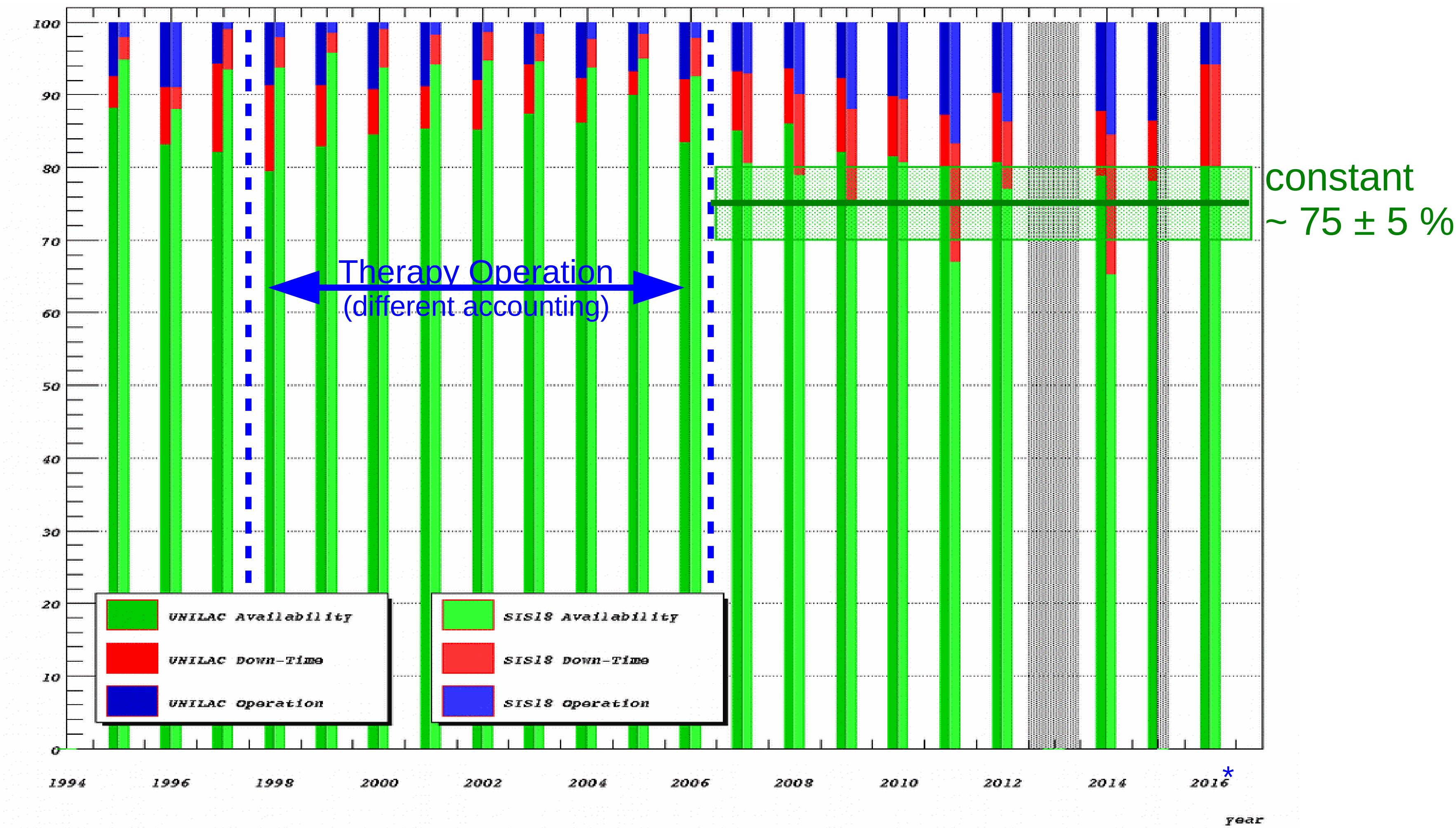
LHC Sequencer References

- M.Lamont et al., "Functional specification 'LHC Sequencer ...'", LHC-CQ-ES-0001, EDMS #810407, 2006-12-21
https://edms.cern.ch/ui/file/810407/0.6/LHC-CQ-ES-0001-00-60.pdf
- V. Baggiolini et al., "A Sequencer for the LHC ERA", CERN-ATS-2009-114, ICALEPS'2009, Kobe, Japan, 2009
http://cds.cern.ch/record/1215886/files/CERN-ATS-2009-114.pdf
- R. Alemany-Fernandez et al., "The LHC Sequencer", ICALEPS'2011, Grenoble, France, 2011 http://accelconf.web.cern.ch/AccelConf/icalepcs2011/papers/mopmn027.pdf
- V. Baggiolini, R. Alemany-Fernandez et al., "LHC Sequencer", extended LTC Workshop, Chamonix, France, 2008
http://indico.cern.ch/event/28066/contributions/638169/attachments/.../LHC_Sequencer.pdf
- D. Anderson et al., "The AccTesting Framework: ... for Accelerator Commissioning and Systematic Testing", ICALEPCS2013, San Francisco, USA, 2013
http://accelconf.web.cern.ch/AccelConf/ICALEPCS2013/papers/thppc078.pdf

Appendix

Accelerator Experience & Efficiency

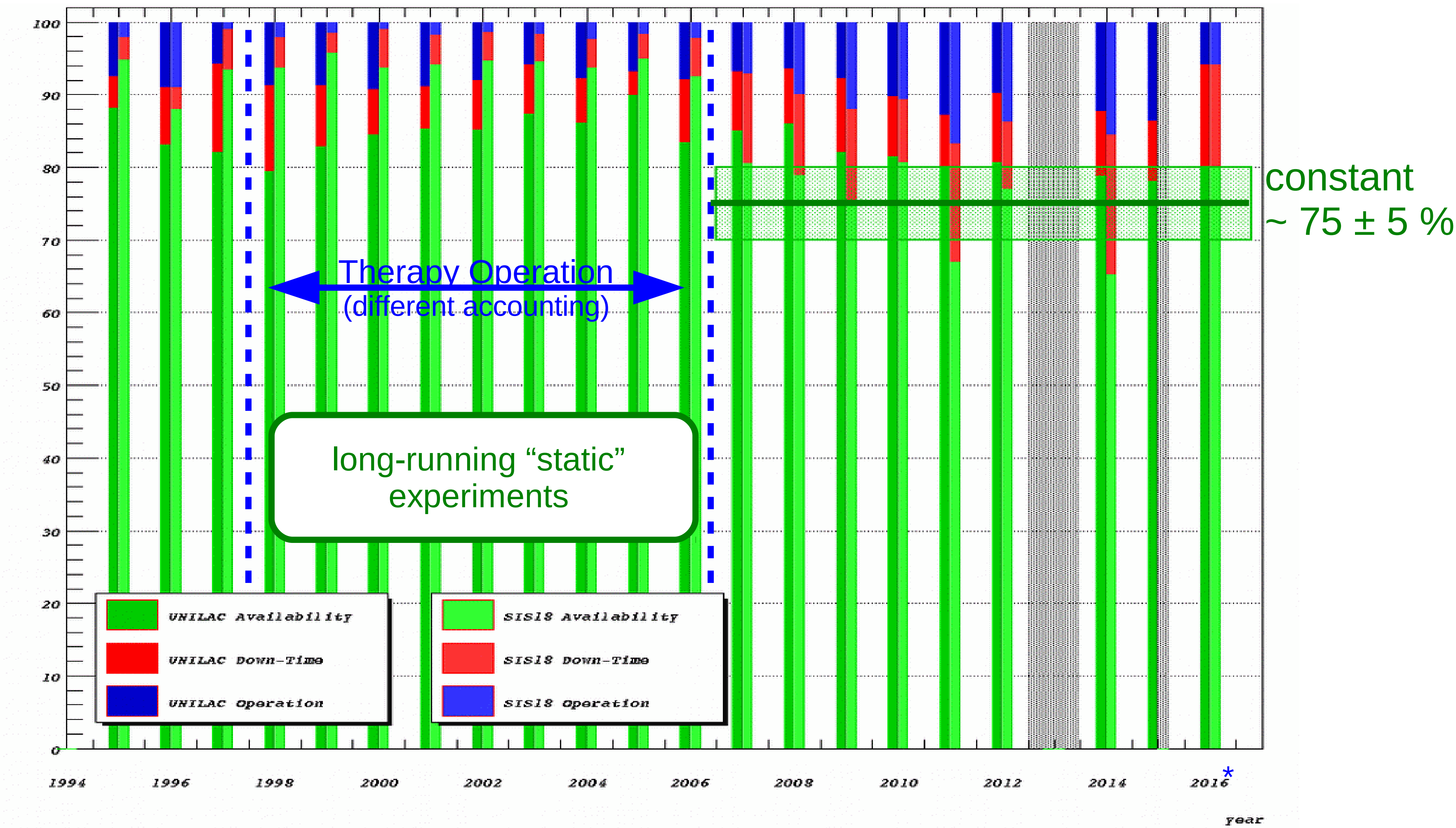
1995-2016: U. Scheeler, S. Reimann, P. Schütt et al.



Based on: U. Scheeler, S. Reimann, P. Schütt et al., "Accelerator Operation Report", GSI Annual Scientific Reports 1992 – 2015 + 2016 (D. Severin)
https://www.gsi.de/en/work/research/library_documentation/gsi_scientific_reports.htm
N.B. ion source exchanges are factored out from UNILAC & SIS18 data (~ constant overhead)
Availability: *experiments + detector tests + machine development + beam to down-stream accelerators;*
Down-time: *unscheduled down-time + standby;* Operation: *accelerator setup + re-tuning*

* 2018 operation limitations:
- only ½ UNILAC (w/o A3 & A4)
- only 1 element in SIS18

Accelerator Experience & Efficiency
1995-2016: U. Scheeler, S. Reimann, P. Schütt et al.

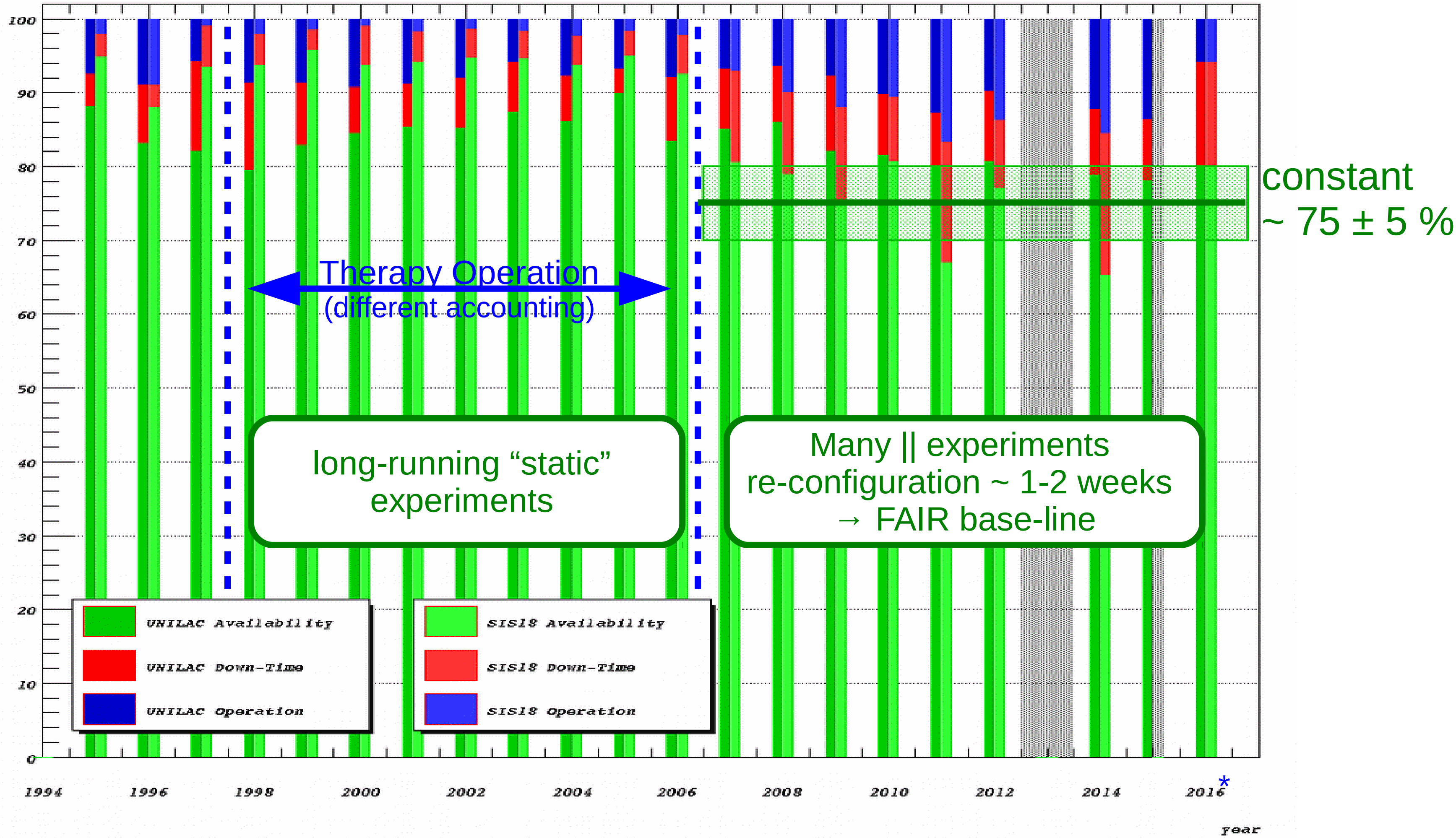


Based on: U. Scheeler, S. Reimann, P. Schütt et al., “Accelerator Operation Report”, GSI Annual Scientific Reports 1992 – 2015 + 2016 (D. Severin)
https://www.gsi.de/en/work/research/library_documentation/gsi_scientific_reports.htm
N.B. ion source exchanges are factored out from UNILAC & SIS18 data (~ constant overhead)
Availability: *experiments + detector tests + machine development + beam to down-stream accelerators;*
Down-time: *unscheduled down-time + standby;* Operation: *accelerator setup + re-tuning*

* 2018 operation limitations:
- only ½ UNILAC (w/o A3 & A4)
- only 1 element in SIS18

Accelerator Experience & Efficiency

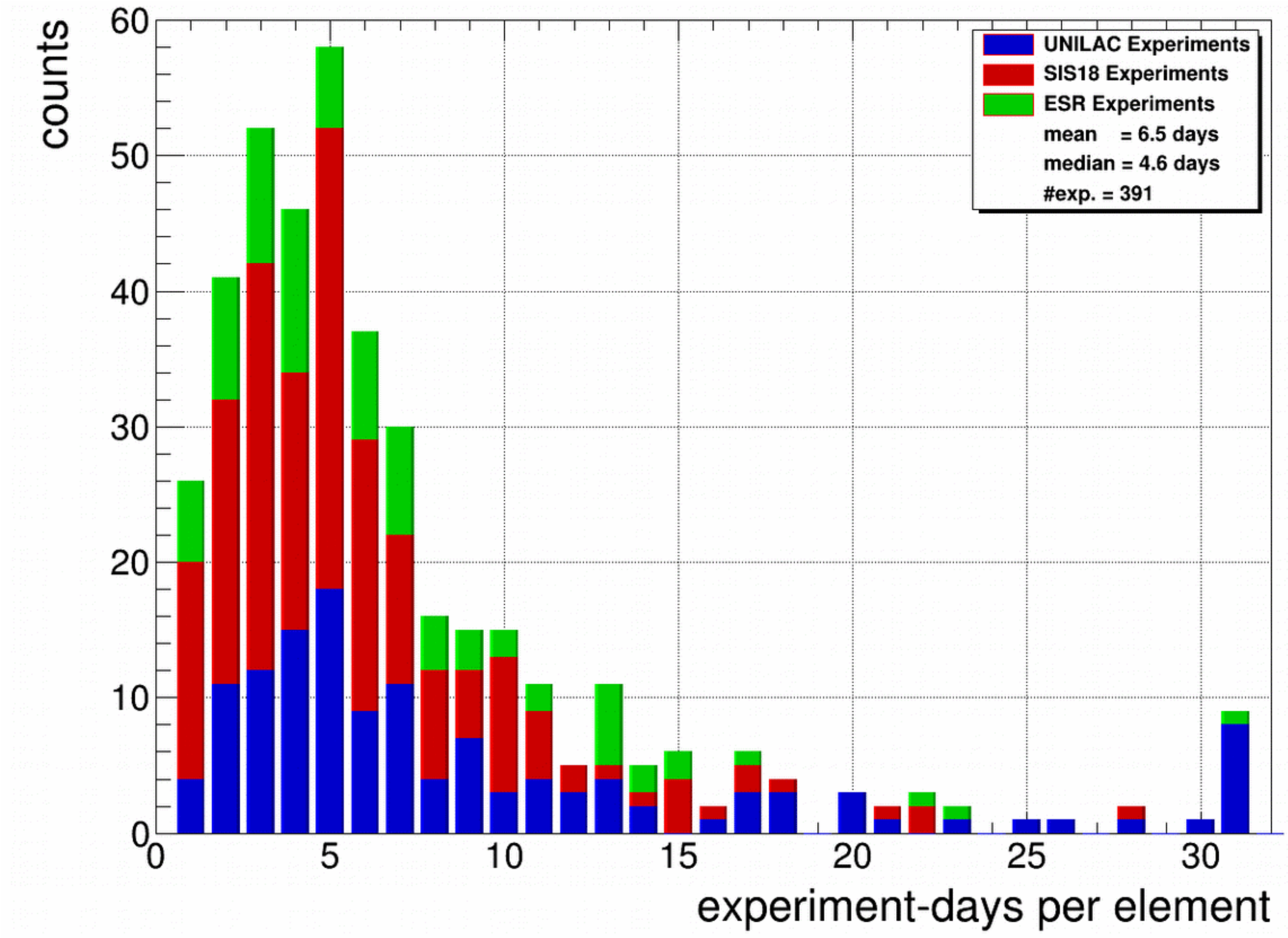
1995-2016: U. Scheeler, S. Reimann, P. Schütt et al.



Based on: U. Scheeler, S. Reimann, P. Schütt et al., "Accelerator Operation Report", GSI Annual Scientific Reports 1992 – 2015 + 2016 (D. Severin)
https://www.gsi.de/en/work/research/library_documentation/gsi_scientific_reports.htm
N.B. ion source exchanges are factored out from UNILAC & SIS18 data (~ constant overhead)
Availability: *experiments + detector tests + machine development + beam to down-stream accelerators;*
Down-time: *unscheduled down-time + standby;* Operation: *accelerator setup + re-tuning*

* 2018 operation limitations:
- only ½ UNILAC (w/o A3 & A4)
- only 1 element in SIS18

Accelerator Experience & Efficiency
a closer look on Exp. Statistics 2006-2016



Open Questions

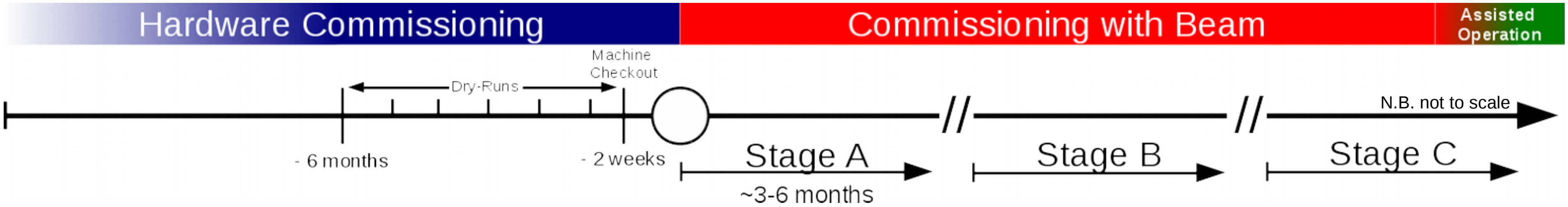
- HWC and BC sequencer are still different implementations? Specs?
- PNuts: still considered useful? Why not plain compiled java?
- Oracle database vs. svn. Pros/Cons?
- Sequence editing? How? Expert Level?
- Representation of sequences (high-level, low-level)? RMI usage?
- Result reporting: via DB? GUI interaction?
- Parallel execution of sequences (mutual blocking for same device, OK for different device). config of sequence/task by device?
- User level parameter & sequence modification (FAIR: e.g. user-level defined mini-ramp parameterisation, sequence(device name/group))
- Why sequence definition in oracle DB? SVN-stored sequences not sufficient?
- Who's editing the sequences routinely? Java-expertise needed as prerequisite?
- Commissioning reporting/error isolation functionality: How? How much? How much DB interaction? (see with Markus).

FC²WG Control Topics – more than "Control System" & Data Supply

- Facility & Interface Analysis
  - Procedures: HWC, HWC-'Machine Check Out', BeamCommissioning, BC-Stage A (pilot beams), BC-Stage B (intensity ramp-up), BC-Stage C (nominal/production operation) Beam parameters, FAIR performance model, optimisation, Accelerator & Beam Modes
- Beam Instrumentation & Diagnostics – System Integration
  - Intensity (DCCTs) & beam loss (BLMs) → Beam Transmission Monitoring System (BTM), trajectory & orbit (BPMs), Q/Q', optics (LOCO & phase-advance), longitudinal & transverse emittance (FCTs. WCM, screens, IPM, etc.), Δp/p, long. bunch shape (FCTs, Tomography), abort gap monitoring, ...
- Accelerator Hardware – System Integration
  - Power converter, magnets, magnet model, RF, injection/extraction kicker, tune kicker/AC-dipole, beam dump, collimation/absorbers, cryogenics, vacuum, radiation monitoring, k-modulation, machine-experiment interfaces
- Control System
  - Archiving system, analog signal acquisition, test-beds, timing, bunch-to-bucket transfer, cyber security, role-based-access, middleware, RT & Feedbacks, daemons
- Components
  - post-mortem, management of critical settings (safe-beam settings), machine protection, interlocks, beam quality checks, daemons, 'Page One', aperture model, ...
- Applications
  - Sequencer (semi-automated procedures), fixed-displays, ...
  - Beam-Based Applications, Cycle-to-Cycle Feedbacks & GUIs → second talk

Commissioning with Beam

https://fair-wiki.gsi.de/FC2WG/BeamCommissioning



- Split Beam Commissioning into three stages:

A) Pilot beams/"easily available" ions (e.g. U28+, Ar)

  - basic checks: threading, injection, capture, cool, convert, acceleration/decelerate, stripping & extraction
  - always done with 'safe' ie. low-intensity/brightness beam
    - Ions: simpler optics, beam dynamics → Protons: transition crossing

B) Intensity ramp-up & special systems

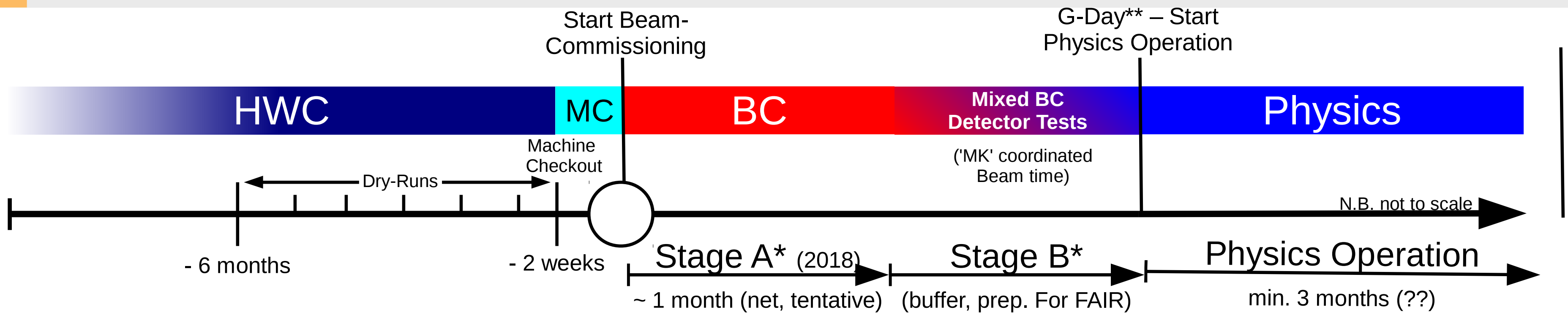
  - achieving and maintaining of nominal transmission and beam losses
  - commissioning of e.g. e-cooler, slow extraction, transverse fast feedbacks
  - commissioning and validation of machine protection & interlock systems
  - Possibly unsafe operations always preceded by checks with safe beam

C) Production operation with nominal intensities
(N.B. first time counted as 'commissioning' or 'assisted operation' → later: 'regular operation')

  - push physics and beam parameter performance (emittance, momentum spread, ...)
  - identify and improve upon bottlenecks impacting FAIR's 'figure-of-merit
  - make fast setup and switch-over between different beam production chains routine

Recommissioning & Operation in 2018 I/II

https://fair-wiki.gsi.de/FC2WG/HardwareCommissioning/



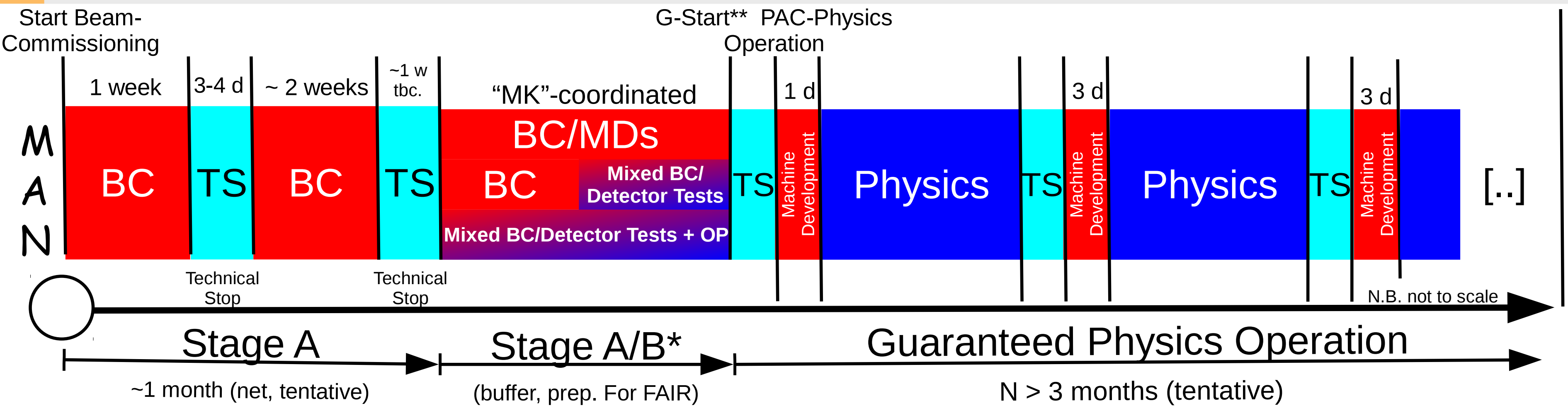
- Hardware Commissioning → coordinated by Sub-Project-Leaders & Machine Coordinators
  - link-existing facilities (GAF), upgrades, machine re-alignment, “SATs”, HW systems (equip. groups),
- Dry-Runs – for all machines post (possibly also UNILAC), each two days, fixed dates (↔ experts availability), starting:
  - Dry-Run #1 – 17.10.2017: CO-core: LSA, Timing System, Archiving System, SCUs, CO core application, …
  - Dry-Run #2 – 14.11.2017: as before + tbd.
  - Dry-Run #3 – 12.12.2017: as before + tbd.
  - Dry-Run #4 – 09.01.2018: as before + BI + related applications
  - Dry-Run #5 – 06.02.2018: as before + Experiments (proposal) + Machine-Experiment Interfaces
  - Dry-Run #6 – 20.02.2018: as before + AEG + “last-minute” checks
  - Dry-Run #7 – 06.03.2018: buffer
- Machine-Checkout – intensive “last minute checks” (N.B closed tunnel/machine):
  - UNILAC: -1 month → BC- 'day 0'
    - patrols, heat runs: RF & power supply conditioning, ...
  - SIS, ESR, CRY: -3 weeks → BC- 'day 0'
    - patrols, heat runs: RF & power supply/AEG conditioning, safety systems: personnel safety, access system, legal ZKS & RP checks (§66 Abs. 2 StrlSchV), “very last-minute” checks/bug fixes: vacuum, power, BI, CO, ...

* in 2018: light-version w.r.t. commissioning of new machines

** “guaranteed” start physics operation (Plan A), no hick-ups, sacrificial buffer being activities related to 'Stage B'

Recommissioning & Operation in 2018 II/II

https://fair-wiki.gsi.de/FC2WG/BeamCommissioning



- Stage-A: Initial Beam Commissioning (BC): 2 dedicated 3 week@24h/7 BC blocks, main aim:
  - drive beam expeditiously through the Beam Production Chain: sources → synchrotrons & beam transfers → exp. targets & storage rings
  - check basic 'accelerator mechanics': threading, injection, capture, cool, convert, acceleration/decelerate, stripping & extraction
  - identify beam-related limitations: polarities, RF, beam instrumentation, machine alignment, effective physical machine aperture, ...
- Immediately followed by dedicated, scheduled Technical Stop (TS)
  - needed for follow-up of HW (tunnel) and SW issues (CO, …)
- Stage-A/B*: Mixed-BC, Machine-Development, Detector Tests (aka. "splash events" for experiments) & Operator Training
  - N.B. "old machine" but completely new CO, substantial modifications
- Physics operation: as promised/targeted nett 3 months (to be confirmed), grouped into 2-3 blocks interleaved with
  - TS: routine maintenance → increases overall availability, follow-up of OP/CO/equipment issues + major ion species/source changes
  - MDs: follow-up of beam physics issues, CO improvements (e.g. beam-based FBs), improve facility to reach nominal FAIR parameters
    - N.B. also better for guaranteeing smooth restart/picking-up of physics operation after technical stops (experts availability)