Methodical Accelerator Design
'Next Generation' Check Point
Informal Review

Laurent Deniau
CERN-BE/ABP

20th January 2021

Laurent Deniau, CERN BE/ABP, 1211 Geneva 23, laurent.deniau@cern.ch

MAD-NG check point and next steps

- **Status**
  - **Long term design:** easy to use and extend (better design and architecture). Development started in 2016 **(=2 FTE!** See achievement below**)**.
  - **Fast, simple, general scripting language ➡ All code is in scripts (100% of the physics).**
  - **Efficient JIT and FFI** ➡ No difference between User scripting and Application code. User can write new physics, customise or parametrise existing physics at full speed!
  - Flexible technologies ➡ portable, self-contained (embedded), all-in-one and modular.
  - **Support for GTPSA (Differential Algebra), DA Maps and User-defined Knobs (see online demo).**
  - **Ahead of MAD-X in many aspects:** Sequences, Tables (TFS), Survey, Track, CoFind, Twiss, Match, Plot, Beam (charge), Backtracking, (Perm.) Misalignments, Fringe (PTC), Orbit correction, Patches, Combined functions, All-thick elements +sub-elements, Elements tracking structure (layout, slicing, frames, user-actions, integrators), Dynamic polymorphic code, Differential algebra (ℝ & ℂ), Linear algebra (ℝ & ℂ), FFT.

Tested on Linac4, PSB, PS, SPS, LHC, HL-LHC*, FCC-ee, CLIC.*

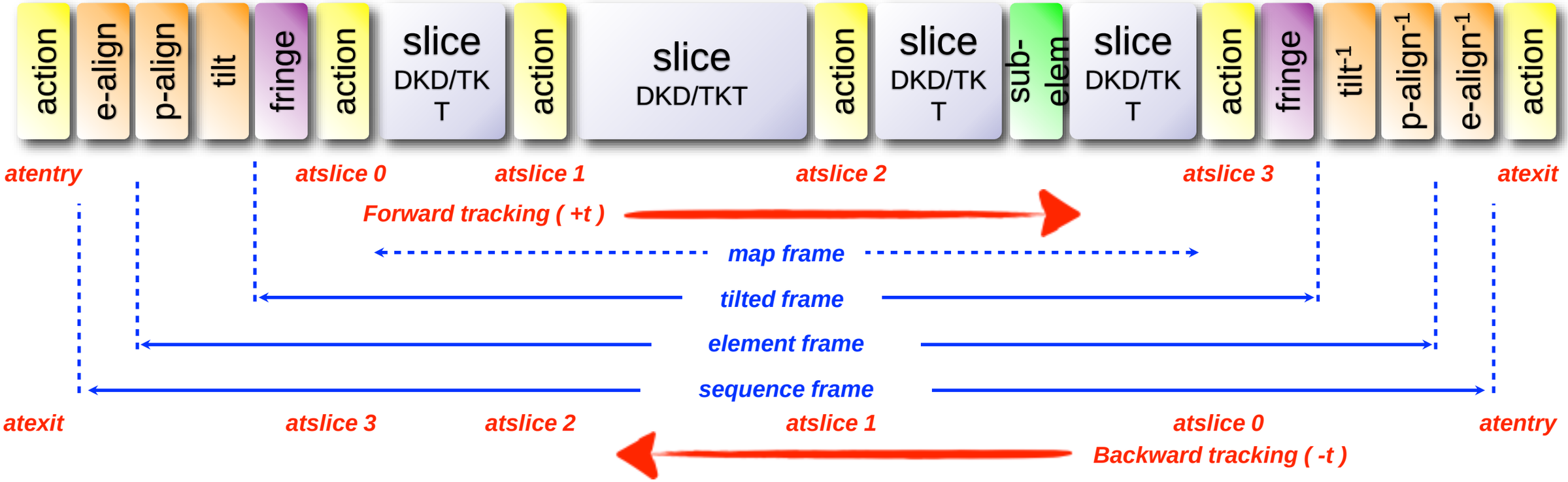
- **Short term (< 2y)**
  - Complete in-depth **review of the maps and alternatives with experts** (+unit tests).
  - Missing modules: Aperture n1 (table post-proc.), IBS (table post-proc.), Emit (matching).
  - Missing physics: **Parametric non-linear normal forms** (+RDT extraction), **Spin** (FCC-ee).
  - Missing techs: parallel tracking, cpymad-like MAD-X interface.

AND continue to: write the manual & keep it up-to-date + thousands of unit tests

- **Mid term (< 4y) — Improvements for speed**
  - **Parametric normal form ➡ x2 chroma, xN vars matching.**
  - **Improve maps for time=true (physics) ➡ improve convergence vs PTC**, x2-3 speed improvement.
  - Improve maps for damap (in C) ➡ reduce GTPSA memory allocations, x2-3 speed improvement.
  - Improve integrators & data structures (in C) ➡ better modern CPU use, x2 speed improvement.
  - **Fast parallel tracking (static over N turns)** ➡ JIT generated C code + previous C improvements.
  - Switch to the alternate object model (no lookup) + tune JIT parameters ➡ x2-3 speed improvement.
- **Long term (> 4y)**
  - **Alleviate dependency on the JIT ➡ push max code to C ➡ loss of features/flexibility/readability!!**

Laurent Deniau, CERN BE/ABP, 1211 Geneva 23, laurent.deniau@cern.ch

MAD-NG - examples



[Track element method code example]

[Track through element code example]

[Track integrator code example]

[Track map code example]

Cofind = Track damap @ order 1 + matching

Twiss = Track damap NF @ order n + actions *(chained to other actions)*

Knobs = damap-w-knobs + Track damap @ order n (+ actions)

[Backtrack FODO lattice example]

[Track FODO lattice with knobs example]

[Match LHC (un)coupling with knobs example] ([pre-installation])

Conclusion
What matters is the quality of the maps and their analysis, and the technology will do the rest for you.

Laurent Deniau, CERN BE/ABP, 1211 Geneva 23, laurent.deniau@cern.ch

Backup slides

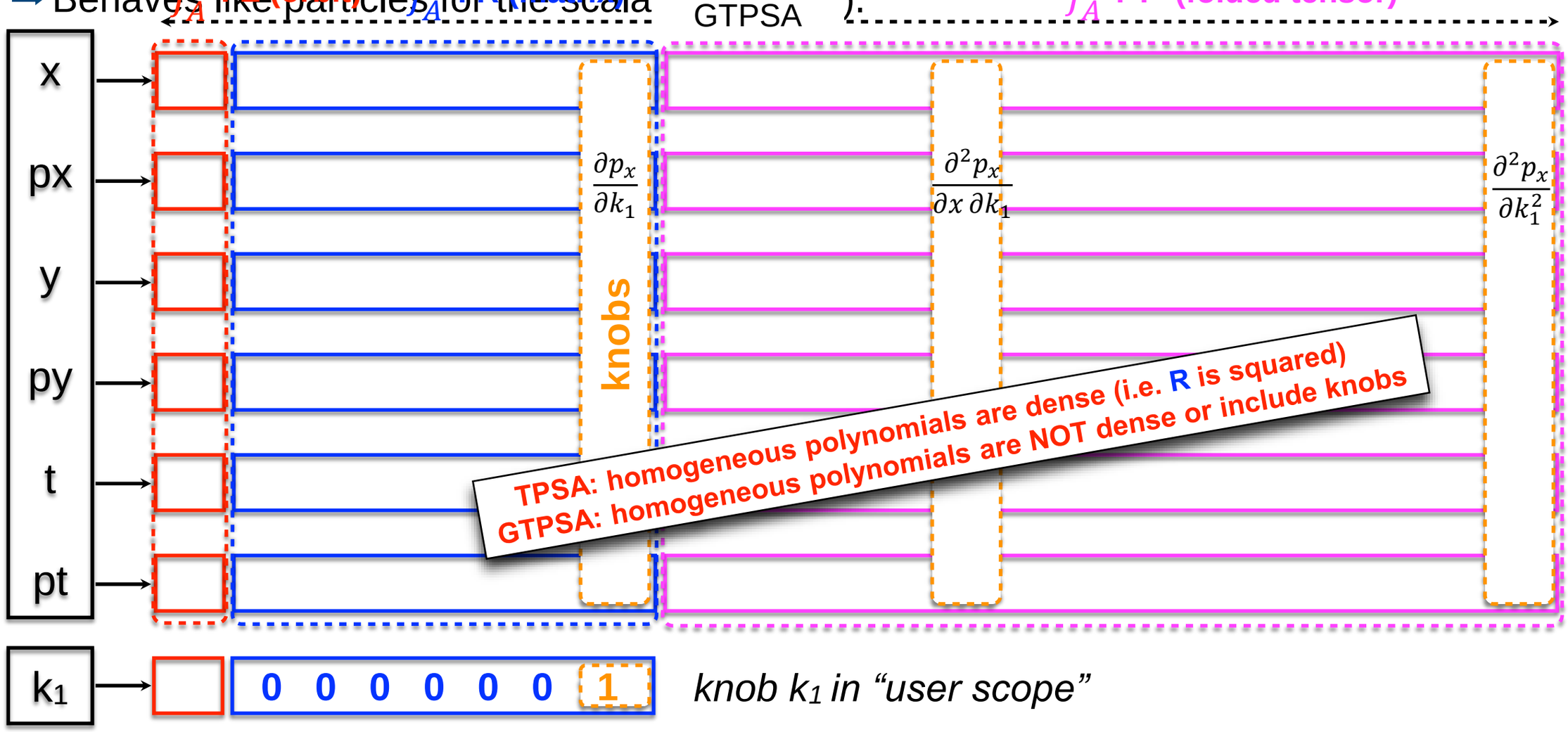
Laurent Deniau, CERN BE/ABP, 1211 Geneva 23, laurent.deniau@cern.ch

DA map

- Differential Algebra maps
  - ➞ Tuple of GTPSA, e.g. 6D phase space uses 6 GTPSA.
  - ➞ Handles user defined knobs.

DA map of 6 variables at order 2 (e.g. MAD-X twiss)

- ➞ Behaves like particles for the scalar part (orbit).



Laurent Deniau, CERN BE/ABP, 1211 Geneva 23, laurent.deniau@cern.ch

DAmap vs. Matrix sizes

TPSA: homogeneous polynomials are dense with $\binom{n+v}{v} = \frac{(n+v)!}{n!\,v!}$ coefficients

GTPSA: homogeneous polynomials are NOT dense (no direct formula, only upper bound)

v \ n	1	2	3	4	5	6	7	8	9	10	11	12
1	2	3	4	5	6	7	8	9	10	11	12	13
2	6	12	20	30	42	56	72	90	110	132	156	182
3	12	30	60	105	168	252	360	495	660	858	1092	1365
4	20	60	140	280	504	840	1320	1980	2860	4004	5460	7280
5	30	105	280	630	1260	2310	3960	6435	10010	15015	21840	30940
6	42	**168**	504	1260	2772	5544	10296	**18018**	30030	48048	74256	111384
7	56	252	840	2310	5544	12012	24024	45045	80080	136136	222768	352716
8	72	360	1320	3960	10296	24024	51480	102960	194480	350064	604656	1007760

DA map: $v\binom{n+v}{v}$

TPSA manipulate only numbers!

TPSA are the only suitable solutions for high orders!

Matrix: $\sum_{k=0}^{n} v^{k+1} = \frac{v(v^{n+1}-1)}{v-1}$

v \ n	1	2	3	4	5	6	7	8	9	10	11	12
1	2	3	4	5	6	7	8	9	10	11	12	13
2	6	14	30	62	126	254	510	1022	2046	4094	8190	16382
3	12	39	120	363	1092	3279	9840	29523	88572	265719	797160	2391483
4	20	84	340	1364	5460	21844	87380	349524	1398100	5592404	22369620	89478484
5	30	155	780	3905	19530	97655	488280	2441405	12207030	61035155	305175780	1525878905
6	42	**258**	1554	9330	55986	335922	2015538	**12093234**	72559410	435356466	2612138802	15672832818
7	56	399	2800	19607	137256	960799	6725600	47079207	329554456	2306881199	16148168400	113037178807
8	72	584	4680	37448	299592	2396744	19173960	153391688	1227133512	9817068104	78536544840	628292358728

Laurent Deniau, CERN BE/ABP, 1211 Geneva 23, laurent.deniau@cern.ch