◉ **Status**
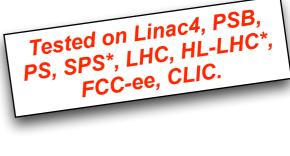
➡ **Long term design:** easy to use and extend (better design and architecture). Development started in 2016 **(=2 FTE!** See achievement below**)**.

➡ **Fast, simple, general scripting language** ⇒ **All code is in scripts.**

➡ **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! User study scripts run blindly fast! Needs much less parameters, replaced by functions & lambda!

➡ Flexible technologies ⇒ portable, self-contained (embedded), all-in-one and modular.

➡ **Support for GTPSA (Differential Algebra), DA Maps and User-defined Knobs** (unique?).

➡ **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 ($\mathbb{R}$ & $\mathbb{C}$), Linear algebra ($\mathbb{R}$ & $\mathbb{C}$), FFT.

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

◉ **Missing Features**

➡ Missing modules: Aperture (need clear general specs for n1), Emit, IBS.

➡ Missing physics: **Tapering & Spin** (FCC-ee), **Parametric non-linear normal forms** (+RDT +stroboscopic NF 6D).

➡ Missing elements: BeamBeam, NonLinELens, CFMultipole. (**added by 2020**)

➡ Missing techs: parallel tracking, cpymad-like MAD-X interface.

◉ **Improvements (mainly 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.

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