Improvements to the Tracks Library

1. Introduction

The new track system, introduced in EEP16, has not yet reached the expected user acceptance. The current plan is to improve it in EEP version 18. It shall be more convenient and thus, shall encourage the users who stayed at EEP15 to move on to a new EEP version.

The development of such improvements shall be based on the experience and needs of the users. I have been asked by Rudi Fey, if I could propose some helpful changes to the tracks library in a short document. And here is the result.

I am writing the document in English for an easier exchange of ideas with all parties. But, as I am used to the German version of EEP and not always sure, what the appropriate English terms for some German words are, you will find here a list of terms and their meanings used in this document:

English	German	Remark
Clothoid	Klothoide	Also known as Euler spiral or Cornu's spiral
Inclination	Neigung	Sideways gradient. This is not the slope ("Steigung")!
Lateral copy	Parallel vervielfältigen	Copies of one or more splines sideways with a given distance
Polygon course	Polygonzug	A spline type
Slope	Steigung	Forward gradient. This is not the inclination ("Neigung")!
Track or Spline	Gleisstück	However, when the spline type "Spline" is meant, it is set in quotation marks.
Track relative	Gleisrelativ, Straßenrelativ or Wegrelativ	The option to take the inclination of the original spline into account, when creating the new spline.
Tripod	Dreibein	Position and orientation of all models in anl3 files.

The document starts with a presentation of some more general change requests, which do not relate to specific user actions. Then follows a chapter describing problems when exchanging start and end of splines, and possible solutions. The most frequent complaints relate to copying splines in all directions. I did not find any issues when copying forwards or backwards. Thus, the next chapter concentrates on lateral copies. Here, "normal" copies and track relative copies are considered separately, as they raise different issues. Lateral copies with a reversed orientation do not cause additional problems to those being already documented in section 3, "Exchanging Start and End of a Spline" and in section 4, "Copying Splines Laterally".

In the last chapter, you will find two, more radical approaches to achieve a better user acceptance. The first one starts with a 2-dimensional construction and adds the 3rd dimension as a subsequent step. While documenting, I realized that most problems arise from inclination and that it could suffice to separate inclination from the other calculations, which forms the second alternative approach.

After proofreading the whole document, I would consider this last approach an especially valuable way to achieve a significantly better user acceptance.

2. General Requirements

2.1. Translations of spline types

Currently, the names of the spline types are always in English, regardless of the selected language. This can discourage new users.

Req. 1	The names of the spline types shall be translated like the user interface in general.

Suggestions for the German translations are:

English	German
Line	Gerade
Arc	Bogen
Rotator	Steigungsänderung
Helix	Schraubenlinie
Clothoid	Übergangsbogen
Cubic and “Spline”	Glatte Kurve

2.2. Selection of Multiple Splines

The creation of lateral tracks (railways, street lanes, sidewalks) is already possible for an arbitrary number of selected tracks (within one layer). The current options to select multiple splines are “rectangular lasso” by Ctrl-B plus LMB and “freehand lasso” by Ctrl-B plus Shift and LMB. This is not always sufficient.

Req. 2	In the 2D editor, an LMB click on a spline with the Alt key pressed, shall add the spline to the current selection, resp. remove it from the selection.

2.3. Connecting a New Spline to an Existing One

If a spline in a layout ends with an inclination (“Neigung”) and you drag a second, newly created spline to its end, it will put the inclination of the previous spline into the tripod of the new spline. Instead, it should change the start and end inclination of the new spline.

Req. 3	If a spline is newly created, and then dragged to the end of an existing one, which has a non-zero inclination, it shall use the same tripod inclination as the existing one, and a start inclination equal to the end inclination of the existing one.

Perhaps, it might be necessary to keep track about the fact being “newly created”.

2.4. Rotators with a Non-Zero Slope

Req. 4	Rotators with non-zero radius shall not add to nor subtract from the inclination, unless explicitly defined by the user. The slope at its end shall be the slope of the tripod plus the slope change of the rotator.

The current behaviour makes it very hard to end a slope with a curved rotator. In order to have an exactly horizontal continuation with the next spline, you have to find the correct values by some “trial and error” in the current version of EEP.

I am not (necessarily) talking about its internal representation, but in the user interface it should behave as described. Potential calculations or conversions should be hidden from the user.

2.5. Helices

Req. 5	Helices shall not have an end slope, nor in degrees nor in metres.

Whenever the end slope differs from the beginning slope (as defined in the tripod), it is not really a helix anymore.

If this request is implemented, existing helices with different start and end slopes (from earlier EEP versions) have to be converted, e.g. in reasonable cubics.

2.6. Spline Type “Spline”

Currently, there are spline types “Cubic” and “Spline”. As far as I can see, they behave identically.

Req. 6	Remove the spline type “spline” and replace their appearances by cubics.

2.7. Polygon courses

Polygon courses are generated by certain copies. Splines of this type cannot be edited and thus, lead to users’ disappointment. There are cases, where none of the existing spline types fulfils the needs. But, instead of introducing another spline type, it seems more user-friendly, to approximate the requested path by a (small) number of shorter splines. Generally, these will be cubics. They should be reasonably long and approximate the need visually good enough. They should be sufficiently long to avoid the problems of the older “Closing the track gap” (“Gleislücke schließen”) with lots of very short splines. I’d suggest a minimum length of 6 metres, which fits well with the texture repetition.

Req. 7	Remove the spline type “polygon course” and replace their appearances by small numbers of lines, arcs or cubics.

2.8. Maximum Angle and Minimum Length of Cubics

In several cases, copies of arcs, rotators and helices result in a cubic. With big z-angles, cubics do not approximate circular arcs very well. And whenever the cubic is long enough, it can be subdivided into shorter ones, which results in a better approximation.

I’d suggest a requirement like:

Req. 8	Whenever a copy results in cubics, **and** the z-angle is greater than 60°, **and** the length is greater than 10 metres, use shorter cubics.

3. Exchanging Start and End of a Spline

3.1. Lines, Arcs, Cubics, and Clothoids

Lines, arcs, cubics, and clothoids are handled correctly.

3.2. Rotators

Currently, exchanging start and end of a rotator leads to a new rotator with all entry fields greyed out.

If you take the shown (but unchangeable) values and create a new rotator at its place, it present a different geometry at its end.

This procedure shows the problem:

1. Place a line on an empty layout.

2. Add a rotator to the line with a radius of 100m, an angle of 60° and a slope of 5°.

3. Add another line to the end of the arc. (We have the expected orientation, a slope of 5° and no inclination.)
4. Exchange start and end of the rotator. The result is no longer editable.
5. Remove the first line and the rotator.
6. Create a new rotator with a radius of 100m, an angle of -60° and a slope of 5°. Add it to the beginning of the remaining line.
7. Add a new line to the end of the rotator and checks its values.

The resulting line should be like the line created in step 1, only with opposite orientation. But it has a different orientation, a different slope, and a different inclination.

Req. 9	Exchanging start and end of a rotator shall result in an editable rotator with the unchanged geometries on both ends.

3.3. Helices

When exchanging start and end of helices, the result is not editable, just as with rotators. But here, the geometry is intact, i.e. snapping a new spline to its start or its end leads to the expected orientation, slope, and inclination.

Req. 10	The parameters of a helix with start and end exchanged shall be editable.

4. Copying Splines Laterally

4.1. Wrong Spline Types and the Lack of Changeability

Please, solve the following nasty bug:

Req. 11	Splines resulting of a copy of an existing spline shall always be editable, unless the resulting spline type is a polygon course.

Currently, there are cases where lateral copies of rotators or helices cannot be edited any more. They are shown as arcs (which they should not be) and the entry fields are greyed out. This is considered a bug and should be solved soon, in a patch or an update.

4.2. Copying splines in the 3D editor

Users tend to use the 3D editor instead of the 2D editor. Therefore, copying tracks should be possible there, too. Multiple selections are done with the Alt key, which is fine here.

Req. 12	After selected one or more splines, a RMB click shall open the context menu, which shall contain an option “copy splines laterally” (“seitlich vervielfältigen”). A popup window shall appear, where the user sets the distance, the direction (left or right) and whether the copy should be track relative. The initial values shall be the same as when this popup was most recently used. After clicking OK, every connected set of splines (there might be several) shall be worked on in two passes: 1. The lines, arcs and helices shall be copied as defined. The newly created arc will have a bigger or smaller radius. The same hold true for helices. Additionally, helices shall have a newly calculated slope, so that the resulting height difference is the same as for the original spline. 2. Rotators, cubics and clothoids shall be handled in the second path. The resulting spline type shall be cubic for these splines. If they are connected to splines being handled in the first pass, they shall use the corresponding start and/or end slopes of these. Unconnected starts and ends shall use the slope of the original spline.

	The inclination at the start and the end of every spline shall be taken from the original spline.

With the outlined procedure, the users are ideally supported when creating parallel tracks, lanes, or sidewalks.

4.3. Normal Copies (not Track Relative)

4.3.1. Lines and Arcs

Lines are simple and behave correctly.

Arcs do the best possible job, even when they have a slope: The slope of the copied spline leads to the same height difference.

4.3.2. Helices

Lateral copies of helices result in arcs which are not editable. This is both wrong and user-unfriendly.

Req. 13	A lateral copy of a helix shall result in another helix with a different slope, which yields to the same height difference as the original.

4.3.3. Rotators, Cubics and Clothoids

A lateral copy of a rotator, a cubic or a clothoid results in a polygon course. This leads to two problems:

Req. 14	In order to be editable by the user, a polygon course shall be avoided generally as outlined under 2.7, “Polygon courses”. Instead, one or more cubics shall be used.

In the current implementation, the resulting polygon course ends in a slope that differs from the original rotator. In the case of a rotator, it looks like a preparation for a following helix with the same radius. For cubics and clothoids, the slope has still another value. Since it is not known, how the track will continue, I’d consider all these cases a bug.

Req. 15	An isolated lateral copy of a rotator, a cubic or a clothoid shall have the same start and end slope as the original.

If, however, the rotator to be copied is part of a bigger selection, appropriate and different slopes make sense, as outlined under 4.2, “Copying splines in the 3D editor”.

4.4. Track Relative Copies

4.4.1. Lines

A lateral, track relative copy of a line results in a polygon course. The result is not editable, and thus, not user-friendly. From my point of view, another line is sufficient.

Req. 16	A lateral, track relative copy of a line shall produce another line. The slope has to be calculated from the requested heights at the start and at the end of the line. The inclination has to be taken from the original.

If I am wrong and a line does not fulfil all needs, a single cubic (basically, it’s a line!) is preferred instead of a polygon course (see 2.7, “Polygon courses”).

4.4.2. All other spline types

In the most general case, a lateral, track relative copy of a spline cannot be represented by a standard spline type. Therefore, the resulting copy is a polygon course in the current implementation, which is not user-friendly, as it is not editable.

Furthermore, the start and end slopes of the copy differ from those of the original. Hence, it is not possible to create parallel lanes by snapping new splines to the ends of both, the original and the copy.

A single cubic will not always be a sufficiently good approximation. But a reasonable number of cubics can do the job and will be editable.

Req. 17	Lateral, track relative copies of all spline types shall result in a reasonable number of cubics (see also 2.7, “Polygon courses” and 2.8, “Maximum Angle and Minimum Length of Cubics”).

5. Two, More Radical Approaches

5.1. 2½ Dimensions

Of course, the title of this section is meant ironically. But nevertheless, we human beings live somehow in a 2½ dimensional world. We can move freely in the (x,y) plane. And we are aware of the 3rd dimension, but we cannot move just as freely there.

5.1.1. Planar view

To adapt this to the track system, the 2-dimensional curve in the (x,y) plane should be considered first. It can also be regarded as the projection of the final spline along to the z-axis. Here, in the plane, we only need the spline types:

- Line,
- Arc,
- Cubic and
- Clothoid

Rotators and helices are replaced by arcs, here.

And there is no inclination, yet.

Instead of the tripod with its six parameters, only four coordinates are required: x, y, z, and an angle α around the z-axis. (An astronomer would call α the azimuth.)

5.1.2. Height

In the next step, the height information has to be applied. When we take our planar curve and add the z-axis, we get a “curved plane”. The curved aspect has already been dealt with (in the plane). So, we can concentrate on the z-axis.

We have a height and a gradient at the beginning of the spline and a height at its end, as long as the end is not yet connected. These three parameters, z_{start}, α_{start}, and z_{end}, allow to calculate an arc (or a line, in the simplest case). Hence, we can build smooth changes in slopes.

The next spline inherits the slope from its predecessor, when snapped to it, and so forth.

If we place a spline between two other existing splines, we can inherit the slope at its beginning from the previous spline. But at the end, we might potentially cause an abrupt change in slope. If that is considered a problem, a third type of vertical curves has to be introduced, the cubic.

5.1.3. Inclination

Finally, we must handle the user-defined inclinations of the splines. Up to now, the orientation of the spline (orientation in the sense of the axes in the Home-Nostructor) has no z-component.

In a last computational step, the defined inclinations turn the orientation of the spline correspondingly.

5.1.4. Criticism

From a user perspective, this approach seems straight forward.

The conversion from the present track system into this system might be tricky at some places.

The proposed system does not support loops (vertical circles). But we are a train simulation, aren't we?

5.2. Isolated Inclinations

Users often complain that the tracks they lay, receive an unwanted inclination. And frequently, it becomes worse as they continue.

Part of the approach described above could be adapted to address this issue:

Here, the orientation of the curves is not derived from the tripods (orientation again in the sense of the axes in the Home-Nostructor). Up to this point, all curves have orientations with no z-component.

It is only the user-defined inclination which affects the orientation of the splines. This happens as the last step in the calculations.

With this approach, lateral copies of splines use the inclination from the original, for both, normal and track relative copies. Normal copies calculate x- and y-coordinates and use unchanged z-coordinates, whereas track relative copies must calculate x-, y- and z-coordinates.