

## Functionality Overview (SRR)

00_Arch - 00 - Architectural Features	Feature
<p>The architecture of the SRR Framework is well-defined and described.</p> <p>The architecture of the SRR Framework poses some requirements on the architecture of any SrrTrains layout.</p>	
00_Arch - 05 - Tracer	Feature
<p>The SRR Framework contains an X3D prototype for tracing purposes. It's the intention of the SrrTracer,</p> <ul style="list-style-type: none"><li>- to get familiar with SrrTrains software</li><li>- to debug SrrTrains software</li><li>- to document SrrTrains software</li></ul>	
00_Arch - 10 - MMF Architecture	Feature
<p>The SRR Framework requires from each and every SrrTrains layout, to be developed according to the "MMF paradigm" (model/module/frame paradigm).</p> <p>This does not preclude having all three kinds of X3D/VRML scenery in one single file, in the case of simple scenes.</p>	
00_Arch - 15 - Static Modules	Feature
<p>Static modules are modules, which are loaded and initialized immediately at the startup of the scene instance.</p>	
00_Arch - 20 - Dynamic Modules	Feature
<p>Dynamic modules are modules, which are loaded, initialized and unloaded on demand, at any instance during the lifetime of the scene instance.</p>	
00_Arch - 25 - SRR Objects	Feature
<p>Providing SRR Objects, is the central service of the SRR Framework.</p> <p>It's the SRR Objects that are used by model authors and by module authors to provide interactive and animated MU capable models.</p>	
00_Arch - 30 - Console Interface	Feature
<p>Each SRR Object can provide parameters, that can be changed and read out by the console interface (a command line interface, that is provided by the SRR Controller).</p>	

## Functionality Overview (SRR)

00_Arch - 35 - Static/Intrinsic Models	Feature
SRR Objects can be used to create static/intrinsic models. Those will be MU capable, interactive and animated.	
Static models are loaded and initialized immediately together with their parent module. However, they can be provided independently of their parent module and they may be used by more than one module.	
Intrinsic models are actually a part of their parent module and cannot be used without their parent module.	

00_Arch - 40 - Modularity of the SRR Framework	Feature
It is possible to extend the SRR Framework by implementing your own SRR Objects.	
However, if those SRR Objects needed new functionality from the SRR Controller or from the Module Coordinator, you would be dependent on the SIMUL-RR project.	
For this case, it is possible to support new types/classes of SRR Objects by extending the SRR Controller and/or the Module Coordinator. This extensibility of the SRR Controller and of the Module Coordinator is called "Modularity of the SRR Framework".	

00_Arch - 45 - Dynamic Models	Feature
THE FEATURE "DYNAMIC MODELS" IS NOT AVAILABLE YET.	
It was planned for step 0033, to support dynamic models of rail vehicles with the Train Manager Extension.	
A very drafty sketch of some ideas is collected in the concept paper concepts/052_DynamicModels.txt.	

00_Arch - 50 - Handover	Feature
THE FEATURE "HANDOVER" IS NOT AVAILABLE YET.	
It was planned for "some step beyond step 0033", to support handover for rail vehicles. That means, some kind of "handover track" should be defined to connect tracks of one module with tracks of another module.	

## Functionality Overview (SRR)

00_Arch - 55 - Moving Modules	Feature
THE FEATURE "MOVING MODULES" IS NOT AVAILABLE YET.	
It was planned for "some step beyond step 0033", to support train movers, i.e. to support moving tracks and turnouts (that could be contained in arbitrary models that could be contained in arbitrary modules in turn, i.e. they could be contained even in dynamic models changing their module). This should be realized by the so-called SRR Object "Module Container".	
The concept of the handover tracks should be generalized, to support <ul style="list-style-type: none"><li>- turntables</li><li>- trains on a ferry ship</li><li>- rail vehicles transported by other rail vehicles</li></ul>	
01_Conv - 00 - Convenience Features	Feature
Convenience Features do not define specific functionality. They are general properties of the product, which make it easier - more convenient - to use the product.	
01_Conv - 05 - X3D Player Support	Feature
This feature documents, which Web3D Browsers are supported by the SRR Framework. Currently, following Web3D Browsers + MU Systems are supported. <ul style="list-style-type: none"><li>- Instant Player (singleuser mode)</li><li>- BS Contact (singleuser mode)</li><li>- BS Contact + BS Collaborate (multiuser mode)</li></ul>	
01_Conv - 06 - MU System Support	Feature
This feature documents, which MU Systems are supported by the SRR Framework. Currently, following Web3D Browsers + MU Systems are supported. <ul style="list-style-type: none"><li>- Instant Player (singleuser mode)</li><li>- BS Contact (singleuser mode)</li><li>- BS Contact + BS Collaborate (multiuser mode)</li></ul>	
01_Conv - 10 - Authoring Support	Feature
THE FEATURE "AUTHORING SUPPORT" IS NOT AVAILABLE YET.	
It was planned for "some step beyond step 0033", to implement Python scripts for the Blender editor, to support (more) easy modeling of SrrTrains layouts.	
01_Conv - 15 - Docu: Project WIKI	Feature
The sourceforge project "simulrr" provides a project WIKI.	

## Functionality Overview (SRR)

01_Conv - 25 - Docu: Concepts' Descriptions	Feature
<p>The sourceforge project "simulrr" provides the "source"code (in .odt format) and the created .txt files of the descriptions of all concepts.</p> <p>The Concept's Descriptions intend to support developers and users (model authors, module authors, frame authors) of the SRR Framework.</p> <p>However, they cannot be compared to a well-defined set of user manuals, which is still missing.</p>	
01_Conv - 30 - Docu: Testcase Descriptions	Feature
<p>Some testcases are officially documented, to support developers getting familiar with the SRR Framework.</p>	
01_Conv - 32 - Docu: Product Database	Feature
<p>The product database is a relational data base that holds a matrix of "objects" and "reasons for change".</p> <p>The intersection of a row of the matrix ("reason") with a column ("object") is a "change".</p> <p>Objects are grouped in "classes" and "subclasses".</p> <p>Reasons are grouped in "target releases".</p>	
01_Conv - 35 - Monolithic Layout (.zip)	Feature
<p>A monolithic layout is a layout, where each file can be reached by a relative URL.</p> <p>SrrTrains defines to have a "two-level" directory structure.</p> <p>The SRR Framework has been tested in a "local monolithic layout".</p>	
01_Conv - 40 - Monolithic Layout (Webspace)	Feature
<p>THIS FEATURE HAS NOT BEEN TESTED.</p> <p>A monolithic layout is a layout, where each file can be reached by a relative URL.</p> <p>SrrTrains defines to have a "two-level" directory structure.</p> <p>The SRR Framework has NOT been tested in a "monolithic layout on a WebSpace".</p>	
01_Conv - 45 - Distributed Layout	Feature
<p>THIS FEATURE HAS NOT BEEN TESTED.</p> <p>A distributed layout is a layout, where each file can be reached by an absolute URL.</p> <p>Each file can be downloaded from a different host.</p> <p>The SRR Framework has NOT been tested in a "distributed layout".</p>	
10_SRR - 00 - Features of the Base Module	Feature
<p>The features of the feature group "SRR - Features of the Base Module" are provided by following components:</p> <ul style="list-style-type: none"><li>- SRR Framework Base Module</li><li>- Example Basic SRR Objects</li></ul>	

## Functionality Overview (SRR)

10_SRR - 05 - Enter Moving Model	Feature
<p>The "bouncing avatars problem" is solved by the help of so-called avatar containers (a special kind of SRR Objects).</p> <p>When a user binds an animated viewpoint, the positions and orientations of his pilot avatar should be transmitted in values relative to the animated coordinate system.</p>	

10_SRR - 10 - Join Avatar	Feature
<p>"Beam to a user" means selecting the user from a list of nicknames and being moved to the position of that user, into a face-to-face position.</p> <p>It's up to the frame to do this, because the frame has access to the &lt;BSCollaborate&gt; node and it's "users" field.</p> <p>However, this action has to be supported by the SRR Framework, because avatars can be contained in avatar containers and position as well as orientation of avatars will be transmitted in relative coordinates.</p> <p>This function is only supported with BS Contact and BS Collaborate, currently.</p>	

10_SRR - 15 - Binary Switch	Feature
<p>The "Binary Switch" SRR Object supports module authors and model authors in situations, where an SFBool state has to be maintained (e.g. to open/close doors, to switch lights on and off and so on). The state is maintained globally (it is shared among all scene instances). A local animation can be output (SFFloat between "0.0" and "1.0", where "0.0" fits to "false" and "1.0" fits to "true").</p>	

10_SRR - 20 - Carousel Drive	Feature
<p>The "Carousel Drive" SRR Object supports module authors and model authors in situations, where a cyclic movement can be switched on and off smoothly.</p> <p>The state (SFFloat "0" to "2.pi") is maintained globally (it is shared among all scene instances).</p> <p>A "binary switch" SRR Object is contained in the "carousel drive" SRR object to switch the rotation of the carousel on and off.</p>	

10_SRR - 25 - Trigger	Feature
<p>THE FEATURE "TRIGGER" (AN SRR OBJECT) IS NOT AVAILABLE YET.</p>	

20_BMM - 00 - Beamer Manager Extension	Feature
<p>The features of the feature group "BMM - Beamer Manager Extension" are provided by the component</p> <ul style="list-style-type: none"><li>- Beamer Manager Extension</li></ul>	

## Functionality Overview (SRR)

20_BMM - 05 - Define Meeting Point	Feature
The "Beamer Manager Extension" of the SRR Framework maintains a list of so-called "beamer destinations".	
A viewpoint can be made a beamer destination, by creating a "Beamer Destination" SRR Object and referencing the viewpoint therein.	
20_BMM - 10 - Beam to Meeting Point	Feature
A "Beamer" SRR Object can be used to output a list of all beamer destinations and to select one of them to be bound.	
Selecting beamer destinations can be additionally done via the user interface of the SRR Controller.	
30_KMM - 00 - Key Manager Extension	Feature
The features of the feature group "KMM - Key Manager Extension" are provided by the component - Key Manager Extension	
30_KMM - 05 - Create/Take/Put/Reset Keys	Feature
Keys can be created within key containers and they can be taken by an avatar or put back to a key container.	
It's possible to reset all key containers and carried keys globally.	
30_KMM - 10 - Carried Keys Lock	Feature
Keys can unlock a "carried keys lock", when they are carried by an avatar.	
30_KMM - 15 - Contained Keys Lock	Feature
THE FEATURE "CONTAINED KEYS LOCK" (AN SRR OBJECT) IS NOT AVAILABLE YET.	
40_TMM - 00 - Train Manager Extension	Feature
The features of the feature group "TMM - Train Manager Extension" are provided by the component - Train Manager Extension	

## Functionality Overview (SRR)

40_TMM - 03 - Tracks and Turnouts	Feature
The author of the models of tracks and turnouts can use the SRR Objects "Track Node", "Track Edge", "Basic Track Section", "Basic 2-Way Turnout" and an "N-Way Switch" to model a mesh of nodes and edges. The SRR Objects for rail vehicles will use this mesh to move the vehicles "along the tracks".	
40_TMM - 06 - Rail Vehicles	Feature
THE FEATURE "SRR OBJECTS FOR RAIL VEHICLES" IS NOT AVAILABLE YET.	
40_TMM - 09 - Create Vehicle	Feature
A "Setup Point" SRR Object can be used to define a location and an initial state for the creation of "1-Vehicle-Trains".  The "Setup Point" outputs a (filtered) list of registered vehicle types and can be used to select one vehicle type and to create a vehicle of this type.  THE FEATURE "CREATE VEHICLE" IS NOT AVAILABLE YET, HOWEVER, THE "SETUP POINT" SRR OBJECT ALREADY EXISTS.	
40_TMM - 12 - Move Train	Feature
THE FEATURE "MOVE TRAIN" IS NOT AVAILABLE YET.	
40_TMM - 15 - Switch the Points Manually	Feature
The switch, which is contained in an "Track Node" object of a turnout, can be the "standard N-Way Switch" SrrSwitchB, which can be used to switch the points manually.	
40_TMM - 18 - Vehicle's Basic User Interface	Feature
THE FEATURE "VEHICLES BASIC USER INTERFACE" IS NOT AVAILABLE YET.	
40_TMM - 21 - Delete Train	Feature
THE FEATURE "DELETE TRAIN" IS NOT AVAILABLE YET.	
40_TMM - 24 - Vehicle's User Interface	Feature
THE FEATURE "VEHICLES USER INTERFACE" IS NOT AVAILABLE YET.	

## Functionality Overview (SRR)

40_TMM - 30 - Gauge Check	Feature
THE FEATURE "GAUGE CHECK" IS NOT AVAILABLE YET.	
40_TMM - 35 - Handling of Derailment	Feature
THE FEATURE "HANDLING OF DERAILMENT" IS NOT AVAILABLE YET.	
40_TMM - 40 - Train Changes Module	Feature
THE FEATURE "TRAIN CHANGES MODULE" IS NOT AVAILABLE YET.	
40_TMM - 45 - Bumpers	Feature
THE FEATURE "BUMPERS" IS NOT AVAILABLE YET.	
40_TMM - 50 - Coupling and Collisions	Feature
THE FEATURE "COUPLING AND COLLISIONS" IS NOT AVAILABLE YET.	
40_TMM - 55 - Decoupling Track	Feature
THE FEATURE "DECOUPLING TRACK" IS NOT AVAILABLE YET.	
40_TMM - 60 - Train Movers	Feature
THE FEATURE "TRAIN MOVERS" IS NOT AVAILABLE YET.	
40_TMM - 65 - Bursting Open the Points	Feature
THE FEATURE "BURSTING OPEN THE POINTS" IS NOT AVAILABLE YET.	
40_TMM - 70 - Derail in Curves (Speeding)	Feature
THE FEATURE "DERAIL IN CURVES (SPEEDING)" IS NOT AVAILABLE YET.	
40_TMM - 75 - Derail on Points	Feature
THE FEATURE "DERAIL ON POINTS" IS NOT AVAILABLE YET.	

## Functionality Overview (SRR)

---

40_TMM - 80 - Lock the Points	Feature
THE FEATURE "LOCK THE POINTS" IS NOT AVAILABLE YET.	

40_TMM - 85 - Interlocking (1900's)	Feature
THE FEATURE "INTERLOCKING (1900's)" IS NOT AVAILABLE YET.	