The Decision

Table of Contents

1 Introduction

Tbd.

2 A Vision of SrrTrains[1]

Some of my readers might already know the history of the SrrTrains project, therefore I am focussing on the most important buzz words that have to be known, if you like to understand SrrTrains:

- Some "inspiration"[2] was happening during the years 1993 – 2001 and hence
- I came to the conclusion that 3D graphics would become an important topic at the telecom industry (my employer), so I phrased the idea DIGITS[3] in year 2002
- Nothing happened
- In year 2007 I started to gain know how about 3D graphics, this was the time, when I invented the SIMUL-RR project, **which should combine a multi player virtual railroad with a real life model railroad (or even with real railways)**
- In 2009 I decided to use the X3D standard with it's NSN[4] and I renamed the project from SIMUL-RR to SrrTrains v0.01

The Mixed Reality Session should consist of

- **Virtual Players**, using some VR equipment or at least flat 3D displays with mouse input
- **Real Players**, using classic control equipment to control the model railroad
- a VR enhanced model railroad
- an **Interface To Reality**, which would connect the VR multiuser session with the model railroad
- Some **Connectivity Platform** to connect the scene instances and the ITR within the **Multiuser Session**

All this is depicted in Figure 1:

1 SrrTrains = Simulated Railroad Trains
2 Please refer to https://letztersein.com/2017/05/28/brief-an-meine-tochter-2017-05-25/ (German language)
3 DIGITS = Distributed Internet Geographic Information Transmission Service
4 NSN = Network Sensor Node

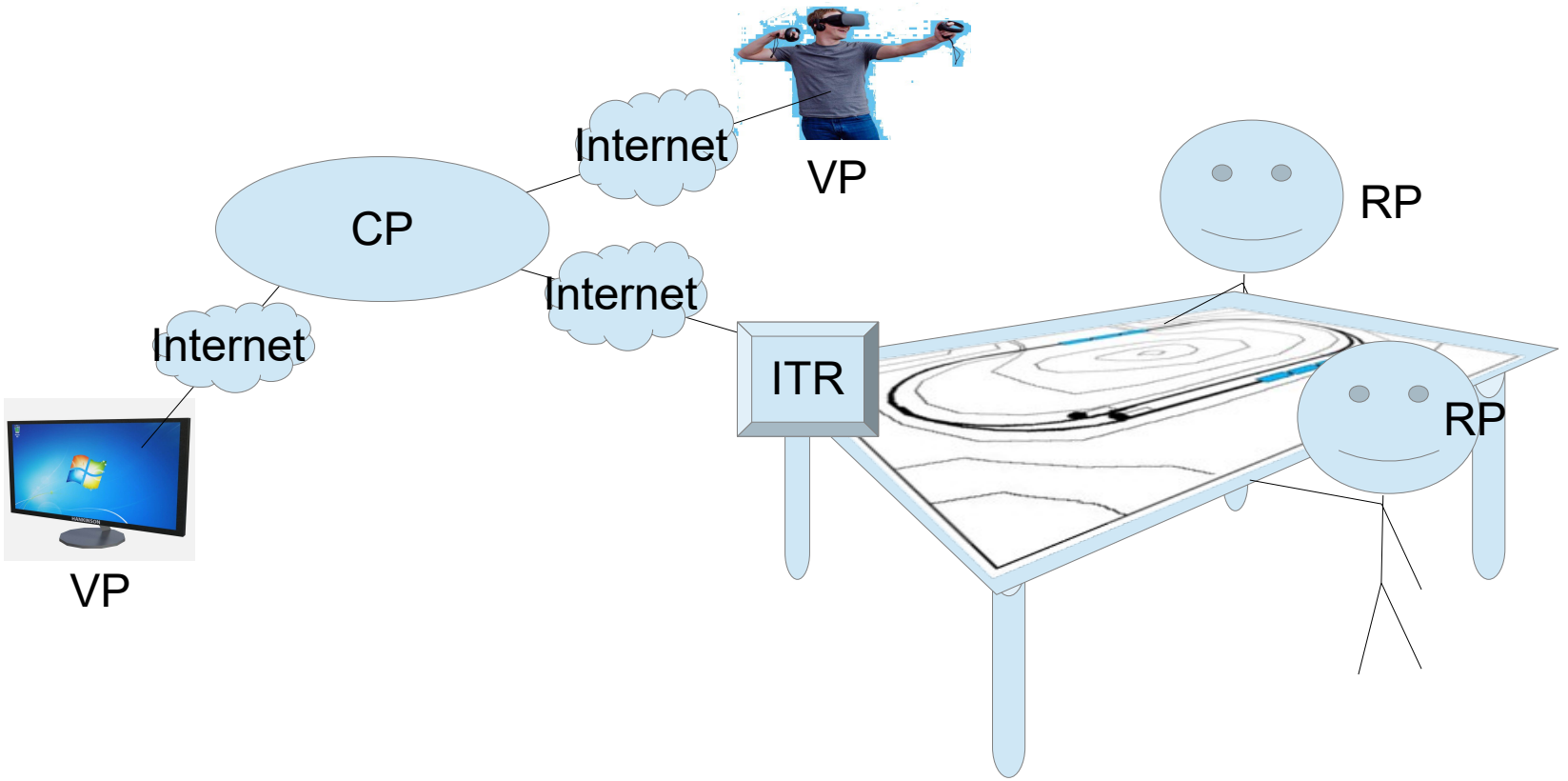


Figure 1: Virtual Players (VP), Real Players (RP) and an Interface to Reality (ITR) are connected via a Connectivity Platform (CP)

3 What Has Happened Until Now (Steps 0001 to 0032, 0033.01 to 0033.10)[5]

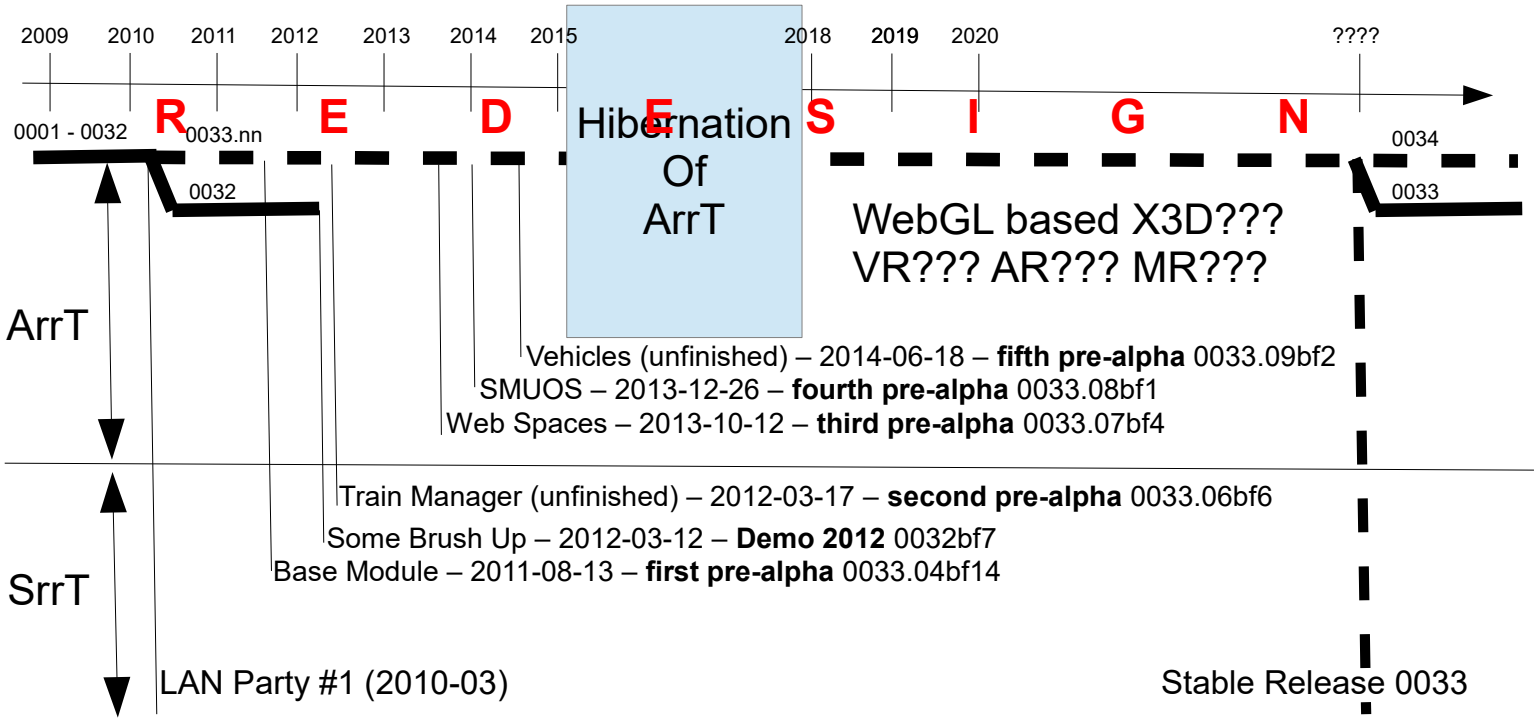


Figure 2: How to get from instable software "Demo 2012" to "Stable Release 0033"

5 Please refer to https://letztersein.com/srrtrains-v0-01 for details.

4 Planning of Release "Fiat A"

This hobby report describes some topics that could(!) be introduced with release "Fiat A" of the SrrTrains v0.01 project, eventually, i.e. Scene Wrappers and GUIs.

Some experimental Scene Wrapper GUI was already available with the former "official pre-alpha releases", e.g. with the "Fifth official pre-alpha release 0033.09bf2" by 2014-06-18 (please refer to https://www.mediafire.com/folder/edhwj4cacq0m5/SrrTrains-v0.01 to download the software).

Since then, the SrrTrains v0.01 project has existed in hibernation mode and had only "inofficial releases" without Scene Wrapper GUI (0033.09bf4 and 0033.10bf3).

Now, when we could(!) be going to start the project again, then we should(!) want to have a next release, which could(!) have all necessary parts including a Scene Wrapper GUI.

Some **still missing functionality** – and **some missing architectural features** – could(!) be added by the following release (beyond 2020).

The **Train Manager Extension** could(!) then be finished by the last release of the 0033 series.

This would then finally lead to the "**Alpha Release of SrrTrains v0.01**".

Prophecy (i.e. Planning):

- **Release Fiat A 0033.10.5 "not before 2020", if at all**
- **Release Arimathea 0033.11 "beyond 2020", if at all**
- **Release Sabbath 0033.12 "even later", if at all**

The next figure shows what could(!) happen in detail, if we started the release "Fiat A" (0033.10.5)

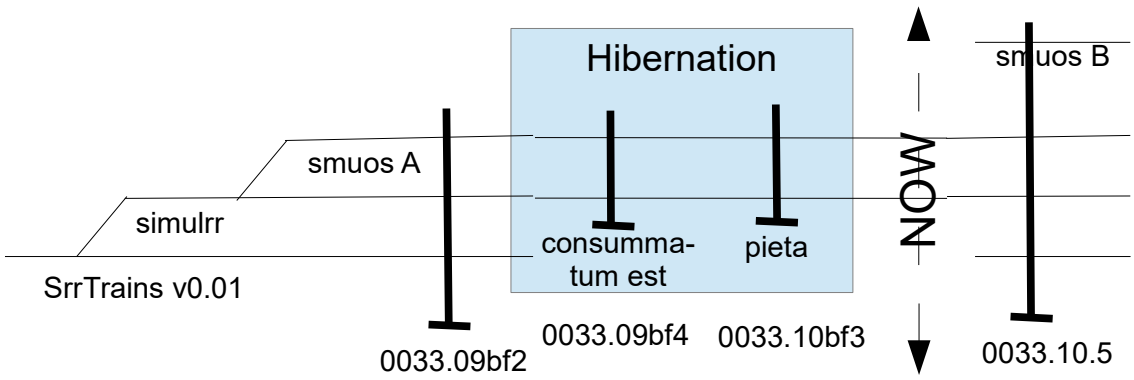


Figure 3: Details of Release "Fiat A"

That means:

- Scene Wrappers/GUIs could(!) be implemented (it could be the "sixth official pre-alpha")
- the work on SMUOS B could(!) be started (SMUOS B is the same like SMUOS A, but based on WebGL)

5 The Envisioned OMs for SrrTrains v0.01

The first step of the "Fiat A" project will be to **define a Testing Scene**, where all features of the SMUOS Framework can be tested within all of the following **Operational Modes (OMs)**:

5.1 Single User VR OM – VR with Classical X3D Player

A classical X3D Player (e.g. BS Contact) will be used with an INTERNAL GUI to play in SINGLE USER VR MODE:

Figure 4: OM Single User VR

5.2 Single User Desktop OM – Desktop VR with Classical X3D Player

An EXTERNAL GUI application will be written to play in SINGLE USER DESKTOP MODE:

Figure 5: OM Single User Desktop

5.3 Single User Desktop NG OM – WebGL Based X3D Player

Experience with WebGL will be gained and a Browser application will be written to play in SINGLE USER DESKTOP MODE:

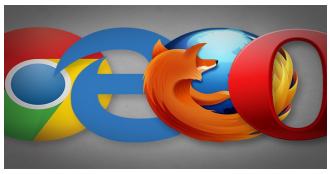


Figure 6: OM Single User Desktop NG

5.4 Single User VR NG OM – VR with WebGL Based X3D Player

A WebGL based X3D Player will be used to play in SINGLE USER VR MODE:

Figure 7: OM Single User VR NG

5.5 Multiuser Desktop+VR OM with any X3D Player

Classical X3D Players, WebGL Based X3D Players and a Connectivity Platform will be used for Multiuser Desktop + VR Mode.

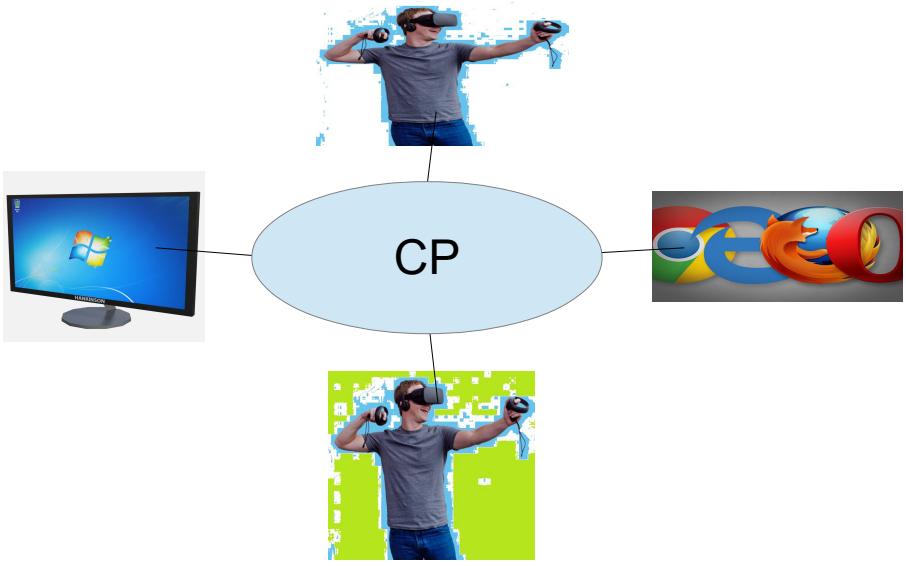


Figure 8: OM Multiuser Desktop + VR

5.6 Mixed Reality OM

Classical X3D Players, WebGL Based X3D Players, an ITR and a Connectivity Platform will be used for Mixed Reality Mode.

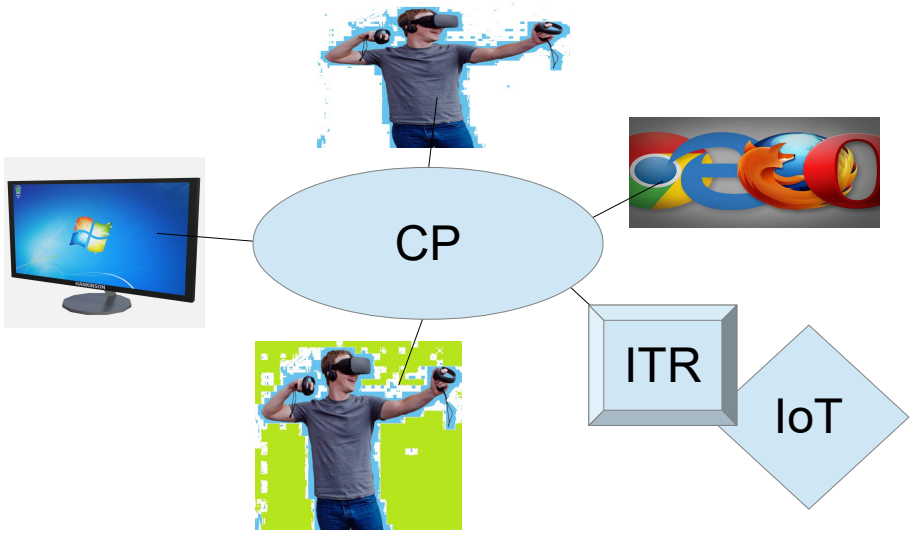
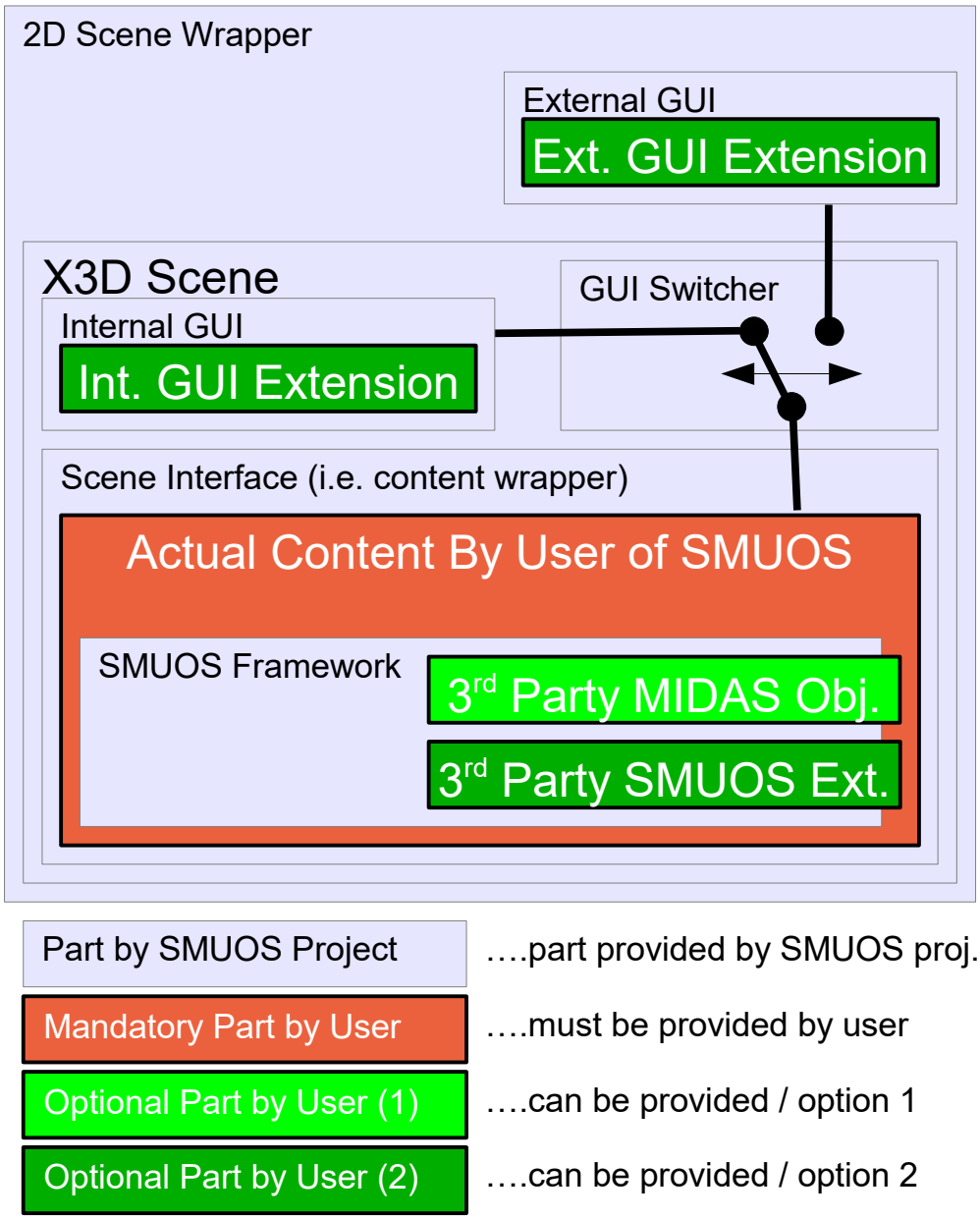


Figure 9: OM for Mixed Reality

6 Sub-Projects for the Fiat A Release

6.1 SMUOS Project (http://smuos.sourceforge.net)

The SMUOS Project could(!) provide all essential software that will be necessary to create – experimental – X3D based Simple Multiuser Scenes (SMS).



SMUOS could(!) specify all external interfaces, so following parts could(!) be replaced by your own software: Scene Interface, GUI Switcher, Internal GUI, External GUI, 2D Scene Wrapper.

Only the SMUOS Framework itself will be essential to create SMS.

6.2 SIMULRR Project (http://simulrr.sourceforge.net)

The SIMULRR project could(!) be an example of a project that uses the SMUOS project.

The SIMULRR project:

- could(!) provide extension MIDAS Objects for Railway Simulations (SRR Objects)
- could(!) provide a new SMUOS Extension – the Train Manager Extension (TME)
- and the according internal and external GUI extensions

6.3 SrrTrains v0.01 (http://members.chello.at/christoph.valentin)

The SrrTrains v0.01 project could(!) be a wrapper project that just uses simulrr and smuos and adds some documentation.

7 Relation to Other Projects

The project SrrTrains is related to three other projects of the projects that I am currently handling:

Project	Home Page	Status (2019-08-15)
SrrTrains	SrrTrains v0.01 SrrTrains 1.0 (tbd.)	Closed Not Started
N.I.L. / DIGITS	N/A	Not Started
S&P-ARK	https://areasharpa.blog/spark	Doesn't matter
P&S-ARK	https://areasharpa.blog/psark	Doesn't matter

The description of all seven (7) projects can be found at https://letztersein.com/die-projekte (German language).

SrrTrains and DIGITS were inspired by some "initial inspiration" (before 2007, as I wrote).

Then the SRR Framework was inspired by the SrrTrains project: we outsourced the SRR Framework to an own sourceforge project in 2009.

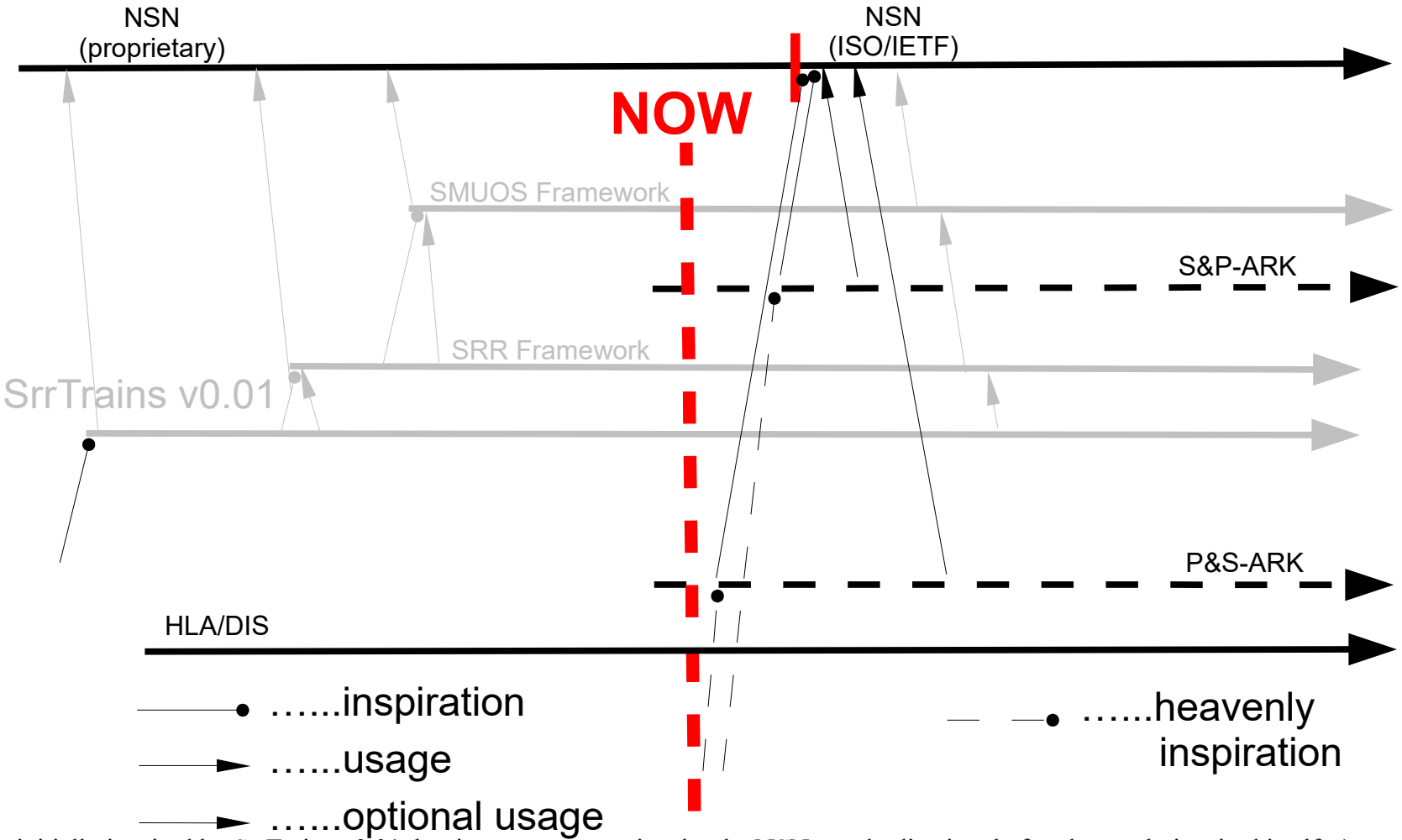
Then the SMUOS Framework was inspired by the SRR Framework: we outsourced the "base module" of the SRR Framework to an own sourceforge project in 2013.

The projects S&P-ARK and P&S-ARK are "template projects". That means, they are not "real" projects but I use them to publish my opinion about projects that SHOULD be started, imho.

S&P-ARK should elaborate the "community requirements" for the Network Sensor Node (NSN) and then – after its implementation – use it.

P&S-ARK should elaborate the "heavenly requirements" for the Network Sensor Node (NSN) and then – after its implementation – use it.

Also S&P-ARK and P&S-ARK could use SMUOS, optionally.



SP-ARK was initially inspired by SrrTrains v0.01, but it cannot start to inspire the NSN standardization, before heavenly inspired itself :-)