Free-viewpoint Immersive Networked Experience

D7.1 Test Protocols and Methodology report for proof-of-concept trials

Project ref. no.	ICT-FP7-248020
Project acronym	FINE
Start date of project (dur.)	1 April, 2010 (36 months)
Document due Date :	31.01.2011
Actual date of delivery	31.01.2011
Leader of this deliverable	Abertis Telecom
Reply to	david.pujals@abertistelecom.com carmen.vicente@abertistelecom.com
Document status	Draft Version

Version	**Date**	**Description**
0.1	14/01/11	Final draft for internal review
1.0	31/01/11	Final version

SEVENTH FRAMEWORK PROGRAMME

Deliverable Identification Sheet

Project ref. no.	ICT-FP7-248020
Project acronym	FINE
Project full title	Free-viewpoint Immersive Networked Experience
Document name	Test Protocols and Methodology report for proof-of-concept trials
Security (distribution level)	PU – Public
Contractual date of delivery	Month 10, 31.01.2011
Actual date of delivery	Month 10, 31.01.2011
Deliverable number	D7.1
Deliverable name	Test Protocols and Methodology report for proof-of-concept trials
Type	Report
Status & version	Final version, v1.0
Number of pages	18
WP / Task responsible	WP7 / RETE
Other contributors	
Author(s)	Abertis Telecom
EC Project Officer	Ertrit Puka
Abstract	Review of the FINE processes chain for the proof-of-concept trials
Keywords	Acquisition, production, processing, production, coding, network and exhibition processes, protocols, formats, test, prototype, trial
Sent to peer reviewer	*15/01/2011 Jordi Ramon (MPRO)*
Peer review completed	*Date of completion*
Circulated to partners	Via plone
Mgt. Board approval	To be approved at the next SB meeting

1. Public Executive Summary

The purpose of this document is to report the work achieved in WP7 regarding the test protocols and methodology for the proof-of-concept trials. This deliverable is based on the user's scenarios definition and on the requirements of technologies involved in every process of the project. Methodology proposed is that every process of the FINE project must be tested as alone as join with its neighbors processes: alone to check if is well-working, and joined with its neighbors to check the compatibility through an interoperability test bed. At the end, a completed end-to-end scenario that responds to the requirements of the FINE project must be obtained.

The information detailed in this document reflects the work done in Task 7.1.

The document is organized as follows:

- Introduction
- Methodology
- Test planning
- Fine processes chain and test protocols
- Conclusions

2. Introduction

The purpose of this document is to report the work achieved in WP7 regarding the test protocols and methodology report for the proof-of-concept trials. This deliverable is based on the user's scenarios definition and on the requirements of technologies involve in every process of the project. The methodology proposed assures that every process of the FINE project is tested both standalone as well as joined with its neighbors processes: standalone to check if is working well, and joined with its neighbors to check the compatibility through an interoperability test bed. At the end, a completed end-to-end scenario that responds to the requirements of the FINE project must be obtained and validated.

This deliverable contains the baseline for planning the test-beds in laboratory and the proof-of-concept trials and field trials for the complete FINE chain.

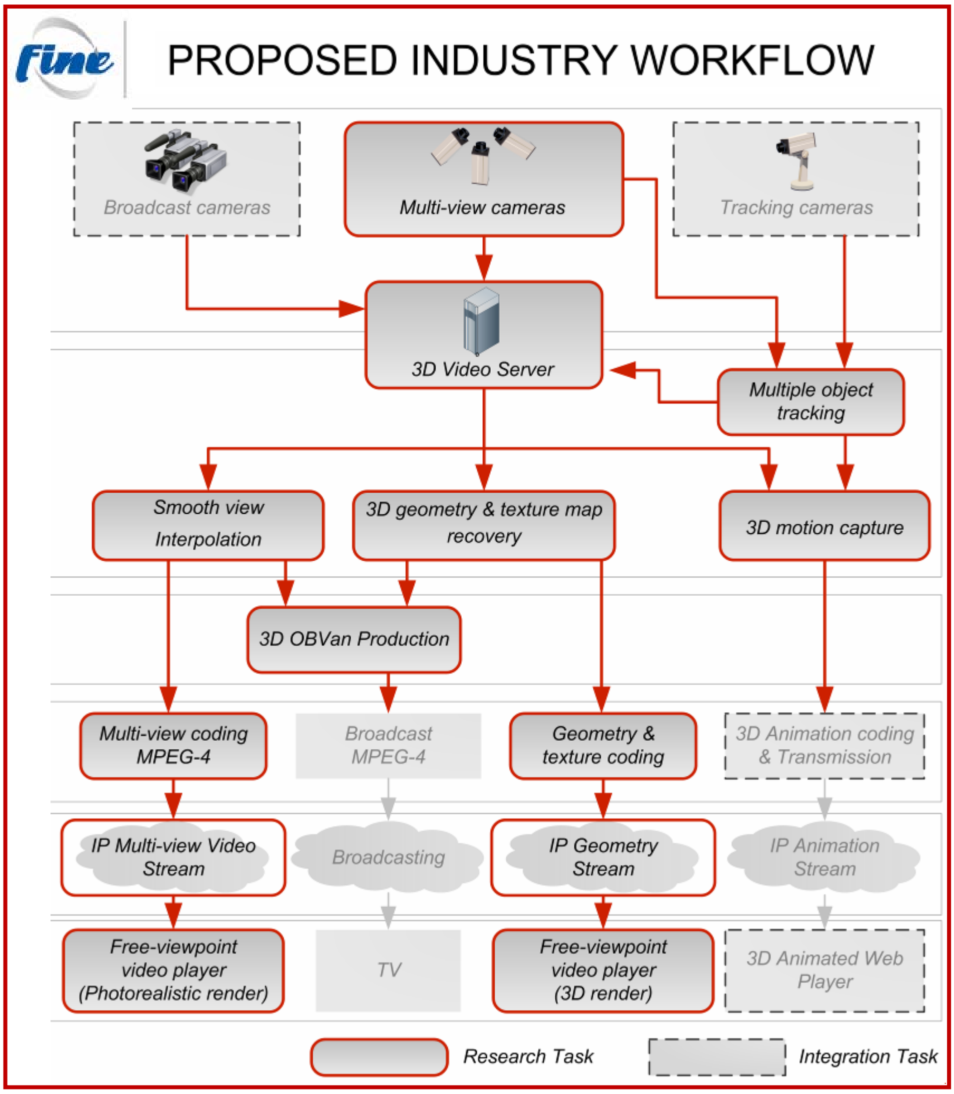
The goal of the WP7 is to provide the test methodology and to evaluate the results of these tests in order to guide the developers and obtain the milestones of the FINE project.

This D7.1 deliverable is the result of the WP7.T1 task. This task has analyzed all the FINE processes chain, the definition of the users' scenarios and the deliverables delivery date in order to propose the best methodology to fulfill FINE's objectives and for planning the test and the trials. Test protocols will be based on the analysis of the FINE processes chain and its interdependencies.

3. Methodology

As this is a very complex project, with distributed partners and a lot of complex processes involved, the best way to complete successfully the FINE project is to realize for every process of the FINE chain two kind of tests: standalone and integrated with its neighbors processes.

The following figure shows all the processes involved in the FINE project. This will be referred to as the FINE processes chain.



For a given process, for example: encoding, standalone tests will check if the process is working correctly. These standalone tests could be done by the partner responsible for developing the process in its own site, without needing coordination with other partners of the project. This way, different processes could be developed and tested in parallel.

But to assure a correct development of the project, in order to make all the processes compatible, some interoperability tests must be done: the integration tests. These tests will be done once every process has successfully passed its preliminary standalone test and are available to integrate with the others.

Integration test will be done first in the laboratory, and once there is a successfully result, field trials would be done to better demonstrate the results of the FINE project.

4. Test planning

The following picture shows the time line for the FINE project.

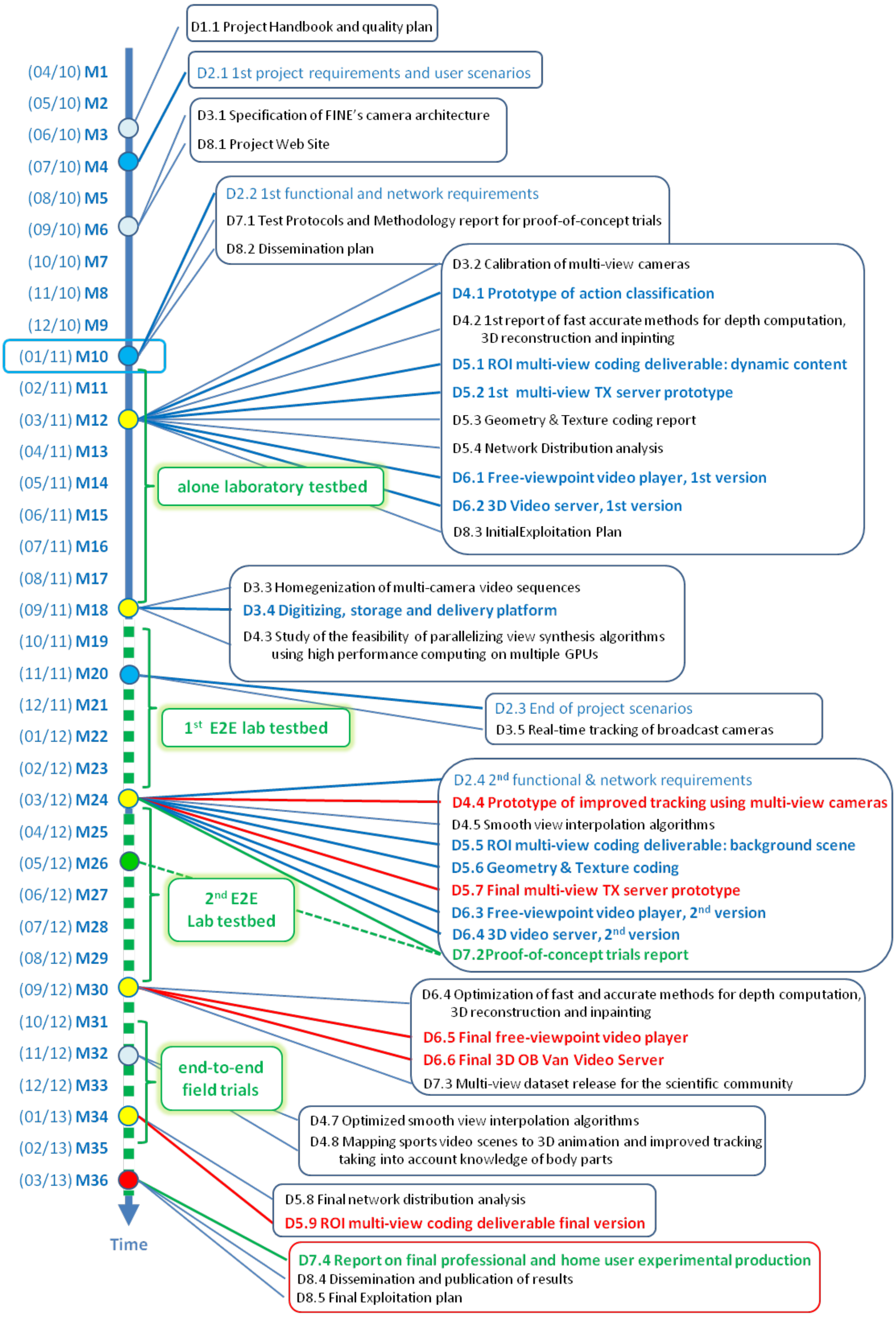


Figure 1: FINE project time line

This time line shows the delivery date of each deliverable, and the tests proposed.

4.1. Standalone laboratory test bed (M10 – M18)

Once users scenarios and project requirements has been established, every process responsible partner can start with the standalone tests to check the correctness of the process.

At M12 some first prototypes will be available for integrations:

- D4.1 Prototype of action classification
- D5.1 ROI multi-view coding deliverable: dynamic content
- D5.2 First multi-view TX server prototype
- D6.1 Free-viewpoint video player, 1st version
- D6.2 3D video server, 1st version
-

At M18 another one will be available for integration:

- D3.4 Digitizing, storage and delivery platform.

4.2. First end-to-end laboratory test bed (M19 – M24)

These first end-to-end laboratory tests will take place from month 19 to 24. These will be a gather of all involved technologies. The aim of these tests is to check that every single development carried out by each partner will work when assembling to the other parts.

At M19:

- All project scenarios will be completely defined.
- Real-time tracking of broadcast cameras will be available for testing.

At M24:

- Prototype of improved tracking using multi-view cameras
- Final multi-view TX server prototype
- Free-viewpoint video player, 2nd version
- 3D video server, 2nd version

We suggest moving the deliverable D7.2 from M24 to M26 to have enough time for reporting all the conclusions from evaluations and tests performed.

4.3. Second end-to-end lab test (M24 – M30)

The aim is to follow with advanced tests and to help check the final chain and to assure that they will work well when we put them all together.

At M30:

- Final free-viewpoint video player
- Final 3D OB Van Video Server

4.4. End-to-end field trials (M32 – M35)

The objective of these field trials is to check all project chain to assure that complete FINE prototype is working properly in a real scenario.

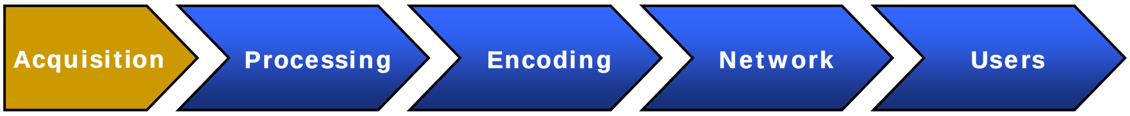
At M34:

- ROI multi-view coding deliverable final version.

All these trials have a period window when they will be carried out, but we estimate that the duration of the tests will actually be smaller than the window period defined.

5. Fine Processes Chain and Tests Protocols

5.1. Acquisition



Acquisition processes will be considered in both professional and home user scenarios. Acquisition is designed by WP3, where WP3T1 describes the architecture and the other shall implement lab and field tests concerning FINE project.

WP3T1 work is about camera architecture and will be the base for these tests.

5.1.1. WP3T2 & WP3T3 Camera calibration

This task will calibrate moving broadcast cameras to enable the integration of broadcast cameras with the rest of acquisitions systems. The task is structured in two main tracks: first track is related to developing image-based tracking algorithms; and the second track relates to developing a simple and inexpensive system with sensors. Both tracks will converge in a hybrid real-time camera tracking.

The methodology to test each track and the hybrid system has different metrics. In image-based algorithms, the first measure is to insert virtual lines on the pitch and measure the error with real lines. Starting with static images that will be tested in a controlled environment such as a small field in a laboratory. Next, the vision algorithms will be tested with real images with good lighting conditions and then changed to more complex images with shadows or areas with few lines. When the system works with still images sequences of images will be tested. A good measure is to insert virtual objects on the pitch and see which are kept in a fixed position without vibration as the camera is moved.

The methodology to validate sensor-based technologies is based on measuring the error between the estimated motion and the ground true obtained with a tripod with encoders. As the same way as was done with image-based algorithms, you should begin testing in controlled environments and simple movements. Then, we will continue to testing in uncontrolled environments and complex movements.

The methodology to test hybrid system is a combination of previous methodologies, adding virtual objects in the images and comparing results with the ground true.

Availability dates:
Month 20: Del 3.5. Tracking of broadcast cameras

5.1.2. WP3T4 Multi-view Storage & Delivery

General testing methodology
Testing will be executed in a laboratory environment: Input and output devices will be simulated where necessary, standard video/audio and metadata streams will be used.

Except for programming unit tests, all test activities will be managed and executed by a dedicated test team.

Validation starts with the documentation of a high level test plan. The inputs for the test plan are the documented requirements and use cases. The test plan will describe the test scenarios to be executed, the data sets (including video, audio and metadata formats) to be considered. As well as load and stress tests based on the non-functional requirements. The test plan will be strongly risk-based, higher risk areas will be tested early in the cycle, more test resources will be allocated to higher risk areas.

Test scenarios will typically describe use case steps and expected results for each step. Video and audio quality will be validated by human inspection.

The test plan will also identify the test tools to be developed for simulating ingest and delivery-clients, and for automating load tests.

For each test run (typical when a new build of the solution is delivered), it will be documented what tests are executed, their execution state (success, failed …), as well a reference to the description of the defects if the test run is not successful.

Task specific testing methodology

The main points that will be tested are:

- The video quality (this will be tested by visual inspection and comparison between the original uncompressed stream and the ingested streams).
- The capability to ingest (and deliver on demand) metadata and those video feeds that are formatted (resolution and coding scheme) in a way that may not be played directly by the server.
- The correctness of the stored calibration data
- The conservation of the synchronization of the video and metadata feeds.
- All the normal operation of the server will be tested for non regression (clip, playlist & time line capabilities, instant replay, slow motion, SDTI video network, IP network, …)

The communication (control/command and data) between the server and the computing module ("Smooth view interpolation", "3D geometric & texture map recovery" and "3D Animated Web Player") will be tested in the following way. EVS will write and also provide to all the partners in the FINE project a SDK in C++ (for Windows) to:

- Request for a specific sequence (video or metadata) stored on the server.
- Send a specific sequence (video or metadata) to the server.
- Add or modify the calibration data of a sequence already stored on the server

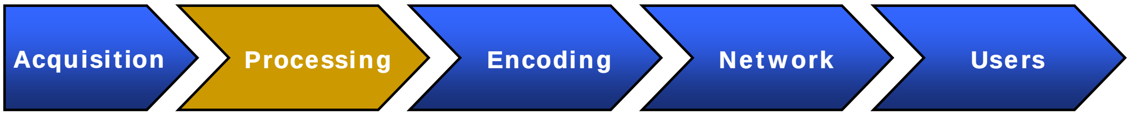
A test program (that uses the SDK) will be written to test these functionalities.

5.1.3. WP3T5 Multi-camera video homogenization

Several methodologies and metrics are going to be used to validate the homogenization of video cameras. One of the main areas of research that this task is focused on is colour matching from a multi-camera setup, several parameters will be evaluated; white balance, contrast, brightness, chromaticity, etc and from this analysis we will work with certain metrics to validate their uniformity, such as histogram matching by means of cumulative distribution functions (CDF), CDF differences, chrome and luminance difference.
Our multi-camera setups will also include a mix of interlaced and progressive camera feeds, and a few empiric metrics will be use to evaluate the success of the homogenization; the interlaced frames will be de-interlaced using several algorithms, and a series of visual test by side-by-side comparison, and image differences will be performed. Test scenes with fast moving cameras and objects will be used to exaggerate and better validate the interlaced vs. progressive differences.
The project FINE also contemplates scenarios where cameras with different resolutions will be combined; both temporal resolution (frames per second) or spatial (size of the camera sensor). A number of image processing operations like frame interpolation, motion compensation and/or

re-scaling with preserving aspect ratio will be performed and both subjective and objective visual quality control tests will be perform to assure correct homogenization of all video feeds.

5.2. Processing



Processing will be considered in both professional and home user scenarios.
These tasks are the two pillars of this project, 3D Motion Capture is the 3D virtual content delivered to end user, where this user could be able to choose any view angle for watching the specified content. Multiple Object tracking sends the position, direction and speed of any object to be rendered in the end user screen.
Smooth View Interpolation is able to generate any point of view (angle) from the real cameras source.

5.2.1.WP4T1 Smooth View Interpolation

The inputs to the algorithms to be developed in this task are:
- Synchronized video sequences from multiple points of view
- Calibration information
- Desired view location, orientation and virtual lens properties.

The output is a video sequence showing an interpolated view for the desired virtual camera. Two use cases are addressed:
- Plausible interpolation in real-time
- Quality interpolation in reasonable time

Algorithms are first tested in lab, using multi-view video sequences captured during FINE field tests (e.g. Stockholm Sep 2010, Genk Oct 2010). Tests include running time and a visual inspection assessing absence of objectionable image artefacts such as tears and flickering in the first use case, or quantitative comparison with an additional captured view not used for the interpolation in the second case.

Once assessed in the lab, algorithms can be provided as building “black boxes” for the FINE architecture, and evaluated in the end-to-end chain. This requires a checking whether communication is properly working (input from EVS server, see Task 2.1.2 testing methodology) and output to OBVan (WP6T2) or MultiView coder (WP5T1).

5.2.2.WP4T2 3D geometry and texture map recovery

This task is aimed at computing depths and a 3D reconstruction of the scene from the input video sequences acquired in WP3. This task is completely computational and does not require special equipment for the tests other than the input data from WP3 and the computer capabilities available within FINE (desktop computers equipped with a GPU and a multiGPU server). The goal of the test trials is threefold:

- To measure the efficiency (in terms of speed and quality) of the developed algorithms
- To assess how the different processes applied to the data in WP3 affect the results attained in this task.
- To study the suitability of the results for the project workflow.

In order to measure the quality of the results, we will perform our initial tests using synthetic data for which a ground truth is available. This will give an idea of the capabilities of the algorithms in optimal conditions. This information will be used for D4.2 (M12). As in M12 it is expected that the first prototypes for the camera system will be available, from M12 on we will test the algorithms for depth computation and 3D reconstruction using data coming from the capture system. By comparing the results obtained with synthetic data with the results obtained with the camera system we can obtain information about how the algorithms behave with respect to the characteristics of the input data. In particular we will test the influence of the camera setup (WP3T1), calibration (WP3T2 and WP3T3) and homogenization of the camera system (WP3T5). We will also use information coming from test results corresponding to other tasks (mainly WP4T1, WP4T2 and WP4T4), that use our results as input data, so that the suitability of the obtained 3D reconstructions inside the project workflow can be tested. The results will be reported in D4.6 (M30).

5.2.3.WP4T3 3D Motion Capture

This task considers the problem of interactive, semi automatic and automatic motion capture of human actions in real world sports events. We will primarily consider non real time processing but with the aim of presenting 3D animations of actions in connection with the ongoing game. Focus will be on single as well as multiple players in team sports such as football. We will build on and extend previous work performed at KTH. The main challenge in this respect is the relatively lower video quality depicting individual player's actions. For this purpose we will consider specially dedicate 2 or three HD camera systems that track individual players. The possibility of automating this tracking using the existing TRAC technology will be investigated. 3D animation in the form of stick figures depicting essential body locations will be generated indexing into libraries of action specific animations. Texture mapping of these will later be provided by BIT.

This task runs from month 3 to month 36 and there are two deliverables associated:

- D4.1 Prototype of action classification, month 12
- D4.8 Mapping sports video scenes to 3D animation and improved tracking taking into a account knowledge of body parts, month 32

5.2.4.WP4T4 Multiple Object Tracking

This task considers the problem of object tracking in a sports scene. In this task, TRAC will advance the state-of-the-art in real-time and near real-time object tracking technology, integrate tracking technology as a cue to other subsystems in the project, and recover body parts using FINE's 3D recovery methods. First, TRAC's existing tracking system will be enhanced by exploiting the high-quality video data from the HD camera clusters in the FINE camera architecture. Second, this task will study improvements available by closing the loop between object tracking and 3D reconstruction (WP4T2), e.g. exploiting information concerning which 3D voxels belong to which objects. Recovering body parts is a key part of TRAC's contribution in this WP. Finally, TRAC will act as an integrator to provide the other partners with tracking data (position and identities of the players, referee and the ball at 25 fps) from real matches. TRAC will provide an API to access both offline and live tracking data.

This task runs from M10 to M32. There are two deliverables associated with this task D4.4 and D4.8. The necessary equipment, location, dates and human resources required for testing the content of these deliverables is outlined below.
D4.4 (M24) a prototype of improved tracking using multi-view cameras.

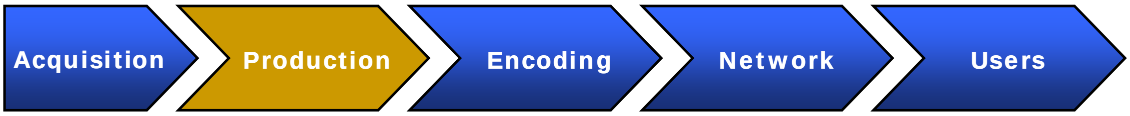
- Equipment: Complete FINE camera architecture.
- Location: a football game at a stadium.
- Dates: The prototype will be available for testing from M24 onwards.

- Human resources: A team of at least three trained operators to set-up and run the equipment.

D4.8 (M32) a mapping of sports video scenes to 3D animation and improved tracking taking into account knowledge of body parts.

- Equipment: Complete FINE camera architecture.
- Location: a football game at a stadium.
- Dates: The prototype will be available for testing from M32 onwards.
- Human resources: A team of at least three trained operators to set-up and run the equipment.

5.3. Production



Production will be considered only in professional user scenarios.
In this task the professional user is able to freely generate any point of view scene based on all the multi-view captured streams.

5.3.1. WP6T2 3D OBVAN Video Server

General testing methodology
Testing of the Video Server itself will be executed in a laboratory environment: Input and output devices will be simulated where necessary, standard video/audio and metadata streams will be used.

Except for programming unit tests, all test activities will be managed and executed by a dedicated test team.

Validation starts with the documentation of a high level test plan. The inputs for the test plan are the documented requirements and use cases. The test plan will describe the test scenarios to be executed, the data sets (including video, audio and metadata formats) to be considered. As well as load and stress tests based on the non-functional requirements. The test plan will be strongly risk-based, higher risk areas will be tested early in the cycle, more test resources will be allocated to higher risk areas.

Test scenarios will typically describe use case steps and expected results for each step. Video and audio quality will be validated by human inspection.

The test plan will also identify the test tools to be developed for simulating ingest and delivery-clients, and for automating load tests.

For each test run (typical when a new build of the solution is delivered), it will be documented what tests are executed, their execution state (success, failed …), as well a reference to the description of the defects if the test run is not successful.

Task specific testing methodology
All the functions of the interface will be tested individually. That include:

- The creation of a virtual viewpoint,
- The management of the viewpoint list,

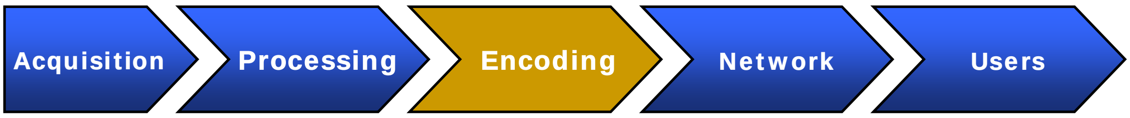
- The input of the parameters necessary to request a free viewpoint clip.
- The capability to modify and to play the requested free viewpoint playlist.
- The setup/configuration interface of the computing modules

The communication (control/command and data) between the server and the computing module ("Smooth view interpolation", "3D geometric & texture map recovery" and "3D Animated Web Player") will be tested in the following way. EVS will write and also provide to all the partners in the FINE project a SDK in C++ (for Windows) to:

- Receive a request from the server to compute a specific free viewpoint sequence with a given viewpoint and a given set of parameters.
- Send a specific free viewpoint sequence (video or metadata) to the server.

A test program (that uses the SDK) will be written to test these functionalities.

5.4. Coding



Coding will be considered in both professional and home user scenarios.
Coding is split in two tasks "Multi-view coding" for the photorealistic rendering and "Geometry and Texture coding" to provide a 3D rendering.

5.4.1. WP5T1 Multi view coding

Application response time must be evaluated. Switching time between consecutive streams must be small enough to achieve a real time 3D multi-view effect.
Encoding should take this into account when generating video streams.

The following methodology aims to test the developed multiview video coding algorithm that supports free-viewpoint experience of live events. The encoding algorithm requires as input synchronized data streams that represent texture, depth, and tracking information. The output will be a coded bitstream that will be distributed by a streaming server. The decoding algorithm requires as input the coded bitstream and the free-viewpoint user interaction. The output is the rendered free-view video, possibly complemented by feedback messages sent to the streaming server. This methodology tests the algorithms in the following deliverables:

D5.1 (M12) ROI multiview coding deliverable: dynamic content
D5.5 (M24) ROI multiview coding deliverable: background scene
D5.9 (M34) ROI multiview coding deliverable: final version

1. The algorithm satisfies functional and network requirements of D2.2.
2. The algorithm is evaluated for given test video data, e.g., video streams from an artificial 3D soccer environment, or recorded video streams from stadium cameras.
3. The quality of the rendered views is measured. To assess the quality of the rendered views for an algorithm, predefined viewpoint trajectories are chosen.
4. The bitrate of the coded bitstream is measured. For predefined viewpoint trajectories, the trade-off between rendered quality and bitrate is assessed.
5. The system response time between free-view user interaction and rendering output is considered. For given free-view user interactions and system response times, the trade-off between rendered quality and bitrate is assessed.

6. Depending on the quality of the provided network service, network packet losses may be considered. For given network conditions, the trade-off between rendered quality, bitrate, and system response time is assessed.

5.4.2. WP5T2 Geometry and texture coding

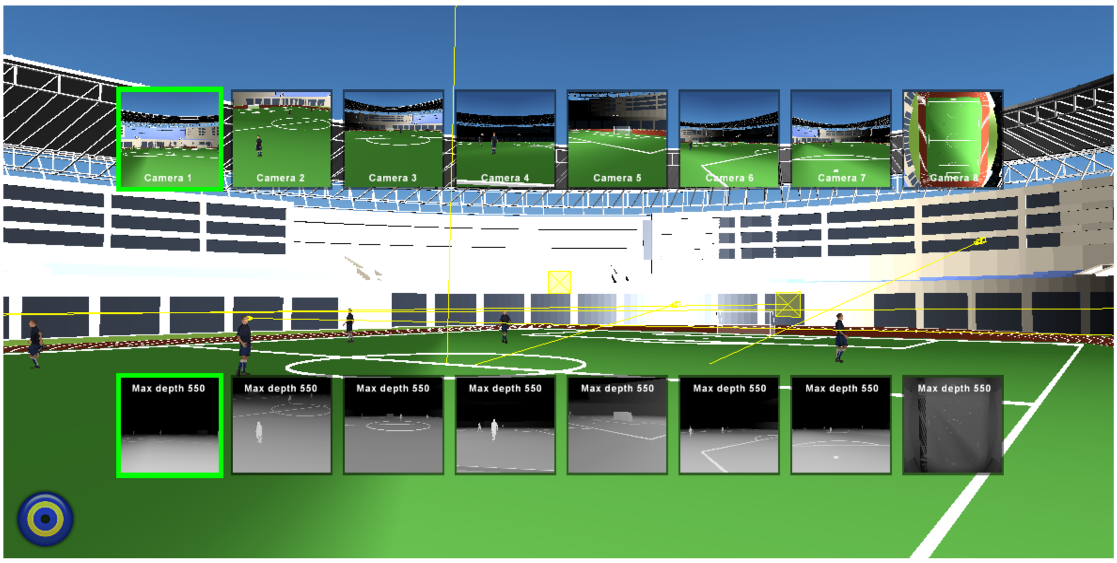
The following methodology addresses conformance of binary encoded geometry and texture streams (FINE-GTS) and conformance of generators and viewers that support this binary stream encoding.

A binary encoding generator conforms to FINE-GTS, if all files that are generated are syntactically correct.
A viewer, both photorealistic and 3D rendering conforms to FINE-GTS if:

a. It is able to decode any binary encoded file/stream that conforms to FINE-GTS.
b. It presents the graphical and animation characteristics of the functionality in any binary encoded file that conforms to FINE-GTS.
c. It correctly handles animation, user interaction and generation of events.
d. It satisfies the requirements in regards to platform and minimum support requirements.
e. It satisfies the general requirements of Del 2.2 from WP 2.

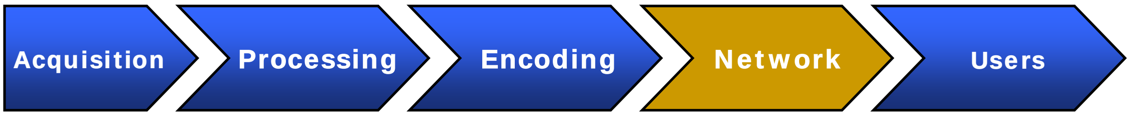
In order to validate the encoding and decoding an artificial test system will be developed throughout the project that can be used to produce geometry and texture coding from an artificial 3D soccer environment with several virtual cameras. It is important to note that many cameras can be set individually to produce appropriate streams of the scene. This data can be saved currently in DDS texture format and can be used for further texture coding experiments.

Fine Artificial Testbed



Availability dates:
Month 12: Del 5.3. Geometry and Texture Coding
Month 24: Del 5.6. Geometry and Texture Coding

5.5. Network



Network will be considered in both professional and home user scenarios.
But network architecture and features will be different for each of the scenarios (Professional and Home users). In professional situation only new generation network is considered but in Home users situation the CDN is also added to that network.

5.5.1. WP5T3 Network distribution

This aims to set out the scenarios for the proof-of-concept trials and give evaluation procedures to be used to assess the usability, suitability and performance of the technologies.

NGN network will be considered as for professional user scenarios and also for home user scenarios. NGN will be focused in professional scenarios due to the CDN constraints, that makes CDN much more restricted than NGN in the case of end user scenarios.

NGN will be tested on technology layer and also at transport protocol layer. Tests to be done will include checking of interfaces and working functionalities, as redundancy and protections in case of failure in order to guarantee the quality of service for the transmission.

In addition, to be able to transmit real time services through the network is very important to check QoS parameters of this network:

- **Bandwidth:** is measured as the average number of bits per second that can travel successfully through the network.
  All the information delivered by the TX server can be transmitted to the users?

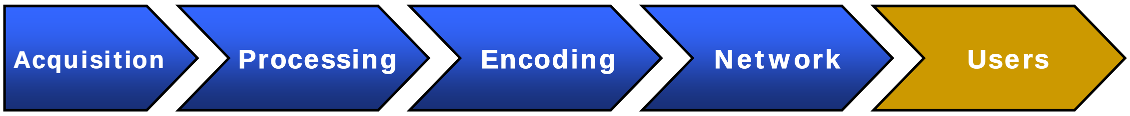
- **End-to-end Delay or latency:** is the average time it takes for a network packet to traverse the network from one endpoint to the other
  There is a good user-perceived response time?

- **Jitter:** is the variation in the end-to-end delay of sequential packets. Large jitter values may cause network data packets to arrive in the wrong sequence or not arrive when their video payload is needed, degrading the video quality.
  Can the receiver application manage the variation arrival time of the packets?

- **Packet loss rate:** is measured as the percent of transmitted packets that never reach the intended destination. Transmitted packets may be lost for several reasons, but the primary cause is due to congestion in the network routers. When too many packets are simultaneously sent to a router, it will simply discard some packets, assuming that the application that sent the packet will retransmit it.
  As retransmissions have no sense in real time services, packet loss means information loss.

There are some mechanisms to provide the QoS required for the video service (network provisioning, queuing and classifying) in order to get that real-time video delivering systems work properly: bandwidth should be as large as possible while latency and jitter are minimized. And packet loss must be zero. This requirement of no packet loss means that network distribution links cannot be oversubscribed, but this cannot be controlled in Internet that is used in the end user scenario!!

5.6. Exhibition



Exhibition will be considered in both professional and home user scenarios.
This process covers the full FINE chain, from the acquisition part to the end user experience.
In case of Professional scenario the end user experience will be performed as 2D content through the broadcast network and through the CDN network. In case of Home user scenario the end user experience will be performed through the FINE free view point video player.

Compatibility between different data and video streams must be evaluated. End user application must work with all different streams provided by the FINE developed devices.

5.6.1. WP5T3 Network distribution

For professional user scenario, this task provides the broadcast network that delivers the produced 2D scene to end users. This 2D scene is the result of the 3D OB Van production process developed by WP6T2.
End user application must be compatible with the received stream but only one stream should be delivered at the same time and the user application should be able to switch among different streams in a remote way Tests for video and audio quality should be done,

For the broadcast scenario, these tests must include network configuration for an optimal capacity and errorless transmission (*Bandwidth, Content format, content encapsulation and user application should be compatible).*

- Must check that the receiver is DVB fully compliant in order to deliver video, audio and data.in a correct way.

- For the IP network scenario, tests must include CDN behaviour (delay, jitter, packet loss), error correction and packet retransmission.

5.6.2. WP6T1 Free view-point video player

For home user scenario, this task will provide the receiver application to feature the FINE multi view point (3D) experience. Both Photorealistic and 3D Rendering are considered in this task.

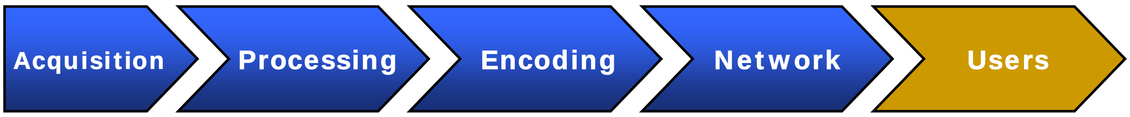
The following methodology addresses conformance of decoding and conformance of viewers that support the free viewpoint video coding (FINE-FVV). A viewer that supports the decoding, both photorealistic and 3D rendering conforms if:

a. It is able to decode any binary encoded file that conforms to FINE-FVV.
b. It presents the graphical and audio characteristics of the functionality in any binary encoded file that conforms to FINE-FVV.
c. It correctly handles user interaction, ie- viewpoint changes.
d. It satisfies the requirements of FINE-FVV in regard to platform and minimum support requirements.
e. It satisfies the general requirements of Del 2.2 from WP 2.

In order to validate the encoding and decoding an artificial test system will be developed throughout the project that can be used to produce depth image sequences, full video image sequences from an artificial 3D soccer environment with several virtual cameras. It is important to note that many cameras can be set individually to produce appropriate streams of the scene. The testbed will allow to playback and switch videos from different point of view.

Availability dates:
D.6.2 Free-viewpoint video player first version M 12
D.6.3 Free-viewpoint video player second version M 24
D.6.5 Free-viewpoint video player third version M 30

5.7. End to end tests



End to end tests including all prototypes will be carried out as described in the schedule. Both professional and home user scenarios will be tested from the end user side. Some test bed periods have been designed to evaluate the interoperability between all prototypes and to be able to fix whatever is wrong for the project to work; these have been split in lab tests, field tests and user satisfaction.
A window period has been defined for each end to end test, but tests can be done in less time than scheduled.

Lab tests: To check every technical part of the FINE chain separately or with other prototypes.

Field tests: To check the FINE chain functionality in a real environment joining all prototypes and testing all scenarios.

User Satisfaction: Throughout questionnaires, forms, and interviews checking the FINE users satisfaction.

6. Conclusions

In this initial phase of the project this document aims to be a guide for evaluating the project evolution and assure a proper development.

As a result of the analysis of the scheduled deliverables dates we have identified some issues to take into account.

- Regarding the deliverable D7.2 Proff-of-concept trials report its deadline is M24 just after the 1st lab testbed when many of the prototypes will be delivered. As we suppose that all partners will use the maximum time to test and finish their prototypes, we suggest moving forward this report one or two months in order to have enough time to test the complete chain (with all the devices put together) and to compile the final report.

- In a similar manner, at the end of the project we have left one month after the testbed window for the final report compilation (D7.4 Report on final professional and home user experimental production) and for project exploitation and dissemination (D8.4 Dissemination and publication of results and D8.5 Final Exploitation plan).