ITEA Office
High Tech Campus 69 - 3
5656 AG Eindhoven
The Netherlands

T + 31 88 003 6136
E info@itea3.org
W www.itea3.org

ITEA 3 is a EUREKA strategic ICT cluster programme

D3.1.1 – Application Design – year 1

MOS2S

Media Orchestration from Screen to Screen

FINAL VERSION

Edited by: J. Belpaire (Kiswe) and O.A. Niamut (TNO)
Contributions from:

- Rolf Biesbroek, Dolf Schinkel (KPN), Cyril Rutten (GameOn)
- Karim DahDah, Luk Overmeire (VRT), Tom Cuypers (Kiswe)
- Seong Yong Lim (ETRI)

Version: v03
Date: 09-11-2017
Delivery date: 10-11-2017

Project key data

ACRONYM and full-length title

15022	MOS2S
Program Call	ITEA 3 Call 2
Full-length Title	Media Orchestration - Sensor to Screen
Roadmap Challenge	Urbanisation

Description

Novel and ubiquitous consumer-priced audiovisual sensors and data in particular, represent an important aspect of the Smart City environment, enabling a variety of applications for citizen information, participation, entertainment, experience, safety and security. Every user becomes a potential source of information, either directly or through social media buzz and its discovery. Audiovisual media provide citizens with Smart City data readily accessible to human senses. With the MOS2S project (Media Orchestration from Sensor to Screen), an international consortium of partners will develop and test audiovisual Smart City technologies and solutions in the context of citizen needs, and embed these solutions within the Smart City Playground.

Project duration & size

Size	Effort: 133.67 PY	Costs: 13.9 M€
Time frame	Start: 2016-10-1	End: 2019-09-30 (37 months)

Coordinator

Netherlands	TNO
Type	Research Institute
Contact Person	Gjalt Loots
Email Address	gjalt.loots@tno.nl

Consortium

Belgium	Nokia, iMinds, Kiswe Mobile, VRT*
Korea, Republic of	ETRI*, Mooovr
Netherlands	Amsterdam ArenA*, Bosch Security Systems B.V., Game On, Inmotio Object Tracking BV, Koninklijke KPN NV, TNO
Turkey	Bor Software inc.*, DİA Yazilim San. ve Tic. A.Ş., KoçSistem, TMOB BİLİŞİM

Table of Contents

List of Figures

Project acronyms

3DoF	3 degrees of freedom
BE	Belgium
CPA	Conformance Points A
CPB	Conformance Points B
DVB	Digital Video Broadcasting
IM	Instant Messaging
IT	Italy
KR	Republic of Korea
MOS2S	Media Orchestration Sensor To Screen
NL	(The) Netherlands
OB	Outside Broadcasting
PTZ	Pan-Tilt-Zoom
TR	Turkey
UHD	Ultra-High Definition
VR	Virtual Reality

1. Introduction

In Deliverable D3.1.1, the MOS2S consortium partners provide an overview of the application designs that have been demonstrated at the first year demonstration event (IBC2017). D3.1.1 is the first deliverable from WP3, in which we outline the design specifications for dedicated applications running on top of the platforms supporting the demonstrators.

In WP3, the platform functionality and generic components researched and prototyped in WP2 need to be combined with applications and use case specific functionality that will be designed and integrated in this WP. The resulting applications and the underlying platforms will be used to drive the MOS2S demonstrators in the respective domains.

D3.1.1 is primarily a software deliverable. This document provides descriptions of the application software, as developed by partners and demonstrated at IBC2017. It is an outcome of T3.1 (Application design).

2. NL Application Designs

2.1. Companion Screen application

The Companion Screen application is an advanced video player for smartphones and tablets that enables professional sports professionals and home viewers to interact with (live) video recordings in an intuitive way. Several important add-ons to the app have been developed in recent months. All code has been written in modern C++ and OpenGL GLSL for cross-platform support using the XCode or AppCode IDE on a MacOS platform. All code is stored in a private GitHub repository that allows sharing with partners at a later stage.

Stitching

Video stitching is necessary to create a single wide-angle, high-resolution view (panorama) from multiple individual camera streams. A wide-angle view offers the viewer both overview (when zoomed out) and detail (when zoomed in). The higher the resolution, the more detail that will be available in the scene. Stitching generally requires several operations: (1) lens undistortion, (2) perspective transform, (3) affine transform, (4) colour equalisation and optionally, (5) warping and (6) blending. In a perfect stitch, the viewer cannot detect the stitch seams and the panorama looks as if it was recorded by a single camera. In addition, both the parallax effect and time discrepancies between cameras must be minimised in order to prevent unwanted artefacts in the stitched view.

Traditionally, stitching is performed as a server-side process due to the fact that it is a computationally intensive task, especially with high-resolution videos. Moving around millions of pixels requires considerable CPU time. Luckily, using hardware acceleration, this task can be offloaded to graphics hardware. This allows us to perform the stitching on the client side, inside the app. The Companion Screen app is able to display a stitch composed of two 12MP video streams, resulting in a 24MP panorama.

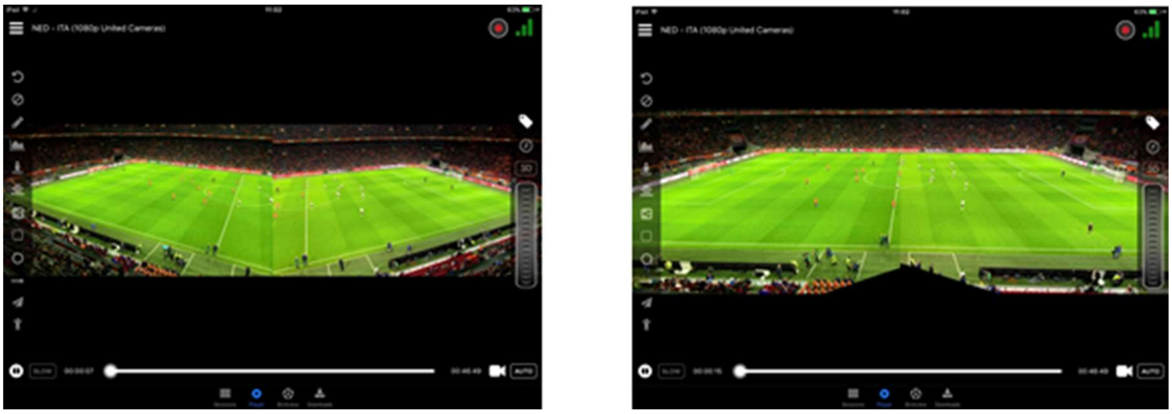

NED - ITA (1080p United Cameras)
NED - ITA (1080p United Cameras)

Figure 1: (left) Unstitched panorama; (right) Stitched panorama.

Virtual PTZ

When zooming and panning into a stitched view (also called “virtual PTZ”), it may occur that the undistortion operation and perspective transform have caused a too large deviation from what the user would consider a ‘normal’ view. Care must be taken to crop out the subregion in such a way, that the result looks like if it would have been captured by a moving TV camera. It is critical that lines that should be straight are indeed straight and that object that stand upright appear indeed upright. For this, the app needs to be able to perform a sophisticated 3D transform.

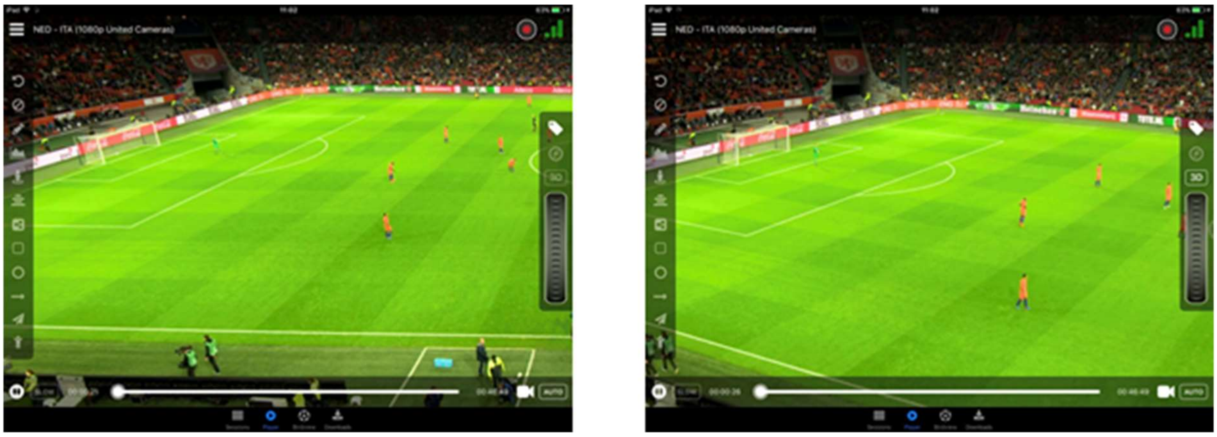

NED - ITA (1080p United Cameras)
3D
AUTO
NED - ITA (1080p United Cameras)
3D
AUTO

Figure 2: (left) Zoomed in with no 3D transform applied; (right) Zoomed in with the 3D transform applied.

Graphical overlays

For the app to be truly interactive, the user needs tools to annotate interesting scenes within the video content. For this, graphical overlays are critical. Using overlays, the user is able to highlight areas, players, movements and other details that are of part of the game. The following overlays were added to the app: (1) polygon, (2) (ballistic) arrow, (3) player move, (4) rectangle, (5) circle, (6) player marker, (7) distance line and (8) free sketch.

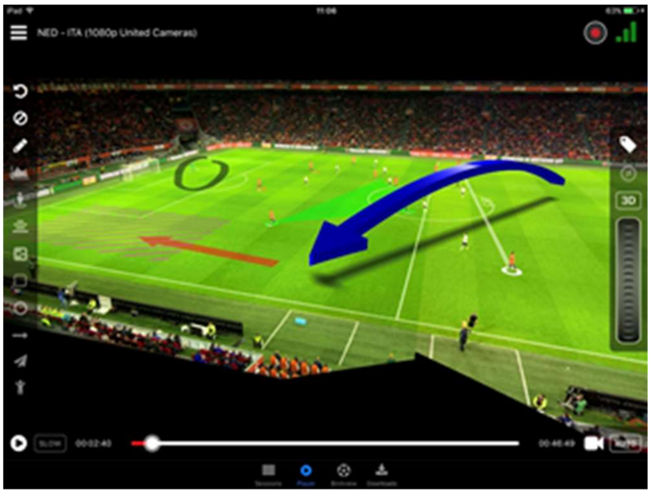


Figure 3: examples of graphical overlays.

2.2. TV application

Metrological built the Proof of Concept (POC) for the MOS2S application for KPN during the KPN Hackathon 2017. The goal of this POC was to show all parties involved what is possible on the current STB's from KPN with respect to the football manager and it's requirements. The goal was to show the end-users of KPN extra information on screen during a football match. When opening the app the user is presented timelines and game-statistics that will enrich the information that they normally have. Player statistics are added to keep track on the users that are on the field and a social experience is added to create "rooms" where people can watch and comment the match together with friends and family. At this moment Metrological is updating the POC to make sure it's connected to the API's and video of GameOn to show all data and also improve the design to prepare it for the upcoming demo's.

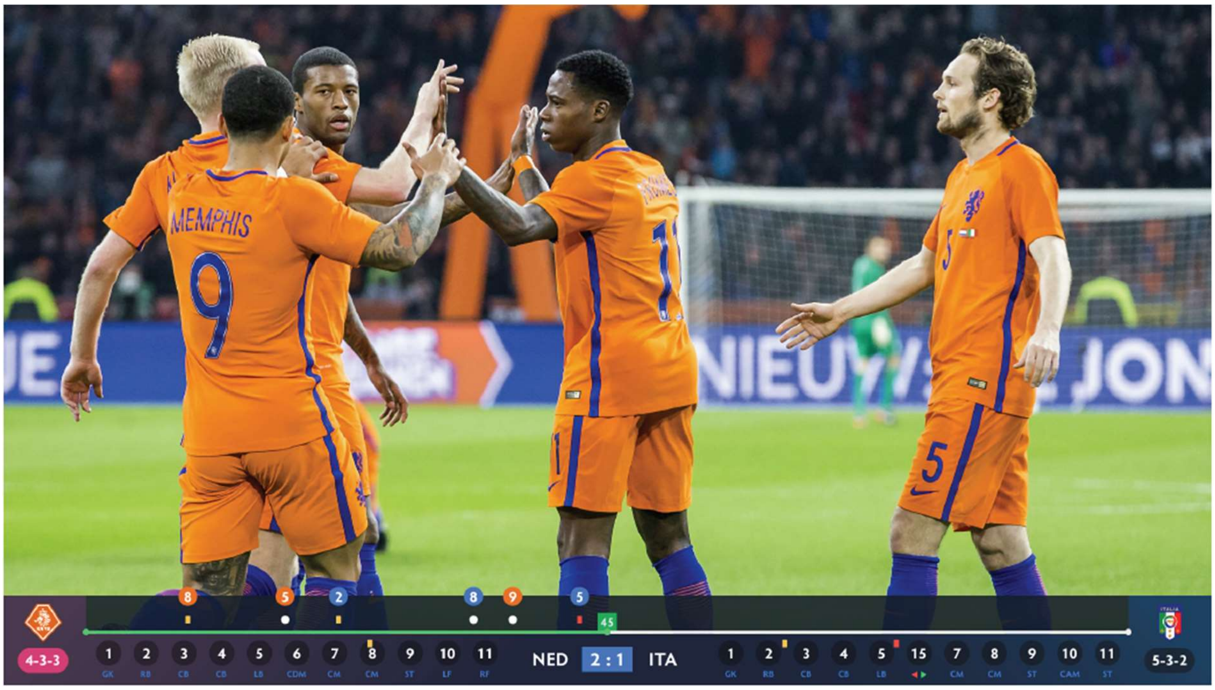
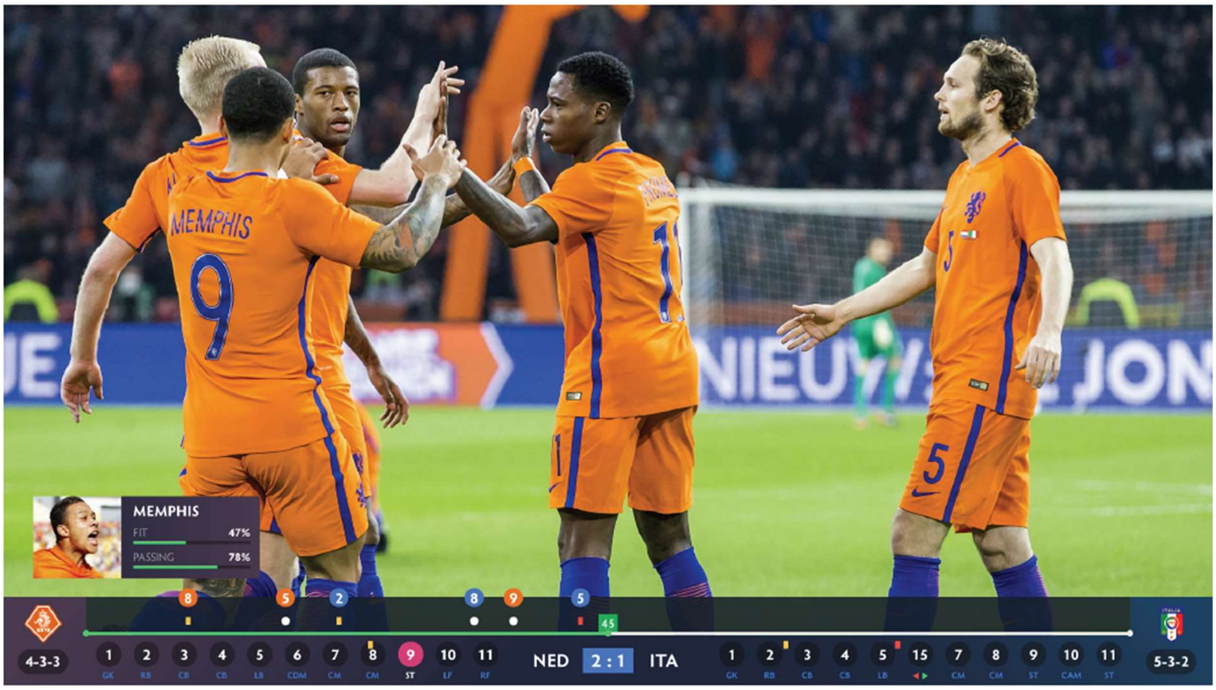


Figure 4: timeline and statistics navigation.

Timeline and Statistics Navigation

On screen you will see both teams that play the match, including the player lineup per team. The user will see the match statistics: formation, score and timeline. Important actions during the match will be shown to the user: yellow cars, red cards, substitutions, missed penalties etc. Another feature of the app is the real-time map. This map will display all the players that are on the field, drawn on a small map that the user is able to open or close himself. This will give the users more insight in the tactics and positioning of the teams. The user is also able to open a chat and video-conference window. By using these features the users will be able to join a "room" together with their friends and/or family to enjoy the match experience and discuss what's happening on the field in real-time. The video-conference option gives the users the option to see (not hear) each other to give it an extra dimension of the social experience. By navigating through the player lineup the user is able to see the statistics of that player, meaning he can see all the data that is measured and is available within the APIs.


MEMPHIS
FIT 47%
PASSING 78%

4-3-3	1	2	3	4	5	6	7	8	9	10	11	NED	2 : 1	ITA	1	2	3	4	5	15	7	8	9	10	11	5-3-2
	GK	RB	CB	CB	LB	CDM	CM	CM	ST	LF	RF				GK	RB	CB	CB	LB		CM	CM	ST	CAM	ST	

Figure 5: data visualization during timeline navigation.

Technology

The current application is built completely in the Metrological Application Framework (MAF) and is built in the Javascript language. By connecting to the API's from GameOn the data is retrieved and will be shown in real-time. The pairing of the 2nd screen devices is also done via the Metrological Application Framework and its capabilities to connect the 2nd screen devices via WebSockets through a QR-code pairing mechanism. This way the chat and video functionalities are possible. Availability The application is available on the Metrological Backend, and can be installed on any KPN App Store when needed. These App Stores can be configured on the KPN STB's by changing 1 file on the local filesystem.

3. BE Application Descriptions

3.1. Editorial tool for professional users

This is the tool where editors can interact with the end users and where they can select and publish content which are shared from the end users. It's also the place where they can initiate polls and set up automated bots which can send messages at a specific moment in time or who can respond to common questions.

The editorial dashboard is a web app, available at a public URL. It is also developed in Angular and Ionic. This enables us to build the software to a mobile app if it should be interesting for editors.

3.2. Mobile app for end users

The mobile app is the tool that's being used by the end users. It's the place where end users can share photos and videos with the editorial team. The end user can also interact with the editors and participate in polls. Bots can automatically help users with common questions, lowering the management load for editors.

The mobile app is built with web technology: Angular & Ionic as framework, combined with SASS based UI refinements which enables us to create a somewhat modified design for each event. Another software stack we use is the open source project Cordova. With it we can build the web based software to mobile applications which are published to the Google Play Store and Apple App Store through the IDE's which are respectively Android Studio and Xcode.

3.3. Social capturing/discovery tool

The social capturing/discovery tool is a set of software components that enables the editor to search the social web for interesting content on a specific topic. The tools are being built on top of Firebase, Node.js, ElasticSearch, Kibana and the API's from Instagram and Twitter.

3.4. MOS2S Extensions

In this project we build on top of the results made from the toolset for crowd contribution and interaction that we have built in the ICoSOLE project, called "Wall of Moments". This toolset consists of an end-user app to contribute content and get updates from the editorial team ('Moments'), and an editorial toolset to request content from end-users, to segment and interact with them ('Trademark') and an app to display content in an interactive way during an event ('the Wall'). ICoSOLE was a -European Union's Seventh Framework Programme (FP7/2007-2013) project under grant agreement n° 610370. More info about that project can be found at the website http://icosole.eu/.

Our goal in MOS2S is to enhance this toolset, and more specifically, expand the editorial app. We rebranded it to ‘Switchboard’ and plan to add location-based functionalities, integration with the Kiswe Cloud Clipping service and a more advanced tagging system to segment users and content. The refactored software is also built as a modular system. This enables us to prototype new insights faster and make pieces of software or services available to partners.

Short overview of the software platform

The platform consists of 4 core parts: a mobile app, an editorial dashboard, several backend services and a social capturing/discovery tool.

Those four core parts have been extended with

- a tool for the creation and publishing of a living and up-to-date long form article
- a digital signage tool, called “the Wall”, to publish selected content on big screens at an event
- an automated service for the creation of personalized aftermovies

3.5. Iterative updates during test cases

In this section we want to outline software updates and refinements we did based on learnings during test cases.

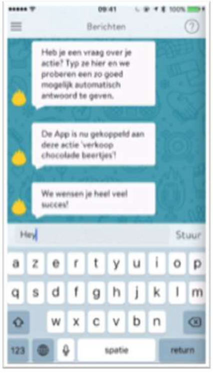
Music for Life (November/December 2016)

This was the first test case with the refactored version of the apps used in ICoSOLE. It included a modular design, tagging system and a first version of the new editorial tool ‘Switchboard’.

Most important new features include:

- Testing co-creation and interaction between production team and event organizers: contributed content was shared on third-party platforms for the first time (user generated photos & videos have been used on Facebook and on television as an overlay)

- First test of using a chat interface as the main interaction between the end user and the editors

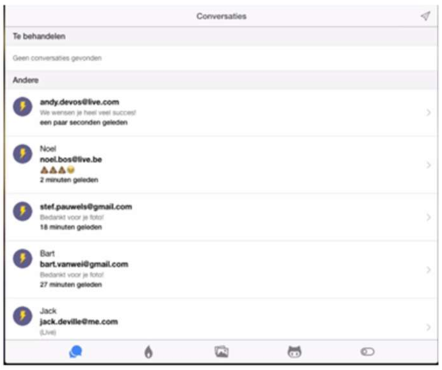


Heb je een vraag over je actie? Typ ze hier en we proberen een zo goed mogelijk automatisch antwoord te geven.

De App is nu gekoppeld aan deze actie 'verkoop chocolade beertjes'!

We wensen je heel veel succes!

Hey



Geen conversaties gevonden

andy.devos@live.com
We wensen je heel veel succes!
een paar seconden geleden

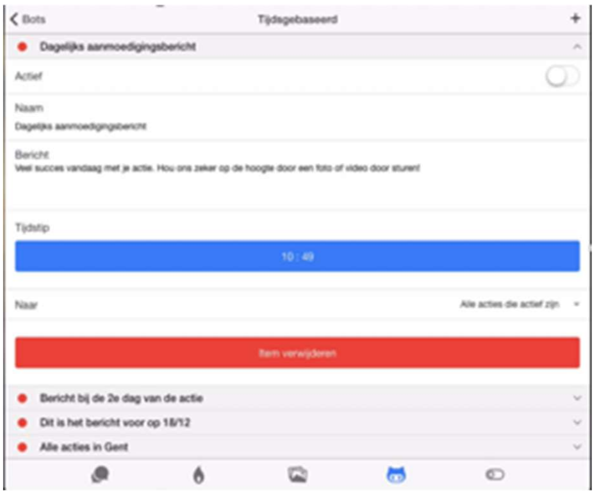
Noel
noel.bos@live.be
2 minuten geleden

stef.pauwels@gmail.com
Bedankt voor je foto!
18 minuten geleden

Bart
bart.vanwei@gmail.com
Bedankt voor je foto!
27 minuten geleden

Jack
jack.deville@me.com
(Live)

- Introduction of the concept of chat bots: end-users were able to ask questions and got an automatic reply (if found). We also introduced time-based bots, which sent out personalized messages for every user on certain points in time.



● Dagelijks aanmoedigingsbericht

Actief

Naam
Dagelijks aanmoedigingsbericht

Bericht
Veel succes vandaag met je actie. Hou ons zeker op de hoogte door een foto of video door sturen!

Tijdstip
10 : 49

Naar
Alle acties die actief zijn

● Bericht bij de 2e dag van de actie

● Dit is het bericht voor op 18/12

● Alle acties in Gent

- End-users received a personalized aftermovie following the event, which consisted of their sent-in photos. This was achieved using an After Effects plugin and a rendering engine.

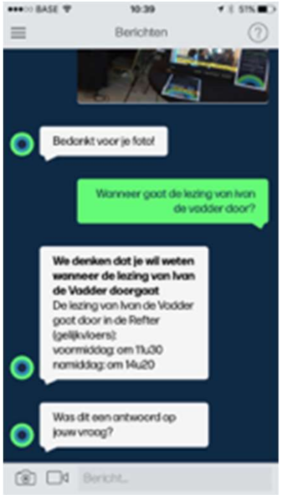
Het Vooruitzicht 2017 (January 21, 2017)

During this event, several simultaneous presentations of VRT News journalists and experts were held. During these presentations, end-users were able to send their questions to the speaker, as well as participate in polls. Lastly, 'the Wall' app was revived and made more interactive by adding beacons.

Added features:

- Asking questions: Switchboard was updated to display separate feeds of messages, based on tagging; these feeds were used to filter incoming messages for each speaker; moreover, important messages could be starred and filtered out

- Interactive polls: the editorial team was able to send out polls (yes/no, multiple choice or score) to (a segment) of users, and follow the results live in Switchboard.
- Curated generated photos and videos were shown on ‘the Wall’ (on a big LED screen); the app driving the screen was connected with beacons near the screen, and adapted what was shown based on people nearby.



Bedankt voor je foto!

Wanneer gaat de lezing van Ivan de vadder door?

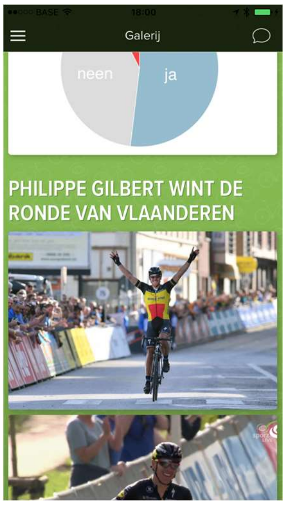
We denken dat je wil weten wanneer de lezing van Ivan de Vadder doorgaat
De lezing van Ivan de Vadder gaat door in de Refter (gelijkvloers):
voormiddag: om 11u30
namiddag: om 14u20

Was dit een antwoord op jouw vraag?

Ronde van Vlaanderen (April 2, 2017)

In partnership with Sporza, we created the app “Rondereporter”, which people could use to send in their best moments of the cycling race by using the app. The main objective was to test a new way of publishing user-generated content by using a longform article.

- The editorial team was able to curate the best content, and to accommodate it with a title and/or description and publish it to the article live.



PHILIPPE GILBERT WINT DE RONDE VAN VLAANDEREN

“Meld Nieuws” app & Trump’s visit to Belgium (May, 2017)

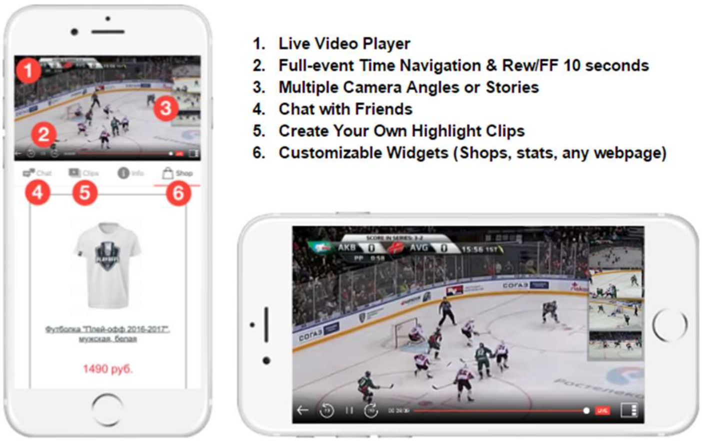
For one month, users were able to report news to the VRT news department, using a dedicated app. At the end of the testing period, Trump’s visit to Belgium was planned, and the (location based feature of the) app was used to locate people nearby the places Trump visited.

Enhanced and new features:

- Push notifications and polls: we wanted to test how to best handle co-creation by setting up several small experiments. These were led by a small professional editorial team inside the news department
  - Collecting local news, proposed by the test users (example: a raid happening in a city)
  - Opinions on news events (example: second hand sale successes)
  - Doing an experiment and submitting feedback (example: traffic experiment)
- Location based services: we included a map in ‘Switchboard’ which showed the location of the users. Based on their location, the team was able to coordinate the people that were using the app. Especially during president Trump’s visit we tested this by sending out 16 reporters in the field guided by an editorial team located at VRT.

3.6. Kiswe mobile application

The Kiswe app offers a multi-story interactive & social streaming experience. It is composed of the Kiswe advanced video player with up to four simultaneously serviced video streams and a selection of interactive widgets.



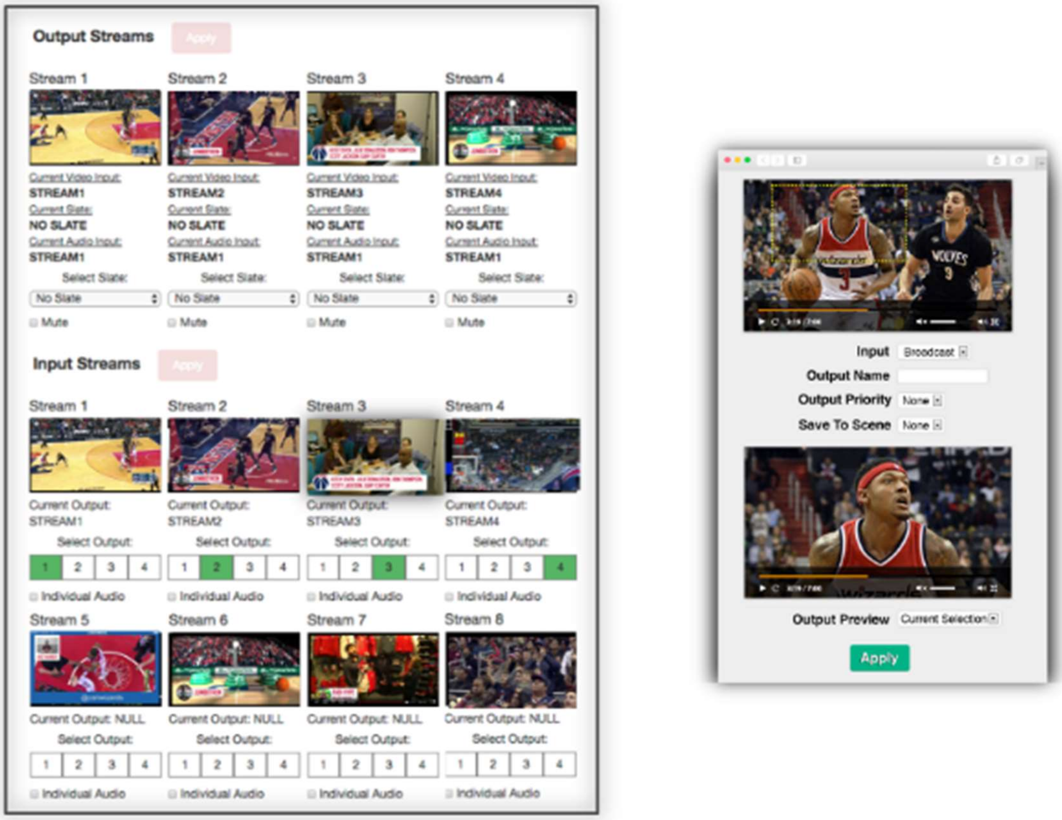
1. Live Video Player
2. Full-event Time Navigation & Rew/FF 10 seconds
3. Multiple Camera Angles or Stories
4. Chat with Friends
5. Create Your Own Highlight Clips
6. Customizable Widgets (Shops, stats, any webpage)

Figure 6: Kiswe mobile application.

This unique multi-stream video player offers the user a series of time navigation features such as the time scrollbar to any point in the event, play/pause video, jump forward/rewind by 10 seconds, and jump to live. You can also toggle the multiple picture-in-picture videos (pips) from the multiple storylines on and off. These streams offer multiple different stories and/or camera angles to the users, and are equipped with Kiswe's ultra-fast interactive switching technology.

3.7. Kiswe Production Studio application

The Kiswe Production Studio toolkit (KPS), provides simplified yet powerful cloud-based digital production features. It gives content creators the tools to craft multiple storylines around a live event from your web browser.



Output Streams Apply

Stream 1	Stream 2	Stream 3	Stream 4
Current Video Input: STREAM1	Current Video Input: STREAM2	Current Video Input: STREAM3	Current Video Input: STREAM4
Current Slate: NO SLATE	Current Slate: NO SLATE	Current Slate: NO SLATE	Current Slate: NO SLATE
Current Audio Input: STREAM1	Current Audio Input: STREAM1	Current Audio Input: STREAM1	Current Audio Input: STREAM1
Select Slate: No Slate	Select Slate: No Slate	Select Slate: No Slate	Select Slate: No Slate
□ Mute	□ Mute	□ Mute	□ Mute

Input Streams Apply

Stream 1	Stream 2	Stream 3	Stream 4
Current Output: STREAM1	Current Output: STREAM2	Current Output: STREAM3	Current Output: STREAM4
Select Output: [✓ 1] 2 3 4	Select Output: 1 [✓ 2] 3 4	Select Output: 1 2 [✓ 3] 4	Select Output: 1 2 3 [✓ 4]
□ Individual Audio	□ Individual Audio	□ Individual Audio	□ Individual Audio

Stream 5	Stream 6	Stream 7	Stream 8
Current Output: NULL	Current Output: NULL	Current Output: NULL	Current Output: NULL
Select Output: 1 2 3 4	Select Output: 1 2 3 4	Select Output: 1 2 3 4	Select Output: 1 2 3 4
□ Individual Audio	□ Individual Audio	□ Individual Audio	□ Individual Audio

Input: Broadcast
Output Name:
Output Priority: None
Save To Scene: None
Output Preview: Current Selection

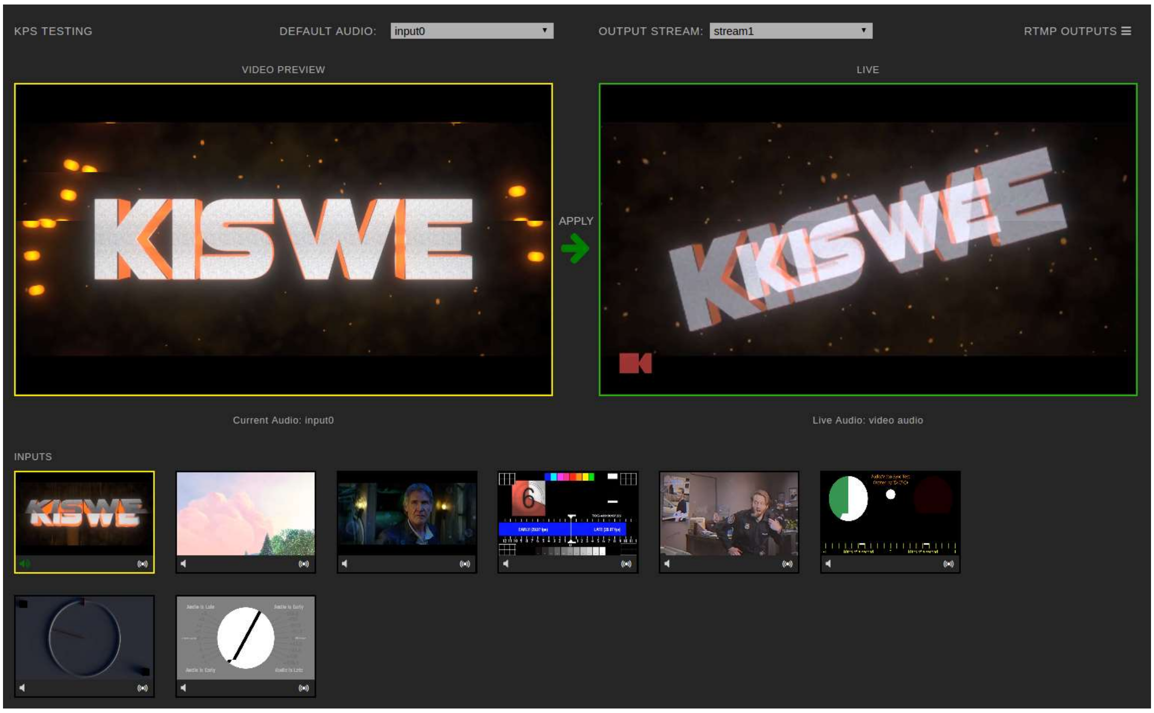
Apply

Figure 7: Kiswe production studio application.

You can switch between cameras (via an SDI input or RTMP stream link), add graphic slates, overlays and lower thirds, introduce VOD or mid-roll commercials on various streams, and adjust your audio levels across every stream through our simplified content management system and user interface.

3.8. UGC contributions in the KPS API

The Kiswe cloud is initially developed to handle incoming streams originating from a KVE accepting HD-SDi input signals from professional cameras. These are usually located at the customer location with a predictable network and decent amount of upload. This makes the ingest a very stable process as we know the hardware and network environment. Working with UGC (User Generated Content) video contributions coming from mobile devices we have to cope with additional challenges to make a stable stream. In the context of UGC, these streams can come and go at any time. The KPS API is an interface to a part of our back-end which allows any (third) party to request incoming feeds and manipulate them. We expanded this interface and backend to work with UGC.



KPS TESTING

DEFAULT AUDIO: input0

OUTPUT STREAM: stream1

Current Audio: input0

Live Audio: video audio

Figure 8: Kiswe interface.

Challenges

As a first step we need to know which streaming devices are sending or stopping a live stream to our back-end for a given event. We want to handle these streams in the back-end as quick as possible and make them available through the KPS API on the fly.

Second, as explained in the goals, the margin of error has been increased when using mobile devices as incoming streams to the back-end. E.g. the library used to create an rtmp stream from the camera on an iOS device does not generate a constant flow of video or audio. Depending on fluctuating network bandwidth it increases and decreases the jitter buffer on the device and therefore video and audio packages received in the back-end are coming in bursts. We also notice clock drift between the iOS device and the back-end that requires to be inverted.

Solution

Incoming connections from a mobile device or KVE are sent to an rtmp server in the Kiswe Cloud. This service is able to capture incoming streams and distribute them to other devices or services. In order for us to handle new incoming connections and disappearing connections we had to create a plugin in this service to notify the Kiswe back-end which streams currently exist and which are appearing or disappearing. The KPS API receives these notifications and makes use of look-up tables to see which streams are connected to which events. It will then create the correct ingests in the back-end to then insert this stream as an additional feed into the event. In addition, these new streams can now also be accessed through the KPS API.

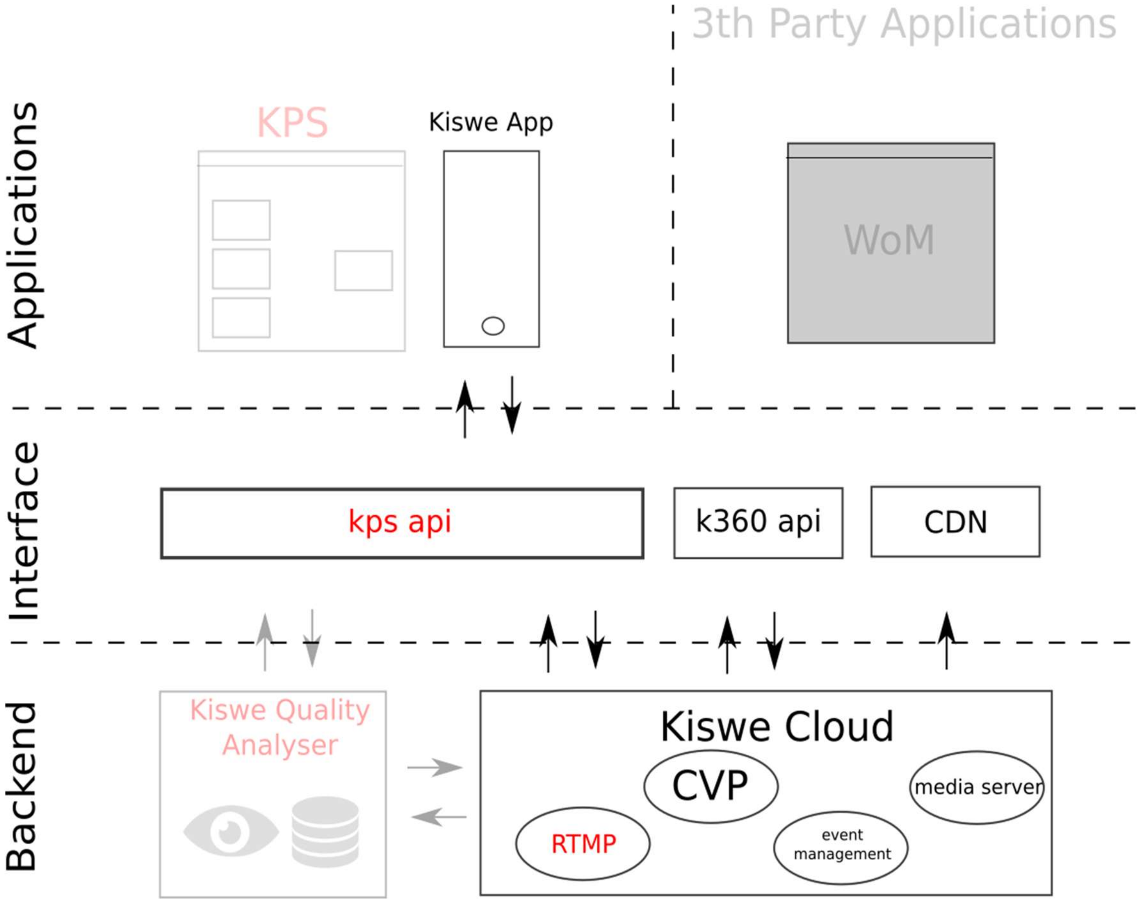


Figure 9: integration of RTMP in Kiswe Cloud

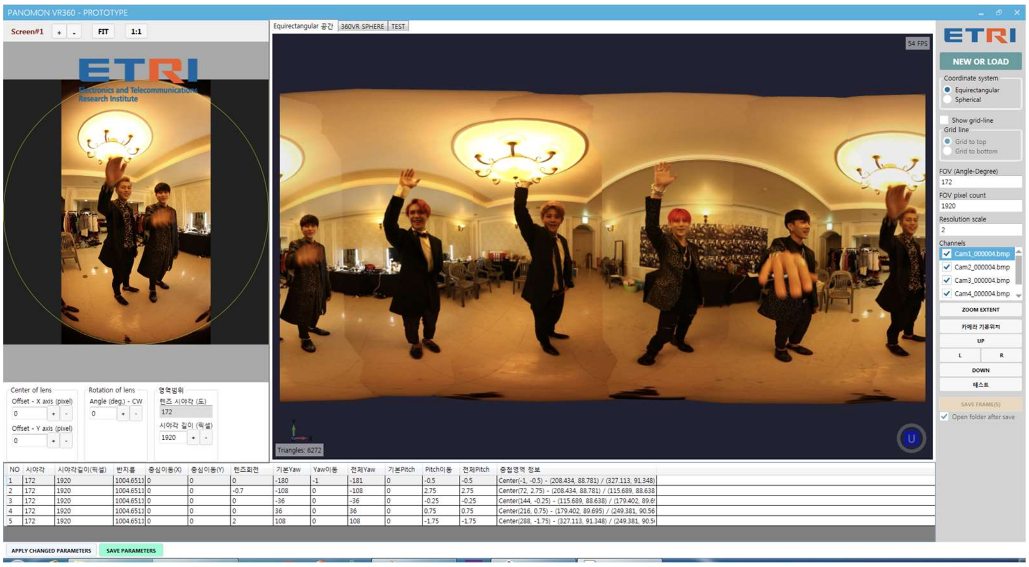
Simply reusing the capturing code of KVEs is not working. As the mobile device and the network are not predictable, we need to be able to handle missing packages or packages arriving in the wrong order. The rtmp protocol contains a clock based on the timing that the packages are sent from the mobile device with an offset. This mechanism aids us to handle packages in the right order. When working with these clocks we noticed clock drifts that also needed to be handled. We calculate the clock drift by measuring the time of arrival of each package over a longer time period. As a counter measure we regularly need to speed-up or slow down the incoming streams in the back-end to handle these drifts. As an additional counter measure we need to work with larger jitter buffers to handle network unpredictability and to continue to deliver a continuous smooth looking stream.

In its current state, the KPS API is not publicly available yet for partners as it is still under development. We have no authorization yet as part of the API to protect private content from being addressed by any 3th party. This has to be tackled first before publishing the API.

4. KR Application Descriptions

4.1. 360VR Monitoring system

The 360VR monitoring system of ETRI provides an easy and fast preview method to 360VR directors. Basically, it receives up to six live feeds at 1080@60p quality, currently. With the original feeds, it also manipulates in between equirectangular domain and sphere domain. For each feed, it applies a field of view angle, the actual seam based on the overlapped regions with concatenated feeds and even exposure control. Also, users can adjust the posture of the feeds in three dimensional rotations; yaw, pitch and roll. The only output format of this monitoring system is just a series of stitched images currently and further output formats will be provided. Also, ETRI is working on a encoding process to provide a video stream of the entire 360VR video. This video processing methods are being implemented using a state of the art GPU functionalities.



Center of lens
Offset - X axis (pixel): 0
Offset - Y axis (pixel): 0

Rotation of lens
Angle (deg.) - CW: 0

영역범위
렌즈 시야각 (도): 172
시야각 길이 (픽셀): 1920

Coordinate system
● Equirectangular
○ Spherical

□ Show grid-line

Grid line
● Grid to top
○ Grid to bottom

FOV (Angle-Degree): 172
FOV pixel count: 1920
Resolution scale: 2

Channels
☑ Cam1_000004.bmp
☑ Cam2_000004.bmp
☑ Cam3_000004.bmp
☑ Cam4_000004.bmp

☑ Open folder after save

NO	시야각	시야각길이(픽셀)	반지름	중심이동(X)	중심이동(Y)	렌즈회전	기본Yaw	Yaw이동	전체Yaw	기본Pitch	Pitch이동	전체Pitch	중첩영역 정보
1	172	1920	1004.6511	0	0	0	-180	-1	-181	0	-0.5	-0.5	Center(-1, -0.5) - (208.434, 88.781) / (327.113, 91.348)
2	172	1920	1004.6511	0	0	-0.7	-108	0	-108	0	2.75	2.75	Center(72, 2.75) - (208.434, 88.781) / (115.689, 88.638
3	172	1920	1004.6511	0	0	0	-36	0	-36	0	-0.25	-0.25	Center(144, -0.25) - (115.689, 88.638) / (179.402, 89.6
4	172	1920	1004.6511	0	0	0	36	0	36	0	0.75	0.75	Center(216, 0.75) - (179.402, 89.695) / (249.381, 90.56
5	172	1920	1004.6511	0	0	2	108	0	108	0	-1.75	-1.75	Center(288, -1.75) - (327.113, 91.348) / (249.381, 90.5

Figure 10: 360/VR monitoring system.

4.2. Adaptive 360VR player

The adaptive 360VR player provides an efficient way to render a 360VR video streams with multiple tiles in real-time. Currently, it receives a set of encoded tiled videos while each tile represents a spatially and temporally separated video. With these multiple tiles, it decodes simultaneously and renders on a single 360VR video buffer on the fly. The screenshot shows the stitched, synchronized and well aligned videos in ERP domain. Users can also see at a sphere domain with a certain viewing angle.

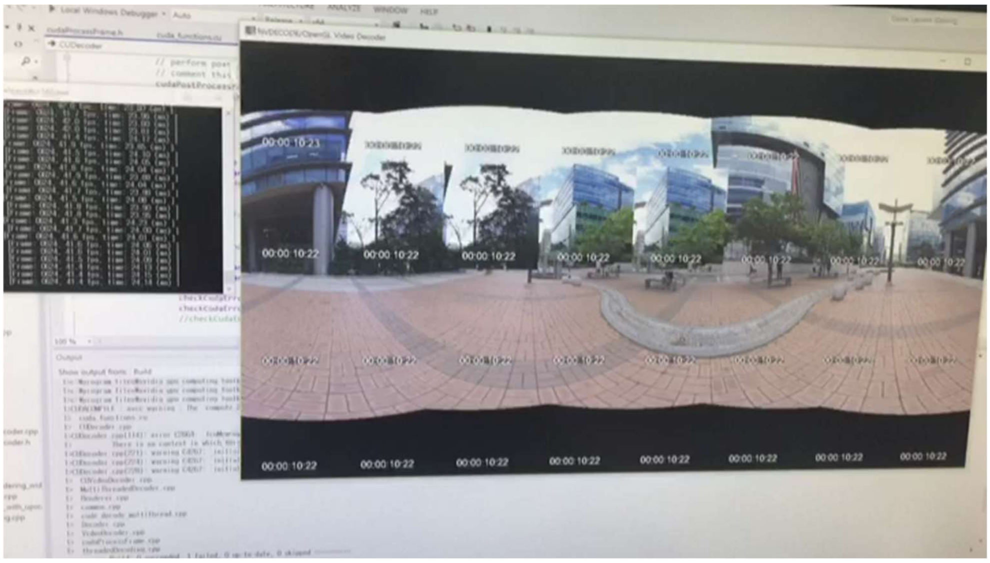


Figure 11: Adaptive 360VR player with multiple tiles.

5. Concluding Remarks

This Deliverable D3.1.1 reports on the application designs for the 1st year milestone. The document outlines the various applications that the MOS2S partners develop to showcase the innovations at the first year demonstrator. With respect to the initial Full Project Proposal, partners have developed application software that focuses on their interests and market developments, concentrating more on live (sports) events. A variety of applications have been developed, in which several form of integration between data and video are explored.

These applications will be used in subsequent demonstrator events to drive developments in platform functionality.