



## D5.4

# IMPLEMENTATION PLAN FOR BUILDING OF ECOSYSTEM

March 2014

### ABSTRACT

This document describes each FI-CONTENT2 Specific Enabler (SE) that will be available for use by phase 3 projects. For each enabler it describes what it does, how it can be used, in what format it is available for use, and what the terms and conditions are for its use.

This document is a deliverable of the FI-CONTENT2 integrated project supported by the European Commission under its FP7 research funding programme, and contributes to the FI-PPP (Future Internet Public Private Partnership) initiative.

#### DISCLAIMER

All intellectual property rights are owned by the FI-CONTENT2 consortium members and are protected by the applicable laws. Except where otherwise specified, all document contents are: “© FI-CONTENT2 project - All rights reserved”. Reproduction is not authorised without prior written agreement.

All FI-CONTENT2 consortium members have agreed to full publication of this document.

The commercial use of any information contained in this document may require a license from the owner of that information.

All FI-CONTENT2 consortium members are also committed to publish accurate and up to date information and take the greatest care to do so. However, the FI-CONTENT2 consortium members cannot accept liability for any inaccuracies or omissions nor do they accept liability for any direct, indirect, special, consequential or other losses or damages of any kind arising out of the use of this information.

#### DELIVERABLE DETAILS

[Full project title]:	Future media Internet for large-scale CONTeNT experimeNTation 2
[Short project title]:	FI-CONTENT2
[Contract number]:	603662
[WP n°]:	WP5 – Ecosystem Building
[WP leader]:	BBC
[Deliverable n°]:	D5.4
[Deliverable title]:	Implementation plan for building of ecosystem
[Deliverable nature]:	Report (R)
[Dissemination level]:	Public (P)
[Contractual delivery date]:	M12
[Actual delivery date]:	1 April 2014
[Editor]:	Dominic Tinley (BBC)
[Internal Reviewers]:	Hélène Waters (BBC) Michael Eble (Fraunhofer IAIS)
[Suggested readers]:	SMEs, developers, and industry players
[Keywords]:	use of SE, SMEs, phase 3, Open Call, Terms & Conditions
[File name]:	FI-CONTENT2_WP5-007_D5.4_V1.0

## **EXECUTIVE SUMMARY**

This document describes each FI-CONTENT2 Specific Enabler (SE) that will be available for use by phase 3 projects. For each enabler it describes what it does, how it can be used, in what format it is available for use, and what the terms and conditions are for its use.

This information forms the basis of a concrete plan for integrating FI-CONTENT2 outputs with phase 3 projects. It enables proposers and new joiners to clearly see the foundation work that has been completed so far upon which new platforms and services can be built.

The terms and conditions are presented in a simple and consistent format with a summary of each so phase 3 projects can make a quick assessment of the suitability of an SE before the need to analyse the detail of the licensing requirement.

## LIST OF AUTHORS

Organisation	Author
BBC	Barbara Zambrini, Chris Needham, Dominic Tinley, Natasha Locker
DFKI	Stefan Lemme
Disney	Chino Noris
Disney (BLRK)	Kenny Mitchell
ETHZ	Marcel Lancelle
FOKUS	Robert Seeliger, Annette Wilson
Fraunhofer IAIS	Michael Eble
IRT	Fabian Schiller, Ralf Neudel, Christoph Ziegler
Orange	Franck Feurtey, Arnaud Brun, Stéphane Geney, Sylvie Gruson
Pixelpark	Dirk Krause
Studio Gobo Ltd	Ali Durrant
Technicolor	Pieter Van Der Linden, Nathalie Cabel
Thales	Farid Benbadis

## TABLE OF CONTENTS

<b>EXECUTIVE SUMMARY .....</b>	<b>3</b>
<b>LIST OF AUTHORS.....</b>	<b>4</b>
<b>TABLE OF CONTENTS.....</b>	<b>5</b>
<b>ABBREVIATIONS.....</b>	<b>6</b>
<b>DEFINITIONS .....</b>	<b>7</b>
<b>1 - INTRODUCTION .....</b>	<b>8</b>
<b>2 - SUMMARY INFORMATION ABOUT ALL SEs.....</b>	<b>9</b>
2.1 - Social Connected TV Platform .....	10
2.2 - City Guide Platform .....	10
2.3 - Games Platform .....	11
<b>3 - DETAILED INFORMATION FOR EACH SE .....</b>	<b>12</b>
3.1 - Social Connected TV Platform .....	12
3.1.1 - BBC: TV Application Layer .....	12
3.1.2 - IRT: Second Screen Framework .....	15
3.1.3 - Fraunhofer FOKUS: Use of Content Enrichment .....	18
3.1.4 - Fraunhofer IAIS: Audio Fingerprinting .....	21
3.1.5 - Fraunhofer IAIS: Audio Mining .....	24
3.1.6 - Fraunhofer IAIS: Content Optimisation.....	27
3.1.7 - Technicolor: Content Atmosphere Search & Discovery .....	30
3.1.8 - Technicolor: Content Similarity Search & Discovery .....	33
3.2 - City Guide Platform .....	36
3.2.1 - Orange – Reperio (Recommendation Services) .....	36
3.2.2 - Orange: KIWANO (Virtual Mixed Reality).....	39
3.2.3 - Thales Communications and Security: Content Sharing.....	42
3.2.4 - Pixelpark: Social Network.....	45
3.2.5 - Fraunhofer FOKUS: Open City Database .....	48
3.2.6 - Fraunhofer FOKUS: Recommendation as a Service .....	50
3.3 - Games Platform .....	53
3.3.1 - ETHZ: Leaderboard .....	53
3.3.2 - ETHZ: Reality Mixer – Camera Artefact Rendering .....	55
3.3.3 - ETHZ: Reality Mixer – Reflection Mapping.....	57
3.3.4 - DFKI: Augmented Reality – Marker Tracking.....	60
3.3.5 - BLRK/ETHZ: Augmented Reality – Fast Feature Tracking.....	63
3.3.6 - DRZ: Game Synchronization .....	66
3.3.7 - Gobo: Spatial Matchmaking .....	69
<b>4 - CONCLUSION .....</b>	<b>72</b>

## ABBREVIATIONS

API	Application programming interface
AR	Augmented reality
ARD	Arbeitsgemeinschaft der öffentlich-rechtlichen Rundfunkanstalten der Bundesrepublik Deutschland (consortium of public broadcasters in Germany)
CMS	Content management system
EPG	Electronic program guide
IoT	Internet of Things
MIT	Massachusetts Institute of Technology
NER	Named-entity recognition
NLP	Natural language processing
NLV	Non-linear video
OCDB	Open City Database
P2P	Peer to peer
PPP	Public-private partnership
RaaS	Recommendation as a Service
S2C	Server to client
SE	Specific enabler
SRT	SubRip Subtitle format
SSF	Second Screen Framework
TAL	TV Application Layer
UGC	User-generated content
VOD	Video on demand
WAR	Web application ARchive file format

## DEFINITIONS

Unity	A cross-platform game engine used to develop video games for web plugins, desktop platforms, consoles and mobile devices
Specific Enabler (SE)	An enabler realized by the FI-CONTENT2 project. Specific Enablers may be layered on top of, or otherwise make use of, Generic Enablers. Please refer to the definition of a FI-CONTENT 2 SE from deliverable D6.1 Architecture specification.

## 1 - INTRODUCTION

This document follows a template format where information for each FI-CONTENT2 specific enabler (SE) is provided in a consistent manner including the full terms and conditions of use. For each SE the following information is included:

- Owner/developer (name of company)
- What it does (plain text description)
- How it works (plain text description)
- What you get (high-level summary and plain text description)
  - Delivery model
  - Languages used
- Examples of use (e.g. where it is being used or how it could be used)
- Terms and conditions of use (high-level summary and plain text description)
  - Licence type
  - Licence features
  - Licence fee
  - Licence summary
  - Copyright statement
  - Full licence – where the full licence can be found
- Performance requirements (e.g. the number of users that can be supported and the specification of machines required to run the code)
- Further documentation (linking to project pages where relevant)
- Contact information



## 2 - SUMMARY INFORMATION ABOUT ALL SES

This section of the document summarises the key elements and licensing arrangements of each specific enabler.

The aim is to enable any developer to make a quick assessment of whether a particular SE is worth investigating before they spend time exploring the detail of the licensing arrangements.

Note in some cases more than one delivery mode, licence, type of use or licence fee option is available in which cases there is a choice as to which can be used.

The table includes the following abbreviations:

### **Delivery mode**

Android = Android library

Hosted = hosted services

SaaS = Software as a Service

Server = Server software

Source = source code available

Unity = Unity 3D package

WAR = Web application ARchive file

### **Licence type**

Eval = can only be used for evaluation purposes

Open = Open source

Prop = Proprietary

### **Type of use**

Com = commercial use available (see full entry for further details)

### **Licence fee**

Per = Per use payment

Recurring = Recurring payment

Upfront = Upfront payment

## 2.1 - Social Connected TV Platform

Name of SE	Delivery model	Licence type	Type of use	Licence fee	Free in FI-PPP
TV Application Layer	Source	Open	Com	Free	Yes
Second Screen Framework	Hosted	Eval Prop		Free in FI-PPP	Yes
Use of Content Enrichment	Hosted	Eval	Com	Per Recurring Upfront	Yes
Audio Fingerprinting	Hosted	Eval Prop	Com	Recurring Upfront	Yes
Audio Mining	Hosted	Eval Prop	Com	Recurring Upfront	Yes
Content Optimisation	Hosted SaaS	Eval Prop	Com	Recurring Upfront	Yes
Content Atmosphere Search & Discovery	Hosted	Eval Prop	Com	Per Recurring Upfront	Yes
Content Similarity Search & Discovery	Hosted	Eval Prop	Com	Per Recurring Upfront	Yes

## 2.2 - City Guide Platform

Name of SE	Delivery model	Licence type	Type of use	Licence fee	Free in FI-PPP
Reperio (Recommendation Services)	WAR	Eval Prop	Com	Recurring Upfront	
KIWANO (Virtual Mixed Reality)	Hosted Source	Eval Open Prop	Com	Per Recurring Upfront	Yes
Content Sharing	Android Server	Eval Prop	Com	Free in FI-PPP	Yes
Social Network	Source	Open	Com	Free	Yes
Open City Database	Hosted Source	Open Prop	Com	Free	Yes
Recommendation as a Service	Hosted	Eval Prop	Com	Per Recurring Upfront	Yes

### 2.3 - Games Platform

Name of SE	Delivery model	Licence type	Type of use	Licence fee	Free in FI-PPP
Leaderboard	Source	Open	Com	Free	Yes
Reality Mixer – Camera Artefact Rendering	Source Unity	Open	Com	Free	Yes
Reality Mixer – Reflection Mapping	Source Unity	Open	Com	Free	Yes
Augmented Reality – Image Marker Tracking	Source	Open	Com	Free	Yes
Augmented Reality – Fast Feature Tracking	Source	Open	Com	Free	Yes
Game Synchronization	Source	Open	Com	Free	Yes
Gobo: Spatial Matchmaking	Source	Open	Com	Free	Yes

### 3 - DETAILED INFORMATION FOR EACH SE

#### 3.1 - Social Connected TV Platform

##### 3.1.1 - BBC: TV Application Layer

TV Application Layer	
<b>Owner/developer</b>	
BBC	
<b>What it does</b>	
<p>The TV Application Layer (TAL) is a software layer for developing Connected TV applications. There are hundreds of devices in the marketplace and they all use slightly different technologies to achieve similar results. The purpose of the TAL is to allow you to write an application once and be confident that it can be deployed to all HTML-based TVs. The TAL provides abstractions for varying device capabilities and interfaces including media playback, remote control input, animation, networking and persistent storage. It also provides a set of user interface controls with widget interaction, navigation and input focus management.</p>	
<b>How it works</b>	
<p>The TAL software layer abstracts the differences between various Connected TV, set-top box, and games console implementations, giving TV application authors a consistent API to develop against. The bulk of your development can be done on a desktop browser that is built on the same origins as the TV browsers. It does not mean there will not be things that work differently once you run your application on TV devices, but it does mean that you can focus on building the features you want in your app rather than worrying about the specifics of TV devices in great detail.</p>	
<b>What you get</b>	
<i>Delivery model</i>	
• Source code	yes
<i>Languages used</i>	
• JavaScript	yes
• PHP	yes
<i>Description</i>	
<p>TAL is an open source JavaScript library, with a PHP (or Node.js) server-side component. The framework is organised as a set of JavaScript modules based on the Require library and a modified version of require.js 0.15 is supplied. You include the JavaScript library in the web page for your Connected TV application. The server-side component detects which browser is used and delivers a device-specific configuration file that adapts the TAL to the capabilities of that particular device.</p>	

## TV Application Layer

### Examples of use

- TAL is the framework used for development of the BBC iPlayer across all devices
  - [www.bbc.co.uk/iplayer](http://www.bbc.co.uk/iplayer)
- An example application is included in the code repository along with associated tutorial pages
  - <https://github.com/fmtvp/talexample>
  - <http://fmtvp.github.io/tal/getting-started/tutorial.html>
- Other notable community projects include The Doctor on TAL and a code fork to provide node.js support
  - <https://github.com/GaryHigham/thedoctorontal>
  - <https://github.com/landeiro/tal>

### Terms and conditions of use

#### *Licence type*

- Open source                      yes
- Proprietary                       subject to agreement
- Evaluation licence               no

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          yes
- Distribution allowed            yes
- Include copyright                must
- Include original                 not required
- State changes                  not required
- Disclose source code            not required

#### *Licence fee*

- Free

#### *Licence summary*

TAL is available and free to use to all under the terms of the Apache 2.0 open source licence.

#### *Copyright statement*

Copyright © 2013 British Broadcasting Corporation and TAL Contributors

#### *Full licence*

<https://github.com/fmtvp/tal/blob/master/LICENSE-2.0>

## TV Application Layer

### Performance requirements

The front-end components of TAL will work in all modern browsers on both PCs and set top boxes. The server side component requires a basic web server running PHP5. The web server has minimal work so can be specified according to the page demands of your specific Connected TV application. TAL will work on Samsung devices (2010 onwards), Panasonic (2011 onwards), Sony (2011 onwards), Toshiba (2011 onwards), LG (2012 onwards) and a whole range of other devices. For a full list of current devices view the config folder in the code repository:

<https://github.com/fmtvp/tal/tree/master/config/devices>

### Further documentation

<http://mediafi.org/?portfolio=tv-application-layer#tab-documentation>

### Contact information

Chris Needham

[chris.needham@bbc.co.uk](mailto:chris.needham@bbc.co.uk)

### 3.1.2 - IRT: Second Screen Framework

## Second Screen Framework

### Owner/developer

IRT

### What it does

The Second Screen Framework (SSF) provides web applications that are running on a TV with all the crucial functionalities to establish a persistent bi-directional communication path to a web application running in the browser of any second screen device.

This includes the possibility to launch applications from one a TV to a second screen. All functionality is provided via a slim JavaScript API and can thus be easily integrated into any web application.

Since the solution is fully compliant with the HbbTV standard, it enables content providers to create fully interactive applications with direct programme relation potentially targeting millions of already deployed devices on the market. Thus, the concept can be implemented without modifications to hardware and only minimal extensions to existing applications.

### How it works

The SSF provides the following features to applications:

- Discovery
- Connection
- App-to-app communication
- App launch

Communication between a TV application and its companion application on the second screen is handled by a web server. Discovery between the devices is accomplished by means of a QR code. Both devices get a unique identifier from the framework server. The IDs are stored in cookies in the devices' browsers. The ID pairs are stored in a database on the framework server. Thus the connection between the devices is persistent. Users need to execute the discovery process only once. The framework is hosted as a service that is independent from the application that is making use of it. End users can benefit from their device connections in every service that supports the framework, without having to manage application-specific connection settings.

Since the app-to-app communication is done over the open internet, there are no specific requirements regarding the network configuration to be implemented by the end-user (e.g. the devices do not need to be in the same local network). The only requirements regarding the technical set-up of the end-users devices are that both devices need an internet connection and that they have a JavaScript-enabled web browser installed.

## Second Screen Framework

### What you get

#### *Delivery model*

- Hosted service                      yes
- Source code                         no

#### *Languages used*

- JavaScript                         yes
- PHP                                 no

#### *Description*

- JavaScript library and API
- Full duplex communication between web applications running in the browser of connected TV devices and second screen devices
- Mechanism for device discovery based on QR code
- Persistent device connection – devices, once coupled, remain associated
- Automatic launch of applications on the second screen device
- Based on standardised web technology

### Examples of use

SSF has been integrated for now into four existing HbbTV services of the ARD network. These are: ARD EPG, ARD-Mediathek, ARD Text and rbbtext. The ARD EPG offers the second screen functionality in a public trial to its viewers. The application is signalled in free to air broadcast on all channels of the ARD network over DVB-C, DVB-S and DVB-T.

### Terms and conditions of use

#### *Licence type*

- Open source                        no
- Proprietary                         yes
- Evaluation licence                yes (to 30 June 2016, only on SSF service, not on source code)

#### *Licence features*

- Commercial use                    no
- Modifications allowed            no
- Distribution allowed              no
- Include copyright                 must
- Include original                    not applicable (no source code distribution)
- State changes                     must
- Disclose source code              not required



## Second Screen Framework

### *Licence fee*

No fee for use in the frame of the FI-PPP

### *Licence summary*

Under this licence, the Second Screen Framework must only be used for testing purposes and must not be used for commercial services. The framework and its API must not be made available to third parties. The copyright holder IRT cannot grant warranty on the framework under this free testing license. By using the framework you agree to the full terms and conditions.

### *Copyright statement*

Copyright © 2013 Institut fuer Rundfunktechnik GmbH

### *Full licence*

<http://mediafi.org/?portfolio=second-screen-framework#tab-terms-conditions>

### **Performance requirements**

There are no performance requirements for developers/application providers to consider regarding the use of SSF. An instance of SSF is maintained by IRT and hosted on the FI-WARE Testbed for use within FI-PPP. In order to make use of second screen functionalities enabled by SSF end users' devices' must have an internet connection and must have a JavaScript-enabled browser.

### **Further documentation**

<http://mediafi.org/?portfolio=second-screen-framework#tab-documentation>

### **Contact information**

Christoph Ziegler

[ziegler@irt.de](mailto:ziegler@irt.de)

### 3.1.3 - Fraunhofer FOKUS: Use of Content Enrichment

## Use of Content Enrichment

### Owner/developer

Fraunhofer FOKUS

### What it does

Use of Content Enrichment provides functions to create, distribute and play interactive video content across platforms and devices by making objects in the video clickable for their viewers. It also provides interfaces to incorporate Web 2.0 capabilities and community functionalities. Thus, the enabler acts as a common building block in future video and multimedia infrastructures. It allows seamless, platform-independent and convenient enrichment of any type of video content using any type of device for a plurality of application cases. Such use cases cover user-generated content (UGC), professional content, advertising and edutainment.

Content Enrichment provides the following capabilities:

- **Media/Text Annotation**
  - Allows any type of media content (e.g. video, audio, photo or text) to be enriched with user-generated video content
  - Add/edit comments, notes, enrich the media content with custom drawings or create custom mash-ups of media content
  - Enable the retrieval of related information (e.g. via access to external resources, location and content sensitive metadata, object tracking etc)
- **Object-based Media Discovery**
  - Enriched media content contains metadata information on available objects within the media
  - Referenced information on available objects can be used to discover other object-related media content
- **Cross-Video Navigation**
  - Allows user interaction to jump back and forth between video content

### How it works

Use of Content Enrichment brings clickable video to Smart TVs, Mobiles, Tablets and PCs by adding a metadata layer to any kind of video content. An entirely web-based (HTML5) interactive video authoring tool provides the interface to tag video content, add object related information and link with related media. The metadata is handled through an API for both creation of interactive content as well as usage of interactive content in apps and services. Metadata related to a video is signalled out of band in parallel to video content and can be requested through the Use of Content Enrichment API.

## Use of Content Enrichment

### What you get

#### *Delivery model*

- Hosted service                    yes
- Source code                        no

#### *Languages used*

- JavaScript                         yes
- PHP                                    yes
- JSON                                 yes

#### *Description*

Use of Content Enrichment is a hosted solution to create interactive video content consisting of a PHP and mySQL based backend. Interactive content can be created through an HTML5 based tagging tool that uses Java Script and communicates with the RESTful API via JSON to create and store metadata. Interactive video players can be implemented in HTML supporting the JSON player API.

### Examples of use

- Use of Content Enrichment is used for development of interactive video apps within FI-CONTENT and by Fraunhofer FOKUS <http://cenr.fokus.fraunhofer.de/play/>

### Terms and conditions of use

#### *Licence type*

- Open source                        no
- Proprietary                         subject to agreement
- Evaluation licence                yes (30 days)

#### *Licence features*

- Commercial use                    yes
- Modifications allowed            yes (via own custom players)
- Distribution allowed                yes
- Include copyright                    must
- Include original                     not required
- State changes                        not required
- Disclose source code                not required

#### *Licence fee*

- Upfront payment / Recurring payment / Per use payment

## Use of Content Enrichment

### *Licence summary*

Use of Content Enrichment is based on Fraunhofer FOKUS background technology to support non-linear-video (NLV). All project partners can use it during the term of the FI-PPP projects for testing and experimentation. For any other use (e.g. commercial use) please contact us.

### *Copyright statement*

Copyright © 2013 Fraunhofer Institute for Open Communication Systems FOKUS

### *Full licence*

Full licence may be provided on request and needs to be negotiated on a case-by-case basis.

### **Performance requirements**

Content Enrichment provides an API that enables interactive content on a variety of platforms and devices such as PC, mobile, tablet and especially SmartTV and hybrid TVs. Any up-to-date browser is able to implement and integrate with Content Enrichment through Javascript and JSON. The hosted service is able to serve up to 1,000 users if the raw video content is streamed from a different source (e.g. Object Storage GE instance which could be linked with Content Enrichment).

### **Further documentation**

<http://wiki.mediafi.org/doku.php/ficontent.common.enabler.contentenrichment>

### **Contact information**

Robert Seeliger

[robert.seeliger@fokus.fraunhofer.de](mailto:robert.seeliger@fokus.fraunhofer.de)

### 3.1.4 - Fraunhofer IAIS: Audio Fingerprinting

## Audio Fingerprinting

### Owner/developer

Fraunhofer Institute for Intelligent Analysis and Information Systems (IAIS)

### What it does

Audio Fingerprinting recommends matching content for second screen devices (e.g. tablet computers). It analyses an incoming audio signal (e.g. from a TV or a video on demand (VOD) stream), computes a fingerprint and checks that fingerprint against a database potentially containing the analysed content data. Finally, the service returns matching content either as links to a repository or as the content itself. The service can be implemented into iOS-based and Android-based applications via SDKs.

### How it works

Audio Fingerprinting consists of an indexing component and a two-part matching component. The indexing component handles the adding and removing of media to and from a fingerprint index and is currently an offline process. As a result an index of acoustic fingerprints is generated. The matching component consists of two parts: a Mobile SDK and a matching server. The mobile SDKs for iOS and Android can be linked to a mobile application that takes care of recording audio, calculating an acoustic fingerprint (while also decrypting the data), sending the fingerprint to a matching server and passing the results back to the application. The matching server, bound to a single fingerprint index, receives fingerprints, performs the lookup on the database and generates a response depending on the matching result. The algorithm is robust with regard to regular background noises, compression and distortions.

### What you get

#### *Delivery model*

- Hosted service                      yes (already deployed for FI-WARE Cloud)
- Source code                         no
- SDK for Android+iOS            yes (in order to integrate it into mobile applications)

#### *Languages used*

- C/C++                                 yes
- Python                                yes
- Objective C (iOS)                 yes
- Java (Android)                     yes

#### *Description*

Audio Fingerprinting can be used as a combination of a hosted service via a RESTful API and as an SDK for integrating audio fingerprinting capabilities into mobile applications for iOS and Android (e.g. a second screen application for a TV series). The programming languages used within the SE are C/C++, Python, Objective C (iOS) and Java (Android). The hosted service can be run on Fraunhofer infrastructure or on FI-WARE Cloud infrastructure where it has already been deployed. The SDKs and documentation can be downloaded from the SE's website (see below).

## Audio Fingerprinting

### Examples of use

FIcontent second screen demo:

<http://fi-content.iais.fraunhofer.de/fc/audiofingerprinting/#/app>

### Terms and conditions of use

#### *Licence type*

- Open source no
- Proprietary yes
- Evaluation licence yes

#### *Licence features*

- Commercial use subject to agreement (within phase 3 of the FI-PPP)
- Modifications allowed no
- Distribution allowed no
- Include copyright negotiable
- Include original not required
- State changes not required
- Disclose source code must

#### *Licence fee*

- Upfront payment / Recurring payment (subject to agreement, negotiable)

#### *Licence summary*

Audio Fingerprinting can be used free of charge within phase 3 of the FI-PPP as described above. It offers functionalities through APIs and has been built using several programming languages. It cannot be provided as open source.

#### *Copyright statement*

Audio Fingerprinting has been generated by Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS, all copyrights are owned by Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V., Hansastraße 27c, 80686 Munich, Germany.

#### *Full licence*

<http://fi-content.iais.fraunhofer.de/fc/audiofingerprinting/#/terms>

## Audio Fingerprinting

### Performance requirements

- The Mobile SDK can generally be integrated into any Android or iOS application
- Android is supported in any version higher than 2.3.x (Gingerbread), iOS in any version higher than 5.0; there are no special hardware requirements
- The indexing and matching component requires a 64-bit machine with a typical Unix-distribution like Ubuntu
- A single CPU can serve approximately 30 permanent, concurrent requests using an index of approximately 10 hours of content (actual hardware needs depend highly on the use case and content)

### Further documentation

- FI-CONTENT wiki: <http://wiki.mediafi.org/doku.php/ficontent.socialtv.enabler.audiofingerprinting>
- SE's website: <http://fi-content.iais.fraunhofer.de/fc/audiofingerprinting/>

### Contact information

Michael Eble  
[michael.eble@iais.fraunhofer.de](mailto:michael.eble@iais.fraunhofer.de)

3.1.5 - *Fraunhofer IAIS: Audio Mining*

## Audio Mining

### Owner/developer

Fraunhofer Institute for Intelligent Analysis and Information Systems (IAIS)

### What it does

Audio Mining analyses a given audio/video file in German spoken language (e.g. content from a TV news show) and returns textual information for indexing (e.g. for search engines). Speech and speaker segmentation as well as speech recognition are performed in order to turn speech into text. The SE delivers segments, the speakers identified, characteristic keywords and various metadata in XML. Finally, the SE builds an index for Multimedia Search.

### How it works

Audio Mining incorporates state-of-the-art multimedia pattern recognition algorithms such as speech detection, speaker diarisation, speaker recognition and speech recognition. By cascading these algorithms, it automatically obtains a broad spectrum of metadata for media files. This enables users to search for terms, quotations or specific speakers, browse through archives using content-based recommendation or obtain media information such as keywords or SRT-compatible subtitles. Audio Mining incorporates a powerful Apache Solr search engine that stores all metadata and makes it available via the provided SOAP/REST interface.

### What you get

#### *Delivery model*

- Hosted service                      yes (already deployed for FI-WARE Cloud)
- Source code                         no

#### *Languages used*

- C/C++                                 yes
- Java                                    yes

#### *Description*

Audio Mining offers a RESTful API which can be used to turn audio/visual content in German spoken language into machine readable text, using automatic speech recognition (ASR), and to recommend matching content for video clips. Via the API, the technology can, for example, be integrated into a CMS. The programming languages used within the SE are C/C++ and Java. The hosted service can be run on Fraunhofer infrastructure or on FI-WARE Cloud infrastructure where it has been deployed already.

### Examples of use

- FI-CONTENT Audio Mining demo: <http://fi-content.iais.fraunhofer.de/fc/audiomining/>
- Could be used for indexing multimedia archives of TV broadcasters



## Audio Mining

### Terms and conditions of use

#### *Licence type*

- Open source no
- Proprietary yes
- Evaluation licence yes

#### *Licence features*

- Commercial use subject to agreement (within phase 3 of the FI-PPP)
- Modifications allowed no
- Distribution allowed no
- Include copyright negotiable
- Include original not required
- State changes not required
- Disclose source code must

#### *Licence fee*

- Upfront payment / Recurring payment (subject to agreement, negotiable)

#### *Licence summary*

Audio Mining can be used free of charge within phase 3 of the FI-PPP as described above. It offers functionality through APIs and has been built using several programming languages. It cannot be provided as open source.

#### *Copyright statement*

Audio Mining has been generated by Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS, all copyrights are owned by Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V., Hansastraße 27c, 80686 Munich, Germany.

#### *Full licence*

<http://fi-content.iais.fraunhofer.de/fc/audiomining/#/terms>

### Performance requirements

- Audio Mining runs on Windows-based and Unix-based 32-bit & 64-bit machines (Unix recommended)
- Minimum requirement is a dual-core machine with at least 4GB RAM (Quad-core with 8GB recommended)
- One instance in the recommended setup can process approximately 32 hours of speech data per day
- The search engine's performance depends highly on the use case. Here typical Solr distribution strategies can be applied

## Audio Mining

### Further documentation

- <http://wiki.mediafi.org/doku.php/ficontent.socialtv.enabler.audiofingerprinting>
- <http://fi-content.iais.fraunhofer.de/fc/audiofingerprinting/>

### Contact information

Michael Eble  
[michael.eble@iais.fraunhofer.de](mailto:michael.eble@iais.fraunhofer.de)

### 3.1.6 - Fraunhofer IAIS: Content Optimisation

## Content Optimisation

### Owner/developer

Fraunhofer Institute for Intelligent Analysis and Information Systems (IAIS)

### What it does

Content Optimisation processes incoming textual content (e.g. from the Audio Mining SE) and extracts characteristic keywords. Afterwards, semantic enrichment based on natural language processing (NLP) will be performed to connect the transcripts and keywords with additional, contextual information. Therefore, the SE integrates and harmonises additional content from disperse sources. The software is intended for SMEs that want to build second screen applications (e.g. for TV documentaries).

### How it works

Content Optimisation leverages technologies for named entity recognition and spotting of entities in textual content. To accomplish this goal it uses normed data that is available open sources (e.g. DBPedia.org) to recognise people, organisations and/or locations in the provided textual data. Those entities will be used to annotate the data and to build up a search index. The latter enables the user to experience an intuitive search with state of the art searching technologies such as a faceted search approach. The search and retrieval of enriched content is accessible via a RESTful API, which enables the user to easily integrate the service into its architecture and/or frameworks.

### What you get

#### *Delivery model*

- Hosted service                      yes (already deployed for FI-WARE Cloud)
- Source code                         no

#### *Languages used*

- Java                                    yes
- Scala                                  yes

#### *Description*

Content Optimisation offers a RESTful API that can be used to enrich textual content in German language with information from open sources like DBPedia.org, using named-entity recognition (NER) and spotting technologies, and to provide a retrieval interface with state-of-the-art search technologies like faceted search and auto-completion. Via the API, the technology can, for example, be integrated into a CMS. The programming languages used within the SE are Java and Scala. The hosted service can be run on Fraunhofer infrastructure or on FI-WARE Cloud infrastructure where it has been deployed already.

## Content Optimisation

### Examples of use

- Demo: <http://fi-content.iais.fraunhofer.de/fc/contentoptimisation>
- Modified version used as backend software for the German digital library demonstrating search capabilities: <http://www.ddb.de>

### Terms and conditions of use

#### *Licence type*

- Open source no
- Proprietary yes
- Evaluation licence yes

#### *Licence features*

- Commercial use subject to agreement (within phase 3 of the FI-PPP)
- Modifications allowed no
- Distribution allowed no
- Include copyright negotiable
- Include original not required
- State changes not required
- Disclose source code must

#### *Licence fee*

- Upfront payment / Recurring payment (subject to agreement, negotiable)

#### *Licence summary*

Content Optimisation can be used free of charge within phase 3 of the FI-PPP as described above. It offers functionalities through APIs and has been built using several programming languages. It cannot be provided as open source.

#### *Copyright statement*

Content Optimisation has been generated by Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS, all copyrights are owned by Fraunhofer-Gesellschaft zur Förderung der angewandten Forschung e.V., Hansastraße 27c, 80686 Munich, Germany.

#### *Full licence*

<http://fi-content.iais.fraunhofer.de/fc/contentoptimisation/#/terms>

## Content Optimisation

### Performance requirements

- Indexing/analysis of new objects is a hardware intensive task and should only be utilized by up to 2 parallel users at once on the currently available FI-WARE computing instances (2 cores + 4GB RAM)
- Querying of the analysed information and retrieval of the annotated content is dependent on the used GE's. While using the SE stand-alone it can handle up to 50 parallel requests at once on the currently available FI-Ware computing instances (2 cores + 4GB RAM)
- The minimum hardware requirement is 2 cores + 4GB Ram + 10GB HDD (sufficient for up to 500.000 unique objects/documents)

### Further documentation

- FI-CONTENT wiki: <http://wiki.mediafi.org/doku.php/ficontent.socialtv.enabler.contentoptimisation>
- SE's website: <http://fi-content.iais.fraunhofer.de/fc/contentoptimisation/>

### Contact information

Timm Kißels

[tim.kissels@iais.fraunhofer.de](mailto:tim.kissels@iais.fraunhofer.de)

### 3.1.7 - *Technicolor: Content Atmosphere Search & Discovery*

## Content Atmosphere Search & Discovery

### Owner/developer

Technicolor R&D France

### What it does

Content Atmosphere Search & Discovery provides a way to discover new movies according to the movie's atmosphere. The emotion is modelled along two dimensions: the arousal (calming to exciting) and the valence (displeasure to pleasure). Content Atmosphere Search & Discovery is a software solution based on a Technicolor algorithm that computes the associated atmosphere to a movie. The target customers for this solution are content providers or cinema information providers in order to offer VOD discovery services.

### How it works

The Content Atmosphere SE relies on an algorithm that processes movie metadata, in this first version genre metadata. The algorithm is based on a supervised machine learning technique called the Support Vector Regression Machine (SVR). The first step consists on a training phase to learn how to compute the arousal and valence values from a training set. The second step is the processing phase that computes arousal and valence values for each movie of from a catalogue. For the scenario "*Get Similar Movies with the same Mood*", the request sent to the SE is a Content ID, and the result is a list of Content IDs with the same atmosphere.

### What you get

#### *Delivery model*

- Hosted service                    yes
- Source code                        no

#### *Languages used*

- JavaScript                         yes
- Python                              yes

## Content Atmosphere Search & Discovery

### *Description*

Content Atmosphere Search & Discovery is a software solution. A back-end stores the movie metadata, computes the arousal and valence values and keeps the results in a database. The backend is based on Python, JavaScript frameworks and a MongoDB database. Content Atmosphere Search & Discovery exposes a RESTful API via JSON. There is no API for the ingest of the VOD catalogue. The algorithm has been tuned and validated on a US-based VOD catalogue. Another training phase is necessary for each new VOD catalogue. The preparation of the dataset requires a deep analysis of the input VOD catalogue requiring close communication between the catalogue provider and Technicolor as the Specific Enabler provider. At this stage, a first implementation of the industrialisation of the SE is available. Some improvements to the algorithm may be necessary depending on final input VOD catalogue that will need to be managed.

### **Examples of use**

The Content Atmosphere Search & Discovery technology is not embedded in the FI-CONTENT2 prototype, submitted to end user evaluation in Brittany. However, a stand-alone demonstrator is available. The demonstrator uses a catalogue with approximately 2,000 movies, each one positioned in a Mood map from its arousal and valence attributes delivered through an innovative user interface.

### **Terms and conditions of use**

#### *Licence type*

- Open source                      no
- Proprietary                        yes
- Evaluation licence                yes

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          no
- Distribution allowed            no
- Include copyright                must
- Include original                  not required
- State changes                    not required
- Disclose source code            not required

#### *Licence fee*

- Upfront payment / Recurring payment / Per use payment  
(subject to agreement, negotiable)

## Content Atmosphere Search & Discovery

### *Licence summary*

The provisions of the collaboration agreement shall govern relationships between projects of the FI-PPP. Some of the main provisions are summarised below for information only. Please refer to the Collaboration Agreement that exhaustively set forth the applicable rights and obligations and that shall prevail in case of conflict with the following summary. For the avoidance of doubt such Access rights do not imply any obligation by the granting Party to provide any support or maintenance for the Software, nor bear any responsibility for any claims for defects in the Software.

During Phase 3, All project partners which are not part of the Collaboration Agreement but have been approved by the European Commission can use it during the term of the FI-PPP projects for testing and experimentation, and under the term of the evaluation license agreement

For any other use, especially commercial use, access to Technicolor Content Atmosphere Specific Enabler, will be granted on a case-by-case basis upon written request and after negotiations.

### *Copyright statement*

Copyright © 2013 Technicolor

### *Full licence*

The evaluation licence agreement is available on FI-CONTENT2 website.

### **Performance requirements**

The machine required to host the SE runs under Linux Ubuntu-based 64-bit, and should be equipped with at least 8GB memory. The software components are Node.js, Python, Flask, Celery and MongoDB. The training phase is the longest phase, but a pre-processing phase.

As mentioned in the description, a study step of the format of the metadata provided by the third-party is necessary in order to understand the categorisation chosen by the content provider. Depending of the composition of the catalogue (i.e. titles of the movies), a training phase may be applied using crowd sourcing. In function of the results of the initial study, some adaptations to the algorithm may be necessary.

### **Further documentation**

[http://wiki.mediafi.org/doku.php/ficontent.socialtv.roadmap#content\\_atmosphere](http://wiki.mediafi.org/doku.php/ficontent.socialtv.roadmap#content_atmosphere)

### **Contact information**

Nathalie Cabel

[Nathalie.cabel@technicolor.com](mailto:Nathalie.cabel@technicolor.com)



### 3.1.8 - *Technicolor: Content Similarity Search & Discovery*

## Content Similarity Search & Discovery

### Owner/developer

Technicolor R&D France

### What it does

Content Similarity Search & Discovery is a software solution to provide content-to-content recommendations based only on content metadata. The object is to offer a solution that allows a list of similar movies to be obtained from an initial movie request. The technology is based on a Technicolor algorithm that computes distances between movies. The target customers for this solution are content providers and cinema Information providers in order to offer VOD discovery services.

### How it works

Content Similarity Search & Discovery relies on an algorithm that processes movie metadata, in the initial version mainly the genres, cast and crew. The first step consists of a pre-processing task to prepare the dataset. The algorithm computes for each movie in the catalogue a vector of weighted genres. The second step allows computation of the distances between movies. This technology is beneficial for several use cases (e.g. "Get Similar Movies" or "Build Clusters of a Catalogue"). For the scenario "Get Similar Movies" the request sent to the SE is a Content ID, and the answer is a list of Content IDs with the respective distances.

### What you get

#### *Delivery model*

- Hosted service                      yes
- Source code                         no
- Software as a service           no

#### *Languages used*

- Java                                    yes

#### *Description*

Content Similarity Search & Discovery is a software solution. A back-end stores the movie metadata, computes the similarities values and persists the results in a database. The backend is based on Java frameworks and a CouchDB database. Content Similarity Search & Discovery exposes a RESTful API via JSON. The algorithm has been tuned and validated on two VOD catalogues, a French-based one and a US-based one. The preparation of the dataset requires a deep analysis of the input VOD catalogue data requiring close communication between the catalogue provider and Technicolor as the Specific Enabler provider. At this stage, a first implementation of the industrialisation of the SE is not yet available. Some improvements to the algorithm are always in progress.

### Examples of use

## Content Similarity Search & Discovery

The algorithm has been implemented in two applications by Technicolor:

- GraphMap application – from a movie, get the list of similar movies
- Cocktail Search application – propose a list of pre-defined “movie cocktails” based on movie clustering.

These two applications are available in the FI-CONTENT2 prototype developed by Technicolor and submitted to an end-user evaluation in Brittany.

### Terms and conditions of use

#### *Licence type*

- Open source                      no
- Proprietary                        yes
- Evaluation licence                yes

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          no
- Distribution allowed            no
- Include copyright                must
- Include original                  not required
- State changes                    not required
- Disclose source code            not required

#### *Licence fee*

- Upfront payment / Recurring payment / Per use payment  
(subject to agreement, negotiable)

#### *Licence summary*

The provisions of the collaboration agreement shall govern relationships between projects of the FI-PPP. Some of the main provisions are summarised below for information only. Please refer to the Collaboration Agreement that exhaustively set forth the applicable rights and obligations and that shall prevail in case of conflict with the following summary. For the avoidance of doubt such Access rights do not imply any obligation by the granting Party to provide any support or maintenance for the Software, nor bear any responsibility for any claims for defects in the Software.

During Phase 3, All project partners which are not part of the Collaboration Agreement but have been approved by the European Commission can use it during the term of the FI-PPP projects for testing and experimentation, and under the term of the evaluation license agreement.

For any other use, especially commercial use, access to Technicolor Content Similarity Search & Discovery, will be granted on a case-by-case basis upon written request and after negotiations.

#### *Copyright statement*

Copyright © 2013 Technicolor

#### *Full licence*

## Content Similarity Search & Discovery

The evaluation licence agreement is available on FI-CONTENT2 website.

### Performance requirements

The machine required to host Content Similarity Search & Discovery runs under Linux Ubuntu 12.04 64-bit and must be equipped with at least 8GB of memory. The tests have been achieved with a volume of approximately 5.000 movies in a catalogue. Some adaptations may be necessary to support larger catalogues. As mentioned in the description, a study step of the format of the metadata provided by the third-party is necessary to know, for instance, the categorisation chosen by the content provider. Depending on the results of this study, some adaptations on the algorithm may be necessary.

### Further documentation

[http://wiki.mediafi.org/doku.php/ficontent.socialtv.roadmap#content\\_similarity](http://wiki.mediafi.org/doku.php/ficontent.socialtv.roadmap#content_similarity)

### Contact information

Nathalie Cabel

[Nathalie.cabel@technicolor.com](mailto:Nathalie.cabel@technicolor.com)

## 3.2 - City Guide Platform

### 3.2.1 - *Orange – Reperio (Recommendation Services)*

#### Reperio (Recommendation Services)

##### Owner/developer

Orange

##### What it does

Reperio is a recommendation engine including collaborative and thematic functionality (recommendation, ranking, prediction). Reperio is content-agnostic and thus can be used on any on collaborative data (logs) or on thematic data (catalogues).

##### How it works

Reperio takes a snapshot of the application service data and uses this to generate similarity matrixes, either on collaborative data (logs) or on thematic data (catalogues). The logs are usually actions of users on items (of the application service) with an optional rating (i.e. user U rated movie M with a rating of 1/5). The catalogue is usually a description of items with further descriptors for metadata association with an attribute and its value (e.g. 'actor: Jack Nicholson' is a descriptor associating the attribute 'actor' with its value 'Jack Nicholson').

Based on this data, Reperio is able to generate similarity matrixes (usually item/item where row and columns are items and cells hold the similarity value of the two items, with a range usually going from 0 = not similar at all to 1 completely similar). The collaborative part is sensitive to cold starts (i.e. you need logs before getting recommendations) but has great discovery capabilities (e.g. users who like 'Rambo' also like 'Fame'). The thematic part doesn't offer the same discovery aspect, but is usually accurate and insensitive to cold start, provided of course that the catalogue is available.

##### What you get

###### *Delivery model*

- Hosted service                      no
- Source code                         no
- WAR archive                        yes

###### *Languages used*

- JavaScript                         no
- PHP                                    no
- Java                                  yes (based on a Tomcat server and a PostgreSQL database)

## Reperio (Recommendation Services)

### *Description*

Reperio is a Java servlet installed in a Tomcat container. The data is stored in a PostgreSQL database. These components can be installed on any platform (Reperio has already been installed on Linux, Windows XP/7/Server). It is delivered in a WAR archive, with documents related to the installation and use of the enabler and the scripts you need to create or migrate your database for Reperio.

### **Examples of use**

On request

### **Terms and conditions of use**

#### *Licence type*

- Open source                   no
- Proprietary                   yes
- Evaluation licence       subject to agreement

#### *Licence features*

- Commercial use           yes
- Modifications allowed   subject to agreement
- Distribution allowed     yes
- Include copyright       not required
- Include original         negotiable
- State changes           negotiable
- Disclose source code   negotiable

#### *Licence fee*

- Upfront payment (for hosted service) / Recurring payment (for ASP use)

#### *Licence summary*

Reperio is available with a one-shot licence (hosted service) or with recurring payments (ASP).

#### *Copyright statement*

© Orange 2014

#### *Full licence*

Orange will provide link to full terms and conditions in April 2014

## Reperio (Recommendation Services)

### Performance requirements

Thanks to optimised algorithms and code, Reperio is able to handle any number of users and/or items depending on hardware configuration. The key resource is the available memory (RAM).

### Further documentation

<http://wiki.mediafi.org/doku.php/ficontent.smartcity.enabler.recommendationservices>

### Contact information

Stéphane Geney  
[stephane.geney@orange.com](mailto:stephane.geney@orange.com)

### 3.2.2 - Orange: KIWANO (Virtual Mixed Reality)

## KIWANO (Virtual Mixed Reality)

### Owner/developer

Orange

### What it does

The Kiwano SE is a scalable spatial database able to manage a large number of moving objects in real time. It is able to record their location with a high frequency of updates and perform searches such as retrieving the list of neighbours of an object. It also implements communication between moving objects with methods for sending data within neighbours.

### How it works

In essence, Kiwano receives streams of location data from moving objects (e.g. users, avatars, bots, drones) and sends back notifications about the neighbourhood, updating the list of neighbours with their new positions and states. It also implements communication between moving objects with methods for sending data within neighbours.

An example of usage of Kiwano, called Hybrid Earth, is provided as a web application and an Android mobile application. Hybrid Earth shows how Kiwano can be used to manage avatars evolving in a virtual world (based on Google Street View) and in the reality (based on geolocation of smartphones).

### What you get

#### *Delivery model*

Hosted service	yes
Source code	yes

#### *Languages used*

JavaScript	yes
PHP	no
Java (Android)	yes
Python	yes

#### *Description*

The Kiwano SE (developed in Python) is provided as a hosted service: it can be accessed via an API described on the SE's website <http://kiwano.li>. Source code will only be provided for the Hybrid Earth application (HTML/JavaScript web application and Android application) to give developers an example of usage of Kiwano.

## KIWANO (Virtual Mixed Reality)

### Examples of use

Current examples of use of Kiwano:

- Manycraft: scaling Minecraft to millions <http://manycraft.net>
- Hybrid Earth: mixed reality at planet scale <http://hybridearth.net>

### Terms and conditions of use

#### *Licence type*

Open source	yes
Proprietary	yes
Evaluation licence	subject to agreement

#### *Licence features*

Commercial use	yes
Modifications allowed	no
Distribution allowed	no
Include copyright	not required
Include original	not required
State changes	not required
Disclose source code	not required

#### *Licence fee*

Upfront payment / Recurring payment / Per use payment  
Free for non-commercial use within the FI-PPP

#### *Licence summary*

Usage of Kiwano is subject to a licence fee, to be negotiated. Free access to the SE is provided for testing and evaluation purposes (access to the full API, with restrictions in the volume of requests permitted). The Hybrid Earth example is provided under open source licence.

#### *Copyright statement*

© Orange 2014

#### *Full licence*

Orange will provide link to full terms and conditions in April 2014

### Performance requirements

A single Kiwano node can support several hundreds to several thousands of simultaneous users, depending on the frequency of location update messages. Moreover, the architecture is designed to scale to any number of users.



## KIWANO (Virtual Mixed Reality)

### Further documentation

- <http://mediafi.org/?portfolio=virtualmixed-reality#tab-documentation>
- <http://kiwano.li>

### Contact information

Franck Feurtey  
[franck.feurtey@orange.com](mailto:franck.feurtey@orange.com)

Arnaud Brun  
[arnaud.brun@orange.com](mailto:arnaud.brun@orange.com)

### 3.2.3 - *Thales Communications and Security: Content Sharing*

## Content Sharing

### Owner/developer

Thales Communications and Security

### What it does

Content Sharing provides the ability to synchronise online content such as text and images seamlessly between the content provider and a high number of users, keeping it up to date. In addition to this feature, it allows direct communications between users with no access to an operator network. This feature adds the possibility of sharing/synchronising content directly between users (e.g. those wishing to avoid roaming). Another feature provided by this enabler adds geographical context to the content to be distributed in order to either target or restrict the content dissemination to a specific geographical area.

### How it works

The Content Sharing Specific Enabler is provided as two pieces of software: a content server and an Android library allowing the development of Android mobile applications. The content server delivers Atom feeds using the AtomPub protocol. It makes use of WebSockets to notify clients of the availability of new content to which they have subscribed. On the client side, the Android library allows the use of content synchronisation with the server and content exchange and synchronisation between users, exploiting direct communications. Both the content server and library include advanced synchronisation mechanisms as well as geographical capabilities allowing geographical content distribution.

### What you get

#### *Delivery model*

- |                   |     |
|-------------------|-----|
| • Hosted service  | no  |
| • Source code     | no  |
| • Android Library | yes |
| • Server software | yes |

#### *Languages used*

- |              |     |
|--------------|-----|
| • JavaScript | yes |
| • PHP        | no  |
| • HTML 5     | yes |
| • Java       | yes |
| • Python     | yes |

#### *Description*

Content Sharing is a piece of software composed of two pieces: a server part provided with its API, and an Android library that allows applications to be built which take advantage of content sharing features.

## Content Sharing

### Examples of use

Content Sharing can be used to support a number of use cases relating to the distribution of news between users:

- Users A, B, and C register on the Olympic games website <http://www.sochi2014.com>
- Users A, B, and C subscribe to the curling news section
- When users are connected to the internet, each time new curling content is published the website server notifies them of its availability
- The terminal connects to the server and gets the content
- If user A adds a comment related to one piece of curling content, then users B and C are notified and get the content directly
- Now user C enters the subway and loses their internet connection
- While user C is travelling, if new curling content is made available both users A and B get it, and if user B then enters the subway and is in the vicinity of user C then user B's terminal will seamlessly connect directly to user C's terminal and send new content to it; users B and C can also add and exchange comments on the news.
- User C leaves the subway and is connected to the internet again at which point the comments generated in the subway are then synchronised with the server.
- User A's terminal is then notified about this new content and can download it.
- If user C passes in the vicinity of a curling store then an advertisement about curling equipment is sent to their terminal

Further examples of use:

[https://bitbucket.org/florian\\_cosnier/demoappscg/downloads/DemoAppSCG.apk](https://bitbucket.org/florian_cosnier/demoappscg/downloads/DemoAppSCG.apk)

### Terms and conditions of use

#### *Licence type*

- Open source                      no
- Proprietary                        yes
- Evaluation licence                yes

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          no
- Distribution allowed            no
- Include copyright                must
- Include original                 must
- State changes                    not required
- Disclose source code            must

#### *Licence fee*

During phase 3, the Specific Enabler is provided for free, subject to users providing feedback on its use. Upon completion of phase 3, the use of this software is subject to a licence fee that will depend on the customer and the conditions of use.

## Content Sharing

### *Licence summary*

Content Sharing is available for free during the phase 3 experimentation phase.

### *Copyright statement*

Copyright © 2013 Thales Communications and Security

### *Full licence*

<http://www.binpress.com/license/view//030cbff7f19819112b730e291abef99b>

### **Performance requirements**

- The content server is based on Apache and can handle hundreds of users
- The client part is an Android library and provides infrastructure and direct communication features, which have been tested with up to ten servers without issues
- The direct communication part has been tested in a topology of four terminals and the scalability issues are currently under investigation
- The server itself does not need much resource in the way of hardware to run, however for scalability issues, it needs high performance CPU and memory.
- The Android library has been used to build proof-of-concept applications running on Samsung Galaxy phones

### **Further documentation**

- <http://mediafi.org/?portfolio=content-sharing#tab-documentation>
- [https://bitbucket.org/florian\\_cosnier/demoappscg/wiki/Home](https://bitbucket.org/florian_cosnier/demoappscg/wiki/Home)

### **Contact information**

Farid Benbadis

[farid.benbadis@thalesgroup.com](mailto:farid.benbadis@thalesgroup.com)

### 3.2.4 - *Pixelpark: Social Network*

## Social Network

### Owner/developer

Pixelpark

### What it does

The Social Network SE is middleware that can be used to create a social network, either temporarily or permanently for a group of users. It is able to provide social interactions in the digital world separate from the major, proprietary social networks (e.g. Facebook) and as such is best suited for intranet-based or extranet-based networking. Users of the Social Network SE keep full control over their data domain.

### How it works

The Social Network SE comprises two parts: a server and a client component. The server component is a web application that creates the authentication layer towards an external authentication component (e.g. the Keyrock Identity Management Generic Enabler) and serves the web pages. Additionally, a noSQL CouchDB database is used to provide a data exchange with the clients. The client is aimed towards mobile devices that can either be a mobile web browser of the latest generation or an Android App. The Android App serves the specific use case of providing offline functionality in environments with unreliable internet access (e.g. public mass events).

### What you get

#### *Delivery model*

- Hosted service                      no
- Source code                            yes

#### *Languages used*

- JavaScript                            yes
- PHP                                        no

#### *Description*

Both the client and server side components are written in Javascript. The client side component makes use of the AngularJS framework. A PouchDB module provides data replication towards the server sided CouchDB. A Node.js server is used to provide the authentication for the web client that runs in a standard browser. The app is created using the PhoneGap mobile framework that outputs a native mobile application embedding the web pages in a shell.

### Examples of use

- The Social Network SE was used in tests with pupils from a Cologne school
- The same group will use it in the Carnival experiment in Cologne

## Social Network

### Terms and conditions of use

#### *Licence type*

- Open source                      yes
- Proprietary                        no
- Evaluation licence                no

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          yes
- Distribution allowed            yes
- Include copyright                must
- Include original                 negotiable
- State changes                    not required
- Disclose source code          not required

#### *Licence fee*

- Free

#### *Licence summary*

The Social Network SE is available under the MIT open source licence. As such it can be used in commercial projects if the copyright is maintained. It is available free of charge.

#### *Copyright statement*

Copyright © 2014, Pixelpark

#### *Full licence*

The MIT License (MIT)

Copyright © 2014, Pixelpark

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. The Software is provided "as is", without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from out of or in connection with the Software or the use or other dealings in the Software.

## Social Network

### Performance requirements

The server side runs on an equivalent of an Amazon m1.medium virtual machine and was tested with 20 users. We expect it to support up to 1.000 users. The client side runs on every smart phone browser that comprises the Chromium engine, version 30 or higher as a web application. The client application needs Android version 4.4.2 ('KitKat') and was tested on a Nexus 5 smartphone.

### Further documentation

- <http://mediafi.org/?portfolio=social-network-enabler#tab-documentation>
- <https://github.com/pixelpark/ppnet/blob/master/README.md>

### Contact information

Dirk Krause  
[dirk.krause@pixelpark.com](mailto:dirk.krause@pixelpark.com)

### 3.2.5 - Fraunhofer FOKUS: Open City Database

## Open City Database

### Owner/developer

Fraunhofer FOKUS

### What it does

The Open City Database (OCDB) is an open source database for any smart city related data (e.g. points of interest, open city data and related media from various sources). Besides its database functionality the OCDB provides a comprehensive API to create, modify and request data sets for their integration with smart city guide apps or any other application or service that may take advantage of open city data.

### How it works

The OCDB is designed to build the backend basis for cross platform apps (e.g. mobile web apps) that make use of open tourist and open city data. Applications can access the OCDB data using the REST endpoint. The data format used by OCDB and exposed through the API is JSON. Even user-generated content (UGC) through bespoke apps as well as further open data (e.g. DBpedia) can be managed by the OCDB.

### What you get

#### *Delivery model*

- Hosted service                    yes
- Source code                        yes

#### *Languages used*

- JavaScript                         yes
- JSON                                yes

#### *Description*

The OCDB is an open source solution with meteor framework server side component. The OCDB can be used as hosted service or a new instance can be installed. It provides a comprehensive API to create, modify and request open city data. Furthermore, it allows for the integration of UGC through bespoke applications

### Examples of use

- FIC2 Smart City Guide Reference implementation <http://scg.fokus.fraunhofer.de:3000>
- <http://scg.fokus.fraunhofer.de:3000/http://wiki.mediafi.org/doku.php/ficontent.smartcity.enabler.opencitydatabase>
- <https://github.com/fraunhoferfokus/OCDB>

### Terms and conditions of use



## Open City Database

### *Licence type*

- Open source                    yes
- Proprietary                    subject to agreement
- Evaluation licence            no

### *Licence features*

- Commercial use                yes
- Modifications allowed        yes
- Distribution allowed         yes
- Include copyright             must
- Include original                not required
- State changes                 not required
- Disclose source code         not required

### *Licence fee*

- Free

### *Licence summary*

OCDB is available and free to use to all under the terms of the Apache 2.0 open source licence.

### *Copyright statement*

Copyright © 2013 Fraunhofer Institute for Open Communication Systems FOKUS

### *Full licence*

<http://www.apache.org/licenses/LICENSE-2.0>

### **Performance requirements**

- A token is required for requests to the Open City Database
- Every request has to be '?token=[valid token / user id]' at the END

### **Further documentation**

- <http://wiki.mediafi.org/doku.php/ficontent.smartcity.enabler.opencitydatabase>
- <https://github.com/fraunhoferfokus/OCDB>

### **Contact information**

Robert Seeliger

[robert.seeliger@fokus.fraunhofer.de](mailto:robert.seeliger@fokus.fraunhofer.de)

### 3.2.6 - Fraunhofer FOKUS: Recommendation as a Service

## Recommendation as a Service

### Owner/developer

Fraunhofer FOKUS

### What it does

The Recommendations as a Service (RaaS) SE provides the ability to create a professional recommendation engine with just a few mouse clicks and no programming skills. This SE can maintain your item and user data and will host your recommendation engine as a service in the cloud or on your own server infrastructure. Thereby, decision makers can choose whether to use ratings, likes, check-ins or implicit feedback, such as clicks or consumption time. In addition, they can adjust the way the personalisation works by selecting from a wide range of well-explained algorithms.

### How it works

You can start your own personalised recommendation system on a website, using the admin interface, by following three simple steps:

- **Allocate data**  
Import existing data, connect external databases or create new users, items and relations between them. The necessary data can be stored in the offered graph database that provides efficient access and a high reliability by distributing the data. It is protected and anonymous, so only the end customer concerned has full control over the data.
- **Create recommendation engines**  
Create and adjust engines presenting different well-explained recommendation approaches and select from a wide range of algorithms.
- **Build services on top:**  
Use an automatically-offered example user interface or connect your custom application via dynamically generated APIs to the engine.

After following these steps, you can analyse your success by retrieving visual illustrations of usage data and customer statistics.

### What you get

#### *Delivery model*

- Hosted service                      yes
- Source code                         no

#### *Languages used*

- JavaScript                         yes
- JSON/XML                         yes
- Java                                 yes

#### *Description*

RaaS provides a hosted recommendation engine ready for tailor-made implementation into your app or service. A set of algorithms is available (e.g. collaborative filters) and can be used via the admin

## Recommendation as a Service

interface or via the built-in RESTful API, which is based on JSON or XML.

### Examples of use

- Used for second screen and in app recommendation within the first German HybridTV Portal on DVB-T
- Recommendation engine in personalized electronic programme guide (EPG) offer called MOVISTO from rtv.de, part of Bertelsmann group <http://www.rtv.de/movisto-registrierung.html>

### Terms and conditions of use

#### *Licence type*

- Open source                      no
- Proprietary                        subject to agreement
- Evaluation licence                yes (30 days)

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          yes (integrate own algorithms)
- Distribution allowed            yes
- Include copyright                must
- Include original                 not required
- State changes                  not required
- Disclose source code          not required

#### *Licence fee*

- Upfront payment / Recurring payment / Per use payment

#### *Licence summary*

All project partners can use RaaS during the term of the FI-PPP projects for testing and experimentation. For any other use (e.g. commercial use) of RaaS, please contact us.

#### *Copyright statement*

Copyright © 2013 Fraunhofer Institute for Open Communication Systems FOKUS

#### *Full licence*

Full license may be provided on request and must be negotiated on a case by case basis.

### Performance requirements

- Integration of own algorithms as new modules written in Java is possible
- Standard version of hosted solution can serve up to 1,000 users, for larger customer bases

## Recommendation as a Service

server extensions/ upgrades are recommended

- Through the RESTful API you are able to use RaaS in a wide range of applications and devices including SmartTVs. HbbTV, iOS, Android and web apps

### Further documentation

Further documentation will be provided on the project wiki.

### Contact information

Robert Seeliger

[robert.seeliger@fokus.fraunhofer.de](mailto:robert.seeliger@fokus.fraunhofer.de)

## 3.3 - Games Platform

### 3.3.1 - ETHZ: Leaderboard

## Leaderboard

### Owner/developer

ETHZ

### What it does

The main functionality that the Leaderboard SE provides, is the storage of high scores and the retrieval of a sorted list of high scores.

### How it works

Leaderboard is a combination of services plugged together to provide the storage of high scores and the retrieval of a sorted list of high scores. As such it is a construction of open source software solutions with their respective documentation found elsewhere. It provides an API to the user. Leaderboard will make use of the 'Identity Management - KeyRock' GE (<http://catalogue.fi-ware.org/enablers/identity-management-keyrock>) to authenticate a user when submitting a score. The requests are a direct access to the database. Be aware that in this version, the high score value submitted by users can easily be manipulated or hacked.

### What you get

#### *Delivery model*

- Hosted service no
- Source code yes

#### *Languages used*

- JavaScript yes
- PHP no
- Java yes
- HTML yes

#### *Description*

Leaderboard is a set of MySQL commands wrapped in Java. Both a standalone server and a plugin for Smartfoxserver2x are implemented. A simple HTML/JavaScript client example is also provided.

### Examples of use

- Augmented Resistance game  
[http://youtu.be/C\\_8wRZsyDs](http://youtu.be/C_8wRZsyDs)
- Any other game using high scores

## Leaderboard

### Terms and conditions of use

#### *Licence type*

- Open source                      yes
- Proprietary                       no
- Evaluation licence               no

#### *Licence features*

- Commercial use                 yes
- Modifications allowed         yes
- Distribution allowed          yes
- Include copyright               must
- Include original                 not required
- State changes                  not required
- Disclose source code          not required

#### *Licence fee*

- Free

#### *Licence summary*

The MIT License is a permissive free software license, meaning that it permits reuse within proprietary software provided all copies of the licensed software include a copy of the MIT License terms.

#### *Copyright statement*

Copyright © 2013 ETH Zurich

#### *Full licence*

<http://mediafi.org/?portfolio=leaderboard-se#tab-terms-conditions>

### Performance requirements

The current implementation just wraps mySQL commands and does not yet feature advanced caching strategies that would be required for thousands of users.

### Further documentation

<http://mediafi.org/?portfolio=leaderboard-se#tab-documentation>

### Contact information

Marcel Lancelle

[marcel.lancelle@inf.ethz.ch](mailto:marcel.lancelle@inf.ethz.ch)

### 3.3.2 - ETHZ: Reality Mixer – Camera Artefact Rendering

## Reality Mixer – Camera Artefact Rendering

### Owner/developer

ETHZ

### What it does

This client-side code modifies the virtual rendered content to match the camera image more closely in an AR context to provide more realistic appearance.

### How it works

Motion blur, noise and colours are measured or estimated and applied to the virtually rendered image.

### What you get

#### *Delivery model*

- Hosted service no
- Source code yes
- Unity package yes

#### *Languages used*

- JavaScript no
- PHP no
- C# yes
- GLSL yes

#### *Description*

The implementation is provided as a client-side Unity package that can simply be imported and used in a Unity project.

### Examples of use

- Augmented Resistance  
[http://youtu.be/\\_C\\_8wRZsyDs](http://youtu.be/_C_8wRZsyDs)
- Any other AR application written in Unity

### Terms and conditions of use

#### *Licence type*

- Open source yes
- Proprietary no
- Evaluation licence no

## Reality Mixer – Camera Artefact Rendering

### *Licence features*

- Commercial use            yes
- Modifications allowed    yes
- Distribution allowed      yes
- Include copyright          must
- Include original            not required
- State changes             not required
- Disclose source code      not required

### *Licence fee*

- Free

### *Licence summary*

The MIT License is a permissive free software license, meaning that it permits reuse within proprietary software provided all copies of the licensed software include a copy of the MIT License terms.

### *Copyright statement*

Copyright (c) 2013 ETH Zurich

### *Full licence*

<http://mediafi.org/?portfolio=reality-mixer-camera-artefact-rendering-se#tab-terms-conditions>

### **Performance requirements**

Hardware accelerated OpenGL is required

### **Further documentation**

<http://mediafi.org/?portfolio=reality-mixer-camera-artefact-rendering-se#tab-documentation>

### **Contact information**

Fabio Zünd  
[fzuend@inf.ethz.ch](mailto:fzuend@inf.ethz.ch)

Marcel Lancelle  
[marcel.lancelle@inf.ethz.ch](mailto:marcel.lancelle@inf.ethz.ch)



### 3.3.3 - *ETHZ: Reality Mixer – Reflection Mapping*

## Reality Mixer – Reflection Mapping

### Owner/developer

BLRK, ETHZ, DFKI

### What it does

All visually-oriented SEs of the Reality Mixer group measure camera properties and adapt the virtual objects to fit to the camera image background visually. The Reflection Mapping SE utilises a light probe to extract a sphere map from the camera image, which contains the environmental lighting conditions. This sphere map will be used to apply an appropriate lighting model to rendered virtual objects. Thus, the additional virtual objects are incorporated into the resulting image in a very seamless fashion leading to a more realistic experience of mixed reality applications.

### How it works

It provides a new level of realistic augmented reality rendering with the following steps: it pinpoints the centre and radius of the material sphere source relative to the augmented reality marker. Then it assigns the camera image of the sphere to the source texturing of the target object(s) and performs spherical environment reflection mapping to the shading of the target object(s).

### What you get

#### *Delivery model*

- Hosted service                   no
- Source code                     yes
- Unity package                 yes

#### *Languages used*

- JavaScript                     yes
- PHP                             no
- UnityScript                 yes

#### *Description*

The implementation includes either an Xflow node as JavaScript library or a cross platform Unity script/package.

### Examples of use

- Augmented Resistance as part of the “Tabletop Augmented Reality Games” scenario  
[http://wiki.mediafi.org/doku.php/ficontent.gaming.scenarios#tabletop\\_augmented\\_reality\\_games](http://wiki.mediafi.org/doku.php/ficontent.gaming.scenarios#tabletop_augmented_reality_games)
- Star Tours as part of the “Seamless Augmented Reality on the Web” scenario  
[http://wiki.mediafi.org/doku.php/ficontent.gaming.scenarios#seamless\\_augmented\\_reality\\_on\\_the\\_web](http://wiki.mediafi.org/doku.php/ficontent.gaming.scenarios#seamless_augmented_reality_on_the_web)

## Reality Mixer – Reflection Mapping

### Terms and conditions of use

#### *Licence type*

- Open source                      yes
- Proprietary                        no
- Evaluation licence                no

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          yes
- Distribution allowed            yes
- Include copyright                must
- Include original                  not required
- State changes                    not required
- Disclose source code            not required

#### *Licence fee*

- Free

#### *Licence summary*

The Reality Mixer – Reflection Mapping SE is available under MIT licensing

#### *Copyright statement*

Copyright © 2013 DFKI - German Research Center for Artificial Intelligence

<http://www.dfki.de/web/forschung/asr>

and ETH Zurich

<http://graphics.ethz.ch>

#### *Full licence*

Template: <http://opensource.org/licenses/MIT>

Xflow: <https://github.com/stlemme/xml3d-xflmix/blob/master/LICENSE>

Unity: see enclosed license file in package

### Performance requirements

- The Xflow implementation runs in any XML3D capable browser with WebGL support.
- All current maintained versions of Unity are supported by this SE. The latest Unity release at time of writing is v4.3.4.

### Further documentation

- <http://mediafi.org/?portfolio=reality-mixer-reflection-mapping-se>
- <http://wiki.mediafi.org/doku.php/ficontent.gaming.enabler.realitymixer.reflectionmapping>

## Reality Mixer – Reflection Mapping

### Contact information

For technical information:

Kenny Mitchell

[kenny.mitchell@disneyresearch.com](mailto:kenny.mitchell@disneyresearch.com)

Stefan Lemme

[stefan.lemme@dfki.de](mailto:stefan.lemme@dfki.de)

For licensing information:

Marcel Lancelle

[marcel.lancelle@inf.ethz.ch](mailto:marcel.lancelle@inf.ethz.ch)

### 3.3.4 - DFKI: Augmented Reality – Marker Tracking

## Augmented Reality – Marker Tracking

### Owner/developer

DFKI

### What it does

All SEs of the Augmented Reality group provide various tracking methods to enable augmented reality (AR) applications. The Marker Tracking SE utilises AR markers to retrieve camera pose information through Xflow. This extends XML3D with the capabilities to apply the matching transformation to 3D scene content and render them onto respective markers in a web-based environment. The Marker Tracking SE follows the declarative approach of XML3D and is real-time capable.

### How it works

It performs AR marker detection on video streams. Therefore, it might utilise the `Navigator.getUserMedia()` API of the browser to gain access to connected camera devices and take advantage of third-party JavaScript libraries to perform the image processing algorithms. Finally, it exposes the current camera pose to XML3D according to a given fiducial marker.

### What you get

#### *Delivery model*

- Hosted service no
- Source code yes

#### *Languages used*

- JavaScript yes
- PHP no

#### *Description*

The implementation includes an Xflow node as JavaScript library. It might be completely obsolete after the Augmented Reality GE from FI-WARE is released.

### Examples of use

- Star Tours as part of the “Seamless Augmented Reality on the Web” scenario  
[http://wiki.mediafi.org/doku.php/ficontent.gaming.scenarios#seamless\\_augmented\\_reality\\_on\\_the\\_web](http://wiki.mediafi.org/doku.php/ficontent.gaming.scenarios#seamless_augmented_reality_on_the_web)

## Augmented Reality – Marker Tracking

### Terms and conditions of use

#### *Licence type*

- Open source                    yes
- Proprietary                    no
- Evaluation licence            no

#### *Licence features*

- Commercial use                yes
- Modifications allowed        yes
- Distribution allowed         yes
- Include copyright              must
- Include original                not required
- State changes                 not required
- Disclose source code         not required

#### *Licence fee*

- Free

#### *Licence summary*

The Augmented Reality – Marker Tracking SE is available under MIT licensing

#### *Copyright statement*

Copyright © 2013 DFKI - German Research Center for Artificial Intelligence  
<http://www.dfki.de/web/forschung/asr>

#### *Full licence*

Template: <http://opensource.org/licenses/MIT>  
Xflow: <https://github.com/stlemme/xml3d-xflar/blob/master/LICENSE>

### Performance requirements

The Xflow implementation runs in any XML3D-capable browser with WebGL support

### Further documentation

- Specification:  
<http://wiki.mediafi.org/doku.php/ficontent.gaming.enabler.augmentedreality.markertracking>
- FIcontent Catalog  
<http://mediafi.org/?portfolio=augmented-reality-marker-tracking-se>

## Augmented Reality – Marker Tracking

### Contact information

Stefan Lemme  
[stefan.lemme@dfki.de](mailto:stefan.lemme@dfki.de)

### 3.3.5 - *BLRK/ETHZ: Augmented Reality – Fast Feature Tracking*

## Augmented Reality – Fast Feature Tracking

### Owner/developer

BLRK/ETHZ

### What it does

All specific enablers of the Augmented Reality (AR) group provide various tracking methods to enable AR applications. The Fast Feature Tracking SE learns targets by colour and then matches the centre of a colour area (for example a coloured football or road sign) in the camera image to retrieve the relative camera pose information. This extends an application with the capabilities to apply the matching transformation to 3D-scene content and render them onto respective targets.

### How it works

With the Fast Feature Tracking SE you will be able to easily create applications with basic markerless augmented reality functionality. With this SE you can learn the colour of targets on the fly in an application and then track the centre and size of the target for camera relative placement of animated interactive graphics such as virtual characters or vehicles. It provides a markerless augmented reality tracking method (blob tracking) with the following steps:

- Pinpoint the centre and radius of the material sphere source relative to the Augmented Reality marker
- Assign the camera image of the sphere to the source texturing of the target object(s)
- Perform spherical environment reflection mapping to the shading of the target object(s)

Implementation is via a cross platform Unity script/package and reference algorithm provided in Skye Wars app source code.

### *Delivery model*

- |                         |     |
|-------------------------|-----|
| • Hosted service        | no  |
| • Source code           | yes |
| • iTunes App Store demo | yes |

### *Languages used*

- |              |     |
|--------------|-----|
| • JavaScript | no  |
| • PHP        | no  |
| • C#         | yes |
| • C/C++      | yes |
| • GLSL       | yes |

### *Description*

The implementation includes a Unity C# script (controlling GPU process and editor access) and a native plugin performing the fast tracking method. It is suitable for implementation in XFlow and delivery as a generic enabler.

## Augmented Reality – Fast Feature Tracking

### *Examples of use*

- Full source to the Skye Wars iTunes app is provided on GitHub  
<https://github.com/fzuendeth/FIcontent.Gaming.Enabler.AugmentedReality.FastFeatureTracking>
- <https://itunes.apple.com/us/app/skye-wars/id672970985?ls=1&mt=8>

### **Terms and conditions of use**

#### *Licence type*

- Open source                      yes

#### *Licence features*

- Commercial use                yes
- Modifications allowed        yes
- Distribution allowed         yes
- Include copyright              must
- Include original                not required
- State changes                 not required
- Disclose source code         not required

#### *Licence fee*

- Free

#### *Licence summary*

The Augmented Reality – Fast Feature Tracking SE is provided under MIT license.

#### *Copyright statement*

Copyright © 2013 ETH Zurich and The Walt Disney Company Ltd

#### *Full licence*

<http://opensource.org/licenses/MIT>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. The Software is provided “as is”, without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and noninfringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with the Software or the use or other dealings in the Software.



## Augmented Reality – Fast Feature Tracking

### Performance requirements

This SE runs on PC, fast Android device and is further highly optimized for Apple devices

### Further documentation

- [http://mediafi.org/?portfolio=augmented-reality-fast"-tracking-se](http://mediafi.org/?portfolio=augmented-reality-fast)
- <http://wiki.mediafi.org/doku.php/ficontent.gaming.enabler.augmentedreality.fastfeaturetracking>

### Contact information

Marcel Lancelle  
[marcel.lancelle@inf.ethz.ch](mailto:marcel.lancelle@inf.ethz.ch)

Kenny Mitchell  
[kenny.mitchell@disneyresearch.com](mailto:kenny.mitchell@disneyresearch.com)

### 3.3.6 - DRZ: Game Synchronization

## Game Synchronization

### Owner/developer

Disney Research Zurich

### What it does

This Game Synchronization SE provides functionality to synchronise the game world. We consider the following taxonomy. For the connection of the parties, we consider peer to peer (P2P) and server to client (S2C). This enabler serves different networking models, such as RTS Lockstep (P2P) and Authoritarian S2C.

### How it works

This enabler has to be integrated to the host application (the game) and deployed on the devices participating in the game. In both P2P and S2C scenarios, the enabler will deal with the synchronization

### What you get

#### *Delivery model*

- Hosted service                   no
- Source code                    yes
- Specify others                 Unity package, SFS2X Extension

#### *Languages used*

- JavaScript                    no
- PHP                            no
- C#                              yes

#### *Description*

This enabler comes as C# library which is included into the game implementation. We provide a Unity package as well as an extension for Smart Fox Server X2.

### Examples of use

- Augmented Resistance RTS
- Spider Demo

## Game Synchronization

### Terms and conditions of use

#### *Licence type*

- Open source                      yes
- Proprietary                        no
- Evaluation licence                no

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          yes
- Distribution allowed            yes
- Include copyright                must
- Include original                 not required
- State changes                    not required
- Disclose source code          not required

#### *Licence fee*

- Free

#### *Licence summary*

The MIT License is a permissive free software license, meaning that it permits reuse within proprietary software provided all copies of the licensed software include a copy of the MIT License terms.

#### *Copyright statement*

Copyright © 2013 Disney Research Zurich

#### *Full licence*

<http://mediafi.org/?portfolio=game-synchronization-se#tab-terms-conditions>

### Performance requirements

A crucial requirement of the RTS-Lockstep model is that the game simulation must be deterministic, i.e. the same inputs will always produce the same output on all devices you plan to support. If this requirement is not met, the simulations will diverge, leading to different states of the game on different clients. Recovering from such de-synchronization is often complicated. If your game cannot be simulated deterministically, consider changing to a different networking model.

## Game Synchronization

### Further documentation

- <http://mediafi.org/?portfolio=game-synchronization-se#tab-documentation>
- <http://wiki.mediafi.org/doku.php/ficontent.gaming.enabler.gamesynchronization>

### Contact information

Mattia Ryffel

[mattia.ryffel@disneyresearch.com](mailto:mattia.ryffel@disneyresearch.com)

### 3.3.7 - Gobo: Spatial Matchmaking

## Spatial Matchmaking

### Owner/developer

Studio Gobo Ltd

### What it does

Spatial Matchmaking facilitates the finding of geographically local players in a game. Finding players who are local is a common problem in massively multiplayer online games.

### How it works

Players connect to the Spatial Matchmaking service and say that they are available and looking to be matched with someone. To do this they provide some information about where they are (typically a longitude and latitude). Additional filters can also be provided such as the name of a wireless network, or indeed any parameter that a potential partner must also match. Upon registration, the service looks for good matches currently available. If there are any, it makes them. If not, the player is placed in a queue to wait for a match to arrive.

### What you get

#### *Delivery model*

- Hosted service no
- Source code yes

#### *Languages used*

- JavaScript no
- PHP no
- Java yes

#### *Description*

The SE is provided as source code as a starting point for developing a robust, application specific solution. It is written in Java (EE), and implemented as a simple REST service.

### Examples of use

The Spatial Matchmaking SE comes with an example use, written in Unity. This is included in the GitHub repository.

## Spatial Matchmaking

### Terms and conditions of use

#### *Licence type*

- Open source                      yes
- Proprietary                        no
- Evaluation licence                no

#### *Licence features*

- Commercial use                  yes
- Modifications allowed          yes
- Distribution allowed            yes
- Include copyright                must
- Include original                  not required
- State changes                    not required
- Disclose source code            not required

#### *Licence fee*

- Free

#### *Licence summary*

The Spatial Matchmaking is released under the MIT license. Under that license it is possible to use, change, redistribute, for commercial or non-commercial purposes. The original copyright and permission notice must be included with the source code. No other warranty or any kind is provided with the software.

#### *Copyright statement*

Copyright © Studio Gobo Ltd 2013-2014

#### *Full licence*

<http://wiki.mediafi.org/doku.php/ficontent.gaming.enabler.spatialmatchmaking.termsandconditions>

### Performance requirements

The SE expects to run on a server within an Apache Tomcat container. It is provided as sample code to be adapted for high performance situations, but in its current form is capable of dealing with hundreds of requests on any reasonable specification server.

### Further documentation

<http://wiki.mediafi.org/doku.php/ficontent.gaming.enabler.spatialmatchmaking>

## Spatial Matchmaking

### Contact information

Studio Gobo Ltd  
[info@studiogobo.com](mailto:info@studiogobo.com)

## 4 - CONCLUSION

This document has described each FI-CONTENT2 Specific Enabler (SE) that will be available for use by phase 3 projects. It has described what each enabler does, how it can be used, in what format it is available for use, and the terms and conditions for its use.

This document is part of an ongoing process in which partners are working towards further releases. They will continue to update the information contained herein during the remainder of FI-CONTENT2 to ensure the licence terms are as consistent and favourable to phase 3 projects as is possible within the commercial constraints under which all partners must operate.

*end of the document*