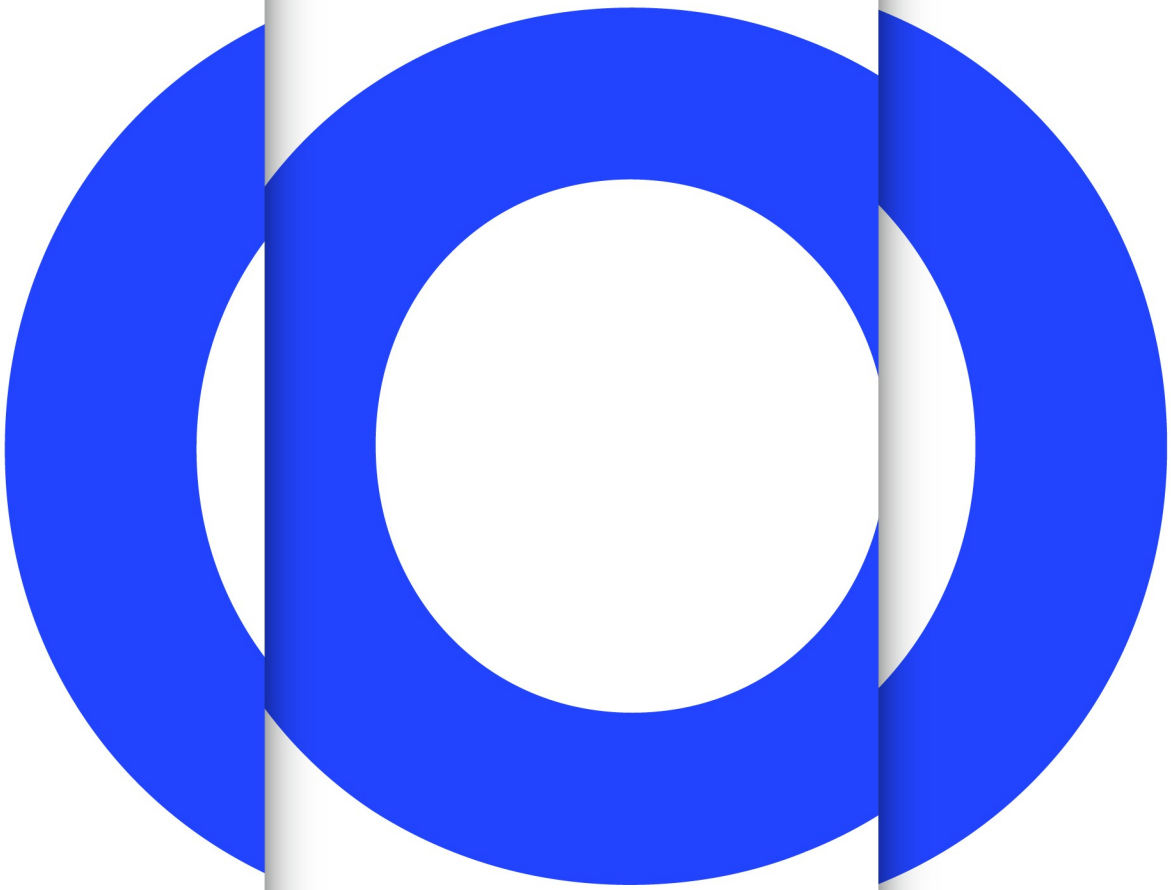
EBU

OPERATING EUROVISION AND EURORADIO



EBU

OPERATING EUROVISION AND EURORADIO

Oscied

Open source cloud infrastructure for encoding to disctribution

Oscied basics

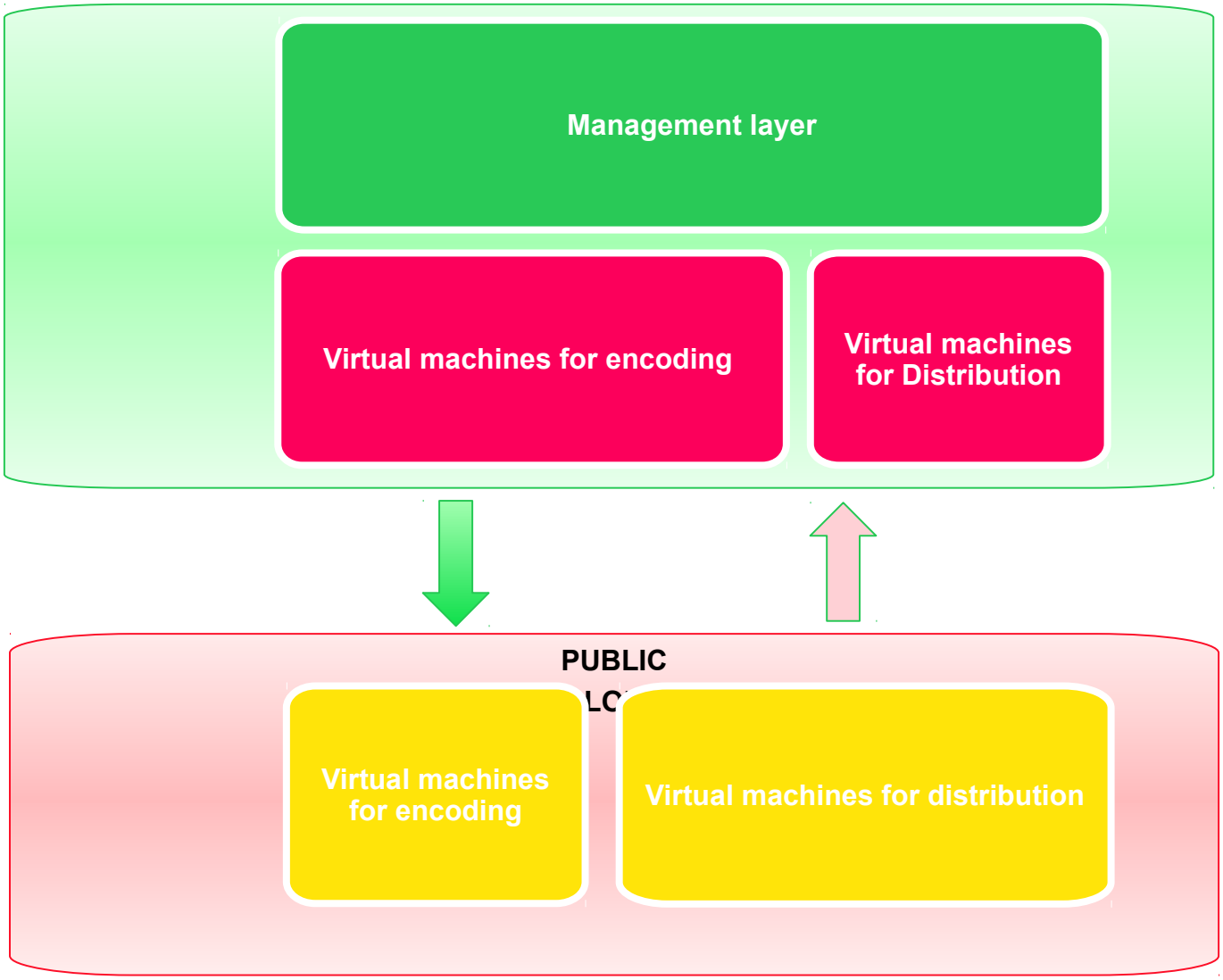
- Scalable cloud infrastructure
  - *Virtualisation of services*
  - *Elasticity: Fast up-/down-scaling*
  - Manageable services
  - *Control scaling of virtual services*
  - *Manage setting of encoding to distribution*
  - All components are open source
  - *Library of code embedded in functional service*
  - *Fast interchange of knowledge*
  - *Availability of development communities*
  - Modular development
  - *Use of interchangeable modules*
  - *Decentralised parallel development*
    - *Remote accessibility of scalable development, test and production environment*

Oscied development cycles

- Basic code (current / first development cycle)
  - *On demand video using X264 as video encoding format.*
- Current developments
  - *A professional management layer for the system as a whole will be at the core of this system.*
  - *MPEG DASH, both for Live and VOD and play out to different devices (laptop, HbbTV, Tablet and Smartphone both for Android and iOS).*
  - *Optimisation of distribution by adding automated management of data flows*
  - *Automation of broadcast processes*
  - *Addition of interactive services*

EBU
OPERATING EUROVISION AND EURORADIO

Cloud infrastructure for broadcasters



Cloud infrastructure for broadcasters

Management layer in private cloud

- *Allow professional users to manipulate setting of the virtual encoders and distribution machines*
- *Management of recourses in private and public cloud*
- *Real-time monitoring of running processes*

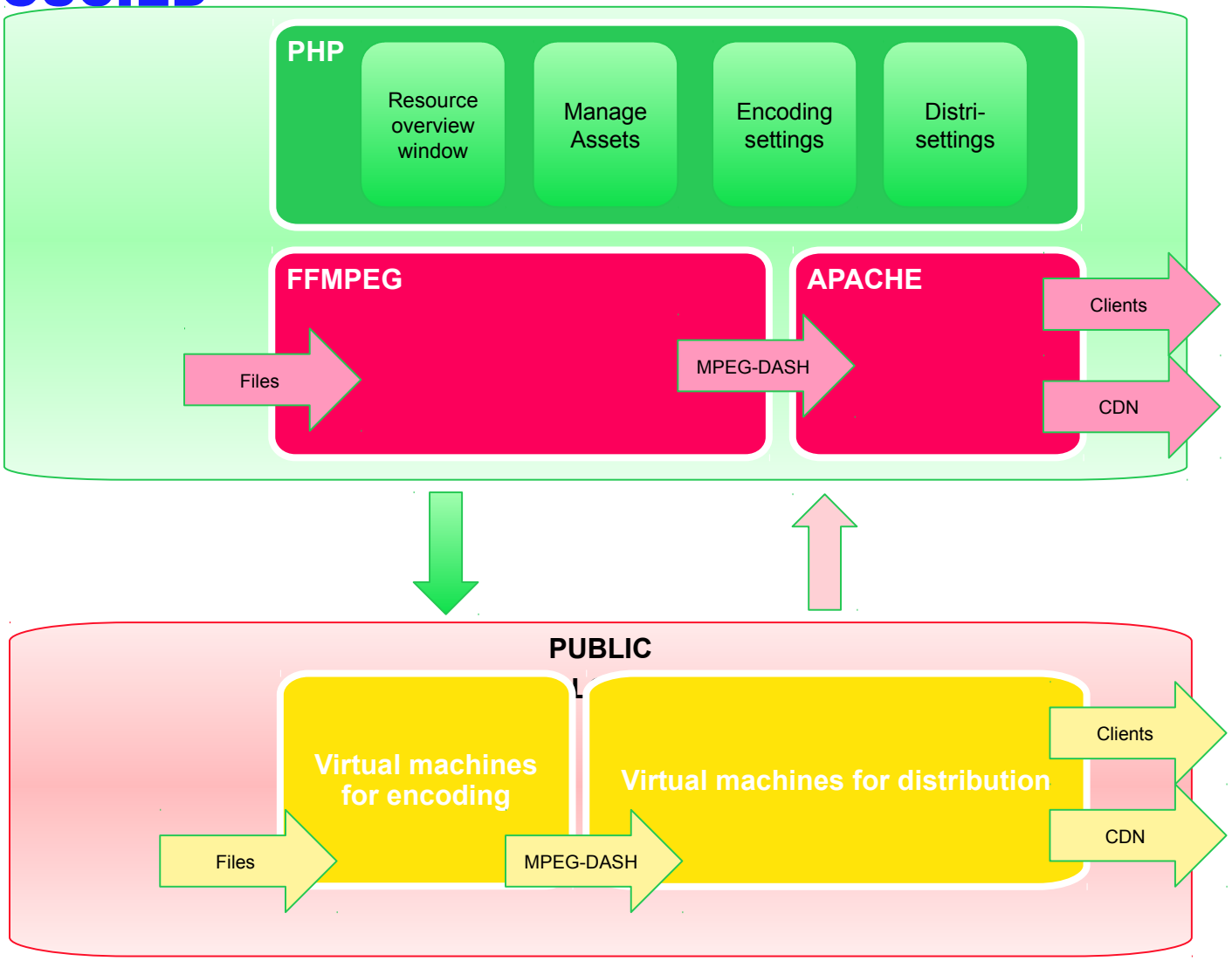
Virtual services in private cloud

- *Use of local machines that run virtual services*
- *Scaling on permanent use in local loop*
  - *Encoding high resolution feeds*

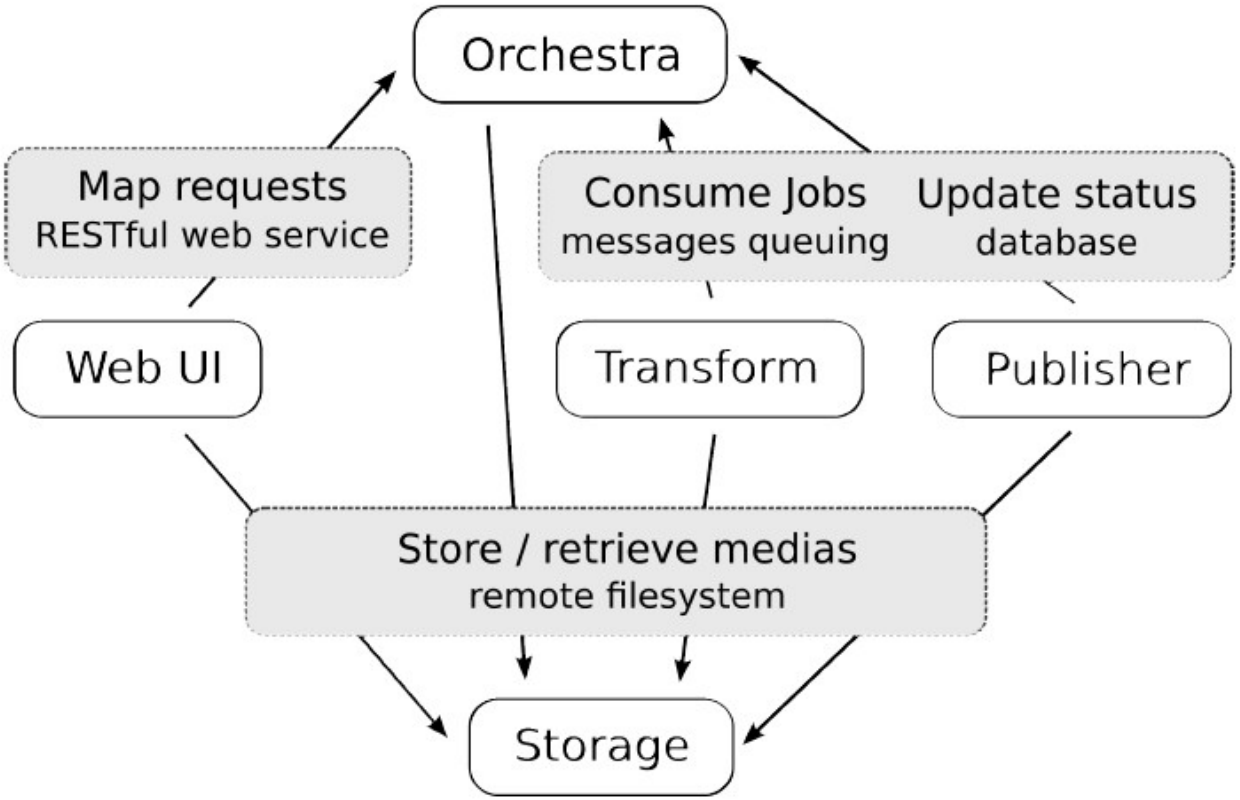
Virtual services in public cloud

- *Fast scaling infrastructure for peak offload*
- *Optimisation of decentralised processes*

OSCIED


PHP
Resource overview window
Manage Assets
Encoding settings
Distri-settings
FFMPEG
APACHE
Files
MPEG-DASH
Clients
CDN
PUBLIC
Virtual machines for encoding
Virtual machines for distribution
Files
MPEG-DASH
Clients
CDN

Current components of oscied



OSCIED basic code

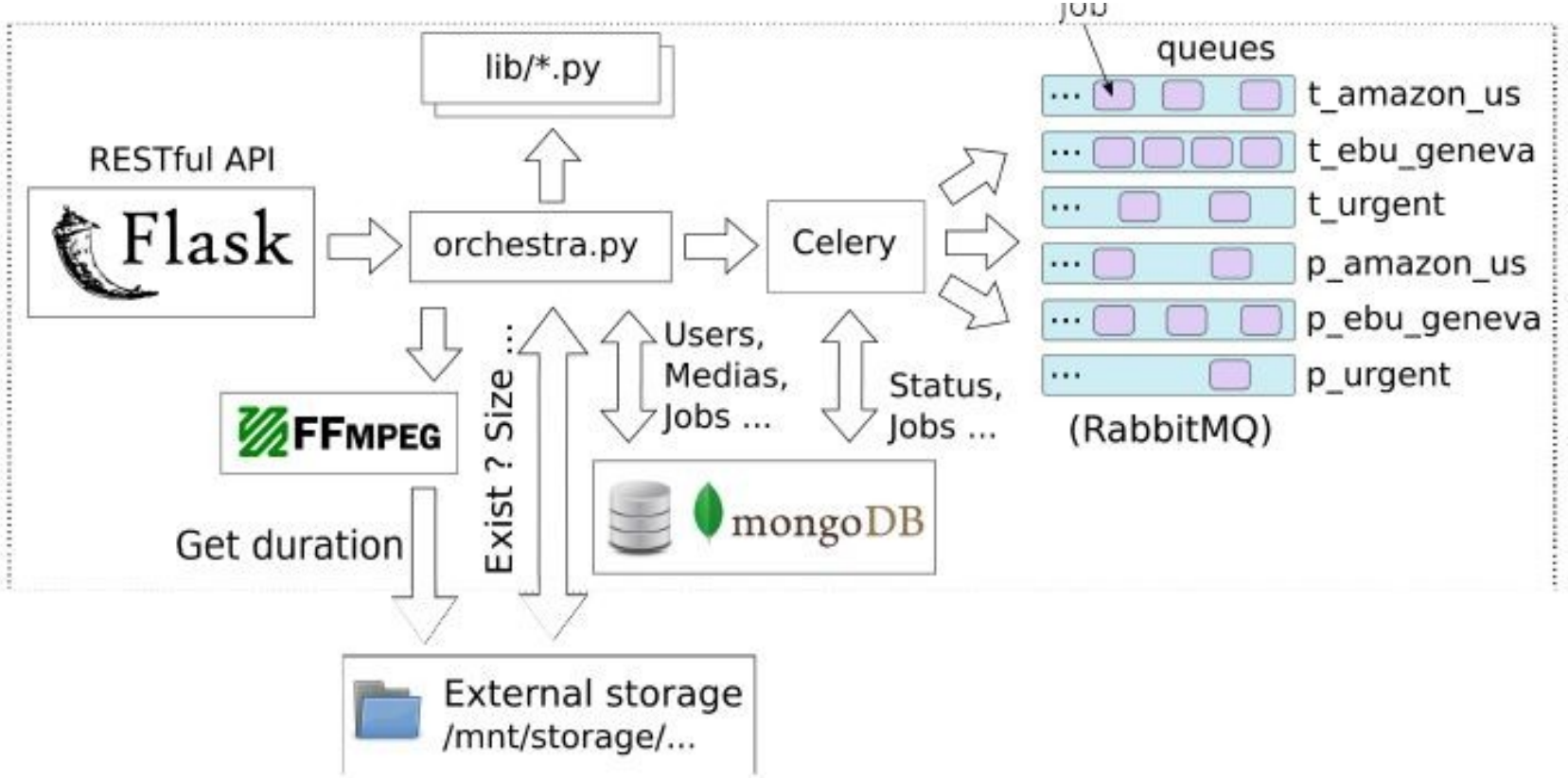
PHP: Management layer in private cloud

- *Monitor window visualising running processes*
- *Settings menu for encoding settings*
- *Setting menu for distribution via private and public cloud Apache virtual machine*
- *> Basic code supports minimal a manual upload of files, a manual input of metadata and proven playout via private cloud an Amazon cloud.*

Cloud infrastructure

- *Virtual machines for encoding (FFMPEG)*
- *Virtual machines for distribution (Apache)*
- *Automatic scaling private to public cloud (JuJu) on basis of business rules*
- *Install script for OSCIED on local hardware*
- *> Basic code supports only Apache virtual machine (with CodeShop streaming module) in private an public cloud. NGIX vm, automated CDN output, streamlink interface with other backends are not available in this version.*

orchestrator



Orchestrator build

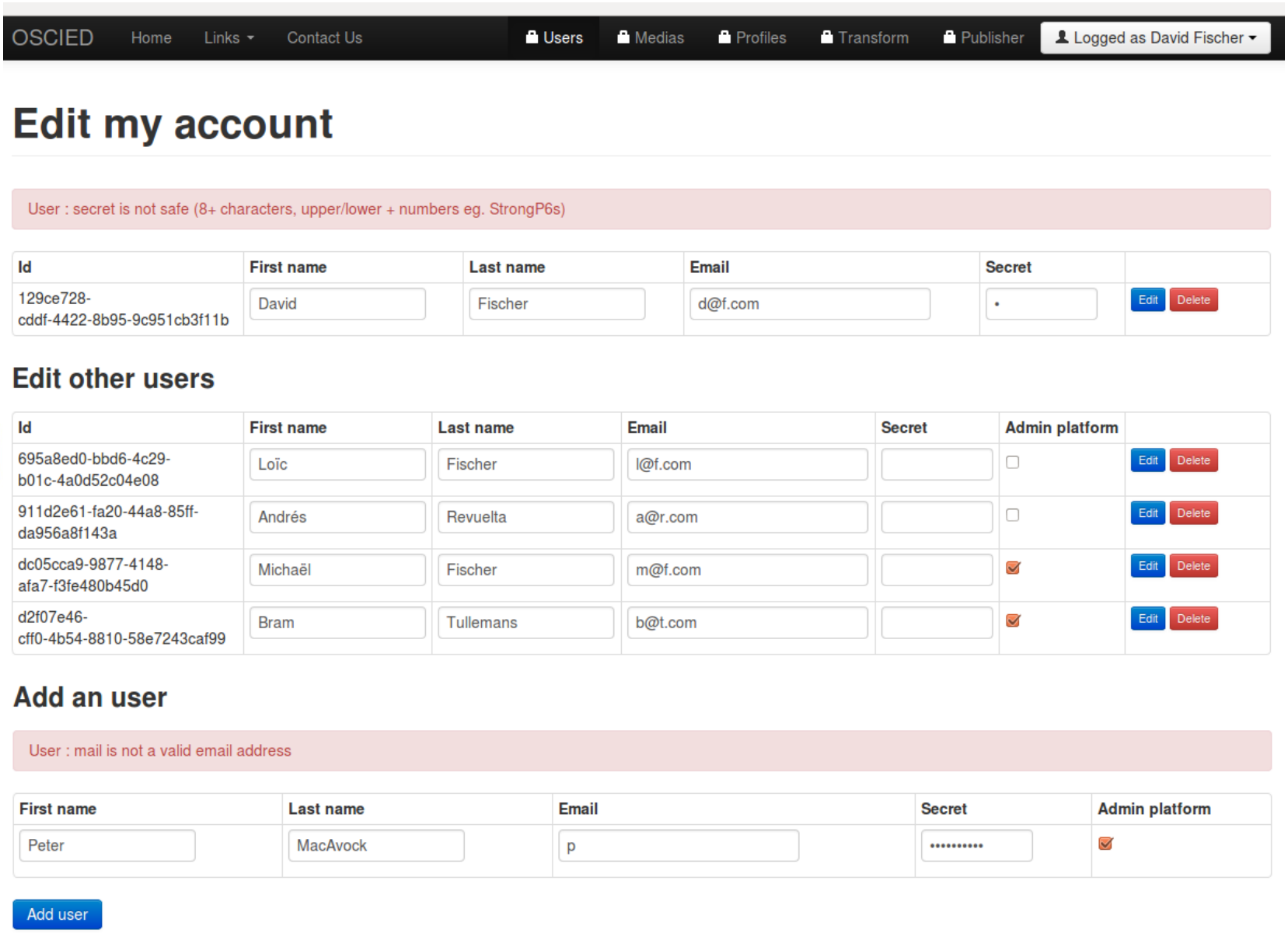
- **OSS Tools**
- • Flask Python Micro Web Framework
- • PyMongo Python module for working with MongoDB
- • MongoDB Scalable, High Performance NoSQL Database from 10gen
- • RabbitMQ AMQP Message Broker from vmware
- • Celery Distributed Task Queue
- • JuJu Cloud Orchestrator from Canonical

Orchestrator manages all actions in OSCIED via:

- • the RESTful API, to expose application's functionalities to user
- • the database, to store application's data (users, profiles, jobs, ...)
- • the message broker, to communicate with workers (transform & publisher)

EBU
OPERATING EUROVISION AND EURORADIO

Account settings



EBU
OPERATING EUROVISION AND EURORADIO

Imported media

OSCIED Home Links Contact Us Users Medias Profiles Transform Publisher Logged as David Fischer

Available medias

Title	Virtual Filename	File size	Duration	Added on	Added by	Status	
Psy - Gangnam Syle 720p	Psy_gangnam_style_720p.mp4	174.8 MB	00:04:12.16	2013-02-02 15:39	David Fischer	PUBLISHED	Delete
Project London MP2	Project_London.mpg	24.4 MB	00:00:01.95	2013-02-02 15:40	David Fischer	READY	Delete
s	s.mp4	0 Bytes		2013-02-02 15:40	David Fischer	DELETED	
Project London - Official Trailer (2009)	Project_London_trailer_2009.mp4	52.3 MB	00:02:44.88	2013-02-02 15:38	David Fischer	PUBLISHED	Delete
Psy - Gangnam Style	Psy_gangnam_style.flv	174.7 MB	00:04:12.16	2013-02-02 15:38	David Fischer	PUBLISHED	Delete
Gaga	gaga.mp2	0 Bytes		2013-02-02 15:46	David Fischer	PENDING	
PSY MP2	PSY.mp2	0 Bytes		2013-02-02 15:47	David Fischer	DELETED	
Project London MP2 Bis	Project_London.mpeg	24.4 MB	00:00:01.95	2013-02-02 15:48	David Fischer	READY	Delete

Add a media

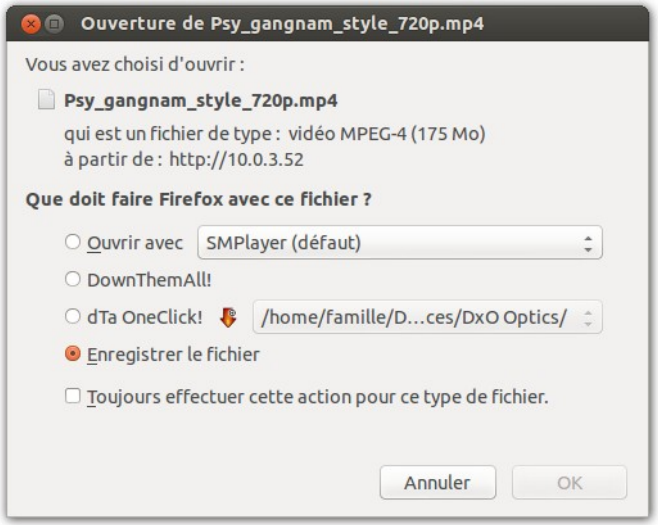
Title

Virtual filename

+ Add Cancel all

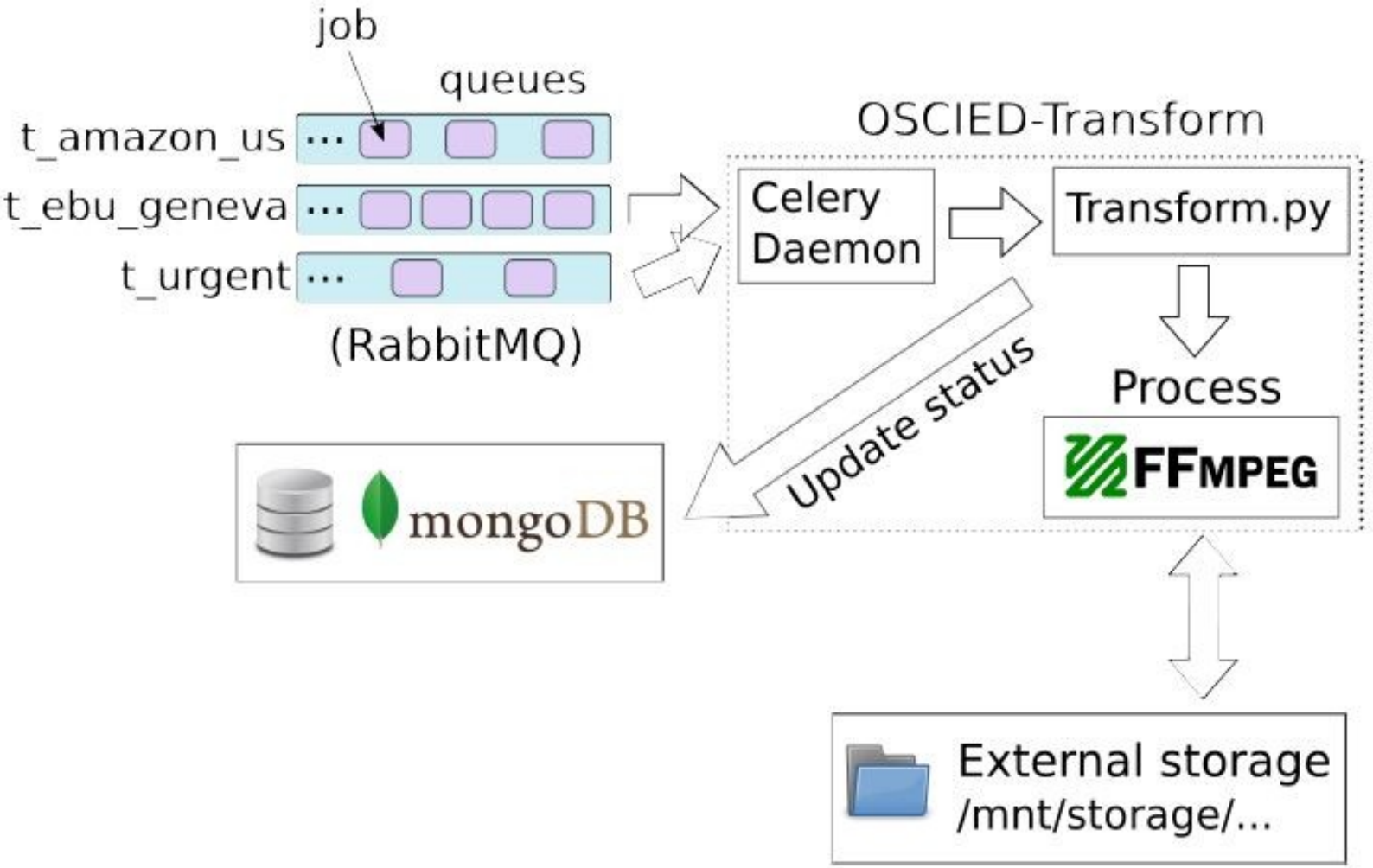
You can drap and drop your files here

Add media



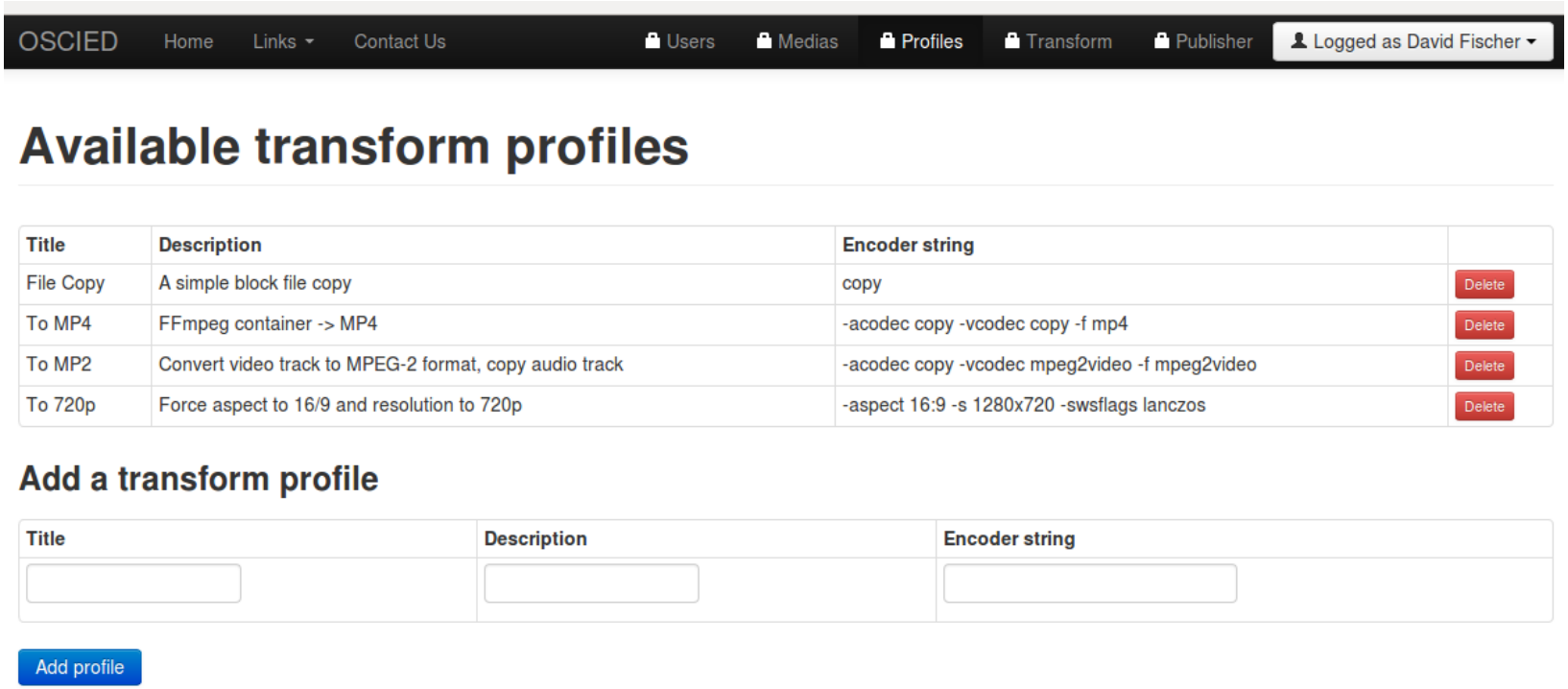
EBU
OPERATING EUROVISION AND EURORADIO

Transform unit



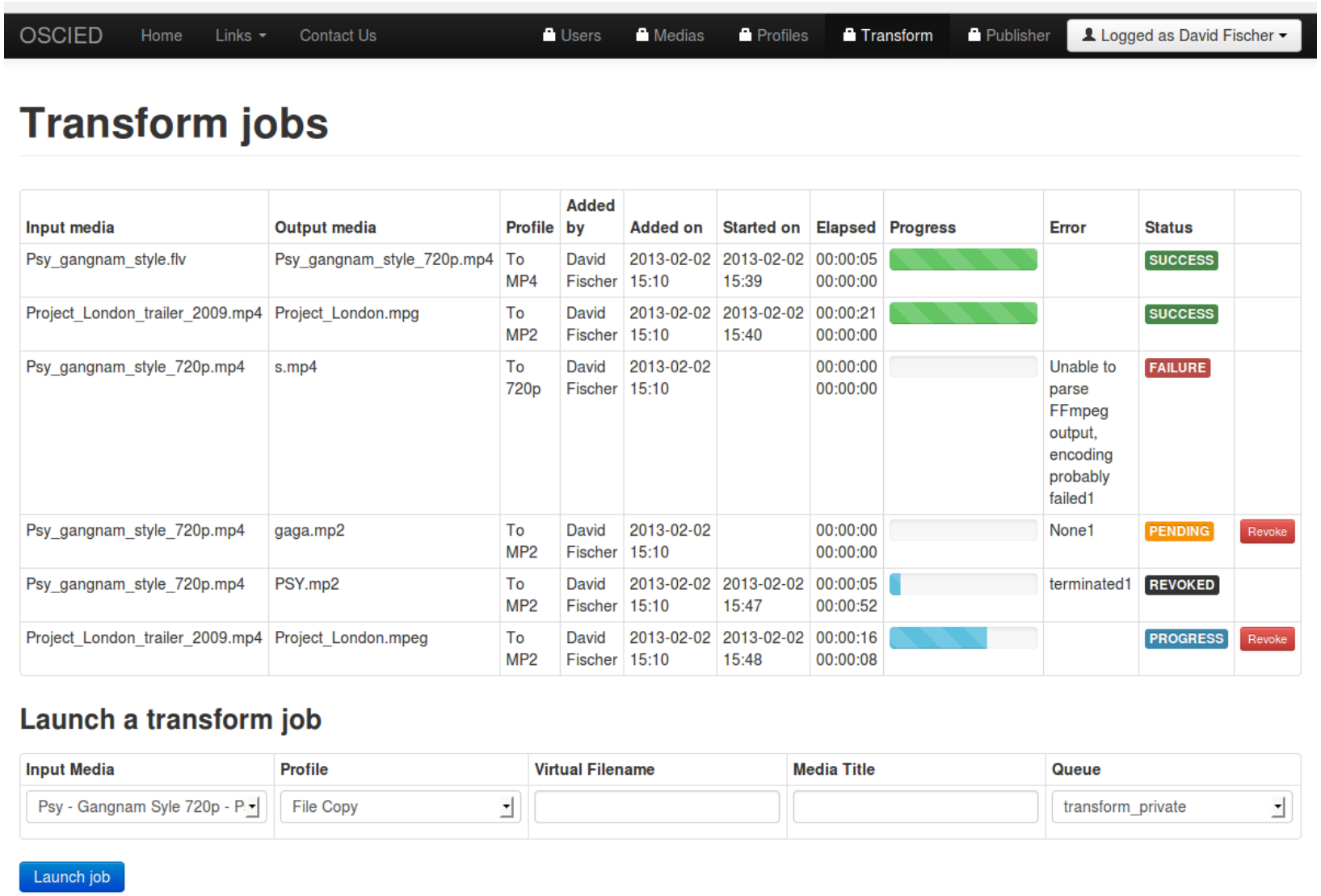
EBU
OPERATING EUROVISION AND EURORADIO

Encoding settings

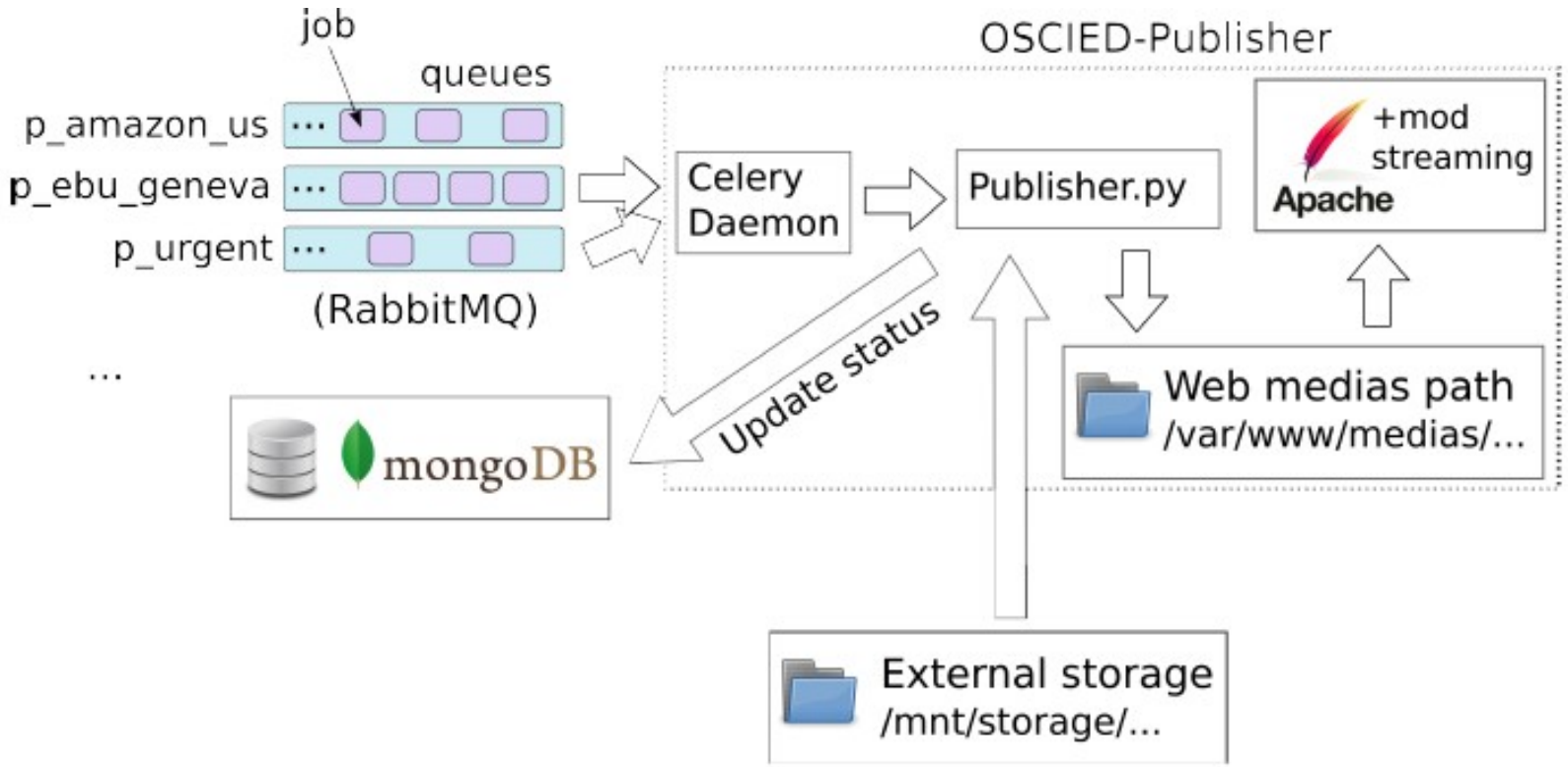


EBU
OPERATING EUROVISION AND EURORADIO

Encoding queue

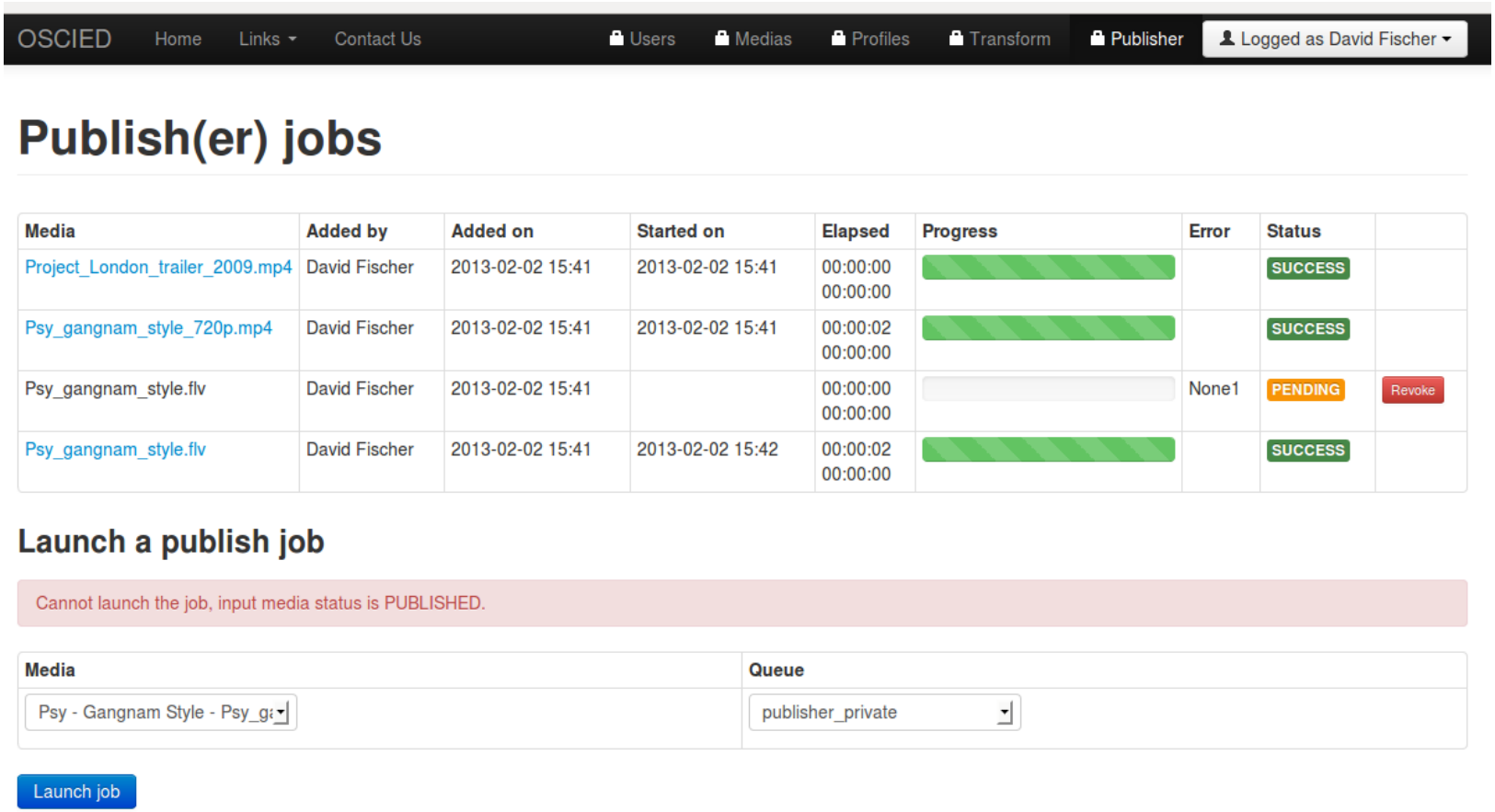


Distribution

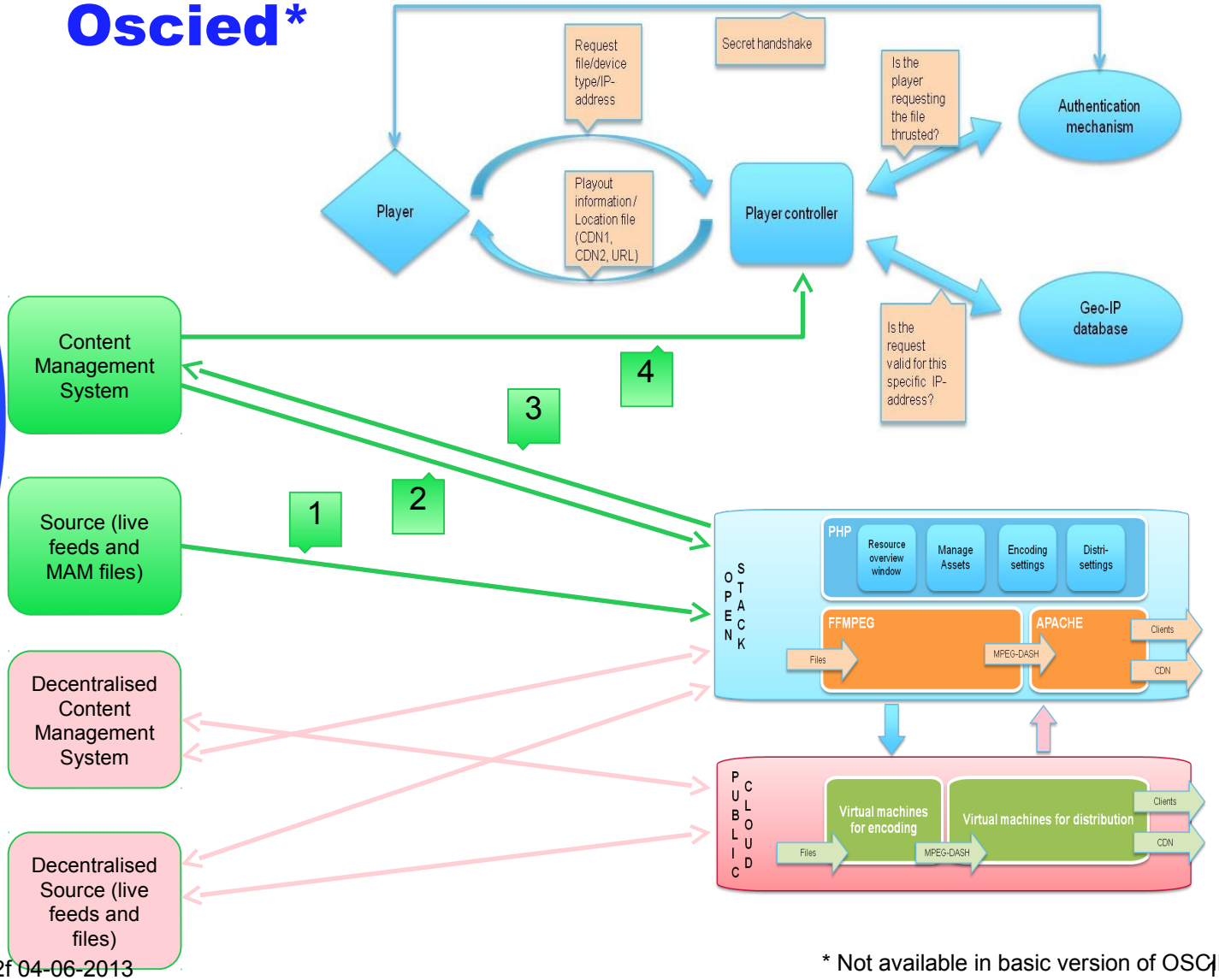


EBU
OPERATING EUROVISION AND EURORADIO

Distribution window

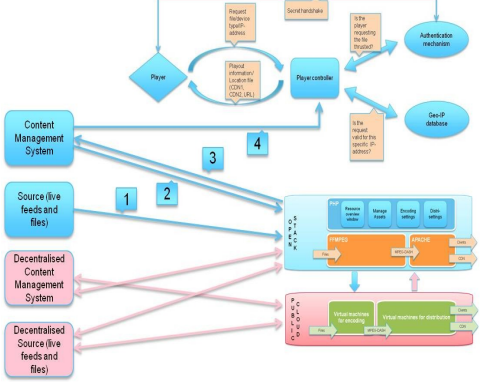


Media Asset management and Oscied*



* Not available in basic version of OSCIED

Media asset management and oscied



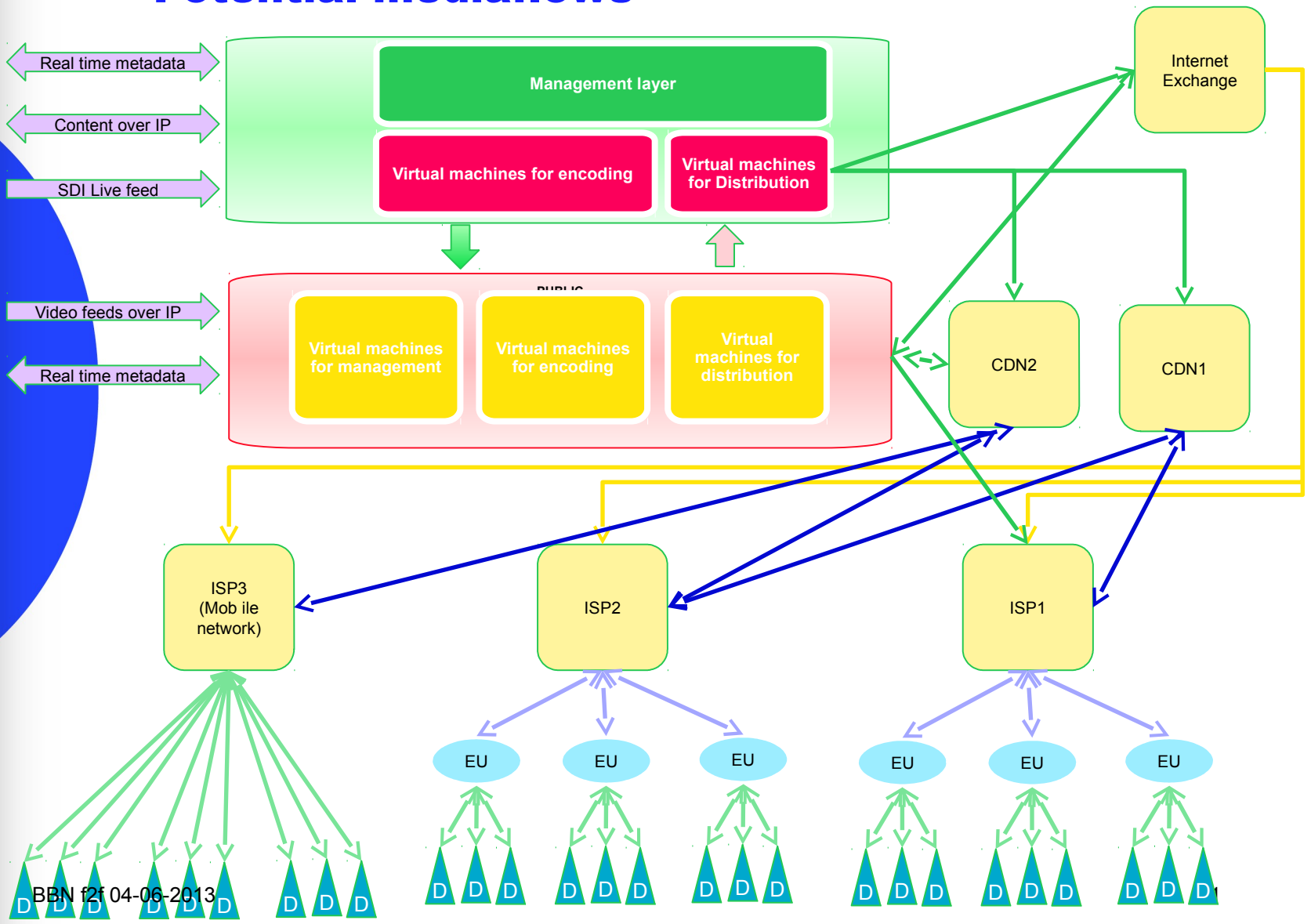
OSCIED needs to be able to interface automatically with MAM backends in order to become useful in a professional environment

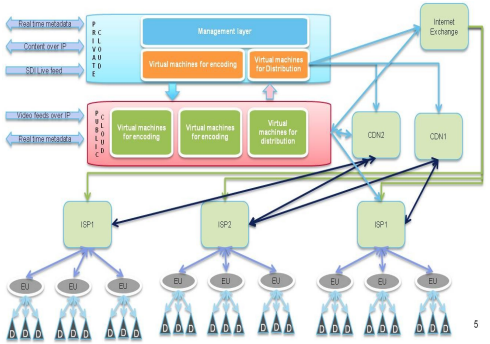
- A source file is uploaded to the private cloud of OSCIED
- The CMS sends a notification XML with details of the file that is uploaded (name, location, requested encoding profile and publication details)
- The management layer of OSCIED will send a notification XML to the CMS when the file is available where in the distribution chain
- The CMS will communicate with the Player Controller the locations of the files

It should be possible to ingest content directly in the public cloud of OSCIED, or from a external location to the private cloud.

EBU
OPERATING EUROVISION AND EURORADIO

Potential mediaflows



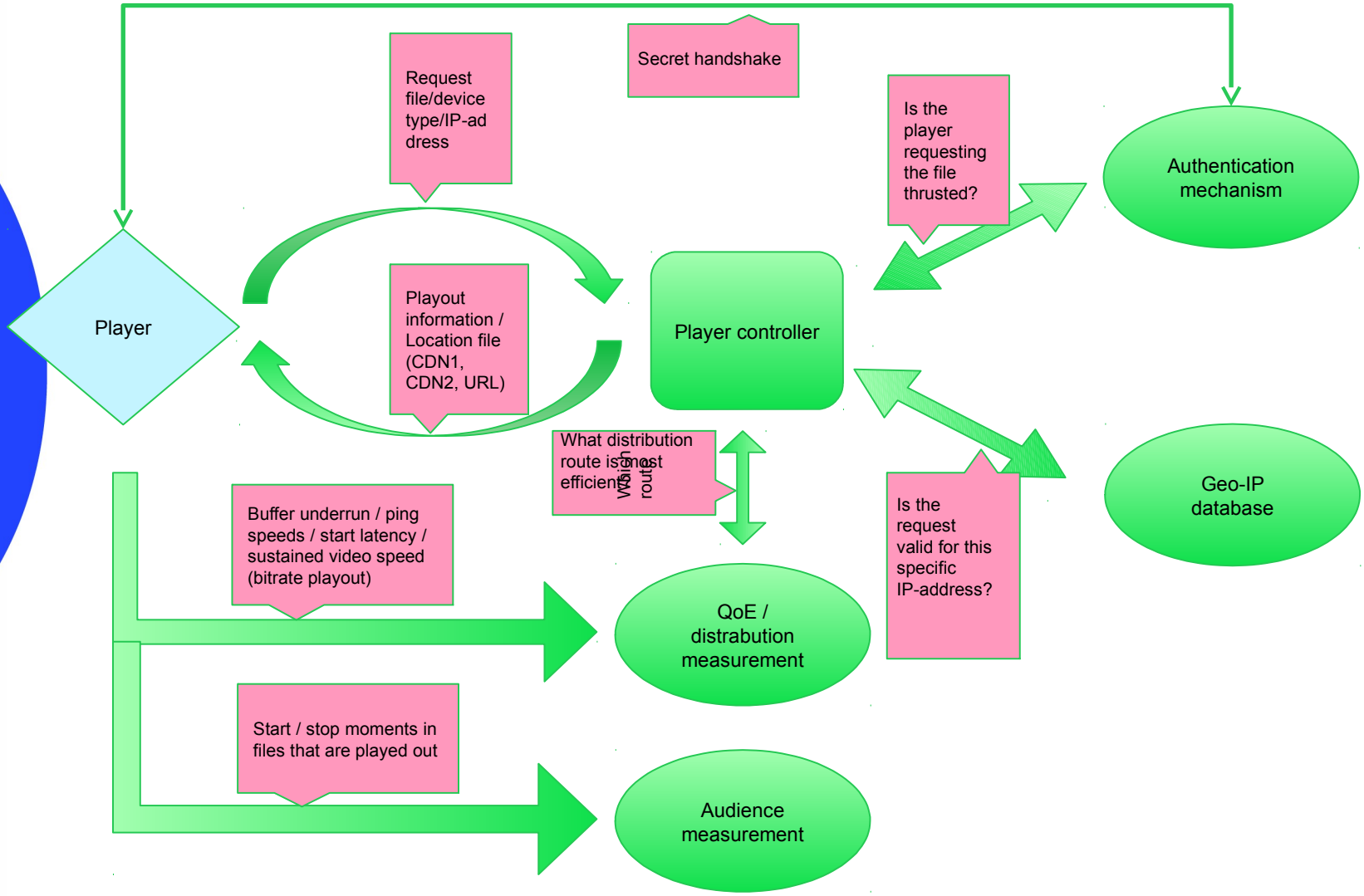


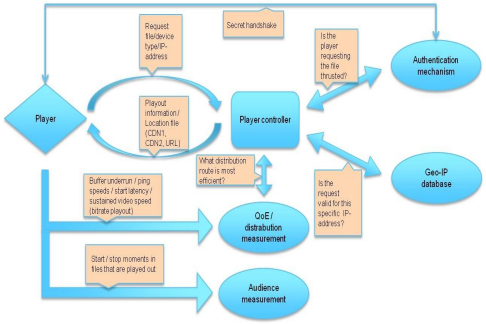
communication overlay

Connect different distribution outlets with automated distribution intelligence to optimise data flows from and to OSCIED

1. Distribution overlay: The management environment decides what the the optimal data route is to distribute content from private or public OSCIED via IXPs, CDNs, ISPs, End User networks (EU) to specific devices (D).
   - *Congestion in network avoided*
   - *Capabitlity of network is taken into account (for example if it can transcode content for playback on specific devices)*
   - *Distribution costs are taken into account (cost per GB or GB/s)*
2. Communication overlay: Dataflows from end user devices (D) are alligned with those of the content provider and processed in the cloud or the network (for example active nodes in the CDN)
   - *Synchronisation of media flows*
   - *Generating interactive interfaces to thin clients / devices with limited playback options*
   - *Upload of source files of media assets that are processed dectralised (remote editing, postproduction, etc.)*

Player backend communication





Future intelligence in oscied

Player backend functionalities and other feedback loops can be part of OSCIED in the future.

- Player sends information to OSCIED that can be used for end to end monitoring of the IP-network. This information can be used to manage traffic flows, for example avoid a busy CDN at a certain moment.
- Player sends information to OSCIED that can be used for audience measurement.
- Player can request a media file for playback from OSCIED. The Player Controller of OCSIED checks:
  - *If the player is authentic*
  - *If the player is used in an accepted location*
  - *What the best available location of the file is*
- Player receives an XML or MPD with information about either:
  - *The location of a still informing that the file cannot be played out in that region*
  - *The location of the media files that can be played out from the easiest to reach cache*