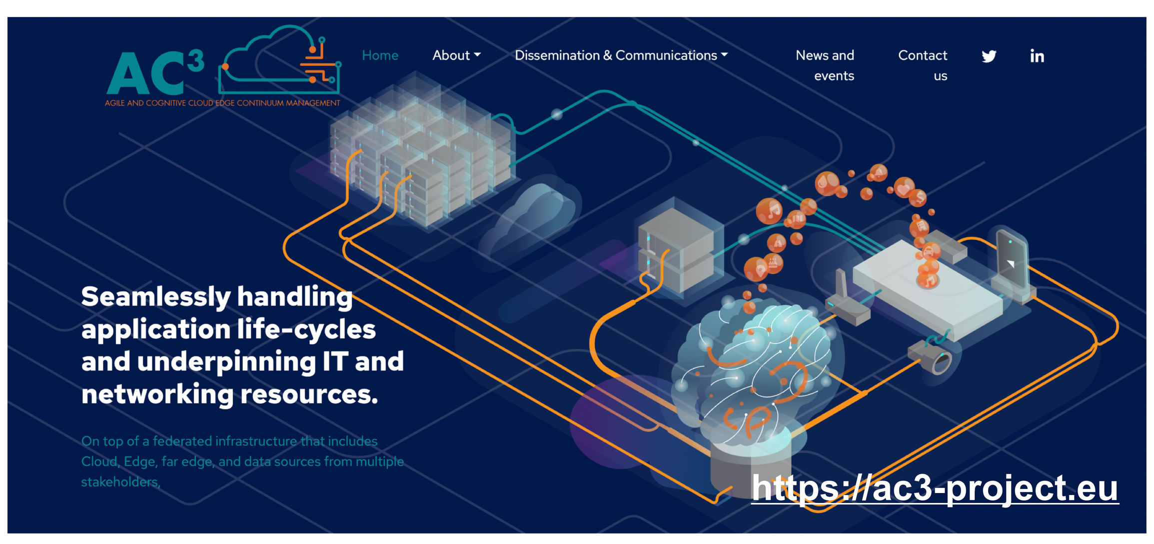
# Distributed processing of Integral Field Spectroscopy data within the European AC³ Project


**Cristina Catalán-Torrecilla[1] (ccatalan@ucm.es), Mario Chamorro-Cazorla[1], Sergio Pascual[1], Armando Gil de Paz[1], África Castillo-Morales[1], Jesús Gallego[1], Christos Verikoukis[2], Adlen Ksentini[3], Ray Carroll[4], Ben Capper[4], Ryan Jenkins[4] , Valeriia Shapoval[4]**

(1) Departamento de Física de la Tierra y Astrofísica, Instituto de Física de Partículas y del Cosmos IPARCOS, Universidad Complutense de Madrid, E-28040 Madrid, Spain
(2) ISI/ATH and University of Patras
(3) Eurecom Institute, Sophia Antipolis, France
(4) Waterford Research Interest Group, Red Hat Research, Office of the CTO (OCTO), Red Hat Ireland, Communications House, Waterford, Ireland

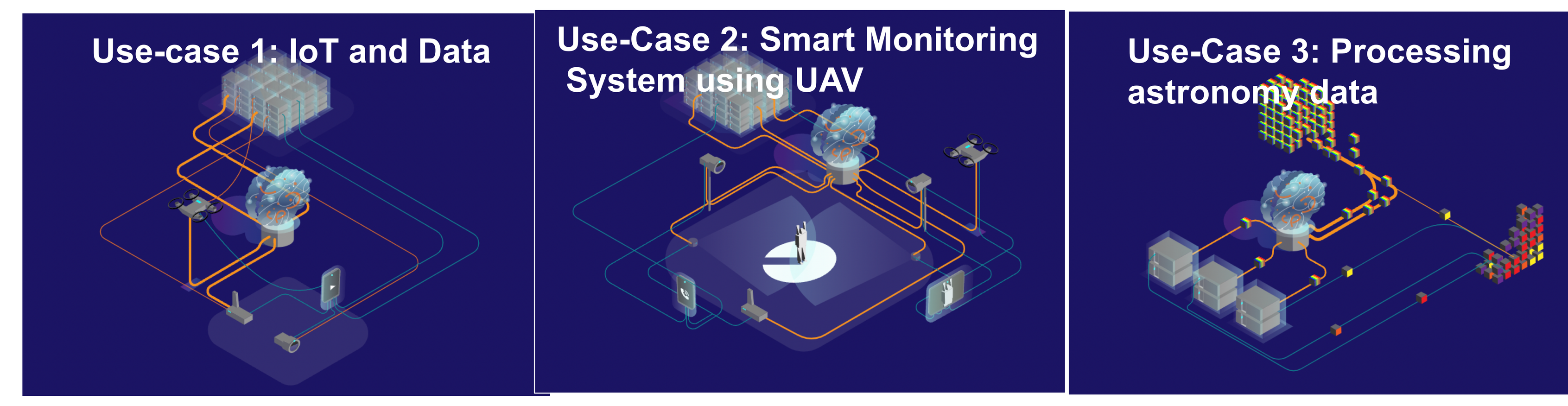

## The AC3 project

The European AC3 project (Agile and Cognitive Cloud-edge Continuum management, https://ac3-project.eu/) aims to develop a novel architecture using a federated Cloud Edge Continuum infrastructure. Central to this effort is the Cloud Edge Continuum Computing Management (CECCM) that guides both the resources and the applications Life Cycle Management. Another important point of AC3 is that it includes the implementation of Artificial Intelligence and Machine Learning technologies to ensure an energy-efficient operation. The project consortium is composed of a total of 14 institutions from 9 European countries.




## Proof of concept: Uses Cases

Within this project, there are a total of three different Use Cases:




UCM lead Use Case 3 whose main goal is working on the distributed processing of Integral Field Spectroscopy (IFS) data. To overcome challenges related to data volume and computational resources, AC3's modular architecture is essential. Use Case 3 focuses on the use of microservice-based astronomy software deployed in containers. Containerization techniques simplify the management of the analysis tools needed for processing the datacubes. At the moment, we are working on testing the scalability of existing astronomical software (like spectral synthesis programs). For that, each astronomical software is being deployed in containers. Later, AC3 will have the capacity to distribute the computing resources based on workload requirements. This process will help to optimize the processing of large volumes of datacubes.

## Use Case 3 proposed solutions

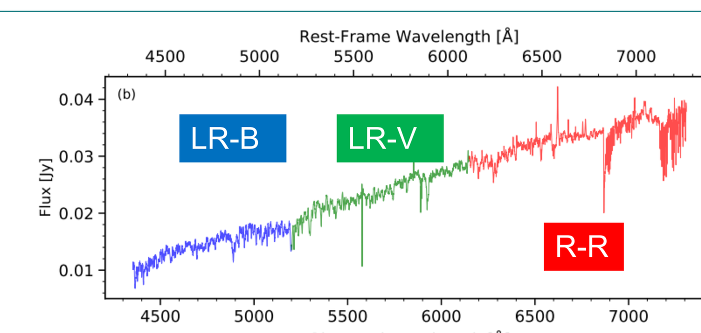
**Use Case 3 will demonstrate the CECCM's capabilities to run astronomical software and process datacubes. This will allow us to integrate scientific applications that will take advantage of hybrid cloud native infrastructures to optimize computing processes.**

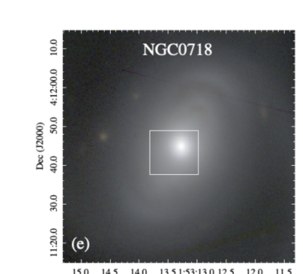
The tools that will be integrated are different spectral synthesis software assets:

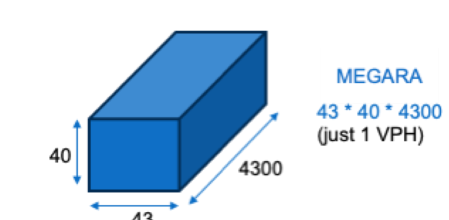
- STECKMAP
- STARLIGHT
- pPXF

**Methodology: Full spectrum fitting**

$$M_\lambda = \left[\sum_{j=1}^{N} x_j b_{j,\lambda} r_\lambda\right] \otimes G(v_*, \sigma_*)$$

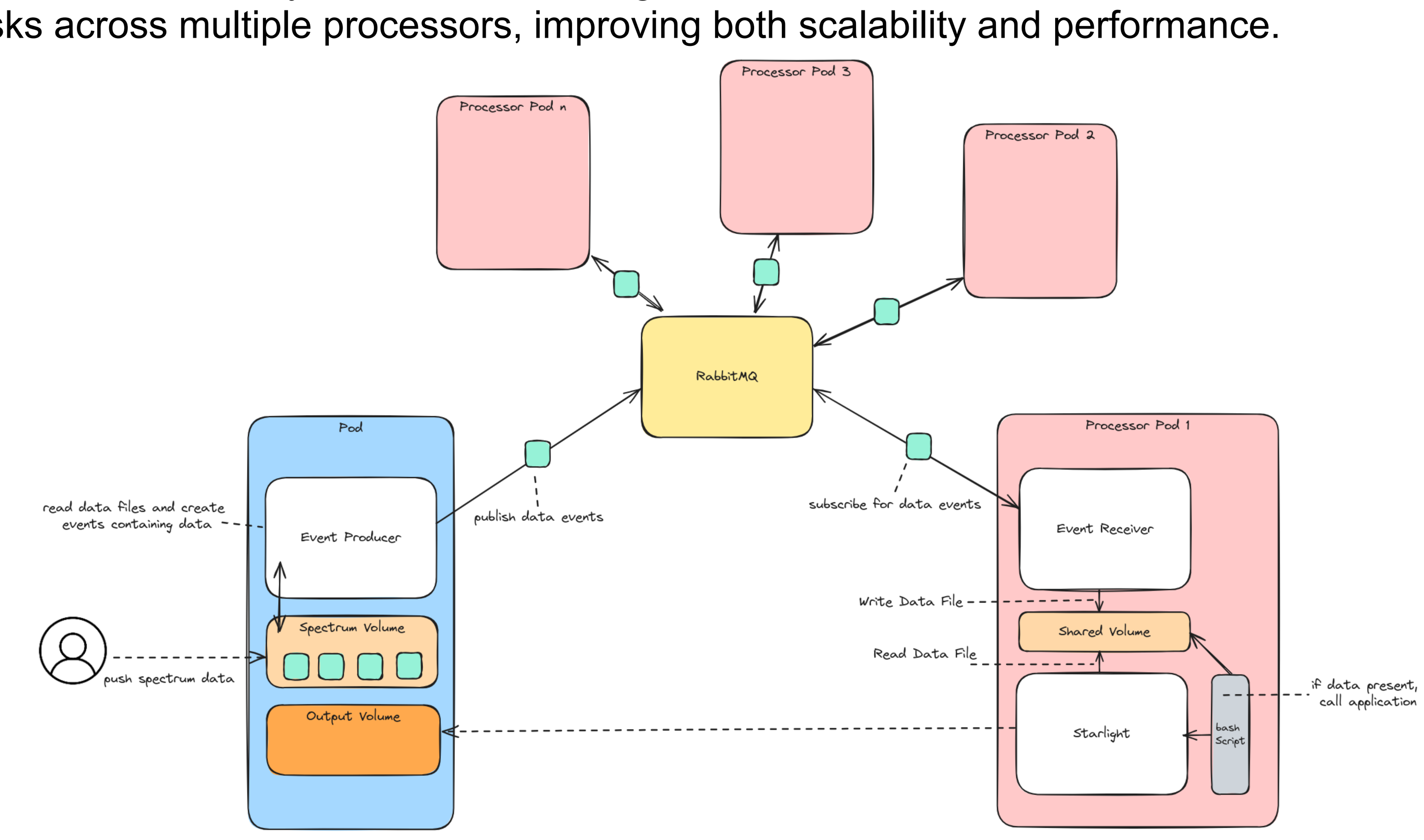



### Containerization (software optimizations & scalability of existing software)

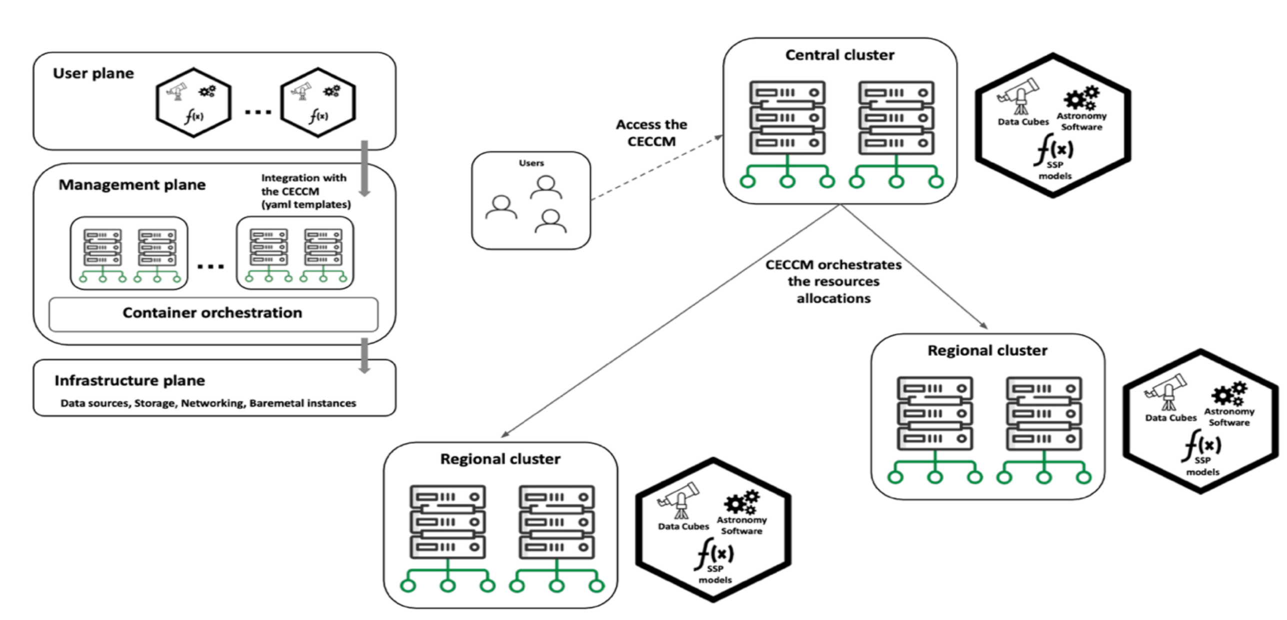
The Use Case 3 implementation will make use of a set of **containerized specific-use analysis tools** to process the data that is distributed by the CECC manager. This architecture ensures efficient data processing by distributing tasks across multiple processors, improving both scalability and performance.




This diagram illustrates a data processing architecture utilising RabbitMQ to enable event-driven communication. This setup helps to parallelise and distribute the processing of large datasets across multiple nodes efficiently.

- Event Producer Pod: This component reads spectrum data files from the mounted Spectral Volume. It creates events (chunks of data) and publishes these events to RabbitMQ.
- RabbitMQ: Acts as the message broker, managing the distribution of data events to various Processor Pods.
- Processor Pods: Each of these pods subscribes to the RabbitMQ queue to receive data events. The Event Receiver container within each pod writes a data file in a format suitable for the Starlight application. The file is saved to a shared volume accessible to both containers within the pod.
- Starlight Application: This application, triggered by a bash script, processes the new data file and saves the results back to a shared volume on the Event Producer Pod.

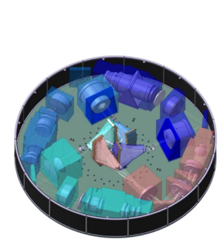
### Arquitecture of UC3 showing the main components & how they interact

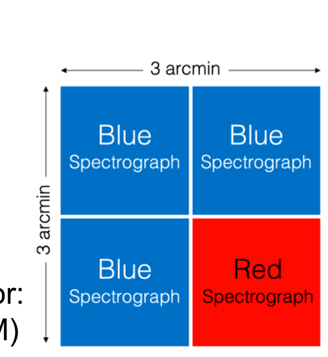



The astronomy software is modeled as a microservice user application based in Linux containers. The users will interact with the CECM to request access to these applications. The CECCM schedules the execution of the applications and the availability of the required resources (datasources, networking, storage, computing). The execution of the astronomy software might be federated into different regions depending on the overall system load and the requirements. Once the software finishes its execution, the result can be stored in the provisioned storage resources. A set of API endpoints will be advertised to interact with all the CECCM services, components and layers, allowing to process the astronomical datacubes.

## Future projects that could benefit from AC3

TARSIS (to be installed in 2028)




Principal Investigator: Dr. Gil de Paz (UCM)

Future prospects with edge computing & TARSIS: ***data filtering and compression*** & ***perform collaborative research***

## Summary

Use Case 3 will demonstrate the CECCM's capabilities to run astronomical software and process datacubes. This will allow us to integrate scientific applications that will take advantage of hybrid cloud native infrastructures to optimize computing processes.

Additonal info:

- https://github.com/ccatalan/astro-containers/
- https://github.com/MarioChC/astro-containers
- https://github.com/raycarroll/ucm/tree/main?tab=readme-ov-file


*"This project has received funding from the European Union's Horizon Europe Research and Innovation programme under grant agreement No 101093129"*