

**PLIEGO DE PRESCRIPCIONES TÉCNICAS CON RECTIFICACIONES  
INCORPORADAS  
RESOLUCIÓN DEL RECTOR DE 23 DE MAYO DE 2023**

**PLIEGO DE PRESCRIPCIONES TÈCNICAS**

**CLOUDLESS:**

**Plataforma de computación de información en el borde**

CONTRATACIÓN DE SERVICIOS PREVISTA EN EL REAL DECRETO 959/2022, DE 15 DE NOVIEMBRE, POR EL QUE SE REGULA LA CONCESIÓN DIRECTA DE UNA SUBVENCIÓN A CENTROS Y FUNDACIONES PÚBLICOS DE INVESTIGACIÓN Y UNIVERSIDADES PÚBLICAS ESPAÑOLAS PARA LA REALIZACIÓN DE PROYECTOS INNOVADORES EN EL ÁREA DE COMPUTACIÓN EN LA NUBE DEL PROGRAMA UNICO I+D CLOUD, EN EL MARCO DEL PLAN DE RECUPERACIÓN, TRANSFORMACIÓN Y RESILIENCIA -FINANCIADO POR LA UNIÓN EUROPEA- NEXT GENERATION EU

## 1. Objeto

El objeto del presente pliego es establecer las especificaciones técnicas y condiciones que han de regir la contratación del **Proyecto CLOUDLESS** de la **Universitat Rovira i Virgili**, que garantice de manera continua a lo largo de la duración del contrato, el desarrollo de software y conformidad de los hitos y condiciones definidas en el presente pliego.

## 2. Condiciones generales

Con carácter general, el adjudicatario deberá tener en consideración los siguientes aspectos:

- El adjudicatario deberá organizar y dirigir al equipo de trabajo que designe para la realización de las tareas objeto de este contrato, no dependiendo este equipo, en ningún caso, de ninguna unidad administrativa de la Universitat Rovira i Virgili.
- Al equipo de trabajo de la empresa adjudicataria de la contratación no es aplicable el régimen jurídico de los empleados de la Universitat Rovira i Virgili, en especial los aspectos relativos al cumplimiento y control de horario, vacaciones, permisos y licencias y otros de análoga naturaleza.
- El equipo de trabajo que desarrolle las tareas objeto de este contrato no formará parte de los organigramas y directorios de la Universitat Rovira i Virgili, ni dispondrá de claves de acceso a recursos propios de los empleados (salvo en los casos que sea imprescindible para la realización del servicio, como acceso a documentación o plataformas para el cumplimiento del contrato).

## 3. Descripción del servicio

Los requisitos exigidos en este pliego deberán ser considerados mínimos y, por tanto, de indispensable cumplimiento por parte de los licitadores. A los efectos de aceptación de las ofertas, se efectuará una comprobación exhaustiva del cumplimiento de todos y cada uno de ellos. Los licitadores, en su oferta, deberán concretar sus propuestas para cada apartado y superar los requisitos mínimos (si lo estiman oportuno) los cuales serán valorados de acuerdo con los criterios definidos en el Pliego de Cláusulas Administrativas Particulares (PCAP).

El responsable del Contrato (RC) es la figura de máxima responsabilidad por parte de la Universitat Rovira i Virgili en el control y vigilancia del cumplimiento de los compromisos en el presente contrato, siendo imprescindible su conformidad para cualquier modificación a los términos expresados en el contrato. Tiene potestad para proponer la aplicación de penalizaciones previstas en el PCAP. El RC es el interlocutor con el adjudicatario. De igual manera, el adjudicatario pondrá a disposición del contrato un único interlocutor responsable de la prestación del servicio objeto del contrato.

### 3.1. Fases del contrato

Se establecen tres fases en este contrato: inicio, ejecución y finalización.

#### 3.1.1. Fase de inicio

La fase de inicio del contrato tiene por objetivo que el adjudicatario se familiarice con el entorno de trabajo y adquiera el conocimiento necesario para prestar dicho servicio.

En ella, el adjudicatario deberá realizar las tareas que permitan el inicio de los desarrollos, incluyendo:

- Contratación y adscripción definitiva de personal al proyecto.
- Implantación de medios tecnológicos adscritos al proyecto
- Creación de canales de comunicación y difusión del proyecto
- Tareas de formación del personal
- Entrega del Plan de ejecución del proyecto al RC.

Todo el personal que se adjudique al proyecto deberá conocer la documentación asociada a la fase de Inicio.

En esta fase, el adjudicatario podrá iniciar la ejecución de las tareas propias del servicio, pero no tendrá penalizaciones asociadas a la falta de calidad.

El plazo de esta fase es de un mes natural a partir de la fecha de inicio del contrato.

Una vez finalizada la fase de inicio, el RC emitirá un certificado de cumplimiento, dando lugar al derecho de cobro del primer pago, definido su importe en el Pliego Administrativo.

### 3.1.2. Fase de ejecución

Esta fase constituye la mayor parte del proyecto. Durante ella, el adjudicatario realizará la ejecución propia del servicio de acuerdo con las condiciones mínimas establecidas en el presente pliego y las aportadas en su oferta, junto con las propuestas adicionales y transformaciones que se consideren necesarias para una gestión eficiente.

La fase de ejecución comprende desde el fin de la fase de inicio hasta el fin del contrato o el fin de la prórroga, de producirse.

Con periodicidad mensual, en la reunión de seguimiento se revisarán los informes de avance de la ejecución, que servirán al adjudicatario para ajustar o corregir las posibles desviaciones de la planificación.

Con periodicidad trimestral, si el cumplimiento es el adecuado, el RC emitirá una certificación que generará el derecho de cobro de las tareas realizadas en dicho periodo, pudiendo retenerse la certificación hasta el cumplimiento efectivo de la planificación por parte del adjudicatario.

### 3.1.3. Fase de finalización.

A la finalización del contrato, el adjudicatario deberá hacer entrega por escrito en formato digital, de los siguientes contenidos:

- Código fuente desarrollado en el proyecto. En las condiciones y formato definidas por el RC.
- APIs y Documentación asociada al desarrollo del proyecto.
- Manuales de usuario individuales para cada perfil del proyecto.
- Ficha técnica, detallando la tecnología empleada
- Manuales de instalación y mantenimiento de la solución desarrollada

La calidad de dicha documentación debe ser validada por el RC, estando obligada la empresa a su corrección o ampliación en los términos que éste determine.

Además, el adjudicatario se compromete a colaborar activamente con la Universitat Rovira i Virgili y, en su caso, con la empresa, departamento o entidad responsable de la gestión del resultado del proyecto, para facilitar la transferencia de conocimiento.

A criterio del RC, si éste considerara que la documentación escrita es insuficiente, el adjudicatario estará obligado a transferir el conocimiento a través de sesiones de formación tanto a técnicos de la Universitat Rovira i Virgili como a la empresa, departamento o entidad responsable de la gestión del resultado del proyecto (si es el caso).

El adjudicatario será completamente responsable de la prestación normal del servicio, con la capacidad y agilidad requeridas, y de las tareas derivadas del paso de conocimiento, hasta que finalice el contrato.

La fase de finalización del servicio convive con el final de la fase de ejecución y su duración máxima será de 15 días naturales.

### 3.2. Descripción de los servicios a prestar

El servicio objeto de contrato abarcará:

Para la prestación de los servicios anteriores la Universitat Rovira i Virgili precisa un servicio que contemple los siguientes elementos:

- Servicio de coordinación del proyecto de desarrollo. Supervisando y definiendo las tareas concretas del personal adscrito al proyecto, para garantizar el cumplimiento de los requisitos en las condiciones de calidad y funcionalidad definidos por la Universitat Rovira i Virgili.
- Garantizar la calidad del código fuente generado
- Garantizar la calidad de la documentación generada
- Facilitar el control y mantenimiento de los errores de desarrollo
- Ofrecer soluciones óptimas para resolver las complejidades de desarrollo
- Establecer mecanismos para facilitar el trabajo en equipo con el personal técnico y científico de la Universitat Rovira i Virgili.
- Adscribir al proyecto personal técnico especializado en el área de desarrollo de software procurando y favoreciendo la máxima estabilidad posible del personal, ofreciendo soluciones para cubrir sus posibles carencias de conocimiento en áreas específicas.

### 3.3. Entorno tecnológico

Definir los lenguajes y ámbito tecnológico y de conocimiento del proyecto. Se puede dividir como:

#### **PLATAFORMAS**

El contratista pondrá a disposición las plataformas necesarias para el desarrollo, pruebas e integración de la solución desarrollada, así como los manuales de instalación y mantenimiento de dicha plataforma, en la duración del contrato.

La empresa deberá trabajar fundamentalmente en AWS Cloud, utilizando los siguientes servicios AWS S3, AWS Lambda, AWS Batch, AWS ASF, AWS EKS, AWS Spot Instances y AWS EC2 VMs. También es conveniente disponer de un cluster de computación donde se puedan desplegar servicios de almacenamiento (Minio) y computación (Kubernetes, Lithops, OpenFaas, Fn, Ray, Dask).

Todos los desarrollos se realizarán en repositorios github privados y se requieren baterías de pruebas automatizadas.

#### **TECNOLOGÍAS**

Los lenguajes de desarrollo son Python y Javascript. Se utilizarán las plataformas Lithops, NextFlow y DATOMA.cloud. DATOMA es una infraestructura Cloud desarrollado por CloudLab URV en AWS Cloud.

Los entornos tecnológicos relacionados implican uso y conocimiento de soluciones Cloud en AWS Cloud como AWS S3, AWS Lambda, AWS Batch, AWS ASF, AWS EKS, AWS Spot Instances y AWS EC2 VMs. También de servicios de almacenamiento de objetos (Minio) y de computación (K8s Kubernetes, Lithops, OpenFaas, Fn, Ray, Dask). El lote 2 además debe tener conocimientos y disponibilidad de uso de TEEs (Trusted Execution Environments) como Intel SGX.

No se empleará software con licencia de uso, APIs externas, o código fuente, scripts, CSS o cualquier recurso externo para la resolución de tareas de desarrollo del proyecto, sin la validación expresa del RC.

**(ANULADO)** ~~Todo el software desarrollado será propiedad de CloudLab URV y en ningún caso se establecerán licencias propietarias o de uso para la explotación de resultados.~~

### 3.4. Ubicación del servicio

El adjudicatario podrá prestar el servicio desde cualquier país de la CEE, si bien se valorará la presencia en España.

El personal adscrito al proyecto que realice tareas en remoto o teletrabajo, será especialmente supervisado por el RC, tanto a través del cumplimiento de tareas de corta periodicidad, como mediante mecanismos de supervisión del equipo de trabajo y conectividad online a través de sistemas de videoconferencia.

Todas las reuniones se realizarán en Tarragona (Campus Sescelades) cada tres meses. No es necesario provisionar presupuesto de viajes fuera de Tarragona.

La ejecución remota del servicio, debe realizarse en unas instalaciones con garantías de seguridad y control de acceso incluyendo la capacidad de establecimiento de una conexión segura entre los equipos informáticos del adjudicatario y la red corporativa de la Universitat Rovira i Virgili a través de Internet, previamente aprobadas por el RC.

### 3.5. Metodología y herramienta de gestión de tareas

Este punto solo se refiere a los lotes 1,2 y 3.

Durante la prestación del servicio, la empresa implantará, gestionará y documentará las actividades relacionadas con la gestión de sistemas, mediante un sistema de Helpdesk que permita automatizar, centralizar y mejorar la gestión de servicios TIC según metodología ITIL. El RC dispondrá de acceso a dicha herramienta en todo momento, debiendo entregarse la máquina virtual como resultado del fin del contrato.

En cualquier caso, el adjudicatario deberá adaptarse al cambio tecnológico que pueda sufrir dicha herramienta de gestión de tareas, así como otras herramientas para el tratamiento de incidencias y soporte de usuarios.

Durante la prestación del servicio, la empresa implantará, gestionará y documentará las actividades relacionadas con la gestión de incidencias de desarrollo, mediante una plataforma que permita la gestión de dicha información con objeto de que el personal técnico y de análisis de la aplicación,

conozcan el estado de las incidencias detectadas, así como de las propuestas de mejora, modificación o ampliación de especificación consideradas necesarias.

Se realizarán reuniones de seguimiento a petición de RC donde se presentarán los informes de ejecución del servicio y se valorarán acciones correctivas en caso de ser necesarias.

Se propone la aplicación de metodologías de calidad, como medida de control de garantía y reducción de riesgos. Para ello se aplicará para los aspectos del proyecto relacionados con la gestión de sistemas, la metodología ITIL, mientras que para los aspectos del proyecto relacionados con la gestión del desarrollo se aplicará la metodología SCRUM.

### 3.6. Horario de prestación del servicio

El personal deberá tener disponibilidad en la misma franja horaria en que se encuentra disponible el personal de la Universitat Rovira i Virgili, con objeto de facilitar y optimizar la comunicación. El horario de prestación del servicio será:

- Disponibilidad de comunicación remota, de lunes a jueves, de 9:00 a 17:00.
- Disponibilidad presencial para reuniones, de lunes a jueves, de 9:00 a 14:00 en horario de invierno, y de lunes a jueves de 9:00 a 14:00 en horario de verano.

El horario de verano cubre los meses de junio a septiembre y el horario de invierno cubre el resto del año.

El servicio se prestará siguiendo el calendario laboral oficial de la Provincia de Tarragona.

Si hay alguna actuación que puede afectar a la disponibilidad y continuidad del negocio, se realizará fuera del horario laboral a fin de no entorpecer el desempeño de las labores propias de los usuarios.

El RC podrá modificar el horario de prestación por necesidades del servicio no superando en ningún caso el cómputo total de horas contratado.

En las reuniones de seguimiento, el licitador deberá presentar un informe que describa la cobertura del servicio realizada, para lo cual deberá estar habilitado un mecanismo de control de presencia de sus técnicos.

### 3.7. Recursos materiales para la prestación del servicio

La empresa adjudicataria deberá contar con los siguientes medios materiales y herramientas para la prestación del servicio:

- Servidores físicos y virtuales y software necesario para el control, desarrollo, integración y presentación de los resultados del proyecto.
- Herramienta de control remoto con las siguientes características: seguridad en el establecimiento de las comunicaciones, trazabilidad sobre el control de los equipos, uso de cuentas personalizadas para los operadores y confirmación de toma de control del equipo por parte del usuario.
- Ordenador personal con conexión a internet, preferiblemente portátil, y software asociado para cada personal adscrito. Será imprescindible que dichos equipos cuenten con todas las medidas de seguridad para evitar la propagación de malware (antivirus y actualización de parches de seguridad) y con el software de control remoto anteriormente citado, que cuente con funcionalidades adicionales, como la posibilidad de coger control

remoto de equipos ubicados en redes externas o con cortafuegos, chat o videoconferencia, intercambio de documentos, etc.

- Herramientas software para la comunicación con el RC.
- Mecanismo para el control horario de sus trabajadores.

En general, el licitador podrá proponer herramientas adicionales que considere adecuadas para una mejor prestación del servicio, corriendo con los costes de alojamiento y licenciamiento del software propuesto.

### 3.8. Equipo de trabajo

Se debe adscribir como mínimo un Coordinador/Jefe de Proyecto durante todo el periodo y un número de programadores suficiente como para cumplir los hitos del proyecto definidos, atendiendo al número de jornadas ofertadas.

El equipo de trabajo dispondrá de una especificación de requisitos acorde a las horas contratadas, como marco de definición de las tareas, gestionando a través del sistema de gestión de incidencias y solicitudes la organización y centralización de las relaciones de los usuarios de la Universitat Rovira i Virgili con el adjudicatario en lo relativo a la prestación del servicio objeto del contrato, la comunicación de incidencias o solicitudes, la priorización de tareas, etc. El RC supervisará los trabajos realizados a través de las reuniones tanto periódicas como a petición del RC.

Para desarrollar la prestación del servicio objeto de este contrato, el adjudicatario pondrá a disposición de la Universitat Rovira i Virgili un equipo de trabajo que atenderá las tareas, incidencias y solicitudes y planificará y dirigirá las actuaciones realizadas por el personal adscrito al contrato en el horario definido en el pliego de condiciones.

Por motivos de enfermedad, vacaciones o situaciones que afecten a la capacidad de desarrollo de los miembros del equipo de trabajo, el adjudicatario deberá proceder, sin repercusión económica adicional, a la sustitución temporal por técnicos con la misma cualificación técnica, de forma que no se interrumpa la prestación del servicio contratado.

Para evitar posibles inconvenientes de adaptación al entorno de trabajo motivado por sustituciones de personal, la empresa adjudicataria contemplará periodos de solapamiento de técnicos durante el tiempo necesario sin coste adicional para la Universitat Rovira i Virgili, a efectos de la adecuada formación del personal integrante del equipo de trabajo siguiendo los procedimientos establecidos en la fase de inicio.

El adjudicatario no podrá modificar la composición del equipo de trabajo vinculado al presente contrato sin la aprobación previa y expresa del RC. Para formular la propuesta del cambio, el adjudicatario solicitará autorización explícita del RC acompañando justificación detallada del motivo que suscita el cambio y el currículum del candidato propuesto. En caso de que el RC no autorizase el cambio de personal propuesto por el adjudicatario, éste no se podrá llevar a cabo.

Por otra parte, el RC podrá solicitar el cambio de cualquiera de los técnicos asignados por otro técnico de igual categoría en caso de no cumplir los niveles de calidad exigidos o si existen razones justificadas que lo aconsejen.



### 3.9. Acuerdos de nivel de servicio

El servicio objeto de contrato deberá garantizar al menos el siguiente acuerdo de nivel de servicio en aquellas incidencias/solicitudes/tareas que entren dentro de su ámbito de competencias.

#### 3.9.1. Definición de Conceptos

Los técnicos deberán realizar su trabajo con corrección, educación y profesionalidad, cuidando en todo momento el trato al usuario, tanto verbal como escrito, así como la calidad de la información generada a través de protocolos, informes, seguimientos y soluciones, etc.

Cuando, a criterio del responsable del contrato, se detecte que los trabajos o información generada no cuenta con la calidad mínima esperada, se establecerá una no conformidad. Esta podrá ser mayor o menor dependiendo de la gravedad del error.

No conformidad menor: cuando se trate de un error o falta leve. Por ejemplo:

- Documentación insuficiente o no ajustada a las plantillas de documentación del proyecto.
- Demoras inferiores a un mes en el cumplimiento de los hitos parciales
- Responder o dirigirse a un usuario de forma inadecuada.
- Mala de calidad del código fuente de acuerdo a los parámetros establecidos por el RC.

No conformidad mayor: cuando se trate de un error relevante que pueda afectar al proyecto o al trabajo de otras personas, que afecte a la imagen del servicio o tenga una repercusión importante. Por ejemplo:

- Faltar el respeto a un usuario.
- Incumplimiento de RGPD y ENS.
- Negligencias en el cumplimiento de las tareas
- Falsedad en los informes o documentación entregada.
- Incumplimiento de las entregas trimestrales

Las no-conformidades menores tendrán un peso de 1 en el indicador. Las no-conformidades mayores tendrán un peso de 3. El indicador será la suma de los pesos de las no-conformidades menores y mayores.

#### 3.9.2. Penalizaciones

En el caso de que un indicador no supere la calidad de servicio mínima establecida (es decir, obtengamos una calidad de servicio insuficiente), se aplicarán las penalizaciones indicadas en la siguiente tabla.

| Indicador | Descripción    | Penalización<br>por calidad<br>de servicio | Penalización<br>por calidad<br>de servicio | Penalización<br>por calidad<br>de servicio |
|-----------|----------------|--------------------------------------------|--------------------------------------------|--------------------------------------------|
|           |                | MEJORABLE                                  | BAJA                                       | MUY BAJA                                   |
| NC        | No conformidad | 3- 5 puntos                                | 6-8 puntos                                 | 9-10 puntos                                |



La penalización mensual implicará un descuento de sobre el importe a facturar en el trimestre atendiendo a la calidad del servicio. Una penalización superior a 10 puntos puede implicar la cancelación del contrato por incumplimiento de la calidad mínima del servicio.

| Penalización<br>sobre factura            | Penalización<br>por calidad<br>de servicio | Penalización<br>por calidad<br>de servicio | Penalización<br>por calidad<br>de servicio |
|------------------------------------------|--------------------------------------------|--------------------------------------------|--------------------------------------------|
|                                          | MEJORABLE                                  | BAJA                                       | MUY BAJA                                   |
| Descuento sobre<br>importe de<br>factura | 10 %                                       | 20 %                                       | 30 %                                       |

### 3.9.3. Informes de servicio

Para la correcta supervisión de los servicios y del adecuado cumplimiento de los niveles de servicio, el adjudicatario a petición del RC deberá generar informes, tanto de detalle como de resumen, de acuerdo con las propuestas (modelos, contenido y sistema de generación y automatización). Entre los diversos modelos de informe, pueden citarse los siguientes:

- Indicadores de nivel de servicio.
- Resumen de rendimientos del personal de los equipos y disponibilidad.
- Incidencias y solicitudes en el Sistema: abiertas, cerradas, pendientes, etc.
- Estado de cumplimiento de los hitos del proyecto.
- Otros.

El adjudicatario deberá incluir en los informes éstos y otros indicadores que sean propuestos por el RC. El control y seguimiento del servicio se llevará a cabo sobre la base de los informes mensuales emitidos por el adjudicatario.

### 3.10. Obligaciones y responsabilidad

El adjudicatario está obligado a comunicar al RC cualquier actuación que pudiera afectar a la seguridad o integridad de los programas y/o ficheros de datos, siendo necesaria su autorización expresa para llevar a cabo dicha actuación.

El adjudicatario se compromete a informar al Responsable de Seguridad de la Universitat Rovira i Virgili, si detecta un uso fraudulento o irregular de cualquier equipo que pudiera afectar a la seguridad de los datos corporativos de la Universitat Rovira i Virgili.

El adjudicatario está obligado a informar al RC en caso de detectar alguna irregularidad o fallo en los equipos que pudiera afectar a la seguridad de los servicios corporativos de la Universitat Rovira i Virgili.

El adjudicatario guardará confidencialidad respecto a cualquier clave de acceso a Sistemas de Información de la Universitat Rovira i Virgili que conozca durante la prestación del servicio, así como sobre el contenido de aquella información que, no siendo pública o notoria, conozca por esta vía, especialmente la de carácter personal que tratará conforme a las instrucciones de Virgili que no podrá copiar o utilizar para fines distintos de los derivados del normal desempeño de las funciones ligadas al presente contrato, ni ceder a terceros, ni siquiera a efectos de conservación.

El adjudicatario se compromete a cumplir la normativa vigente en materia de protección de datos de carácter personal.

El adjudicatario será responsable de los daños o desperfectos causados en los equipos, y aplicaciones de la Universitat Rovira i Virgili a terceros en la medida de que éstos sean provocados por fallos o negligencias de su personal durante el cumplimiento del objeto del contrato.

El adjudicatario adoptará las medidas oportunas que dictan las buenas prácticas de gestión de residuos y será responsable de la correcta segregación y gestión de los residuos derivados de su actividad.

**(ANULADO)** ~~La propiedad intelectual de los trabajos, resultados, código y cualquier información generada como resultado del presente pliego, pertenecen a la Universitat Rovira i Virgili, no estando autorizado a su uso, venta o modificación sin la autorización o acuerdo expreso del RC o responsable legal.~~

El código entregado está sujeto a una garantía de 6 meses, durante la cual, si se detectan negligencias en la realización de los trabajos (por ejemplo, funciones vacías o que simulen el comportamiento especificado sin realizar su funcionalidad descrita), el adjudicatario deberá corregir todos los errores de la aplicación reportados en ese plazo y actualizar la documentación asociada.

## Lote 1: Orquestación de aplicaciones en el continuo nube-periferia

Presupuesto del lote: 530.000€, IVA excluido

### 1. Introducción

El crecimiento explosivo de la Internet de las cosas, de dispositivos periféricos, junto con el rápido crecimiento de datos generados en el borde de Internet está desafiando la computación en la nube tradicional. Para muchas aplicaciones, el alto rendimiento de la nube no compensa los costes elevados de sus servicios ni la mayor latencia de red causada por el transporte de datos desde la periferia de Internet. Para resolver este problema, recientemente ha surgido la llamada computación perimetral o «“Edge Computing” en inglés».

La computación perimetral es un nuevo paradigma que se aleja de las nubes centralizadas y lleva a cabo procesos computacionales lo más cerca posible de los dispositivos móviles, sensores y usuarios finales. La computación perimetral ha sido identificada como uno de los principales disruptores en el sector de la infraestructura de datos y la nube, y también como una tecnología estratégica para la transformación digital de Europa. Hoy en día, el 80% de los datos se procesan en la nube, un mercado actualmente dominado por empresas de fuera de la UE. Para el mercado de IoT inteligente, evidencias recientes demuestran que la demanda en el borde está creciendo rápidamente, y se espera que en 5 años el 80% del procesamiento de datos se llevará a cabo en el borde, revirtiendo completamente el escenario actual.

A pesar de la necesidad urgente de definir la infraestructura necesaria para brindar servicios informáticos en la periferia, y del crecimiento dramático en la inversión en investigación en los últimos años, la computación perimetral aún sigue siendo una tecnología emergente. Dentro de este contexto, aún no está claro, por ejemplo, qué abstracción de computación es la más adecuada para poder ejecutar procesos computacionales de forma integrada, desde el borde hasta la nube.

En este sentido, **el modelo de computación basado en funciones**, llamado «“Function as a Service (FaaS)” en inglés», podría proporcionar la abstracción tan necesaria para la computación perimetral. En el modelo FaaS, la unidad de cálculo es una función que se ejecuta de forma efímera y protegida por un entorno de ejecución seguro llamado «“sandbox” en inglés». El modelo FaaS tiene muchos beneficios en comparación con los modelos tradicionales basados en aplicaciones, como por ejemplo, una inicialización más rápida debido a los tamaños de imagen más pequeños y la reutilización de funciones. Además, las funciones pueden explotar mejor los recursos heterogéneos disponibles en el borde, proporcionar privacidad y confiabilidad adicionales y reducir el coste de los servicios en la nube.

Por otro lado, más allá de la abstracción de ejecución y desarrollo de aplicaciones, los hiperescaladores como Amazon Web Services, Google Cloud, etc., se están apresurando a posicionarse como proveedores de infraestructura en el borde, tratando de expandir su dominio actual en el mercado de la nube también al borde, y ofreciendo sus plataformas como una nueva estándar de facto.

Competir con los hiperescaladores requiere de una alternativa que permite **la orquestación federada de los recursos nube-periferia**, con tecnologías de código abierto y estándares abiertos. El nuevo sistema deberá ser compatible con todo tipo de máquinas virtuales, contenedores de aplicaciones, Kubernetes, y especialmente sobre tecnologías emergentes de virtualización ligeras como las basadas basadas en WebAssembly como WasmEdge (<https://wasmedge.org/>) o Krustlet (<https://krustlet.dev/>),

o las basadas en GraalVM (<https://www.graalvm.org/>) que se están usando para la construcción de entornos de ejecución seguros para FaaS en el borde.

## 1.1 Objetivos

El objetivo general de este lote es el desarrollo de una plataforma de código abierto para orquestar de forma holística los recursos del continuo nube-periferia con los siguientes requisitos:

- **Soporte para virtualización ligera:** La nueva plataforma debe soportar tecnologías de virtualización basadas en **GraalVM** y **WebAssembly** que permiten traer los paradigmas de aplicaciones nativas de la nube y las plataformas sin servidor (o FaaS) “al borde”. Como estas plataformas ofrecen un rendimiento mucho mayor y consume muchos menos recursos que los contenedores tipo Docker o las micro máquina virtuales (microVMs) como **FireCracker** (<https://firecracker-microvm.github.io/>), la plataforma deberá poder soportar la ejecución ligera y ultrarápida de funciones sobre tecnologías ligeras.

El objetivo es diseñar e investigar nuevas arquitecturas de computación ligera virtualizada que se adapten a entornos de computación Serverless. Basándonos en tecnologías de código abierto como OpenStack o OpenNebula, se optimizarán los tiempos de despliegue y arranque de máquinas virtuales ligeras para servicios de Funciones como Servicio, permitiendo alta concurrencia en grupos de tareas. La plataforma debe gestionar clouds privados elásticos a escala, los cuales pueden incrementar automáticamente la infraestructura disponible haciendo uso bajo demanda de recursos de proveedores públicos cloud y Edge.

- **Orquestación federada de aplicaciones nube-periferia:** Dado que muchos entornos de computación perimetrales se basan en un modelo de gestión distribuida, un modelo de gestión desagregado con controladores federados entre sí sería altamente deseable. La plataforma debería permitir a las empresas expandir las capacidades de sus nubes privadas hasta “el borde” mediante la combinación de sus propios recursos con recursos perimetrales multi-proveedor. Se espera que la nueva plataforma puede aprovechar herramientas de contenedores como Kubernetes, Docker y CRI-O (<https://github.com/cri-o/cri-o>) para administrar y ejecutar aplicaciones ligeras con GraalVM.

El orquestador debe permitir a las empresas ejecutar aplicaciones a lo largo del continuo nube-periferia. Para ello es necesario integrar los paneles de control y monitorización de servicios de computación paralela como LITHOPS con entornos de virtualización ligera nube-periferia de código abierto como OpenStack o OpenNebula. La orquestación eficiente debe tener una visión holística de los recursos computacionales y de los recursos de almacenamiento (bloques, objetos). Se diseñará comunicación entre capas (cross-layer) que permitan la optimización de flujos de trabajo en el continuo nube-periferia.

- **Integración con sistemas de computación de datos sin servidor como Lithops** (desarrollado en CloudLab, URV, <https://lithops-cloud.github.io/>): Lithops es un sistema de análisis de datos para plataformas FaaS de alto rendimiento escrito en Python. El nuevo sistema de orquestación debe permitir Lithops ejecutar aplicaciones altamente paralelizables y nativas de la nube, haciendo uso recursos de proveedores públicos tanto en la nube como en la periferia, así como recursos privados. Se optimizará e integrará LITHOPS para la ejecución de aplicaciones altamente paralelizables en entornos híbridos multi-proveedor.

Se deberá demostrar la integración del orquestador con Lithops. La integración de Lithops con el panel de control de ejecución de la plataforma de virtualización permitirá alta concurrencia entre grupos (bursts) de funciones ejecutadas sobre tecnologías de virtualización ligera. Se valorará con benchmarks el tiempo de arranque (cold start) y grado de concurrencia entre las funciones ejecutadas. También se permitirán mecanismos de “gang scheduling” que permitan agrupar funciones en nodos con varias CPUs. El orquestador deberá permitir ejecutar funciones en GraalVM como parte del flujo de trabajo de aplicaciones de análisis de datos escritas en Lithops. Se deberá demostrar la integración en al menos 5 flujos de trabajo definidos por el grupo CloudLab de la URV.

- **Soporte para ejecución altamente concurrente de grupos de tareas (bursts) en paralelo siguiendo el patrón lithops.map.** Se valorará la concurrencia obtenida en el lanzamiento de grupos de tareas de diversos tamaños sobre clusters federados utilizando virtualización ligera. Se implementarán también mecanismos que permitan agrupar funciones (FaaS) que puedan comunicarse de manera rápida entre ellas. Para ello se adaptará el Lithops Standalone runtime para permitir su mapeo a nodos grupales con diversas CPUs. También se ofrecerán mecanismos de monitorización avanzada entre la plataforma de virtualización ligera i el entorno de ejecución LITHOPS. También se prestará atención a los flujos de datos desde Lithops a servicios de almacenamiento de objetos y de bloques.

## 2. Tareas

Hay cuatro tareas a desarrollar en este lote. Cada tarea se corresponde con un objetivo:

**T1. Diseño y optimización de una plataforma de virtualización ligera nube-periferia** adaptada a cargas de trabajo Serverless. Se extenderá una plataforma de computación como OpenStack o OpenNebula para soportar de manera eficiente modelos de computación serverless bajo el paradigma de Funciones como servicio. Esta plataforma aprovechará la capacidad de OpenNebula/OpenStack de gestionar clouds privados elásticos a escala, los cuales pueden incrementar automáticamente la infraestructura disponible haciendo uso bajo demanda de recursos de proveedores públicos cloud y edge. Como parte de esta integración se estudiará la extensión y compartición de métricas desde OpenNebula a sistemas serverless como LITHOPS, y viceversa. También se diseñará y definirán configuraciones de la arquitectura de cluster en el Edge para la optimización de las cargas de trabajo.

Esta tarea tiene como objetivo la confección de un entorno de ejecución o “runtime” que permite la ejecución eficiente de aplicaciones compiladas a WebAssembly en “el borde”. La confección de este “runtime” debe permitir la ejecución de módulos de WebAssembly de forma eficiente y optimizado para la periferia. Esto es, debe permitir ejecutar funciones en multitud de dispositivos, desde servidores multinúcleo hasta sistemas de bajo coste como Raspberry PI. El nuevo “runtime” deberá ser construido sobre tecnologías de código abierto soportados por la Cloud Native Cloud Foundation (CNCf) como WasmEdge o Kruslet para facilitar su adopción, independientemente del grado de maduración tecnológico. Se valorará el soporte para compilación en tiempo de ejecución, también «conocida por sus siglas inglesas, JIT, “just-in-time”»).

Esta tarea se debe ocupar de disponer de los elementos de almacenamiento necesarios para poder guardar los módulos de WebAssembly en el continuo nube-periferia, por ejemplo, mediante el uso de un registro de código abierto como wasm-to-oci (<https://github.com/engineerd/wasm-to-oci>) y gestionar las dependencias y librerías compartidas.

El runtime deberá garantizar tiempos de inicio ultrabajos en frío que habilitan el lanzamiento de cientos o miles de funciones y microservicios en un breve período de tiempo. También, debe garantizar tiempos de ejecuciones cercanos a su ejecución nativa. Por ello, esta tarea incluye un evaluación exhaustiva del runtime mediante banco de pruebas estándares como PolyBench/C (<https://web.cse.ohio-state.edu/~pouchet.2/software/polybench/>) y aplicaciones reales compiladas a WebAssembly, por ejemplo, para realizar inferencias de modelos de aprendizaje automático «“Machine Learning (ML)” en inglés» con rendimiento casi nativo e incluso beneficiarse de aceleración de hardware con GPUs, TPUs, FPGAs, etc.

## **T2. Confección de un entorno de ejecución y orquestación ligero de funciones utilizando aislamiento a nivel de software (software isolation).**

La segunda tarea de este lote consiste en la creación de un orquestador federado nube-periferia para aplicaciones compiladas a WebAssembly. El principal requisito del orquestador es que permita la ejecución de aplicaciones compiladas a WebAssembly a lo largo del continuo nube-periferia. Los recursos disponibles en “el borde” son heterogéneos. A menudo, disponen de pocos recursos, por lo que es esencial que el orquestador despliegue y ejecute las aplicaciones implementadas con WebAssembly de forma eficiente, minimizando el consumo de memoria y/o ejecución. Por este motivo, el orquestador debe permitir ejecutar código compilado a WebAssembly de forma nativa, o bien sobre contenedores o microVMs, que presenten sobrecargas bajas con un mayor aislamiento. El orquestador también debe proveer de los medios de almacenamiento necesarios para ofrecer procesamiento de datos de forma local en los nodos perimetrales.

Como parte de esta actividad, se desarrollarán e integrarán métodos para la recopilación continua de datos generados por los dispositivos perimetrales y su procesamiento escalable en tiempo real. A partir de estos datos, se aplicarán técnicas y métodos de IA/ML para permitir que los entornos perimetrales actúen de manera proactiva para reducir el tiempo de despliegue, transferencias de datos y anticiparse a los problemas como falta de recursos.

Para evitar este tipo de problemas, anticipar cuántos recursos necesitará un contenedor o un módulo de WebAssembly para funcionar correctamente es vital. Muchos sistemas, p. ej., Kubernetes, permiten, si bien de forma opcional, garantizar una cantidad de recursos para un contenedor. Si hay más CPU o RAM disponible en el host, un contenedor puede usar más recursos que los especificados en las solicitudes. También, se puede limitar la cantidad máxima de recursos que puede usar un contenedor. Si un contenedor intenta asignar más de sus límites, el sistema lo reducirá o terminará.

El orquestador deberá poder anticipar problemas y migrar tareas de un dispositivo perimetral a otro en caso de que sea necesario. Para ello, se añadirán a cada dispositivo perimetral algún tipo de agente que permite supervisar la ejecución de las microservicios compilados a WebAssembly.

Se validará esta tarea con flujos de trabajo de computación paralela ofrecidos por el grupo CloudLab sobre la plataforma LITHOPS. Se validarán variables como tiempos de arranque de funciones, nivel de concurrencia conseguido, y escalabilidad de la arquitectura frente a cargas de trabajo dispares. Se hará especial hincapié en el soporte a grupos de tareas trabajando en ráfagas en paralelo con diferentes niveles de escalabilidad (grupos de diversos tamaños).

## **T3. Integración con la herramientas de análisis de datos Lithops.**



La tercera tarea de este lote consiste en la integración del “runtime” confeccionado a través del orquestador nube-periferia con Lithops. El objetivo es integrar el trabajo de las tareas **T1** y **T2** con un entorno de ejecución siguiendo el modelo “Function-as-a-Service” proporcionado por Lithops.

Esta integración debe permitir ejecutar flujos de trabajos híbridos dónde se mezclan tareas desarrolladas en Python con otras tareas escritas en otros lenguajes: C, C++, Rust, JavaScript, etc., y en algunos casos compiladas a WebAssembly “bytecode”. La ejecución de cada tarea paralela debe poder realizarse en cualquier localización del continuo nube-periferia. Se espera un soporte integral a la API `map-reduce` de Lithops ([https://lithops-cloud.github.io/docs/source/api\\_futures.html](https://lithops-cloud.github.io/docs/source/api_futures.html)).

Adicionalmente, se deberá diseñar monitores software, por ejemplo, con “sidecars” para capturar estadísticas de ejecución que permitan optimizar en el futuro con técnicas de AI los flujos de ejecución implementados en Lithops. Estas estadísticas incluirán el tiempo de arrancada o «“start-up time” en inglés» en frío y caliente, tiempo de ejecución y consumos de recursos (CPU, memoria y red).

Finalmente, se deberá demostrar la integración en al menos cinco flujos de trabajo definidos por el grupo CloudLab de la URV. Estos flujos se centrarán en procesamiento de datos genómicos y metabolómicos. El grupo CloudLab colaborará con la empresa adjudicataria para la correcta ejecución de estos flujos.

#### **T4. Soporte a grupos de tareas paralelas (burst) sobre los mecanismos de virtualización ligera.**

En relación a la tarea T3, se optimizará la ejecución de grupos de tareas concurrentes de diversos tamaños sobre los mecanismos de virtualización ligera. Se validarán diversos benchmarks que calculen el grado de concurrencia y tiempos de arranque de grupos de tareas de diversos tamaños. Se probarán tareas tanto computacionalmente intensivas como tareas intensivas en datos con flujos paralelos al Object Storage. Se validará también el soporte a tareas muy breves (<1s) en grupo con benchmarks que lancen muchos grupos de estas tareas sobre el cluster.

También se implementará soporte a grupos de tareas en el mismo nodo, permitiendo que un grupo de tareas (burst) se lancen en nodos colocados de manera que permitan comunicación interna rápida. La idea es que se lancen por ejemplo 1000 tareas en grupos de 100, y estas particiones caigan en un mismo nodo con diversas CPUs. El mapeo de particionado de datos o tareas a nodos permitirá eficiencia en la ejecución de pipelines. Aquí se requerirá una integración entre Lithops y los recursos computacionales o a nivel de orquestación de recursos en el panel de control.

### **3. Régimen de pagos**

Tras la certificación de cumplimiento del primer mes, se establecerá el pago del 25% del contrato. Trimestralmente, a partir de dicho mes, una vez recepcionado de conformidad el servicio prestado. Previa a la emisión de cada factura, se realizará una revisión del cumplimiento de los servicios prestados en el trimestre correspondiente por el responsable del contrato por parte de la Universitat Rovira i Virgili.



## Lote 2. Plataforma de computación confidencial Cloud/Edge en espacios de datos

Presupuesto del lote: 530.000€, IVA excluido

### 1. Introducción

El objetivo de este lote es el desarrollo de una plataforma de computación confidencial y segura para espacios de datos (International Data Spaces). Esta tarea requiere tanto de desarrollos y soluciones de ingeniería como de investigación activa en el campo para resolver retos aún abiertos. Por ello, el perfil de la empresa y del equipo deben incluir una fuerte componente investigadora con trayectoria demostrada en el campo de seguridad distribuida.

Los espacios de datos se están consolidando como arquitecturas integrales para diferentes dominios verticales (turismo, salud, transporte) que facilitan la compartición y acceso a datos entre proveedores de datos y consumidores de datos. La Unión Europea hace una apuesta decidida por la investigación y desarrollo de nuevas infraestructuras de datos que permitan liderar la digitalización en estos campos diversos.

El problema fundamental es la existencia de barreras importantes a la compartición de datos o el acceso distribuido por razones de seguridad y confidencialidad. Muchas empresas y proveedores de datos no confían sus datos a repositorios de datos (datahubs) públicos por desconfianza en la plataforma informática de almacenamiento y computación sobre los mismos. Esto afecta tanto a repositorios en la nube como a repositorios en el extremo de la red.

Por ello, este lote plantea el desarrollo e investigación de computación confidencial Cloud/Edge en espacios de datos. Los objetivos fundamentales son:

**O1. Investigación sobre la seguridad y la privacidad del ecosistema serverless.** Dado que la computación sin servidor ha surgido recientemente, ni su superficie de ataque ni sus mecanismos de seguridad han sido analizados en profundidad por la comunidad investigadora. Esta tarea tendrá como objetivo arrojar luz sobre la seguridad y privacidad del ecosistema serverless, en primer lugar identificando conceptos erróneos, debilidades y limitaciones en los ecosistemas serverless actuales que nos permitan identificar nuevos ataques, y luego proponiendo soluciones de seguridad que permitan lograr mejores garantías de seguridad y privacidad, teniendo en cuenta también los estrictos requisitos asociados con la computación sin servidor (por ejemplo, en términos de rendimiento o latencia).

**O2. Investigación sobre computación confidencial mediante entornos de ejecución confiables** (Trusted Execution Environments -TEEs) en el contexto de la computación serverless. La tarea comenzará analizando la seguridad de las diferentes soluciones de computación confidencial utilizadas actualmente por los principales proveedores de cloud computing, con el objetivo de conocer mejor los modelos de amenaza que asumen y sus posibles limitaciones. Se hará especial hincapié en la investigación de nuevos ataques de canal lateral que permitan a los adversarios deducir información sensible sobre la aplicación/función/servicio que se ejecuta dentro del TEE, y en la propuesta de contramedidas eficaces para reducir o eliminar posibles fugas de información.

**O3. Desarrollar una plataforma de orquestación eficiente y segura de recursos en el continuo cloud-edge** para mejorar el rendimiento y coste energético de las aplicaciones. Para ello, se deben implementar dos estrategias claves: i) optimizar la distribución de cargas de trabajo entre el edge (dispositivos de usuarios o oficinas centrales) y el cloud, de forma que podamos reducir la latencia; ii) utilizar el cloud para compartir y agregar modelos entre los distintos edge, de forma de que los edge tengan acceso a recursos para realizar tareas más complejas, todo esto de una forma segura por diseño y tomando la privacidad como configuración predeterminada, ya que no se compartirán datos, solo modelos.

**O4. Desarrollo de una plataforma de computación confidencial Cloud/Edge** en espacios de datos. Se implementarán los servicios de confidencialidad y encriptación de datos sobre un datahub existente (EMBL Metaspace) en el campo de la metabolómica y sobre un directorio de servicios de computación (Marketplace) denominado DATOMA desarrollado por la URV en AWS Cloud. La encriptación homomórfica y los TEEs deben proteger el ciclo de los datos en METASPACE y DATOMA en entornos Cloud pero también en servicios en el extremo. Para ello, se integrarán los servicios confidenciales en la plataforma Lithops (Cloud/Edge) del grupo CloudLab tanto en almacenamiento como en computación. Lithops es utilizado por METASPACE y DATOMA y permitirá la protección de todo el ciclo de vida de los datos.

## 2. Tareas

### T1. Investigación sobre computación segura en entornos Serverless.

Definición del panorama de amenazas y análisis del estado del arte en cuanto a ataques y soluciones de seguridad existentes para la computación serverless

Análisis de la seguridad y propuesta de ataques novedosos contra infraestructuras serverless. Esto incluirá el análisis de imágenes serverless en repositorios públicos (por ejemplo, Docker Hub), de diferentes soluciones de virtualización en las que se pueden ejecutar funciones serverless (no solo contenedores normales), de los servicios en la nube con los que interactúan y, por último, pero no menos importante, de la interacción entre todos estos componentes.

Diseño de soluciones de seguridad ligeras que mejoren significativamente la seguridad y la privacidad de las infraestructuras serverless en su conjunto. Se prestará especial atención a la propuesta de diversos tipos de técnicas y métodos de seguridad que permitan la definición, aplicación y verificación de las políticas especificadas por los proveedores de la nube y los propietarios de las aplicaciones.

Desarrollo de una metodología para proteger la ejecución de funciones “sin servidor” y de los flujos de ejecución con el objetivo de construir un entorno de ejecución transparente y seguro, pero también respetuoso con las restricciones de privacidad de las aplicaciones construidas sobre el mismo. Esta actividad realizará un estudio exhaustivo de la seguridad de las plataformas de ejecución “sin servidor”, de la superficie de ataque, y propondrá contramedidas efectivas para mitigar las posibles vulnerabilidades, bien mediante criptografía, bien por hardware mediante Entornos de Ejecución de Confianza ("Trusted Execution Environment" en inglés) como Arm Trustzone, o finalmente con políticas de seguridad dinámicas.

### T2. Diseño de una plataforma de computación confidencial Cloud/Edge mediante entornos de ejecución confiables.

Estudio y comparación de las tecnologías TEEs utilizadas actualmente por diferentes proveedores de nube para lograr la computación confidencial. El objetivo de esta tarea será comprender mejor las opciones de diseño de los TEE modernos y sus implicaciones en la seguridad y el rendimiento que ofrecen. Identificación de nuevos ataques de canal lateral y diseño de contramedidas que permitan reducir o eliminar por completo las fugas de información identificadas. Investigar posibles formas de utilizar TEEs (o tecnologías similares) para mejorar la transparencia y “accountability” de los proveedores de la nube en infraestructuras serverless.

Estudio de los requerimientos de seguridad y amenazas de todo el ciclo de vida de los datos en un espacio de datos. Se estudiarán las amenazas sobre el repositorio de datos y la computación de los datos en todo el ciclo incluyendo su publicación, almacenamiento, indexación, localización y acceso, transferencia segura, y finalmente su computación confiable en la nube o el extremo. Se utilizará como referencia el servicio METASPACE internacional y sus mecanismos públicos y privados de acceso a datos. Se diseñará un nuevo servicio confidencial con tecnologías TEE y de encriptación homomórfica que proteja todo el ciclo. Se garantizará que los propietarios del repositorio de datos en la nube nunca pueden acceder a datos o metadatos de clientes en el modo confidencial. Se propondrá una certificación de seguridad externa que valide la robustez del modelo siguiendo estándares internacionales.

### **T3. Desarrollo de una plataforma segura y eficiente de orquestación de recursos Cloud-Edge**

Estudio del estado del arte para comprender los desafíos de orquestar tareas/aplicaciones cuando hay dos capas (cloud y dispositivo de usuario/IoT) o tres capas (cloud, nodos edge, dispositivo final). En esta tarea, será importante seleccionar una o varias cargas de trabajo que se utilizarán para evaluar algoritmos de orquestación (por ejemplo, tareas que entrenan modelos de aprendizaje automático, tareas que utilizan (inferencia) modelos de aprendizaje automático en el edge, entre otras).

Evaluar los resultados en términos de rendimiento, latencia y consumo energético al utilizar el edge o el cloud. En este paso, se debe examinar el impacto de los aceleradores que comúnmente se utilizan en el edge (por ejemplo, Google Edge TPU, Intel Neural Compute Stick 2, etc.).

Analizar los resultados para diferentes estrategias de orquestación (por ejemplo, round-robin, usar siempre el edge si hay espacio libre, considerar el consumo energético, etc.). Se validará la plataforma de orquestación con varios flujos de trabajo serverless intensivos en datos de genómica y metabolómica. También se validará un flujo cloud-edge de datos geoespaciales ofrecido por el grupo Cloudlab.

### **T4. Desarrollo de una plataforma confidencial de espacios de datos sobre Lithops**

Desarrollo e integración en Lithops de un sistema de protección de datos que permita a los desarrolladores ejercer un control de grano fino sobre los flujos de información en sus aplicaciones. Parte de los flujos de trabajo de este nuevo tipo de aplicaciones se basan en la ejecución de funciones desarrolladas por terceros. Estas funciones se pueden comportar de formas inesperadas, o incluso de forma maliciosa. El objetivo fundamental será añadir a Lithops los mecanismos de seguridad necesarios para proteger los flujos de información y asegurar la ejecución segura de los distintos flujos de trabajo de acuerdo a unos requisitos de seguridad y privacidad predeterminados por el usuario. Idealmente, estos requisitos deberían expresarse como políticas de seguridad generadas de forma automática para que se pudiera dar cumplimiento mediante técnicas criptográficas en tiempo de ejecución.

Implementación y validación de la plataforma de computación confidencial sobre Lithops demostrando todo el ciclo de vida de los datos en las plataformas metabolómicas METASPACE y DATOMA basadas en Lithops. Participación de miembros de EMBL y URV (CloudLab) en la validación del ciclo de vida de la plataforma en un experimento real en el AWS Cloud.

Se implementará un prototipo usando Lithops con TEEs que permita la ejecución remota en repositorios en el extremo de flujos de datos confidenciales. Se garantizará con encriptación la protección de datos sensibles. El grupo CloudLab ofrecerá un flujo de datos en Lithops de genómica que será validado con las tecnologías confidenciales. Se valorará la transparencia y no intrusividad en el software existente para garantizar la confidencialidad, así como el rendimiento y costes adicionales añadidos por la capa de seguridad.

### 3. Régimen de pagos

Tras la certificación de cumplimiento del primer mes, se establecerá el pago del 25% del contrato. Trimestralmente, a partir de dicho mes, una vez recepcionado de conformidad el servicio prestado. Previa a la emisión de cada factura, se realizará una revisión del cumplimiento de los servicios prestados en el trimestre correspondiente por el responsable del contrato por parte de la Universitat Rovira i Virgili.

### Lote 3. Plataforma NOCODE de Manipulación de datos

Presupuesto del lote: 351.545€, IVA excluido

#### 1. Introducción

El objetivo de este lote es el desarrollo de una plataforma NOCODE de manipulación de datos no estructurados en la nube. La plataforma debe gestionar la gestión y acceso a los datos, la preparación y la transformación de los mismos, y la ejecución de flujos de trabajo (pipelines) de análisis de dichos datos.

Como datos no estructurados nos referimos a texto, imágenes y video sin metadatos de consulta disponibles en una base de datos. En particular, el proyecto comenzará enfocándose en datos no estructurados de genómica (SAM, BAM) y metabolómica (mzML).

Como sistemas de computación trataremos DATOMA.cloud (desarrollado por la URV en el grupo CloudLab), Lithops (desarrollado en ClouLab, <https://lithops-cloud.github.io/>) y NextFlow (<https://www.nextflow.io/>). Los datos se localizarán fundamentalmente en Cloud Object Storage como Amazon S3 y similares.

La plataforma NOCODE tiene dos componentes fundamentales: Servicio de visualización y composición de datos y Sistema de monitorización y control de recursos. El primero sería la interfaz gráfica NOCODE orientada a usuarios finales (con pocos conocimientos técnicos) que puedan visualmente conectar tareas y pipelines con fuentes de datos no estructuradas, y ejecutarlas sin necesidad de provisionamiento de recursos previo. El segundo componente está orientado a los administradores de la plataforma, una interfaz de control, monitorización y gestión de recursos que les permita optimizar y gestionar la ejecución eficiente de los pipelines creados por el primer componente.

Los objetivos fundamentales de este lote son los siguientes:

**O1. Creación de una interfaz web gráfica NOCODE que permita la composición visual de flujos de datos no estructurados.** La interfaz gráfica desarrollada con Javascript permitirá tener una paleta de componentes reutilizables y extensibles (conectores de datos, procesadores de datos) que podrán ser arrastrados a una paleta central y conectados entre si para definir un flujo de datos. Dicha interfaz se integrará con las herramientas Lithops, NextFlow y DATOMA gestionadas por el grupo CloudLab de la URV.

**O2. Creación de un sistema de monitorización y control de recursos de computación y almacenamiento en la nube.** Se desarrollarán mecanismos de monitorización de máquinas virtuales y contenedores que interceptarán los recursos computacionales utilizados en los flujos de datos de O1. El objetivo es gestionar de manera eficiente estos recursos, pudiendo optimizar el uso eficiente de los recursos y su coste económico.

**O3. Creación de un sistema de ejecución de flujos basado en la interfaz NOCODE y el sistema de monitorización.** Se deberá demostrar la integración con herramientas existentes de computación como Lithops, DATOMA y NextFlow en al menos 5 flujos de trabajo definidos por el grupo CloudLab de la URV.

**04. Creación de un panel de control de recursos económicos y de facturación en la nube conectado al sistema de monitorización.** El panel de control debe permitir un seguimiento exhaustivo de los costes económicos de ejecución de los flujos de datos en la nube. Deberá permitir configurar la ejecución en base a parámetros como la velocidad de computación, horarios de ejecución y límites temporales así como coste económico.

## 2. Tareas

### T1. Servicio de visualización y composición de datos

Esta tarea tiene como objetivo la implementación de una interfaz NOCODE de composición de flujos de datos no estructurados. La interfaz se desarrollará en Javascript para entornos Web aunque permitirá su ejecución en escritorio. La interfaz contendrá una paleta superior de componentes extensibles donde se instalarán los componentes de acceso a datos y los elementos de computación. Permitirá instalar nuevos componentes en la paleta que se corresponderán con ejecuciones en el Cloud o conectores de datos de Lithops.

La interfaz permitirá arrastrar los componentes a un marco central, donde se podrán enlazar de manera visual al estilo de herramientas RAD (Rapid application development) como Visual Studio o Azure Logic Apps. Se podrán etiquetar las conexiones y los componentes con propiedades externas en una consola (Property Inspector) con claves y valores. Se definirán dependencias con los tipos de datos y su volumen que permitan el posterior provisionamiento basado en datos (data-driven resource provisioning). Se utilizarán conectores de datos de código abierto generados por el grupo CloudLab (<https://github.com/aitorarjona/cloud-native-datasets>).

En colaboración con el grupo URV (CloudLab) se conectará esta interfaz visual a los sistemas Lithops, DATOMA y NextFlow. Finalmente, la interfaz generará un fichero con los metadatos del flujo similar a AWS ASF, Apache AirFlow y NextFlow. Este lenguaje simplificado de exportación se definirá en colaboración con el grupo CloudLab. Los flujos deberían ser poder controlados por herramientas de orquestación estándar como AWS ASF o Apache Airflow.

### T2. Sistema de monitorización y control de recursos

Esta actividad define la creación de un tablero (dashboard) de control web de los recursos computacionales usados por los flujos de datos de la tarea T1.

El dashboard permite el control y la monitorización avanzada de la ejecución de procesamiento de datos mostrando en tiempo real los trabajos desplegados, el estado de las distintas tareas, su tiempo de ejecución, errores y otra información sobre el sistema.

Se desarrollarán sensores de diferentes servicios Cloud y Edge que envíen eventos al sistema de monitorización. En particular, se diseñarán monitores software (sidecar) que deben acompañar las VMs o contenedores utilizados por Lithops, DATOMA y Nextflow y que compondrán la tarea anterior T1. En concreto, este monitor permitirá calcular y diferenciar entre tiempo de transferencia de datos y tiempo de computación, coste económico, así como registrar variables sobre recursos computacionales como uso de CPU y memoria durante el tiempo del experimento.

La información generada debe ser enviada a un servicio de eventos en la nube (como SQS o DELL Pravega). También se desarrollará un módulo de carga y análisis de eventos periódicos con funciones serverless que almacenen los mismos en una base de datos de eventos o series temporales.



Finalmente, el Dashboard utilizará herramientas existentes como Grafana para visualizar el uso de recursos durante un flujo de trabajo. Y se diseñará una interfaz web visual que permita seguir en tiempo real el estado de ejecución de un flujo de datos a partir de la información de la tarea 1. Esta herramienta permitirá mostrar en qué fase se encuentra el flujo, y visualizar la información capturada.

### **T3. Integración con herramientas de análisis de datos en la nube**

La tercera tarea de este lote consiste en la integración de la herramienta NOCODE con sistemas de análisis de datos en la nube como Lithops, DATOMA y NextFlow. El objetivo es integrar el trabajo de las tareas T1 y T2 con estos tres entornos de ejecución en la nube. La herramienta NOCODE y la de monitorización serán un complemento ideal al MarketPlace de flujos de datos ofrecidos por DATOMA y su integración en el DataHub Internacional de Metabolómica creado por Metaspaces.

La herramienta NOCODE debe ofrecer interfaces visuales amigables que permitan la conexión a datos en almacenamiento de objetos (Object Storage) tanto en la nube como en entornos en el extremo (Edge). Se crearán diversos componentes visuales en el editor NOCODE que permitan la conexión a repositorios estándar abiertos como METASPACE o el NCBA's SRA archive. También se integrará con conectores de datos a tipos nativos no estructurados ofrecidos por CloudLab en la librería dataplug (<https://github.com/aitorarjona/cloud-native-datasets>). Dicha librería soporta datos genómicos (SAM, BAM, gzip SAM, SRA), metabolómicos (mzML) y también datos geoespaciales (LIDAR, COPC).

En esta tarea también se ofrecerán prototipos sencillos de integración con Spark, Dask y Ray en clusters desplegados sobre Kubernetes. El grupo CloudLab ofrecerá aquí conectores de datos para estas plataformas, que serán integrados en la interfaz NOCODE.

Por último, se validará la tarea con cinco flujos de datos de genómica (variant calling 1), metabolómica (annotation pipeline, Hermes, rMSI) y datos geoespaciales (NDVI/water consumption). Se validará la conexión visual a fuentes de datos y su monitorización de recursos usando el Dashboard.

### **T4. Panel de control de recursos económicos (Cost plane)**

La tarea final de este lote es un panel de costes económicos para la ejecución de flujos de datos en la nube. El componente de control económico debe incluir un mecanismo de facturación (billing) de los recursos consumidos en la nube por los flujos de trabajo utilizando pagos por tarjeta de crédito.

Este componente se enfocará en la facturación de recursos en la nube de Amazon, y debe incluir soporte a On demand VMs pero también a recursos Spot. Soportará también servicios de computación como AWS Batch, AWS Lambda, o AWS EC2 VMs. El plano de control debe registrar y cuantificar todos los costes económicos tanto computacionales, como de almacenamiento y transferencias, como de orquestación de servicios y servicios añadidos de colas o gestión del sistema.

El objetivo de esta tarea es integrarse con las tareas anteriores para poder optimizar el uso de recursos de los flujos de trabajo sobre diferentes tipos de recursos en la nube. La idea es permitir cambios visuales en la plataforma NOCODE y de monitorización que faciliten el cambio a otros recursos que reduzcan costes.

Esta tarea ofrecerá también un servicio de monitorización (profiling) de flujos de datos con diferentes datos de diferentes volúmenes. La idea es poder entender los costes de un flujo para poder cambiar su modelo de ejecución sobre diferentes recursos computacionales en base a variables como deadlines de ejecución, precio máximo y topes de facturación.



Se demostrará la efectividad de la herramienta optimizando al menos dos flujos de datos (Lithops y Nextflow) sobre recursos computacionales en la nube con diversos volúmenes de datos. CloudLab colaborará aquí en la integración de este panel de coste con un optimizador de recursos Spot desarrollado en el Lote 4.

### 3. Régimen de pagos

Tras la certificación de cumplimiento del primer mes, se establecerá el pago del 25% del contrato. Trimestralmente, a partir de dicho mes, una vez recepcionado de conformidad el servicio prestado. Previa a la emisión de cada factura, se realizará una revisión del cumplimiento de los servicios prestados en el trimestre correspondiente por el responsable del contrato por parte de la Universitat Rovira i Virgili.

## Lote 4. Sistema de Optimización de costes en recursos computacionales en la nube

Presupuesto del lote: 117.000€, IVA excluido

### 1. Introducción

Uno de los aspectos esenciales de la computación en la nube es una gestión correcta de los costes económicos de los recursos utilizados. En particular si queremos ejecutar tareas computacionalmente intensivas que mueven grandes cantidades de datos, el problema de la optimización de costes es sin duda un problema clave.

Además, los proveedores de nubes públicas ofrecen una gran variedad de servicios con precios muy diferenciados. Por ejemplo, una vCPU puede costar 6 veces más barata en AWS Spot Instances que una vCPU en Amazon Lambda, o dos veces más barata que una vCPU en AWS EC2 (on demand).

El servicio de Spot Instances de Amazon sería el que ofrece precios más competitivos, aunque con ciertas restricciones de ejecución. En concreto, una máquina virtual (VM) Spot puede ser eliminada con un cierto aviso temporal, por lo que no es adecuada para un servicio continuado en producción. En cambio, dicho servicio puede ser muy adecuado para tareas que se pueden ejecutar en cualquier momento (batch analytics) o para ejecutar servicios de funciones serverless (FaaS o Function as a Service).

El problema de monitorizar y modelizar el comportamiento de los recursos Spot ha sido analizado en la literatura científica en los últimos años.. Además, han surgido iniciativas empresariales como spotinst.com (ahora spot.io adquirido por NetApp) que ofrecen optimización de costes y plataformas de computación gestionadas sobre estos recursos.

En general, la monitorización y optimización de recursos en la nube es un problema abierto, y diversas iniciativas como Crayon.com, CloudZero.com o SkyPilot (<https://github.com/skypilot-org/skypilot>) intentan optimizar el uso de recursos en la nube.

#### 1.1 Objetivos

Los objetivos clave de este trabajo son los siguientes:

**O1. Crear un sistema de monitorización de recursos computacionales (AWS Spot VMs)** que capture información de disponibilidad y la envíe a un servicio de eventos y series temporales para su posterior análisis

**O2. Entrenar modelos de aprendizaje automático que predigan la disponibilidad y coste de dichos recursos.** Ofrecer una API sencilla que permita calcular costes de diferentes tipos de recursos.

**O3. Desarrollar un plano de costes (Cost Plane) o mecanismo de optimización del provisionamiento de recursos en la nube** en base a diferentes cargas de trabajo y o flujos de trabajo de análisis de datos usando Lithops y sistemas de ejecución FaaS como OpenWhisk, OpenFaas o K8s.

**O4. Demostrar O3 con dos tipos de experimentos:** (i) Ejecución de plataformas FaaS sobre recursos Spot efímeros de manera eficiente y (ii) Ejecución de flujos de datos en Lithops sobre recursos efímeros y no efímeros.

## 2. Tareas

### T1. Sistema de monitorización

Desarrollar módulos software que puedan ser desplegados en diversos recursos computacionales (AWS Spot VMs) y que monitoricen la disponibilidad y coste de dichos recursos en diferentes regiones y centros de datos. Esta información se enviará a un servicio de eventos que recopilará los datos en una base de datos de series temporales. El sistema debe permitir la monitorización adaptativa de diferentes recursos computacionales.

El sistema debe también calcular los costes de transferencia (S3) entre diferentes regiones y tener un mapa de latencias que permita evaluar costes de movimientos de datos entre regiones.

Es necesario desarrollar también algoritmos que analicen esta información y la almacenen de manera estructurada obteniendo información de los distintos centros de datos.

Validación 1 (M3): Primer prototipo capturando información de al menos 10 regiones de AWS. Pruebas de consultas de datos.

Validación 2 (M6): Software de referencia capturando información de al menos 30 regiones de AWS. Pruebas de consultas de datos y demostración de escalabilidad y tolerancia a fallos del sistema.

### T2. Modelo de aprendizaje entrenado con información de monitorización

El objetivo de esta tarea es entrenar diversos modelos de aprendizaje automático, con aprendizaje profundo si es posible, para afinar la predicción de disponibilidad y costes para los recursos computacionales Spot en diferentes regiones.

Se utilizarán los datos recogidos en el sistema de monitorización (T1) para entrenar los diferentes modelos. Un objetivo clave es conseguir una alta precisión en la disponibilidad y coste de los recursos computacionales. Se ofrecerá un servicio o API abierta sobre Serverless functions que responda a consultas utilizando el modelo entrenado.

Validación 1 (M6): Comparación con literatura previa en Spot instances, y demostración de la precisión del modelo con diferentes recursos y regiones de la nube de Amazon. Se busca un error de menos del 5% en la disponibilidad de los recursos.

Validación 2 (M12): Software de referencia del modelo entrenado con gran precisión. Servicio sobre API Gateway y Amazon Lambda de una API abierta de consultas al modelo entrenado. Demostración de la robustez, precisión y escalabilidad de la solución.

### T3. Plano de costes en la nube

El objetivo de esta tarea es desarrollar un plano de control de costes que permita la optimización en el provisionado de recursos computacionales en la nube. El plano de control se centrará en tres entornos software:

1. Cluster K8s sobre VMs
2. Runtime FaaS como OpenWhisk, OpenFaas o Fn.
3. Lithops sobre K8s o runtime Faas.

En base a variables como el tiempo de computación, la flexibilidad de horarios, y techos de coste, el plano de control propondrá despliegues concretos en recursos Spot de diferentes Data centers. Se requiere un trabajo de investigación para realizar la optimización multi-criterio que permita adaptarse a los requerimientos de los clientes.

Se desarrollarán también módulos de control en al menos una plataforma FaaS, que permita tener en cuenta las notificaciones de eliminación de nodos, de forma que las funciones no se ejecuten en nodos que carecen de tiempo por su futura eliminación. Se podrán limitar los tiempos de ejecución de funciones para adaptarse a los tiempos de eliminación (eviction) de las funciones en el sistema.

Se calculará también en este plano los costes de transferencia de datos entre centros de datos, para evaluar la viabilidad de ejecuciones de Lithops cuando las transferencias entre regiones sean muy costosas. Se estudiará la integración con flujos de datos Lithops en al ejecución mediada por el plano de control.

Validación 1 (M12): Primer prototipo del panel de control con soporte a despliegue de cluster K8s sobre Spot VMs en base a variables proporcionadas como la flexibilidad horaria, recursos computacionales y localización geográfica.

Validación 2 (M18): Software de referencia del panel de control con soporte a despliegue de K8s, FaaS y Lithops. Se validará el despliegue de K8s, Faas y Lithops en recursos computacionales Spot especificando restricciones de movimientos de datos, ejecución, recursos computacionales y franjas horarias. Se validará con cargas de trabajo que la selección del panel de control es adecuada y obtiene buen rendimiento comparado con ejecuciones sobre AWS Lambda o EKS estándar. Se permitirá un 10% menos de rendimiento en el sistema comparado con recursos no Spot.

#### **T4. Validación con experimentos K8s, FaaS y Lithops sobre el plano de control**

Esta tarea se centra en la validación de toda la plataforma con experimentos reales sobre K8s, Faas y Lithops. El grupo de investigación CloudLab ofrecerá diversos flujos de datos en Lithops de análisis de datos geoespaciales y genómicos. Se deberán validar dichos flujos comparando su ejecución sobre entornos AWS Lambda y AWS EC2 VMs con los recursos AWS Spot VMs sugeridos por el plano de control (T3). Se persigue entre un 5% y un 10% de coste adicional máximo respecto al caso de referencia (baseline).

También se probarán ejecuciones auto-escalables de Ray, Dask y Spark sobre recursos proporcionados por el panel de control. Este punto incluye investigaciones abiertas enfocadas a combinar tecnologías cluster (Ray, Dask) con tecnologías serverless (Lithops) que se benefician del plano de control y su gestión de recursos. También evaluaremos a nivel científico despliegues multi-región que permitan optimizaciones del continuo de recursos entre los diferentes extremos.

Validación (M18): Ejecución de dos flujos de datos (Lithops) de manera optimizada sobre recursos Spot desplegados gracias al panel de costes. Validación de coste y tiempo de ejecución comparado con ejecución en recursos Lambda. Primeras pruebas de auto-escalado de servicios FaaS y K8s sobre Spot VMS gestionadas por el plano de control

Validación (M24): Validación final de diversos flujos de datos en Lithops, Dask , Ray y Spark sobre recursos habilitados por el panel de control. Ejecución sobre K8s y OpenWhisk (o plataforma FaaS equivalente como OpenFaas o Fn). Se permitirá un 5% de manera ideal o un máximo de 10% de penalización (overheads) en la ejecución sobre recursos Spot proporcionados por el panel de control. Se comparará también el coste y rendimiento de un cluster Spot provisionado por AWS Batch y otro suministrado por el panel de control.

### 3. Régimen de pagos

Tras la certificación de cumplimiento del primer mes, se establecerá el pago del 25% del contrato. Trimestralmente, a partir de dicho mes, una vez recepcionado de conformidad el servicio prestado. Previa a la emisión de cada factura, se realizará una revisión del cumplimiento de los servicios prestados en el trimestre correspondiente por el responsable del contrato por parte de la Universitat Rovira i Virgili.

Pedro A. Garcia  
Responsable Científico y responsable del contrato  
CU del *Departament d'Enginyeria Informàtica i Matemàtiques*

Tarragona, mayo de 2023